



**T.C.
İSTANBUL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**



Doktora Tezi

**FARKLI PLATFORMLARDA SUNULAN F2P OYUNLARINDAKİ
SATIN ALMA KARARLARININ MAKİNE ÖĞRENMESİ
YÖNTEMLERİYLE TAHMİN EDİLMESİ**

KAAN ARIK

Enformatik Anabilim Dalı

Enformatik Programı

**DANIŞMAN
Prof. Dr. Seda TOLUN TAYALI**

**II. DANIŞMAN
Doç. Dr. Murat GEZER**

Mart, 2024

İSTANBUL

Bu çalışma, 11.03.2024 tarihinde aşğıdaki jüri tarafından Enformatik Anabilim Dalı, Enformatik Programında Doktora tezi olarak kabul edilmiştir.

Tez Jürisi

Prof. Dr. Seda TOLUN TAYALI(Danışman)
İstanbul Üniversitesi
İşletme Fakültesi
İşletme Bölümü

Prof. Dr. Sevinç GÜLSEÇEN
İstanbul Üniversitesi
Enformatik Bölümü

Prof. Dr. Adem KARAHOCA
İstanbul Medeniyet Üniversitesi
Mühendislik Fakültesi

Prof. Dr. Çiğdem EROL
İstanbul Üniversitesi
Enformatik Bölümü

Prof. Dr. Gökhan Silahtaroğlu
İstanbul Medipol Üniversitesi
İşletme ve Yönetim Bilimleri Fakültesi

İntihal Programı Beyanı

20.04.2016 tarihli Resmi Gazete’de yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince; Bu Lisansüstü teze, İstanbul Üniversitesi’nin abonesi olduğu intihal yazılım programı kullanılarak Fen Bilimleri Enstitüsü’nün belirlemiş olduğu ölçütlere uygun rapor alınmıştır.

Proje Destekleri

--

Tezden Üretilmiş Yayınların Künye Bilgileri

Arik, K., Gezer, M. ve Tolun Tayalı, S. (2022). Bibliometric Analysis of Scientific Studies Published on Game Customer Churn Analysis Between 2008 and 2022 . *Journal of Politics Economy and Management* , 5 (1) , 55-75.

Arik K., Gezer M. ve Tolun Tayali S. (2022). The study of indicators affecting customer churn in MMORPG games with machine learning models. *Upravlennets / The Manager*, 13(6), pp. 70–85. DOI: 10.29141/2218-5003-2022-13-6-6.

ÖNSÖZ

Çalışma sürecimde her türlü yol gösterici olan, olumlu tavrıyla beni cesaretlendiren, bilgi birikimiyle çalışmama farklı açılardan bakmamı sağlayan beraber çalışmaktan ve her zaman öğrencisi olmaktan gurur duyduğum değerli danışman hocalarım Prof. Dr. Seda TOLUN TAYALI ve Doç. Dr. Murat GEZER'e sonsuz teşekkürlerimi sunarım.

Doktora eğitimi sürecim boyunca sağladığı desteklerden dolayı Prof. Dr. Sevinç GÜLSEÇEN'e ve İstanbul Üniversitesi Enformatik bölümü ailesine, tez sürecinde değerli tecrübelerini aktaran Prof. Dr. Adem KARAHOCA'ya ayrıca teşekkürü borç bilirim.

Kıymetli hocam Dr. Öğr. Üyesi Ali KIRKSEKİZ'e, kıymetli arkadaşım Öğr. Gör. Dr. Mehmet Tevfik AĞDAŞ'a, kıymetli araştırma görevlisi arkadaşlarım Münür İPEK ve Şafak Yüksek VURAL'a, ayrıca tez kapsamında kullandığım verilere erişme konusunda bana yardımcı olan Luiz Bernardo Martins KUMMER'e ve Ben C. LE'ye de ayrıca teşekkür ederim

Tez süreci boyunca her zaman destek olan ve sabırlı davranan canım eşime ve canım oğluma minnettarım.

Mart 2024

KAAN ARIK

İÇİNDEKİLER

Sayfa No

ÖNSÖZ	iv
İÇİNDEKİLER.....	v
ŞEKİL LİSTESİ	viii
TABLO LİSTESİ.....	xi
SİMGE VE KISALTMA LİSTESİ.....	xiii
ÖZET	xv
SUMMARY	xvii
1. GİRİŞ.....	1
2. GENEL KISIMLAR.....	5
2.1. ELEKTRONİK, VİDEO VE DİJİTAL OYUNLAR	7
2.2. DİJİTAL OYUN TARİHİ VE YILLARA GÖRE GELİŞİMİ.....	9
2.2.1. 1950 ile 2000 Yılları Arası Oyun Tarihi	10
2.3. DİJİTAL OYUN TÜRLERİ VE PLATFORMLAR.....	15
2.3.1. Kişisel Bilgisayarlar	16
2.3.2. Konsollar	16
2.3.3. Mobil	17
2.3.4. Sanal Gerçeklik	18
2.3.5. Web Tarayıcı	18
2.4. DİJİTAL OYUN TÜRLERİ.....	19
2.5. MÜŞTERİ VE MÜŞTERİ KAYBI.....	22
2.5.1. Gönüllü Müşteri Kaybı.....	24
2.5.2. Gönülsüz Müşteri Kaybı.....	25
2.5.3. Müşteri Koruma ve Sadakati	25
2.5.4. Müşteri Metrikleri	26
2.5.5. Müşterilerin Hareketsizliği ve Periyotları	28
2.6. OYUN SEKTÖRÜNDE ÖDEME TÜRLERİ VE MODELLERİ	30
2.7. OYUN SEKTÖRÜNDE İŞ MODELLERİ	34
2.7.1. Geçmiş Dönem Modelleri	34
2.7.2. Güncel Modeller	35
2.7.2.1. Mikroetkileşim (Microtransactions)	36

2.7.2.2. Abonelik	38
2.7.2.3. Freemium İş Modeli	38
2.8. MÜŞTERİ KAYIP ANALİZİ	42
2.9. VERİ PİRAMİDİ VE BİLGİ KEŞFİ	44
2.9.1. VERİ MADENCİLİĞİ VE UYGULAMA ALANLARI	47
2.9.2. VERİ MADENCİLİĞİ SÜRECİ	49
2.9.3. CRISP-DM METODOLOJİSİ	50
2.9.3.1. İşin Anlaşılması	51
2.9.3.2. Verinin Anlaşılması	52
2.9.3.3. Verinin Hazırlanması	52
2.9.3.4. Modelleme	56
2.9.3.5. Değerlendirme	58
2.9.3.6. Uygulama	65
2.10. LİTERATÜR TARAMASI	65
2.11. YAPAY ZEKA VE MAKİNE ÖĞRENMESİ	83
2.11.1. Makine Öğrenimi Sistemlerinin Türleri	84
2.11.1.1. Gözetimli Öğrenme	86
2.11.1.2. Gözetimsiz Öğrenme	87
2.11.1.3. Yarı Gözetimli Öğrenme	88
2.11.1.4. Pekiştirmeli Öğrenme	89
2.12. SINIFLANDIRMA VE REGRESYON	89
2.12.1. K-En Yakın Komşu Algoritması	94
2.12.2. Karar Ağaçları Algoritması	98
2.12.2.1. ID3 ve C4.5 Karar Ağacı Algoritmaları	101
2.12.2.2. Sınıflandırma ve Regresyon Ağaçları (CART)	102
2.12.3. Topluluk Öğrenmesi Yöntemleri (Ensemble Methods)	104
2.12.3.1. Torbalama (Bagging)	105
2.12.3.2. Rastgele Orman (Random Forest)	106
2.12.4. Yükseltme (Boosting)	109
2.12.5. Yığınlama (Stacking)	112
2.12.6. Destek Vektör Makineleri	113
2.12.6.1. Doğrusal Sınıflandırma	115
2.12.6.2. Doğrusal Modellerde Esnek ve Keskin Kenar Payı Sınıflandırma	118
2.12.6.3. Doğrusal Olmayan Sınıflandırma	121

2.12.7. Sade Bayes (Naive Bayes).....	122
2.12.8. Lojistik Regresyon Analizi	124
2.12.9. Yapay Sinir Ağları.....	129
3. MALZEME VE YÖNTEM.....	137
3.1. İŞİN ANLAŞILMASI.....	137
3.1.1. Araştırmanın Problemleri	137
3.1.2. Araştırmanın Sınırlılıkları	138
3.2. VERİNİN ANLAŞILMASI	138
3.3. VERİNİN HAZIRLANMASI.....	143
3.4. MODELLEME.....	147
3.5. MODEL PERFORMANS DEĞERLENDİRME	148
4. BULGULAR.....	150
4.1. K-EN YAKIN KOMŞU ALGORİTMASINA DAİR BULGULAR	150
4.2. KARAR AĞAÇLARI ALGORİTMASINA DAİR BULGULAR	154
4.3. RASTGELE ORMAN ALGORİTMASINA DAİR BULGULAR.....	156
4.4. AŞIRI GRADYAN ARTTIRMA ALGORİTMASINA DAİR BULGULAR.....	161
4.5. DESTEK VEKTÖR MAKİNELERİ ALGORİTMASINA DAİR BULGULAR.....	166
4.6. SADE BAYES ALGORİTMASINA DAİR BULGULAR	172
4.7. LOJİSTİK REGRESYON ALGORİTMASINA DAİR BULGULAR	175
4.8. ÇOK KATMANLI PERSEPTRON ALGORİTMASINA DAİR BULGULAR	180
5. TARTIŞMA VE SONUÇ	186
KAYNAKLAR.....	197
EKLER	214
EK-1	214
ÖZGEÇMİŞ	220

ŞEKİL LİSTESİ

Sayfa No

Şekil 2.1: İlk elektronik Tic Tac Toe oyunu olan XOX'e dair bir kesit (EDSAC, 1952).....	11
Şekil 2.2: "Spacewar" oyunundan bir kesit (MIT Museum Collections, 1961)	12
Şekil 2.3: Atari'nin Pong adlı oyunundan bir kesit (wikipedia.org, 2020b)	13
Şekil 2.4: Dijital oyun platformları ve alt dallar	16
Şekil 2.5: Oynanışa ve kamera açılarına göre dijital oyun türleri.....	20
Şekil 2.6: Kayıp türlerinin sınıflandırılması (Shaaban ve diğ., 2012)	24
Şekil 2.7: Oyuncu metriklerinin hiyerarşik diyagramı (Drachen ve diğ., 2013).....	26
Şekil 2.8: Bilgisayar ve mobil platformda baz alınan inaktiflik süreleri	29
Şekil 2.9: Ödeme türüne göre oyunların sınıflandırılması	32
Şekil 2.10: Overwatch oyununa ait bir şans kutusu mikro etkileşim örneği.....	37
Şekil 2.11: Freemium modeli yaşamsal döngü şeması (Reime, 2011)	39
Şekil 2.12: Büyük veri sınıflandırılması (Fayyad, 2005'ten uyarlanmıştır.)	46
Şekil 2.13: Veri madenciliği ile ilişkili çalışma alanları (Kelleher ve Tierney, 2022)	49
Şekil 2.14: CRISP-DM modeli yaşam döngüsü.....	51
Şekil 2.15: One-hot encoding örneği	54
Şekil 2.16: Temel Bileşenler Analizinde örnek boyut birleştirme	56
Şekil 2.17: Izgara ve rassal hiperparametre arama yöntemlerinin şeması	58
Şekil 2.18: ROC eğrisi ve altında kalan alanın grafiği.....	62
Şekil 2.19: Yıl bazında oyun kaybı tahmini üzerine yayınlanmış çalışma sayısı	79
Şekil 2.20: Araştırmalarda kullanılan veri setlerinin oyun isimlerine göre dağılımı	80
Şekil 2.21: Sınıflandırma probleminde kullanılan algoritmaların dağılımı	81
Şekil 2.22: Araştırmalarda kullanılan makine öğrenmesi türlerinin dağılımı	81
Şekil 2.23: Çalışmaların oyun platformuna göre frekans grafiği	82

Şekil 2.24: Veri madenciliğinin sınıflandırılması	85
Şekil 2.25: Makine öğrenmesi ve uygulamaları.....	86
Şekil 2.26: Regresyon sınıflandırması	90
Şekil 2.27: Doğrusal regresyon modeli	91
Şekil 2.28: Çoklu Regresyon canlandırması	92
Şekil 2.29: KNN algoritma yapısı	95
Şekil 2.30: Uzaklık ölçümlerinin şeması (Grootendorst, 2021’den uyarlanmıştır.)	96
Şekil 2.31: KNN regresyonunda eğri uydurması (Pedregosa, 2019’dan alınmıştır.).....	98
Şekil 2.32: Karar ağacı çalışma şeması.....	100
Şekil 2.33: Topluluk Öğrenmesi yöntemlerinin sınıflandırılması.....	105
Şekil 2.34: Torbalama algoritmasında öğrenenlerin birleştirilmesi	106
Şekil 2.35: Rastgele Orman Algoritması şematik yapısı	107
Şekil 2.36: Rastgele orman algoritmasının işlem basamakları	108
Şekil 2.37: Yükseltme yöntemi	110
Şekil 2.38: XGBoost algoritmasının çalışma prensibi	111
Şekil 2.39: Yığınlama yöntemi akış şeması	113
Şekil 2.40: Doğrusal ve doğrusal olmayan problemlerin şeması	114
Şekil 2.41: DVM algoritmasının terimleri ve şemalandırılması	114
Şekil 2.42: Lineer sınıflandırmada çizilebilecek çizgi örnekleri	116
Şekil 2.43: Destek Vektörleri ve hiperdüzlem çizimleri	117
Şekil 2.44: Doğrusal modellerde esnek ve keskin kenar payı çizgileri.....	118
Şekil 2.45: Lagranj çarpanlarının fonksiyon çizimi	120
Şekil 2.46: Sigmoid fonksiyon (Molnar, 2022’den uyarlanmıştır.)	126
Şekil 2.47: Gradyan İnişin canlandırması (Raschka, 2022’den uyarlanmıştır.)	128
Şekil 2.48: Biyolojik sinir ağının yapay sinir ağıyla karşılaştırmalı canlandırması	130
Şekil 2.49: Yapay sinir ağı çalışma prensibi	131
Şekil 2.50: Basit sinir ağır modeli ve katmanları	133

Şekil 2.51: Yapay sinir ağı sınıflandırması	134
Şekil 2.52: Tek ve çok katmanlı algılayıcıların canlandırması	135
Şekil 2.53: İleri ve Geri Beslemeli Yapay Sinir Ağlarının Canlandırması	135
Şekil 3.1: Özniteliklerin kayıt grupları.....	140
Şekil 3.2: Çok dosyalı ham verilerin tek bir .csv dosyasında işlenmesi süreci.....	143
Şekil 3.3: Hayatta kalma süresi değişkeninin kayıt niteliklerine dağıtımının şeması	145
Şekil 3.4: Oyuncu kayıp tanımlanmasında gözlem ve kayıp periyodları.....	145
Şekil 3.5: Etkileşim ve kullanıcı ham verilerinin işlenmesine dair akış diyagramı	146
Şekil 4.1: Blade and Soul K-En Yakın Komşu algoritmasına dair karmaşıklık matrisi	152
Şekil 4.2: Blade and Soul veri seti K-En Yakın Komşu dair ROC eğrileri	153
Şekil 4.3: Blade and Soul Karar Ağaçları algoritmasına dair karmaşıklık matrisi	155
Şekil 4.4: Blade and Soul veri seti Rastgele Orman karmaşıklık matrisi	157
Şekil 4.5: Blade and Soul veri seti Rastgele Orman ROC eğrileri.....	158
Şekil 4.6: Sınıflandırma probleminde Blade & Soul veri seti için Rastgele Orman algoritması öznitelik önemliliği sonuçları	159
Şekil 4.7: Blade and Soul veri seti Aşırı Gradyan Artırma ROC eğrileri	162
Şekil 4.8: Sınıflandırma probleminde Blade & Soul veri seti için Aşırı Gradyan Artırma algoritması Öznitelik Önemliliği Sonuçları.....	164
Şekil 4.9: Blade & Soul veri seti için DVM için C ve Gamma F-1 Skor sonuçları.....	170
Şekil 4.10: Blade and Soul veri seti Destek Vektör Makineleri ROC eğrileri.....	171
Şekil 4.11: Blade and Soul veri seti Destek Vektör Makineleri karmaşıklık matrisi	171
Şekil 4.12: Blade and Soul veri seti Destek Vektör Makineleri Kernel F1-skorları.....	172
Şekil 4.13: Blade and Soul veri seti Sade Bayes algoritmasına dair ROC eğrileri	174
Şekil 4.14: Blade and Soul veri seti Sade Bayes algoritmasına dair karmaşıklık matrisi	174
Şekil 4.15: Blade and Soul veri seti Lojistik Regresyon ROC eğrileri	176
Şekil 4.16: Blade and Soul veri seti Lojistik Regresyon karmaşıklık matrisi.....	177
Şekil 4.17: Blade & Soul Lojistik Regresyon öznitelik önemliliği sonuçları	178

TABLO LİSTESİ

Sayfa No

Tablo 2.1: Farklı sektörlerde freemium iş modelini kullanan uygulamalar	40
Tablo 2.2: Veri türleri ve örnekleri	52
Tablo 2.3: Sınıflandırma probleminde kullanılan performans değerlendirme ölçütleri	60
Tablo 2.4: Denetimli ve denetimsiz öğrenmenin farkları	87
Tablo 2.5: K-En Yakın Komşu Algoritmasında En Sık Kullanılan Uzaklık Formülleri.....	96
Tablo 2.6: CART algoritması hiper parametreler ve açıklamaları	104
Tablo 2.7: Doğrusal ve doğrusal olmayan DVM'lerin karşılaştırılması	115
Tablo 2.8: DVM'de kullanılan farklı çekirdek hileleri.....	122
Tablo 3.1: Veri toplama periyodu ve dönem bilgileri	139
Tablo 3.2: Veri seti için kayıp durumu sütun dağılımları.....	139
Tablo 3.3: Oyuncu kayıt hareketlerinin kategorilerine göre örnek listesi	140
Tablo 3.4: 9 farklı mini oyuna dair kodlar ve süreleri	141
Tablo 3.5: Veri setlerinde yer alan özneliliklerin adları ve açıklamaları	142
Tablo 4.1: K-En Yakın Komşu sınıflandırma probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri.....	151
Tablo 4.2: K-En Yakın Komşu regresyon probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri.....	153
Tablo 4.3: Karar Ağaçları algoritması sınıflandırma probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri.....	154
Tablo 4.4: Karar Ağaçları regresyon probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri.....	156
Tablo 4.5: Rastgele Orman sınıflandırma probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri.....	156
Tablo 4.6: Rastgele Orman regresyon probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri.....	158
Tablo 4.7: LengBear veri seti için Rastgele Orman algoritması performans sonuçları.....	160

Tablo 4.8: Aşırı Gradyan Artırma sınıflandırma probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri.....	161
Tablo 4.9: Aşırı Gradyan Artırma sınıflandırma probleminde Blade & Soul veri seti için ızgara ve rastgele arama sonuçları.....	162
Tablo 4.10: Aşırı Gradyan Artırma regresyon probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri.....	164
Tablo 4.11: Destek Vektör Makineleri sınıflandırma probleminde Blade & Soul veri seti farklı çekirdekler için performans değerlendirme ölçü değerleri.....	166
Tablo 4.12: Destek Vektör Makineleri sınıflandırma probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri	167
Tablo 4.13: Destek Vektör Makineleri regresyon probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri.....	168
Tablo 4.14: Destek Vektör Makineleri sınıflandırma probleminde Blade & Soul veri setine dair ızgara ve rastgele arama yöntemlerine göre performans değerlendirme ölçü değerleri	168
Tablo 4.15: Sade Bayes sınıflandırma probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri.....	172
Tablo 4.16: Lojistik Regresyon sınıflandırma probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri.....	175
Tablo 4.17: Lojistik Regresyon sınıflandırma probleminde LengBear veri seti için performans değerlendirme ölçü değerleri.....	179
Tablo 4.18: Çok Katmanlı Perseptron sınıflandırma probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri.....	180
Tablo 4.19: Çok Katmanlı Algılayıcı regresyon probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri.....	183
Tablo 4.20: Çok Katmanlı Perseptron sınıflandırma probleminde Blade & Soul veri setine dair arama yöntemlerine göre performans değerlendirme ölçü değerleri	183
Tablo 4.21: Çok Katmanlı Perseptron sınıflandırma probleminde LengBear veri seti için performans değerlendirme ölçü değerleri.....	184
Tablo 5.1: Test 1 veri seti için farklı makine öğrenimi algoritmalarının belirli metrikler üzerindeki performansları açısından bir karşılaştırması – anlamlılık testleri.....	187
Tablo 5.2: Test 2 veri seti için farklı makine öğrenimi algoritmalarının belirli metrikler üzerindeki performansları açısından bir karşılaştırması – anlamlılık testleri.....	187

SİMGE VE KISALTMA LİSTESİ

Simgeler	Açıklama
C	: Ceza parametresi
Γ	: DVM hiperparametresi
ϕ	: Haritalama Fonksiyonu
ε	: Epsilon
Π	: Toplam Sembolü
$\hat{y}$	: Tahmin edilen y değeri-regresyon
B	: Beta katsayısı
P	: Olasılık Değeri
$P(x)$	: Olasılık Fonksiyonu
W	: Ağırlık Vektörü
Σ	: Standart Sapma
γ	: Düzeltme Parametresi
E	: Aylak Değişken
λ	: Lambda Değişkeni

Kısaltmalar	Açıklama
CRISP-DM	: Cross Industry Standard Process for Data Mining
KNN	: k-En Yakın Komşu
KA	: Karar Ağaçları
XGB	: Aşırı Gradyan Artırma
RO	: Rastgele Orman
NB	: Sade Bayes
ÇKA	: Çok Katmanlı Algılayıcı
LR	: Lojistik Regresyon
DVM	: Destek Vektör Makineleri
YSA	: Yapay Sinir Ağı
MSE	: Hataların Karelerinin Ortalaması
RMSE	: Hataların Karelerinin Ortalaması Kökü
SST	: Kareler Toplamı

SSE	: Artıkların Karelerin Toplamı
SSR	: Tahminlerin Kareler Toplamı
Kernel	: Çekirdek
Hold-out	: Dışarıda Tutma Yöntemi
P2P	: Oynayabilmek İçin Öde
F2P	: Oynaması Ücretsiz
ROC	: Alıcı İşletim Karakteristiği
TBA	: Temel Bileşen Analizi
YZ	: Yapay Zeka
MOBA	: Çok Oyunculu Çevrimiçi Savaş Arenası
MMORPG	: Devasa Çok Oyunculu Çevrimiçi Rol Yapma Oyunu

ÖZET

DOKTORA TEZİ

FARKLI PLATFORMLARDA SUNULAN F2P OYUNLARINDAKİ SATIN ALMA KARARLARININ MAKİNE ÖĞRENMESİ YÖNTEMLERİYLE TAHMİN EDİLMESİ

Kaan ARIK

İstanbul Üniversitesi

Fen Bilimleri Enstitüsü

Enformatik Anabilim Dalı

Danışman : Prof. Dr. Seda TOLUN TAYALI

II. Danışman : Doç. Dr. Murat GEZER

Son yirmi yılda, oyun endüstrisi popülaritesi hızla artarak eğlence sektöründe en üst sıralarda yer almayı başarmıştır. Bu yükselişle birlikte, müşteri ilişkileri ve müşteri kaybı analizi, oyun sektöründe daha da ön plana çıkmıştır. Oyunlar sektöre hızla entegre olmakta ve yüksek gelir getiren elektronik ürünler olarak nitelendirilmektedir. Oyunlara olan ilginin artışıyla birlikte şirketlerin gelirleri artış göstermiş, bu durum ise müşteri kaybı kavramını daha kritik hale getirmiştir. Bu bağlamda, sektördeki müşterinin davranışları analiz edilmeye değer bir yapı olduğunu ortaya koymuştur. Bu tez, Blade and Soul ve LengBear oyunlarında belirli dönemlerdeki oyuncu kayıt verilerinin makine öğrenmesi algoritmaları ile müşteri kayıp analizini gerçekleştirmektedir. Bu bağlamda problem sırasıyla, oyuncu kaybı sınıflandırması ve oyuncunun hayatta kalma süresinin tahmin edilmesi olarak modellenmiştir. Çalışmanın teorik temelini davranışsal ekonomi ve kayıp yönetimi başlıkları oluşturmaktadır. Araştırmada, problemlerin çözüm önerisinde makine öğrenimi yöntemleri kullanılmıştır. Araştırma kapsamında her iki veri setinde CRISP-DM metodolojisi uygulanmış ve bu metodolojiye bağlı kalarak süreç detaylı bir şekilde açıklanmıştır. Blade and Soul oyununda, oyuncu kaybı ve hayatta kalma süreleri, XGBoost sınıflandırma problemi için en yüksek tahmin performansını ortaya koyarken, ÇKA ve XGBoost regresyon ise hayatta kalma sürelerinin tahmininde başarılı performans göstermiştir. LengBear veri setinde ise Çok Katmanlı Algılayıcı (ÇKA) ve Rastgele Orman (RO) algoritmalarının ızgara arama yöntemi ile başarılı sonuçlar elde etmiştir. Sonuçlar, müşteri kaybı analizinin işletmelerin müşteri davranışlarındaki eğilimleri ve kalıpları

belirlemelerine yardımcı olabileceğini göstermiştir. Bununla birlikte, müşterilerin oyunu bırakma olasılığının (churn) ne zaman en çok olduğu, hangi özelliklerin ayrılmasına neden olduğu ve müşteri eğilimleri ve öngörüler ortaya koyulmaya çalışılmıştır.

Mart 2024, 221+xviii. sayfa.

Anahtar kelimeler: Müşteri Kaybı Tahmini, İş Analitiği, Dijital Oyunlar, Makine Öğrenmesi, Yapay Zeka



SUMMARY

Ph.D. THESIS

PREDICTION OF PURCHASING DECISIONS IN F2P GAMES OFFERED ON DIFFERENT PLATFORMS WITH MACHINE LEARNING METHODS

Kaan ARIK

İstanbul University

Institute of Graduate Studies in Sciences

Department of Informatics

Supervisor : Prof. Dr. Seda TOLUN TAYALI

Co-Supervisor : Assoc. Prof. Dr. Murat GEZER

In the last two decades, the gaming industry has grown rapidly in popularity and has become one of the top entertainment industries. With this rise, customer relationships and churn analysis have become more prominent in the gaming industry. Games are rapidly being integrated into the industry and are characterized as high revenue generating electronic products. With the increase in interest in games, companies' revenues have increased, making the concept of customer churn more critical. In this context, the behavior of the customer in the industry has revealed a structure worth analyzing. This thesis analyzes customer churn in Blade and Soul and LengBear games by using machine learning algorithms on player registration data over certain periods. In this context, the problem is modeled as player churn classification and prediction of player survival time, respectively. The theoretical basis of the study is behavioral economics and churn management. In the research, machine learning methods were used to propose solutions to the problems. Within the scope of the research, CRISP-DM methodology is applied to both data sets and the process is explained in detail based on this methodology. In Blade and Soul, player loss and survival times showed the highest prediction performance for the XGBoost classification problem, while MLP and XGBoost regression performed well in predicting survival times. On the LengBear dataset, the grid search method of the Multilayer Perceptron (MLP) and Random Forest (RF) algorithms achieved successful results. The results

showed that churn analysis can help businesses identify trends and patterns in customer behavior. In addition, when customers are most likely to churn, what characteristics cause them to leave, and customer trends and insights are revealed.

March 2024, 221+xviii. pages.

Keywords: Customer Churn Prediction, Business Analytics, Digital Games, Machine Learning, Artificial Intelligence



1. GİRİŞ

Bilgisayar, tablet ve akıllı telefon gibi teknolojik ürünlerin günlük hayatımıza girişiyle birlikte ürün, satış ve pazarlama gibi konular işletme sektörü için önemli unsurlardan biri haline gelmiştir. Özellikle müşteri ihtiyaçları ve tercihleri daha da ön plana çıkmaya başlamıştır. Müşteri, işletmelerin marka ürününü, idari ve kişisel hedefler doğrultusunda satın alan kişi veya kuruluştur. Bir işletmenin ekonomik kaynak bütünlüğünün temel yapı taşlarından en önemlisidir. Pazarlama ve ürün yönetimi konusunda karşımıza çıkan bu kavram müşteri ilişkileri yönetimiyle birlikte daha da önem kazanmıştır. İnternet'in toplumun büyük bir kesimi tarafından erişilebilir hale gelmesiyle birlikte müşterilerin tercihleri ve fikirleri işletmeler için kilit konumdadır (Selvi, 2014). Çünkü müşteriler e-ticaret sitelerinde bir ürünün farklı modellerine ilgi göstermekte, aynı zamanda daha uygun fiyatlara bu ürünlere erişebilmektedirler. Bu durum, ürünlerin sayısının artması ve rekabet ortamı seviyesinin yükselme sebebi olmuştur (Koçoğlu, 2017).

Müşteri tercihlerinin önem kazanmasıyla birlikte şirketler müşteriye elde tutabilmek için farklı analizlere ve yöntemlere başvurmuşlardır. Satılan ürünler için müşterilerden geri dönüt alabilmek için uygulamaları plan dahiline alarak hizmet kalitesini ölçmeye çalışmışlardır. Bu çalışma günümüzde kaçınılmaz olmuştur. Çünkü, rekabet faktörü şirketlerin satışlarını ve müşteri oranlarını korumada en önemli unsurdur. Pazar değer kazandıkça, şirketler müşteri sadakatini, memnuniyetini ve elde tutmayı iyileştirmenin yollarını belirlemeye yönelmiştir. Pek çok kuruluştaki pazarlama yaklaşımı, ürün odaklılıktan müşteri odaklılığa doğru aşamalı olarak gelişmiştir. Bu yaklaşım şirketlerin müşterilerinin kim olduğu, ne satın aldıkları, ne zaman satın aldıkları ve davranış tahminleri hakkında bilgi edinmesini sağlayan modern veritabanı teknolojileri tarafından desteklenmektedir (Manjunatha ve Shivalingaiah, 2004).

Müşteri İlişkileri Yönetimi (Customer Relationship Management) çerçevesinde müşteri kaybını önleme işletmeler için üst düzey hedefler arasında yer almaktadır. Büyük şirketler, şirket adına kayıp olarak nitelendirilebilecek kayıpları tespit edebilmek için kayıp tahmin modellerini uygulamaktadır. Bu kayıpları tahmin edebilmek için, günümüzde popüler hale gelen veri madenciliği ve yapay zeka tekniklerini de sıklıkla kullanmaktadırlar (Burez ve Poel, 2009). Makine öğrenimindeki gelişmelerle birlikte, İş Zekası (Business Intelligence) uygulamaları,

makine öğrenimi algoritmaları tarafından desteklenen gelişmiş veri madenciliği tekniklerinden yararlanarak daha iyi müşteri içgörülerini sağlayabilmektedir. Özellikle gelişmekte olan şirketler için müşteri kaybını belirleme, üzerinde çok fazla durulmayan bir konu olsa da, görünenden daha fazla önem arz ettiği söylenebilir.

Müşteri kaybına yönelik analizlerin son yıllarda banka, telekomünikasyon, müzik uygulamaları, video platformları, dijital medya ve oyun sektöründe sıklıkla kullanıldığı görülmektedir. Özellikle oyun sektörü 2000'li yılların başından bu yana hızlı gelişim göstermesinden dolayı eğlence endüstrisinde üst sıralarda yer bulmuştur. Bu yükselişle birlikte dijital oyunların da sektöre hızlı şekilde entegre olması, yüksek gelir getirmesi oyun sektöründe müşteri ilişkileri ve müşteri kayıp analizinin önemini daha da arttırmıştır. Müşterilere aralarından seçim yapabilecekleri çok çeşitli hizmet sağlayıcıları sunulmuş ve yeni müşteriler kazanmayı zor ve maliyetli hale getiren bir ortam ortaya çıkmıştır (Poel ve Larivière, 2004). Bu doğrultuda Müşteri İlişkileri Yönetimi kavramı uzun vadeli önemli konulardan biri haline gelmiştir.

Müşteri İlişkileri Yönetimi kapsamında müşterinin şirkete katılımını ve bağlılığını arttırmak en zorlu görevlerden biridir. Bu sebeple, yeni müşteri kazanmanın yüksek maliyeti nedeniyle, yerleşik işletmeler müşteri kazanmaya çalışmak yerine müşteriyi elde tutmaya odaklanmaktadır. Müşteriyi elde tutmada müşteri kaybı riskini tahmin etmek önemli bir görevdir. Müşteri kaybı tahmin analizinde elde edilen doğruluğun her bir yüzdelik artışı, şirketler için önemli potansiyel gelir tasarrufu sağlayabilir. Diğer bir ifadeyle, müşteriyi elde tutmada küçük bir gelişme, kârda önemli bir artış getirebilir (Rudin, 2019). Bu, özellikle her müşterinin kayda değer miktarda kâr katkıda bulunabileceği, müşteri katılımının ve sadakatinin nispeten düşük olduğu finansal hizmetler sektörü için geçerlidir. Finansal hizmetler alanında, verimli ve etkili elde tutma stratejileri oluşturmak için veriye dayalı bir müşteri kaybı tahmin modeli geliştirmek esastır (Vo ve diğ., 2021).

Dijital oyunlar, diğer dijital ürünlere göre kendine özgü bazı niteliklere sahiptir. Yani oyuncu perspektifinden bakıldığında, her oyuncu oyunu farklı hızlarda oynayabilir ve oyun deneyimi her oyuncu için birbirinden farklı olabilir. Bu, oyuncuların oyunu nasıl oynadıklarına ve aktif katılım düzeyine göre değişiklik gösterir. Ayrıca dijital oyunlar, beceri ağaçları (skill trees), yan görevler (sub-quests), kullanıcı sıralamaları (player rank), ödüller (rewards) ve bölüm tasarımı (level design) gibi değişkenler bakımından diğer sektör ürünleriyle kıyaslandığında

nitelikler bakımından farklılaşır. Yukarıda belirtilen unsurlara ek olarak, oyunların en önemli bileşeni, ilk oynayıştan sonra azalmayan oyuncu deneyimidir. Bu nedenle oyuncular belirli süre periyotlarında doyuma ulaşana kadar ürünleri tüketmeye devam edebilirler (Grüsser ve diğ., 2007; Mutanen ve diğ., 2010; Hadiji ve diğ., 2014).

Bu tez çalışmasında, makine öğrenmesi yöntemlerinin kullanılarak kayıp eğilimi gösteren müşterilerin tahmin edilmesi, bu doğrultuda farklı makine algoritmalarının performanslarının karşılaştırılması ve bu yöntemlerin kullanılabilirliğinin oyun şirketleri özelinde müşteri kaybını azaltmaya yönelik değişkenlerin incelenmesini amaçlamaktadır. Tez kapsamında oyun sektöründe yer alan MMORPG (Massively Multiplayer Online Role-Playing Game) ve mobil türünde şans oyunu olmak üzere iki farklı veri seti üzerinde çalışılmış ve bu veri setleri araştırmacılar için özel izin alınarak tez kapsamında kullanılmıştır. Oyunlarda müşteri kayıp analizinde sınıflandırma problemine dair ülkemizde lisansüstü düzeyde bir çalışma olmadığı ve bu konuda ilk olma özelliği taşımaktadır. Tez çalışmasının sonuçları, öncelikli olarak İş Analitiği ve Müşteri Kayıp Analizi alanında yapılacak çalışmaları iyileştirilmesinde katkı sağlayacağı ve oyun alanında öncü bir çalışma olacağı göstermektedir.

Tez çalışmasının GENEL KISIMLAR başlığı altında oyunun tanımı, kronolojik gelişimi ve günümüzde nasıl kategorize edildiğine dair bilgiler verilmiştir. Ardından veri, veri piramidi, veri madenciliği, yapay zeka tarihçesi, bu yapıların kullanım alanları ve makine öğrenmesinde tez kapsamında kullanılan algoritmaların teorik içeriklerinden bahsedilmiştir. Sonrasında müşteri, müşteri kaybı, oyun sektöründe müşteri kayıp analizi ve geçmişten günümüze kadar oyun sektöründe kullanılan pazarlama modelleri üzerinde durulmuştur. Tezin devamında uygulanan farklı makine öğrenmesi algoritmalarına ve izlenen adımlara detaylı olarak yer verilmiş, ayrıca çözülmesi hedeflenen problemde oyun şirketleri için işlenebilecek göstergelere dair analizler gerçekleştirilmiştir. Bölümün sonunda oyunda müşteri kayıp analizine dair alanyazın taraması yapılmış ve yapılan çalışmaların hangi yönleriyle ön plana ortaya çıktığı ve hangi konularda yetersiz olduğuna dair bir tarama gerçekleştirilmiştir.

MALZEME VE YÖNTEM ana başlığında CRISP-DM (Cross-Industry Standard Process for Data Mining) metodolojisine bağlı kalınarak çalışma kapsamında kullanılan veriler ayrıntılı şekilde şema haline getirilmiş, işi anlama ve veriyi anlama basamağında istatistiksel analizlere yer verilerek ön işleme basamağında yapılan işlemler akış şemasıyla açıklanmıştır. Ayrıca tez

kapsamında MMORPG verisi üzerinde kullanılan ve araştırmacı tarafından oluşturulan bir değerlendirme puanı eklenmiş ve buna puana göre sonuçlar oluşturulmuştur.

BULGULAR kısmında ise uygulanan algoritmalara dair farklı parametreler denenerek elde edilen sonuçlar tablolastırılmış ve farklı performans ölçütlerine göre tüm sonuçların karşılaştırmalı analiz sonuçları verilmiştir. Tez kapsamında iki farklı veri seti kullanıldığından ötürü hem MMORPG hem de mobil veri setleri üzerinde pazarlama modeline göre karşılaştırmalı analizler ve şirketler için değerlendirilebilecek göstergelerin de açıklamaları yer almaktadır.

TARTIŞMA VE SONUÇ bölümünde müşteri kayıp analizinde geliştirilen yapay zeka modellerinin sadece analiz kısmında başarılı olmasına odaklanılmamış aynı zamanda şirketlerin kayıp önlemelerine karşı alınabilecek aksiyonların göstergeleri de değerlendirilmiştir ve tartışılmıştır.

2. GENEL KISIMLAR

Oyun, insanlık tarihinin ilk dönemlerinden bu yana günlük yaşamın ve kültürün bir parçası olmayı başarmıştır. Eski dönem toplumlarında farklı oyun örneklerine rastlamak mümkündür. Mısır'da satranç, tavla ve senet, Aztek İmparatorluğu'nda hem halkın hem de soyluların oynayabildiği, şans ve strateji becerileri gerektiren "Patolli" oyunu bu dönem oyunlarına ait örneklerden bazılarıdır. Türk kültüründe ise Dedem Korkut Hikayelerinde yer alan oy, av, hayvan güreşi ve ok atmaca gibi oyunlara rastlamak mümkündür (Ant, 2003). Bunlar, geleneksel oyunlar şeklinde başlamış ve zaman içinde değişime uğrayarak kimliğini yitirmeden günümüze kadar gelmeyi başarmışlardır.

Oyun kavramı felsefe, tarih ve sosyoloji gibi alanlarda çalışan bilim insanlarının farklı tanımlamalarıyla alanyazında yerini almıştır. Aynı zamanda oyunların sahip olduğu içerikler ve birbirleriyle nasıl ilişki kurabileceklerini anlamak ve açıklamak için de oldukça fazla çalışma gerçekleştirilmiştir. Sürekli bir değişim içinde süreç veya yapıyla ilgili kesin bir tanım oluşturmak zordur. Çünkü oyun, yapısı itibarıyla farklı dönemlerde farklı kitlelere hitap eden bir etkinliktir (Ang ve Zaphiris, 2008; Avedon ve Sutton-Smith, 2015; Costikyan, 2002; Debus ve diğ., 2020; Juul, 2001; Maroney, 2001; Myers, 1998; Room, 2010; Suits, 1978; Tavinor, 2008; Whitton, 2014; Zimmerman, 2003). Buradaki amaç, önceki tanımları dikkate alarak, tanımların ortak unsurlarını bulmak, sorunları ayırt etmek ve ardından oyunun temel tanımını ortaya koymaktır.

Psikoloji, fizyoloji, felsefe, tarih, sosyoloji ve kültür araştırmaları alanlarından farklı bilim insanları yalın oyun kavramının değişimini ve yapısını tanımlamaya çalışmışlardır. Oyun kuramına ait çalışmalara katkı sağlayan önemli isimlerden filozof ve tarih profesörü olan Johan Huizinga, insan toplumunun eski çağlardan beri oyuna aşina olduğunu ve kapsamlı ilkel faaliyetleri temelinde barındırdığını belirtmektedir (Huizinga, 1938). Ayrıca, oyunun birtakım unsurlar barındırdığını, bütünsellik ve gönüllülüğe dayalı olduğunu ve belli kurallar çerçevesinde gerçekleşmesi gerekliliğini de vurgulamaktadır. Bu doğrultuda oyun, belli bir mekanda gerçekleşen, zaman ve istenç sınırları içinde, bir düzene uygun, bireyin kendi rızası doğrultusunda ve güncel hayatın gerçekliğinden uzak gerçekleşen kültürden eski bir olgu olup, toplum ve kültür faaliyetlerine yön veren bir faaliyettir (Huizinga, 1938; Crawford, 2003; Maroney, 2001).

Oyuna dair ülkemizde yapılan araştırmalarda ise sanat tarihi alanında araştırmalar yapmış bilim insanı Metin Ant öne çıkmaktadır. Oyun ve Büğü adlı kitabında oyun ve yapısına ait tanımlamaların yanında Türk Kültürü'nde oyunun ne denli önemli bir yer tuttuğunu da ortaya koymaktadır (Ant, 2003). Huizinga'nın çalışmasında oyunun kültürden eski olduğunu ve kültürlerin oyunlardan etkilendiği fikrine katılarak, Türkler'e ait oyun kavramına yaptığı araştırmalarda, oyun kelimesinin eski Türk toplumlarında dini ayin ve törenleri yapan şamanların insanlarla ruhlar arasında aracılık yaptığı dini törenlere dikkat çekmektedir.

1960'lı yılların önemli sosyologlarından biri olan Caillois'a (1961) göre oyun serbestlik, kurallara dayalı olma, bir sonuca ulaştırma ve gönüllülük ilkelerinin ön plana çıktığı faaliyet bütünüdür. Aynı zamanda Suits (1967) oyunun teknik bir aktivite olmaktan ziyade belirli unsurlar bakımından farklılık kazandığını dile getirmiştir. Bu açıdan oyun, belirli kuralların izin verdiği çerçevede yapısına göre araçlar seçilerek, belirli bir durumla sonuçlanan ve bunu ortaya çıkarmaya yönelik faaliyet olarak nitelendirmiştir. Abt (1970) ise oyunu iki veya daha fazla bağımsız birey arasında karar verme mekanizmalarını gözetererek, belirlenen hedeflere ulaşılan ve içeriğin sınırlı olduğu bir aktivite olarak tanımlamaktadır. Felsefi bakış açısıyla oyun, hedefleri ve kuralları olan, gönüllülüğün esas olduğu ve ciddiyet olgusundan uzak bir aktivitedir (Suits, 1978). Kart oyunları ve masa oyunları üzerinde kapsamlı çalışmalar gerçekleştiren David Parlett (1982) oyunu, kazanma odaklı bir amacı olan ve belirlenmiş birtakım kurallara göre oynanan faaliyet olarak tanımlamaktadır. Adams'e (2014) göre ise oyun, katılımcının kurallara uygun davranarak keyfi olarak, önemsiz bir hedefe ulaşmaya çalıştıkları, gerçeklik bağlamında yürütülen bir tür etkinliktir. Bu tanımlar doğrultusunda oyun, ölçülebilir bir sonuç ile sonlanan, belirli kuralları olan, rekabet durumunun yaşandığı ve sonuca ulaştıran sistemik faaliyet bütünü olarak tanımlamıştır (Zimmerman, 2003).

Oyun tanımlarının ortaya konduğu süreçte oyunun temel unsurları da dile getirilmiştir. Bu unsurlar arasında yer alanların en önemlisi gerçeklik olgusundan uzak olmasıdır. Bu doğrultuda, Kolster (2013) oyunların aslında bireyin kendi fikrinde gerçeğe yakın olduğu ve bir bakıma gerçek hayatın özeti niteliğinde olduğunu dile getirmektedir. Oyunu diğer faaliyetlerden ayıran bir özelliği de rekabettir. Satrancın amacı rakibi mat etmektir, futbolun amacı rakip takımdan daha fazla gol atmak veya dijital bir oyunda oyuncuların rakip oyuncuları mağlubiyete uğratmasıdır. Bu kısımda ortak nokta rakiplerin bir hedef doğrultusunda birbirlerini yenmeye çalışmasıdır (Rowe, 1992).

Günlük yaşamda bireyler kendilerinden önceki nesillere göre teknolojik aygıtların kullanımına daha fazla vakit ayırmaktadırlar. Bu gelişim süreci aynı zamanda yukarıda bahsedilen oyun kavramının da dijital ortama taşınması ile farklılaşmış ve dijital oyun kavramının da gelişim sürecini hızlandırmıştır. “Tennis for Two” adlı elektronik programlama ile başlayan oyun süreci, sonraki dönemlerde bilgisayarların gelişip, metin tabanlı oyunlardan nesne tabanlı oyunlara doğru geçişiyle devam etmiştir. Ancak “oyun”, günümüz teknolojisinin geldiği noktada sadece bir bilgisayar oyunu olarak değil, aynı zamanda tabletlerin, konsol aygıtların ve akıllı telefonların da kullanımıyla birlikte “dijital oyun” olarak adlandırılmaya başlanmıştır (Binark, Sütçü ve Fidaner, 2009; Frasca, 2001; Gregersen, 2011; Whitton, 2014; Wolf, 2008; Yengin, 2011; Salen ve Zimmerman, 2004).

2.1. ELEKTRONİK, VIDEO VE DİJİTAL OYUNLAR

Bilgisayar, tablet ve akıllı telefon her geçen gün hayatımıza farklı bir yenilik sunmaktadır. Günlük alışveriş yaparken, eğlenirken ve iletişim kurarken bu cihazlara sıklıkla ihtiyaç duymakta ve aktif olarak kullanılmaktayız. Oyunlar, teknolojinin gelişimiyle birlikte artık farklı platformlarda oynanabilir olmasının yanında yaşamımızın vazgeçilmez bir parçası olmayı da başarmıştır. Bu boyutta şüphesiz teknolojiye ayrılan sürenin büyük bir çoğunluğu, oyuncu sayısı ve oynanan süreye göre bakıldığında dijital oyunlar için harcanmaktadır (Gaming in Turkey, 2019).

Sürekli değişim ve gelişim içinde olan dijital oyun farklı disiplinleri de bir araya getiren bir çalışma alanı haline gelmiştir. Dijital oyunlar bilgisayar bilimleri, film, sanat, canlandırma(animasyon), yeni medya, işletme ve yönetim bilimleri, göstergebilim ve psikoloji gibi bilim alanlarıyla disiplinler arası bir çalışma içindedir (Wolf ve Perron, 2003). Sanatsal yaklaşıma göre bu dijital oyunlar; drama, opera veya film gibi yeni bir ifade türü, sosyal bilim yaklaşımına göre yeni kolektif davranış biçimi, bilgisayar bilimciler, mühendisler ve endüstri tasarımcılara göre ise yeni bir buluş olarak nitelendirilmektedir (Murray, Bogost, Mateas, ve Nitsche, 2006). Birleşik Krallık Bath Spa Üniversitesi’nde oyun bilimi ve bilgisayar oyunları üzerine çalışmalar gerçekleştiren Newman (2013) oyun kuramı üzerine yapılan akademik çalışmaların yetersiz kaldığını ve göz ardı edildiğini dile getirmiştir. Aynı şekilde Wolf ve Perron (2003) dijital oyun alanında akademik çalışmaların kısırlığından dolayı ortaya net bir çalışma grubunun çıkmadığını ifade etmiştir. Güncel tanımıyla video ve bilgisayar oyunlarını

da barındıran dijital oyunlar kültürel bir form olarak karşımıza çıkmakla birlikte, medya teknolojisi ve küresel bir endüstri olarak hızla büyümeye devam etmektedir.

Video oyunları ve elektronik olmayan öncülleri arasında birtakım farklar mevcuttur. Bilgisayar oyunları otomasyon ve karmaşıklık katarak oyuncuların davranışlarına ilişkin oyun kurallarını destekleyebilir ve hesaplayabilir. Aynı zamanda ölçülebilir bir skorla sonuçlanan yapay bir sahnenin canlandırmasıdır (Juul, 2001). Ayrıca, dijital oyunlar oyuncular için görsel geri bildirim oluşturmak adına kullanıcı arabirimi veya oyun çubuğu(joystick), denetleyici, klavye veya hareket algılama cihazı gibi giriş cihazı ile etkileşimi içeren elektronik yazılımlardır. Bu geri bildirim, televizyon seti, monitör, dokunmatik ekran veya sanal gerçeklik kulaklığı gibi bir video görüntüleme cihazında gösterilir (Rozali ve diğ., 2007). Ayrıca, hoparlörler veya kulaklıklar aracılığıyla sağlanan sesli geri bildirimlerle ve bazen de dokunsal teknoloji dahil olmak üzere diğer türden geri bildirimlerle zenginleştirilir (Clark ve Scott, 2009). Barr, Noble ve Biddle (2007) dijital oyunların etkin ve mali getirisi yüksek yazılımlar olduğu vurgusunu yapmışlar ve bireylerin eğlenmek adına bu yazılımları almak için bütçe ayırdıklarını da belirtmişlerdir.

Dijital oyunlar, kişisel bilgisayar veya konsollar gibi herhangi bir elektronik platformu kullanan ve bir veya birden fazla oyuncuyu fiziksel veya ağa dayalı bir ortamda, metin tabanlı veya görüntü tabanlı eğlence yazılımlarıdır (Frasca, 2001; Sardone ve diğ., 2008; Shabanah, 2011; ScienceDaily, 2020; Hui, 2019; Arjoranta, 2019; Yengin, 2011). Kolko (2006)'a göre dijital oyun, bir veya daha fazla oyuncuya yönelik etkileşimli olan ve en basit düzeyde eğlenceyi olabildiğince artırabilmesi için geliştirilen interaktif bir program olmasının yanı sıra, dijital aygıtlarda oynanabilen herhangi bir oyundur. Yoğun ilgi ve rekabet içeren normal oyuna benzeyen ancak kendine özgü ve kurallarıyla bir cihaz aracılığıyla gerçekleştirilen etkinliktir (Przybylski ve Weinstein, 2016). Dijital oyun, bir video ekranındaki görüntülerle oynanan ve genellikle hızlı aksiyonu vurgulayan elektronik bir oyundur (Merriam-Webster Sözlüğü, 2020). Bir bilgisayarda veya video cihazında görsel geri bildirim oluşturmak için bir kullanıcı arayüzü ile etkileşimi gerçekleştiren, eğlence, oyun, kazanma / kaybetme ve rekabet gibi birçok unsuru kullanan yazılım sistemleridir (Shabanah, 2011). Oyuncunun dijital bir ara yüzle etkileşime girdiği ve oyunun konusuna bağlı olarak çeşitli türden zorluklarla karşılaştığı özel bir dijital eğlence türüdür (Zackariasson ve Wilson, 2012). Dijital oyun, belli bir topluluğun boş zamanlarındaki eğlence amaçlı aktivitelerinden doğan ve modern toplumların kültürünün ana

yapılarından biri haline gelen faaliyettir (Quandt ve diğ., 2015). Dijital oyun en temel haliyle bilgisayar, konsol, akıllı telefon ya da tablet aracılığıyla oynanabilen etkileşimli dijital eğlencedir (Owen, 2016). Yine farklı tanımla dijital oyun, bireyi eğlendirmek ya da eğitmek için tasarlanmış yazılım kodlarının genel adıdır (Computer Hope, 2018).

Son yıllarda gerçekleştirilen sektör raporlarında dijital oyunların eğlence sektöründe büyük bir paya sahip olduğu söylenebilir (Wijman, 2020). Ayrıca bu ürünlerin popüler olması ve yüksek getiri kazandırmasının yanı sıra var olan kültürün bir parçası olarak kendi kültürünün oluşumuna da ortam hazırlamaktadır. Dijital oyunlar için dergiler, web siteleri ve internet içerikleri piyasaya sürülürken bu durum bir üst seviyeye çıkarak internet üzerinden sanal dünya için gerçek para harcama yapılan bir yapı haline gelmiştir. Dahası, günümüzde çevrimiçi oyunlar olarak adlandırdığımız MOBA ve MMORPG oyun kategorisinde yer verilen League of Legends, DOTA2, Counter-Strike Global Offensive, Blade and Soul, Heroes of Storm, Path of Exile vb. oyunlarda güncel mücadele ligleri ve yıl ortası ve yıl sonu turnuvaları da son yıllarda gerçekleştirilmeye başlanmıştır.

Bilgisayar oyunlarını diğer eğlence sektörü ürünlerinden farklı kılan yapı ise, sürekli bireyin tatminkarlık seviyesini yüksek düzeyde tutmasıdır. Örneğin, bir birey beğendiği bir filmi izledikten sonra onu çok sık arayla tekrar tekrar izlemeyecektir. Yani birey uzun zaman sonra tekrar hatırlamak isterse filmi izlemek isteyebilmekte ve bunun dışında tekrar tekrar izleme durumu gerçekleşmemektedir. Ancak dijital oyunlar bu şekilde olmamakla birlikte birey tek bir seferde doygunluğa ulaşamamakta ve devamlı oyunla etkileşim kurmak istemektedir.

2.2. DİJİTAL OYUN TARİHİ VE YILLARA GÖRE GELİŞİMİ

Video oyunları, 1950'lerde bilim fuarında teknolojik ticari bir ürün olarak ortaya çıkmasından günümüze kadar dünyanın en karlı eğlence endüstrilerinden biri haline gelmiştir. Son yıllardaki mobil teknolojilerin gelişimi endüstride devrim niteliğindedir. Oyun, modern popüler kültürle bir hayli entegre hale gelmiş durumdadır. Belli dönemlerde iniş çıkış yaşasa bile günümüzde eğlence sektörünün en çok gelir getiren sektörü olmayı başarmıştır (Chikhani, 2015).

NES (Nintendo Eğlence Sistemi) 1986 yılında Amerika pazarına giriş yaptığında, dijital oyun endüstrisinde yer alan yatırımcıların çoğu, ürünün ticari açıdan başarılı olmayacağına inanmışlardır. 1970'lerin sonu ve 1980'lerin başında, TV tabanlı eğlence sistemlerinin ilk geliştiricilerinden biri olan Atari'nin yükseliş ve düşüşü ile oyun endüstrisinin bittiği

düşünülmüştür. NES, Amerika'da oyunculara sunulduktan sonra büyük satış oranıyla dikkat çekmiş ve o dönemde geliştirdiği yeni konsol ve Super Mario Bros. adlı oyunuyla oyun piyasasında üst sıralarda yerini almıştır. Ardından SEGA tarafından geliştirilen ev konsolları ile oyun pazarı farklı bir yöne doğru evrilmiştir (Pruett, 2003). Bu süreçte SEGA konsollarından sonra daha az maliyetli, daha iyi grafikler sunan ve aynı cihazlar üzerinde çalışabilen PlayStation piyasaya sürülmüş ve rekorla döneminin en çok satan konsol oyunu olmayı başarmıştır. Yaklaşık yirmi yıl sonra, oyun endüstrisi devasa boyutlara ulaşmış ve oyunların karmaşıklığı da katlanarak artmıştır. NES için geliştirilen bir oyun birkaç kişilik grupta birkaç ay içinde tamamlanabilirken, Playstation 2 için bu süre yirmi ila altmış kişilik ekiplerin çalışmasıyla yaklaşık birkaç yıl sürmüştür (Gamble, 2008).

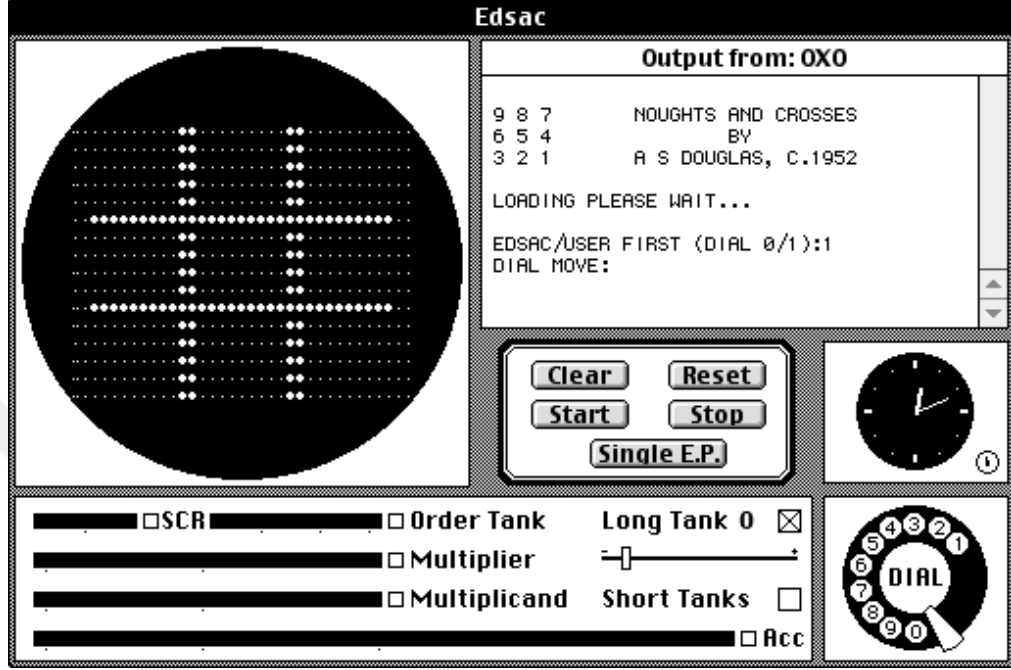
Oyun endüstrisinin arkasındaki gücün büyük bir kısmı bilgisayar teknolojisindeki yenilikler ve gelişmeler olmuştur. Video oyunları giderek daha güçlü teknolojik sistem gereksinimlerine ihtiyaç duyduğundan, oyuncular grafik ve oynanış açısından daha iyi geliştirilmiş oyunlar talep etmişlerdir. Özellikle, gerçekçi üç boyutlu grafiklerin üretimiyle grafik kartları, oyun donanımı üreticileri için en önemli satış noktası haline gelmiştir. Bu gelişim sürecinde günümüz oyun sistemleri güçlü bilgisayarlardan oluşmakta ve oyun içeriğinin oluşturulması daha zor hale gelmiş durumdadır. Video oyunlarının yıllara göre gelişimi ve değişimi dönem bazlı aşağıda detaylı şekilde verilmiştir.

2.2.1. 1950 ile 2000 Yılları Arası Oyun Tarihi

Oyun tarihinde dijital ortamda üretilen oyunların çoğu tek bir yazılım mühendisi tarafından veya bilgisayar bilimi öğrencilerinin dönem projeleri olarak geliştirilmiştir (Kent, 2001). Bu dönemde yer alan oyunlar, daha çok yazılım paketi halinde programlanmıştır. Bu dönemde karşılaşılan zorluklardan birisi güncel donanım, yazılım ve diğer yapılarda bu oyunları destekleyecek türden olmaması ve disiplinler arası bir çalışmayı gerektirmemesidir.

1952 yılında OXO, Alexander Sandy Douglas tarafından geliştirilen dünyanın ilk elektronik tabanlı oyunudur. Douglas, oyunu Cambridge Üniversitesinde doktora öğrencisi iken doktora tezi için geliştirmiştir. Oyun, kâğıt ve kalemle, iki kişi tarafından oynanan EDSAC bilgisayarı için geliştirilmiş 35x26 piksellik video aygıtı ile arayüz sağlayan bir “Tic Tac Toe” oyunudur. (Brookhaven National Lab, 2008; Cohen, 2019; MuseumofPlay, 2016). Bu oyunun uluslararası

anlamda oynanmamasının sebebi EDSAC cihazının teknolojik durumu ve sürekli üniversitenin hizmetinde olmasıdır.



Şekil 2.1: İlk elektronik Tic Tac Toe oyunu olan XOX'e dair bir kesit (EDSAC, 1952)

Elektronik oyun niteliğinde geliştirilen bir sonraki oyun, 1958 yılında Amerikalı Fizikçi William Higinbotham tarafından çalışma laboratuvarında bilimsel araştırmalarını gerçekleştirirken, aynı zamanda ziyaretine gelen arkadaşlarını eğlendirmek için yazılmıştır (Kowert ve Quandt, 2016; Nosowitz, 2008). Söz konusu oyun beş inçlik bir osiloskop cihazı ekranı üzerinde ortadan geçen bir çizgiyle birlikte topu temsil eden bir noktanın iki adet kumanda ile topu karşı tarafa geçirmeye çalıştıkları “Tennis For Two” adlı analog bir yazılım olarak karşımıza çıkmıştır. Analog ortamda geliştirilen bu oyun video oyunları tarihinde geliştirilen ilk oyunlardan biri arasında kayıtlara geçmiştir (Brookhaven National Lab, 2008; Tezel, 2016).

1960'lı yıllara gelindiğinde, MIT Üniversitesi'nde Steve Russell, J.Martin Graetz ve Wayne Witanen tarafından 1962 yılında iki oyuncunun ekranın ortasındaki bir yıldızın karakterleri içine çeken ve bunu önlemek içinde oyuncuların manevra yaptığı, PDP1 olarak adlandırılan skorların görülebildiği bilgisayarların yanında katot ışın tüplerinin desteklediği elektronik cihazlarda da oynanabilen oyun olan “Spacewar” piyasaya çıkmıştır. Skor görüntüleme özelliğiyle ön planken, bu özellik osiloskop cihazlarında mevcut değildir.

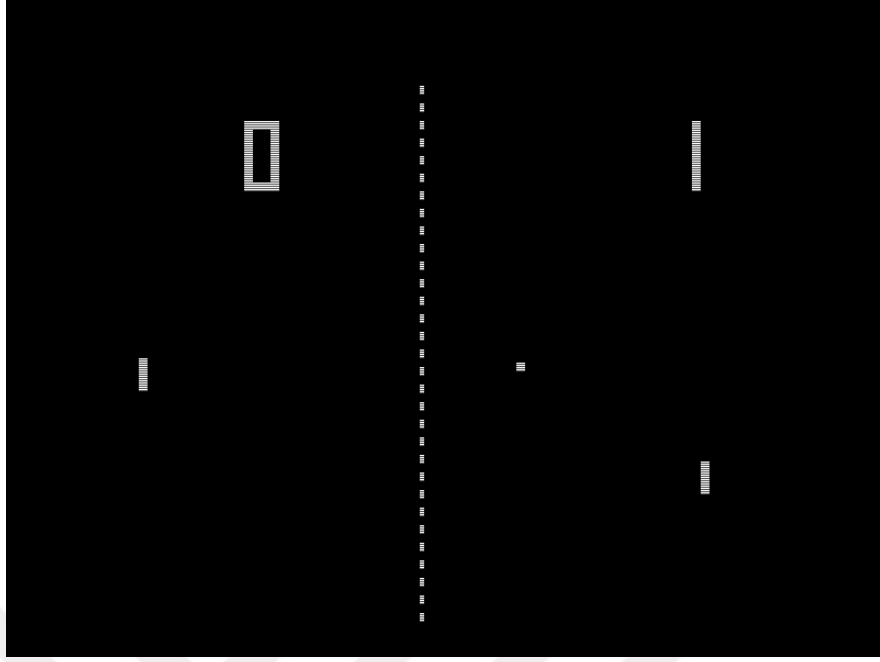


Şekil 2.2: “Spacewar” oyunundan bir kesit (MIT Museum Collections, 1961)

Spacewar bu bakımdan analog teknolojiden ziyade dijital derleme yapabilen bir cihazda çalıştığından ötürü ilk olarak ticarileştirilen ürün olarak kayıtlara geçmiştir. Aynı zamanda birden fazla cihazda oynanabilen ve ticarileştirilmiş ilk oyun olma özelliğini korumaktadır (King, 2002; Monnens ve Goldberg, 2015; Morristown, 1981; Weller ve diğ., 2004).

Video oyunları kategorisinde 70’li yıllar Altın Çağ olarak adlandırılmaktadır. Aynı şekilde Spacewar’ın hemen ardından Computer Space 1971 yılında Amerika Birleşik Devletleri’nde oyunculara sunulmuştur. Ancak fazla tutulmamasıyla kısa süre içinde unutulmuştur. Yaklaşık bir sene sonra Atari Inc. tarafından piyasaya sürülen Pong adlı oyun ile oyun endüstrisinde büyük bir gelişme yaşanmıştır. Pong’un ardından aynı yıl Magnavox Odyssey ile büyüme hızla devam etmiştir. Bu oyunun piyasaya sürüldüğü cihaz türüne bakıldığında evlerde oynanan ilk ticari oyun konsoludur (Chang ve diğ., 2007; Lowood, 2009; M. J. P. Wolf, 2008b).

Atari Pong adlı oyunun ticarileştirilmesinde atılan ilk adımlardan biriyse oyunların evlere girmesini sağlamasıdır. 1975 yılında Pong oyununun ev versiyonu tanıtılmış ve bunun gerçekleştirilmesinde en büyük rolü Atari’nin kurucusu Nolan Bushnell üstlenmiştir.



Şekil 2.3: Atari'nin Pong adlı oyunundan bir kesit (wikipedia.org, 2020b)

1980'li yıllara doğru gelindiğinde sektörde rekabet canlanmış ve birçok yatırımcı tarafından oyun endüstrisine yatırım yapılmaya başlanmıştır. Eğlence sektörünün değişmez bir parçası olmaya doğru bu dönemde önemli bir adım atılmıştır. Bu dönemin ilk popüler oyunlarından biri olan Pac-Man karşımıza çıkmaktadır. Pac-Man oyunu 1980 yılında Amerika'da çok kısa sürede en çok oynanan oyunlar arasında yer almıştır (Gallagher ve Ryan, 2003; Klein, 1984).

Yine bu yıllarda Donkey kong adlı oyunun piyasaya sürülmesiyle birlikte oyunların "Altın Çağı" daha da parlamaya başlamıştır. Donkey Kong, Nintendo tarafından 1981 yılında piyasaya sürülen bir platform oyunudur (Aloupis ve diğ., 2015; Barnes, 1994; DeWinter, 2015). Oyun, kısa bir süre sonra karakterin mesleği plumber (tesisatçı) olarak değiştirilerek ismi Mario şeklinde piyasaya tekrar sunulmuştur (Wesley ve Barczak, 2010).

1990 yılında Microsoft, Windows 3.0 sürümlü işletim sisteminin içine klasik kart oyunu olan Solitaire'i paket halinde kullanıcılara sunmuştur. Dönem bazlı bakıldığında kendi döneminde oynanması kolay ve sıradan oyun modelini kullanan Solitaire (Halliburton ve Pearson, 2002) oyuncular için en popüler oyunlar arasında üst sıralarda yer bulmuştur. 1991 yılında SEGA firması oyun sektöründe adeta büyük bir adım atarak Japonya'da MEGA DRIVE olarak bilinen GENESIS adlı 16-bit hafızaya sahip evlerde oynamaya yönelik olarak üretilen oyun konsolunu piyasaya sürmüştür. Bu konsol için SONIC adlı oyunu geliştirmiştir.

1996 yılında önceden bahsedildiği üzere Tomb Raider popüler olarak oynanan oyunlar arasında yer almayı başarmıştır. Core Design tarafından geliştirilen bu oyun birçok platformda yayınlanmış ve büyük bir kitle tarafından oynanmıştır. 1998 yılında Legend of Zelda: Ocarina of Time oyunu Legend of Zelda sersinin beşinci oyunu olarak tekrardan piyasaya sürülmüştür. Karakter yapısı, uzamsal bulmacaları ve o zamana kadar diğer oyunlarda çok fazla üstünde çalışılmayan bir unsur olan oyun müzikleriyle ön plana çıkan oyunlardan biri olmayı başarmıştır. İnternet'in icadından sonra halkın kullanımına sunulmasıyla birlikte oyun sektöründe daha ciddi adımlar atılmış ve farklı türde oyunlar geliştirilmeye başlanmıştır. 1999 yılında yine SONY tarafından geliştirilen 3 boyutlu fantastik temalı ve çok oyunculu rol yapma olan Everquest piyasaya sürülmüştür. Everquest oyun piyasasında 3 boyutlu bir oyun motoru kullanan ve ticari anlamda büyük bir başarılı olan MMORPG oyunudur (Walker, 2006). Aynı zamanda çevrimiçi oyunlar arasında türünün en çok oynanan örneklerinden biridir.

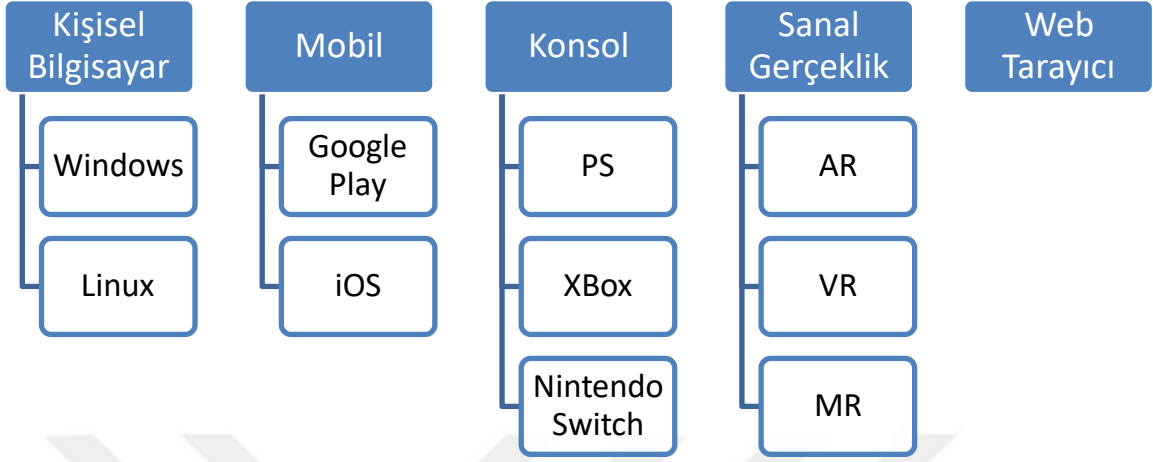
2000'li yılların başında oyun sektöründe büyük bir payı olan SEGA, Dreamcast adını verdikleri yeni nesil konsolları piyasadan çektiklerini duyurarak artık sadece oyun geliştirmeye odaklandıklarını açıklamıştır. Aynı sene içinde PlayStation 2 Sony tarafından tanıtılmış ve çıktığı ilk günden itibaren 13 yıl içinde 157,68 milyon adet satarak yeni bir rekorla tüm zamanların en çok satan video oyun konsolu olmayı başarmıştır (Plant, 2018). Aynı süreçte Microsoft tarafından üretilen ve günümüz en popüler oyun konsollarından biri olan XBOX cihazı pazara giriş yapmıştır. Daha önce SEGA ile başlayan internet oyunları Microsoft tarafından 2002 yılında geniş bant genişliği ve indirilebilir içeriğe sahip olmasıyla Xbox Live ile daha büyük kitlelere ulaşılabilir hale gelmiştir.

Akıllı telefon üreten firmaların sayısının artması kullanıcıların akıllı telefonları daha az maliyetle satın almasını sağlamıştır. Hemen hemen herkes günümüzde akıllı cihazlara erişim sağlayabilmektedir. 2020'de, küresel akıllı telefon kullanıcılarının sayısının 2019'a göre yüzde 9,3 artışla 3,5 milyar olacağı tahmin edilmekteydi (O'Dea, 2020). Ülkemize baktığımızda 2019 yılına göre %4,4 artışla Ocak 2020 verilerine göre 62,07 milyon internet kullanıcısı mevcuttur. Öte yandan akıllı cihazlarda mobil erişim oranları Türkiye'de Ocak 2020'de 77,39 milyon mobil bağlantı sayısına ulaşmıştır. Bu da Türkiye özelinde akıllı cihaz kullanımının yüksek olması, Telekom, mobil oyun, sosyal medya ve reklamcılık gibi sektörlerinde potansiyelinin artırılmasında etkili olmaktadır (Kemp, 2020).

2.3. DİJİTAL OYUN TÜRLERİ VE PLATFORMLAR

Dijital oyunlar hem içerik hem de oynanışları bakımından kendi öncüllerinin piyasaya çıkışından bu yana büyük bir değişim ve gelişim göstermiştir. Her geçen yıl dijital oyun platformlarının çeşitliliği artmaktadır. Bu platformlar oyunculara farklı deneyimler sunarken, oyuncuları sanal olarak kurgusal bir dünyaya götürmektedir. Dijital oyunlarda oyuncular arası etkileşim kişisel bilgisayar ya da cep telefonlarıyla olabildiği gibi jetonla çalışan makinelerde veya belirli oyunlar için özel olarak geliştirilmiş konsollarda da olabilir. Dijital oyunlar, nispeten basit ve metin tabanlı oyunlardan, grafik açısından zengin ve büyük bir ekip tarafından geliştirilmiş sanal ortamlara kadar değişkenlik gösterebilir. Bazı oyunlar tek kullanıcı (singleplayer) olarak küçük gruplar halinde oynanırken, bazıları ise çoklu oyuncularla (multiplayer) coğrafi olarak birbirine yakın olmayan kitlelerce de oynanabilir (Winget, 2011).

Elektronik oyunlar kategorisinde ilk oyun, Katot Işın Tüpü Eğlence Cihazı (Cathode Ray Tube Amusement Device) üzerinde 1947 yılında geliştirilmiştir. Dijital oyun endüstrisinde ise 1960'lı yıllardan günümüze kadar çeşitli konsollar oyuncuların kullanımına sunulmuştur. 1960'lı yıllarda oyunlar analog sistem elektronik cihazlar üzerinden oynanırken artık teknolojinin gelişmesiyle birlikte bilgisayar, konsol ve mobil cihazlar olmak üzere farklı kategorilere ayrılmıştır. 1990'lı yıllara kadar bilgisayarlar ve konsollar kullanılırken, günümüzde mobil cihazlar bu gruba dahil olmuştur. Dijital oyunların türünü ifade ederken "platform" kategorisi, oyunun oynandığı donanımları ve sistemleri ifade etmektedir. Buna kişisel bilgisayarlar, konsollar ve akıllı mobil cihazlar da eklenebilir (Apperley, 2006). Kişisel bilgisayar platformunda Windows ve Linux, konsol kategorisinde Playstation, XBOX, Nintendo Switch ve Wii, mobil platformda ise Android ve IOS olarak kendi içlerinde gruplandırılmaktadırlar. Platformlara dair detaylı görsel şema Şekil 2.9'da verilmiştir.



Şekil 2.4: Dijital oyun platformları ve alt dallar

2.3.1. Kişisel Bilgisayarlar

Kişisel bilgisayar oyunları, video oyunlarının en iyi deneyim sunduğu platformlardan biridir. Bu platformlar, yüksek kaliteli görselleri desteklemesi ve daha fazla oynanabilirlik sunması avantajıyla ön plana çıkmaktadır. Çoğu zaman, oyun grafikleri kişisel bilgisayar platformunda oyun konsolundan daha iyi performans gösterirler. Çünkü bilgisayar oyunları, deneyimi geliştirebilecek çeşitli aksesuarlar ve seçenekler aracılığıyla daha fazla özelleştirme sağlamaktadır. Örneğin, oyun oynamak için klavye ve fare kullanılabilirken aynı zamanda bir oyun kumandası da entegre ederek kullanılabilir. Kişisel bilgisayar oyunları ayrıca Steam ve Epic Store gibi oyun satımı yapılan ortamlar sayesinde oynanabilecek daha geniş bir oyun yelpazesi sunmaktadır (Galehantomo, 2015).

2.3.2. Konsollar

1985 yılında oyun sektöründe baskınlığını elinde tutan Nintendo kendisinden sonra daha az maliyetli, yüksek grafikleri destekleyen ve standart donanım özelliklerine sahip konsolları geliştiren Sega, Sony ve Microsoft ile karşı karşıya kalmıştır. Bu firmaların sektöre girişiyle birlikte Nintendo konsollar artık evlerde oynanmayarak popülarlığını yitirmiştir. İlerleyen süreçte Sony tarafından piyasaya sürülen PlayStation ve Microsoft tarafından piyasaya sürülen

XBox konsoluyla birlikte oyun endüstrisi pazarında konsol çeşitliliği artmıştır. Bu değişime ayak uyduramayan Nintendo sonradan etkisini yitirmeye başlayarak, pazarı Sony ve Microsoft'a bırakmak zorunda kalmıştır. Ardından satış patlamaları ile rekorlar kıran Sony PS günümüzde de hala en çok oynanan konsol olma özelliğini devam ettirmeyi başarmıştır.

Yukarıda da bahsedildiği üzere Microsoft, Sony ve Nintendo'nun oyun konsolları yıllardır oyun platformlarında kullanılan cihazların en başında gelmektedir. Oyun konsollarının kullanımı kişisel bilgisayarlara göre biraz daha kolaydır çünkü konsol oyunları genellikle oyun kolu (joystick) ile oynanmaktadır. Pek çok şirket yeni nesil konsollar geliştirmeye devam ederken, halihazırdaki konsollar ise yeni sürümlerini oyunculara sunmaktadır (Gamble, 2008).

Konsol platformu sadece oyun oynama hedefine hizmet ettiğinden ötürü oyun firmalarının gözde cihazlarından biri olmayı başarmıştır. 2000'li yıllardan günümüze birçok yüksek bütçeli oyun konsol cihazlar için geliştirilmiştir. Bu dönemde geliştirilmiş daha fazla oyun olmakla birlikte bazı önemli oyunlara aşağıda yer verilmiştir.

Battlefield, 2002 yılında piyasaya sürülen Battlefield 1942 ile Microsoft Windows ve OS X üzerinde başlayan bir dizi birinci şahıs nişancı video oyunudur. Seri İsveçli EA DICE tarafından geliştirilmiş ve Amerikan şirketi Electronic Arts tarafından yayınlanmıştır. Yaklaşık 50 milyon oyuncu tarafından oynanmıştır.

Assassin's Creed, Ubisoft tarafından yayınlanan ve esas olarak Anvil oyun motoru ve daha gelişmiş türevlerini kullanan stüdyosu Ubisoft Montreal tarafından geliştirilen bir açık dünya aksiyon-macera gizli video oyunu imtiyazıdır. Call of Duty, Activision tarafından yayınlanan birinci şahıs nişancı video oyunu serisidir. Far Cry, tamamı Ubisoft tarafından yayınlanan birinci şahıs nişancı oyunlarının bir imtiyazıdır. İlk oyun olan Far Cry Crytek tarafından CryEngine oyun motorunda geliştirilmiş ve Mart 2004'te piyasaya sürülmüştür. Yine dönemin popüler oyunlarından Deus Ex rol yapma oyunu olarak karşımıza çıkmaktadır.

2.3.3. Mobil

Akıllı cihazlarda ve tabletlerde geliştirilen oyunlar son 10 yılda hızlıca popüler hale gelen yeni bir platform olarak karşımıza çıkmaktadır. Günümüzde kullanıcıların çoğu akıllı mobil cihazda farklı türlerde oyun oynamakta ve bu ilgi oyun endüstrisini son zamanlarda çok daha popüler hale getirmektedir.

Özellikle Angry Birds, FarmVille, Clash of Clans, PUBG ve Fortnite gibi popüler oyunların tercih edilmesiyle daha oynanır hale gelen mobil platform, oyun sektöründe kendine üst sıralarda yer bulmuştur. Son yıllarda mobil oyunlar, Pokémon Go ve yeni çıkan Harry Potter: Wizards Unite gibi etkileşimli oyunlarla artırılmış gerçeklikten de yararlanarak gelişmeye devam etmektedir. Mobil oyunlarda son yıllarda Hyper Casual türünde birçok oyun geliştirilmiştir ve popüler olarak oyuncular tarafından oynanmaya başlanmıştır. Son yıllarda akıllı mobil cihazlarda geliştirilen oyunların çoğu kolay oyun (hyper casual) akımından etkilenerek geliştirilmiş ve halen geliştirilmeye devam etmektedir (Facebook, 2019).

2.3.4. Sanal Gerçeklik

Sanal gerçeklik, genel tanımıyla gerçek dünyayı yapay bir dünya ile deneyimlemeye izin veren ve kullanıcıyı başka bir dünyada olduğuna inandıran teknolojidir (Casas ve diğ., 2017). Kullanıcıların tıpkı gerçek hayatta yaptıklarına benzer şekilde etkileşimde bulunabilecekleri, bilgisayar tarafından üretilen sanal ortamlar oluşturmak için kullanılan bir dizi teknolojiyi içerir. Bu amaçla, stereoskopik ekranları, hareket izleme donanımını, girdi cihazlarını ve yazılım platformlarını entegre etmektedir (Parisi, 2015). Mobil oyun platformunun yanı sıra, oyun endüstrisinde sanal gerçeklik uygulamaları gittikçe daha da geliştirilerek platformda büyük gelişmeler yaşanmaktadır. Sanal gerçeklik, oyuncularına sanal dünya deneyimi yaşatmakta ve mevcut olan en sürükleyici oyun deneyimlerinden birini sunmaktadır. Oculus, Sony, HTC, Samsung ve diğer şirketler birçok farklı türde sanal gerçeklik platformuna cihaz geliştirmektedir. Sanal gerçeklik, oyun oynamanın ötesinde en büyük potansiyele sahip olabilecek oyun platformu olmasıyla ön plana çıkmakta olup, önümüzdeki yıllarda daha popüler hale geleceği düşünülmektedir. (Cruz-Neira ve diğ., 2018).

2.3.5. Web Tarayıcı

İnternet, bugün her yaştan insan için oyun oynamayı en erişilebilir hale getirmiş ve kullanıcıların internete kolay erişim sağlamasıyla oyun oynamak da kolay hale gelmiştir. Televizyon programlarından esinlenerek geliştirilen oyunlardan RuneScape ve FarmVille gibi klasiklere kadar İnternette binlerce oyun ücretsiz olarak oyunculara sunulmaktadır. Web tarayıcı üzerinden geliştirilen birtakım oyunlar özellikle günümüzde alt kademe eğitim sınıflarında bir kavramı detaylı şekilde öğretebilmek için etkin şekilde kullanılmaktadır. Ayrıca

internet üzerinden oynanan oyunlarda güçlü sistem gereksinimleri ve özel bir yükleme süreci içermediğinden ötürü bazı oyuncuların sevdiği bir ortam haline gelmiştir.

2.4. DİJİTAL OYUN TÜRLERİ

Oyun türlerini kategorize etmek, oyunları belirli bir düzende anlayabilmek ve onlara kolay erişim sağlama konusunda önemli bir husustur. Video oyun türleri farklı kategoriler altında değerlendirilmektedir. Kategorize etme, oyunların oynanış tarzları ve kamera açıları baz alınarak gerçekleştirilmektedir (Jin Ha Lee ve diğ., 2014).

Bir video oyunu kategorize edilirken oyunun tek oyunculu mu, çok oyunculu mu, yoksa bir ağ üzerinden oynanabilecek şekilde mi tasarlandığını sormak yerine oyunun kendi içindeki etkileşimin temel yapısı dikkate alınır. Bir oyun, oynandığı platforma göre (kişisel bilgisayar, mobil veya konsol), sağladığı oyun tarzına göre (çok oyunculu, ağ bağlantılı veya tek oyunculu), oyun dünyasıyla (birinci şahıs veya üçüncü şahıs), oyun tarzını oluşturan kurallar ve hedeflerle (yarış oyunu veya aksiyon macera vb.) veya temsili yönleriyle (bilim kurgu, fantastik veya aksiyon) sınıflandırılabilir (Carr, 2006). Bu türler; aksiyon/macera, masa, dövüş, platform, simülasyon, yarış, rol yapma ve strateji olarak listelenebilir. Şekil 2.5'te oynanış (gameplay) ve kamera açısı (perspective and view point) unsuruna göre detaylı tanımlara yer verilmiştir. Aşağıda oyun türlerinin tanımları ve kategorizasyonu Apperley (2006) ve Wolf (2002) tarafından geliştirilmiş ve ilerleyen süreçte Krzywinska ve Brown (2015) düzenlemeler yapmıştır. Bu oyun türleri aşağıda detaylı olarak verilmiş olup, örnek oyunlar ise araştırmacı tarafından her bir türe özel eklenmiştir.



Şekil 2.5: Oynaniş ve kamera açılarına göre dijital oyun türleri

Aksiyon/Macera (Action/Adventure): Genellikle birden fazla seviyenin birbiriyle bağlantılı olduğu, oyun ortamında yakalama, ateş etme ve kaçma gibi basit ve temel oyun mekaniklerini içeren oyun türüdür. Bu tür oyunlarda seviyeler genellikle birkaç adımda tamamlanır. Örneğin, oyunun başka bir yerinde ihtiyaç duyulan nesneleri almak veya diğer alanlara açılan kapıları açmak oyun içinde tanımlanan nesnelere erişmek gerekir. Oyun içinde karakterler genellikle silah, mühimmat, anahtar ve buna benzer nesneleri taşıyabilirler.

- **FPS (Birincil Şahıs Oyunları – First Person Shooter):** Birincil şahıs gözüyle oynanan aksiyon oyunları olarak adlandırılmaktadır. Half-Life, Quake, Counter-Strike, Overwatch bu kategoride yer alır.
- **TPS (Üçüncül Şahıs Oyunları – Third Person Shooter):** Üçüncül şahıs gözünden farklı perspektiflerle oyuncu karakterini gösteren oyun türüdür. Resident Evil, GTA serisi, Max Payne, Tomb Raider, Fortnite bu kategoride yer alır.

Masa Oyunları (Board Games): Mevcut masa oyunlarının bir uyarlaması şeklinde dijital ortamda sunulan veya daha önce masa oyunu olarak sunulmasa bile tasarımında ve oynanışında

masa oyunlarının mekanik ve dinamiklerinin kullanıldığı oyunlardır. Paperback, 7 Wonders ve Splendor gibi oyunlarda bu kategoride yer alır.

Dövüş (Fight): Oyuncu ve bilgisayar karşılaşması şeklinde veya iki ya da daha fazla oyuncuya oynama imkânı sunan, dengeli şekilde geliştirilmiş dövüş oyunlarıdır. Mortal Kombat, Street Fighter ve Tekken serileri bu kategoride yer alır.

Platform: Oyunda karakterin koşma, tırmanma, zıplama ve diğer hareketler gibi bir dizi mekaniklerin yer aldığı oyunlardır. Spirit of Maya, TownFall, Caveblazers ve Celeste bu kategoride yer alır.

Yarış (Racing): Oynayanın bir yarışı kazanmasını veya rakipten daha fazla alan ele geçirmesini hedefleyen oyunlardır. Forza Horizon, Gran Turismo ve Need for Speed (NFS) bu kategoride yer almaktadır.

Simulasyon (Simulation): Simülasyon türü sporları, farklı araçların sürüşünü canlandıran oyunları içermektedir. Bir bakımdan gerçek olayların sanal dünyada canlandırılması olarak nitelendirilebilir. Euro Truck Simulator, Cities: Skylines ve Bus Simulator bu kategoride yer alır.

Rol Yapma (Rol Playing): Oyuncuların kurgusal bir ortamda karakterlerin rollerini üstlendiği ve oyuncuların hikâyeyi baştan sonra kadar öğrendikleri ve başarılı bir şekilde bitirdikleri oyun türüdür (Tychsen, 2006). Oyuncular gerçek rollerle ya da karakter gelişimiyle ilgili yapılandırılmış bir karar verme süreciyle birlikte bir anlatı içinde bu rolleri canlandırma sorumluluğunu üstlenirler. The Witcher serisi, World of Warcraft, Undertale ve Dark Souls gibi oyunlar bu kategoride yer alır.

MMORPG (Devasa çok oyunculu çevrimiçi rol yapma oyunu – Massively multiplayer online role playing game): Geniş kitlelerce oynanan çevrimiçi rol yapma oyun türüne verilen isimdir. Farklı coğrafi konumlardan oyuncuların internet bağlantısıyla oyunda aktif olarak bir karaktere büründüğü ve o karakteri yönlendirdiği oyun türüdür. Blade and Soul, Albion Online ve EVE online buna örnek verilebilir.

- **MOBA (Çok oyunculu çevrimiçi savaş arenas – Multiplayer Online Battle Arena):** Çevrimiçi çok oyunculu savaş arenas ya da MOBA, bir gerçek zamanlı strateji savaş video oyunu türüdür. League of Legends, DOTA II ve Heroes of the Storm bu kategoride yer alır.

Strateji (Strategy): Bu oyunlar genellikle hızlı olmayı veya reflekslerin kullanılmasını gerektirmeyen strateji kullanımını vurgulayan oyunlardır (Apperley, 2006; Wolf, 2008; Krzywinska ve King, 2006).

- **RTS (Gerçek Zamanlı Strateji Oyunu- Real Time Strategy):** Gerçek zamanlı strateji adıyla oyuncuların kendi aralarında bir sıra bekleme olmaksızın karşılıklı olarak oynanan oyun türüdür. Starcraft, Spore ve Age of Empires serisi örnek verilebilir.
- **TBS (Sıra tabanlı strateji – Turn Based Strategy):** Sıra tabanlı strateji oyunları temel olarak sıra ile oynanan strateji oyunları olarak tanımlanabilir. Civilization, Advance Wars ve Panzer General bu tarz oyunlara örnek verilebilir.

2.5. MÜŞTERİ VE MÜŞTERİ KAYBI

Teknoloji, müşteri ve şirket ilişkisi arasında önemli bir köprü görevi üstlenir. İnternet ve mobil cihazların yükselişiyle, müşteriler bilgiye ve şirketlerin sunduğu imkanlara daha kolay erişmiştir. Bu durum, müşteri beklentilerinin artmasına ve müşteri sadakati kavramının ortaya çıkmasına temel hazırlamıştır. İşletmeler değişikliklere uyum sağlamak ve müşterilerinin ihtiyaçlarını karşılamak için teknolojiden yararlanmayı hedeflerler. Bu bağlamda teknoloji, müşteri etkileşimlerinin şirketlere iletilmesine, e-ticaret siteleri ve sosyal medya gibi çevrimiçi platformlarda üretilen bilginin aktarılmasında yardımcı olur. Aynı zamanda müşteriler mekan ve zaman gözetmeksizin mal ve hizmet satın alımı yapabilmekte ve daha geniş bir ürün ve marka yelpazesine erişebilmektedir. İnternet, müşterilerin ürünleri araştırması, fikir sahibi olması ve işletmelerle bağlantı kurmasını kolaylaştırmıştır. Teknoloji, müşterilerin etkileşim biçimini büyük ölçüde etkilemiş ve müşteri beklentilerinde de değişime sebep olmuştur. Bu beklentileri karşılamak için teknolojiye uyum sağlayan ve teknolojiden yararlanan işletmeler, günümüzün dijital dünyasında müşterileri elde tutmak için daha anlamlı ve operasyonel faaliyetler gerçekleştirmektedir.

Müşteri, pazarlama ve müşteri ilişkileri yönetimi alanlarının merkezinde yer alır ve işletmeler için kilit role sahip olduğu kabul edilir. Müşteri bir şirketin gelir kaynağı olduğundan ötürü, müşteri sadakati de şirketler için önemlidir. Akademik alanyazın incelendiğinde, müşteri kavramının genellikle pazarlama ve müşteri ilişkileri yönetimi bağlamında ele alındığı görülmektedir. Peki, müşteri nedir? Müşteri, belirli bir işletmenin belirli bir marka malını idari veya kişisel amaçlarla satın alan kişi veya kuruluştur (Gold ve Tzuo, 2020; Beyer ve Holtzblatt, 1995; Meyer ve Schwager, 2007; Buttle, 2004; AMA, 2017).

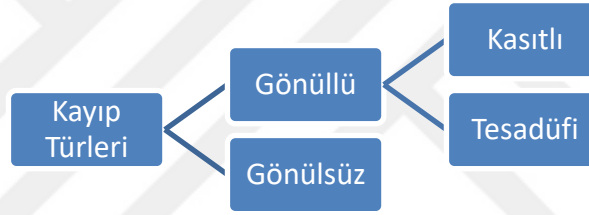
Müşterilerin ihtiyaçlarını ve tercihlerini anlamak, ürünlerini veya hizmetlerini geliştirmek ve rekabet avantajı elde etmek isteyen işletmeler, müşterilerin ürün ve hizmetlerden ne kadar memnun olduklarını öğrenmeye odaklanırlar. İnternette sunulan ürün sayısının ve rekabetin artmasıyla birlikte Müşteri İlişkileri Yönetimi (CRM) daha da önemli hale gelmiştir. Müşteri İlişkileri Yönetimi (CRM), müşteri ilişkilerini geliştirmek ve verimliliği arttırmak için birçok farklı türde işletme ve kuruluş tarafından kullanılabilen çok yönlü bir araçtır. Satış ve pazarlama alanında müşteri etkileşimlerini yönetmek, hizmet ve destek alanında hizmet ve destek ekipleri tarafından müşteri şikayetlerini ve hizmet taleplerini yönetmek, e-ticarete web sitelerindeki ve mobil uygulamalardaki müşteri etkileşimlerini analiz etmek, finansal hizmetlerde müşteri etkileşimlerini yönetmek ve finansal işlemleri takip etmek için ve sağlık hizmetlerinde ise hasta etkileşimlerini yönetmek gibi hedeflerde kullanılmaktadır.

Müşteri ilişkileri yönetimi özellikle çevrimiçi ve mobil oyun platformları ile yakından ilişkilidir. Müşteriler, şirketlerin gelir getirisinde en önemli kilit unsur olarak değerlendirilirken belirli süre sonra gelecek dönemlere yönelik ayrılacak abonelerini tespit etme ve tanımlama sürecinde de oldukça önem arz etmektedirler. Bu bakımdan şirketler mevcut müşterileri korumak için müşteri kaybına dair birtakım analizler gerçekleştirirler. İşletme alanında bu süreç müşteri kayıp analizi olarak karşımıza çıkar. Herhangi bir abone, kullanıcı veya oyuncunun aldığı hizmeti bırakması "kayıp" olarak nitelendirilirken, müşterinin bir hizmeti kullanmayı bırakması veya aboneliğini iptal etmesi üzerine şirketin buna karşı gerçekleştirdiği analizler ise "müşteri kayıp analizi" olarak nitelendirmektedir (Forhad ve diğ. 2014).

Bir kullanıcı grubuna ait geçmiş hizmetlerin kullanımı, harcama ve diğer davranış kalıplarının tamamı müşterilerin kayıp analizi kapsamında değerlendirilirken, müşterinin hizmeti sonlandırmak isteyip istemediği olasılığı da bu analizlerle belirlenebilir. Çoğu hizmet sağlayıcı firma, satın alma sürecine odaklanır fakat bir şirketin başarısındaki temel yapı taşlarından biri

de verdiği hizmette dalgalanmayı en aza indirecek şekilde faaliyetlerini yürütmesidir. Müşteri kaybı sürekli ve proaktif (bilinçli ya da bilinçsiz, olumlu ya da olumsuz mevcut koşulların sonucunu değiştirmek için inisiyatif kullanma) şekilde ele alınmazsa, ürün veya hizmet tam potansiyeline ulaşamayacaktır.

Müşteri kaybı, işletmenin herhangi bir nedenle müşteri kaybetmesi olarak tanımlanır. Bu doğrultuda, müşteri yaşam döngüsünün normal bir parçasıdır. Aynı zamanda şirket için sürekli aktif kalamazlar, dolayısıyla müşteri kaybı bir şirketin uzun vadeli sağlığının önemli bir göstergesidir (Poel ve Lariviere, 2004). Bu doğrultuda Şekil 2.6'da farklı sektörlerde müşteri kaybı konusunda müşteriler gönüllü ve gönüllü olmayan olmak üzere iki ana başlık halinde verilmiştir.



Şekil 2.6: Kayıp türlerinin sınıflandırılması (Shaaban ve diğ., 2012)

2.5.1. Gönüllü Müşteri Kaybı

Gönüllü müşteri kaybı, müşterinin kendi istekleri doğrultusunda güncel işletmeden aldığı hizmeti sonlandırarak farklı bir işletmeyi tercih etmesidir (Hadden ve diğ., 2006; Ahn ve diğ., 2020). Tesadüfi (incidental) ve kasıtlı (deliberate) kayıp olmak üzere iki alt kategoriye ayrılır.

Tesadüfi kayıp, müşterilere plan dahilinde uygulananndan ziyade, hayatlarında oluşan bir ansızın gelişen problemten dolayı gerçekleşir. Örneğin, müşterinin vefat etmesi, bir sağlık problemi ile karşılaşması, müşterinin mali durumunda değişiklik olması, şirketin verdiği hizmeti karşılayamaması veya şirketin hizmetini kullanmadığı farklı bir bölgeye taşınması olarak sıralanabilir (Shaaban ve diğ., 2012).

Kasıtlı kayıp ise müşterinin kendi isteği doğrultusunda belli sebeplerden ötürü gerçekleşen kayıp türüdür. Örneğin, bir bankanın müşterisinin diğer bankalardan aldığı cazip faiz

oranlarından etkilenecek hesabını farklı bir bankaya taşıması veya telekom hizmeti alan bir kullanıcının cazip görüşme paketlerini değerlendirerek diğer bir telekom firmasına geçiş yapması bu kapsamda değerlendirilir. Özellikle oyun sektöründe donanım yetersizliği, işletim sistemleri farklılıkları veya oyun ücretlerinin yüksek olmasından dolayı oluşan olumsuzluklar müşterinin kasıtlı kayıp olarak nitelendirilmesine yol açar ve oyun sektörü müşterileri gönüllü kayıp grubunda değerlendirilir (Şeker, 2016; Shaaban ve diğ., 2012). Aynı zamanda, müşterinin kendisi için daha avantajlı olduğuna inandığı şartlarda gerçekleştiği için gönüllü müşteri kaybını tahmin etmek daha zordur.

2.5.2. Gönülsüz Müşteri Kaybı

Gönülsüz kayıp, tanımlanması en kolay ve şirketin abone listesinden çıkarmaya karar verdiği müşteri grubudur (Shaaban ve diğ., 2012). Dolayısıyla, bu kategori dolandırıcılık, ödeme yapmama ve hizmeti kullanmama nedeniyle abonelikten çıkarılan müşterileri içerir. Gönüllü müşteri kaybına göre tahmin etmek daha kolaydır. Ayrıca, Hadden ve diğ. (2006) gönülsüz kayıpların genellikle bir şirketin gönüllü kayıplarının yalnızca küçük bir yüzdesini oluşturduğunu dile getirmişlerdir. Bu sebepten ötürü şirketler daha çok gönüllü müşteri kaybı analizi üzerinde durmaktadır.

2.5.3. Müşteri Koruma ve Sadakati

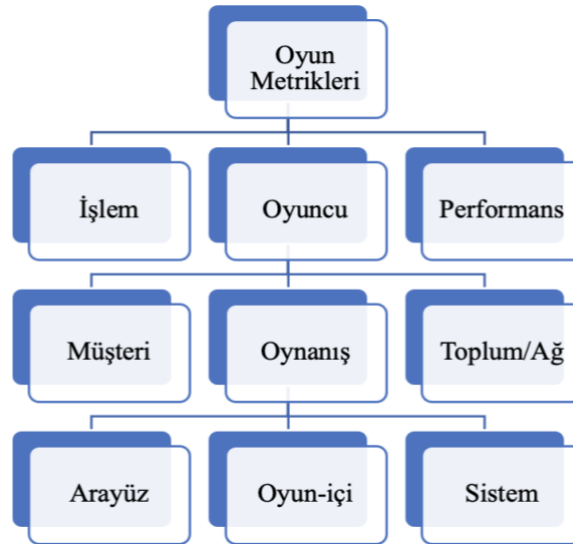
Müşteriyi koruma (customer retention), bir işletmenin mevcut müşterilerini zaman içinde elinde tutma becerisini ifade etmektedir. Müşteriyi korumak hedef bir hizmetin kullanılmaya devam etmesini ve aboneliklerin yenilemesini sağlamaktır. Ayrıca, müşteri koruma müşteri kaybının tam tersi olarak nitelendirilir. Müşteri koruma sürecinde, müşteriler çoğunlukla ürün hakkında bilgi sahibidir ve kendi çıkarlarını gözetmektedirler. Uzun vadede müşteri kaybını azaltmak için, ürünün müşteriye sağladığı değeri iyileştirmek veya maliyeti düşürmek gerekir. Bu doğrultuda işletmelerin müşteri kaybını önlemek için müşterileri ve ürünle olan ilişkilerini anlamak için bir plan hazırlaması gerekir. Bu süreçte en kritik nokta, alınan kararlarla sunulan hizmetin müşterilerin problemini çözmeye yakın olmasıdır.

Müşteriyi elde tutmanın anahtar kavramlarından bir diğeri de müşteri sadakatidir. Müşteri sadakati müşterinin tercih edilen bir ürünü/hizmeti gelecekte sürekli şekilde yeniden satın alması ve tekrar tercih etmesi olarak tanımlanmaktadır (Oliver, 1999). Kiseleva ve diğ. (2016)'a göre müşteri sadakati, bir müşterinin bir ürün veya hizmetle ilgili beklentilerinin ne kadar iyi

karşılandığının veya aşıldığının bir ölçüsüdür. Müşteriyi elde tutma oranını %5 artırmak, kârı %25-95 arasında artırabilir. Şirketin halihazırda sahip olduğu bir müşteriye satış yapması başarı oranı %60-70 iken, yeni bir müşteriye satış yapması başarı oranı %5-20'dir (Landis, 2022). Ayrıca, telekom ve finans sektöründe yeni bir müşteri kazanmanın maliyetinin mevcut müşteriyi elde tutmanın maliyetinden beş kat daha fazla olduğunu bilinmektedir. Bu istatistikler müşteri korumanın oldukça kritik ve uzmanlıkla yönetilmesi gereken bir süreç olduğunu açıklar niteliktedir.

2.5.4. Müşteri Metrikleri

Müşteri metriği, tüm müşteriler üzerinde bireysel olarak yapılan her ölçüme verilen isimdir. Metrikler özellikle güncel problemlerin çözülmesinde ve işletmelerin gerekli analizleri gerçekleştirebilmesi için önemli verilerdir. Yapay zeka algoritmalarının kullanımında veya model seçiminde katkıda bulunan en önemli yapıların bu metrikler olduğu söylenebilir. Örneğin, müşteriye sunulan ürünü ayda kaç kez kullandığı veya günlük, aylık ve yıllık kullanım süreleri bu metrikler arasında yer alır. Ayrıca bu metrikler her sektör için farklı olmakla birlikte her birinin kendine özgü şekilde analiz edilmesi gerekir.



Şekil 2.7: Oyuncu metriklerinin hiyerarşik diyagramı (Drachen ve diğ., 2013)

Oyun sektöründe müşteri davranışını ve etkileşimini izlemek ve analiz etmek için kullanılan çeşitli metrikler mevcuttur. Bu metrikler, oyun geliştiricilerinin ve yayıncılarının oyuncuların oyunla nasıl etkileşime girdiğine dair iç görü kazanmalarına ve oyuncu deneyimini iyileştirmek

için veriye dayalı kararlar almalarına yardımcı olabilir. Oyun sektörü farklı tür ve platformlara göre kategorize edildiğinden ötürü her oyun türünün kendine özgü metrikleri bulunmaktadır.

Şekil 2.7’de oyuncu metriklerinin hiyerarşik diyagramına yer verilmiştir. Müşteri metrikleri oyuncu ana grubunun alt metriği olarak karşımıza çıkmaktadır. Aşağıda yer alan metrikler çoğu platform geliştirilen oyunlar için ortak kullanılan metriklerdir. Bu ortak metriklerden bazıları açıklamalarıyla aşağıda verilmiştir. Bunlar;

- **Yükleme işlemi sayısı:** Oyuncuların oyunu toplam yükleme sayısıdır.
- **Kullanıcı Başına Günlük/Aylık/Yıllık Ortalama Kazanç (ARPU):** Oyunculardan elde edilen günlük, aylık ve yıllık ortalama kazançtır. Belirli periyotlarla değerlendirilmesinde fayda vardır.
- **Günlük/Aylık/Yıllık aktif oyuncu sayısı:** Belirli periyotlarla hesaplanan aktif oyuncu sayısıdır.
- **Günlük/Aylık/Yıllık oturum süreleri:** Belirli periyotlarda hesaplanan aktif oyuncu süreleridir.
- **Elde tutma oranı:** Belirli periyotlarla incelenen oynamaya devam eden oyuncu sayısıdır.
- **Oyun içi satın alım kayıtları:** Oyunda oyuncunun satın alma işlemine dair kayıt hareketleridir.
- **Oyun içi aksiyon kayıtları:** Oyunda oyuncunun gerçekleştirdiği her aksiyona dair kayıt hareketleridir.
- **Satın alma zamanı:** Oyuncunun satın alım yaptığı zaman etiketidir.
- **Yükleme başına maliyet (CPI):** Oyunun yayımlandıktan sonra indirme başına şirkete olan maliyetidir. Şirketler bu metriği düşük tutmayı hedeflemektedir (El-Nasr ve diğ., 2013).

Bu metrikler arasında en önemlilerinden biri müşteri Yaşam Boyu Değeri (Customer Lifetime Value) olarak bilinir (Sohrabi ve Khanlari, 2007). Mevcut ve gelecekteki değerlerin müşteri bazında hesaplanmasını sağlar. Müşteri ilişkileri yönetimi stratejisinin belirlenmesi, pazarlama kaynaklarının başarılı analizler gerçekleştirilmesi bu metriğe dayanmaktadır. Bu nedenle şirketlerden mevcut müşterilerin satın alma davranışlarını tahmin etmeye yönelik analizler gerçekleştirmesi beklenir (Castéran ve diğ., 2017). Müşteri yaşam boyu değeri metriği, işletme

ve pazarlama alanında sıklıkla kullanılan bir metriktir. Şirketler için en karlı olan müşterilerin uzun süreli ilişki kuranlar olduğu belirtilmiştir. Ayrıca, yeni müşteri edinmenin maliyeti daha yüksek olduğu bilinmekle beraber bir müşterinin kar getirisi gelecek yıllarda yapacağı birçok satın alma işlemi gerçekleştirildiğinde anlaşılabilir (Borle ve diğ., 2008).

Bir müşteriyi gelir getiren pozisyonda önemli kılan en önemli faktör nedir? Kotler ve Armstrong (2018) bu soruyu “zaman içinde şirketin müşteriyi çekme, ürün satma ve bakımını yapma maliyetini aşan kişi” olarak tanımlamaktadır. Bu aşan kısımdan dolayı oluşan fazlalık müşteri yaşam boyu değeri olarak nitelendirilir. Borle ve diğ. (2008) bir müşterinin gelecekteki nakit akışlarının bugünkü değeri olduğunu ifade etmişlerdir. Gupta ve Zeithaml (2006) ise bireysel müşterilerin geçmiş, şimdiki ve öngörülen gelecekteki nakit akışlarına dayanan değeri olduğunu ve müşteri yaşam boyu değerini oluşturabilmek için müşteriyi elde tutma ve kayıp oranlarının önemli olduğunu vurgulamıştır.

2.5.5. Müşterilerin Hareketsizliği ve Periyotları

Oyun sektörü müşteri kayıp analizinde karşılaşılan en temel problemlerden biri oyuncunun tercih ettiği platformda oynamak için yüzlerce oyuna sahip olmasıdır. Yani oyuncu, geliştirilen oyunda bir şeyi beğenmediğini hissettiğinde başka bir oyuna geçme durumu sıklıkla yaşanır. Finans ve telekom sektöründe kayıp olan müşteriler hesaplarını ve bilgilerini değiştirdiği işletmeden sildirirken oyun sektöründe durum böyle değildir. Oyuncular üyeliklerini silmezler ancak uzun süre hareketsiz kalabilirler. Bu bağlamda özellikle oyun tasarımcıları ve içerik üreticileri için ilk değerlendirilmesi gereken yapı, oyunda ilgi çekici deneyimler sunabilmek ve Oyun Akışının (Game Flow) tamamlayıcı olmasını sağlamaktır. Özellikle oyunda uygulama içi satın almalarla gelir elde etmeye dayalı bir model kullanılıyorsa, şirketin finansal başarısı büyük ölçüde hitap ettiği kitle büyüklüğüne ve sadakatine bağlıdır. Oyunlara ait müşteri ve oyun içi metriklerin analiz edildiği GameAnalytics’in iş analistleri ekibinin gerçekleştirdiği çalışmaya göre uzun süreli bir oyuncuyu elde tutma oyunun finansal başarısıyla ilişkilidir (Lovato, 2015).

Oyuncuların oynadıkları oyunları neden bıraktıklarına dair açık ve net kriterler olmadığı söylenebilir. Temelde oyundan ayrılma, arayüzün kullanılabilirliği ve basitliği, ücretsiz denemenin mevcudiyeti, ilk oynamada oyuna uyum sağlama durumlarından kaynaklanabilir (Mustač ve diğ., 2022). Oyun sektöründe bu faktörler oyun yapımcılarının fikirleri, oyunun dinamik ve mekaniği, teknolojik imkanlar ve oyuncu istekleri gibi farklı başlıklar altında

kategorize edilebilir. Oyuncuların oyunu bırakma sebeplerinden bazıları aşağıda listelenmiştir. Bunlar;

- Zaman ve para kaybı,
- Sosyal izolasyona sebep olma,
- Teknolojik cihaz eksikliği,
- İnternet bağlantısının zayıf olması,
- Oyun aralıklarının uzun olması,
- Oyun girişlerinin ve kullanıcı arayüzünün (user interface) kötü olması,
- Ücret politikası
- Oyuncunun istekleri olarak sıralanabilir.

Çalışma	Oyun	Tür	Gelir Yöntemi	Hareketsizlik Periyodu
Runge ve vd., 2014	Diamond Dash Monster World Flash (Woga)	Sosyal oyunlar (social game) - Mobil	F2P	2 hafta
Perianez vd., 2016	Age of Istharla	RPG - Mobil	F2P	10 gün
Tamassia vd., 2016	Destiny	MMO	P2P	4 hafta
Kim vd., 2017	Dodge the Mud, Undisclosed, TagPro	Sıradan oyun (casual game) - Mobil	F2P - Mobil	10 gün
Lee vd., 2018	Blade and Soul	MMORPG	F2P ve P2P	5 hafta

Şekil 2.8: Bilgisayar ve mobil platformda yayınlanan oyunlarla gerçekleştirilen çalışmalarda baz alınan inaktiflik süreleri

Bilgisayar ve mobil platform olmak üzere farklı oyunlarda kullanılan hareketsizlik periyotları Şekil 2.8’te verilmiştir. Bilgisayar kategorisindeki oyunların hareketsizlik süreleri 2 ila 5 hafta arasında değişmektedir. Mobil türüne bakıldığında ise bu sürenin 7 ila 14 gün aralığında değiştiği görülmektedir. Bu aşamada her oyun için kullanılabilecek standart bir gün yoktur, her oyunun müşteri kitlesi ve mikro etkileşimleri baz alınarak yukarıda verilen aralıklarla kayıp tahmini denenebilir.

2.6. OYUN SEKTÖRÜNDE ÖDEME TÜRLERİ VE MODELLERİ

21. yüzyılda İnternet ortamında sunulan platformların sayısında ve oluşturulan içeriklerin verisel boyutunda büyük bir artış yaşanmıştır. Oyun sektörü de zamanla bu grubun içerisinde yer bulmuştur. Özellikle ilk bölümde belirtildiği üzere MMO (çoklu devasa çevrimiçi) oyunların piyasaya çıkması bu süreci hızlandırmıştır. 1990'ların sonlarında, Nexus: The Kingdom of the Winds (1996), Ultima Online (1997), Lineage (1998) ve EverQuest (1999) gibi oyunlarda oyuncu sayılarında beklenmedik bir artış meydana gelmiştir. Özellikle Blizzard'ın kendi türünde ilk olmayan ancak gerek mekanikleri ve oynanış şekliyle oyuncuların ilgisini çeken "Warcraft: Orcs ve Humans" oyunuyla birlikte bu gelişim hızlı şekilde artarak devam etmiştir. Bu gelişimin devamını günümüzde bu tür oyunların yeni güncelleme paketleriyle de görmek mümkündür.

20. yüzyılda gerçekleşen ikinci sanayi devrimiyle birlikte kullanıma sunulan birçok kamu hizmeti vardır. Bunlar su, gaz, elektrik ve telefon hizmeti olarak sıralanabilir. 20 yüzyıl sonlarına doğru ise kablolu televizyon, cep telefonu ve internetin gelişmesiyle tüketici ile hizmet veren arasındaki tekrarlanan ödeme ilişkileri ortaya çıkmıştır. Bu ilişkiler abonelik sistemi olarak adlandırılmıştır. Abonelikler yalnızca periyodik olarak ödenen sabit ücret uygulanmamış, tüketici ve tedarikçi arasında üç tür ödeme modeliyle detaylandırılmıştır (Lejeune, 2001).

- **Yinelenen ödemeler:** Her hizmet dönemi için gerçekleştirilen aynı tutarda sabit ödemelerdir.
- **Kullanıma dayalı ödemeler:** Bazı ölçü birimlerine göre kullanılan hizmet miktarı için yapılan ödemelerdir.
- **Tek seferlik ödemeler:** Genellikle kurulum için ücret alınmasının yanı sıra hizmete yönelik geçici (tekrar etmeyen) yükseltmeler veya bir kerelik (uygulama içi) satın alımlar için alınan ödemelerdir.

Güncel hayatta kullandığımız hizmetler (oyun satışı, müzik uygulaması indirilmesi, telekom ve dergi abonelikleri arasında tercih yapma vb.) arasında birçok alternatif mevcuttur. Hizmetler bazen peşin (her hizmet döneminin başında) ve bazen de gecikmeli olarak (hizmet sağlandıktan sonra) ödenir. Ancak tüm bu hizmetlerde ortak olan, tüketici ile hizmet sağlayıcı arasındaki süregelen ilişkidir. Ancak oyun sektöründe, kullanıcı oyun satın almayı oyun satış platformu

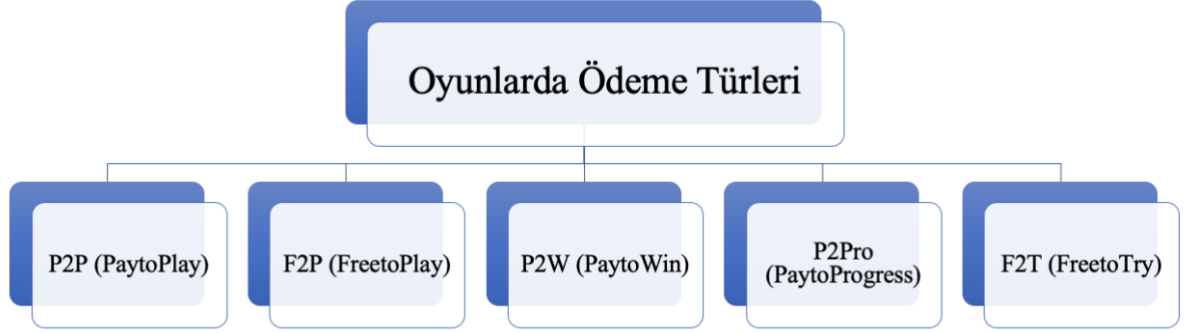
üzerinden yapar. Çünkü bu oyunlar genellikle farklı sitelerde satılmaktan ziyade oyun satış platformları üzerinden gerçekleşir. Örneğin: Steam, Epic Store, XBox Game Pass, Kinguin ve G2A olarak sıralanabilir.

Oyunlar için platform seçiminde karar aşaması oyun geliştirme ve oyun pazarlamasının önemli faktörlerinden biridir. Bu platformların hepsi farklı özelliklere ve her biri kendi içinde farklı oyuncu gruplarına sahiptir. Son yıllarda, iOS ve Android işletim sistemine sahip akıllı mobil cihazların kullanım sayılarındaki artış sebebiyle mobil oyun piyasasında büyük bir artış olmuştur. Bunun yanında konsol platformunda AA ve AAA olarak adlandırılan büyük bütçeli oyunların da boy göstermesiyle birlikte oyun endüstrisi, eğlence endüstrisinde büyük bir pay sahibi olmuştur. AA ve AAA kavramları oyun endüstrisinde yapılan iş ve ürün bütçesi için kullanılan bir terimdir.

Konsol oyun pazarı günümüzde Microsoft (XBox), Sony (Playstation) ve Nintendo (Wii) cihazlarına çıkarılan oyunlar üzerinde yoğunlaşmıştır. Konsol cihazları, bilgisayar veya akıllı telefonlardan mobilité ve pratik kullanım bakımından ayrıldığından ötürü, salt hedefi oyun oynatma üzerine geliştirilmiş cihazlardır. Cihazların bu özelliği de konsolların bu tarz bir pazarlama politikasına doğru yönelmesinde etkili olmuştur. Oyun geliştirme sürecinin en zor taraflarından biri, oyuncu davranışını tahmin etmek üzere kullanılan ve şirkete pahalıya mal olan prototipleme sürecidir.

Günümüzde birçok oyun firması, gelirlerini tek sefer ödemeli modeli kullanmak yerine kademeli olarak yayınlamayı tercih etmiştir. Bu süreçte oyun içi reklamlardan ve satın almalarından gelir elde etmişlerdir (Pedersen ve diğ., 2010). 2019 yılında geliştiriciler mobil oyunlardan 10 milyar \$, PC 3.3 milyar \$ ve konsol oyunlarında ise 1.1 milyar \$ dolar gelir sağlamıştır (Statista, 2022). Oyun firmalarının AAA türünde oyunlar geliştirmeye yönelik yatırımların yetersiz olması veya bütçe bulunmaması gibi sebeplerle mobil oyunların sektörde daha hızlı tüketilmesine ve gelişmesine neden olmuştur. Özellikle oyun ve grafik motorlarının son yıllarda geliştiricilere ücretsiz olarak sunulması hem geliştiricilerin hem de firmaların uyguladığı yeni bir pazarlama politikası olarak ön plana çıkmaktadır. Yüksek donanımsal özellikli bilgisayarlara sahip olmadan ve çalışanlara çok fazla bütçe ayırmadan reklam ve uygulama içi satın alım üzerinden gelir elde etmeye yönelik pazarlama politikası günümüzde çok fazla uygulanabilir hale gelmiştir. Ancak her oyunun kendi türünde kendine özgü bir

oyuncu kitlesi olduğunu belirtmek gerekir. Bu doğrultuda oyun sektöründe ödeme türüne göre oyunlar Şekil 2.9’te şema olarak verilmiştir.



Şekil 2.9: Ödeme türüne göre oyunların sınıflandırılması

Ödeme türüne göre oyunlar P2P, F2P, P2W, P2Pro ve F2T olmak üzere beş grup olarak kategorize edilmektedir. Bu modellerin arasında en popüler olanları F2P olmakla birlikte halen P2P modelini kullanan oyunların da olduğu görülmektedir. Bahsedilen ödeme türleri aşağıda detaylı şekilde açıklanmıştır (El-Nasr ve diğ., 2013).

- **F2P (Free to Play):** Oyunun pazarlama sürecinde farklı kitleler tarafından oynanabilmesi için oyuncuya oyunun ücretsiz olarak sunulmasıdır. Oyun oyuncuya ücretsiz olarak sunulmakta ancak reklamlar, mikro-ödemeler, indirilebilir içerikler, eşya ve kostüm güncellemeleri vb. aracılığıyla gelir elde edilmektedir. Paladins, Dota II, League of Legends ve Smite örnek olarak verilebilir.
- **P2P (Pay to Play):** Oyun içinde belirli ödemelerin gerçekleştiği, oyunların kendi aralarında eşya takas edebildiği ve oyuncuya aylık veya yıllık bazda maliyetlerin yer aldığı oyunlara verilen isimdir. World of Warcraft ve Overwatch bu kategoriye örnektir.
- **P2W (Pay to Win):** Oyuncuların oyun içinde para harcayarak sadece kazanmaya odaklı eşyaları veya güçlendirmeleri satın aldığı ödeme yöntemidir. 2010 yılında yayınlanan ve halen oynanmakta olan FIFA Ultimate Team adlı oyunda oyuncular gerçek para karşılığında satın alarak kadroya dahil edilmektedir.
- **P2Pro (Pay to Progress):** Oyun içinde oyuncuların gerçek para ödeyerek oyunda yönettiği karakterler için tecrübe puanı, yetenek puanı ve sürekli güçlendirmeye sahip olduğu ve karakter geliştirmeye devam etmelerine olanak tanıyan modeldir.

- **F2T (Free to Try):** Para harcamadan oynanabilen ancak tamamlanamayan oyunlar ise bu modeli kullanmaktadır. Oyunu kısıtlı versiyonda oynatıp daha sonra satışını gerçekleştirmek üzere kullanıcılara teklif götüren satın alma modelidir.

Bu modellerin kullanımında oyun şirketlerinin temel hedefi, oyuncuyu aktif olarak oyunun içinde tutarak ürünü kolay ve erişilebilir hale getirmektir. Özellikle mobil platformlarda ücretsiz olarak sunulan oyunlarda F2P modelinin kullanımı artış göstermektedir. Bu konuda en iyi örneklerden biri “HeartStone”dur. HeartStone, çevrimiçi iki oyuncunun kartlar aracılığıyla karşılıklı hamleler yaptığı çevrimiçi bir kart oyunudur. Aynı zamanda ücretsiz olarak indirilebilir ve oynanabilir. Ancak bu noktada, farklı kart desteleriyle farklı rakiplere karşı güçlü olabilmek için ücret ödeyerek kart desteleri satın alınmaktadır. Ancak bu şart olmamakla birlikte oyuncunun oyun içinde daha aktif ve farklı seviyeleri görebilmesi için ücretsiz adıyla sunulan bir pazarlama politikasıdır.

F2P modeli masaüstü ve konsol platformuna hitap eden bir model olmamakla birlikte mobil platformda da sıklıkla kullanıldığını bilinmektedir. Örneğin, Supercell firması tarafından geliştirilen Clash of Clans 2012 yılında piyasaya sürülmüş ücretsiz strateji oyunu olmakla birlikte mobil kategoride indirme rekorunu kıırarak bu alanda adından sıkça söz ettirmiştir. Buna benzer oyunlarda farklı eşyalar alarak veya hesaplar arası takas yaparak farklı geliştirmeler kullanılabilir. Candy Crush F2P modelini kullanan en popüler oyunlardan bir diğeridir. Oyun, oyunculara farklı bölümleri geçmek için günlük oyun oynama hakkı sunmakta ve bu hak dolduğunda ise puan satın alarak yeni bölümleri geçmek için ekstra oyun hakkı sunmaktadır (Yang ve diğ., 2020). MOBA oyun türünde ise League of Legends bu kapsamda örnek verilebilir. Bu oyunda karakterler oyun içi kazanılan puanlarla açılabilirdiği gibi oyunda sürekli oynanarak kazanılan özel puanlar aracılığıyla da içeriklerin kilitleri kaldırılabilir. Oyuncuların, oyun içi yönetmekte olduğu karaktere ait farklı kostümleri alabilmesi için oyun içi ücret ödemesi gerekir. Bu bakımdan oyun ücretsiz olarak sunulurken, oyun içi kostüm satışı ek ücrete tabii olarak gerçekleştirilmektedir.

F2P modeli, şirketlerin oyun lisansı için tek bir peşin ücret talep etmek yerine, ödeme yapan oyunculardan ömürleri boyunca abonelikler, reklamlar veya mikro etkileşim yoluyla para kazanmayı hedefleyerek çevrimiçi (online) oyun pazarında en çok kullanılan model konumundadır (Castro ve Tsuzuki, 2015). Ancak, şirketleri ve oyuncuları birbirine bağlayan

aboneliklerin zayıf olduğu F2P oyunlarda müşteri ve şirket bağlantısı oldukça zayıftır. Bunun sebepleri yukarıda oyuncunun olası oyunu bırakma sebepleri listesinde bahsedilmiştir.

2.7. OYUN SEKTÖRÜNDE İŞ MODELLERİ

Oyun geliştiricileri sektörde her yıl yüzlerce oyun piyasaya sürerken, ücretsiz kullanıma sunulan oyun motorları sayesinde sektöründe rekabet faktörü ön plana çıkmıştır. Oyuncuların, keyif alarak oynadığı oyun sayısı önceki dönemlere göre artarken, oyun geliştiricileri piyasaya sundukları oyunlar için tek seferlik ödeme modelini kullanmayı tercih etmemişlerdir. Ancak bu modeli uygulayanlar ise belirli bir ciroya ulaşır sürekli yeni proje için girişimde bulunmuşlardır. Bu da sürekli iş gücü, teçhizat ve proje maliyeti olarak şirketlere geri dönmektedir. Bu sebepten ötürü oyun geliştiricileri oyunları ücretsiz olarak sunmak ve gelir elde etmek için farklı para kazanma politikaları ve modelleri uygulamışlardır. Bu doğrultuda bu başlığın devamında oyun sektöründe geçmişten günümüze gelir elde etme modellerinden detaylı şekilde bahsedilmiştir.

2.7.1. Geçmiş Dönem Modelleri

Toplumun hizmetine sunulan ilk oyunlar, 1970'lerden 1980'lere oyun salonlarında oyuncularla buluşmuştur. Bu süre zarfında, oyunlarının donanım gereksinimlerini her oyun kendi ayrı fiziksel cihazında barındırılmıştır. Bu durumdan ötürü işletmeler, tüketicilere Pacman'dan Galaga'ya kadar günümüz standartlarına göre sınırlı sayıda oyun seçeneği sunmak için fiziksel lokasyon ve yüksek maliyetleri karşılamak zorunda kalmıştır (Perrotta, 2020). Bu dönemde P2P iş modelinde oyun makineleri jetonla çalışmış ve oyun içinde can (health) karşılığında oyuncuların jeton atmasını gerektiren ilk mikro etkileşimlerin ortaya çıkmasına ortam sağlanmıştır. Bu iş modeli, 1990 yılına kadar 3,5 milyar doların üzerinde gelir elde eden Namco'nun Pacman'ı ve 1982'de 2,7 milyar doların üzerinde gelir elde eden Taito'nun Space Invaders'ı da dahil olmak üzere “büyük kazanç” olarak nitelendirilebilecek bir başarıya imza atmıştır. Bu dönemde şirketlerin jetonlar aracılığıyla fiziksel bir lokasyonda oyun hizmeti sunması P2P modelini kullanmaya ortam hazırlamıştır. Çünkü, fiziksel oyun cihazlarının toplumun büyük kesiminin özelinde olmaması da bu modelin uygulanmasında etkili olmuştur. Zamanla toplumun oyun oynayabileceği cihazları satın almalarıyla birlikte, oyun satışı modellerinde de değişiklik yaşanmıştır. Teoride farklı modeller ortaya atılmış ve İnternet

üzerinden çoklu oyuncularla oynanan oyunların artmasıyla birlikte farklı modeller uygulamada kullanılmıştır.

2.7.2. Güncel Modeller

Yeni nesil konsollar (Xbox One veya PS5), oyun bilgisayarları ve akıllı mobil cihazlar günümüzde popüler olan başlıca oyun platformlarını oluşturmaktadır. Diğer endüstrilerde olduğu gibi, oyun endüstrisi de dijital çağa geçişten büyük ölçüde etkilenmiş ve geliştiricilerin oyunlarını dijital bir kütüphane kullanarak oyunculara sunmalarına ortam sağlamıştır. Bu kütüphanelerden hizmet alan oyuncular, oyunun fiziksel kopyalarına ihtiyaç duymamakla birlikte, bulunduğu bölgenin oyun satış platformu aracılığıyla bu oyunlara ve en son çıkan sürümlerine de erişim sağlamışlardır. Bu kütüphanelerin oyuncuya sağladığı faydaların ötesinde, yayıncıların ve geliştiricilerin perakende satışını da engelleyerek oyunu İnternet üzerinden indirme (download) imkanı sunmuştur. Bu da paketleme maliyetlerini ortadan kaldırmış ve oyun satışlarından daha fazla gelir elde etmelerini sağlamıştır.

Yüksek bütçeye sahip konsol üreticileri, Nintendo eShop'tan PlayStation Store'a kadar oyunların indirilebileceği dijital kütüphanelere yatırım yapmış, bu platformlarda yapılan satışlarla oyun sektöründe satış ve pazarlama zincirinin bir kısmına hakim olmuşlardır. Hızlı gelişen dijital endüstride oyun satın alma süreci aynı zamanda iş modellerini de etkilemiş ve farklı kazanç elde etme modellerinin oyunlarda uygulanabilir olmasını sağlamıştır. Güncel olarak halen kullanılan bu satış modellerinde belirli unsurlar ön plana çıkartılmış ve sektöre kazanç sağlanması daha da desteklenmiştir. Bu unsurlar ve güncel modellere uygulanan alt başlıklar aşağıda detaylı şekilde bahsedilmiştir.

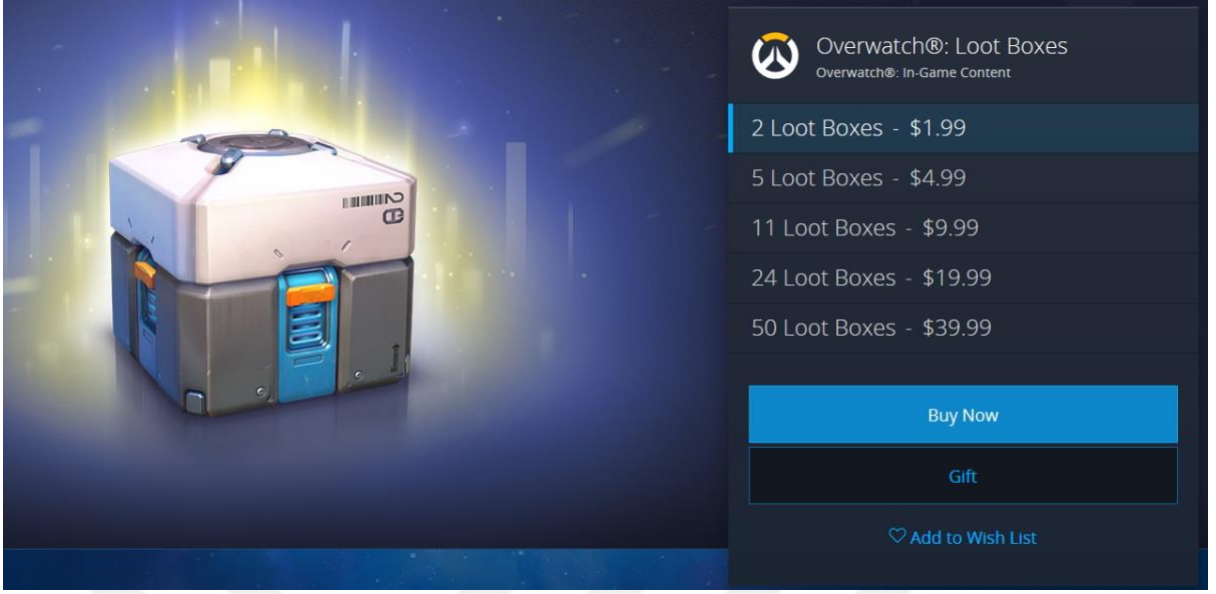
Activision Blizzard'ın Battle.net, Electronic Art'ın Origin ve Valve'nin Steam gibi oyun yayıncılarının oyun satışlarından elde ettikleri gelir artış gösterirken, Microsoft ve Sony gibi şirketler maliyetleri düşürmek için dijital satış platformlarında yerini almıştır. Valve'nin Steam'ine benzer platformlar, kendi oyunlarının dağıtımını kontrol etmenin ötesinde, Apple App Store veya Google Play Store gibi diğer geliştiricilerin oyunları için dijital bir pazar yeri haline gelerek yeni para kazanma fırsatları sunmuşlardır. Steam, %30'luk bir gelir payı karşılığında, diğer oyun geliştiricilerine geniş oyuncu tabanlarına erişim sağlayarak kullanıcı sadakatine önem vermiş, platformlar arasında en öne çıkanların başında yer almıştır. Bu model,

Valve'in yeni bir gelir akışı sağlanmasının yanı sıra kendi oyun satışlarını arttıran ekosisteme daha fazla oyuncunun çekilmesini sağlamıştır.

Steam sadece bir oyun satış platformu olduğundan ötürü oyuncu ekosistemini tamamen elinde tutması mümkün olmamıştır. Apple, Microsoft ve Sony gibi hem oyun hem de cihaz üreticilerinin de satış platformlarına dahil olmasıyla birlikte Steam bir nebze müşteri kaybı yaşamıştır ancak halen popüler satış platformların başında yer almaktadır. Örneğin, Microsoft Gamepass, Sony Playstation Plus ile oyunların satışını gerçekleştirirken, Apple ve Google gibi firmalar market uygulamalarını kullanarak oyuncuların hizmetine sunmuştur. Microsoft ve Sony şirketlerinin tez çalışmasının ilk bölümünde bahsedildiği üzere Xbox ve Playstation cihazlarının üreticisi olduğu bilinmektedir. Bu durumda kendi ürettikleri cihazlara özel oyunları kendi platformları üzerinde satışa sunmuş ve aylık düşük bir miktar karşılığında oyunculara oyun kütüphanelerini kullanma fırsatı vermiştir. Özellikle aylık düşük sabit ücret karşılığında platform kütüphanesinde yer alan bütün oyunları farklı platformlarda oynama imkanı sunması diğer modellerinin de gözden geçirilmesine sebep olmuştur. Mikroetkileşim, abonelik ve freemium güncel gelir elde etme modelleri arasında yer almaktadır ve aşağıda detaylı şekilde açıklanmıştır.

2.7.2.1. Mikroetkileşim (Microtransactions)

Mikro etkileşim, oyuncuların oyun içinde yaptıkları küçük veya büyük oranlı uygulama içi satın alımlardır. Satın alımlar, bir karakterin görünümünü değiştiren eşyaları, silahları, özel karakterleri veya oyundaki zorlukları tamamlamak için ekstra canları içerebilir. Oyuncular ayrıca, oyunda ilerlemek için ihtiyaç duydukları öğeleri veya yükseltmeleri almak için oyun içi market ekranında sunulan ganimetleri ve oyun içeriklerini elde etmek için mikro etkileşimi kullanabilirler (Evers ve diğ., 2015;Tomic, 2018). Şekil 2.10'da Overwatch oyununda oyunculara sunulan bir lootbox (şans kutusu) teklifi gösterilmektedir. Bu teklif oyun içinde karaktere eşya, görünüm, kostüm ve değişiklikler içermekle birlikte belirli bir ücret karşılığında oyuncuya sunulmaktadır.



Şekil 2.10: Overwatch oyununa ait bir şans kutusu (loot box) teklifinde mikro etkileşim örneği

Dijital oyunlarda ilk mikroetkileşim modelini 2006 yılında Bethesda Oyun Stüdyosunun “The Elder Scrolls IV: Oblivion” adlı oyununda at zırhının 2,50 dolara satmasıyla kullanılmıştır. Tek bir oyun içi eşya için 2,50 dolar yüksek bir ücret olarak nitelendirilmiş ve birçok oyuncu olumsuz yönde tepki göstermiştir. Bu tepkilere rağmen, eşya oyuncular arasında popülerlik kazanarak bu oyunda en çok indirilebilir içerikler arasında yer almayı başarmıştır. Bu doğrultuda, oyun içinde eşya satışını şirketin yararına olmasını sağlayan ve mikro işlemler alt unsurlarından biri olan indirilebilir içerik diğer ismiyle DLC (downloadable content) karşımıza çıkmaktadır. Örneğin, Fortnite piyasaya sürülmesinden on ay sonra 1,2 milyar dolar gelir elde etmiştir. Oynaması ücretsiz bir oyun olmasına rağmen, Fortnite’ın gelirleri esas olarak oyuncuların kostümler ve görünümüler için uygulama içi satın alımlar yaptığı mikro işlemlerden oluşmaktadır. Oyuncular bu eklentiler olmadan da oyunu oynamaya devam edebilirken, oyuncuların çoğu ek eşyalar ve öğeler için ödeme yapmayı tercih etmektedir. Bu doğrultuda oyun sektöründe para kazanma modellerinin artık değiştiği ve doğrudan abonelik veya tek sefer ödeme teklifleri sunarak oyuncuyu elde tutmanın zor olduğu görülmektedir. Oyun sektörünün sürekli rekabetin olduğu ve farklı demografik özelliklere sahip gruplara hitap eden bir sektör olması da bunda oldukça etkilidir.

2.7.2.2. **Abonelik**

Oyun sektöründe geliştirici ve yayıncı olmak üzere farklı gruplar bulunur. Şirketler bazen hem geliştirici hem de yayıncı olabilir. Ancak genellikle oyunu geliştiren firma bir yayıncı ile çalışmak zorundadır. Çünkü, geliştirilen oyunun piyasaya hakim bir yayıncı tarafından oyunculara sunulması daha büyük bir etki oluşturabilir (Oh ve Ryu, 2007). Bu sebepten ötürü geliştirici ve yayıncı kavramlarının iyi anlaşılması kilit noktadır. Sadece oyun sektörüne has olmamakla birlikte film, dizi ve müzik gibi dijital içerik üreten şirketler için de durum benzerdir. Dijital içerikler yapımcı firmalar tarafından tamamlanarak dijital yayın platformlarında sunulur. Burada bahsi geçen firmalar bir bakıma yayıncı konumundadır.

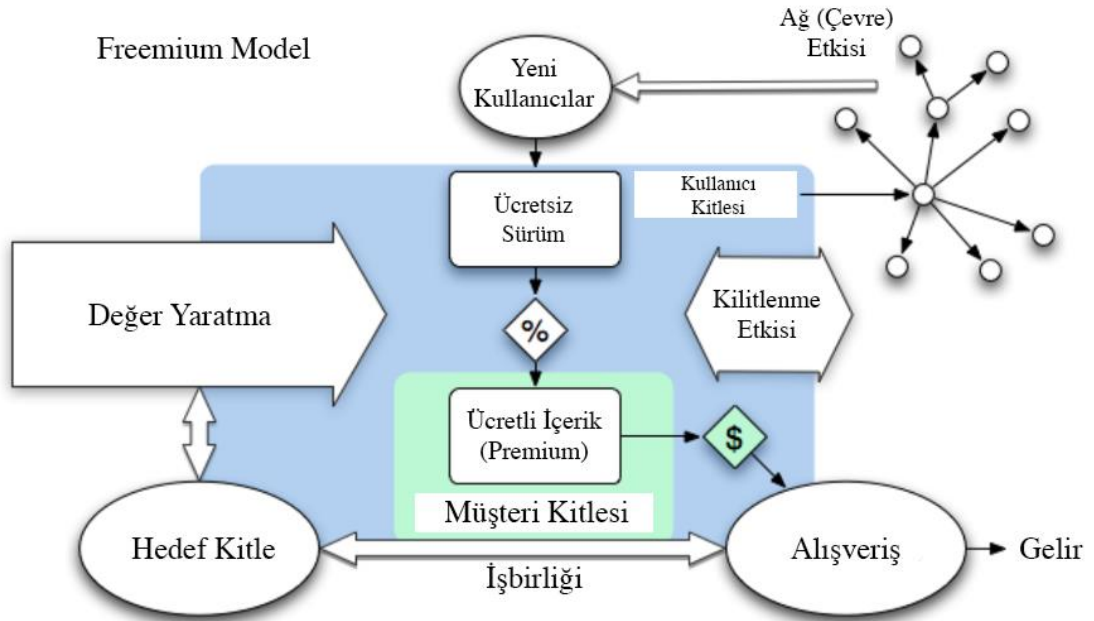
Sony ve Microsoft gibi konsol üreticileri, son yıllarda PlayStation Now ve Xbox Game Pass gibi abonelik hizmetleri oyuncuların hizmetine sunmaktadır. Özellikle konsol platformunda oyunların çok pahalı olması, oyuncuların abonelik sistemine dahil olmasını da kolaylaştırmıştır. Bu hizmetler, oyunculara oynayabilecekleri geniş oyun kütüphanesi sunmasının yanında, oyunculara sağladığı oyun çeşitliliğiyle oyuncuları sistemde tutmaya devam etmektedir. Tipik olarak bu abonelik hizmetleri, oyun kütüphanelerindeki çoğunlukla daha eski oyunlara erişim sunmaktadır. Bu kütüphanelerde genellikle yeni çıkan bir oyun serisine ait önceki sürümler yer alabilir. Ancak oyuncular, bir oyunu piyasaya sürüldükten hemen sonra oynamak isterse, ek bir ödeme yapmak zorundadır. Örneğin, World of Warcraft gibi oyuncuların oyunu oynamak için aylık bir ücret ödediği ve en güncel pakete sahip olması için yaklaşık 40 \$ karşılığında satın aldığı benzersiz abonelik örnekleri de mevcuttur.

2.7.2.3. **Freemium İş Modeli**

“Freemium” terimi ilk olarak 2006 yılında Union Square Ventures'tan Fred Wilson tarafından önerilmiştir. Freemium iş modeli, bir iş modelinin temel işlevlere sahip ücretsiz ürünler sağladığı ve kullanıcıları ücretsiz hizmetler aracılığıyla Premium modele döndürmeye çalıştığı iş modelidir. Fred Wilson Freemium iş modeli çalışma prensibi içinse şunları ifade etmiştir:

Hizmetinizi müşteriler aracılığıyla dilden dile duyurun, birbirlerine yönlendirin ve daha sonra aramaya yönelik pazarlama tekniklerini kullanarak büyük bir kitleye kullanılabilir ücretsiz şekilde dağıtın, sonrasında toplanan müşterileri cazip bir ücret karşılığında gelişmiş sürüme yükseltin (Wilson, 2006).

İnternet kullanıcılara birçok hizmeti ücretsiz olarak sunmaktadır. Özellikle kullanılan hizmetin birçok alternatifinin olması interneti bu konuda mutlak güç kılan özelliklerden biridir. Tüketiciler zaman içinde internet üzerinden ücretsiz hizmet almaya alıştıklarından dolayı şirketler için farklı pazarlama stratejileri kullanma gerekliliği ortaya çıkmıştır. Anderson (2009) tarafından gerçekleştirilen bir araştırmada kullanıcılar, İnternet üzerinden satın alınan ürün veya hizmetin ücreti ne kadar düşük olursa olsun maliyet kavramını her daim önemli kriterlerden biri olduğunu dile getirmiştir. Makul fiyata sunulan bir ürünün İnternet ortamında ücretsiz olarak edinilmesini öngören kullanıcıların farklı yollardan bu ürüne veya hizmete ulaşmak istemeleri ise son derece doğaldır. İşte bu noktada freemium iş modeli kullanıcıyı hem kazanma hem de ayrıcalıklı (premium) hizmeti etkin olarak uyarlama konusunda ön plana çıkmaktadır (Anderson, 2009).



Şekil 2.11: Freemium modeli yaşamsal döngü şeması (Reime, 2011)

Şekil 2.11’de görüldüğü üzere, ücretsiz sürüm kullanıcıları sisteme çekmek için kullanılırken ayrıcalıklı sürüm hizmeti kullanıcıları yükseltmeye (upgrade) ve ödeme yapmaya ikna etmek için kullanılmaktadır. Sosyal etkiden ötürü müşteriler sisteme dahil olarak yeni kullanıcı olarak nitelendirilir ve daha sonra bu müşterilere ücretsiz sürüm sunulur. Hemen ardından belirli sınırlamaların olduğu bir metin ve mesaj kullanıcıya iletilir. Ardından müşteriye belirli

dönemlerde yeni teklifler sunularak ayrıcalıklı sisteme dahil olması beklenir. Hedef kitle ile yapılan satın almaların arasında iş birliği bulunurken, hedef kitle aynı zamanda değer yaratma aşamasında tüm sistemin girdilerini etkiler. Hesabı ayrıcalıklı sürüme yükselten müşteriler elde tutma etkisini devam ettirmek için birtakım aksiyonlar uygulanır. Bu süreç, hizmeti sürekli kullanıma teşvik etmek için eylemler planlanmasını içerir. Kilitlenme etkisi müşteri ilişkileri yönetiminde müşteri sadakati altında değerlendirilirken, şirket ayrılmaya yönelik müşterilere özel teklifler ve güncel modelde olmayan sistem seçenekleri sunabilir.

Freemium iş modeli, son yıllarda dijital sektörde yaygın olarak benimsenen iş modellerinin en iyi türlerinden biridir. Aynı zamanda son yıllarda akademik dünyada yaygın olarak tartışılan önemli bir konu haline gelerek (Anderson, 2009; Lyons ve diğ., 2012) tüm oyun ve yazılım şirketlerinin ortak odak noktası olmuştur. Bu noktada işletmeler arasındaki rekabet, ürünlerin değil iş modellerinin rekabetine dayanmaktadır. Freemium iş modeli yalnızca oyun sektöründe kullanılmayıp, günümüzde farklı sektörlerde aktif olarak kullanılmaktadır. Bu sektörler örnekleriyle birlikte Tablo 2.1’de verilmiştir.

Tablo 2.1: Farklı sektörlerde freemium iş modelini kullanan uygulamalar

Sektör	Uygulama İsimleri
Müzik	Spotify, YouTube Music, Deezer
Veri Depolama	Dropbox, Google Drive, Apple iCloud
Dijital İçerik Üreticileri	YouTube, Netflix, Disney Plus, Amazon Prime Video
Mobil Uygulama	Google Play, AppStore
Planlama	Evernote
Blog İçerik Üretimi	WordPress

İşletmeler, para kazanmanın sürekli hale dönüştürülmesini hedeflediği bu tür modellerde sürekli güncellenebilirliğe dikkat çekmişlerdir. Özellikle teknolojinin hızlı şekilde hayatımıza girmesiyle birlikte bu modelin göz ardı edilmesinin, yeni iş modellerinin kullanılmasının şirketlerin zararına olacağı aşikardır. Bu model, aynı zamanda inovasyon ve yenilik kavramlarının da önemini bir kat daha artırmaktadır. Şirketler, iş dünyasına yeni bir ürün sürebilir veya mevcut bir ürünün yeni sürümünü çıkarabilir. Piyasa sürülmesi düşünülen ürün başarıyla veya başarısızlıkla sonuçlanabilir. Birkaç dönem öncesine bakıldığında iş dünyasında

içeriği ücretsiz olarak kullanıcıya sunma modelleri halihazırda kullanırken, aynı zamanda Freemium iş modeli tamamı ile bir yenilik olarak ortaya çıkmamıştır. Güncel modellerde minimal değişiklikler yapılarak geliştirilmiştir. Freemium iş modeli çevrimiçi hizmet sunan şirketler arasında en yaygın olarak kullanılanı olup Web 2.0 teknolojisiyle birlikte gelen temel iş modellerinden biri haline gelmiştir (Reime, 2011).

Freemium iş modeli internet üzerinden değil klasik yazılım dünyasında ortaya çıkmış bir modeldir (Huang, 2016). Teknoloji şirketleri çevrimiçi çalışma zorunluluğu olmayan yazılımları kullanıcılara ücretsiz olarak açmış ve bu kapsamda gelişmiş sürümü ancak satın alma opsiyonu olarak sunmuştur. Örneğin; Kaspersky ağ güvenliği konusunda cihazları zararlı yazılımlardan korumak amacıyla geliştirilen anti virüs yazılımı buna en önemli örneklerdendir. Freemium modelinin içeriğinin yüzde 90-95'i ücretsizdir ve yalnızca İçeriğin yüzde 5-10'u ücrete dayalıdır. Son yıllarda güncel olarak kullanılan çevrimiçi müzik dinleme uygulaması da bu kapsamda verilebilecek en iyi örnekler arasındadır. Spotify kullanıcılarına günlük sadece altı müzik değiştirme hakkı sunmakta ve dinlenen her beş müzikte bir üç reklam vermesi bu iş modeli kapsamında değerlendirilir. Birçok kullanıcının tercih ettiği YouTube ise müzik hizmetini başlatarak ücretsiz olarak sunduğu modelde müziği dinlerken tuş kilidinin kapatılmaması ve cihazlara indirmeye izin vermemesi bir diğer örnek olarak karşımıza çıkar. Witell ve Löfgren (2013) ücretsiz hizmetin bir iş modelinde inovasyon için kritik öneme sahip olduğunu öne sürmüştür. Ayrıca, uygulamada kullanıcıya hizmetin ücretsiz olarak sunulmasının kilit nokta olduğunun altını çizmektedir. Ancak bu kapsamda uygulamayı ücretli olarak kullanan kullanıcılar aynı zamanda ücretsiz olarak kullananların da maliyetini karşılamış olmaktadır. Kastalli ve Van Looy (2013) ücretsiz sunulan hizmetin ürün satışını önemli ölçüde etkilediğini dile getirmiştir.

Birçok şirket, iş modeli yeniliği için freemium iş modelini güncel olarak kullanmaktadır. Bilişim sektöründe yer alan firmalar arasında özellikle oyun endüstrisinde, daha fazla potansiyel kullanıcı çekmek ve freemium iş modelini kullanarak rekabet avantajları oluşturmak için internet pazarlamasının önemli olduğu vurgusu yapılmaktadır. Freemium iş modeli şirketin hedefleri doğrultusunda oyuncuların hizmetine sunulabilirse ve oyuncular geliştirilen modeli benimserlerse freemium iş modelinin çevrimiçi endüstriye büyük fayda sağlayacağı düşünülmektedir.

Özellikle WoW (World of Warcraft), Everquest, Guild Wars, Starwars: Old Republic ve Eve Online gibi oyunlarda çıkış yaptığı ilk dönemlerde “PaytoPlay” şeklinde bir satış politikası kullanmaktaydı. Yani kullanıcı aylık, altı aylık veya yıllık abonelik ödeyerek sadece oyunu satın alım yapabiliyordu. Ayrıca oyuncular oyunun sunucularında dahil olabilmek için bile Premium hesaba geçmek zorunda ve ödeme yapmak zorundaydı. Ancak, oyun sektöründe farklı oyunların kendisini göstermesiyle hem oyun endüstrisinde rekabet faktörü ortaya çıkmış hem de oyuncuların istekleri bir bakıma değişmiştir. Değişen oyuncu isteklerine yönelik birtakım pazarlama stratejileri de boy göstermiştir. Oyunların popülerliğinin artış gösterdiği yıllarda Premium sistemi aktif olarak kullanıyorken günümüzde artık Freemium modele geçişin olduğu belirgin olarak kendini göstermektedir (CaliberAdvisors, 2017). Ek olarak indirilebilir dosyaların yerine artık Freemium iş modelinde uygulama içi ürün, kişiselleştirme ve reklam gibi yeni yapılar sisteme dahil olmuştur.

2.8. MÜŞTERİ KAYIP ANALİZİ

Müşteri davranışı ve ihtiyaçları hakkında anlamlı bilgilere sahip olmak, müşteri odaklı pazardaki ihtiyaçlardan biridir. Yeni hizmetler ve ürünler sunarak müşterilerin çıkarlarını koruyabilmek müşteri-şirket ilişkilerini sürdürmeye yardımcı olur. Şirketlerin içinde bulunduğu rekabet ortamında müşteri ihtiyaçlarını anlayabilmek aynı zamanda bunu karşılayabilmek ve memnuniyeti yüksek tutabilmek için, müşteriyi elde tutmayı arttırmak adına yeni pazarlama stratejileri geliştirmenin önemine dikkat çekilmektedir (Liu ve Shih, 2005). Canning (1982), müşterilere daha fazla satış yapmanın artık karlı bir satış stratejisi olmadığını ve sürekli olarak daha rekabetçi büyüyen bir pazar yerinin satış kaynaklarının en verimli kullanımına odaklanan bir yaklaşım gerektirdiğini belirtmektedir.

İşletmeler, şirket-müşteri ilişkisini iyi kuramaz ve müşterilerin ihtiyaçlarını doğru bir şekilde analiz edemez ise müşteri karmaşası (consumer confusion) durumu oluşur. Müşteri karmaşası, günümüzde şirketlerin müşteri davranışlarını anlama ve çözüm bulma konusunda gecikmeleri nedeniyle karşılaştıkları büyük bir sorundur (Mitchell ve diğ.,2005). Oyun sektöründe bu durumlarla karşılaşmak olasıdır. Çünkü, dijital oyunlar insan yapımı ve eğlence amaçlı oluşturulan ancak içinde kompleks bilgisayar grafiklerinin ve kod yığınlarının çalıştığı elektronik ürünlerdir. Bazen şirketin elinde olmadan birtakım hatalar (bug) oyunda yer alabilir veya oyuncular sunuculara bağlanmakta zorluk çekebilir. Bu tür sorunlar oyun türüne ve platformuna göre farklılık gösterebilir. Şirketlerin müşteri karmaşası durumuna çözüm

üretmeleri gerekmekte olup, oyun pazarındaki tüm işletmeler, şirketin gelir rakamlarını ve dolayısıyla politika kararlarını etkileyebileceği için müşteri kaybı ile ilgili analizler yapma ihtiyacındadır. Müşteri Kayıp Analizi ile şirketler, müşterilerin geçmiş davranışlarını analiz ederek kaybın gerçek nedenini öğrenmeye çalışır. Aynı zamanda müşteri kaybının gelecekte hangi müşteriler tarafından gerçekleşip gerçekleşmeyeceğine dair analizler yapmaya yardımcı olur. Kayıp analizinin farklı sektörlerde kullanımının temel amacı, müşteri ihtiyaçlarını belirlemek ve müşteriyi korumak değil, aynı zamanda içinde bulunduğu sektöre faydalı olabilme için makine öğrenmesini modellerini geliştirmektir (Castéran ve diğ., 2017; Forhad ve diğ., 2014; Kulkarni ve diğ., 2019; Rogers, 2018).

Müşteri kayıp analizinde sınıflandırma yaklaşımları en çok kullanılan analizler arasında yer almaktadır. Diğer bir tabirle, veri madenciliği kavramının günümüzde daha popüler olmasıyla birlikte kayıp analizinin daha detaylı şekilde karşımıza çıktığını görmekteyiz. Daha önceki dönemlerde müşteri sepet analizi ve birliktelik kuralları uygulanırken artık verinin artmasıyla birlikte farklı makine öğrenmesi yöntemlerinden de yararlanılır hale gelmiştir. Bu süreçte veri madenciliği ve makine öğrenmesi gibi hususların açıklanması gerekmektedir. Bu kavramlardan bir sonraki ana başlıkta detaylı şekilde bahsedilmiştir.

Veri madenciliğinin bu tez kapsamında kullanılan önemli alt başlıklarından biri müşteri kayıp analizidir. Veriler, şirketler için faydalı ve sürdürülebilir kararlar alınabilmesi için elde edilmesi gereken temel yapıların başında yer alır. Çünkü artan veri aynı zamanda örtük bilgiyi de beraberinde getirmiş ve şirketler bu örtük bilgiyi keşfederek müşterilerin isteklerini ve ihtiyaçlarını belirlemeyi hedeflemişlerdir. Bu bakımdan müşteri kayıp analizi veri madenciliği ve yapay zeka alanında ön plana çıkan temel analizlerden biri olarak karşımıza çıkmaktadır. Müşteri kaybı analizi, müşterinin geçmişe ait verileri aracılığıyla (Forhad ve diğ., 2014) herhangi bir şirketin müşteri tabanındaki kayıp oranının hesaplanması işlemidir. Bir hizmet veya ürün kullanmayı bırakma olasılığı en yüksek olan müşterilerin keşfedilmesini içermektedir. Müşteri kayıp analizi aynı zamanda, bir şirkette müşteriyi elde tutmak için sürdürülebilir ve sağlam bir strateji geliştirmede son derece yararlıdır (Han ve Kamber, 2012).

Şirketlerin mevcut verileri ve metrikleri önceki yıllarda tanımlayıcı istatistik (descriptive statistic) analizleriyle açıkladığı bilinmektedir (Gold ve Tzuo, 2020). Ancak örtük bilginin artmasından ötürü artık müşterilerin ihtiyaçları da bir o kadar değişiklik göstermiştir. Bu konuda veri madenciliği alanında değerlendirilen Makine Öğrenmesi yöntemleri bu

problemlerin çözümünde ön plana çıkmaktadır. Müşteri kayıp analizi, makine öğrenmesi yöntemleri ile sınıflandırma yaparak ve mevcut özniteliklerden etki oranı hesaplanarak farklı göstergeler sunmaya yardımcı olmaktadır. Müşteri kayıp analizi aynı zamanda farklı makine öğrenmesi algoritmaları ile bankacılık ve finans, müzik yayıncılığı, sigortacılık ve telekomünikasyon gibi sık bilinen sektörlerde farklı sorunları çözmek için kullanılmıştır. Bu nedenle makine öğrenmesi modellerinin tahmin probleminde uygulanabilecek ilk adımı, müşteri davranışlarını müşteri veri noktalarından anlamaktır.

Oyun şirketleri, oyun içi metrikleri içeren verileri araştırmacılarla paylaşmaktan kaçınırlar. Şimdiye kadar sınırlı sayıda oyun şirketi oyuncu verilerine açık erişim sunmuştur. Bu şirketlerin verileri kamuya açık hale getirme konusundaki isteksizliği, oyun madenciliğine özgü tekniklerin ve yapay zeka araştırmalarının yaygın kullanımını ve geliştirilmesini sınırlamaktadır.

Oyun şirketleri, oyuncuların aksiyonlarına, oyun içi gelişimlerine ve satın alımlarına göre büyük miktarda oyuncu verilerini farklı analizlerde kullanmaktadır (Lee ve diğ., 2019). Mevcut verilerle, kullanıcıların genel davranışlarını modellemek (El Nasr, 2013) ve oyun içi dinamikleri kullanıcının şirkete gelir olarak geri gelme ihtimalini araştırmaktadırlar. Çoğu istatistiksel yaklaşım, geçmiş verilere göre bir tahmin modeli geliştirmekte ve bu modele oyuncuların genel profiline dair verileri veri tabanında toplanan değerleri de dahil etmektedir. Örneğin, oyun içi kayıt verileri, oyun ile ilgili platformlarda yayınlanan yorumlar, müşteri memnuniyet anketleri veya oyun içi verilen görevlerde görev yapma sıklığı gibi durumlar tahmin modelleri geliştirmek için önemli veriler içermektedir. Bu bölümde müşteri, müşteri ilişkileri yönetimi ve müşteri kayıp analizine dair bilgiler verilmiştir. Veri madenciliği ve makine öğrenmesine yönelik bilgiler ise çalışmanın devamında yer almaktadır.

2.9. VERİ PİRAMİDİ VE BİLGİ KEŞFİ

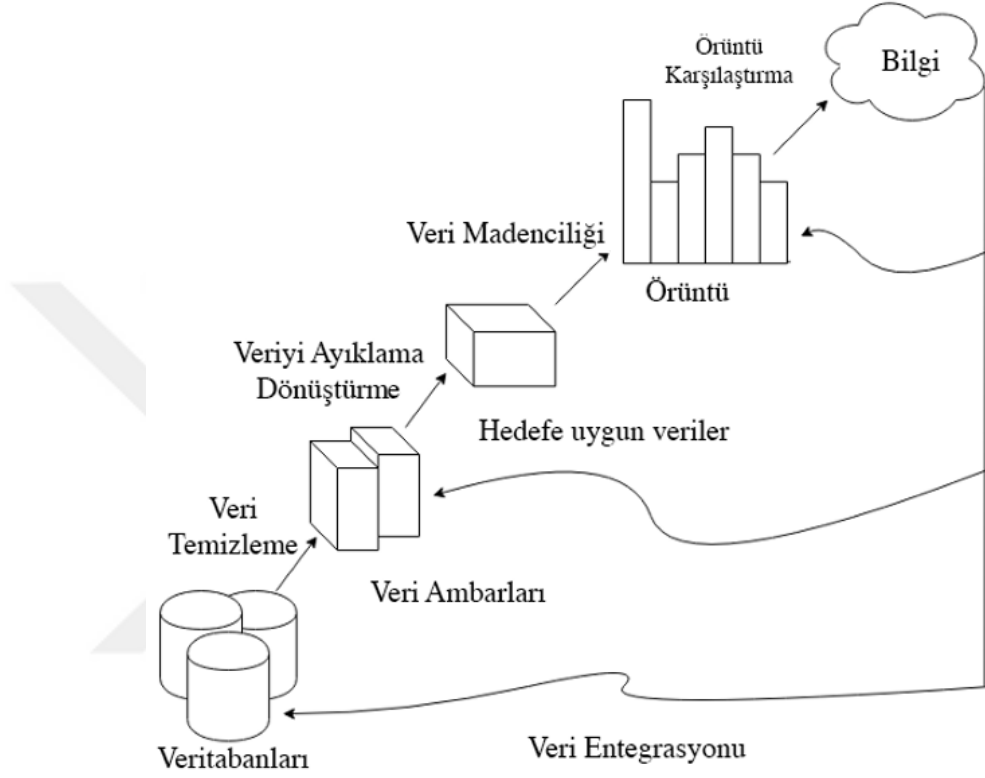
Gelişen ve yenilik içerisinde teknoloji ile verilerin daha değerli olduğu dijital dünya içerisinde bulunuyoruz. Veriler, günlük yaşantımızda bireylerin kişisel yaşamlarında karar vermede, ihtiyaçlarının belirlenmesinde, davranışlarının anlaşılmasında ve analiz edilmesinde önemli rol oynamaktadır. Veri; sağlık, finans, eğitim, ulaşım ve daha pek çok alanda kullanılmaktadır. Ayrıca, veri kullanımı kamu ve toplum sağlığı, suç önleme ve afet yönetimi gibi alanlarda iç görü (insight) sağlayarak toplumun genel yaşam kalitesinin iyileştirilmesine yardımcı olur.

Genel olarak veri, modern toplum için vazgeçilmez bir araç haline gelmiştir ve geleceğimizi şekillendirmede her geçen gün daha önemli bir rol oynamaya devam etmektedir. Veriye sahip olmak; varsayımlar üretebilmek, çıkarımlar yapabilmek ve planlanan uygulamalara yön verebilmek açısından oldukça önemlidir. İnternet ve akıllı cihazlarda üretilen veri miktarı önceki yıllara göre değerlendirildiğinde büyük bir artış olduğu görülür. Bu da beraberinde örtük bilgiyi getirir ve bireylerin ihtiyaçlarına cevap vermek için birtakım analizleri gerçekleştirme gerekliliğini doğurur.

Çoğu uygulamanın birincil amacı, karmaşık bir veri akışındaki kalıpları anlamlandırabilmektir. Bu süreç daha sonra bilginin keşfi ve karar verme süreci ile birleştirilir. Veri kümeleri büyük ve karmaşık olduğunda, geleneksel istatistiksel yöntemleri kullanarak kalıpları işlemek ve analiz etmek zorlaşır (Awad ve Khanna, 2015). Verinin boyutu artış gösterdikçe analiz edilmesi ve depolanma süreci yapısı gereği zor bir hal alabilir. Bu konuda çözüm olarak veri tabanlarında bilginin keşfi sürece yardımcı olur. Veri tabanlarında bilginin keşfi olarak da bilinen veri madenciliği; veri tabanlarında depolanan verilerden örtük, önceden bilinmeyen ve potansiyel olarak yararlı bilgilerin önemli olgu halinde çıkarılmasını ifade eder. Aynı zamanda bilgi çıkarma, temeldeki büyük ve anlamsal olarak bulanık veri kümelerinden anlamlı ve faydalı kalıpları belirlemek adına bilgi veri tabanı oluşturmak için yapılandırılmış ve yapılandırılmamış kaynaklardan bilgi toplama sürecidir (Clarke ve diğ., 2009).

Verinin boyutundaki artış büyük veri kavramının ortaya çıkmasına zemin hazırlamıştır. Büyük veriler, standart veri tabanı yönetim sistemleri kullanılarak işlenemeyecek kadar büyük hacimlerde toplanan ve çok karmaşık ve yapılandırılmamış formlar oluşturan verilerdir. Büyük verilerle ilgili ortaya çıkan zorluklar arasında yalnızca artan hacimle değil, aynı zamanda çıkarılması, dönüştürülmesi, analiz edilmesi, depolanması ve görselleştirilmesi gereken veri akışlarının geniş çeşitliliği ve karmaşıklığı da yer alır. Günümüz dünyasında, insanlar, nesneler ve bunların teknoloji etkileşimleri aracılığıyla çok büyük miktarda veri üretilmektedir. Google, Twitter, Facebook, Wikipedia, YouTube vb. web siteleri, veri merkezlerinde hemen hemen her saat, terabayt veya petabayt veriden daha büyük hacimli veriler oluşturur (Zikopoulos ve diğ., 2012). Modern teknolojik gelişmeler, endüstrinin büyük veri ve büyük veri analitiğine giriş yapmasını sağlamıştır. 21. yüzyılda Büyük Veri (Big Data), endüstrinin çeşitli alanlarında ve akademik alanda araştırma ve geliştirme uygulamalarının vazgeçilmez bir parçası haline gelmiştir. Bu kavram, geleneksel veri tabanı teknikleri ve teknolojileri aracılığıyla saklanması,

işlenmesi ve analizi zor olan veri hacimlerindeki artışı ifade eder (Nayyar ve Puri, 2017). Aynı zamanda, çeşitli, karmaşık ve yüksek ölçekli büyük miktardaki veri setlerinden gizli değerleri çıkarmak için yeni entegrasyon formları gerektiren kapsamlı araç, teknik ve teknolojilerin bütünü olarak tanımlanabilir.



Şekil 2.12: Büyük veri sınıflandırılması (Fayyad, 2005'ten uyarlanmıştır.)

Şekil 2.12'de büyük verinin basamaklarının sınıflandırılması verilmiştir. Bu basamaklara göre veri tabanından gelen verilerin öncelikle veri temizleme (data cleaning), ardından veri seçme ve dönüştürme (data selection and transformation), sonrasında veri madenciliği (data mining) yani veriden anlamlı bilgi çıkarma süreci, hemen ardından örüntü karşılaştırma (pattern evaluation) ve bilgiye erişim ile sonlandırma olarak gösterilmiştir. Veri madenciliği süreçlerinden bir sonraki başlıkta detaylı olarak bahsedilmiştir.

Çevrimiçi veriler, müşterilerin tüm ihtiyaçlarını karşılamak için geliştirilmekte olan farklı kaynaklardan ve hizmetlerden gelmektedir. Bulut hizmetleri, sosyal medya ve sensör verileri vb. çeşitli hizmetler ve kaynaklar yüksek hacimli veriler üretir ve verilerin kullanıcı ihtiyaçlarına göre analiz edilebilmesi ve kullanılabilmesi için bir yönetim sürecine dahil

edilmesi gereklidir. Bu kaynaklardan elde edilen veriler kişi ve kuruluşlar için avantajlı olsa da verilerin yönetimi ve analizi oldukça zor bir süreçtir.

İnternet üzerinde devasa boyutta veri oluşturulmakta ve bu veriler çeşitli veri depolama aygıtlarına yedeklenmektedir. Mevcut veri hacmindeki bu denli büyüme; toplumun kişisel bilgisayara sahip olma oranının artması, güçlü veri toplama ve depolama araçlarının hızlı gelişiminin bir sonucu olarak ortaya çıkmıştır.

Dünya çapında tıp, fen bilimleri, işletme ve telekomünikasyon dijital endüstri olmak üzere birçok işletme yüksek kapasiteli veri akışı sağlamaktadır (Han ve diğ., 2012). Marketler günlük alışverişlerde yüksek miktarda veri üretmektedir. Telekomünikasyon firmalarında ise iletişim hizmeti sunduklarından dolayı anlık büyük veri akışı oldukça fazladır. Tıp ve sağlık alanlarında, tıbbi kayıtlar ve hasta izleme uygulamaları ciddi miktarda veri üretmektedir. Benzer şekilde Web ortamında sosyal medya aracılığıyla oluşturulan içerikler de büyük veri miktarının artmasında önemli rol oynamaktadır. Aynı şekilde oyun endüstrisinde oyuncuların anlık akan verileri, satın alma işlemleri ve güncel başarıları ise büyük veri üretiminin gerçekleştiği diğer bir alan olarak karşımıza çıkmaktadır.

2.9.1. VERİ MADENCİLİĞİ VE UYGULAMA ALANLARI

Veri madenciliği yeni bir çalışma alanı olmamakla birlikte önceki yıllarda, "veri tarama (data dredging)" ve "veri üzerinden balık avlama (fishing through data)" gibi isimlerle de adlandırılmaktaydı (Hand ve diğ., 2001). Veri madenciliğine dair ilk yapılan tanımlardan biri "verilerden örtük, önceden bilinmeyen ve potansiyel olarak yararlı bilgilerin elde edilmesi" olarak karşımıza çıkmaktadır (Fayyad ve diğ., 2001).

İlerleyen dönemlerde veri madenciliği, bir veri setinde örüntü veya ilişki bulmak için çeşitli algoritmaların uygulanmasına atıfta bulunarak mevcut tanım genişletilmiştir (Fayyad ve diğ., 1992). Veri madenciliği, bilgisayarlarda depolanan büyük miktardaki verilerin analizini ifade eder (Olson ve Delen, 2008). Veri madenciliği, büyük miktarda veriden ilginç (örtük, önceden bilinmeyen ve potansiyel olarak yararlı) modellerin veya bilgilerin çıkarılması sürecidir (Tan ve diğ., 2018). Birçok araştırmacı veri madenciliğini yaygın olarak kullanılan başka bir terim olan veriden bilgi keşfi veya KDD ile eşanlamlı olarak görürken, bir diğer kısmı ise veri madenciliğinin bilgi keşfi sürecinde yalnızca önemli bir adım olduğunu düşünmektedir. KDD sürecinde örtük bilginin ortaya çıkarılması hedeflenirken, veri madenciliğinde tüm bilgi keşfi

sürecini ifade etmek için kullanılır. Bu sebepten ötürü veri madenciliği, büyük miktarda veriden ilginç kalıpları ve bilgileri keşfetme sürecidir (Han ve diğ., 2012). Aynı zamanda veri madenciliği, şirketlerin iç ve dış problemlerine çözüm üretmesine, müşteri davranışlarını tahmin etmesine, riskleri azaltmasına ve yeni fırsatlar bulmasına yardımcı olmak için çok büyük miktarda bilgi ve veri kümesini analiz etme, yararlı bilgi çıkarma veya madencilik sürecidir (Calders ve Custers, 2013). Diğer bir tanımla veri madenciliği, kullanılabilecek olan ham verilerden gerekli olan bilgileri elde etmek ve onları anlaşılabilir bir yapıya dönüştürmek için kullanılan bir yöntemdir (Giudici ve Figini, 2009).

Yukarıda bahsedilen tanımlara paralel olarak, veri madenciliği büyük veri tabanlarından tahmine dayalı bilgilerin otomatik olarak çıkarım yapılmasıdır. Veri madenciliği verilerde ilginç yapı bulma sürecidir (Roiger, 2017). Yapıdan kasıt bir dizi kural, bir grafik veya bir denklem ve daha fazlasını içeren birçok biçim olabilir. Veri madenciliği, örüntü tanıma teknolojilerinin yanı sıra istatistiksel ve matematiksel teknikleri kullanarak, depolama alanlarında depolanan büyük miktarda veriyi eleterek anlamlı yeni korelasyonlar, örüntüler ve eğilimler keşfetme sürecidir (Larose ve Larose, 2014). Ayrıca veri madenciliği bilgisayar bilimi, makine öğrenimi ve veri tabanı teknolojisi ile farklı fikirleri, araçları ve yöntemleri bir araya getirmektedir.

Veri madenciliği, iş zekası (business intelligence), sosyal medya, biyoloji, tıp bilimi ve sağlık hizmetleri gibi popüler alanlarda uygulamalı ve aktif olarak kullanılmaktadır. Aynı zamanda oyuncu analizi, finansal veri analizi, perakende sektörü, telekomünikasyon endüstrisi, biyolojik veri analizi ve izinsiz giriş tespiti uygulamalarında da kullanıldığı bilinmektedir. Günümüzde veri madenciliği başlığı altında uygulanan yapay zekâ ile görüntü işleme süreci tıp uygulamalarına öncü olmuştur. Dolandırıcılık tespiti olarak adlandırılan Fraud Detection uygulamasının ilk örneğinin bankacılık sektöründe kullanıldığı bilinmektedir. Bir diğer kullanımı da uzay ve havacılık alanında geçmişe dair kanıtlar için deneylerde kullanılmasıdır. Hava araçlarının sensör ve otomasyon sistemlerinde de aktif olarak kullanıldığını söyleyebiliriz.



Şekil 2.13: Veri madenciliği ile ilişkili çalışma alanları (Kelleher ve Tierney, 2022)

Veri madenciliğinin uygulama alanlarının yanında ilişkili olduğu çalışma alanlarını açıklamak, yapısı gereği daha mantıklı olacaktır. Bölümün başında belirtildiği üzere istatistik temelli modellerin zamanla veri madenciliği altında toplandığını görmek mümkündür. Bu doğrultuda istatistik alanı; veri madenciliği ile ilgili alanlar grubunda ilk sırada görülür. Bir diğeri ise veri madenciliğinin üzerine algoritmalar geliştirilmiş olan ve aynı zamanda yapay zekanın temellerinden birini oluşturan makine öğrenmesidir. Uzun yıllar şirketler tarafından kullanılan örüntü tanıma işlemi de bu gruba dahil edilebilir. Şekil 2.13'te veri madenciliği ile ilişkili olan alanlar şema haline getirilmiştir.

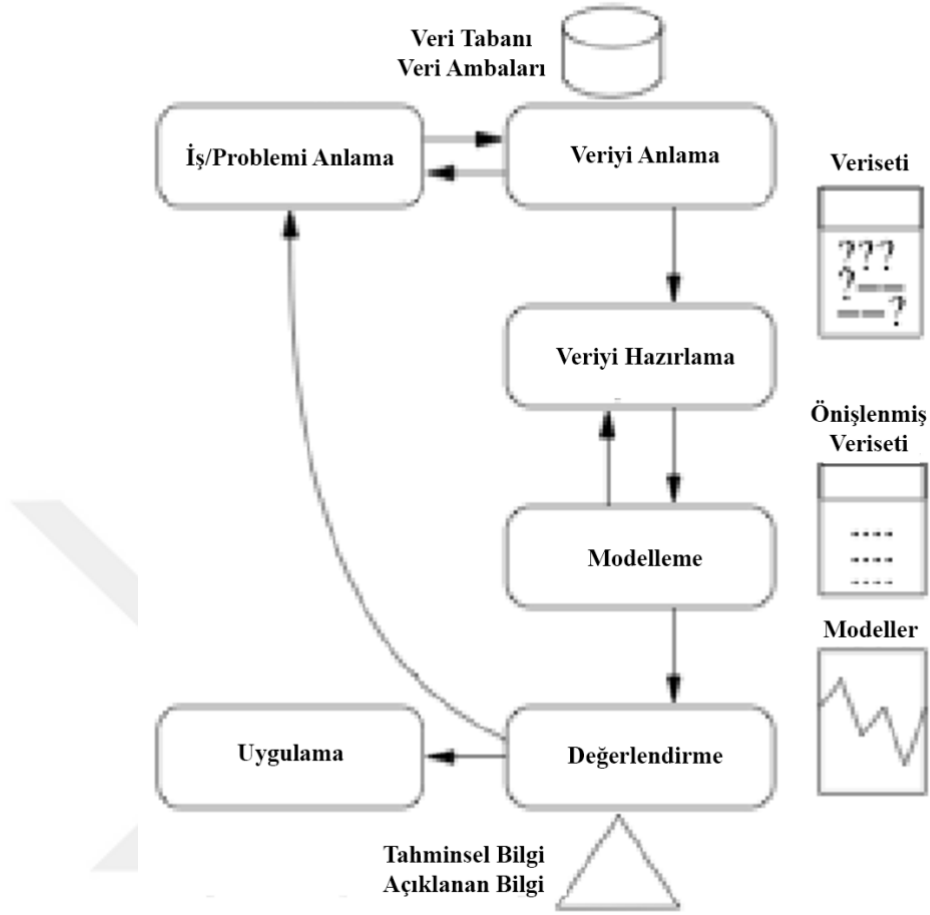
2.9.2. VERİ MADENCİLİĞİ SÜRECİ

Veri madenciliğinde bir probleme çözüm üretmek için ham veriler incelenmeli, mevcut bilgiler tanımlanmalı, mevcut raporlardan sonuç çıkarabilmek adına bir model oluşturulmalıdır. Ham veriler birçok farklı kaynaktan ve biçimde mevcut olduğundan, veri kaynaklarından ve veri formatlarından bir model oluşturmak kendini tekrar eden bir süreçtir. Önceki bölümlerde verinin günlük hayatta hızlı şekilde üretildiğinden ve bu istatistiğin gittikçe arttığından bahsedilmişti. Ham petrolün nasıl işlendiğine dair bir yöntem ve süreç var ise, verinin de

işlenmesi ve anlamlı bilgiler elde edilebilmesi için de bir modele ihtiyacı vardır. Bu süreçte sıklıkla kullanılan üç modelin isimlerine yer verilmiştir. Bunlar; KDD, CRISP-DM ve SEMMA modelleridir. Bu tez çalışmasında CRISP-DM metodolojisine göre veri madenciliği gerçekleştirildiğinden, yalnızca bu modelin teorik bilgilerine yer verilmiştir. Tüm adımlarda gerçekleştirilen uygulamaların süreçleriyle ilgili detaylı bilgiler Malzeme ve Yöntem bölümünde bahsedilmiştir.

2.9.3. CRISP-DM METODOLOJİSİ

CRISP-DM Türkçede karşılığı “Veri Madenciliği için Sektörler Arası Standart Süreç” olarak adlandırılmaktadır. Toplamda altı aşamadan oluşmaktadır. DaimlerChrysler AG, SPSS, NCR ve OHRA gibi veri madenciliği kullanıcıları ve tedarikçilerinden oluşan şirketler tarafından önerilmiş güvenilir bir veri madenciliği metodolojisidir. Veri madenciliği sürecine yapılandırılmış bir yaklaşım sağlayan döngüsel bir süreçtir. Altı aşama herhangi bir sırayla uygulanabilir. Ancak bazen önceki adımlara geri dönmeyi ve eylemlerin tekrarını gerektirebilir (Wirth ve Hipp, 1999). CRISP-DM modelinin temel hedefleri veri madenciliği sürecinde projeleri yürütmek ve farklı kişiler tarafından güvenilir, verimli şekilde tekrarlanabilen ve farklı durumlara uyarlanabilen bir süreç belirlemektir. Ek olarak büyük veri madenciliği projelerini daha az maliyetli, daha güvenilir, daha tekrarlanabilir, daha yönetilebilir ve daha hızlı hale getirmeyi amaçlamaktadır.



Şekil 2.14: CRISP-DM modeli yaşam döngüsü (Chapman ve diğ., 2000’den uyarlanmıştır.)

CRISP-DM, bir veri madenciliği projesinin yaşam döngüsüne genel bir bakış sağlar. Bir projenin aşamalarını, ilgili görevlerini ve çıktılarını içerir. Metodolojinin adımları Şekil 2.14’te verilmiştir (Chapman ve diğ., 2000).

2.9.3.1. İşin Anlaşılması

Bu adımda şirketlerin hedefleri belirlenir ve hedefe ulaşmada yardımcı olacak önemli unsurlar keşfedilir. Çözülmesi gereken iş probleminin anlaşılmasını ve projenin hedeflerinin tanımlanmasını içerir. Bu aşamanın amacı, genel iş stratejisinin belirlenen hedeflerle uyumlu olmasını sağlamaktır.

2.9.3.2. Verinin Anlaşılması

Analiz için kullanılacak verilerin keşfedilmesini ve bunların tanınmasını kapsayan adımdır. Verilerin toplanması, eksiksizlik, doğruluk ve tutarlılık açısından kontrol edilmesi ve eksik veya yanlış verilerin belirlenmesi aşamasıdır. Veriler bu aşamada betimsel istatistiklerle değerlendirilir. Aynı zamanda verilerde analiz sonuçlarını etkileyebilecek olası sorunların veya sınırlamaların belirlenmesini hedefleyen adımdır.

İstatistik ve ilişkili alanlarda bilimsel araştırma için veya başka bir amaçla toplanan veriler farklı tipte olabilirler. Veri tipleri görselleştirme ve model kurma aşamasında oldukça önem taşır. Nümerik ve kategorik olarak iki sınıfa ayrılırlar. Tablo 2.2’de bu veri türlerinin yapıları ve örneklerine yer verilmiştir (Kartal ve Balaban, 2015, Sınıksaran, 2011).

Tablo 2.2: Veri türleri ve örnekleri

Kategorik	İkili (Binary)	İki kategorik değişkenin olduğu veri türüdür.	E-posta adresine gelen mailin "spam" ya da "spam değil" durumu
	Sıralı (Ordinal)	Kategorik değişkenlerin mantıksal olarak bir arada yer almasıdır.	Büyük orta küçük gibi ayrılabilen sıralı değişkenler
	Kategorik (Nominal)	Verinin mantıksal olarak bir arada yer almadığı her kodun farklı bir sınıfı içerdiği veri türüdür.	Sıra ifade etmeden 1’den 3’e kadar farklı meslek kodlarının oluşturulması
Nümerik	Aralıklı (Interval)	Başlangıç noktasının 0 olarak kabul edildiği ancak 0 olmasının yokluk belirtmediği veri türüdür.	Hava sıcaklığının 0 C olması hava sıcaklığının olmadığı anlamına gelmemesi, 0 değerinin ölçüt olarak kabul edilmesi
	Oran (Ratio)	Başlangıç noktasının 0 oluşu gerçek anlamda yokluk belirtmesinde kullanılan veri türüdür.	Bir hastaya ait boy ve kilo verileri buna örnek verilebilir.

2.9.3.3. Verinin Hazırlanması

Veri dönüştürme verilerin analiz için uygun bir formata dönüştürülmesini içermesinin yanı sıra birden fazla kaynaktan gelen verilerin temizlenmesini, dönüştürülmesini ve modelleme ve analiz için kullanılabilir tek bir veri setine entegre edilmesini içermektedir. Normalizasyon,

kategorik verilerin nümerik tipe dönüştürülmesi ve öznitelik türetme gibi işlemler bu aşamada gerçekleştirilir (Sınıksaran, 2011).

Bir probleme yönelik çözüm üretilmek istenen veri setinde farklı boyutta satırlar ve sütunlar yer alabilir. Ele alınan bu değerler farklı veri türlerini barındırabilir. Bu verilerin değişim aralıkları bazen çok uzak olabilirken bazen çok yakın olmasıyla durumuyla karşılaşılır. Veri setindeki büyük değerlerin küçük değerleri domine etmesi sonucu veri üzerindeki baskıyı artırarak sonuçların farklılaşmasına sebep olmaktadır (Sınıksaran, 2011; Gürsakal, 2021). Bu durum özellikle KNN gibi parametrik algoritmaları olumsuz yönde etkileyebilmektedir (Pandey ve Jain, 2017). Bu soruna çözüm olarak verinin belirli değerler arasında bir formata uygun hale getirilmesi önerilir. Bu çözümler normalizasyon teknikleri altında değerlendirilmektedir. Minimum ve Maksimum normalizasyon, ondalık ölçekleme, standardizasyon bu yöntemler arasında yer almaktadır. Kelime anlamı olarak normalizasyon ve standardizasyon eş anlamlı olarak kullanılırken, teorik olarak birçok bilim dalında farklı anlamlarda kullanılır. Bu sebepten ötürü, veri normalleştirilmesi ya da veri standartlaştırılması olarak kullanmak daha uygundur (Akpınar, 2018).

Normalizasyon, verilerin istenilen değerler arasına indirilmesi işlemidir. En bilinen ve veri setleri üzerinde model geliştirirken en çok kullanılan ise 0-1 arası normalizasyondur. Bunun için Denklem 2.1'den yararlanılır.

$$x' = \left(\frac{x - \min_a}{\max_a - \min_a} \right) * (\text{yenimax}_a - \text{yenimin}_a) + \text{yenimin}_a \quad (2.1)$$

Burada veriler 0-1 arası bir normalizasyon indirgenecek ise çarpma işleminden sonraki kısma ihtiyaç duyulmaz. Çünkü yeni maksimum değer 1 ve yeni minimum değer 0'dır.

Diğer normalizasyon yöntemi standart olarak da bilinen z skor normalizasyondur. Belirli bir veri noktasının ortalamadan kaç standart sapma uzakta olduğunu ölçer. Bir z-skoru negatif veya pozitif olabilir.

$$z = \left(\frac{X - \mu}{\sigma} \right) \quad (2.2)$$

Denklem 2.2'de μ özelliğin ortalama değeri, σ ise özelliğin standart sapmasıdır. Bir özelliğin tüm değerleri ortalamasına tam olarak eşitse, 0'a normalize edilecektir. Ortalamanın altındaysa,

negatif bir sayı olacak ve ortalamanın üzerindeyse pozitif bir sayı olacaktır. Bu negatif ve pozitif sayıların büyüklüğü orijinal özelliğin standart sapması tarafından belirlenir. Bu yöntemde hesaplanan değerlerin uç değerleri aşabilme ihtimali de vardır. Burada dikkat edilmesi gereken hususlardan biri ise veri seti kapsamında normalizasyon işleminin gerekli olup olmadığını modele uygulamadan önce ve uyguladıktan sonra elde edilecek sonuçların karar verici olduğudur.

Birçok makine öğrenimi algoritması doğrudan ham veriler üzerinde çalışamaz. Modelin ön işleme aşamasında kategorik değişkenlerin nümerik olarak dönüştürülmesi gerekir. Bu işlem yapılmadığı takdirde algoritma bunu nümerik bir değer olarak algılar ve bu durumda yapılan tahminlerin güven derecesi düşer (García ve diğ., 2015). Bu sebepten ötürü verinin dönüştürülmesi kritik derecede önemlidir. Ancak belirtilmesi gereken bir diğer husus ise karar ağacı gibi bazı algoritmaların doğrudan kategorik verilerle çalışabilmesidir.

Sıralı bir ilişkinin bulunmadığı kategorik değişkenler için tamsayı kodlaması yeterli değildir. Bu durumda, one-hot encoding uygulanabilir. Burada tamsayı kodlu değişken kaldırılır ve her benzersiz tamsayı değeri için yeni bir ikili değişken eklenir. Şekil 2.15'te görüldüğü üzere kodlama şekli genel olarak istatistik alanında kukla değişken (dummy variable) olarak adlandırılır (Garavaglia ve Sharma, 1998).



renk
kırmızı
yeşil
mavi
kırmızı

renk_kırmızı	renk_mavi	renk_yeşil
1	0	0
0	0	1
0	1	0
1	0	0

Şekil 2.15: One-hot encoding örneği

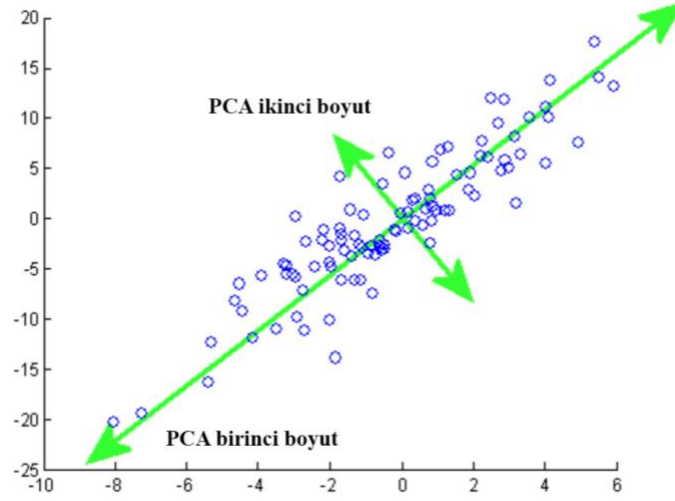
Orijinal formatındaki veriler, sınıflandırma veya regresyon modellerini eğitmek için neredeyse hiçbir zaman doğrudan kullanılmaz. Bunun yerine veri bilimciler, makine öğrenimi

algoritmalarını eğitmek için zamanlarının büyük bir bölümünü veri ön işleme basamağında öznitelik oluşturmaya ayırır (García ve diğ., 2015).

Öznitelik mühendisliği olarak da bilinen öznitelik oluşturma, bir makine öğrenimi modelinin doğruluğunu ve performansını artırmak için bir veri kümesindeki özelliklerden yeni özellikler veya öznitelikler oluşturma sürecidir (Zheng ve Casari, 2018). Öznitelik oluşturma, veri ön işleme ve özellik seçiminde çok önemli bir adımdır ve verilerden ilgili bilgileri çıkarmak ve modelin girdi değişkenleriyle hedef değişken arasındaki ilişkileri daha iyi anlamasına yardımcı olur.

Boyut azaltma, makine öğrenimi ve örüntü tanıma (pattern recognition) görevlerinde çeşitli amaçlarla kullanılır. Hesaplama verimliliğini arttırma, aşırı uyum problemini çözme, gürültülü veriyi azaltma gibi işlevlerde uygulanmaktadır. Faktör analizi (FA), Doğrusal Diskriminant Analizi (LDA), Temel Bileşen Analizi (PCA) ve t-SNE gibi popüler boyut azaltma yöntemleri mevcuttur. Bu çalışmada Temel Bileşenler Analizi ön-işleme kısmında boyut azaltma probleminde kullanıldığından ötürü bu yöntemin açıklanmasına yer verilmiştir.

Temel Bileşenler Analizi (PCA), boyut azaltma ve veri görselleştirme için kullanılan istatistiksel bir tekniktir (Jolliffe, 1986). İlk olarak Pearson (1901) tarafından ortaya atılmış ve daha sonra Hotelling (1933) tarafından geliştirilmiştir (Hotelling, 1933). Boyut azaltma, özellik çıkarma ve veri görselleştirmek için kullanılır.



Şekil 2.16: Temel Bileşenler Analizinde örnek boyut birleştirme

Verinin boyutları arttıkça, görselleştirme ve üzerinde hesaplamalar yapmanın zorluğu da artar. Bir veri setinin analizinde boyut azaltmak isteniyorsa, gereksiz boyutlar kaldırılarak en önemli boyutları saklanır. Bu noktada veride istatistiksel olarak etkisi düşük ve diğer değişkenlerle birleştirilmesi düşünülen özneliliklerin birleştirilme işlemi gerçekleştirilir. Çok düşük etkiye sahip iki boyut TBA uygulandığında artık nispeten daha iyi bir etki sahibi olabilir (Hotelling, 1933).

Şekil 2.16’da yer alan Temel Bileşenler Analizinin arkasındaki temel fikir, orijinal verilerdeki maksimum varyansı yakalayan yeni bir ilişkisiz değişkenler (temel bileşenler) kümesi bulmaktır. Bu yeni değişkenler, orijinal özelliklerin doğrusal kombinasyonları kullanılarak orijinal verilerin dönüştürülmesiyle elde edilir. Dönüştürülen veriler daha sonra orijinal verileri daha düşük boyutlu bir uzayda temsil etmek için kullanılabilirken varyansın mümkün olduğunca büyük bir kısmı korunur (Abdi ve Williams, 2010).

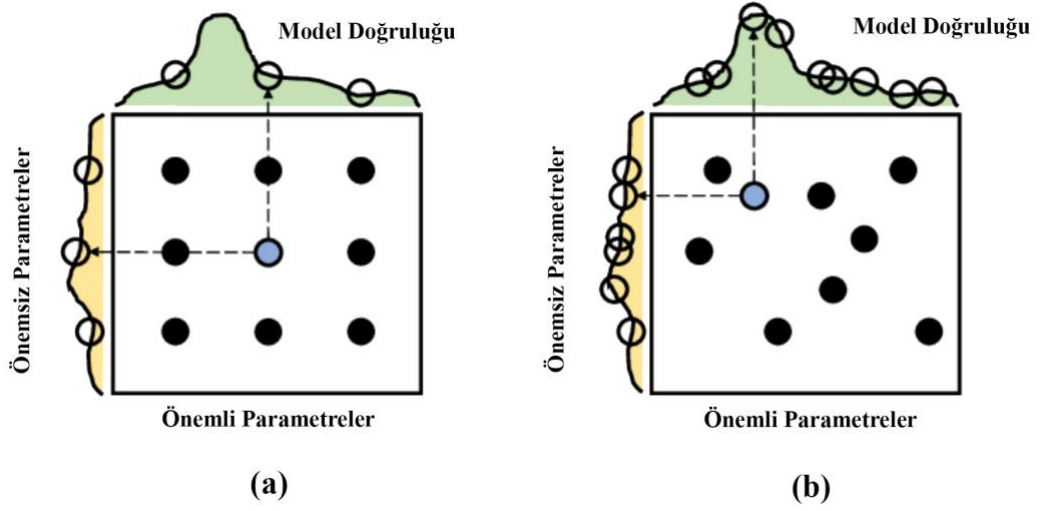
2.9.3.4. *Modelleme*

Seçilen modeli değerlendirmek için test süreci oluşturma, veri setinden modeller oluşturma ve oluşturulan modelin uzmanlar ile sonucu tartışmak için değerlendirilmesi gibi veri madenciliği tekniğinin seçimi bu adımda yapılır. Ayrıca uygulanan modelde aşırı uyma, eksik uyma veya modelin beklenen performansı göstermemesi durumunda, belirlenen değerler kapsamında hiper parametre araması yapılarak modelin performansının iyileştirilmesini içerir.

Parametre ve hiper parametre kavramları makine öğrenme algoritmalarının en önemli değişkenleridir. Ancak bazı noktalarda birbirinden ayrılırlar. Parametre, modele dahil olan ve değeri verilerden tahmin edilebilen bir yapılandırma değişkenidir. Parametreler makine öğrenmesi algoritmalarının anahtarıdır (Brownlee, 2021). Modelin geçmiş eğitim verilerinden öğrenilen kısımdır. Yapay sinir ağlarında yer alan ağırlıklar, destek vektör makinesindeki destek vektörleri, doğrusal regresyon veya lojistik regresyondaki katsayılar buna örnek olarak verilebilir. Hiperparametre ise modelin dışında olan ve değeri verilerden tahmin edilemeyen değişkenlerdir. Ağırlıklı olarak model parametrelerini tahmin etmeye yardımcı olmak için süreçlerde kullanılır ve uygulayıcı tarafından belirlenirler. Sinir ağlarının eğitiminde kullanılan öğrenme oranı (learning rate), destek vektör makineleri için C değeri veya k-en yakın komşu algoritmasında yer alan k değerleri buna örnek verilebilir (Pedregosa ve diğ., 2020).

Bir problemin çözümünde geliştirilen model için en iyi hiperparametre değerini farklı değerlerle denemeden bulmak mümkün değildir. En iyi hiperparametre seçim sürecinde temel kurallar kullanılabilir, diğer modellerde yer alan değerleri denenebilir veya deneme yanılma yoluyla en iyi değer bulunabilir. Deneme yanılma yöntemi zaman, performans ve kaynak bakımından oldukça maliyetli olabilir. Makine öğrenmesinde buna alternatif olarak arama algoritmaları geliştirilmiştir. Arama algoritmaları, modeli uygulayan kişi tarafından belirlenen hiperparametlerin belirli aralıklarla model uzayında farklı kombinasyonlarla aranmasına dayanır. Makine öğrenmesinde en sık kullanılan arama algoritmaları Şekil 2.17’de verilen Izgara Arama (Grid Search) ve Rastgele Arama (Random Search) yöntemleridir (Goodfellow ve diğ., 2016; Géron, 2019; Alpaydin, 2020).

Izgara arama, hiperparametrelerin optimum değerlerini hesaplamaya çalışan bir ayarlama tekniğidir (Chollet, 2021). Bir modelin belirli parametre değerleri üzerinde gerçekleştirilen kapsamlı bir aramadır. Model aynı zamanda bir tahmin edici olarak da bilinir. Zaman ve kaynak kullanımında oldukça yardımcıdır.



Şekil 2.17: Izgara ve rassal hiperparametre arama yöntemlerinin şeması

Rastgele arama, modelin en iyi performansı sunmasında hiperparametrelerin rastgele kombinasyonlarını kullanmasına dayanan yöntemdir (Chollet, 2021). Tüm parametre değerlerinin denenmediği ve bunun yerine parametrelerin rassal olarak verilen sabit sayıda iterasyonla örnekleneceği anlamına gelir.

Arama uzayı yüksek boyutlu olduğunda rastgele arama daha geniş bir hiperparametre uzayını keşfedebildiğinden ötürü, rastgele arama'nın ızgara arama'ya göre kullanılması tavsiye edilir. Çünkü, çok fazla hiperparametre ızgara arama için zaman karmaşıklığını artıracaktır. Bununla birlikte, ızgara arama hesaplama açısından çok pahalı olsa da kapsamlı bir arama olarak, belirtilen hiperparametrelerin tüm kombinasyonlarına araştırmak için daha kullanışlıdır (Liashchynskyi ve Liashchynskyi, 2019).

2.9.3.5. Değerlendirme

Bu adım modelin yeni verilere genelleme yeteneğini değerlendirmek için test veri seti olarak bilinen, ayrı bir veri kümesi üzerindeki performansının ölçülmesini içerir. Ayrıca, seçilen modelin genel kullanım için uygun olup olmadığını veya daha fazla ayarlama gerekip gerekmediğinin belirlenmesine yardımcı olabilir.

Geliştirilen modellerin tahminleri bazen eğitim veri setinden çok fazla etkilenecek öğrenme değil ezberleme gerçekleştirir. Sisteme dışarıdan bir tahmin yaptırıldığında modelin performansı oldukça düşer. Bu durum aşırı uyum ve eksik uyum problemlerini ortaya çıkarır.

Aşırı uyum, bir modelin eğitim verilerindeki gözlem sayısına göre çok fazla parametreye sahip olması durumunda ortaya çıkar ve modelin altta yatan örüntüler yerine verilerdeki gürültüyü veya rastgele dalgalanmaları öğrenmesine neden olur. Model eğitim verileri ile çok fazla özdeşleşir, bu da veriler üzerinde iyi performans gösterir ancak yeni veriler üzerinde kötü performans göstermesine yol açmaktadır (James ve diğ., 2013, Goodfellow, 2016).

Eksik uyum ise modelin verilerdeki temel kalıpları yakalayamadığı hem eğitim verilerinde hem de geleceğe dönük verilerde düşük performansla sonuçlanan bir problemdir (Bishop, 2006). Geliştirilen modelde çok az parametre olması veya yeteri kadar kompleks olmaması da eksik uyumu doğuran faktörler arasındadır. Sonuç olarak, eksik uyum gösteren modelin hem eğitim hem de test verilerinde performans ve genelleştirme becerisi düşük olur (Chollet, 2021).

Mevcut veri kapsamında uygulanan model performans değerlerinde yöntemleri verinin ne ölçüde iyi performans gösterdiğine odaklanır. Model Değerlendirme, model geliştirme sürecinin ayrılmaz bir parçasıdır. Verilerimizi temsil eden en iyi modelin bulunmasına yardımcı olur. Ayrıca seçilen modelin gelecekte ne kadar iyi çalışacağına odaklanır (Kartal, 2015; Akpınar, 2018; Brownlee, 2021). Model performansının eğitim verileriyle değerlendirilmesi veri biliminde kabul edilebilir değildir.

Veri biliminde modelleri değerlendirmenin iki yöntemi vardır: dışarıda tutma (hold-out) ve çapraz geçerleme (cross validation). Aşırı uyumdan kaçınmak için, her iki yöntem de model performansını değerlendirmek amacıyla bir test kümesi (model tarafından görülmeyen) kullanır.

- **Dışarıda Tutma (Hold-Out):** Eğitim kümesi, tahmine dayalı modeller oluşturmak için veri setinin bir alt kümesidir. Doğrulama (validation) seti, eğitim aşamasında oluşturulan modelin performansını değerlendirmek için veri kümesinin bir alt kümesidir.
- **Çapraz Doğrulama (Cross Validation):** Yalnızca sınırlı miktarda veri mevcut olduğunda, model performansının tarafsız bir tahmin gerçekleştirmesi için k-kat çapraz doğrulama kullanılır. K-kat çapraz doğrulama, verileri eşit büyüklükte k alt kümeye böler. Her seferinde alt kümelerden biri eğitimden çıkarılır ve test kümesi olarak kullanılır. Bu işlem k kez tekrarlanır.

- **Yeniden Örnekleme (Bootstrap):** Bootstrap örnekleme, makine öğreniminde istatistiksel bir modelin veya algoritmanın belirsizliğini tahmin etmek için kullanılan bir yeniden örnekleme tekniğidir. Bir veri kümesinden tekrar tekrar rastgele örnekler çekmeyi ve her örnek üzerinde bir model oluşturmayı içerir. Ortaya çıkan modeller daha sonra modelin genel performansını tahmin etmek veya bir model topluluğu oluşturmak için birleştirilir.

Model performans değerlendirme metrikleri, makine öğrenmesi modellerinin doğruluğunu değerlendirmek için kullanılan ölçütlerdir. Bu ölçütler, farklı modelleri karşılaştırmak ve belirli bir sorun için en iyi modeli belirlemek için kullanılır. Farklı metrikler farklı model ve problem türleri için değişiklik gösterebileceğinden ötürü modele uygun metriği seçmek önemlidir.

Tablo 2.3: Sınıflandırma probleminde kullanılan performans değerlendirme ölçütleri

		TAHMİN		
		Pozitif	Negatif	
GERÇEK	Pozitif	Doğru Pozitif (DP)	Yanlış Negatif (YN) Tip 2 hata	Duyarlılık (Sensitivity) $\frac{DP}{DP + YN}$
	Negatif	Yanlış Pozitif (YP) Tip 1 hata	Doğru Negatif (DN)	Belirleyicilik (Specificity) $\frac{DN}{DN + YP}$
		Kesinlik (Precision/Positive Prediction Value) $\frac{DP}{DP + YP}$	Negatif Öngörü Değeri (False Negatif Rate) $\frac{DN}{DN + YN}$	Doğruluk (Accuracy) $\frac{DP + DN}{DP + YP + YN + DN}$

Araştırmalar kapsamında hipotezler kabul edilir veya reddedilir. Öte yandan hipotezler doğru ya da yanlış olabilir. Bu süreçte dört mümkün durum vardır. Bunlar;

1.Hipotezi kabul edilir ve hipotez doğrudur.

2.Hipotez reddedilir ve hipotez doğrudur.

3.Hipotezi kabul edilir ve hipotez yanlıştır.

4.Hipotez reddedilir ve hipotez yanlıştır.

Yukarıda verilen 1. ve 4. karar doğru iken, 2. ve 3. karar yanlıştır. İstatistikte doğru hipotezin reddedilme hatasına birincil hata, yanlış hipotezin kabul edilmesine de ikinci tip hata denir (Sınıksaran, 2011). Tablo 2.3'te sınıflandırma problemleri için model performans değerlendirme ölçütleri formülleriyle birlikte verilmiştir.

İkili sınıflandırma gerçekleştirilen bir modelde müşteri kaybının yaşanması 1, yaşanmaması ise 0 ile temsil edilirse;

DP(Doğru Pozitif): Gerçekte kayıp olarak nitelendirilen ve tahminde de kayıp olarak nitelendirilen müşterilerin sayısıdır.

YP(Yanlış Pozitif): Gerçekte kayıp olarak nitelendirilmeyen, tahminde ise kayıp olarak nitelendirilen müşterilerin sayısıdır. Tip 1 hata olarak adlandırılır.

YN(Yanlış Negatif): Gerçekte kayıp olarak nitelendirilmeyen ve tahminde de kayıp olarak nitelendirilmeyen müşterilerin sayısıdır.

DN(Doğru Negatif): Gerçekte kayıp olarak nitelendirilen, tahminde ise kayıp olarak nitelendirilmeyen müşterilerin sayısıdır. Tip 2 hata olarak adlandırılır.

F-Ölçütü: Kesinlik ve Duyarlılık ölçütlerinin harmonik ortalamasıdır. Denklem 2.3'te görüldüğü üzere her iki ölçütün birlikte değerlendirilmesine olanak tanır.

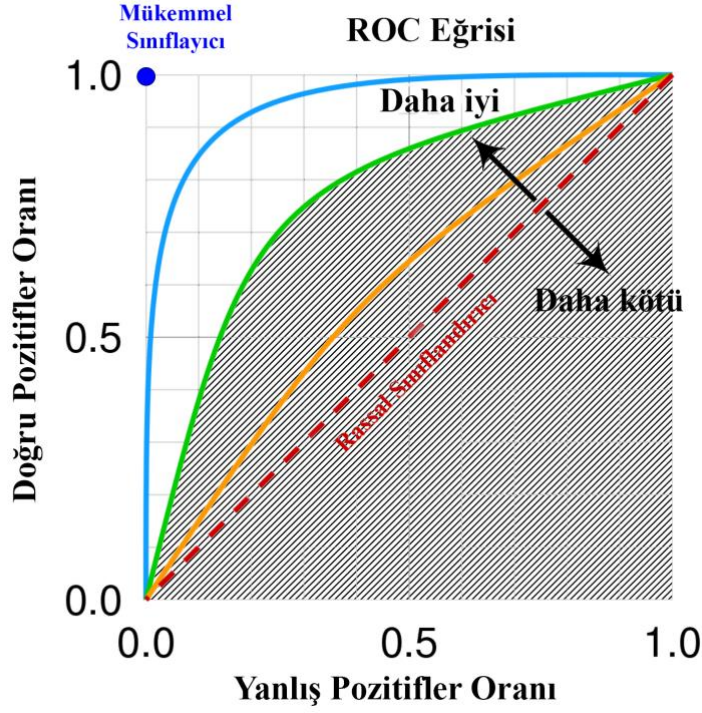
$$F \text{ Ölçütü} = 2 * \frac{\text{Kesinlik} * \text{Duyarlılık}}{\text{Kesinlik} + \text{Duyarlılık}} \quad (2.3)$$

Hata Oranı: Doğruluk yüzdesinin 1 değerinden çıkarılmış halidir (Denklem 2.4).

$$\text{Hata Oranı} = 1 - \text{Doğruluk} \quad (2.4)$$

ROC eğrisi/analizi: Y eksenini doğru pozitif oranını (DP) temsil ederken, X eksenini ise yanlış negatif oranını (YN) temsil etmektedir. Şekil 2.18'de kırmızı eğri rassal sınıflandırıcıyı temsil ederken sarı, yeşil ve mavi eğriler ise sınıflandırıcının performansını gösterir. Sonuç olarak,

eğrinin sol üst köşeye yaklaşması modelin ayırt ediciliğini ve performansını artırır (Kılıç, 2013).



Şekil 2.18: ROC eğrisi ve altında kalan alanın grafiği

ROC analizi II. Dünya Savaşı sırasında, radar tarafından tespit edilen sinyalleri doğru bir şekilde belirlemek ve düşmanı ayırt etmek için Büyük Britanya'da geliştirilmiştir. 1967'de Lee Lusted tarafından, tıbbi görüntülemelerde karar verme aşamasında ROC analizinin kullanılmasını önerilmiş ve 1969 yılında uygulamaya geçirilmiştir (akt. Centor, 1991; Zou ve diğ., 2007).

AUC Eğrisi: AUC, ROC eğrisinin altında kalan alan (Şekilde yer alan çizgili bölge) olmakla birlikte sınıflandırma modellerinin kalitesinin ölçüsüdür. Şekil 2.24'te çizilen kırmızı eğri 0.5 değerle rassal sınıflayıcı ifade ederken, mükemmel sınıflandırıcı için bu değer 1'e eşittir. Ayrıca, pozitif sınıflardan gelen olasılıkları negatif sınıflardan ne kadar iyi ayrıldığını göstermektedir (Fawcett, 2006; Kartal, 2018)

Bir regresyon modelinin becerisi veya performansı, tahminlerdeki bir hata olarak hesaplanır. Tahminlerin beklenen değerlere ne kadar yakın olduğunu bilmek modelin performansını değerlendirmek için mantıklıdır. Bu kapsamda tahminlenen ile tahminleyici arasındaki hatalar belirli ölçütlerde ifade edilir. Bunlar;

R^2 ve düzeltilmiş R^2 (adjusted R^2), bir regresyon modelinin uyum iyiliğinin (goodness of fit) ölçütleridir. Bağımlı değişkendeki varyansın modeldeki bağımsız değişkenler tarafından açıklanan oranını ölçer. $0 \leq R^2 \leq 1$ arasında değer alır (Denklem 2.5).

R^2 değerlerinin yüksek olması iyi bir uyumu gösterir ve açıklama gücü yüksektir, düşük olması uyum seviyesinin düşük ve açıklama gücünün düşük olduğu anlamına gelebilir.

$$R^2 = 1 - \frac{SSR}{SST} \quad (2.5)$$

Tahmin edilen veri noktaları $\hat{y}_i$, gözlenen veri noktaları y_i olduğu varsayılırsa;

$$SSE = \sum (y_i - \hat{y}_i)^2 \quad (2.6)$$

SSE tahmin edilen veri noktaları ile gözlenen veri noktaları arasındaki karesel farkların toplamıdır.

$$SSR = \sum (\hat{y}_i - \bar{y})^2 \quad (2.7)$$

SSR (Sum of Squared Residuals) yani artık kareler toplamıdır.

$$SST = \sum (y_i - \bar{y})^2 \quad (2.8)$$

$$SST = SSR + SSE \quad (2.9)$$

SST, toplam kareler toplamı olup bağımlı değişkenin gözlenen değerleri ile ortalaması arasındaki karesel farkların toplamıdır.

Bazı durumlarda bağımsız özniteliklerin çok fazla olması modelin gereksiz değişkenler tarafından etkilenmesine ve bu da R^2 değerinin artmasına sebep olur. Bu durumda R^2 değeri açıklayıcılığının çok yüksek olduğuna işaret etse de, model aşırı uyum göstermiş olabilir. Buna çözüm olarak Düzeltilmiş R^2 değeri üretilir ve model tekrardan değerlendirmeye alınır.

Burada n örneklem büyüklüğü ve k bağımsız değişken sayısı olduğu varsayılırsa,

$$\text{Düzeltilmiş } R^2 = 1 - \frac{\frac{SSE}{(n-k-1)}}{\frac{SST}{(n-1)}} \quad (2.10)$$

R^2 ve *Düzeltilmiş R^2* dışında değerlendirme metriklerinden bir diğeri ise Açıklanan Varyans Skoru (Explained Variance Score)'dur. Açıklanan Varyans skoru R^2 skoruna benzer, ancak aradaki önemli fark tahmindeki sistematik sapmaları hesaba katmamasıdır. Çoğu zaman R^2 skorunun tercih edilmesi tavsiye edilir (Pedregosa, 2018).

Ortalama Mutlak Hata (MAE), regresyon analizinde bağımlı değişkenin gerçek ve tahmin edilen değerleri arasındaki farkın bir ölçüsüdür. Veri kümesindeki tüm gözlemler için tahmin edilen ve gerçek değerler arasındaki mutlak farkların ortalamasıdır. Ayrıca, hataların mutlağının alınması veride oluşabilecek olan negatif değerlerin de önüne geçilmesi açısından önemlidir.

y_i gözlenen değer, $\hat{y}_i$ ise tahmin edilen değer ise;

$$\text{Ortalama Mutlak Hata} = \frac{1}{N} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (2.11)$$

Veri kümesinde tahmin edilen ve beklenen hedef değerler arasındaki karesel farkların ortalaması veya ortalaması olarak hesaplanır. Kareleme aynı zamanda büyük hataları şişirme veya büyütme etkisine sahiptir. Yani, tahmin edilen ve beklenen değerler arasındaki fark ne kadar büyükse, ortaya çıkan pozitif hatanın karesi de o kadar büyük olur. Bağımsız öznitelik değeri x 'i temsil ediyorsa, MSE x^2 olacaktır. Bu büyük değerler analizde kafa karıştırıcı olabilir ve bu sebepten ötürü sonuçlar raporlanırken genellikle MSE yerine Hata Kareler Ortalamasının Karekökü (RMSE) kullanılır.

Burada y_i veri kümesindeki i 'inci gözlenen değer, $\hat{y}_i$ ise i 'inci tahmin edilen değer ve n ise örneklem büyüklüğü olarak varsayıldığında;

$$\text{Hata Kareler Ortalaması} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (2.12)$$

$$\text{Hata Kareler Ortalamasının Karekökü} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (2.13)$$

MSE başlığında yer alan notasyonlar RMSE için de geçerli olmak üzere;

Burada ise hatanın karekökü hesaplanır. Bu da RMSE birimlerinin tahmin edilen hedef değerin orijinal birimleriyle aynı olduğu anlamına gelir.

2.9.3.6. Uygulama

Bu adımda bir uygulama planı yapılır. Veri madenciliği modeli sonuçları değerlendirilir ve modelin yararlılığı tartışılır. Herhangi bir hata varsa kontrol edilir ve tüm sürecin tekrar gözden geçirilmesi sağlanır. Herhangi bir adımda değişiklik yapılmak istenirse önceki adımlara dönüş gerçekleştirilebilir.

Bu bölümde verinin yapısı, veri madenciliği, kullanım alanları ve veri madenciliği sürecinden bahsedilmiş ve tez kapsamında çalışılan konuya dair alanyazın taramasına yer verilmiştir. Çalışmanın devam eden kısmında yapay zekanın tarihi, teorik tanımlamaları, makine öğrenmesi türleri ve algoritmaları, sınıflandırma ve regresyona dair teorik bilgiler yer almaktadır.

2.10. LİTERATÜR TARAMASI

Kawale ve diğ. (2009) popüler bir MMORPG olan Everquest II'e dair bir veri setiyle gerçekleştirdikleri çalışmada oyuncu kaybı sorununu incelemiş ve araştırmalarında oyuncular arasındaki sosyal etkiyi ve oyuncuların oyuna kişisel katılımlarını incelemeye dayanan bir kayıp tahmin modeli önermişlerdir. Sosyal etkinin, negatif etki ve pozitif etki bileşenlerine sahip bir vektör niceliği olduğunu varsaymaktadırlar. Sosyal etki ve oyuncu bağlılığı faktörlerini bir araya getiren kayıp tahmini yöntemlerinin, geleneksel yayılım modeli veya oyuncu bağlılığı faktörü kullanılarak yapılan tahminlere kıyasla kendi veri kümeleri için tahmin doğruluğunu önemli ölçüde arttırdığı görülmüştür.

Lewis ve Wardrip-Fruin (2010) World of Warcraft Armory veri setiyle oyuncuların oyun içinde seçtikleri karakterlerin sınıflandırması amacıyla gerçekleştirilen niceliksel sonuçların ön plana çıktığı çalışmada %88,16 ile %98,91 arasında değişen bir başarı oranı elde etmiştir. Yapısı

itibariyle çalışma, müşteri kayıp tahmini alanına doğrudan etki etmese de yayınlandığı yıla bakıldığında günümüz çalışmalarına katkı sağlamıştır.

Borbora ve diğ. (2011) bir MMORPG oyununda oyuncu kaybını tahmin etmeye yönelik gerçekleştirdikleri çalışmada probleme dair farklı algoritmaları uygulamışlardır. Aynı zamanda araştırmacılar, teori ve veri odaklı bir model olmak üzere oyuncu kaybını tahmin etmek için iki farklı modeli karşılaştırmaktadır. Teori odaklı modelin, veri odaklı model kadar başarılı olmadığını ancak yorumlanması ve anlaşılmasının daha kolay olduğu vurgusu yapılmaktadır. Veri kümesini bölümlere ayırmak ve duyarlılık (recall) değeri açısından tek sınıflandırıcıdan daha iyi performans gösteren bir topluluk modeli oluşturmak için bir kümeleme tekniği kullanmışlardır. Araştırmanın sonuçları, duyarlılık değeri açısından topluluk yöntemlerinin daha iyi performans gösterdiğini ve müşteri kayıp analizinde iyi sonuçlar verdiğini göstermektedir.

Borbora ve Srivastava (2012) tarafından gerçekleştirilen bir diğer çalışmada oyuncu yaşam döngüsü temelli yaklaşımda, oyundan ayrılma noktasına kadar geçen haftalarda oyuncuların aktivite özellikleri analiz edilmiş ve oyunda kalan oyuncunun aktivite özellikleri ile karşılaştırmışlardır. Araştırmacılar müşteri kayıp analizine dair wClusterdist kümeleme algoritmasını önermiş ve araştırmanın sonuçları, önerilen sınıflandırma şemasının bu problem formülasyonu için çok uygun olduğunu ve performansının diğer geleneksel sınıflandırma şemalarından diğerine göre daha iyi veya onlarla karşılaştırılabilir olduğunu göstermektedir.

Runge ve diğ. (2014) Diamond Dash iOS ve Monster World Flash oyun veri setleriyle müşteri kayıp tahminine yönelik sınıflandırma problemi çözümüne dair sosyal oyunlarda (social games) oyuncuların oynamayı ne zaman bırakacaklarının nasıl tahmin edilebileceğini incelemektedir. Oyuncu kaybını tahmin etmek için yapay sinir ağı, lojistik regresyon, karar ağaçları ve destek vektör makineleri algoritmalarıyla modeller geliştirilmiş ve en iyi sonuçların yapay sinir ağı ile elde edildiği ön plana çıkmıştır.

Hadiji ve diğ. (2015) tarafından, oynaması ücretsiz (Free To Play) oyunlarda kullanılan bir pazarlama modeli olan ‘freemium’u kullanan bir oyun verisi üzerinde, müşteri kayıp tahmini gerçekleştirmiştir. Çalışmada beş farklı video oyunundan derlenen veriler GameAnalytics uygulaması üzerinden toplanmıştır. Araştırmada veri üzerinde Karar Ağaçları, Lojistik Regresyon, Yapay Sinir Ağları ve Naif Bayes algoritmalarıyla geliştirilen modellere ve

sonuçlara yer verilmiştir. Kayıp olarak nitelendirilen ve nitelendirilmeyenler sınıflar üzerinde F1 skoru paylaşılmış ve verilerin değerlendirilmesi bu sonuçlara göre yapılmıştır. Araştırmanın sonuçları Karar Ağaçları (0.92) ve Yapay Sinir Ağları (0.89) ile başarılı bir performans göstermiştir.

Castro ve Tsuzuki (2015) tarafından gerçekleştirilen bir çalışmada oturum açma kayıtları kullanarak özellik seçimi için bir frekans analizi yaklaşımı önermişlerdir. Bu kayıtlar (gerçek verilerden) dört farklı yöntem kullanılarak sabit uzunlukta veri dizilerine dönüştürülmüş ve daha sonra bunlar k-En Yakın Komşular algoritması ile olasılıksal sınıflandırıcıları eğitmek için girdi olarak kullanılmıştır. Algoritmalar tahmine dayalı performans ölçütleri kullanılarak değerlendirilmiş ve karşılaştırılmıştır. Araştırmada zaman frekans düzlemi etki alanı analizi, teorik olarak RFM yaklaşımına kıyasla elde tutma kampanyalarının kârını %20'den fazla artırarak tatmin edici sonuçlar vermiştir.

Rothenbuehler ve diğ. (2015) müşteri kaybı tahmininin önlenmesi için bir uzman sistemin geliştirilmesini ortaya koymaktadırlar. Uzman sistem olarak, tahmin gerçekleştirmek için Gizli Markov Modeli (Hidden Markov Model) kullanmışlardır. Gizli Markov Modeli'nin tahmin performansı; Lojistik Regresyon, Sinir Ağı ve Destek Vektör Makinesi gibi müşteri kaybı tahmini için yaygın olarak kullanılan diğer algoritmalarla karşılaştırılmıştır. Sonuçlar, Gizli Markov Modeli'nin diğer algoritmalarından daha iyi performans göstermediğini ortaya koymaktadır.

Ding ve diğ. (2015) çalışmasında MMORPG oyunda kayıp davranışını daha iyi açıklamak için bir oyuncunun kayıp davranışının, tüm bir oyun grubunda arkadaşlarının vazgeçmesiyle ilişkili olup olmadığını incelemek adına boylamsal istatistiksel modeller ortaya koymuştur. Aktif ve aktif olmayan kişilerin içerisinde fark edilebilir oyuncuların oluşan veri seti, kişilerin yayılma davranışları arasındaki ilişkiyi üç kategoriye ayırmıştır. İki kişinin sahip olduğu ortak arkadaş sayısı arttıkça, aralarındaki etkinin daha önemli olduğunu ortaya koymaktadır.

Tsymbalov (2016) Facebook üzerinde yayınlanan 'Ghost Tales' projesinden 4 aylık bir döneme ait elde edilen verileri kullanarak sosyal çevrimiçi oyunlar için kayıp tahminine yönelik kohort tabanlı bir sınıflandırma yaklaşımı sunmaktadır. Araştırmanın modeli gerçek veriler üzerinde test edilmiş olup, kayıpların önlenmesi ile gelirden iyi bir artış sağlandığı kanıtlanmıştır. Yaklaşımın temeli, ağaç tabanlı topluluk sınıflandırıcıları ve karar sınırı ile eşik optimizasyonu

gibi bileşenleri içermektedir. Araştırmanın sonuçlarına göre, probleme dayalı tahmin uygulamasında 0.74 AUC skoru kayıp tahmini probleminde Ekstra Rastgele Orman (Extra Random Forest) algoritması elde etmiştir.

Lee ve diğ. (2016) F2P Rol Yapma Oyunu (Role Playing Game) türünde bir oyun olan Crazy Dragon adlı oyunda dengesiz veri setiyle oyuncuların kayıt verilerini kullanarak makine öğrenmesi algoritmaları ile müşteri kayıp tahmini gerçekleştirmişlerdir. Geliştirilen modelin sonuçlarına göre, SMOTE ve Rastgele orman algoritmasının birlikte uygulandığı modelde 0.87 F1 skor elde etmişlerdir. Ağaç tabanlı modellerin bu konuda iyi çalıştığını gösteren çalışmalardan biri olma niteliğindedir.

Liao ve diğ. (2016) tarafından gerçekleştirilen bir çalışma gelir ve kullanıcı davranış özelliklerini dikkate alan hibrit bir sınıflandırma modeli oluşturmaya odaklanmaktadır. Bu model test edilmiş olup geleneksel perakende ve finans sektörlerinde kullanılan mevcut müşteri kaybı tahmin yöntemleriyle karşılaştırılmıştır. Araştırma kapsamında RFM analizi gerçekleştirilirken aynı zamanda araştırmacılar tarafından hibrit bir modelde sunulmuştur. Sonuçlar, önerilen hibrit modelin sanal dünyalarda müşteri kaybını tahmin etmek için daha etkili olduğunu göstermiştir.

Bir diğer mobil platforma ait oynaması ücretsiz (F2P) modeli kullanan mobil veri seti üzerinde Drachen (2016) tarafından gerçekleştirilen çalışmada sezgisel model kullanılmıştır. Veri setinde 1 günlük ve 1 haftalık verilerin yer alması farklı dönemlerde tahmin yürütülmesine yardımcı olmuştur. En başarılı sonuçlar 10 katmanlı çapraz doğrulama ile birinci haftanın sonunda topluluk yöntemleriyle 0.71 F1 skor değeri ile elde edilmiştir.

Tamassia ve diğ. (2016) MMORPG (çevrimiçi çok oyunculu) birinci şahıs nişancı oyunu olan Destiny üzerinde oyuncuların oyunu ne zaman bırakacaklarını tahmin etmenin zorluğunu tartışmışlardır. Ayrıca, oyuncuların oyunu ne zaman bırakacaklarını tahmin edebilecek bir model oluşturmak için Gizli Markov Modelleri (Hidden Markov Models) adı verilen bir tür matematiksel model kullanmışlardır. Araştırmanın sonuçlarına göre, Doğrusal Diskriminant Analizi (Linear Discriminant Analysis), Lojistik Regresyon ve Rastgele Orman algoritmalarında 0.63 F1 skor oranı elde edilmiştir. Aynı zamanda Gizli Markov Modellerinin 0.77 AUC değeri ile diğer algoritmalara göre daha başarılı bir performans gösterdiği belirtilmiştir.

Saas ve diğ. (2016) oyuncuların oyunla nasıl etkileşime girdiği ve oyuncu kayıp tahminini daha iyi anlamak için aynı gruptaki oyuncuların dinamiklerini incelemek amacıyla gerçekleştirdikleri çalışmada oynaması ücretsiz oyunlarda kümeleme analizi üzerinde çalışmıştır. Çalışmada belirlenen oyunlar kapsamında oyuncu davranışlarıyla ilgili verileri kümeleme için en iyi yöntemlerin hangileri olduğu araştırılmıştır. Son olarak, yöntemlerin ne kadar iyi çalıştığını görmek için belirlenen veri setleri üzerinde karşılaştırmalı analiz gerçekleştirilmiştir. Araştırmanın sonuçları, Cox regresyonun doğruluk ve ayırıcılık olarak ön plana çıktığını göstermektedir.

Liu ve diğ. (2017) çalışmasında, mobil oyunlarda kullanıcı kaybını tahmin etmek için yeni bir çözüm sunmaktadır. Bu çözüm, bağlamsal bilgileri ve ilişki dinamiklerini yakalamak için derin sinir ağlarıyla yeni bir yarı denetimli ve tümevarımlı gömme modeline dayanmaktadır. Yazarlar, kayıp tahmininin performansını test etmek için Samsung Game Launcher platformundan çok sayıda oyun ve milyonlarca kullanıcı-uygulama etkileşimi içeren gerçek verileri toplamışlardır. Bu verilerle yapılan deneylerin sonuçları, önerilen modelin mevcut yöntemlerden daha iyi olduğunu göstermiştir.

Ha-bin ve diğ. (2017) Blade ve Soul veri seti üzerinde kullanıcılar, oyun botu, otomatik avlanma programları, kötü amaçlı kodlar vb. nedeniyle oyuncuların kötü bir deneyim yaşadıklarında oyunu terk etme eğiliminde olduklarını vurgulayarak, oyun operatörleri bu tür olayları zamanında tespit etmezse, kaybın çevrimiçi oyunun kârına ve hizmetin uzun ömürlülüğüne zarar verebileceğini dile getirmiştir. Veri analizinin verimliliğini arttırmak için yeni bir PNG görüntü dönüştürme tabanlı kayıp tahmini tekniği önerip, sıkıştırma tekniğini kullanarak günlük dosyalarının boyutunu 52.849 kat daha küçülterek ve özellik analizi olmadan analiz hızını arttırmayı hedeflemişlerdir. Sonuçlar, potansiyel müşteri kaybının %97 gibi yüksek bir doğrulukla tespit edildiğini göstermektedir.

Xue ve diğ. (2017) çalışmasında, oyuncunun oyuna katılımını en üst düzeye çıkarmak için tasarlanmış oldukları Dinamik Zorluk Ayarı (Dynamic Difficulty Adjustment) adlı bir çerçeve önermektedir. Bu çerçeve önerilirken, seviye tabanlı oyunlar örnek olarak kullanılmış ve bir oyuncunun oyun boyunca ilerlemesi olasılıksal bir grafik olarak modellenmişlerdir. Bu DDA çerçevesi, oyuncuyu mümkün olduğunca uzun süre oyuna bağlı tutmak için seviyeler arasındaki geçiş olasılıklarını optimize ederek çalıştırmayı hedeflemektedir. Yazarlar bu sistemi

Electronic Arts, Inc. şirketinin birçok oyununda test etmiş ve oyunun para kazanma oranını etkilemeden oyuncu katılımında %9'a varan bir iyileşmeye gidildiğini gözlemlemişlerdir.

Kim ve diğ. (2017) mobil ve çevrimiçi oyunlardan olan Dodge the Mud, Undisclosed ve TagPro'ya ait veri setleri üzerinde makine öğrenmesi yöntemleri ile müşteri kayıp tahmini gerçekleştirmişlerdir. Araştırmanın sonuçları, Dodge the Mud adlı oyunda 0.79 F1 skorla LSTM algoritması , Undisclosed adlı oyunda 0.73 F1 skorla Gradient Boosting Sınıflandırıcı ve TagPro adlı oyunda 0.82 F1 skor oranlarıyla Lojistik Regresyon ve Gradient Boosting algoritmaları ön plana çıkmaktadır.

Bertens ve diğ. (2017) Age of Istharia adlı oyun için müşteri kayıp tahmini üzerine yeni bir yaklaşım önermişlerdir. Önerilen yöntemde, her oyuncunun oyundan ne zaman ayrılacağını ve o noktaya kadar ne kadar süre oynayacağını doğru bir şekilde tahmin edilmesine odaklanılmaktadır. Araştırma kapsamında hayatta kalma süresi veya olaya kadar geçen süre olarak bilinen, bir olayın gerçekleşmesine kadar geçen süreyi tahmin etmek için kullanılan makine öğrenimi algoritmaları olan Survival Ensemble ve bir veya daha fazla ortak değişkenin olaydan olaya veriler üzerindeki etkisini modellemek için kullanılan yarı parametrik bir model olan Cox Regression modelleri kullanılmıştır. Sonuçlar, bir oyuncunun ne zaman ayrılacağını ve ne kadar süre oynayacağını doğru bir şekilde tahmin edebildiğini göstermiştir. Öte yandan, Cox regresyonu nispeten iyi performans gösterdiği ancak çok fazla ön işleme süreci gerektirdiği vurgulanmıştır.

Milosevic ve diğ. tarafından (2017) yine mobil bir oyun olan 'Top Eleven - Be a Football Manager' verisiyle gerçekleştirilen çalışmada şirket, kullanıcı kaybını önlemek için kullanıcı faaliyetlerini izlemiş ve kullanıcının ilgisini çekebilecek özellikleri belirlemiştir. Ardından, kullanıcıya kişiselleştirilmiş anlık bildirimler gönderilerek oyuna geri dönmesi sağlanılmaya çalışılmıştır. Araştırmanın sonuçlarına göre bu yaklaşım, kullanıcı kaybını %28'e kadar azaltmada başarılı olmuştur ve bu da işletme için büyük bir avantaj olarak geri dönmüştür. Aynı zamanda müşteri kaybını tahmin etmek için çeşitli algoritmaların performansları değerlendirilmiş ve Gradient Boosting modelinin en iyi performansı gösterdiği tespit edilmiştir. Son olarak, Lojistik Regresyon'un basit ve ölçeklenebilir bir model olduğu düşünüldüğünde başarılı bir performans göstermiştir.

Jeon ve diğ. (2017) müşteri kayıp tahmini üzerine gerçekleştirdiği çalışmada, Crazy Dragon adlı bir mobil RPG oyununun oyun kayıtlarını incelemişlerdir. Oyun kayıtlarından yedi adet bilişsel-psikolojik özellik çıkarmışlar ve bunları oyuncuların oyunu ne zaman bırakacaklarını tahmin etmek için kullanmışlardır. Ayrıca satın alma özelliklerinin müşteri kayıp tahmini üzerindeki performans etkisini de incelemişlerdir. Sonuçlar, bilişsel-psikolojik özelliklerin kullanılmasının temel ham kayıt verilerini kullanmaktan daha etkili olduğunu ve satın alma özelliği olmadan da yüksek bir oyundan ayrılma tahmin performansı elde edebildiklerini ortaya koymuştur.

Perianez ve diğ. (2017) Google Play ve iOS markette yayınlanan ve geniş kitlelere hitap eden oynaması ücretsiz (F2P) tabanlı bir mobil oyuna ait verilerle müşteri kayıp analizi üzerine yeni bir model geliştirmişlerdir. Bu model ‘hayatta kalma topluluğu’ (Survival Ensemble) oyuncuların modeli olarak adlandırılmış ve bir oyuncunun zaman içinde oyundan ayrılma (churn) olasılığını tahmin etmek için kullanılmıştır. Ayrıca model, bir oyuncunun kalma ya da ayrılma kararını etkileyen faktörleri de belirlemeye yönelik çözümler sunmaktadır. Çalışmanın sonuçları, hayatta kalma topluluğu modelinin Cox regresyonu gibi geleneksel yöntemlerden daha doğru ve güvenilir olduğunu göstermiştir.

Woh ve Lee (2018) ‘Soulgace’ adlı oyuna ait veri seti üzerinde müşteri kayıp tahmini gerçekleştirmişlerdir. Oyunda müşteri kayıp tahmini problemine çözüm olarak araştırmacılar, kullanıcıların oyunda nasıl davrandıklarını gösteren, kaç kişinin oynadığı ve ne kadar para harcadıkları gibi bir dizi değişken olan bir oyun kayıt dosyası tanımlamışlardır. Bir takım makine öğrenmesi algoritmaları, problem çözümünde kullanılmış ve araştırmanın sonuçları derin öğrenme ve ağaç tabanlı makine öğrenmesi yöntemlerinin üzerine kurgulanmıştır. Derin öğrenme kısmında RNN, CNN ve LSTM katmanlı bir sinir ağı yapısı oluşturarak bir tahmin yürütmüş ancak yeterli seviyede tahmin yapacak bir başarı oranına ulaşamamıştır. Aynı zamanda araştırmacılar ağaç tabanlı modellerin derin öğrenmeye göre daha iyi sonuç verdiğini ve hata oranında oldukça düşük yüzdelere ulaştığını dile getirmiştir.

Kummer ve diğ. (2018) 10 GB boyutunda MMORPG olan Blade and Soul oyununun verilerine dair sınıflandırma problemi olarak müşteri kayıp tahmini ve regresyon problemi olarak hayatta kalma süresi tahmini gerçekleştirmişlerdir. Araştırmanın sonuçlarına göre, sınıflandırma probleminde Test 1 ve Test 2 veri seti ayrı olmak üzere ortalama test puanlarında en iyi sonucu

0.72 RepTREE ve C4.5 algoritmaları sağlarken, regresyon probleminde ise 0.44 ile RepTREE algoritması en iyi başarı gösterenler arasında ilk sırada yer almaktadır.

Lee ve diğ. (2018) çalışmasında, uzun vadede kâr optimizasyonu sağlayan kayıp tahminine yönelik çevrimiçi oyunlarda sadık müşteriler üzerine odaklanmıştır. Çevrimiçi oyunlarda kullanıcı kaybını tahmin etmek için bir yöntemi tartışmaktadır. Araştırmanın sonuçları, gerçek dünyadaki bir çevrimiçi oyun hizmetine uygulanmış ve kampanya maliyeti ile dönüşüm oranı dikkate alındığında toplam kullanıcılar için tahmin modelinden daha uygun maliyetli olduğu gösterilmiştir.

Kwon ve diğ. (2018) çalışmasında, Princess Maker adlı oyunda oyun-içi kayıtların oyuncuların davranışlarını nasıl etkilediğini analiz etmek ve oyuncuların oyundan ne zaman ayrılacaklarını tahmin etmek için geliştirdikleri modelde, sonuçlar önerdikleri yöntemin oyun içi olayların oyuncuları nasıl etkilediğini anlamak ve oyuncuların oyundan ne zaman ayrılacağını tahmin etmek için kullanılabileceğini göstermektedir. Zaman serisi verilerinin kümelenmesi ve DBSCAN+LSTM algoritmalarıyla birlikte 0.97 F1 skora ulaşmışlardır.

Wang, Xu ve Hussain (2018) çalışmasında, bir müşterinin gelecekte oyundan ayrılıp ayrılmayacağını tahmin etme üzerine gerçekleştirdikleri araştırmada, Gradyan Artırma Karar Ağacı (Gradient Boosting Decision Tree) adı verilen bir model kullanmışlardır. Modeli büyük ölçekli bir müşteri veri seti üzerinde test ettikten sonra, statik ve dinamik özelliklerin tamamlayıcı olduğunu ve modelin test seti üzerinde 0,84 AUC (ROC eğrisi altındaki alan) değerine ulaştığını tespit etmişlerdir. Bu modelin, yakın gelecekte hangi müşterilerin reklam platformundan ayrılacağını tahmin etmek için kullanışlı olduğunu dile getirmişlerdir.

Drachen ve diğ. (2018) çalışmasında, sosyal faktörlerin mobil oyunlar üzerindeki etkisine odaklanmasını tartışmaktadırlar. Sosyal faktörlerin genel olarak oyunlarda elde tutma açısından önemli olduğu gösterilmiş olsa da mobil oyunlarda elde tutma ve para kazanma üzerindeki etkisinin iyi anlaşılmadığı belirtilmektedir. Araştırma kapsamında sınıflandırma ve regresyon problemlerine çözüm üretmek için 200.000 kullanıcının bilgisini içeren bir veri seti kullanılmıştır. Sonuçlar, sosyal aktivitenin kullanıcıların premium kullanıcı olup olmayacağı üzerinde doğrudan bir etkiye sahip olmadığını ancak sosyal aktivitenin bir grup kullanıcı içinde zamanla arttığını göstermiştir.

Bergstorm (2018) Facebook tabanlı oyunlar olan YoWorld ve YoVille verilerini kullanarak gerçekleştirdiği çalışmada, geçmişinde sıklıkla Social Network Game (Sosyal Ağ Oyunu) oynayan 147 oyuncuya anket yönlendirmesi gerçekleştirmişlerdir. Anketin bulguları oyuncuların oyunu oynamayı genellikle artık oyunla ilgilenmedikleri için değil, boş zamanlarını kısıtlayan dış faktörler nedeniyle bıraktıklarını göstermiştir.

Demediuk ve diğ. (2018) belirli davranışsal özelliklerin bir oyuncunun League of Legends oyununu oynamaya devam etme olasılığı üzerindeki etkisini anlamak amacıyla gerçekleştirdiği çalışmada, oyuncu kaybını tahmin etmek için Cox regresyonun farklı bir modeli olan Karma Etkiler Cox Regresyonu (Mixed Effect Cox Regression) kullanılarak hayatta kalma analizinin uygulanmasına dayanan bir çözüm sunmuştur. Oyundan ayrılma riski taşıyan oyuncuların özelliklerinin değerlendirilmesinin yanı sıra oranları da sağladığından oyundan ayrılma tahmini problemi için faydalı bir yaklaşım oluşturmaktadır.

Bertens ve diğ. (2018) çalışmada, bir oyuna dair tavsiye motoru (recommendation engine) oluşturmak için kullanılabilecek iki algoritma önermişlerdir. Bu motor, oyunculara çekici bulmaları öngörülen öğelerin önerilmesi üzerine kurgulanmıştır. Topluluk tabanlı bir model ve derin bir sinir ağı kullanılarak gerçekleştirdikleri tavsiye sisteminde bir oyuncunun hangi öğeleri beğeneceğini tahmin etmede başarılı olacağını, aynı zamanda gerçek dünya ortamlarında kullanılabilir olması kadar hızlı ve güvenilir olduğunu da ortaya koymuştur.

Kristensen ve Burelli (2018) 'Cookie Cats Pop' adlı oyunun verileriyle müşteri kayıp tahmini üzerine model geliştirdiği çalışmada iki tür veriyi birleştirerek, bir oyuncunun zaman içindeki davranış değişikliklerine bakan sıralı veriler ve bir oyuncunun genel davranışına bakan toplu verileri analiz etmişlerdir. Bu iki veri türünü birleştirmek için farklı türde sinir ağları kullanmış ve analizlerinin sonuçları, iki veri türünü birleştirmenin her iki veri türünü tek başına kullanmaktan daha doğru tahminlere yol açtığını göstermiştir. Dört farklı sinir ağı mimarisini tahmin probleminde kullanan araştırmacılar, sonuçların LSTM tahmini veya aktivasyonlarla birleştirilmesinin temel çizgiye göre bir gelişme gösterdiğini ortaya koymaktadır. Aynı zamanda, LSTM temelli farklı modeller üzerinde 0.87 AUC skora ulaşmıştır.

Chen ve diğ. (2018) çalışmada müşteri Yaşam Boyu Değeri (Customer Lifetime Value) tahmin etmek için kullanılan matematiksel modellerden Pareto/NBD gibi parametrik modelleri karşılaştırmışlardır. Sonuçlar, konvolüsyonel sinir ağlarının bireysel oyuncuların ekonomik

değerini tahmin etmede en etkili yöntem olduğunu göstermek ile birlikte regresyon tahmininde ise, Kök Ortalama Karekök Logaritmik Hatası (RMSLE) hata türünde CNN modeliyle %3.96 hata oranı elde etmişlerdir.

Yang ve diğ. (2019) çalışmasında, 'Thirty-Six Stratagems' ve 'Thirty-six Stratagems Mobile' oyunlarına dair oyuncuların ücretsiz bir çevrimiçi oyununda oyuncunun kayıp tahmini üzerine gerçekleştirmişlerdir. Çalışmada, oyuncuların oyunda ne kadar zaman geçirdiklerinin düzen oranına bakılmış ve bu bilgiyi oyuncuların ne zaman ayrılacağını tahmin etmek için kullanılabilecek öznitelikler oluşturmak için kullanmışlardır. Aynı zamanda araştırmacılar ham veriler ve öznitelik mühendisliği yaptıkları sonuçları karşılaştırmış ve her iki veri kümesin de Lojistik Regresyon algoritmasının sırasıyla 0.71 ve 0.78 AUC değerlerini ortaya koyduğu görülmüştür.

Bonometti ve diğ. (2019) 5 farklı veri seti kullanarak gerçekleştirdiği çalışmasında bir kullanıcının oyun ile ne kadar meşgul olduğunu tahmin etmek için kullanılabilecek teorik bir tanı ortaya koymuştur. Bu yaklaşımın, bir kullanıcının bir oyunu ne kadar süre oynayacağını ve oyunu oynamayı bırakma olasılığını tahmin etmek için kullanılabileceğini göstermek adına deneyler yapılmıştır. Araştırmanın sonuçları, MLP algoritmasında LSTM katmanı kullanarak başarı oranının 0.70 bulunduğu belirtilmiştir.

Guitart ve diğ. (2019) oyunda kayıp ve "satın alma vazgeçmesi" (ödeme yapan kullanıcıdan ödeme yapmayan kullanıcıya geçiş) olarak adlandırılan duruma ilişkin derin bir davranışsal analiz sunduğu çalışmasında, oyuncuların belirli bir süre oyunu bırakma ve geri dönme sürelerini izlemişlerdir. Araştırma kapsamında hayatta kalma modelleri kullanarak gerçek ve yanlış kayıplara dair istatistiksel veriler ortaya koymuşlardır. Sonuçlar, oyun oynama süresi arttıkça hata oranlarının artış gösterdiği ancak belirli bir süre sonra hata oranının durduğunu göstermektedir.

Jang ve diğ. (2019) çalışmasında 'MAZE X BRAVE' adlı oyun verilerini kullanarak oyunda müşteri kaybını tahmin etmede kendi oluşturdukları çıkma vektörüyle tahmin modelleri oluşturmuşlardır. Bir kayıp vektörünün kullanım avantajlarının iki yönlü olduğunu, kayıp tahmininin doğruluğunu artırabileceğini ve kullanıcı kaybını azaltmak için promosyon kampanyalarının yardımcı olabileceğine dikkat çekmiştir. Araştırma kapsamında Lasso, SVM, NN, DT, RF ve GB algoritmalarını kullanarak regresyon ve sınıflandırma problemine dair bir

tahmin gerçekleştirmişlerdir. NN ve DT için 0.95 R^2 açıklanabilir değerine ulaşmışlardır. Vektörel verilerin gün bazında oluşturulan verilere göre analizde sınıflandırma probleminin yapay sinir ağları 0.95 başarı oranı ile çok iyi denilebilecek bir başarıya ulaşmıştır.

Kilimci ve diğ. (2020) mobil oyunlarda müşteri kaybını tahmin etmek için yeni bir model sunmuştur. Bu model, oyuncuların oyunlarla ilgili yorumlarından oyuncu duygularını analiz etmek için derin öğrenme algoritmaları ve Kelime Gömme Modelleri (Word Embeddings Models) kullanmıştır. Sınıflandırma için konvolüsyonel sinir ağları, tekrarlayan sinir ağları ve uzun kısa süreli bellek ağları gibi derin öğrenme algoritmaları kullanılmıştır. Metin gösterimi için Word2Vec, GloVe ve FastText kullanılmıştır. Önerilen modelin etkisini göstermek için dört farklı Türkçe oyun veri kümesi üzerinde deneyler yapılmıştır. Sonuçlar, mobil oyunlardaki oyuncuların duygu analizinin müşteri kaybını tahmin etmek için güçlü bir gösterge olabileceğini göstermiştir.

Roohi ve diğ. (2020) ‘Angry Birds Dream Blast’ adlı bir oyuna geçen ve oyunu bırakan oyuncuların oranını tahmin edebilen yeni bir simülasyon modeli önermektedir. Bu model, Yapay Zeka (YZ) oyununu, oyuncu nüfusunun zaman içinde nasıl değiştiğine dair bir simülasyonla birleştirmektedir. YZ oyuncuları seviyelerin zorluğunu ölçmekle nitelendirilmiş ve bu da oyuncuların beceri (skill), niyet (intention) ve bıkkınlıklarını (boredom) dikkate alan bir oyuncu popülasyonu modeli oluşturmak için kullanılmıştır. Araştırmacılar, bu modelin ayrıca farklı beceri seviyelerine sahip oyuncuların seviye zorluğundan nasıl etkilendiğini ve bunun oyuncuların oyunu bırakma oranını nasıl etkilediğini açıklamak için kullanılabileceğini savunmuştur.

Rothmeier ve diğ. (2020) ‘The Settlers Online’ adlı bir oyunun veri üzerinde geçmişe yönelik oyuncuların oyundan ne zaman ayrıldığını tespit etmek için dört farklı yöntem kullanmışlardır. Müşteri kayıp tahmini için sekiz farklı makine öğrenimi algoritması kullanılmıştır. Araştırmacılar Rastgele Orman algoritmasının en iyi çözümü sunduğunu ve %97’lik bir doğruluk oranı yakaladığını dile getirmişlerdir.

Ahn ve diğ. (2020) çeşitli endüstrilerde ve akademik alanlarda kullanılan müşteri kaybı tahmin tekniklerini karşılaştırdıkları çalışmada; işletme, pazarlama, bilgi teknolojileri, telekomünikasyon, gazete, sigorta ve oyun gibi alanlarda müşteri kaybı tanımlarını bir araya getirerek aralarındaki farkları açıklamışlardır. Ayrıca açıklamaları kayıp, özellik mühendisliği

ve tahmin modellerini sınıflandırarak detaylandırmışlardır. Farklı sektörlerden ve akademik alanlardan farklı kayıp çalışmalarını birleştirerek araştırmacıların ilgilendikleri hizmet alanı için en uygun kayıp tanımına ve ilişkili modelleri seçmelerine yardımcı olabilecek analizleri sunmuşlardır.

Zheng ve diğ. (2020) ‘ChurnPred’ adı verilen yeni bir sinir ağı türü önermişlerdir ve bu ağ müşteri kayıp tahmini üzerine yoğunlaşmaktadır. Kaybı tahmin etmek için, kullanıcı davranış kalıplarını daha iyi anlamak amacıyla hem oturum kayıtlarını hem de oyun içi davranışları incelemişlerdir. Oturum açma kayıtlarında uzun-kısa süreli bellek (LSTM) algoritmasını kullanmışlardır. Oyun içi davranışlar için belirli bir dönemde ortaya çıkan kalıpları belirlemek üzere zamana duyarlı bir filtreleme bileşeni ve bu davranışların farklı perspektiflerden çoklu kombinasyonlarını çıkarmak için Çoklu Görünüm yöntemi (Multi View Method) uygulanmıştır. Modeli gerçek dünya verileri üzerinde test etmiş ve modelin mevcut yöntemlerden daha etkili olduğunu ortaya koymuşlardır.

Charles ve Cowley (2020) müşteri kayıp tahminine yönelik davranışsal veriler üzerinde ‘Behavlets’ adı verdikleri oyuncu kaybını tahmin etme ve oyuncunun oyundan aldığı keyfi artırmak için oyunu uyarlamak adına nasıl kullanılabileceğine odaklanan bir model geliştirmiştir. Çalışmada araştırmacılar, geliştirilen modelin teorik yapısını açıklamış ve müşteri kayıp analizinde kullanılabilecek bazı öneriler dile getirmişlerdir.

Yang ve diğ. (2022) çevrimiçi oyunda müşteri kayıp tahmini üzerine gerçekleştirdikleri çalışmada, entropi (entropy) ve çapraz entropi (cross entropy) adı verilen bilgi teorisinden bir kavram kullanarak oyuncuların oyun oynama sürelerinin düzenliliğini ölçmüşlerdir. Daha sonra bu bilgileri, oyuncuların ne zaman oyundan ayrılacağını tahmin etmek için uygulanabilen veri öznitelikleri oluşturmak için kullanmışlardır. Son olarak, bu özniteliklerin performansını mevcut özniteliklerin performansı ile karşılaştırmış ve önerilen özelliklerin oyuncu kaybını tahmin etmede daha etkili olduğu sonucuna ulaşmışlardır.

Loria ve Marconi (2021) oyuncuların oyunda diğer bireylerle etkileşime girdikleri değişkenlerin, oyunda müşteri kayıp tahmini yapmak için kullanılıp kullanılamayacağını incelemişlerdir. Sonuçlar, verilerin oyuncuların oyunu ne zaman bırakacaklarını tahmin etmek için kullanılabileceğini göstermiştir. Ayrıca, bir oyuncu oyunu oynayarak ve sistemi kullanarak

ne kadar uzun süre geçirirse, oyundan ayrılma olasılığının o kadar yüksek olduğu sonucuna ulaşmışlardır.

Jang ve diğ. (2021) oyunda müşteri kayıp tahminine yönelik olarak bireysel kullanıcıların kullanım süresini dikkate alan bir yöntem ortaya koymaktadır. Çalışma, bir kullanıcının ne zaman kayıp olarak nitelendirileceğine %96,6 oranında doğru bir şekilde tahmin edebildiğini göstermektedir. Ayrıca elde edilen sonuçlarda kullanılan veri setlerinin sonuçları içinde en iyi performans gösteren yöntem olduğunu dile getirmişlerdir.

Karmakar (2021) bir oyunda oyuncuların kaybını analiz etmek için istatistiksel bir çerçeve sunmaktadır. Oyuncuların katılım süreleri, iş birliği ve oyunun her seviyesinde oluşan başarı puanları gibi birçok özelliği incelemiş ve bu özellikler ile oyundan ayrılma oranı arasındaki karşılıklı bağımlılıkları yakalamak adına genelleştirilmiş bir doğrusal karma model (GLMM) kullanmıştır. Modeldeki parametreleri tahmin etmek için büyük veri setlerini işleyebilen böl ve yeniden birleştir yaklaşımı olan bir Hamiltonian Monte Carlo algoritması kullanmışlardır. Araştırmanın sonuçları, geliştirdikleri modeli mevcut modeller ile karşılaştırarak kendi modellerinin kaybı tahmin etmede daha iyi olduğunu göstermiştir.

Su ve diğ. (2021) oyun endüstrisinde İş Zekasının (Business Intelligence) kullanımını, özellikle de oyun analitiğini incelemiştir. Oyun analitiğinin beş yönünü geleneksel oyun değer zincirine dayalı olarak literatürdeki oyun analitiği yönlerini keşfetmek, oyun endüstrisinde analitik kullanmanın temel amaçlarını bulmak, oyun analitiği kullanılarak ele alınabilecek oyun alanındaki sorunları veya zorlukları sunmak, tahmin için oyun analitiğinde kullanılan farklı algoritmaları listelemek ve literatürde zaten ele alınmış ancak daha fazla geliştirilmesi gereken araştırma alanlarını özetlemek olarak sıralamaktadır. Ayrıca araştırmacılar oyun analitiğinin sınırlamalarını tartışmakta ve gelecekteki araştırmalar için potansiyel araştırma noktaları önermektedir.

Biçer (2021) yüksek lisans tez çalışmasında basit oyun (hyper casual) adı verilen mobil oyun türünü oynamayı bırakma nedenlerini incelemiştir. Oyunu bırakma nedenlerini belirlemek için Google ve iOS platformlarında sunulan hyper-casual türünde bir oyundan elde edilen verileri kullanılmıştır. Bir kişinin yedi gün boyunca oyunu oynamadığı takdirde "kayıp" olarak nitelendirmiş olup devamında sonuçları karşılaştırmak ve hangi faktörlerin oyunu bırakmaya neden olduğunu tahmin etmek adına farklı makine öğrenimi algoritmaları kullanmıştır.

Araştırmanın sonuçlarına göre, Karar Ağaçları ve Rastgele Orman algoritmaları %89 başarı oranıyla en iyi performans göstermiş, ardından %88 başarı yüzdesiyle KNN algoritması ve devamında da Lojistik Regresyon ile Naive Bayes algoritmaları gelmiştir.

Mustac ve diğ. (2022) oynaması ücretsiz bir mobil oyunda, makine öğrenmesi yöntemleri ile müşteri kayıp tahmini gerçekleştirmiştir. Araştırma, Avrupalı bir oyun geliştiricisinden dört yıl boyunca toplanan ve çoktan seçmeli hikaye anlatımı türünde bir oyun oynayan oyuncuların kayıtlarını içeren verilere dayanmaktadır. Araştırmanın sonuçları, çok sayıda oyuncunun oyunu yalnızca kısa bir süre oynadığını ve bunun da gelir kaybına yol açabileceğini doğrultudur.

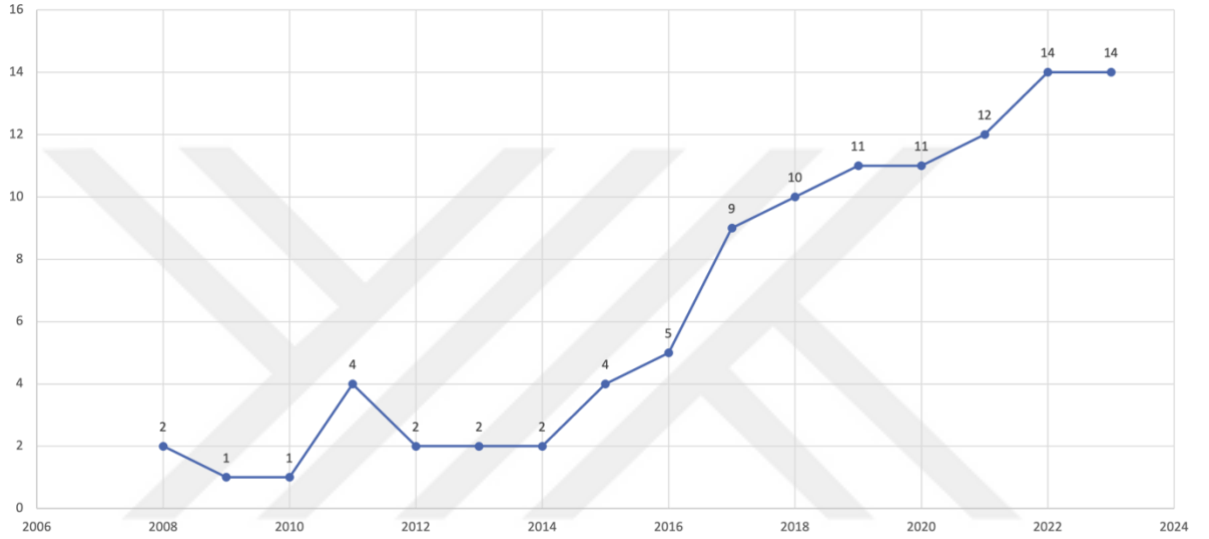
Kristensen ve Burelli (2022) bir oyuncunun oyunu ne zaman bırakacağını tahmin etmenin doğruluğunu artırmanın yollarını araştırdıkları çalışmalarında, iki tür veriyi birleştirerek bir oyuncunun davranışında zaman içinde meydana gelen değişikliklere bakan sıralı verileri ve bir oyuncunun genel davranışına bakan toplu verileri belirlemişlerdir. Bu iki veri türünü birleştirmek ve sonuçları karşılaştırmak için farklı sinir ağı mimarileri kullanılmış ve karşılaştırmalı sonuçlar, iki veri türünü birleştirmenin, her iki veri türünü tek başına kullanmaktan daha sağlıklı ve başarılı bir yöntem olduğunu ortaya koymuştur.

Zhao ve diğ. (2022) çevrimiçi oyunlarda müşteri yaşam boyu değerini (CLTV) tahmin etmenin zorluklarını tartıştıkları çalışmada, müşteri kayıp tahmini probleminde çoğu önerilen yöntemin ortaya koyduğu çözümlerden sadece birine odaklandığını ve ikisi arasındaki korelasyonu dikkate almadıklarını dile getirmiştir. Ayrıca, çoğu çözümün geçmiş verilere dayandığını ve gerçek dünya testleri olmadan yeterince açıklanabilir olmadığını dile getirmiştir. Bu sorunları ele almak için araştırmacılar, farklı türlerdeki oyunlar arasında CLTV'yi modelleyen 'perCLTV' adlı bir sistem önermektedir. Gerçekleştirilen deneyler ve çevrimiçi A/B testleri, perCLTV'nin etkili olduğunu ve diğer yöntemlerden daha iyi performans gösterdiğini ortaya koymuştur.

Canbek (2022) yüksek lisans tez çalışmasında makine öğrenmesi yöntemleri ile oyun endüstrisinde para kazanma kavramı üzerine gerçekleştirdiği araştırmasında, bir oyuncunun oyunda ne kadar para harcayacağına dair tahmin modeli geliştirmek için sıradan bulmaca oyunundan alınan verileri kullanmıştır. Araştırmanın sonuçları, sıradan oyun sağlayıcıları tarafından bir oyuncunun gelecekte ne kadar para harcayacağını tahmin etmek ve ürünlerini

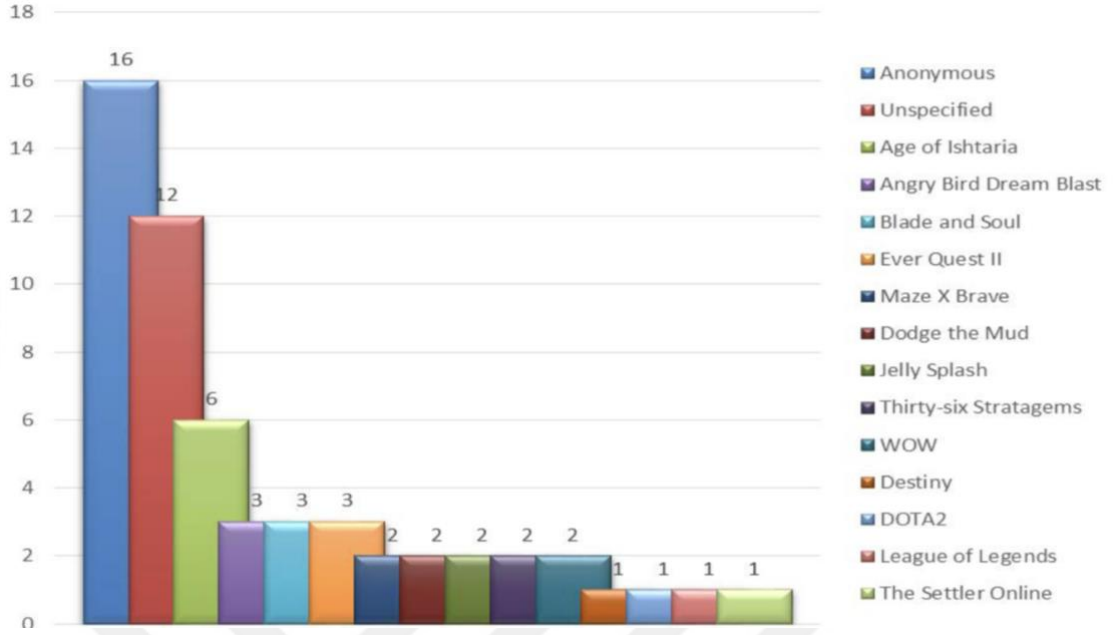
geliştirmek için hangi stratejileri kullanacaklarına karar vermelerine yardımcı olmak için geliştirilen modelin kullanılabileceğini göstermektedir.

Oyunda müşteri kayıp analizinde yer alan alanyazın çalışmalarının sayısına, veri setleri kapsamında kullanılan oyunların isimlerine, kullanılan algoritmaların ve hedef probleme yönelik türlerine ve platform bazında frekans grafiklerine aşağıda yer verilmiştir (Arık ve diğ., 2022).



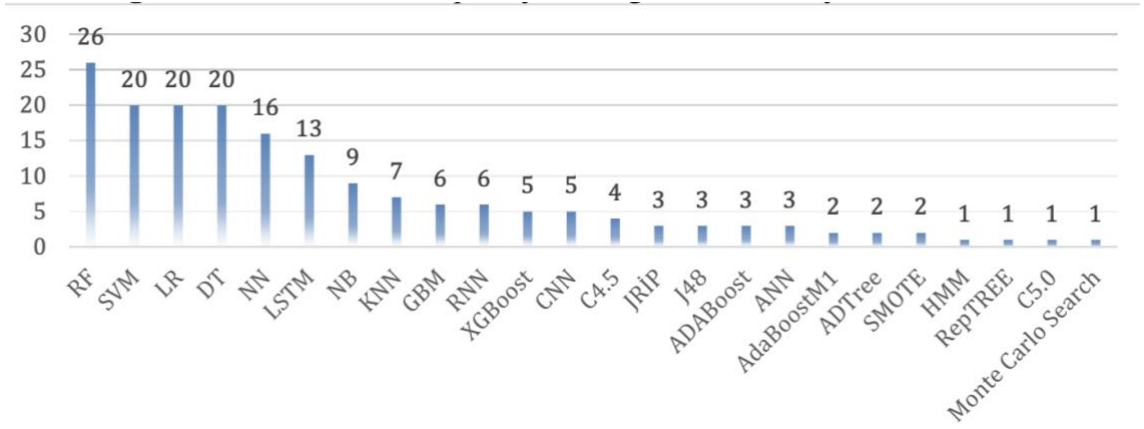
Şekil 2.19: Yıl bazında oyun kayıp tahmini üzerine yayınlanmış çalışma sayısı

Şekil 2.19’da 2008 yılından bu yana oyun müşterisi kayıp tahmini üzerine çalışmaların frekans dağılımına yer verilmiştir. 2016 yılına kadar az sayıda çalışma yayınlanırken, 2016 yılından sonra çalışma sayısındaki artış görülmektedir. Son yılların popüler konularından biri olan oyunda müşteri kayıp analizi, her geçen gün akademik çalışmaların arttığı önemli bir alan haline gelmiştir. Bu çalışmaların önümüzdeki yıllarda daha da artacağı söylenebilir.



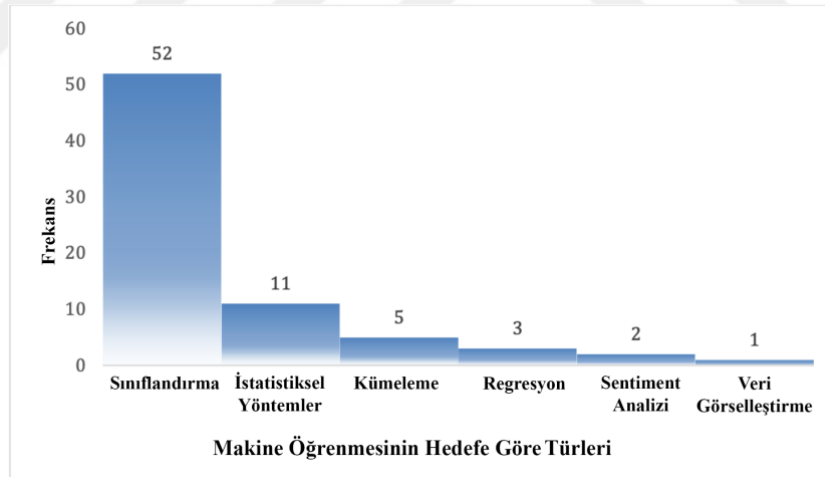
Şekil 2.20: Araştırmalarda kullanılan veri setlerinin oyun isimlerine göre dağılımı

Şekil 2.20’de görüldüğü gibi en çok tekrar edenleri ($n=16$ ve $n=12$) isimlerini etik kaygılardan dolayı paylaşmayan anonim veya tanımlanmamış oyunlar oluşturmaktadır. Ardından “Age of Istharía” ($n=6$) ve aynı frekansa sahip olan “Angry Birds”, “Blade and Soul”, “Everquest II” ($n=3$) gibi oyunlar sıralamayı takip etmektedir. Diğer oyunlar bir veya iki kere çalışmalar kapsamında kullanılmış olup, bu durum oyunda müşteri kayıp analizinde veriye ulaşmanın kısıtlı olduğunu da ortaya koymaktadır. Daha önceki bölümlerde bahsedildiği üzere şirketler veriyi açık şekilde paylaşmamakta ve bu da bu alanda yapılacak çalışmaları kısıtlamaktadır.



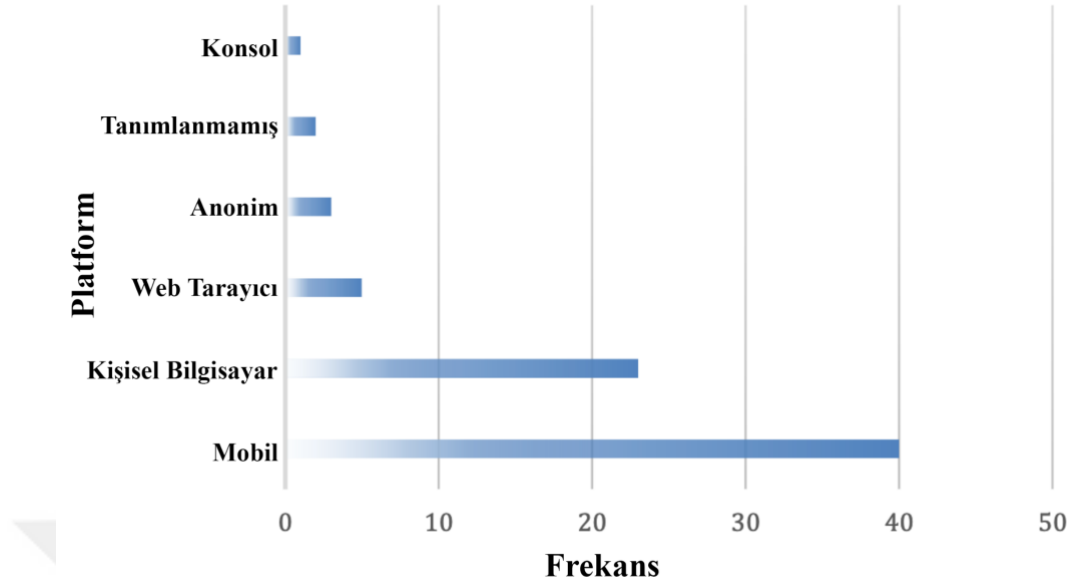
Şekil 2.21: Sınıflandırma probleminde kullanılan algoritmaların dağılımı

Oyunda müşteri kayıp analizinde en iyi performans gösteren algoritmalar Şekil 2.21’de grafik haline getirilmiştir. En iyi performans gösteren algoritmalar sırasıyla Rastgele Orman, Destek Vektör Makineleri, Lojistik Regresyon, Karar Ağaçları ve Yapay Sinir Ağları olarak sıralanmaktadır. Bu algoritmalar aynı zamanda makine öğrenmesinde en sık kullanılan ve iyi tahmin yapabilen algoritmalarlardır.



Şekil 2.22: Araştırmalarda kullanılan hedefe yönelik makine öğrenmesi türlerinin dağılımı

Özellikle yayınlanan araştırmalarda sınıflandırma yaklaşımının açık ara önde olduğu Şekil 2.22’de görülmektedir.



Şekil 2.23: Çalışmaların oyun platformuna göre frekans grafiği

Mobil oyun sektörü, özellikle akıllı telefonların dünya çapında yaygınlaşmasından bu yana önem kazanmıştır. Şekil 2.23'te gösterildiği gibi mobil oyun verilerine dayalı modeller daha ağırlıklı oluşturmuştur. Bilgisayar ve web tarayıcılarında oynanan oyunlar, mobil platformda ve bilgisayar ve web tarayıcı platformlarında sırasıyla ikinci ve üçüncü sırada yer almaktadır. Diğer istatistiksel veriler, anonim ve tanımlanmamış olanların önemli ölçüde daha düşük olduğunu göstermektedir.

Literatür taramasının sonucunda;

- Müşteri kayıp analizinde açık verinin kısırlığı,
- Yapılan analizlerin ağırlıklı olarak mobil platformda olduğu,
- Gerçekleştirilen analizlerin büyük çoğunluğunun sınıflandırma problemine dayalı olduğu,
- Rastgele Orman, Destek Vektör Makineleri, Lojistik Regresyon, Karar Ağaçları ve Yapay Sinir Ağları gibi algoritmaların araştırmalarda en sık kullanılan algoritmalar olduğu,
- Son yıllarda oyunda müşteri kayıp analizinin bilimsel alanyazında popüler ve ilgi çekici bir konu olduğu sonucuna varılabilir.

2.11. MAKİNE ÖĞRENMESİ

Günümüzde yapay zeka programların insan gibi düşünme ve öğrenme becerisi olarak karşımıza çıkarken makine öğrenmesi yapay zekanın bir alt kategorisinde yer almaktadır. Belirgin programlama (explicitly programmed), bir değişikliği manuel olarak gerçekleştirmek için makineye özel talimatlar vererek programlama işlemidir. Makine öğrenmesi, belirgin (explicitly) programlama olmadan öğrenme yeteneğine sahip algoritmalar olarak tanımlanır (Koza ve diğ., 1996). Makine öğrenmesi bu noktada belirgin programlamadan, herhangi bir özel talimat gerektirmeden kendi kendine öğrenmeye çalışan sistemler olmasıyla ayrılır (Ng, 2019).

Makine öğrenmesi, insanların öğrenme faaliyetlerini taklit etmek için bilgisayarların nasıl kullanılacağını ve yeni bilgi ve yeni beceriler elde etmek, mevcut bilgiyi tanımlamak ve performansı ve başarıyı sürekli olarak iyileştirmek için bilgisayarların kendini geliştirme yöntemlerini inceleyen çalışma alanıdır (Wang ve diğ., 2009). İnsan öğrenmesi ile karşılaştırıldığında, makine öğrenmesi daha hızlı öğrenir ve öğrenme sonuçlarının daha kolay genelleştirilmesi sağlanır. Makine öğrenmesine dair alanyazında yer bulmuş ve sıklıkla kullanılan tanımlardan biri ise Mitchell (1997) tarafından yapılmıştır.

Bir bilgisayar programının, T 'deki görevlerde P ile ölçülen performansı E deneyimi ile geliyorsa, bazı T görev sınıfları ve P performans ölçütüne göre E deneyiminden öğrendiği söylenir.

Makine Öğrenimi, verilerden öğrenebilmeleri için bilgisayarları programlama bilimidir (Geron, 2017). Aynı zamanda, belirli bir probleme çözüm üretebilmek için bir olguya dair örneklerin bir araya getirilmesine dayanan algoritmalar oluşturmakla ilgilenen bilgisayar biliminin alt alanıdır (Joiner, 2018). Bu örnekler insanlar tarafından elle işlenebilir veya başka bir algoritma tarafından oluşturulabilir. Öte yandan, pratik bir probleme dair veri toplamak ve bu veri kümesine dayalı olarak algoritmik bir istatistiksel model oluşturmak yöntemlerle çözme süreci olarak da tanımlanabilir (Burkov, 2019). IBM (2022) tarafından ise insanların öğrenme şeklini taklit etmek ve doğruluğunu kademeli olarak arttırmak için veri ve algoritmaların kullanımına odaklanan yapay zeka ve bilgisayar biliminin bir alt çalışma dalı olarak tanımlanmıştır.

2.11.1. Makine Öğrenmesi Sistemlerinin Türleri

Makine öğrenmesi türlerinin açıklanabilmesi ve kavram kargaşasının önlenmesi için literatürde yer alan farklı tanımlamalara yer verilmesi gerekir. Veri biliminde ortak olarak kullanılan ve her araştırmacının problem çözümüne ulaşırken kullandığı bir terminoloji mevcuttur.

Verinin temelinin oluşturan aynı zamanda verideki her bir özelliği temsil eden yapılar öznitelik olarak adlandırılır. Ancak literatürde özellik ve bağımsız değişken olarak da kullanılmaktadır. Makine öğrenmesi denetimli, denetimsiz, yarı denetimli ve pekiştirmeli öğrenme gibi kavramları içerir. Bu kavramlar modelin veriden neyi nasıl öğreneceğine karar veren öğrenme modelleridir ve aynı zamanda makine öğrenmesi sistemlerinin türleri olarak adlandırılır. Makine öğrenmesi sistemleri, eğitim sırasında aldıkları gözetim miktarına ve türüne göre sınıflandırılır. Makine öğrenmesi türleri yabancı literatürde “supervised-unsupervised” şeklinde nitelendirilirken Türkçede karşılığı farklı kaynaklarda gözetimli/danışmanlı/denetimli ve gözetimsiz/danışmansız/denetimsiz gibi kavramlarla anılmaktadır. Bu tez kapsamında bu kavramlar öznitelik, gözetimli/gözetimsiz/yarı-gözetimli ve pekiştirmeli öğrenme şeklinde bahsedilecektir.

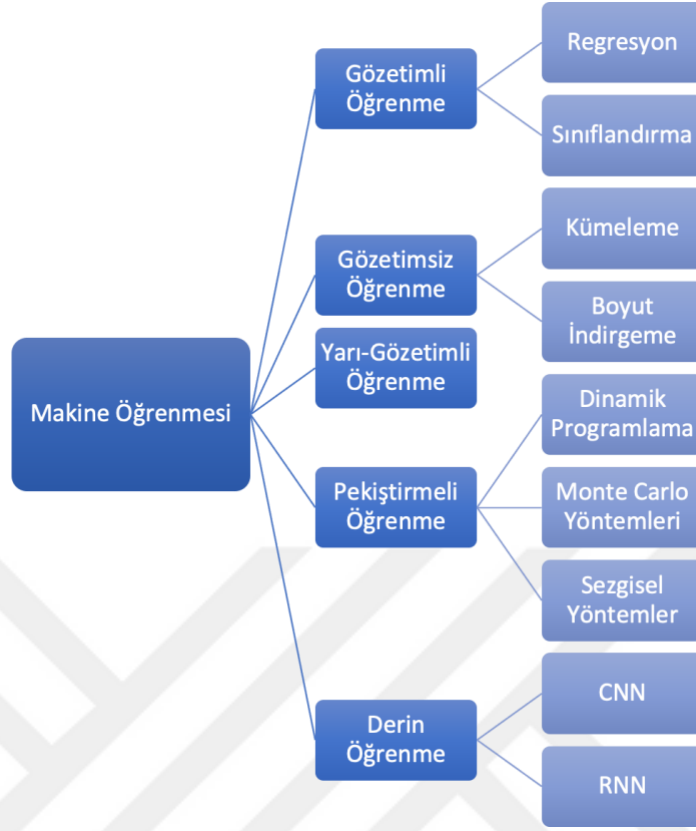
Veri madenciliği başlığı altında kategorize edilen makine öğrenmesi modelleri tanımlayıcı ve tahminleyici olmak üzere iki gruba ayrılır. Tahmine dayalı ve tanımlayıcı modeller, verileri analiz etmek ve tahminlerde bulunmak ya da verilerdeki örüntüleri anlamak için kullanılan iki tür istatistiksel modellerdir (Oracle, 2023). Sınıflandırma ve regresyon problemlerine çözüm sunan gözetimli öğrenme modelleri tahminleyici grupta yer alırken, kümeleme, yoğunluk analizi ve boyut indirgeme problemlerine çözüm sunan gözetimsiz öğrenme yöntemleri ise tanımlayıcı grupta yer almaktadır. Veri madenciliğinin sınıflandırılması Şekil 2.24’te şema olarak verilmiştir.



Şekil 2.24: Veri madenciliğinin sınıflandırılması

Tanımlayıcı modeller, verilerdeki kalıpları anlamak ve tanımlamak için kullanılır. Bu modeller gelecekteki olaylar hakkında tahminlerde bulunmaz. Bunun yerine verilerdeki ilişkilerin ve kalıpların ayrıntılı bir şekilde anlaşılmasını sağlar (Williams, 2011).

Tahmine dayalı modeller, geçmiş verilere dayanarak gelecekteki sonuçlar hakkında tahminlerde bulunmak için kullanılır. Hem tahmine dayalı hem de tanımlayıcı modellerin kendine özgü uygulamaları olmakla birlikte işletme, finans, ekonomi ve sağlık hizmetleri dahil olmak üzere birçok alanda kullanılırlar. Bu modeller, verilerdeki örüntüleri tanımlamak için istatistiksel teknikler uygular ve bu örüntüleri gelecekteki olaylar hakkında tahminlerde bulunmak için kullanır. Örneğin, bir müşterinin satın alma geçmişine dayanarak satın alma yapma olasılığını tahmin etmek için tahmine dayalı bir model kullanılabilir (Williams, 2011).



Şekil 2.25: Makine öğrenmesi ve uygulamaları

Makine öğrenmesi ise bu kapsamda gözetimli, gözetimsiz, yarı-gözetimli, pekiştirmeli ve derin öğrenme olarak beş alt başlıkta sınıflandırılır. Bunların her biri kendine özgü algoritma ve yöntemleri içermektedir. Bu alt başlıklar ve yöntemler Şekil 2.25’te verilmiştir.

2.11.1.1. Denetimli Öğrenme

Denetimli öğrenme, algoritmalarda mevcut veriden yeni bir tahmin yaparken önceden belirlenen etiketleri öğrenmesine dayalı gerçekleşen öğrenme yöntemidir (Alpaydın, 2020). Örneğin; farklı cihazlardan elde edilen verileri üç farklı çıktı grubuna ait bir problemi çözmede kullanırken, girdi olarak adlandırılan kısımda verilerin etiketleriyle birlikte makine sunulmasıdır. Bu süreçte makine verileri tahmin ederken mevcut etiketli veriden öğrenme yapar ve bu da tahmin performansını etkiler. Denetimli öğrenmeyi denetimsiz öğrenmeden ayıran temel farklardan biri eğitilme şekilleri ve sisteme verilen girdi türleridir. Bir diğer önemli fark ise, denetimli öğrenme algoritmalarının doğru çıktıyı sağlamak için insan müdahalesine ihtiyaç duyarken, denetimsiz öğrenme algoritmalarının ihtiyaç duymamasıdır. Bu, denetimsiz öğrenme algoritmalarının daha özerk olduğu ve insan rehberliği olmadan çalışabileceği anlamına gelir.

Gözetimli öğrenme algoritmaları, etiketli eğitim verilerinden öğrenerek öngörülemeyen veriler için sonuçları tahmin etmeye yardımcı olur. Sınıflandırma ve regresyon makine öğreniminde yaygın olarak kullanılan temel kavramlardır ve bu problemlere sunulan çözümler gözetimli öğrenme grubuna aittir. Bu kısma kadar makine öğrenmesinin türleri ve alt başlıklarına yer verilmiş ve tartışılmıştır. Bölümün devamında ise sınıflandırma ve regresyonun kavramsal çerçevesinden bahsedilecek ve kullanılan algoritmaların teorik içerikleri açıklanacaktır.

2.11.1.2. Denetimsiz Öğrenme

Denetimsiz öğrenme, makinelerin kendi kendilerine öğrenmelerini sağlar. Bu tür makine öğrenimi, yapay zeka uygulamalarına uygulayıcı gözetimi olmadan sınıflandırılmamış veya etiketlenmemiş veri noktaları içeren büyük veri kümelerindeki gizli kalıpları öğrenme ve bulmayı hedefler. Veriler denetimli öğrenmede olduğu gibi etiketlere (label) sahip değildir (Chollet, 2021, Alpaydın, 2020, Géron, 2019). Kümeleme, yoğunluk tahmini ve boyut azaltma gibi yöntemler bu grup altında değerlendirilir. Tablo 2.4'te gözetimli ve gözetimsiz öğrenmenin farklarına yer verilmiştir.

Tablo 2.4: Denetimli ve denetimsiz öğrenmenin farkları

Denetimli Öğrenme	Denetimsiz Öğrenme
• Etiketli veriler kullanır	• Etiketli veriler kullanılmaz
• Sınıflandırma ve regresyon problemlerini çözmek için kullanılır	• Kümeleme, yoğunluk tahmini ve boyut indirgeme problemlerini çözmek için kullanılır
• Eğitim veri seti kullanır.	• Tüm veriyi girdi olarak kullanır. Test ve Eğitim olarak ayrılma yapılmaz.
• Doğru çıktıyı sağlamak için insan müdahalesi gerektirir	• Doğru çıktıyı sağlamak için insan müdahalesi gerektirmez

Kümeleme, bir küme içindeki noktaların diğer kümelerdeki noktalara göre birbirine daha çok benzediği bir dizi veri noktasını kümeler halinde gruplama işlemidir. Kümeleme algoritmaları verilerdeki örüntüleri ve ilişkileri bulmaya ve benzer noktaları bir araya getirmeye çalışır.

Kümeleme genellikle benzer satın alma alışkanlıklarına sahip müşteri grupları veya benzer içeriğe sahip belge grupları gibi verilerdeki gizli yapıları keşfetmek için kullanılır.

Yoğunluk tahmini, rastgele bir değişkenin olasılık yoğunluk fonksiyonunu tahmin etme sürecidir. Olasılık yoğunluk fonksiyonu, verilerde farklı değerlerin ortaya çıkma olasılığını tanımlar. Yoğunluk tahmini, aykırı değerlere işaret edebilecek kümeleri veya olağandışı değerleri belirlemek gibi verilerdeki kalıpları ve eğilimleri tanımlamak için kullanılabilir.

Boyut azaltma, bir veri kümesindeki özelliklerin veya boyutların sayısını azaltma işlemidir. Bu genellikle verileri daha yönetilebilir hale getirmek veya gereksiz özellikleri kaldırmak için yapılır. Boyut azaltma, verilerin karmaşıklığını azaltarak makine öğrenimi algoritmalarının performansını arttırmak için kullanılabilir. Ayrıca yüksek boyutlu verileri daha düşük boyutlu bir uzayda görselleştirmek için de kullanılabilir (Han ve diğ., 2012).

Özetle, kümeleme veri noktalarını benzerliğe dayalı olarak kümeler halinde gruplamak için, yoğunluk tahmini rastgele bir değişkenin olasılık yoğunluk fonksiyonunu tahmin etmek için ve boyut azaltma bir veri kümesindeki özelliklerin veya boyutların sayısını azaltmak için kullanılır. Tüm bu teknikler, etiketli eğitim verilerine ihtiyaç duymadan verilerdeki örüntüleri ve ilişkileri bulmak için denetimsiz öğrenmede kullanılır.

2.11.1.3. Yarı Denetimli Öğrenme

Yarı denetimli öğrenme, eğitim için etiketsiz verilerin kullanılmasını da sağlayan bir denetimli öğrenme görevleri ve teknikleri sınıfıdır. Genellikle, etiketsiz veri sayısı etiketli veri sayısından daha fazladır. Yarı denetimli öğrenme, denetimsiz öğrenme (etiketli eğitim verisi olmadan) ile denetimli öğrenme (tamamen etiketli eğitim verisi ile) arasındadır. Birçok makine öğrenimi araştırmacısı, az miktarda etiketli veri ile etiketsiz verilerin öğrenme doğruluğunu önemli ölçüde artırabileceğini bulmuştur. Başka bir deyişle, büyük miktarda etiketsiz veri ve az miktarda etiketli veri kullanılarak oluşturulan bir öğrenme tekniğidir (Nabiyev, 2012).

Örneğin elimizde büyük miktarda veri olduğunu ve öğrenme yöntemi olarak denetimsiz öğrenmeyi seçtiğimizi varsayalım ancak küçük bir kısımla veriyi eğiterek daha iyi sonuçlar elde etmeye yönelik ayrılan küçük kısmı da denetimli öğrenme ile gerçekleştirme durumudur. Buna öğrenme sürecine denetimli öğrenme denmektedir.

2.11.1.4. *Pekiştirmeli Öğrenme*

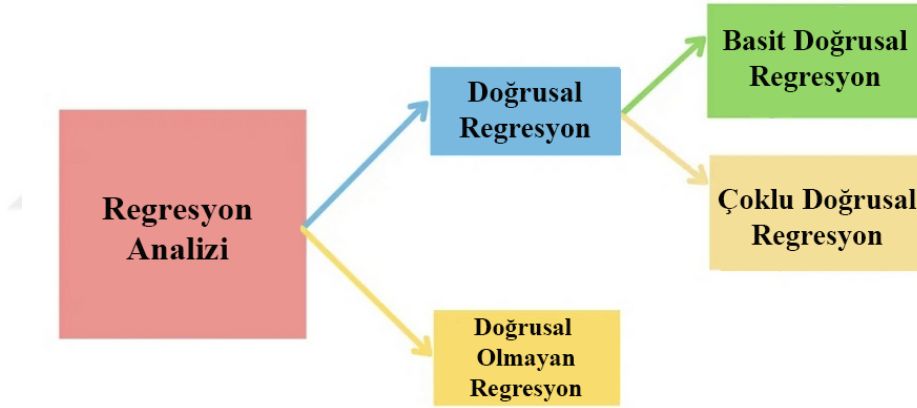
Gerektiğinde oyuncu veya ajan (agent)'in müdahale ettiği bir öğrenme tekniğidir. Bu öğrenme modeli tamamen öğrenme sürecini oyunlaştırmakla ilgilidir. Takviyeli öğrenme aslında deneme-yanılma, ödül-ceza mantığına dayanır. Ajan çevre ile etkileşime girer ve finalde karar verir veya seçim yapar. Makine doğru hamleyi yaparsa ödüllendirilir, hata yaparsa cezalandırılır (Ding ve diğ., 2020). Buradaki amaç toplam ödülü maksimize etmektir. Yani, bu algoritmayı yazan kişi, makineyi teşvik etmek için istenen eylemlere pozitif değerler ve istenmeyen eylemlere negatif değerler atar. Böylece makine optimum çözüme ulaşmak için uzun vadeli bir şekilde ödülü maksimize edecek şekilde programlanır. Bu şekilde makine kendi hatalarından ders çıkararak doğru olanı yapmayı öğrenir. Günümüzde dijital oyunlarda düşman hamleleri, masa oyunlarında rakip oyuncu, web kullanıcı arayüzleri, otonom araçlar, kişiselleştirilmiş ürün tavsiye sistemi, sohbet botları ve robotik gibi alanlarda sıklıkla kullanılmaktadır.

2.12. SINIFLANDIRMA VE REGRESYON

Sınıflandırma, en bilinen makine öğrenimi yöntemlerinden biridir. Görüntü tespiti, örüntü tanıma, hastalık tanısı koyma, dolandırıcılık tespiti, kalite kontrol ve pazarlamada sıklıkla kullanılır. İkili sınıflandırma ve çoklu sınıflandırma problemleri olarak ayrılır. İkili sınıflandırma bir probleme dair oluşturulan modelin iki çıktı grubunu ifade ederken, çoklu sınıflandırma ise bu çıktı gruplarının ikiden fazla olması durumudur (Alpaydın, 2020). Bir müşterinin aboneliğini iptal edip etmeyeceğine dair gerçekleştirilecek bir problemde evet ya da hayır olduğu için ikili sınıflandırma, bir çiçeğin belirli özelliklerinin yer aldığı veri setinde üç farklı çiçeğin hangi gruba ait olduğunu tahmin etmeye yönelik problem ise çoklu sınıflandırmaya örnek verilebilir.

Regresyon, bir bağımlı değişken ile bir veya daha fazla bağımsız değişken arasındaki ilişkiyi modellemek için kullanılan istatistiksel bir yöntemdir. Regresyonun amacı, bağımsız değişkenlerin değerlerine dayanarak bağımlı değişken hakkında tahminler yapmak için kullanılabilecek en iyi uyum çizgisini bulmaktır (Chatterjee, 2012; James ve diğ., 2013). Sürekli nümerik veri ve kategorik veri ile çalışabilir. Ancak, kategorik verilerle çalışırken, genellikle kategorik değişkenleri sayısal değişkenlere dönüştürmek gerekmektedir.

Regresyon, doğal ya da toplumsal olguların belirlendiği, çıkarsama yapılmak istenen öznitelikler arasında ilişkilerin ortaya konulmasında kullanılan ve matematiksel ifadelerle ortaya konulan süreçtir (Köklü ve diğ., 2021). Bu olgular bazen deterministik olurken bazıları ise olasılıksal olarak nitelendirilir. Örneğin, bir probleme çözüm üretirken formül kullanılır ve bu formül belirli şartlarda uygulandığında bizi gerçek çözüme ulaştırır. Aynı zamanda bu formüller bahsedilen şartlarda uygulandığında doğrulanabilir. Fen Bilimleri alanında sıklıkla kullanılan uygulamaya yönelik bu formüller gerçek yaşam problemlerini çözme konusunda yardımcı olurlar. Bu doğrultuda, sosyal olguların deterministik yöntemlerle açıklanması olanaksız olarak görülür. Bu tarz konuların çözümünde tahmin yoluna başvurulur. Bu doğrultuda regresyon, bağımlı bir değişken ile bir dizi bağımsız değişken arasındaki ilişkinin gücünü belirlemeye çalışan ve hem tanımlayıcı hem de çıkarımsal istatistik sağlayan bir ölçüdür (Sınıksaran, 2011).



Şekil 2.26: Regresyon sınıflandırması

Regresyon modelleri sınıflandırılırken doğrusal ve doğrusal olmayan modeller olarak iki gruba ayrılır. Doğrusal regresyon modellerinde basit doğrusal regresyon ve çoklu doğrusal regresyon olarak iki alt başlık yer alır. Çalışmanın ilerleyen bölümlerinde bu kavramlar detaylı şekilde açıklanmıştır.

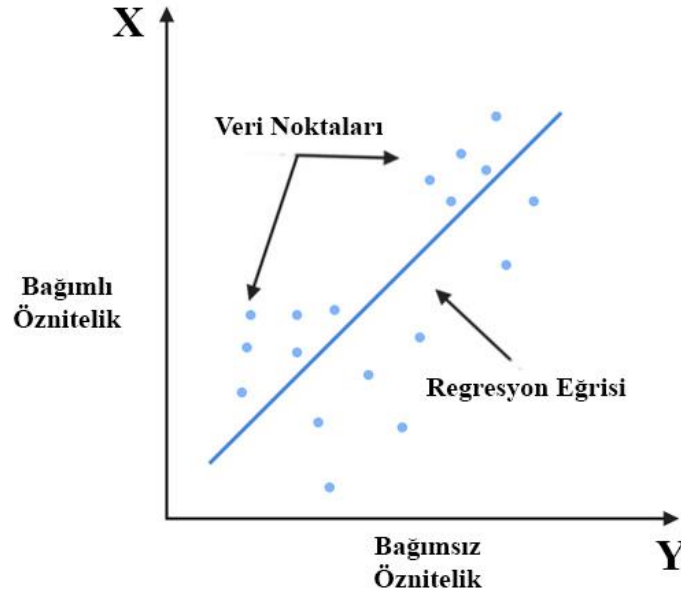
Regresyon analizinde temel amaç özniteliklerden birinin diğerine olan bağımlılık düzeyini matematiksel olarak ortaya koyabilmektir. y değişkeni x değişkeni tarafından deterministik bir ilişki dahilinde açıklanabiliyorsa, matematiksel olarak y , x 'in fonksiyonu ya da y , x üzerinde bir regresyon olarak nitelendirilebilir (Sınıksaran, 2011; Rawlings ve diğ., 2016). Burada y bağımlı öznitelik iken x bağımsız öznitelik veya açıklayıcı öznitelik olarak adlandırılır. Ayrıca

bu süreçte, üzerinde çalışılan özniteliklere ilişkin gözlem değerlerinin oluşturduğu veri göz önünde bulundurularak ortaya konulan ilişkinin matematiksel biçimi yani Regresyon eğrisi elde edilir.

$$Y_i = f(X_i, \beta) + e_i \quad (2.14)$$

Y_i = bağımlı öznitelik, f = fonksiyon, X_i = bağımsız öznitelik, β = bilinmeyen parametreler, e_i = hata terimini göstermektedir.

Bu analiz, bağımlı değişkenin değerini en iyi tahmin eden bir veya daha fazla bağımsız değişkeni içeren doğrusal denklemin katsayılarını tahmin etme üzerine kuruludur. Tahmin edilen ve gerçek çıktı değerleri arasındaki tutarsızlıkları en aza indiren düz bir çizgiye veya yüzeye uydurma işlemi gerçekleştirilir. Şekil 2.27’de doğrusal regresyon analizinde oluşturulan çizgi ve tahmin değerleri şema halinde verilmiştir.



Şekil 2.27: Doğrusal regresyon modeli

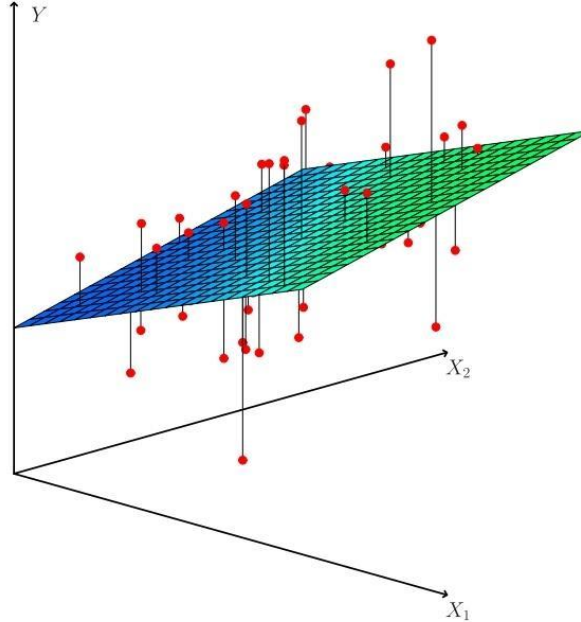
Doğrusal regresyon, veriler için en uygun doğruyu keşfetmek için en küçük kareler (Least Square) yöntemini kullanır. Daha sonra y'den, x'in değerini tahmin etme işlemi gerçekleşir (Newbold ve diğ., 2020). Sonuç olarak regresyon modeli, analiz edilen veya incelenen yanıt değişkeni olan bağımlı özneliğin değerini tahmin eder. Aynı zamanda bağımlı öznelik y'deki

değişmeyi bağımsız öznelilik x ile $f_x(x)$ fonksiyonu ile açıklanmasıdır. Dolayısıyla doğrusal regresyon, değişkenler arasında matematiksel bir ilişkiyi simüle eden ve satış, maaş, yaş, ürün fiyatı gibi sürekli veya nümerik öznelilikler için tahminler yapan bir denetimli öğrenme algoritmasıdır.

Regresyon doğrusu, veri noktalarından regresyon doğrusuna olan dikey mesafeyi mümkün olduğunca küçük yapan doğrudur. Çünkü en uygun çizginin varyansını (hataların karelerinin toplamı) en aza indiren çizgidir. Bunu görselleştirmek biraz zor olabilir ancak burada temel hedef bu noktalara mümkün olduğunca yakın uyan denklemi bulmaktır.

$$\frac{\sum (x - \bar{x}) * (y - \bar{y})}{\sum (x - \bar{x})^2} \quad (2.15)$$

Yukarıda belirtilen kısımda tek bir bağımsız özneliliğe ait değerlerin yer aldığını ve bunu bir katsayıyla çarparak yeni üretilen tahmin değerinin oluştuğu görülmektedir. Bu tek öznelilik baz alınarak yapılan doğrusal regresyondur.



Şekil 2.28: Çoklu Regresyon canlandırması

Sınıflandırma ve regresyon probleminde sıklıkla kullanılan algoritmalar k-en yakın komşu (k-nearest neighbor), karar ağaçları (decision tree), rastgele orman (random forest), aşırı gradyan artırma (xgboosting), gradyan artırma (gradient boosting), lojistik regresyon, sade Bayes (Naive

Bayes), destek vektör makineleri (support vector machines) olarak sıralanabilir. Bu algoritmalarla ilgili teorik bilgilere ve formüller detaylı şekilde verilmiş ve her algoritma için öncelikle sınıflandırma ardından regresyon modellerinin açıklaması yapılmıştır.

Birden fazla bağımsız öznitelikğin yer aldığı ve bir tahmin gerçekleştirmeye yönelik gerçekleştirilen regresyon sürecine ise çoklu regresyon denir. Çoklu regresyonda, doğrusal tek bağımsız öznitelikle gerçekleştirilen regresyon modelinde uygulanan varsayımlar çoklu regresyon içinde geçerlidir (Özdil ve diğ., 2015).

y = bağımlı öznitelik ve x_n = bağımsız öznitelikler ise,

$$y = b_0 + b_1x_1 + b_2x_2 + \dots + b_nx_n \quad (2.16)$$

Çoklu regresyon, çoklu açıklayıcı değişkenlere sahip doğrusal ve doğrusal olmayan regresyonları kapsayan daha geniş bir regresyon sınıfıdır. Doğrusal regresyonda ilişkinin eğimini etkileyen yalnızca bir bağımsız öznitelik varken, çoklu regresyonda birden fazla bağımsız öznitelik bulunmaktadır. Farklılıklarına rağmen hem basit hem de çoklu doğrusal regresyon modelleri doğrusal modellerdir, doğrusal bir denklem biçimini benimserler. Buna lineer varsayım denir. Oldukça basit bir şekilde, bağımsız öznitelikler kümesi ile bağımsız öznitelikler arasındaki ilişki türünün doğrusal olduğunu varsaydığımız anlamına gelir.

Doğrusal olmayan regresyon, iki veya daha fazla öznitelikten elde edilen çıktıyı grafiksel olarak çizmeye çalışmasından dolayı doğrusal regresyona oldukça benzer. Doğrusal olmayan modellerin geliştirilmesi doğrusal modellerden daha karmaşıktır çünkü fonksiyon, deneme-yanılma olmadan kaynaklanabilecek tekrarlar (iterasyon) yoluyla oluşturulur.

Modelin amacı, doğrusal regresyonda olduğu gibi hata kareleri toplamını mümkün olduğunca küçük yapmaktır. Kareler toplamı, Y gözlem değerlerinin y 'yi tahmin etmek için kullanılan doğrusal olmayan (eğri) fonksiyondan ne kadar farklı olduğunu izleyen bir ölçüdür. Bu süreçte veri setindeki her y veri noktası arasındaki fark bulunarak hesaplanır. Ardından, bu farkların her birinin karesi alınır. Son olarak, tüm kareli rakamlar toplanır. Bu kareli rakamların toplamı ne kadar küçükse, fonksiyon kümedeki veri noktalarına o kadar iyi uyar.

Doğrusal olmayan regresyon logaritmik fonksiyonları, trigonometrik fonksiyonları, üstel fonksiyonları, güç fonksiyonlarını, Lorenz eğrilerini, Gauss fonksiyonlarını ve diğer yöntemleri

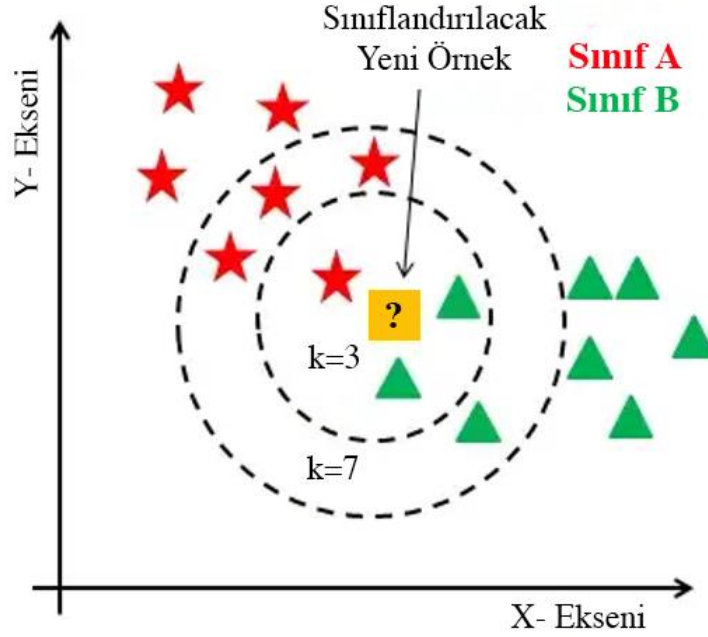
kullanabilir. Doğrusal regresyon iki değişkeni düz bir çizgi ile ilişkilendirir; doğrusal olmayan regresyon ise değişkenleri bir eğri kullanarak ilişkilendirir. Doğrusal olmayan regresyonda kullanılan bağımsız ve bağımlı öznitelikler nicel türde olmalıdır. Kategorik özniteliklerin nümerik özniteliklere dönüştürülmesi gerekir.

2.12.1. K-En Yakın Komşu Algoritması (K Nearest Neighbor)

Bazı algoritmalar, eğitim örneklerini ezberler ve sınıflandırma işlemi gerçekleştirmeden önce sisteme test verisi sunulana kadar bir işlem yapmazlar. Aynı zamanda, modelin eğitiminde girdilerin çıktılarına dönüştürülmesi sürecinde öğrenme açıkça (explicitly) gerçekleşmez. Belirgin (explicitly) kavramından yukarıda bahsedilmişti. Bu yaklaşım tembel öğrenme (lazy learning) veya örnek tabanlı öğrenme olarak adlandırılır. Aynı zamanda bu tip öğrenmeler, eğitim setinin çok büyük olduğu ve yeni örneklerle dair tahmin problemine hızlı ve efektif çözümler üretilmesi ele alındığı durumlarda faydalı olabilir. K-en yakın komşu, temel naif Bayes kuralları (lazy naive bayes rules) ve durum tabanlı çıkarma (case-based reasoning) bu sınıfta değerlendirilen algoritmalarından bazılarıdır.

KNN algoritmaları 1951 yılında ilk olarak Evelyn Fix and Joseph Hodges tarafından geliştirilen (Fix ve Hodges, 1951) daha sonra 1967 yılında Thomas Cover ve Peter Hard tarafından istatistiksel olarak genişletilmiştir (Cover ve Hart, 1967) .

Tembel öğrenme algoritmalarının en bilineni k-en yakın komşu, sınıflandırma ve regresyon problemlerinde sıklıkla kullanılan denetimli bir makine öğrenimi algoritmasıdır (Fix,1951; Altman,1992). Tahmin, girdiler arasındaki mesafeyi kullanarak gerçekleştirilir. Bu doğrultuda, algoritma komşuların nasıl sınıflandırıldığına bağlı olarak veri noktalarını sınıflandırma süreci izler (Hastie ve diğ., 2001; Pirayonesi ve El-Diraby, 2020). Bazı veri kümeleri için hızlı ve iyi bir performansa göstermektedir. Sınıflandırma ve regresyon problemine çözüm üretirken yüksek bir hesaplama maliyetine ve zayıf genelleme performansa sahip olabilir. K-en yakın komşu algoritmasının temel çalışma yapısı Şekil 2.29'da verilmiştir.

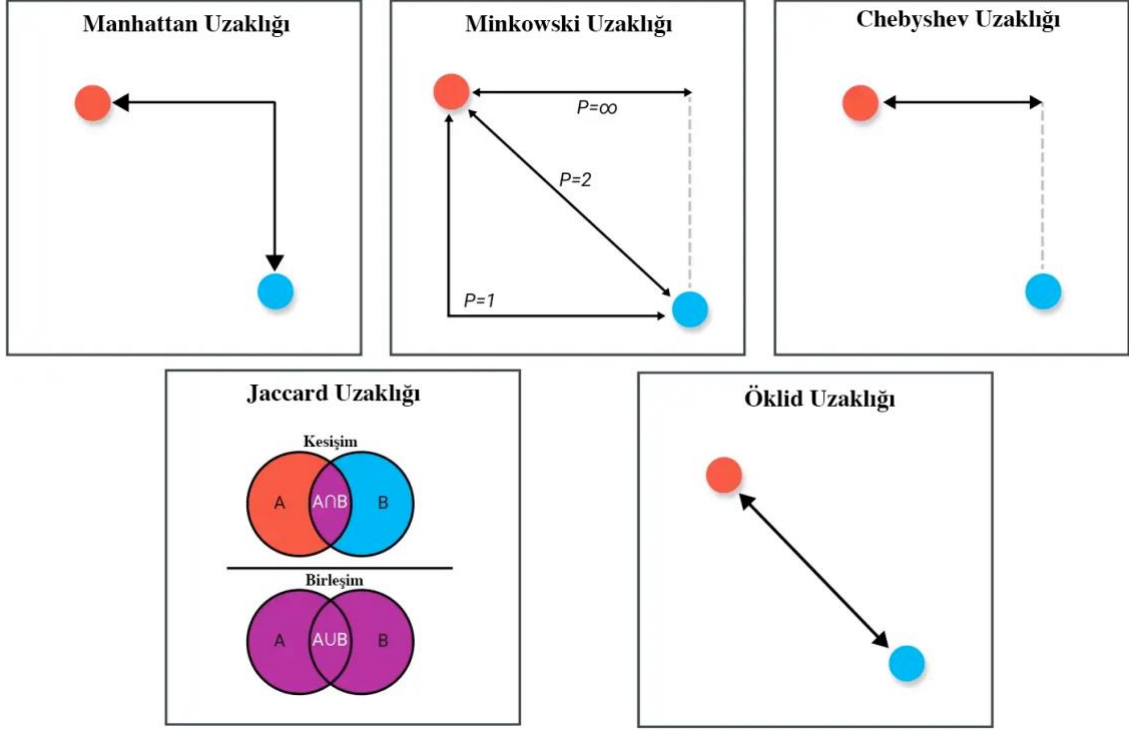


Şekil 2.29: KNN algoritma yapısı

K-en yakın komşu algoritması, belirli sınıflara sahip bir örneklem kümesindeki verilerden yararlanılarak kullanılır. Örnek veri setine eklenecek yeni verinin mevcut verilere göre uzaklığı hesaplanır ve k-yakın komşuları kontrol edilir. Örneğin; mevcut verilerin 3 sütun ve 7 satırdan oluştuğu varsayılırsa, bu kayıtlardan her sütundaki 7 değer birbirlerine olan uzaklıkları yeni giriş ile hesaplanır. Hesaplama işlemi yapıldıktan sonra çoğunluk oylaması ile sonucun hangi sınıfa ait olduğu tahmin edilir. Bu noktada çoğunluk oylamasında dikkate alınacak k sayısı temel parametre olarak önemlidir. k hiperparametresi genellikle tek sayı olarak seçilir ve bunun temel nedeni olarak çoğunluk oylamasında eşit bir durumun ortaya çıkmasını engellemektir.

Sınıflandırma problemlerinde;

Girdiler arasındaki mesafeyi hesaplarken öncelikle bir formül belirlenmelidir. Öklid, Manhattan, Chebyshev, Minkowski ve Jaccard en popüler uzaklıklardan bazılarıdır. Bu uzaklıkların teorik yapısı Şekil 2.30'da şemalandırılmış ve formüllere Tablo 2.7'de yer verilmiştir.



Şekil 2.30: Uzaklık ölçümlerinin şeması (Grootendorst, 2021’den uyarlanmıştır.)

Tablo 2.5: K-En Yakın Komşu Algoritmasında En Sık Kullanılan Uzaklık Formülleri

Uzaklık	Formül
Öklid	$D(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$
Manhattan	$D(x, y) = \sum_{i=1}^k x_i - y_i $
Minkowski	$D(x, y) = \sum_{i=1}^k (x_i - y_i ^p)^{\frac{1}{p}}$
Chebyshev	$D(x, y) = \max_i (x_i - y_i)$
Jaccard	$D(x, y) = 1 - \frac{ x \cap y }{ y \cup x }$

K-en yakın komşu algoritmasında kullanılan uzaklık ölçüleri aşağıda açıklamalarıyla birlikte verilmiştir (Chomboon ve diğ., 2015) . Bunlar;

Öklid uzaklığı, k-komşu algoritmasında en yaygın kullanılan ve çoğu araştırmacının hakkında bilgi sahibi olduğu uzaklıktır. Öklid uzaklığı iki nokta arasındaki doğrusal mesafedir.

Manhattan uzaklığı ise sadece iki nokta arasındaki farkın mutlak değeri alınır. Manhattan uzaklığı, Öklid uzaklığına göre aykırı değerlerden daha az etkilenir ve verinin boyutuna daha az duyarlıdır, bu da onu bazı uygulamalarda daha işlevsel haline getirebilir.

Minkowski uzaklığı, Öklid uzayında bir metrik olup, Öklid ve Manhattan uzaklıklarının genelleştirilmesi ile oluşturulmuştur. Gözlem değerleri arasındaki uzaklık belirli sayıda değişken dikkate alınarak hesaplanıyorsa Minkowski uzaklık bağıntısı kullanılabilir.

Chebyshev uzaklığı, herhangi bir koordinat boyutu boyunca iki vektör arasındaki farkın en büyüğü olarak tanımlanır. Başka bir deyişle, bir eksen boyunca maksimum mesafedir. Satrançta şahın bir kareden diğerine gitmek için ihtiyaç duyduğu minimum hamle sayısı Chebyshev mesafesine eşit olduğundan, genellikle Satranç Tahtası mesafesi olarak da adlandırılır.

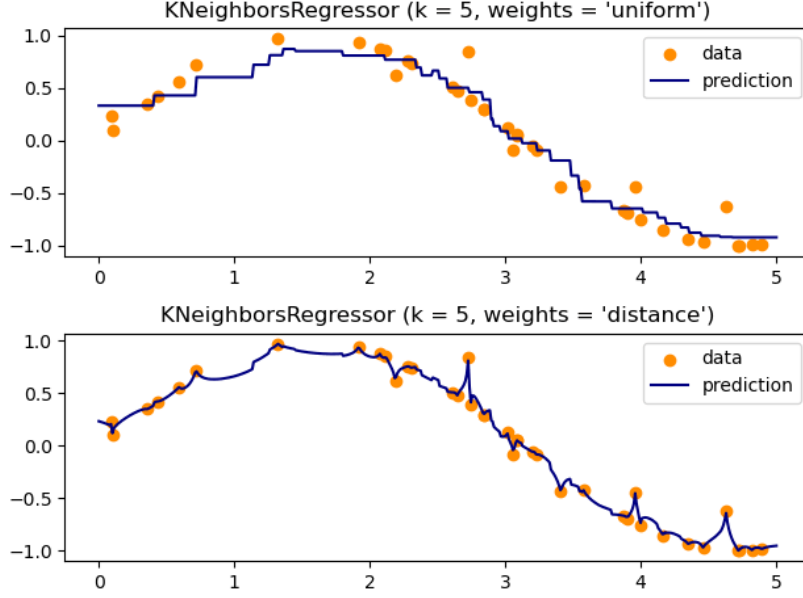
Jaccard uzaklığı, örnek kümelerin benzerliğini ve çeşitliliğini hesaplamak için kullanılan bir metriktir. Kümeler arasındaki benzer varlıkların toplam sayısının toplam varlık sayısına bölünmesiyle elde edilir.

Regresyon problemlerinde,

Daha önce de belirtildiği gibi KNN, o noktaya en yakın K komşunun sonucuna göre tahminler yapar ve bu tahmini yapabilmesi için örneklem noktaları arasındaki mesafenin kullanılması gerekir. Bu metrikler Tablo 2.30'de verilmişti. KNN regresyonu, aynı komşuluktaki gözlemlerin ortalamasını alarak bağımsız öznitelikler ile sürekli güncelleyerek sonuca yakınlaşan bir yöntemdir.

Yöntem ilk başta kullanışlı gibi görünse de verinin çok sayıda bağımsız özniteliğe sahip olması durumunda kullanışsız hale gelebilir. Avantajı, doğrusal regresyonda olduğu gibi öznitelikler arasında doğrusal bir ilişki gerektirmemesidir. KNN regresyonunun uygulanması yavaştır çünkü tüm örnekler arasındaki mesafenin hesaplanmasına dayanır; bu da yüzlerce özellik ve

binlerce gözlem içeren büyük veri kümeleri için çok zaman alıcı olabilir. Şekil 2.31’de k-en yakın komşu algoritmasının eğriyi modele uydurmasına dair örneklere yer verilmiştir.



Şekil 2.31: KNN regresyonunda eğri uydurması (Pedregosa, 2019’den alınmıştır.)

x_i, y_i eğitim örnekleri olarak varsayıldığında, x_i örneklemin öznitelik değerleri, y_i ise hedeflenen çıktı özneliği olarak tanımlanır. KNN algoritması $D(x, x_i)$ uzayında k değeri seçilir ve belirlenen uzaklık ölçüsüne göre hesaplanarak en yakın komşuların ortalaması alınarak yeni bir değer tahmini gerçekleştirir (Zhang ve diğ., 2017).

$$\hat{y} = f(x) = \frac{1}{k} \sum_{j=1}^k y_{i_j} \quad (2.17)$$

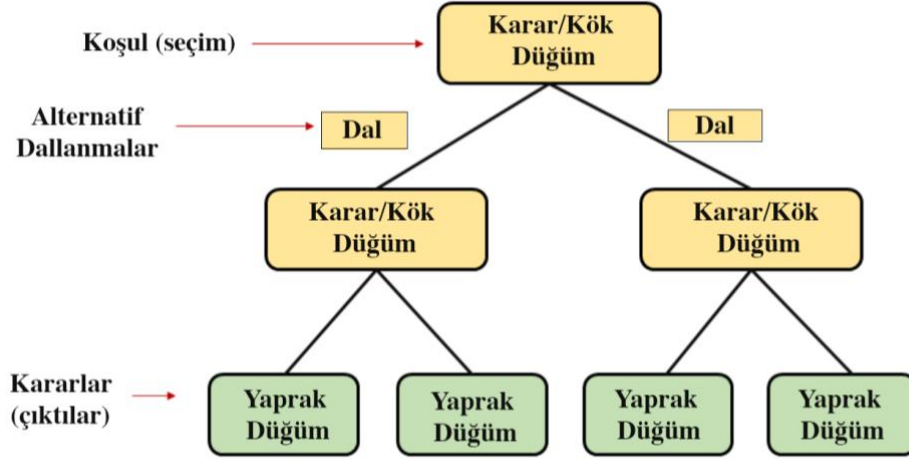
2.12.2. Karar Ağaçları Algoritması

Karar ağaçları sınıflandırma ve regresyon dahil olmak üzere çeşitli tahmine dayalı modelleme uygulamalarında sıklıkla kullanılmaktadır. Karar ağaçları algoritması temelde her bir düğümün (node) bir nitelik üzerindeki bir koşulu, her bir dalın (branch) deneme sonucunu temsil ettiği ve her bir yaprak düğümünün (leaf node) bir sınıf etiketi oluşturduğu ağaç yapısına dayanan bir algoritmadır. Diğer bir ifadeyle, büyük miktardaki kayıtlara basit karar vermeye yönelik koşullar uygulanarak, küçük kayıt gruplarına bölünmesi ile kullanılan bir algoritmadır. Bu süreç "if-else-then" durumunun çok koşullu olarak kullanılması süreci olarak düşünülebilir.

1980 yılında tarafından Gordon Kass CHAID algoritması ve 1986 yılında Ross Quinlan tarafından ID3 algoritması önerilmiştir. Bu önerilerin ardından, 1993 yılında yine Quinlan tarafından C4.5 algoritması sınıflandırma problemine çözüm için literatürde yerini almıştır (Loh, 2014; Akpınar, 2018). Karar ağaçları 1984 yılında Leo Breiman, Jerome Friedman, Charles J. Stone ve R.A. Olshen tarafından önerilmiştir (Moore, 1987). Sınıflandırma ve regresyon ağaçları anlamına gelen CART (Classification and Regression Trees) karar ağaçları algoritmasının bir diğer adıdır.

Karar ağaçları, kolay uygulanabilir yapısı performansı ile öne çıkmaktadır. Aynı zamanda sinir ağları (neural network), destek vektör makineleri (support vector machines) ve derin öğrenme (deep learning) gibi algoritmaların aksine hızlı çalışmasıyla popülerdir (Rokach ve Maimon, 2005; Kingsford ve Salzberg, 2008). Öte yandan, çok sayıda kayıt içeren veri kümesini bir karar kuralı (decision rule) uygulayarak daha küçük kümelerle bölmek için kullanılır. Ayrıca, veri madenciliği sürecinde çıkarım sağlanması beklenen tercihleri, riskleri, dönütleri, hedeflerin analizinde yardımcı olan başta işletme ve yönetim olmak üzere birçok önemli çalışma alanında kullanılmaktadır.

Karar ağaçlarının diğer algoritmalarla göre daha az zahmetli olması, alt ağaç (sub-tree) yorumlamanın daha kolay olması, anlaşılabilir kurallar oluşturma becerisine sahip olması ve sürekli ve ayırık değerlerle çalışabilmesi avantajları arasında yer alırken, sürekli verinin tahmininde çok iyi olmaması, öznitelik sayısı yüksek ve örnek kümesi eleman sayısı az olduğunda çok başarılı olmaması, her problem için özel ağaç yapıları olduğundan, standart bir ağaç yapısı tanımlamanın zorluğu ve çok boyutlu verilerde bazen kuralları yanlış kurgulama sonucu başarısız performans sergilemesi dezavantajları arasında yer alır (Swain ve Hauska, 1977).



Şekil 2.32: Karar ağacı çalışma şeması

Şekil 2.32’de görüldüğü üzere, algoritma otomatik olarak bir kök düğüm (root node) oluşturur. Daha sonra bu kök düğüm dallanmalara ayrılıp karar düğümlerini (decision node) oluştururlar. Son olarak karar düğümlerinin her birinin altında yaprak düğümleri (leaf node) yer alır. Bu şemaya göre karar ağaçları bir kurala göre yapı oluşturur ve tahmin gerçekleştirir (Altunkaynak, 2019).

Peki, kök düğüm ve diğer düğümlerin oluşturulması neye göre belirlenir? Bu sorunun cevabı Gini katsayısı ve Entropi değeri ile açıklanmaktadır.

Düğüm oluşturmayı belirleyen bir diğer yöntem ise Entropi (maksimum karmaşıklık) formülüdür. Veri madenciliği haricinde fizik alanında bir sistemin mekanik olarak iş çevrilemeyen termal enerjisini temsil eden termodinamik bir terim olarak kullanılır (Akpınar, 2018; Altunkaynak, 2019). Veri madenciliğinde ise, karmaşıklığı ve beklenmedik olayların gerçekleşme olasılığını temsil eder. Maksimum entropi veya düzensizlik seviyesi 1 ile, minimum entropi ise 0 değeri ile temsil edilir. Tüm örneklerin 1 sınıfa ait olduğu yaprak düğümlerin entropisi 0 olurken, sınıfların eşit olarak bölündüğü bir düğümün entropisi 1 olur.

$$Entropi = - \sum_{i=1}^n p_i \log_2 (p_i) \quad (2.18)$$

Karar Ağacı algoritmasında kök düğümü belirlemek için her özniteliğin ve potansiyel bölünmelerin entropisi hesaplanır. Bunun için her öznitelikten bölünme yüzdesi, her iki

düğümdeki veya tüm düğümlerdeki ortalama entropiyi bularak ana düğüme göre entropideki değişimin hesaplanması gerekir. Entropideki bu değişim Bilgi Kazancı (Information Gain) olarak kabul edilir ve bir özelliğin hedef öznitelik için ne kadar bilgi sağladığını temsil etmektedir. Bilgi Kazancı formülü Denklem 2.19’da verilmiştir.

$$\text{Bilgi Kazancı} = \text{entropi}(\text{kök düğümler}) - [\text{ortalama entropi}(\text{alt düğümler})] \quad (2.19)$$

2.12.2.1. ID3 ve C4.5 Karar Ağacı Algoritmaları

ID3 (Iterative Dichotomiser 3) ve C4.5, makine öğrenmesinde sınıflandırma problemlerinde kullanılan popüler karar ağacı algoritmalarıdır. Her iki algoritma da tahminler yapmak için ağaç tabanlı bir yapı kullanmasının yanı sıra her bir düğüm bir girdi niteliği üzerindeki bir testi, her dal bu testin bir sonucunu ve her bir yaprak düğümü bir sınıf etiketini temsil etmektedir.

Kazanç (X_i): i . bağımsız öznitelik için kazanç değeri, $H(Y)$: Sınıfın entropi değeri, k_i : i . öz niteliğin kategori sayısı, $P(X_{ij})$: i . öz niteliğin j . düzeyinin ortaya çıkma olasılığı, $H(Y/X_{ij})$: i . öz niteliğin j . düzeyi için sınıfa ait entropi değeri, m öznitelik olduğu ve $i=1,2,3,\dots,m$ olduğu varsayıldığında,

$$\text{Kazanç}(X_i) = H(Y) - \sum_{j=1}^{k_i} P(X_{ij}) H\left(\frac{Y}{X_{ij}}\right) \quad (2.20)$$

ID3 algoritmasında kazanç değeri büyük olan öznitelik, sınıf için en fazla belirleyiciliğe sahip öznitelik olarak esas alınır ve buna göre dallanma gerçekleştirilir. Ayrıca, bir düğüme (node) sınıf değeri tek bir çıktı ile sınıflanıyorsa, o düğüme saf (pure) düğüm adı verilir (Çelik ve diğ., 2017).

C4.5, ID3 algoritmasının bazı sınırlamalarının üstesinden gelen bir uzantısıdır. C4.5 hem sürekli hem de kategorik verileri işleyebilir ve ayrıca sınıflandırma performansına fazla katkıda bulunmayan dalları kaldırmak için ağacı budayabilir. C4.5 ayrıca bölünecek en iyi öz niteliği seçmek için kazanç oranı kavramını kullanır ve bu da bölünmeden kaynaklanan dalların sayısını dikkate alır.

Ağaç yapısının karmaşıklık düzeyi arttığında ve eğitim verilerinde iyi performans gösterdiğinde, ya da görünmeyen verilerde kötü performans gösterdiği aşırı uyumdan problemi

ortaya çıkabilir. Bu sorunu çözmek için, ağacın boyutunu küçültmek ve genelleme performansını iyileştirmek amacıyla çeşitli budama teknikleri geliştirilmiştir.

Budama (pruning), ID3 ve C4.5 gibi karar ağacı algoritmalarında ağacın boyutunu küçültmek ve genelleme performansını arttırmak için kullanılan bir işlemdir. Buradaki fikir, ağacın sınıflandırma doğruluğuna fazla katkıda bulunmayan dallarını çıkarmaktır. Budama, ağacın çok karmaşık hale geldiği ve eğitim verilerine iyi uyum sağladığı ancak görünmeyen verilerde kötü performans gösterdiği durumlarda, aşırı uyumu önlemeye yardımcı olur. Doğruluğa fazla katkıda bulunmayan dalların kaldırılmasıyla ağaç daha basit hale gelir ve yeni verilere daha iyi genelleme yapabilir (Pedregosa, 2018).

ID3 algoritmasına ait formülde yer alan şartlar kabul edildiğinde $H(X_i) : X_i$ özniteliliğine ait entropi değerini temsil ettiği varsayıldığında Denklem 2.21 uygulanarak hesaplanır.

$$\text{Kazanç Oranı}(X_i) = \text{Kazanç} \frac{H(X_i)}{H(X_i)} \quad (2.21)$$

Bu kısımda kazanç oranı en yüksek olan öznitelik sınıflar üzerinde en fazla belirleyiciliğe sahip olan öznitelikler olarak seçilir ve dallandırma buna göre gerçekleşir.

Budama işlemi ön budama (pre-pruning) ve son budama (post-pruning) olarak iki sınıfa ayrılır. Ön budama özniteliklerin arasında değersiz olanların dallanma sürecine dahil edilmemesine dayanırken, son budama tüm ağaç oluştuktan sonra belirli yöntemlerle bazı dalların kaldırılmasına dayanır.

2.12.2.2. Sınıflandırma ve Regresyon Ağaçları (CART)

ID3, C4.5 ve C5.0'dan sonra geliştirilen CART algoritması sınıflandırma ve regresyon problemleri için kullanılabilen popüler bir karar ağacı algoritmasıdır. Algoritma, öncüllerinden olduğu gibi veride yer alan değerlere dayalı tahminler yapmak için ağaç modeli oluşturur. Sınıflandırmada amaç, her yaprak düğümündeki Gini ya da entropideki gibi safsızlığı (impurity) en aza indirgeyerek ikili ağaç oluşturmaktır. Regresyon problemlerinde ise, yaprak düğümlerden gelen değerlerin o yaprakta yer alan örnekler için ortalamasını alarak hedefi tahmin ettiği aynı zamanda Hata Kareler Ortalamasını (MSE) en aza indirecek şekilde bir model oluşturmaktır (Müller ve Guido, 2016).

Bu ağaçlar oluşturulurken hem sınıflandırma hem de regresyon problemlerinde bazı kriterlere göre dallanma gerçekleştirilir. Bunlar Twoing ve Gini kriterleri olarak karşımıza çıkar.

Twoing kriteri, büyük olan öznitelik sınıfının üzerinde en çok belirleyicilik oranına sahip öznitelik olarak alınır ve dallanma buna göre gerçekleştirilir.

D_i : i . Bölünmeyi, $\phi(D_i)$: i . bölünme için twoing kriteri, P_L : i . bölünme için gözlemin solunda kalma olasılığı, P_R : i . bölünme için gözlemin sağında kalma olasılığı, $P(Y_j|R)$: Ağacın sağında yer alan sınıfın j . düzeyinin olasılığını ifade ettiği varsayılırsa,

$$\phi(D_i) = 2P_L P_R \sum_{j=1}^k \left[\left(P\left(\frac{Y_j}{L}\right) - P\left(\frac{Y_j}{R}\right) \right)^2 \right] \quad (2.22)$$

Gini katsayısı, 1912 yılında İtalyan istatistikçi Corrado Gini tarafından geliştirilen istatistiksel bir ölçüdür. Rastgele seçilen bir ögenin ne sıklıkla yanlış tespit edildiğini ölçmek için kullanılır. Aynı zamanda, alt kümenin saflık değeridir. Katsayı 0 ile 1 arasında değişmekte olup 0 mükemmel eşitliği, 1 ise mükemmel eşitsizliği temsil eder. Sonuç 0'a ne kadar yakınsa veriler arasındaki ayrılma seviyesi o kadar iyidir (akt. Çami, 2017; Yin, 2019). Kök düğüm oluşturulurken Gini katsayısı düşük olan bir öznitelik tercih edilmelidir. Gini endeksi kategorik hedef değişken için ikili (binary) sınıflandırma problemine tahmine dayalı olarak çalışır. Her sınıf için Gini katsayısı hesaplanır ve sonuçların karelerinin toplamı çıkarılır.

Denklem 2.23'te Gini katsayısının hesaplanması formülize edilmiştir. Her iki formül için de p_i 'nin, i sınıfının olasılığını temsil ettiği varsayılırsa;

$$\text{Gini Katsayısı} = 1 - \sum_{i=1}^n p_i^2 \quad (2.23)$$

Hiperparametre ve parametre kavramlarına önceki bölümlerde bahsedilmişti. Ağaç tabanlı modellerde (karar ağaçları ve topluluk öğrenimi yöntemleri vb.) aşırı uyum ve yetersiz uyum gibi durumlar ile modelin performansının artırılmasının hedeflendiği durumlarda modelde bazı hiperparametreler düzenlenerek bu sorunlar çözülmeye çalışılır. Karar ağaçlarında kullanılan hiperparametreler açıklamalarıyla birlikte Tablo 2.8'de verilmiştir.

Tablo 2.6: CART algoritması hiper parametreler ve açıklamaları

Hiperparametre	Açıklama
Ayırıcı (Splitter)	Her düğümde bölünmeyi seçmek için kullanılan strateji
Kriter	Bölmenin işlevinin kalitesini ölçme yöntemi
Maksimum öznitelik (max_features)	En iyi bölünmeyi ararken göz önünde bulundurulması gereken özelliklerin sayısı
Maksimum Derinlik (max_depth)	Ağacın maksimum derinliği
Düğüm için minimum gözlem sayısı	Bir düğümü bölmek için gereken minimum örnek sayısı
Yaprak için minimum gözlem sayısı	Yaprak düğümde olması gereken minimum örnek sayısı

Karar ağacı regresyonu ise bir nesnenin özelliklerini gözlemler ve anlamlı sürekli çıktı üretmek üzere gelecekteki verileri tahmin etmek için ağaç yapısındaki bir modeli eğitir. Bir regresyon ağacının oluşturulması da verileri bölümlere ayıran yinelemeli bir süreç olan ikili yinelemeli bölümlendirmeye dayanır. Başlangıçta, tüm ağacın yapısını belirlemek için eğitim örnekleri kullanılır. Algoritma daha sonra verileri olası her ikili bölünmeyi kullanarak kırar ve verileri iki parçaya bölen bölünmeyi seçer. Böylece ayrı parçalardaki ortalamadan karesel sapmaların toplamını en aza indirir (Xu ve diğ., 2005).

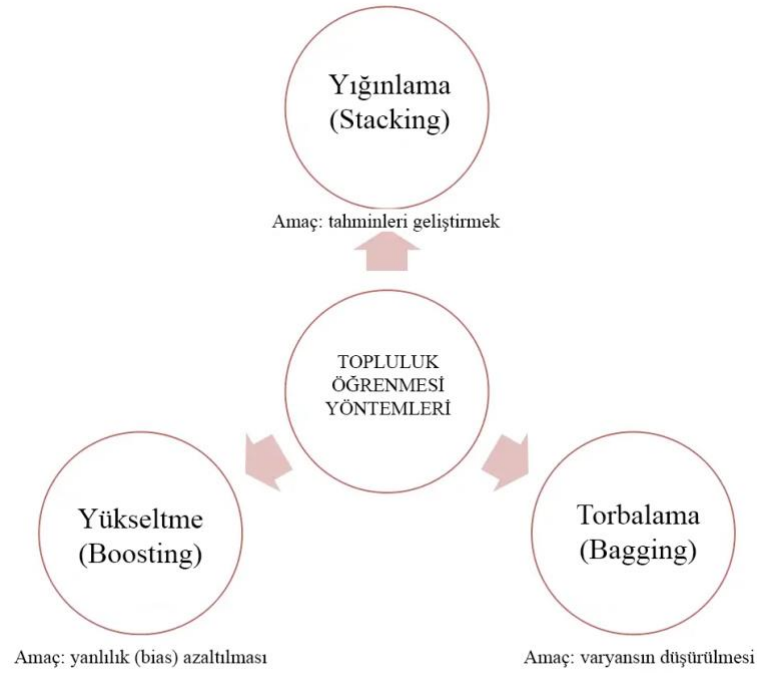
Bölme (split) işlemi daha sonra yeni dalların her birine uygulanır. İşlem, her bir düğüm kullanıcı tarafından belirlenen minimum düğüm boyutuna (yani düğümdeki eğitim örneklerinin sayısı) ulaşana ve bir terminal düğüm haline gelene kadar devam eder (Mohamed ve diğ., 2012).

2.12.3. Topluluk Öğrenmesi Yöntemleri (Ensemble Methods)

Çözülmesi hedeflenen bir sorun varsa ve buna çözüm üretilmesi bekleniyorsa, bazı durumlarda bireylerin verdikleri cevaplar, çoğunluğun verdiği cevaplardan daha yetersiz olabilir. Tek bir kişi tarafından verilen kararların hata yüzdesi ile topluluk tarafından verilen kararların hata yüzdesi arasında muhakkak fark olacaktır. Mevcut problemi çözmek için tek bir kişinin fikrini sormak yerine, bir topluluğun fikrini sorarak bir sonuç elde edilir ve bu topluluk öğrenmesi olarak adlandırılır (Opitz ve Maclin, 1999; Polikar, 2006). Bu yöntemler, topluluğun kararının

bir kişinin kararına olan üstünlüğü fikrine göre geliştirilmiştir. Topluluk yöntemleri genellikle çok sayıda bağımsız tahmin modeli oluşturmayı ve ardından bunları yüksek performans elde etmek amacıyla oylama veya ortalama alma yoluyla birleştirmeyi ifade etmektedir (Rokach, 2010).

Topluluk yöntemleri en iyi performansı, tahmin edicilerin (predictors) birbirinden mümkün olduğunca bağımsız olduğu durumlarda gösterir. Başarılı tahminler elde edebilmek için, farklı algoritmaları bir araya getirerek verinin eğitim sürecine dahil edilmesi sürecidir. Bu sayede, model çok farklı hata türleri yapma şansını topluluğun doğruluğunu da doğrusal oranda artırmaktadır (Geron, 2017). Şekil 2.33'te topluluk öğrenmesi yöntemlerinin farklı sınıfları temel amaçlarıyla birlikte yer verilmiştir.

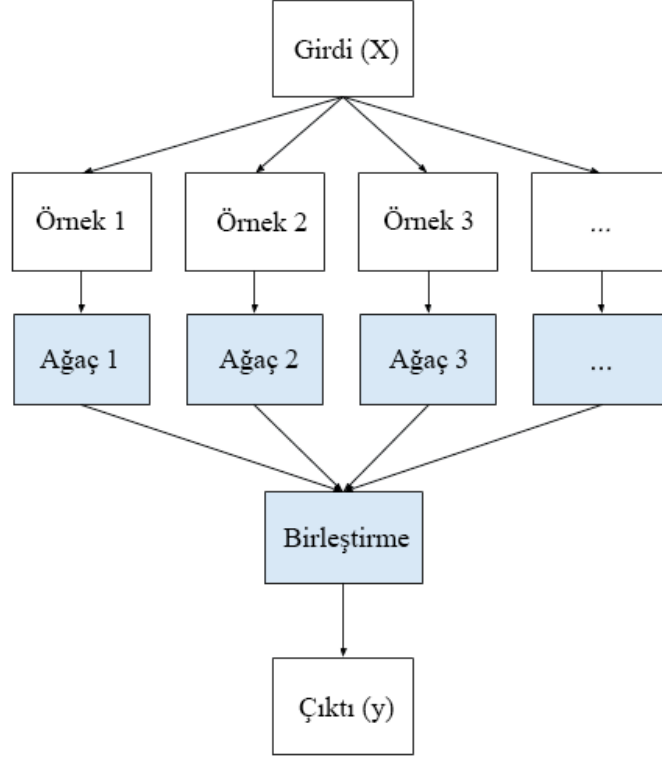


Şekil 2.33: Topluluk Öğrenmesi yöntemlerinin sınıflandırılması

2.12.3.1. Torbalama (Bagging)

Torbalama terimi Berkeley Üniversitesinden Profesör Leo Breiman tarafından ortaya atılmıştır. Breiman, istatistiksel öğrenme için karar ağaçlarının geliştirilmesinde etkili olmuş ve farklı rastgele veri alt kümeleri üzerinde çok sayıda ağacın eğitilmesi ve ortalamasının alınmasının varyansı azaltacağını ve kararlılığı artıracığını fark etmiştir.

Bu terim Bootstrap AGGREGATING ifadesinin kısaltılmasından gelmektedir ve bootstrap örnekleme ile ilişkilendirilmektedir (Breiman, 1994; Breiman 1999). Bagging'in iki temel bileşeni bootstrap ve aggregation algoritmasıdır.



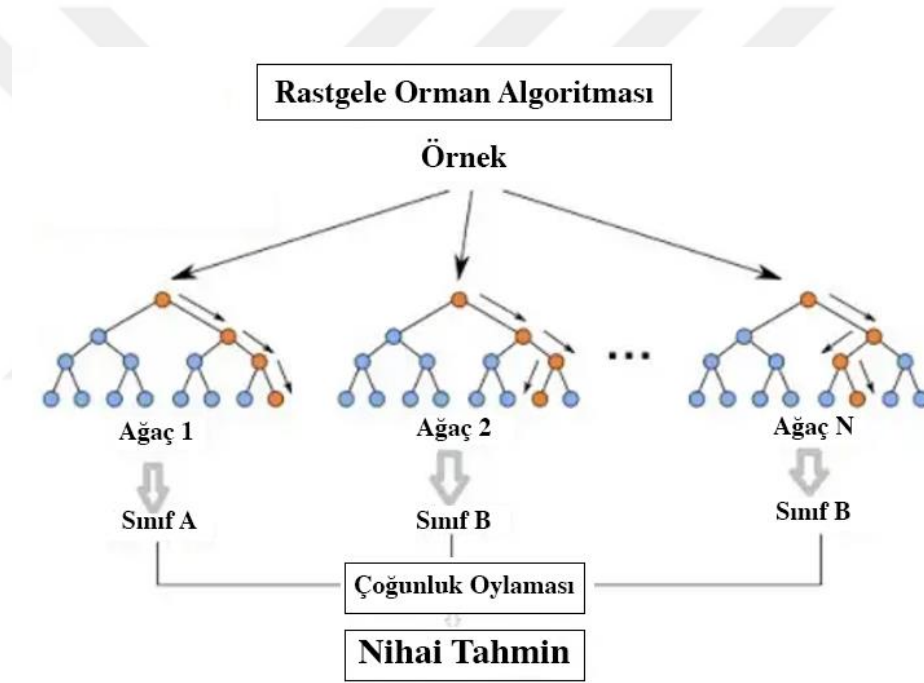
Şekil 2.34: Torbalama algoritmasında öğrenenlerin birleştirilmesi

Temel öğrenenleri eğitmek üzere veri alt kümelerini elde etmek için bootstrap örnekleme uygular. Örnekler veri kümesinden rassal biçimde ve iadeli olarak çekilir. Bunun yanında, karar ağacına benzer şekilde aynı temel modelin birden fazla örneğinin eğitim verilerinin farklı alt kümeleri üzerinde eğitilmesi ve ardından oylama veya ortalama alma yoluyla tahminlerinin birleştirilme işlemi gerçekleştirir. Örneğin, eğitim veri setinde daha fazla değişiklik yapılabilir, eğitim verilerine uyan algoritma değiştirilebilir ve tahminleri birleştirmek için kullanılan mekanizma değişikliği yapılabilir (Breiman, 1994).

2.12.3.2. Rastgele Orman (Random Forest)

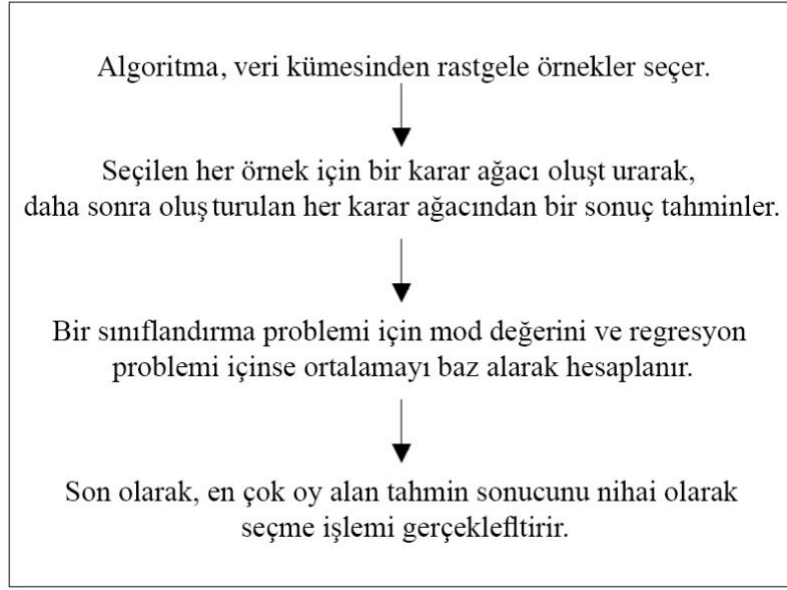
Şekil 2.35'te görüldüğü üzere torbalama farklı torbalardan (bag) gelen tahminlerin tek bir yerde toplanması yani birleştirilmesi işlemidir. Rastgele orman (random forest), ekstra ağaçlar (extra trees) gibi birçok popüler topluluk algoritması bu yaklaşıma dayanmaktadır (Gashler, 2008).

Rastgele orman, 2001 yılında Leo Breiman tarafından geliştirilen bir topluluk yöntemidir ve birden fazla karar ağacı tahminleyicilerini gruplandırır. Aynı zamanda, makine öğrenmesi problemlerinde sıklıkla kullanılan güçlü ve basit bir algoritmadır. Kaggle (açık veri paylaşım platformu) üzerinde gerçekleştirilen yarışmalarda sıklıkla kullanılan ve iyi performans gösteren ve hem sınıflandırma hem de regresyon problemlerini çözmek için kullanılan algoritmalarından biridir (Breiman, 2001). Rastgele orman farklı çalışma alanlarında sıklıkla kullanılmaktadır. Tıpta hastanın tıbbi kaydını analiz ederek hastalığını tanısını koymada, bankacılık sektöründe bir müşterinin sahte mi yoksa gerçek mi olduğunu kolayca tespit etmede ve son yıllarda iş analitiği (business analytics) alanında müşteri kaybı analizinde sıklıkla kullanıldığı bilinmektedir (Massaro ve diğ., 2019).



Şekil 2.35: Rastgele Orman Algoritması şematik yapısı

En iyi özneteliği bulmak yerine ekstra rastgelelik bulmaya çalışarak ağaçların dallanmasını gerçekleştirir. Rastgele ormanın nihai tahminleri, her bir ağacın tahminlerinin ortalaması alınarak yapılır. Bireysel karar ağaçları eğitim verilerine aşırı uyum sağlama eğilimindedir, ancak rastgele bir orman, farklı ağaçlardan gelen tahmin sonuçlarının ortalamasını alarak bu sorunu hafifletebilir. Bu, rastgele ormanlara tek bir karar ağacından daha yüksek bir tahmin doğruluğu sağlar. Rastgele orman algoritmasının işlem basamakları şema halinde Şekil 2.35'te verilmiştir.



Şekil 2.36: Rastgele orman algoritmasının işlem basamakları

Veri bilimi alanında çalışan araştırmacılar sıklıkla aşırı uyum (overfitting) sorunuyla karşılaşır. Bu da sonuçları olumsuz etkileyen kritik bir sorundur ancak rastgele orman algoritması için ormanda yeterli sayıda ağaç varsa bu kısmen önlenir. Ayrıca, rastgele orman algoritmasının eksik değerleri işleyebilmesi ve kategorik değerler için modellenir olması algoritmanın avantajları arasında yer almaktadır. Özellikle büyük veriler üzerinde hesaplama yaparken çok maliyetli olması, ağaç sayısının çok fazla arttığında aşırı uyuma yönelmesi, dengesiz veri setleri üzerinde daha az etkili çalışması ve kayıp değerlerle başa çıkma konusunda daha düşük performans göstermesi ise algoritmanın dezavantajları arasında yer almaktadır (Ziegler ve König, 2014).

B : ağaç sayısı, T_b : b .inci ağaç, C_b : b . ağacın sınıf değeri varsayıldığında $b = 1$ to B ve olmak üzere,

1. Eğitim verilerinden Z^* kadar veriyle N bootstrap örneği çekilir.
2. Minimum $n_{\min}$ düğüm boyutuna ulaşılan kadar ağacın her bir terminal düğümü için aşağıdaki adımları özyinelemeli olarak tekrarlayarak önyükleme verilerine bir rastgele orman ağacı T_b genişletilir.
 - i. p öznitelikler arasından rastgele m değişken seçilir.

- ii. m değişken arasından en iyi değişkeni diğer bir tabirle bölünme noktasını seçilir.
- iii. Düğümü iki alt düğüme ayrılır.

Çıktı değerler $\{T_b\}_1^B$ için model yeni tahmin oluştururken x :

Regresyon için,

$$\hat{f}_{\text{rf}}^B(x) = \frac{1}{B} \sum_{b=1}^B T_b(x) \quad (2.24)$$

Sınıflandırma için,

$\hat{C}_b(x)$ b. rastgele orman ağacının sınıf tahmini olarak varsayılırsa, bu durumda;

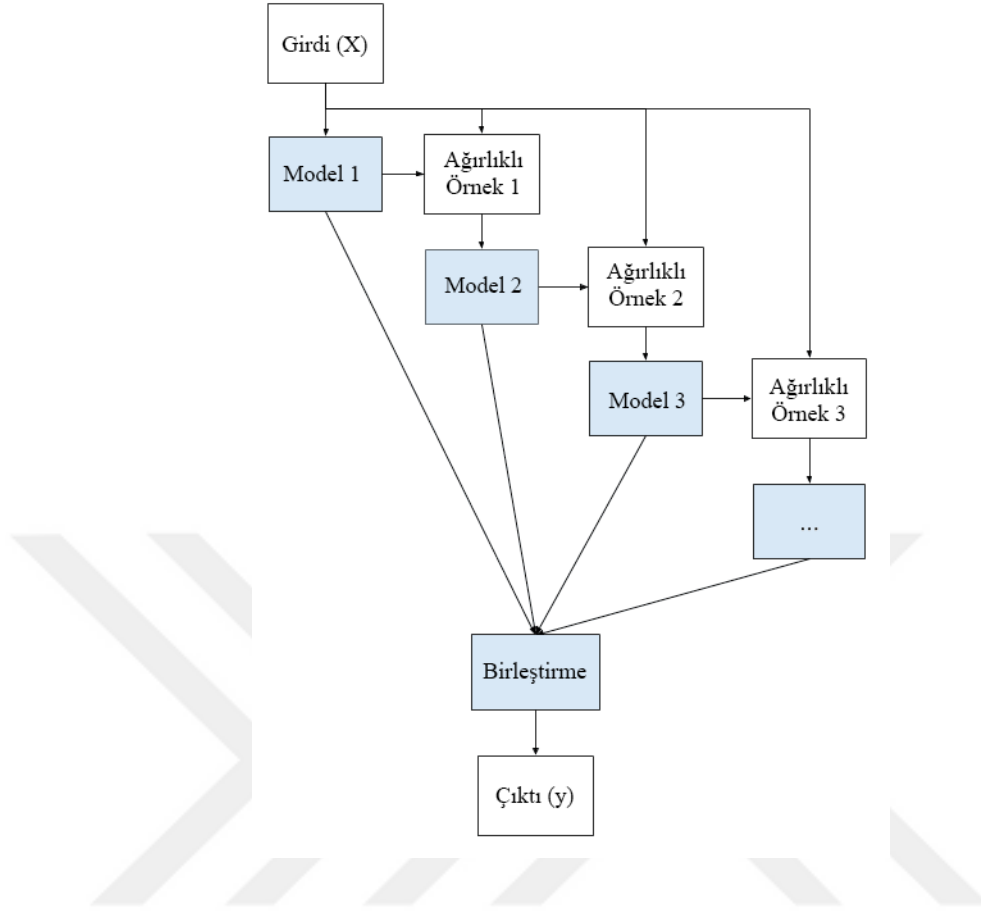
$$\hat{C}_{\text{rf}}^B(x) = \text{çoğunluk oylaması } \{\hat{C}_b(x)\}_1^B \quad (2.25)$$

2.12.4. Yükseltme (Boosting)

Yükseltme yöntemleri 1988 ve 1989 yıllarında Pennsylvania Üniversitesi'nden Michael Kearns ve Harvard Üniversitesi'nden Leslie Valiant tarafından ortaya atılan “bir dizi zayıf öğrenen (weak learner) tek bir güçlü öğrenen (strong learner) oluşturabilir mi?” sorusuna çözüm olarak geliştirilmiştir (Kearns, 1988; Kearns ve Valiant, 1994; akt. Akbaş ve diğ., 2013;). 1990 yılında ise Robert Schapire tarafından bu soruya çözüm önerisi getirilmiştir ve istatistiksel olarak literatürde anlam kazanmaya başlamıştır (Schapire, 1990).

Yükseltme (boosting) torbalama sürecindeki gibi verilerin eğitilmesini ancak her modelin bir önceki modelin zorlandığı örneklerle daha fazla ağırlık vermesine dayanır (Breiman, 2000; Zhou, 2012). Her sınıflandırıcı için eğitim veri kümesi giderek daha önce oluşturulan sınıflandırıcılar tarafından yanlış sınıflandırılan örneklerle odaklanır. Aynı zamanda, eğitim veri setindeki önceki uygun modellerin yanlış yaptığı örneklerle odaklanmak için eğitim verilerini değiştirmeyi amaçlayan bir topluluk yöntemidir (Liu ve diğ., 2012).

Yükseltme yöntemlerinin temel özelliği modelin tahmin hatalarını düzeltmesidir. Her model kendinden önceki modeli düzeltmeye çalışarak tahminler sırayla yerleştirilir ve topluluğa eklenir. Zayıf öğrenenlerin tahminleri basit oylama ya da ortalama kullanılarak birleştirilir. Yükseltme algoritmalarının yöntemi Şekil 2.37’de şema halinde verilmiştir.

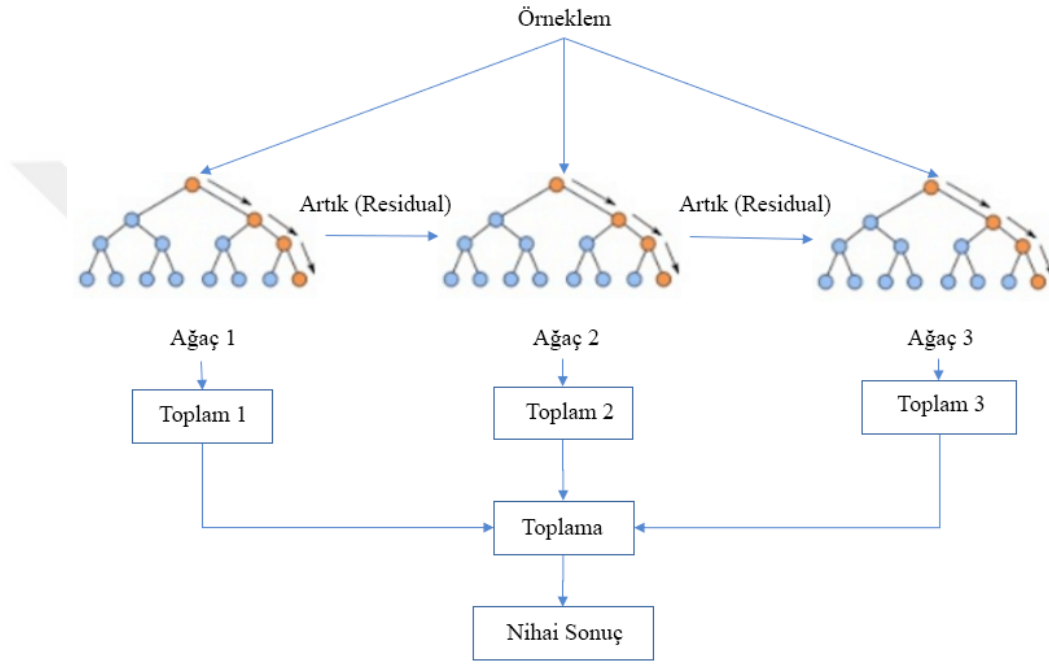


Şekil 2.37: Yükseltme yöntemi

Adaptif Yükseltme (AdaBoost) algoritması geliştirilinceye kadar yükseltme algoritmaları etkili bir topluluk yöntemi olarak görülmemiştir. Adaptif Yükseltme, Gradyan Güçlendirme Makineleri (Gradient Boosting Machines), Aşırı Gradyan Yükseltme (XGBoost) ve Kategori Yükseltme (CATBoost-Category Boosting) gibi birçok topluluk algoritması bu yaklaşıma dayanmaktadır. Aşırı Gradyan Yükseltme (XGBoost-Extreme Gradient Boosting) ilk olarak Tianqi Chen tarafından geliştirilmiştir ve Chen ve Carlos Guestrin 2016 yılında yayınlanan makalelerinde dile getirmişlerdir. XGboost, standart Gradyan artırma algoritmasının verimli ve etkili bir şekilde uygulanmasını sağlayan bir yöntemdir (Chen ve Guestrin, 2016).

XGBoost, geliştirilmesinden ve ilk sürümünden kısa bir süre sonra, makine öğrenimi yarışmalarında sınıflandırma ve regresyon problemleri için kazanan çözümlerin temel bileşeni haline gelen topluluk öğrenmesi yöntemlerinden biridir. XGBoost diğer gradyan yükseltme yöntemlerine göre daha hızlıdır (Pafka, 2015). Ayrıca, sınıflandırma ve regresyon tahmini modelleme problemlerinde yapılandırılmış (structured) veya tablosal (tabular) veri kümelerinde daha iyi performans göstermektedir.

Bilgi ve soru paylaşım platformlarından biri olan Quora’da paylaşılan bir gönderide Tianqi Chen’e GBM ile XGBoost arasındaki farkın ne olduğu bir soru sorulmuş ve yanıt olarak XGBoost’un işleme hızı ve model performansı olduğunu dile getirmiştir. 2015 yılında veri paylaşım platformu Kaggle üzerinde gerçekleştirilen bir veri yarışmasında 29 yaklaşım arasından kazanan 3 çözümün XGBoost algoritmasına ait olduğu ve yarışmada toplam 17 modelin XGBoost algoritmasını kullandığı belirtilmiştir (Quora, 2017).



Şekil 2.38: XGBoost algoritmasının çalışma prensibi

Şekil 2.38 incelendiğinde, sınıflandırma problemi için değerler tipik olarak olasılıkların ve olasılıkların logaritması kullanılarak hesaplanır. Ağacın çıktısı, başka bir ağaç oluşturmak için kullanılan veri kümesi için yeni artık (residual) oluşturur. Bu işlem artıklar azalmayı durdurana kadar veya belirli bir sayıda tekrarlanır. Sonraki her ağaç önceki ağaçlardan öğrenir. XGBoosting’i rastgele orman algoritmasından ayıran kritik noktası ağırlıkların eşit olarak atanmaması ve ağaçlardan gelen değerlerin toplanmasıdır.

$\{(x_i, y_i)\}_{i=1}^N$ eğitim kümesi, $L(y, F(x))$ kayıp fonksiyonu, M zayıf öğrenenlerin sayısı, α ise öğrenme oranı olarak varsayıldığında, Denklem 2.27 ile model sabit bir değerden başlatılır.

$$\hat{f}_{(0)}(x) = \arg \min_{\theta} \sum_{i=1}^N L(y_i, \theta) \quad (2.26)$$

$m = 1$ 'den M 'ye kadar olmak üzere, Denklem 2.28 ile 'gradyanlar' ve 'hessianlar' hesaplanır.

$$\begin{aligned}\hat{g}_m(x_i) &= \left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)} \right]_{f(x)=\hat{f}_{(m-1)}(x)} \\ \hat{h}_m(x_i) &= \left[\frac{\partial^2 L(y_i, f(x_i))}{\partial f(x_i)^2} \right]_{f(x)=\hat{f}_{(m-1)}(x)}\end{aligned}\quad (2.27)$$

$$\hat{\phi}_m = \arg \min_{\phi \in \Phi} \sum_{i=1}^N \frac{1}{2} \hat{h}_m(x_i) \left[-\frac{\hat{g}_m(x_i)}{\hat{h}_m(x_i)} - \phi(x_i) \right]^2 \quad (2.28)$$

$$\hat{f}_m(x) = \alpha \hat{\phi}_m(x) \left\{ x_i, -\frac{\hat{g}_m(x_i)}{\hat{h}_m(x_i)} \right\}_{i=1}^N \quad (2.29)$$

Denklem 2.29 ve 2.30 uygulanarak eğitim setini kullanarak bir temel öğrenen (veya ağaç gibi zayıf öğrenici) modele sunulur. Denklem 2.37 ile model güncellenir.

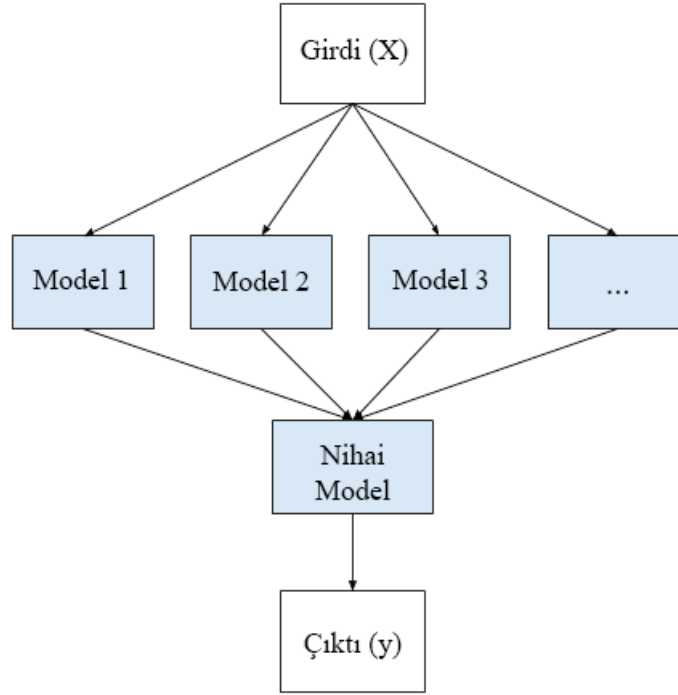
$$\hat{f}_{(m)}(x) = \hat{f}_{(m-1)}(x) + \hat{f}_m(x) \quad (2.30)$$

Denklem 2.31 ile çıktı değeri elde edilir.

$$\hat{f}(x) = \hat{f}_{(M)}(x) = \sum_{m=0}^M \hat{f}_m(x) \quad (2.31)$$

2.12.5. Yığınlama (Stacking)

Yığınlama (stacking) ise yine aynı eğitim sürecinde tahminlerin nihai tahmin sonucuna ulaşması için eğitilen bir meta model girdi olarak kullanılmasını içerir. Bireysel öğrenenleri (individual learner) bir araya getirmek için bu süreçte öğrenenlerin eğitildiği bir algoritmadır (Ozay ve Vural, 2012). Burada, bireysel öğrenenler birinci seviye öğrenen (first-level learner) olarak adlandırılırken, birleştirici (combiner) ise ikinci seviye öğrenen (second-level learner) veya meta öğrenen (meta learner) olarak adlandırılır.



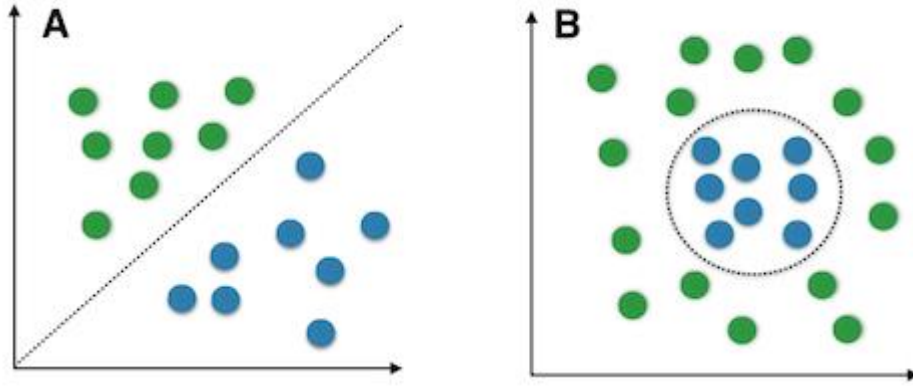
Şekil 2.39: Yığınlama yöntemi akış şeması

Yığınlama en popüler meta öğrenme tekniklerinden biridir. Bu yöntem, bir meta öğrenen kullanarak hangi sınıflandırıcıların güvenilir olduğunu ve hangilerinin olmadığını ortaya çıkarmaya çalışır. Ancak temel hedefi yukarıda belirtildiği üzere tahmin sonuçlarını geliştirmektir. Yığılmış Modeller (Stacked Models), Harmanlama (Blending) ve Süper Topluluk (Super Ensemble) gibi birçok topluluk algoritması bu yaklaşıma dayanmaktadır. Şekil 2.39’da yığınlama yönteminin akış diyagramına yer verilmiştir.

2.12.6. Destek Vektör Makineleri

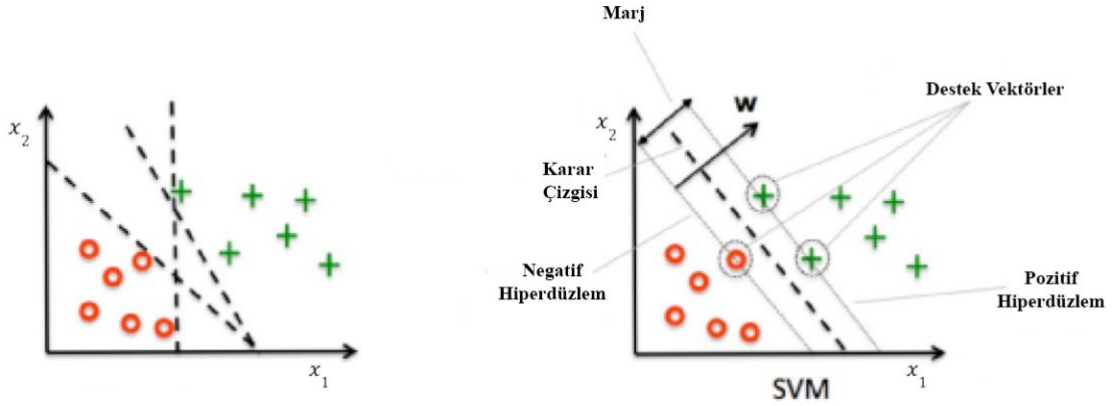
Temelleri Vladimir Vapnik ve Alexey Chervonenkis tarafından öğrenme teorisi kapsamında 1960’lı yıllarda ortaya koyulan DVM (Destek Vektör Makineleri veya Destek Vektör Ağları), 1992 yılında Vladimir Vapnik, Bernhard Boser ve Isabelle Guyon tarafından teorik olarak şekillendirilmiştir.

1990’lı yıllarda makine öğrenmesinin el yazısı ve metin tanıma, kanser teşhisi ve zaman serisi analizi birçok problemin çözümünde başarılı şekilde kullanılmıştır. DVM, sınıflandırma ve regresyon görevi için kullanılan popüler denetimli öğrenme algoritmalarından biridir (Cortes ve Vapnik, 1995).



Şekil 2.40: Doğrusal ve doğrusal olmayan problemlerin şeması

DVM algoritmasının geliştirilmesinde temel amaçlardan biri çok boyutlu ve lineer olmayan sınıflandırma problemi üzerine çözüm üretmektir. Bazı problemlerde doğrusal bir sınıflandırma çizgisi çekerek sınıfların ayrımını yapmak mümkündür. Ancak bazı durumlarda bu çizgi doğrusal olmamakla birlikte farklı sınıfları ayırabilmek için farklı şekillerde ayırma noktaları oluşturmak gerekebilir. Şekil 2.40'ta görüldüğü gibi A grafiğinde doğrusal çizgi sınıflandırma probleminde yeterli iken, B grafiğinde yeterli değildir. B grafiğindeki çizgi aslında sınıfları birbirinden ayırır ancak bunu doğrusal olmayan yöntemlerin yardımıyla yapar.



Şekil 2.41: DVM algoritmasının terimleri ve şemalandırılması

DVM algoritmasında iki farklı sınıfa ait verilerin ayrılabilmesi için algoritmanın temelinde yer alan marj çizgisi karşımıza çıkar. Bu noktaları ayırmak için iki sınıftaki değerlerin uç noktalarına bir çizgi çizilir. Bu çizgilerin düzgün çizilmesi ve açısının iyi ayarlanması önemlidir. Bu noktada, DVM bu çizgileri en uzak mesafeden çizmeye çalışır. Çünkü amaç,

çizgiye yakın değerlerin yanlış sınıfta yer almasını engellemek ve çizginin en yakın sınıf noktalarına ayrılmasını sağlamaktır.

Bu çizgiler arasında yer alan alana marj (margin) adı verilir w ile ifade edilir. Aynı zamanda DVM, w değerini maksimize etmeye çalışan bir algoritmadır. Bu yapının anlaşılması için karar çizgisi (decision boundary), negatif ve pozitif hiperdüzlem (negative-positive hyperplane) kavramlarının açıklanmasında fayda vardır. Hiperdüzlem, DVM'de iki sınıfı birbirinden ayıran bir karar sınırır. Hiperdüzlemin her iki tarafına düşen bir veri noktası farklı sınıflara atfedilebilir. Hiperdüzlemin boyutu, veri kümesindeki girdi özelliklerinin sayısına bağlıdır. Karar çizgisi ise bir sınıfa ait tüm örneklerin bu çizginin bir tarafında, diğer sınıfa ait tüm örneklerin ise çizginin karşı tarafında yer aldığı ve sınıfları birbirinden ayıran çizgidir. Şekil 2.41'de görüldüğü üzere, sınıflandırma probleminin çözümünde kırmızı noktalara yakın çekilen çizgi negatif hiperdüzlem d^- ile ifade edilirken, yeşil noktalara yakın çekilen çizgi ise pozitif hiperdüzlem d^+ ile ifade edilir.

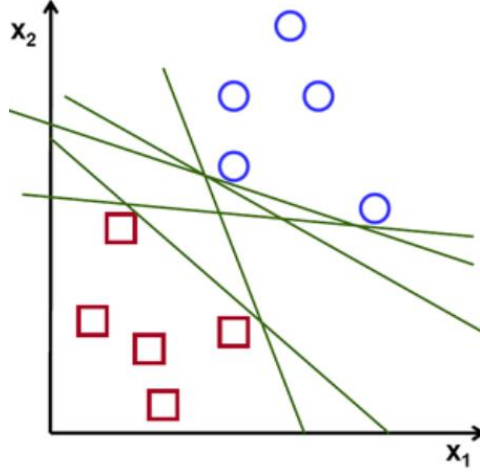
Tablo 2.7: Doğrusal ve doğrusal olmayan DVM'lerin karşılaştırılması

Doğrusal DVM	Doğrusal Olmayan DVM
Doğrusal bir düz çizgi ile kolayca ayrılabilir.	Doğrusal bir düz çizgi ile kolayca ayırmak zordur.
Veri, hiperdüzlemlerin yardımıyla sınıflandırılır.	Çekirdek (kernel) kullanarak ayrılamayan veriyi ayrılır hale getirebilir.
Veri, bir doğru çizilerek kolayca sınıflandırılabilir.	Veriyi daha büyük uzay boyutuna yükselterek sınıflandırma yapabilir.

Tablo 2.9'da doğrusal ve doğrusal olmayan DVM'leri özelliklerinin karşılaştırılmasına yer verilmiştir. Bu doğrultuda DVM sınıflandırıcıları doğrusal ve doğrusal olmayan olarak 2 gruba ayrılır.

2.12.6.1. Doğrusal Sınıflandırma

Sınıfların düz bir doğru ile ayrılabilirdi destek vektör sınıflandırma türüdür. Burada en uygun noktalardan düz üç tane çizgi çekilir ve bu çizgilerin biri pozitif sınıfı diğeri negatif sınıf temsil ederken, ortadan çekilen çizgi ise en uygun hiperdüzlem olarak karşımıza çıkar (Tolun, 2008; Uğuz, 2020).



Şekil 2.42: Lineer sınıflandırmada çizilebilecek çizgi örnekleri

Denklem 2.32 doğrusal regresyon modeli olarak bilinir. (Bkz. Bölüm 2.14)

$$y = h_{\theta}(x) = \theta_0 + \theta_1 x \quad (2.32)$$

Denklem 2.32’da $x = x_1$ ve $y = x_2$ yazılırsa bu durumda Denklem 2.33 ile 0 sonucuna ulaşılır.

$$x_2 = \theta_1 + \theta_1 x_1 \Rightarrow x_2 = \theta_1 x_1 - x_2 + \theta_0 = 0 \quad (2.33)$$

x ve w iki boyutlu vektörler olarak varsayıldığında,

$$x = (x_1, x_2) \text{ ve } w = (\theta_1, -1) \quad (2.34)$$

Denklem 2.34’de yer alan değerlerin vektörler iç çarpımlarından Denklem 2.35 ve tekrar yerine koyulursa Denklem 2.36’elde edilir.

$$\theta_1 x_1 - x_2 = \langle w, x \rangle \quad (2.35)$$

$$\langle w, x \rangle + \theta_0 = 0 \quad (2.36)$$

Burada ifade edilen yapı, doğrusal regresyon modelinin bir hiper düzleme dönüştürülmesi sürecidir. $y_i \in \{-1, 1\}$ olmak üzere, x_i vektörlerinin her biri y_i sınıfına ait olduğu ve bu sınıfların doğrusal olarak ayrılabilirdiği varsayıldığında,

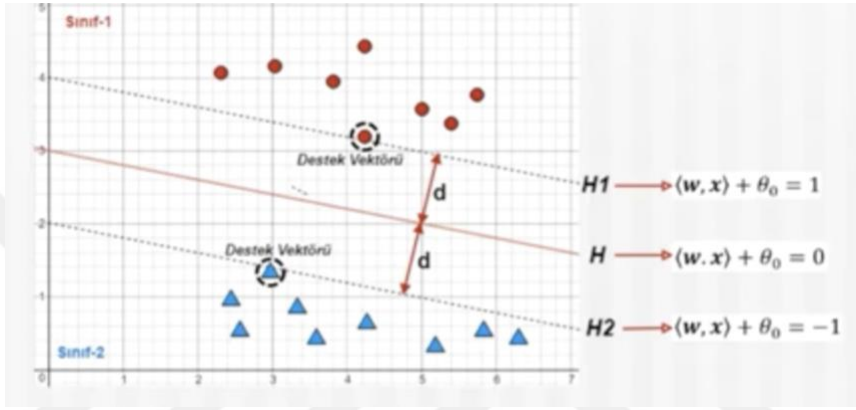
$$\langle w, x_i \rangle + \theta_0 \geq +1 \text{ ise bu durumda } y_i = +1 \text{ (Şekil 2.52’de mavi noktalar)}$$

$$\langle w, x_i \rangle + \theta_0 \leq -1 \text{ ise bu durumda } y_i = -1 \text{ (Şekil 2.52’de kırmızı noktalar)}$$

Devamında $y_i=1$ ya da -1 yazılırsa Denklem 2.37 elde edilir.

$$y_i(\langle w, x_i \rangle + \theta_0) \geq 1 \quad (2.37)$$

DVM algoritmasının giriş kısmında lineer sınıflandırma yaklaşımında çizilen çizginin sınıfları doğrusal olarak ayırdığından bahsedilmişti. Bu çizilen doğrular bir önceki sayfada görüldüğü üzere farklı şekillerde çizilebilir. Ancak optimal olan çizgiyi çekmek, hem sınıflandırıcının daha doğru tahmin yapmasını hem de sınıfların birbirinden daha iyi ayrılmasını sağlayacaktır.



Şekil 2.43: Destek Vektörleri ve hiperdüzlem çizimleri

Burada H1 ve H2 hiperdüzlemleri arası mesafe olan marjin $2d$ 'nin maksimize edilmesine odaklanılmaktadır. Ayrıca bu iki hiperdüzlemin tam ortasından geçen H hiperdüzlemi ise optimal düzlem olarak ifade edilmektedir. Bu doğrultuda,

$$\langle w, x_i \rangle + \theta_0 = 0 \text{ ise H hiperdüzlemi üstünde,}$$

$$\langle w, x_i \rangle + \theta_0 - d = 0 \text{ ise H1 hiperdüzlemi,}$$

$$\langle w, x_i \rangle + \theta_0 + d = 0 \text{ ise H2}$$

hiperdüzlemi altında kalan alanda olduğu söylenebilir.

Marjin mesafesinin geometrik karşılığı ise veri noktası ile hiper düzlem arasındaki Öklid mesafesidir. Noktaların birbirine olan izdüşümleri sonucu d noktası m kadar mesafeye eşit olur.

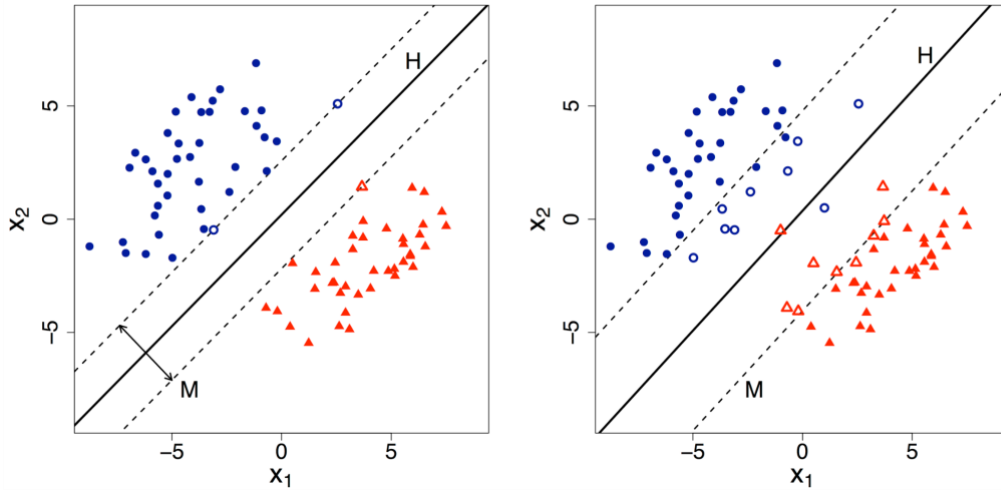
Bu durumda $d = \frac{1}{\|w\|}$ olur ve $2d$ mesafesi olduğundan ötürü $d = \frac{2}{\|w\|}$ ifade edilir. Burada amaç

bu değerin maksimize edilmesini aynı zamanda $||w||$ minimize etmeyi hedeflemektir (Tolun, 2008; Akpınar, 2018; Metlek ve Kayaalp, 2020).

2.12.6.2. Doğrusal Modellerde Esnek (Soft) ve Keskin (Hard) Marjin Sınıflandırma

DVM'lerde keskin marjin ile esnek marjin arasındaki fark, verilerin ayırma çizgisinde tanınan hata toleransına ve ayrılabilirliğe dayanmaktadır. Verilerimiz doğrusal olarak ayrılabilir ise keskin marjin daha uygun olacaktır. Ancak, zıttı bir durum söz konusu olduğunda, bunu yapmak mümkün olmayacaktır. Doğrusal bir sınıflandırıcı bulmayı imkansız kılan veri noktalarına sahip olduğunda, çizgiyi biraz daha tolere ederek bazı veri noktalarının yanlış sınıflandırılmasına izin verilebilir. Bu durumda, esnek marjlı DVM olarak adlandırılır.

Bazen veriler doğrusal olarak ayrılabilir, ancak marjin o kadar küçüktür ki model aşırı uyum sağlamaya veya aykırı değerlere karşı çok hassas olmaya eğilimli hale gelir. Ayrıca, bu durumda, modelin daha iyi güncelleştirilmesine yardımcı olmak için esnek marjlı DVM kullanarak daha büyük bir marjı tercih edilebilir (Rätsch ve diğ., 2001, Hamasuna ve diğ., 2008).



Şekil 2.44: Doğrusal modellerde esnek ve keskin kenar payı çizgileri

Şekil 2.44'te sol tarafta yer alan grafikte keskin kenar yapı çizilirken, sağda yer alan görselde esnek kenar payı çizildiği görülmektedir. Esnek kenarda bazı noktaların alanın dışında kalmasına izin verilir ve buna göre sınıflandırma gerçekleştirilir. Bu noktada marjinin maksimize edilmesi ve bu noktada belirli kısıtların oluşturulmasında aylak değişken (slack variable) kullanılır (Akpınar, 2018).

Aylak değişken, doğru sınıflandırma yapabilmek için herhangi bir i veri noktasının, karar çizgisinin öteki tarafında yer alması durumunda kullanılmaktadır. Burada amaç marj mesafesinin maksimize edilmesi iken, diğer s_i noktalarının minimize edilmesi hedeflenir. Hatta bu minimizasyonun 0'a yakınlaştırılması gerçekleştirmek istenen hedeflerden biridir. Bu sürece düzenleme (regularization) adı verilir.

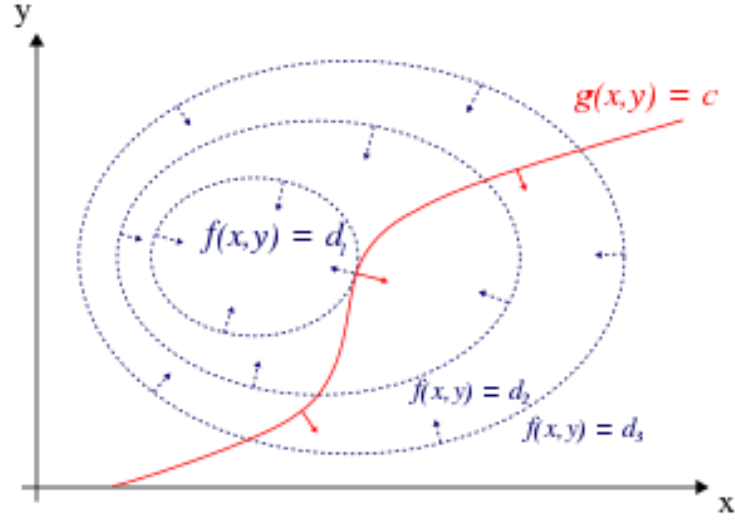
Aylak değişken ξ_i ile temsil edilir. Bu konuda optimal karar çizgisinin çizilebilmesi için amaç fonksiyonu minimize edilir. Burada kilit nokta, ξ_i parametresinin belirlenmesinde bir ölçütün olmaması ve deneme yanılma yoluyla gerçekleştirilmesidir. Bu değer yüksek tutulması hata toleransının sıfıra yaklaşmasını sağlar.

$y_i(\leq w, x > + \theta_0 \geq 1$ denklemi kukla değişken kullanılarak tekrar yazılırsa, $y_i(\leq w, x > + \theta_0 - 1 + \xi_i \geq 0, \xi_i \geq 0 \forall i$ kısıtı kullanılarak Denklem 2.38'de amaç fonksiyonun minimize edilmesidir.

$$\frac{1}{2} \leq w, w > + C \sum_{i=1}^n \xi_i \quad (2.38)$$

Burada C değeri düzenleme için aşırı uyuma karşı oluşturulan ve esnek kenar payını değiştiren hiperparametredir. C değerinin çok düşük olması esnek kenar payına, çok yüksek olması ise keskin kenar payının oluşmasına sebep olabilir.

Yukarıda verilen denklemde iki tane kısıt olduğundan dolayı kısıtlı doğrusal olmayan maksimizasyon problemi ortaya çıkmaktadır. Burada kısıtlar eşitlik ve eşitsizlik altında iki gruba ayrılabilir. Eşitlik kısıtlı problemi Lagranj çarpanları ile çözülürken, eşitsizlik kısıt problemi için ise Lagranj fonksiyonu oluşturulur veya KKT (Karush-Kuhn-Tucker) koşullarını sağlayacak yöntemler aranır.



Şekil 2.45: Lagranj çarpanlarının fonksiyon çizimi

Şekil 2.45'te Lagranj çarpanlarına göre $f(x, y)$ fonksiyonunun $g(x, y) = c$ şeklinde bir kısıt fonksiyona bağlı olarak tepe ve dip noktaların bulunması için her iki fonksiyonun birbirine eşit olması gerekir.

$$\nabla f = \lambda \nabla g \quad (2.39)$$

λ Lagranj çarpanını temsil ettiği varsayılırsa ve değişkenlere göre kısmi türev işlemi gerçekleştirilirse Denklem 2.40 elde edilir.

$$\nabla f - \lambda \nabla g = 0 \Rightarrow \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right) - \lambda \left(\frac{\partial g}{\partial x}, \frac{\partial g}{\partial y} \right) = 0 \quad (2.40)$$

Sonuç olarak iki değişkenli bir ifadeyle, $\mathcal{L}(x, y, \lambda) = f(x, y) - \lambda g(x, y) = 0$ yazılabilir.

Her doğrusal problem kendiyle ilişkili olan başka bir probleme dönüştürülebilir. Bu yapıda mevcut problem primal olarak tarif edilirken, ilişkili olan ise dual olarak nitelendirilir. Amaç fonksiyonundan kısıtların çıkarılmasıyla elde edilebilir.

Bu durumda, a parametresinin maksimize edildiği ve $0 \leq a_i \leq c, i = 1$ kısıtı ile Denklem 2.41'de yer alan a parametresine bağlı bir hale dönüşmüştür.

$$\mathcal{L}(a) = \sum_{i=1}^n a_i - \frac{1}{2} \sum \sum_{i=1}^n y_i y_j a_i a_j \leq x_i, x_j > \quad (2.41)$$

2.12.6.3. Doğrusal Olmayan Sınıflandırma

Doğrusal DVM'nin birçok problemde başarılı sonuçlar elde etmesine rağmen bazen veri kümeleri üzerinde ayırt edici olmayabilir. Bu duruma DVM türlerinden biri olan "doğrusal olmayan sınıflandırıcılar" ile çözüm üretilmeye çalışılır. Karmaşık veri kümelerini açıklamak için düşük seviyede boyutlar veri analizinde yeterli olmayabilir. Boyutu arttırıldığında işlem süreleri de artacağı için model kurma aşaması zaman, maliyet ve kaynak açısından zorlu bir hale dönüşür.

DVM sınıflandırma probleminde kullanılan kernel noktaların üs fonksiyonuna tabii tutularak ve ardından çarpma işlemi uygulayarak çekirdek hilesi kullanılır. Çekirdek fonksiyonu, verileri doğrusal olarak ayrılabilir olma olasılığının daha yüksek olduğu bir özellik uzayına dönüştürür. Bu, 'çekirdek hilesi' olarak bilinir. Çekirdek hilesinin temelinde yer alan ana fikir, verinin daha üst boyutta bir uzaya sığdırılmasına dayanmaktadır.

DVM algoritması etiketli eğitim setine olan ihtiyacı önemli ölçüde azaltabildikleri için metin ve hiper metin sınıflandırmasında oldukça başarılı performans gösterirler. Aynı zamanda, görüntü işleme probleminde de kullanılabilirler. Literatüre bakıldığında DVM algoritmasının tıp, havacılık, görüntü işleme ve biyoenformatik gibi alanlarda kullanıldığı görülmektedir.

Bu kısımda veriler bir düzlem ile ayrılmıştır ancak iki boyutlu bir yapıda olmadığı da ortadadır. Destek vektör makinelerini sınıflandırma probleminde uygulamak için bir çekirdek fonksiyonu seçmek gerekir ve veriler için hangi çekirdeğin en iyi şekilde çalıştığını bulabilmek için standart bir kural yoktur.

Verinin sınıflandırılmasında çekilen çizginin bir üst boyuta çıkarılması $\phi(x)$ notasyonu ile sağlanmakta olup haritalama (mapping) fonksiyonu olarak nitelendirilir. Çekirdeklerde $K(x,x)$ eşleme (mapping) fonksiyonunda $\phi(x)$ karesi olarak işleme alınır (Akpınar, 2018, Gürsakar, 2021).

U ve v gibi iki vektör bulunduğu durumda kernel Denklem 2.42 olarak kullanılacaktır.

$$K(u, v) = \phi^T(u)\phi(v) \quad (2.42)$$

DVM'de kullanılan farklı çekirdek hileleri Tablo 2.10'da çekirdek tipi, eşitlik ve parametre şeklinde verilmiştir.

Tablo 2.8: DVM’de kullanılan farklı çekirdek hileleri (Akpınar, 2018’den alınmıştır.)

Çekirdek Tipi	Eşitlik	Parametreler
Polynomial	$K(u, v) = (au^T + b)^d$	$a, b, d \geq 2$
Gaussian Radial Basis	$K(u, v) = \exp\left(-\frac{\ a - b\ ^2}{2\sigma^2}\right)$	a, b, σ
Neural Network Sigmoid	$K(u, v) = \tanh(a * u^T * v + b)$	a, b
Linear	$K(u, v) = u^T * v$	Yok
Power	$K(u, v) = u^T * v^d$	$d \geq 2$

2.12.7. Sade Bayes (Naive Bayes)

Sade Bayes sınıflandırıcı, İngiliz matematikçi Thomas Bayes’in kendi adını verdiği Bayes olasılık teorisine dayanır. Makine öğrenmesinde sınıflandırma problemine yönelik kullanılan olasılıkçı ve popüler algoritmalarından biridir (Kartal, 2015; Silahtaroglu, 2016; Taha, 2017).

Algoritmanın adında sade veya naif sözcüğü, veri kümesine ait tüm özniteliklerin birbirinden bağımsız olduğu varsayımını gerçekleştirmesinden kaynaklanmaktadır. Naive Bayes algoritması birçok sınıflandırma görevinde iyi performans göstermekle birlikte özellikle metin sınıflandırma problemleri için oldukça uygun bir yapıya sahiptir (Zhang ve Su, 2004).

P(A|B): Belirlenen B için A’nın koşullu olasılığıdır.

P(B|A): Belirlenen A için B’nin koşullu olasılığıdır.

P(B) ve P(A): A ve B olayları için önsel (prior) olasılıktır.

$P(X|Y) = P(x_1, x_2, \dots, x_n|y)$ her x değeri bir özniteliği temsil ettiği varsayılırsa Denklem 2.43 elde edilir.

$$P(X|Y) = \frac{P(Y|X) * P(X)}{P(Y)}$$

$$= P(x_1, x_2, \dots, x_n | y) * P(x_2 | x_3, \dots, x_n, y) \dots P(x_n | y) \quad (2.43)$$

Denklem 2.43'te yer alan eşitlikte her değişken için bir olasılık hesaplayarak zincir kuralı (chain rule) uygulanır ve Denklem 2.44 elde edilir.

$$P(X|y) = P(x_1|y) * P(x_2|y) \dots P(x_n|y) \quad (2.44)$$

Denklem 2.45'te yer aldığı üzere sınıflandırma yapmak için tüm olasılıklar Bayes kuralına göre olasılıkların çarpılarak önsel olasılığa bölünmesiyle elde edilir.

$$P(y|X) = \frac{P(x_1|y) * P(x_2|y) \dots P(x_n|y) * P(y)}{P(x_1) * P(x_2) \dots P(x_n)} \quad (2.45)$$

Son olarak bu olasılıklar çarpılarak sonucun maksimum değeri elde edilir. Denklem 2.46'te çıktı özniteliği sadece ikili sınıfa ait olduğu görülmektedir.

$$y = \operatorname{argmax}_y P(y) \prod_{i=1}^n P(x_i|y) \quad (2.46)$$

Örneğin, ikili sınıflandırma probleminde Naive Bayes algoritması, belirli bir metin parçasını spam ve spam olmama sınıfına ait olma olasılığını metindeki belirli kelimelerin frekanslarına göre hesaplayabilir.

Naive Bayes algoritmasının en önemli avantajlarından biri, uygulamasının çok basit olması ve doğru tahminler yapmak için çok az veriye ihtiyaç duymasıdır. Ayrıca nispeten hızlıdır, bu da onu büyük veri kümeleri veya hızın önemli olduğu uygulamalar için iyi bir seçim haline getirir. Ancak, Sade Bayes algoritması tüm tahmin özniteliklerin bağımsız olduğunu varsayar, bu durum gerçek verilerde çok fazla karşılaşılmayan bir durumdur. Bu durum algoritmanın gerçek veriler kullanım durumlarında uygulanabilirliğini sınırlamaktadır. Son olarak çıktı değerlerin tahmini bazı durumlarda çok düşük performans göstermekle beraber olası çıktıların sebeplerinin bu bağlamda araştırılması çok fazla önerilmez (Alpaydın, 2018; Geron, 2019; Wickramasinghe ve Kalutarage, 2021).

2.12.8. Lojistik Regresyon Analizi

1958 yılında istatistikçi David Cox tarafından geliştirilen lojistik model (Cramer, 2003) (diğer adlarıyla logit model veya logit regresyon), bir olayın log-olasılığının (log-odds) bir veya daha fazla bağımsız özniteliğin doğrusal kombinasyonu olmasıyla bir olayın gerçekleşme olasılığını modelleyen istatistiksel bir modeldir (Géron, 2019; James, 2013). Temelinde, özniteliklerle çıktı olasılığı arasında bir ilişki bulmanın üzerine kuruludur (Tabachnick ve Fidell, 2013).

Lojistik regresyon ikili, çoklu ve sıralı sınıflandırma problemlerinde kullanılabilir. İkili Lojistik Regresyon, modeli bağımlı değişkenin yalnızca iki olası sınıfa ait olması durumudur. Örneğin, müşterinin kayıp veya kayıp değil olarak nitelendirilmesi. Multinomial Lojistik Regresyon modeli bağımlı değişkenin sıralama olmaksızın yalnızca iki 3 veya daha fazla olası sınıfı ait olması durumudur. Örneğin, İç Anadolu, Ege veya Marmara bölgeleri gibi farklı sınıfa ait gruplar. Ordinal lojistik regresyon modeli ise bağımlı değişken, sıralamaya sahip yalnızca iki adet 3 veya daha fazla olası sınıfa sahip olması durumudur. Örnek: 1'den 5'e yıldız derecelendirme yapılması olarak sıralanabilir.

Parametre tahmini için Sıradan En Küçük Kare (Least Square) yöntemini kullanan doğrusal regresyon modelinin aksine, lojistik regresyonda Maksimum Olabilirlik Tahmini (Maximum Likelihood Estimation) kullanılmaktadır. Lojistik regresyon, maksimum olabilirlik tahmini yöntemi kullanarak gözlemlediğimiz verileri elde etme olasılığının maksimum olduğu regresyon katsayıları kümesi oluşturur. Lojistik regresyondaki amaç, eldeki veriyi en uygun lojistik fonksiyonuyla a ve b katsayılarını optimize ederek hesaplanmasını içerir. En temel özelliği ise $+\infty$ ile $-\infty$ arasında değer alması ve $0 < x < 1$ arasında sonuç çıktısı üretmesidir (Wright,1995; LaValley,2008; Baydemir, 2014).

Aşağıda ikili lojistik regresyon algoritmasının varsayımlarına yer verilmiştir (Kartal, 2018). Bunlar;

- Lineer regresyonda olduğu gibi bağımlı ve bağımsız değişkenler arasında bir doğrusal ilişki olmadığını,
- Çıktı özniteliğinin iki sınıftan birine ait olduğunu ve gözlemlerin birbirinden bağımsız olduğunu,
- Varyans eşitliğine (homoscedasticity) belirli aralıkta öznitelik değerlerine ve normal dağılıma ihtiyaç duymadığı varsayılmaktadır.

$$P(y = I|x) \quad (2.47)$$

$$P(X) = \beta_0 + \beta_1 X \quad (2.48)$$

Doğrusal regresyon aykırı değerlerden oldukça etkilenir. Bu soruna çözüm olarak ise log-olasılığı fonksiyonu veya logit fonksiyonu kullanılır.

Log-olasılığı fonksiyonu, verilerin birbirlerine olan olasılıksal olabilirliğini ifade eder. π bir olayın olasılığı, Θ ise bir olayın olabilirliği olarak varsayılırsa, Denklem 2.49 ve 2.50 olarak formüle edilir. Eğim katsayısı (B), x değiştikçe p'nin değişim oranı olmak yerine, eğim katsayısı X değiştikçe log-olasılığı değişim oranı olarak yorumlanmaktadır. Olasılık oranı, B'nin varlığında A'nın olasılığı ile B'nin yokluğunda A'nın olasılığının oranı olarak tanımlanır. Veyahut bu durumun tersi de olabilir.

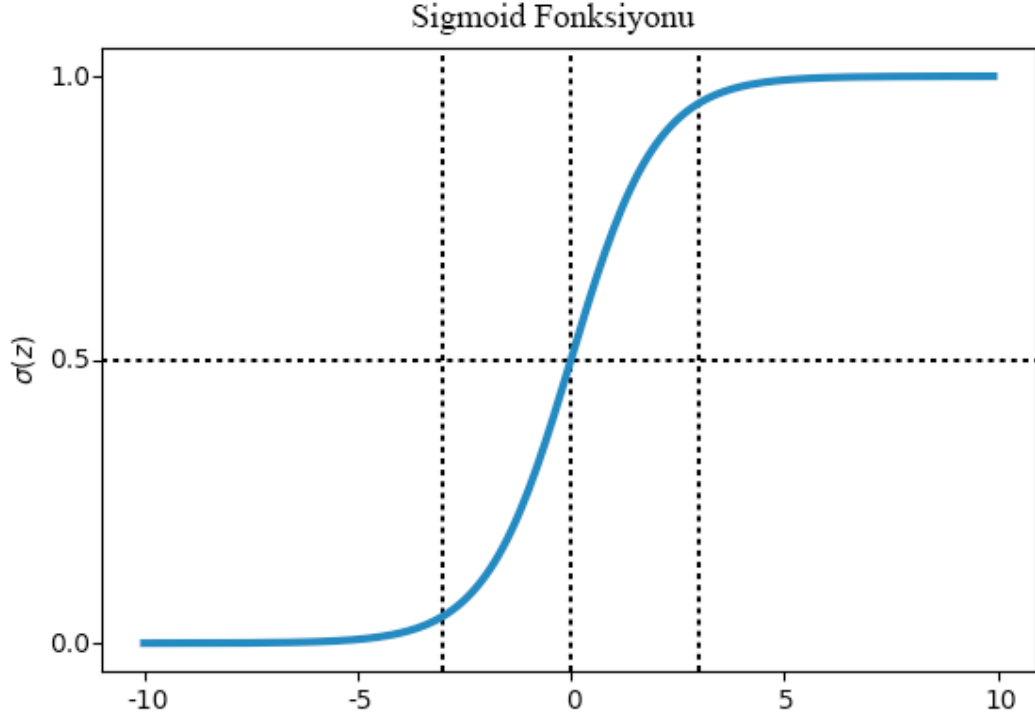
$$Olabilirlik = \frac{P}{1 - P} \quad (2.49)$$

$$Olabilirlik = \frac{\frac{p_1}{1 - p_1}}{\frac{P_0}{1 - P_0}} \quad (2.50)$$

Olasılık ve olabilirlik kavramları farklıdır. Rastgele atılan bir zarın 6 gelme olasılığı $\frac{1}{6}$ olarak ifade edilirken, olabilirliği ise $\frac{1}{5}$ olarak ifade edilir.

Lojit ya da log-üstünlük fonksiyonu, 0'dan 1'e ve negatif sonsuzdan pozitif sonsuza kadar olan olasılık değerlerini temsil eden bir fonksiyondur. Bu fonksiyon, 0 ile 1 arasındaki değerleri X eksenine yerine Y eksenine boyunca sınırlayan sigmoid fonksiyonunun tersidir. Logit fonksiyonu 0 ile 1 arasında olduğundan, bu fonksiyon sıklıkla olasılıkların hesaplanmasında kullanılır. p olasılığı tahmin ettiği ve $a + bX$ modelin doğrusallığını temsil ettiği varsayılırsa Denklem 2.58 lojik fonksiyon olarak ifade edilir.

$$Lojit Fonksiyon = \log\left(\frac{p}{1 - p}\right) \Rightarrow P = \frac{e^{a+bX}}{1 + e^{a+bX}} \quad (2.51)$$



Şekil 2.46: Sigmoid fonksiyon (Molnar, 2022'den uyarlanmıştır.)

Denklem 2.51'de fonksiyonun tersini alındığında, sigmoid fonksiyonu elde edilir ve S şeklinde bir eğri verir. Bu aynı zamanda karar çizgisi (decision line) olarak da adlandırılır. Her zaman $0 < p < 1$ arasında değişen bir olasılık değeri verir. 0.5'in üzerinde yer alması bir sınıfa, altında yer alması ise diğer sınıfa ait olduğu sonucunu taşır.

$$\beta_0 + \beta_1 X == 0 \text{ ise } p \text{ değeri } 0.5,$$

$$\beta_0 + \beta_1 X > 0 \text{ ise } p \text{ değeri } 1'e \text{ doğru,}$$

$$\beta_0 + \beta_1 X < 0 \text{ ise } p \text{ değeri } 0'a \text{ doğru gidecektir.}$$

Lojistik Regresyon Analizinde (LRA) veri setinin eğitiminde amaç, doğrusal modelde en iyi ağırlıkları seçmektir. Makine öğrenmesinde, maliyet fonksiyonu (cost function) olarak tanımlanan yöntemle optimize edilerek istenen optimum ağırlıklar hesaplanır. Maliyet fonksiyonu algoritmanın ulaşmaya çalıştığı bir hedef noktası olarak nitelendirilebilir. Gerçek ile tahmin edilen sınıf arasındaki ortalama hatanın ne kadar olduğunu kontrol eder. Çok büyük ve çok küçük hatalara cezalandırma işlemi uygular. LRA'da bu fonksiyon Çapraz Entropi

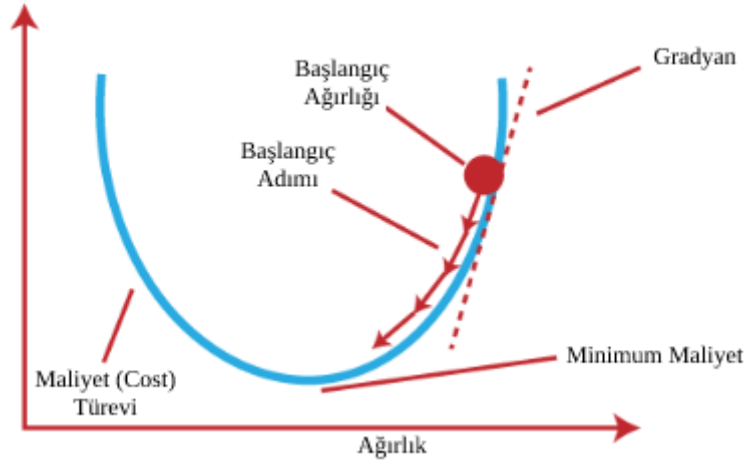
(Cross-Entropy) ya da Logaritmik Kayıp (LogLoss) olarak adlandırılır. Burada amaç maliyet fonksiyonunu minimize ederek optimizasyon gerçekleştirmektir.

$$J(\theta) = -\frac{1}{m} \sum \left[y^{(i)} \times \log(h_{\theta}(x^{(i)})) + (1 - y^{(i)}) \times \log(h_{\theta}(x^{(i)})) \right]_{i=1}^m \quad (2.52)$$

Maliyet fonksiyonu uygulanırken model hatayı minimize etmeye çalışır. Peki bu minimize edilmeye çalışılan değerlere nasıl ulaşılır? Bu sorunun cevabı Gradyan İniş (Gradient Descent) olarak verilebilir. Gradyan iniş, her veri için kayıp fonksiyonuna bağlı kalarak ağırlıkları değiştirir. Her girdi ile çıktı değerler arasında Logaritmik Kayıp fonksiyonuna göre hesap yapar. Her noktada maliyet fonksiyonunun eğimini elde etmek için ağırlık ve sapmanın kısmi bir türevi alınır. Eğime bağlı olarak, gradyan iniş bias ve ağırlık değerlerini günceller, ardından eğitim aşamasında yeni değerler üzerinden yeniden tekrarlar.

Bu yinelemeli yaklaşım (iterative approach), minimum hataya ulaşılan ve gradyan inişi maliyet fonksiyonunu daha fazla minimize etmeyi bitirene kadar tekrarlanır. Denklem 2.52’de α ile temsil edilen öğrenme oranını (learning rate) ayarlayarak en yüksek minimum değere ulaşma hızını değiştirilebilir (Bottou, 2012). Yüksek öğrenme oranı ağırlıkları daha büyük aralıklarla değiştirirken, düşük öğrenme oranı ağırlıkları daha düşük aralıklarla değiştirir.

Öğrenme oranını belirlemede değer çok düşükse, modelin en iyi ağırlıklara ulaşması için çok uzun süre işlem yapması gerekecektir; ancak durum tam tersi yani öğrenme oranı çok yüksekse, modelin en iyi ağırlıklara ulaşması için geçen sürede en iyi ağırlıkların göz ardı edilme riski vardır (Goodfellow ve diğ., 2016).



Şekil 2.47: Gradyan İnişin canlandırması (Raschka, 2022'den uyarlanmıştır.)

y çıktı özniteliği, x her bir öznitelik, a öğrenme oranı, m ise öznitelik sayısını temsil ettiği varsayıldığında Denklem 2.60 ile ifade edilebilir.

$$\theta_j \leftarrow \theta_j - \alpha \frac{1}{m} \sum_{i=1}^m a_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_j^{(i)} \quad (2.53)$$

Burada üzerinde durulması gereken bir diğer husus ise düzenleştirme (regularization) uygulanmasıdır. Her algoritma da olduğu gibi LRA'da da hiperparametreler mevcuttur. Bu hiperparametrelerin düzenlenmesi önemlidir. Düzenleştirme kapsamında kullanılan bazı hiperparametreler aşağıda verilmiştir.

Çözücü (Solver): Optimizasyon probleminde kullanılacak algoritmadır. newton-cg, lbfgs, liblinear, sag ve saga gibi farklı çözücüler mevcuttur.

- **lbfgs:** Diğer yöntemlere kıyasla nispeten iyi performans gösterir ve çok fazla bellek tasarrufu sağlar, ancak bazen minimum değere yaklaşma konusunda eksik kalabilir.
- **sag:** Hem örnek sayısı hem de öznitelik sayısı büyük olduğunda, büyük veri kümeleri için diğerlerinden daha hızlıdır.
- **saga:** Çok sınıflı lojistik regresyon için tercih edilen çözücü olup çok büyük veri kümeleri için de uygundur.
- **newton-cg:** Hessian matrisi kullanan bir yöntemdir ancak hesaplama süresi uzun olduğundan dolayı çok tercih edilmez.

- **liblinear:** Yüksek boyutlu bir veri seti olduğunda bu yöntem önerilir - büyük ölçekli sınıflandırma problemlerine çözüm üretebilir. (Pedregosa, 2018)

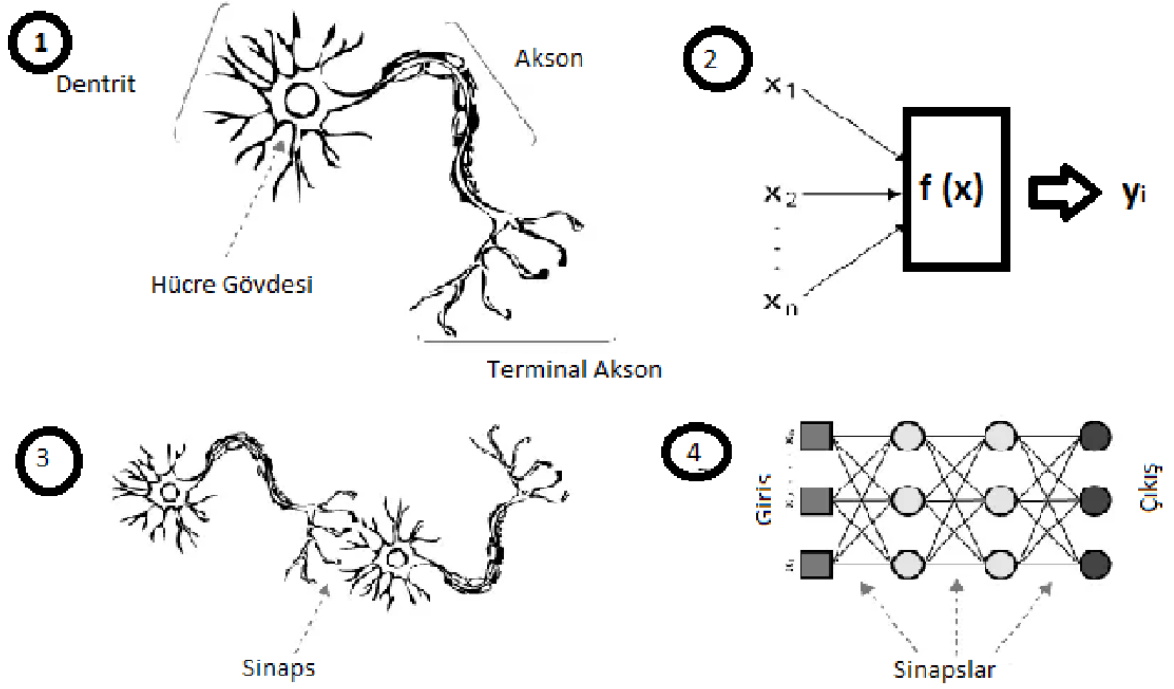
Ceza (Penalty): Düzenleştirme sırasında modelin genelleme hatasını düşürmek ve aşırı uyuma karşı çözüm üretebilmesini sağlamak için kullanılır. Modeli genel yapıdan karmaşık yapıya doğru değiştirir. L1, L2 ve Elastic olmak üzere üç farklı seçenek mevcuttur. Ayrıca bazı cezaların bazı çözümlerde çalışmadığı da bilinmektedir.

C değeri: Modelin düzenleştirme gücünü belirten pozitif ondalık değer alan hiperparametredir. C değeri aşırı uyum problemini dengelemek için ceza uygulama ile çalışır. Daha küçük değerler daha güçlü düzenli bir durumu belirtirken ve yüksek değer, modele eğitim verilerine yüksek ağırlık vermesini söyler.

2.12.9. Yapay Sinir Ağları

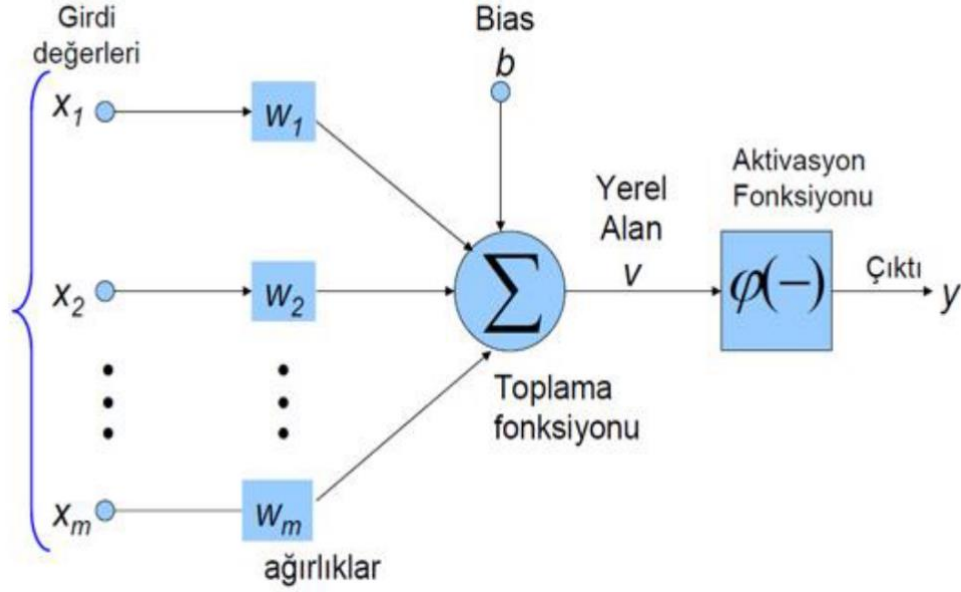
Sinir ağları, insan beyninin yapısı ve işlevinden esinlenen makine öğrenimi algoritmalarından biridir. Teorik olarak insan beyninin bilgiyi işleme modelinin temel alındığı yapay sinir ağlarında amaç yeni bilgiler üretme ve keşfedebilme yeteneklerinin herhangi bir yardım olmaksızın makine aracılığıyla otomatikleştirilmesi amacıyla geliştirilen bilgisayar sistemleridir (Öztemel, 2012). Karmaşık bir sistem olması sebebiyle yapay sinir ağlarının tanım kapsamı farklılık göstermektedir. Yapay sinir ağları (YSA), insan beyninin öğrenme yolunu taklit ederek beynin öğrenme, hatırlama, genelleme yapma yolu ile topladığı verilerden yeni veri üretebilme gibi temel işlevlerini gerçekleştirildiği bilgisayar yazılımlarıdır (Öztürk ve Şahin, 2018). Yapay sinir ağları; insan beyninden esinlenerek, öğrenme sürecinin matematiksel olarak modellenmesi uğraşı sonucu ortaya çıkmıştır (Chollet, 2021).

Sinir ağları özellikle görüntü ve konuşma tanıma, doğal dil işleme ve karar verme gibi geleneksel algoritmaların üstesinden gelmesi zor olan karmaşık problemleri çözmek için çok uygundur. Özellikle finanstan mühendisliğe ve tıpa kadar birçok alanda farklı problemlerin çözümünde kullanılmaktadır (Narendra ve Parthasarathy, 1990; Efe ve diğ., 1999; Efe ve Kaynak, 1999; Kelleher, 2019)



Şekil 2.48: Biyolojik sinir ağının yapay sinir ağıyla karşılaştırmalı canlandırması (Goncalves ve diğ., 2013'den uyarlanmıştır.)

YSA'da yapılar birbirine hiyerarşik olarak bağlı ve paralel çalışabilen yapay hücrelerden oluşmaktadır. Teknik olarak, temelde girdi olarak verilen veriden bir çıktı seti oluşturulmasına dayanır. Bunu gerçekleştirebilmesi için ise bir öğrenme sürecine tabii tutulur. Bu yapılar yapay nöron adı verilen ve bilgiyi işleyen ve ileten birbirine bağlı birimlerden oluşurlar. İnsan beyninin biyolojik yapısı ve yapay sinir ağlarının temelinde yatan bu süreç Şekil 2.48'de canlandırılmıştır. Karşılaştırma yapıldığında nöronlar işlemci elemanını, dentritler toplam fonksiyonunu, hücre gövdeleri transfer fonksiyonunu, aksonlar yapay nöron çıkışlarını ve sinapslar ise ağırlıkları temsil etmektedir (Öztemel, 2012; Özen, 2016; Akpınar, 2018).



Şekil 2.49: Yapay sinir ağı çalışma prensibi

Yapay sinir ağının yapısında yer alan hücreler mühendislikte “proses elemanları” olarak adlandırılır. Bunlar beş farklı temel elemandan oluşmaktadır (Şekil 2.50).

- Girdiler ($x_1, x_2, \dots$), yapay sinir ağına dış dünyadan aktarılan girdi bilgileridir.
- Ağırlıklar ($w_1, w_2, \dots$) sinir ağına gelen bilginin önemi ve hücreye olan etki derecesidir. Ağırlıklar negatif ve pozitif değer alabilirler. Eksi değer alması önemsiz olduğunu anlamına gelmemekle birlikte pozitif değer alması da önemli olduğunu temsil etmemektedir. Ancak 0 olması durumunda etkinin olmadığı anlamına gelir.
- Toplama fonksiyonu, hücreye gelen net bilginin hesaplanmasında görev alır. En yaygın olarak kullanılan ağırlıklı toplamdır. Her bir girdi değerinden gelecek değerler kendi ağırlıklarıyla çarpılarak toplama işlemi uygulanır.

Girdiler x_i , ağırlıklar w_i , n ise hücreye gelen toplam girdi sayısı olarak varsayıldığında *net* değeri,

Toplam fonksiyonu $net = \sum_{i=1}^n x_i * w_i$,

Çarpım fonksiyonu $net = \prod_{i=1}^n x_i * w_i$,

Maksimum ve minimum fonksiyonu $net = \min(x_i * w_i)$ veya $\max(x_i * w_i)$

Çoğunluk fonksiyonu $net = \sum_{i=1}^n \text{sgn}(x_i * w_i)$

Kümülatif toplam fonksiyonu $net = net(eski) + \sum_{i=1}^n x_i * w_i$ olarak hesaplanır.

Aktivasyon fonksiyonu ise toplama fonksiyonundan gelen net bilginin işlenerek bir çıktı değeri oluşturulmasıdır. Farklı aktivasyon fonksiyonlarına alanyazında yer verilmiştir. Bunlar; sigmoid, step, sinus, eşik değeri ve hiperbolik tanjant olarak sıralanabilir. Toplama fonksiyonundan gelen değerler net , uygulanan aktivasyon fonksiyonu ise f olarak varsayıldığında çıktı değeri,

Doğrusal (Lineer) fonksiyon $f(net) = NET$

Step fonksiyonu $f(net) = \text{eğer } net > \text{eşikdeğer} = 1$

$= \text{eğer } net \leq \text{eşikdeğer} = 0$

Sinus fonksiyonu $f(net) = \sin(net)$

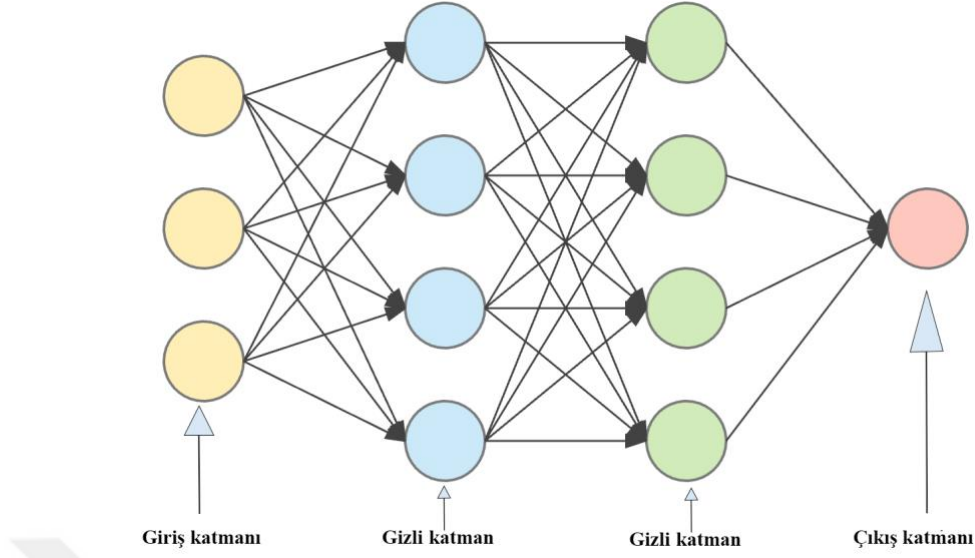
Eşik değeri fonksiyonu $f(net) = \text{eğer } net \geq 1 = 1$

$= \text{eğer } net \leq 0 = 0$

$= \text{eğer } 0 < net < 1 == net$

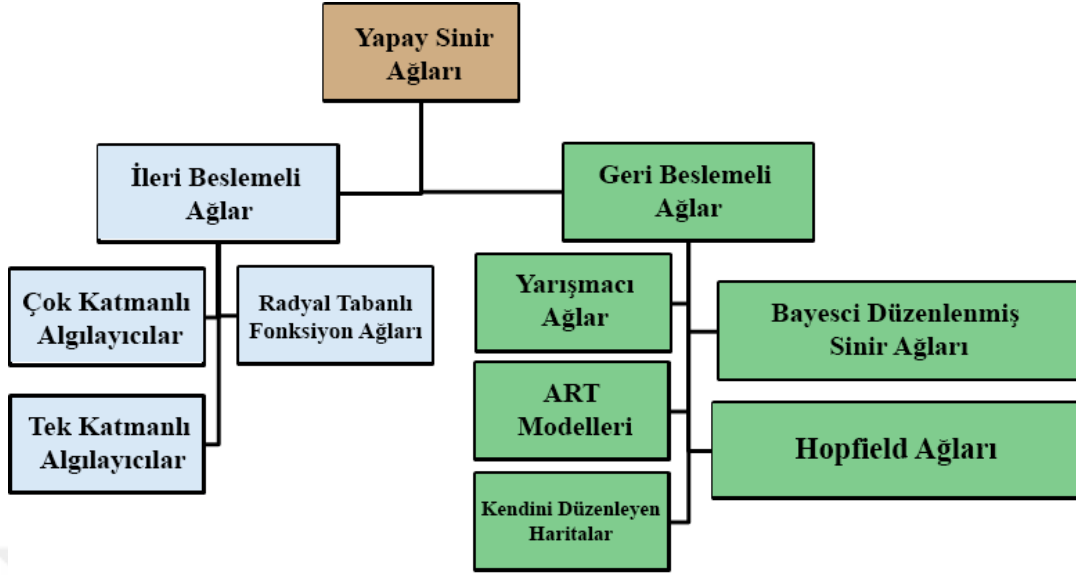
Hiperbolik tanjant fonksiyonu $f(net) = \frac{(e^{net} + e^{-net})}{(e^{net} - e^{-net})}$

Son proses elemanı ise çıktıdır. Bu değeri aktivasyon fonksiyonunun uygulanması sonucu oluşan değerdir.



Şekil 2.50: Basit sinir ağı modeli ve katmanları

Bir yapay sinir ağı, girdi katmanı, ara katman ve çıktı katmanından oluşur. Girdi katmanı dış dünyadan gelen bilgilerin ara katmanlara iletilmesinde görevlidir. Ara katman gelen bilgilerin işlenerek çıktı katmanına yönlendirildiği katmandır. Bilgilerin işlenmesi bu katmanda gerçekleşir. Sayısı artırılıp azaltılabilir. Çıktı katmanı ise ağı girdilerden işlenerek gelen verinin bir çıktıya dönüştürüldüğü katmandır. Üretilen çıktılar dış dünya ile paylaşılır.

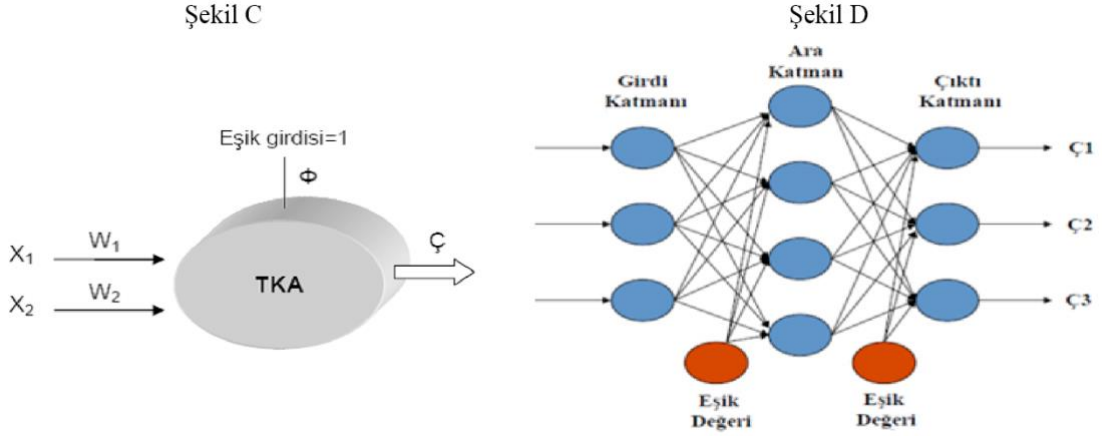


Şekil 2.51: Yapay sinir ağı sınıflandırması

YSA ileri beslemeli ağlar ve geri beslemeli ağlar olarak iki gruba ayrılır. Bu grupların altında farklı algılayıcılar, sinir ağları yapıları, modeller ve haritalar yer almaktadır. Şekil 2.51’de YSA ağ türleri şekil olarak verilmiştir.

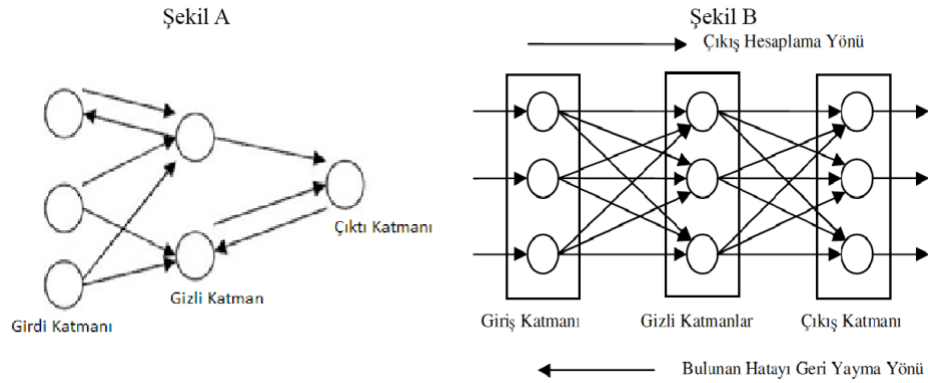
Bir sinir ağını eğitmek için ağı büyük bir veri kümesi beslenir ve ağ, giriş verilerine göre nöronlar arasındaki bağlantıların gücünü ayarlar. Bu süreç ağı eğitilmesi olarak bilinir ve ağı kalıpları tanımayı öğrenmesini ve girdi verilerine dayalı kararlar almasını sağlar.

Sinir ağlarının en önemli avantajlarından biri öğrenme ve yeni verilere uyum sağlayabilmesidir. Daha fazla veriye maruz kaldıkça performanslarını sürekli olarak artırabilirler. Bu da yapay sinir ağlarını çeşitli problemleri çözmek için yüksek performanslı araç haline getirir. Bununla birlikte, sinir ağlarının eğitilmesi için büyük miktarda veri ve hesaplama kaynağı gerekir ve yorumlanması zor olabilir. Bu nedenle belirli problem çözümlerinde daha nadir kullanılır. Ayrıca yapay sinir ağlarını en büyük dezavantajı kara kutu (black box) bir yapıya sahip olmasıdır. En temel düzeyde "kara kutu", derin sinir ağları için tüm nöronların nihai çıktıya ulaşmak için birlikte nasıl çalıştığını bilmediğimiz anlamına gelir. Çoğu zaman herhangi bir nöronun kendi başına ne yaptığı bile net değildir (Blazek ve Lin, 2021). Bu doğrultuda, yapay sinir ağı tahmin sonuçlarını nasıl oluşturduğu hakkında açıklama becerisine sahip değildir. Bu durum ağı yapısına olan güveni sarsarken aynı zamanda işlem süresinin de uzamasına sebep olmaktadır.



Şekil 2.52: Tek ve çok katmanlı algılayıcıların canlandırması

Tek katmanlı ağlar yalnızca girdi ve çıktılarından oluşur. Tek katmanlı algılayıcılarda çıkış fonksiyonu doğrusaldır ve 1 veya -1 değerlerini alır. Çok katmanlı algılayıcılar ise yapısal olarak doğrusal olmayan aktivasyon fonksiyonuna sahip birçok nöronun belirli bir üstünlükle ilişkilendirildiği bir yapıya sahiptir. Bazı yöntemlerin verimsizliği, çok katmanlı algılayıcıların geliştirilmesinde rol oynar.



Şekil 2.53: İleri ve Geri Beslemeli Yapay Sinir Ağlarının Canlandırması

İleri beslemeli sinir ağlarında, nöronlar girdiden çıktıya kadar düzenli katmanlar halinde düzenlenir. Katmanların yalnızca kendilerinden sonra gelen katmanla iletişimi vardır. Yapay sinir ağının girişine giren bilgiler değişmeden merkez noktanın yani hücrelere iletilir. Gizli katman daha sonra buna uygun olarak çıkış katmanından geçer ve dış ortama iletilir (Hamzaçebi, 2021).

Regresyon içinse,

YSA’larda geriye ve ileriye yayılım algoritmalarının olduğunu ve bunların nasıl çalıştığı bölümün girişinde verilmişti. Regresyonda geriye yayılım, hataya göre ağırlık ve bias (yanlılık) ayarlamalarını yapmak için kullanılır ve hatanın kendisi, performans değerlendirme ölçütü ile ölçülebilir.

Yapay sinir ağına başlarken gizli katmanın ağırlıkları $x^l_i U$ formülüyle temsil edilir ve x^l ’de kullanılan üs işareti transpozunu ifade eder. Gizli katmanın çıkışında model Denklem 2.61 formülünü uygular. Burada ξ_i ve x_{tj} değerleri verildiğinde, w ve u en iyi değerleri aranır.

$$\frac{1}{e^{-x^l_i U}} \quad (2.61)$$

$$\xi_t = w_0 + \sum_{k=1}^n w_k \left(\frac{1}{1 + \exp(\sum_{j=1}^l -x_{ij} u_{ik})} \right) \quad (2.62)$$

Burada çıkış nöronları için doğrusal bir aktarım fonksiyonu ve gizli katman nöronlarında sigmoid bir aktarım fonksiyonu kullanılır.

3. MALZEME VE YÖNTEM

Bu tez kapsamında biri çok oyunculu rol yapma oyunu diğeri ise mobil şans oyunu olmak üzere iki farklı veri seti üzerinde çalışılmıştır. Bu veri setleri çalışma kapsamında karışıklık oluşturmaması için Blade & Soul ve Lengebear veri setleri olarak adlandırılacaktır. Veri setleri ile ilgili bilgiler ve yapılan işlemler bu bölümde detaylı şekilde verilmiştir. Ayrıca, verilerin işlenmesinde Bölüm 2.9.3'te yer verilen CRISP-DM metodolojisine sadık kalınmıştır. Ek olarak, uygulanan metodolojinin her alt başlığının altında veri setleri için ayrıca bölümler açılmış ve buna göre detaylandırılmıştır.

Tez kapsamında kullanılan veri setleri kapsamında, veri ön işleme, veri hazırlama, model performans değerlendirmesi ve uygulama için Python dilinde açık kaynak "Scikit Learn" kütüphanesi, veri görselleştirme ve veri düzenlenmesi için ise Tableau kullanılmıştır.

3.1. İŞİN ANLAŞILMASI

3.1.1. Araştırmanın Problemleri

Blade & Soul veri setinde MMORPG oyununa ait sınıflandırma ve regresyon problemine farklı makine öğrenmesi yöntemleri uygulanarak çözüm üretilmeye çalışılmıştır. Blade & Soul verisetinde, oyuncuların oyundan ayrılıp ayrılmayacağını makine öğrenmesi yöntemleriyle tespit edilmesine dayalı olarak ikili kayıp sınıflandırması, oyuncuların oyunda geçirdiği süreyi tahmin etmek içinse regresyon modeli uygulanmıştır.

Lengebear veri setinde kayıp etiketleri veriyi araştırmacılara sunan şirket tarafından verilmemiştir. Bu nedenle farklı günler baz alınarak karşılaştırmalı sonuçlara yer verilmiştir. Müşteri kayıp tahminine yönelik makine öğrenmesi yöntemleriyle ikili sınıflandırma problemine çözüm aranmış ve 4, 5 ve 6 günlük kayıp sürelerine göre sınıflandırma sonuçlarının performansı analiz edilmiş ve bu doğrultuda en iyi performans gösteren kayıp süresi gününün tespit edilmesi hedeflenmiştir.

Her iki veri setine yönelik geliştirilen modellerde oyuncu kaybını en aza indirgeyecek bazı göstergeler öznitelik önemliliğine göre belirlenmiş ve şirket için müşteri kaybını önlemede kullanılabilecek önerilere yer verilmiştir.

3.1.2. Araştırmanın Sınırlılıkları

Akademik oyun çalışmaları özellikle son yıllarda popülerlik kazanmış olmasına rağmen, gelir elde etme ve müşteriye koruma gibi konularda şirketlerin çok fazla veri paylaşmadığı belirtilmelidir. Özellikle mobil oyun platformunda oyun içeriği üreten firmaların oyunculara ait veriyi paylaşmaması bir nebze yapılan çalışmaların sınırlılığını oluşturmaktadır. Özellikle, araştırmalar kapsamında İnternet ortamında veriye erişim sağlanması durumunda araştırmacılara açık hale getirilmiş ve şirketin veri paylaşım politikasına sadık kalınmış şekilde kullanılması bilimsel etik açısından da son derece önemlidir. Bu sebepten ötürü araştırmada kullanılan verilerin şirketler tarafından araştırmacılar tarafından kullanılabilir olduğunu gösteren metinler yer almaktadır. Burada bahsedilen potansiyel problemler, en güncel verilere ulaşma konusunda araştırmada sınırlılık olarak karşımıza çıkmaktadır.

3.2. VERİNİN ANLAŞILMASI

Veri toplama, çevrimdışı ve çevrimiçi kaynaklar da dahil olmak üzere birden fazla kaynaktan verileri kazıyarak, yakalayarak ve yükleyerek bir havuzda toplamak anlamına gelir. Yüksek hacimli veri toplama veya veri oluşturma, özellikle ölçekli bir makine öğrenimi projesinin en zor kısmı olabilir.

Blade & Soul veri seti,

Blade & Soul, oyuncuların birbirleriyle etkileşime girebildiği ve oyunda seçtikleri karakterleri yönetebildiği sunucuda birçok oyuncuyla oynanan Haziran 2012'de piyasaya sürülmüş bir MMORPG oyunudur. Blade & Soul oyununa ait yaklaşık 100 GB oyuncu kayıtları verisi araştırmacılarla paylaşılmıştır. Veriler 28 Mart 2016 ile 25 Ağustos 2017 tarihleri arasında belirli dönemlerde toplanmıştır. Şirket tarafından paylaşılan verinin akademik çalışmalarda kullanılabileceğine dair bilgi verilmiş ve araştırmacıların erişimine sunulmuştur. Tablo 3.1 veri setine ilişkin veri toplama sürecini ve iş modeli bilgilerini göstermektedir.

Tablo 3.1: Veri toplama periyodu ve dönem bilgileri

Veri Seti	Veri Toplama Periyodu	Hafta	İş Modeli	Oyuncu Sayısı	Boyut (GB)
Eğitim	Nisan 1, 2016 – Mayıs 11, 2016	6	Üyelik	4,000	48
Test 1	Temmuz 27, 2016 – Eylül 21, 2016	8	Üyelik	3,000	30
Test 2	Aralık 14, 2016 – Şubat 8, 2017	8	Ücretsiz	3,000	30

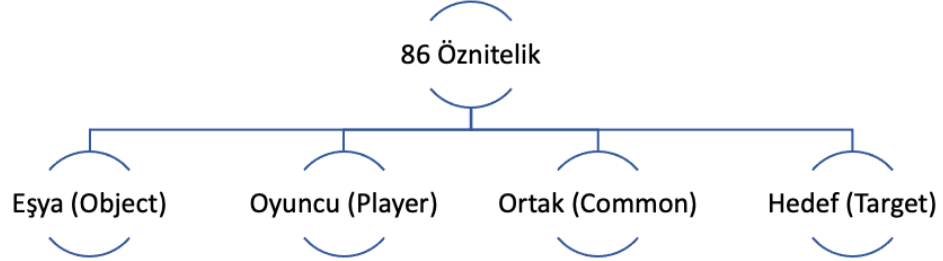
Tablo 3.2’de görüldüğü üzere 10.000 oyuncunun ham verileri işlenerek tek bir veri haline getirilmiştir. Araştırma kapsamında, sınıflandırma ve regresyon problemleri için hedef öznitelik olarak atanan kayıp durumu (churn) ve hayatta kalma süresi (survival time) öznitelikleri veri setinde paylaşılmıştır. Burada şirket tarafından paylaşılan veri kullanma yönergesi her iki problemin çözümünde “hayatta_kalma_süresi” özniteliğinin doğrudan modelleme aşamasında kullanılmamasını içermektedir. Bu sebepten ötürü bu öznitelik sınıflandırma probleminde kullanılmamış, regresyon probleminde ise hedef nitelik olarak karşımıza çıkmaktadır.

Tablo 3.2: Veri seti için kayıp durumu sütun dağılımları

Veri Seti	Kayıp	Kayıp Değil	Toplam
Eğitim	2,650 (%66.25)	1350 (%33.75)	4,000
Test 1	2,092 (%69.73)	908 (%30.27)	3,000
Test 2	2,102 (%70.03)	898 (%29.97)	3,000
Toplam	6,844 (%68.4)	3156 (%31.6)	10,000

Test 1 ve Test 2 verileri ayrı ayrı sunulmuş ve hem Test 1 hem de Test 2 verileri üzerinde sınıflandırma problemine çözüm üretilmesi amaçlanmıştır. Veri seti için kayıp durumuna ait sınıfların dağılımları ve toplam sayıları Tablo 3.2’de verilmiştir. Şirket, Aralık 2016’dan sonra iş modeli değişikliği uygulamış P2P’den (PaytoPlay) F2P’ye (FreetoPlay) geçiş yapmıştır. Test 1 için P2P’den gelir elde ederken, Test 2 veri setinin toplandığı sürede F2P’den gelir elde etmeye başlamıştır. Test 1 ve Test 2 performans değerlendirme ölçütleri ayrı tablolarda verilmiştir. Buradaki amaç, geliştirilecek modelin P2P ve F2P iş modelleri kapsamında performanslarını karşılaştırmaktır.

Oyuncu kayıtları, oyunun analizi için bir oyuncunun davranışını, olayını ve durum değişikliğini anlık olarak kaydeden yapıdır. Oyunda tutulan bu kayıtlar dört grupta değerlendirilmiş ve Şekil 3.1’de verilmiştir.



Şekil 3.1: Özniteliklerin kayıt grupları

- **Oyuncu kategorisi**, oyuncu kimliği, hesap kimliği, ırk, seviye ve sınıf gibi bireysel bilgileri içermektedir.
- **Eşya kategorisi**, eşyanın kimliği, sınıf ve miktar gibi aktörün eylemiyle bağlantılı bilgileri içermektedir.
- **Ortak kategorisi**, kaydın oluşturulması, kayıt hareketi gibi tüm eylemlerin ortak bilgilerini içermektedir.
- **Hedef kategorisi**, rakip karakter kimliği, rakip hesap kimliği, ırkı, seviyesi ve sınıfı gibi bilgileri içermektedir.

Tablo 3.3: Oyuncu kayıt hareketlerinin kategorilerine göre örnek listesi

Kategori	Kayıt Numarası	Kayıt Hareketi	Açıklama	Veri Tipi
Bağlantı (Connection)	1003	EnterWorld	Oyuncu oyun sunucusuna girdiğinde	Nümerik
	1004	LeaveWorld	Oyuncu oyun sunucusuna ayrıldığında	Nümerik
Karakter (Character)	1017	GetMoney	Oyuncunun envantere oyun içi parası arttığında	Nümerik
	1022	GetItem	Envanterdeki eşya sayısı arttığında	Nümerik
Eşya (Item)	2004	DestroyItem	Eşya yok edildiğinde	Nümerik
	2113	RepairItem	Eşya tamir edildiğinde	Nümerik
Yetenek (Skill)	4001	AcquireSkill	Oyuncu yetenek kazandığında	Nümerik
	4002	SkillLevelUp	Oyuncu yetenek yükselttiğinde	Nümerik
Görev (Quest)	5001	AcquireQuest	Oyuncuya bir görev atandığında	Nümerik
	5004	CompleteQuest	Oyuncu bir görevi tamamladığında	Nümerik
Lonca (Guild)	6001	CreateGuild	Oyuncu, lonca yöneticisine oyun parası ödeyerek lonca kurduğunda	Nümerik
	6002	DestoryGuild	Lonca yok edildiğinde	Nümerik

Ham veriler çoğunlukla tamsayı ve float veri türleridir. Ayrıca, 'zaman (time)' niteliği Yıl-Ay-Gün-Saat-Dakika-Saniye biçimindedir. Örneğin: 2016-07-13 18:06:13. Her ham verinin kendi içinde '*oyuncu_kayıt_numarası*' adında bir özniteliği vardır ve oyuncu adını anonimleştirerek verinin yayınlanmasına yardımcı olmaktadır.

Kayıt Numarası özniteliği, 86 adet alt kod ile oyuncuların oyun içi hareketlerine karşılık gelmektedir. Bu öznitelik, Blade and Soul oynarken meydana gelebilecek eylemlere göre şirket tarafından sağlanmaktadır. Tablo 3.3'te oyuncuların her kategori altında ikişer kayıt hareketine yer verilmiştir. Tam liste ise Ek-1'de açıklamalarıyla birlikte paylaşılmıştır.

Lengbear veri seti için,

Lengbear Khmer Card Games tarafından geliştirilen Kamboçya'da özel olarak mobil platformda yayınlanan bir sosyal kumarhane (casino) oyunudur. App Store ve Google Play Store üzerinden indirme seçenekleri mevcuttur. Oyunda çok oyunculu mini oyun sistemi yer almasının yanı sıra, günlük giriş yapan oyunculara ücretsiz ödül, optimize edilmiş performans ile mobilden tablete tüm cihazlarda oynatma desteği sunmaktadır.

Oyunda 9 farklı mini kumar oyunu bulunmakla birlikte oyuncular istediği türü seçip bu oyunlara çevrimiçi olarak dahil olmaktadır. Bu oyunlar Tablo 3.4'de oyun süreleriyle birlikte verilmiştir. Ayrıca ödeme yöntemi kredi kartı veya sanal kartlar aracılığıyla değil mobil operatörler üzerinden sağlanmaktadır.

Tablo 3.4: 9 farklı mini oyuna dair kodlar ve süreleri

Mini Oyun Adı	Kodu	Süresi
SLOT_100_LINE	1008	20 Saniye
BINH	9001	80 Saniye
POKDENG	9002	80 Saniye
XOCDIA	9003	20 Saniye
SHOW	9004	80 Saniye
BAUCUA	9005	20 Saniye
SIKU	9006	80 Saniye
SLOT_20_LINE	9008	20 Saniye
TIENLEN_CAM	9009	120 Saniye

Veri seti 01.03.2020 - 15.05.2020 tarihleri arasında COVID-19 pandemisi döneminde toplanmıştır. Aynı zamanda oyun F2P gelir yöntemini mikro-etkileşimler tercih ederek para kazanmayı hedeflemektedir. Alanyazın incelendiğinde mobil platformda kayıp periyodu da kabul edilen süre farklı oyun türlerine göre ortalama 4 ila 8 gün aralığında değişiklik göstermektedir. Bu veri setine dair çözülmesi hedeflenen sınıflandırma problemi, oyuncuların kayıp olarak nitelendirilmesinde 4-5-6 günlük periyotlar baz alınarak gerçekleştirilmiştir.

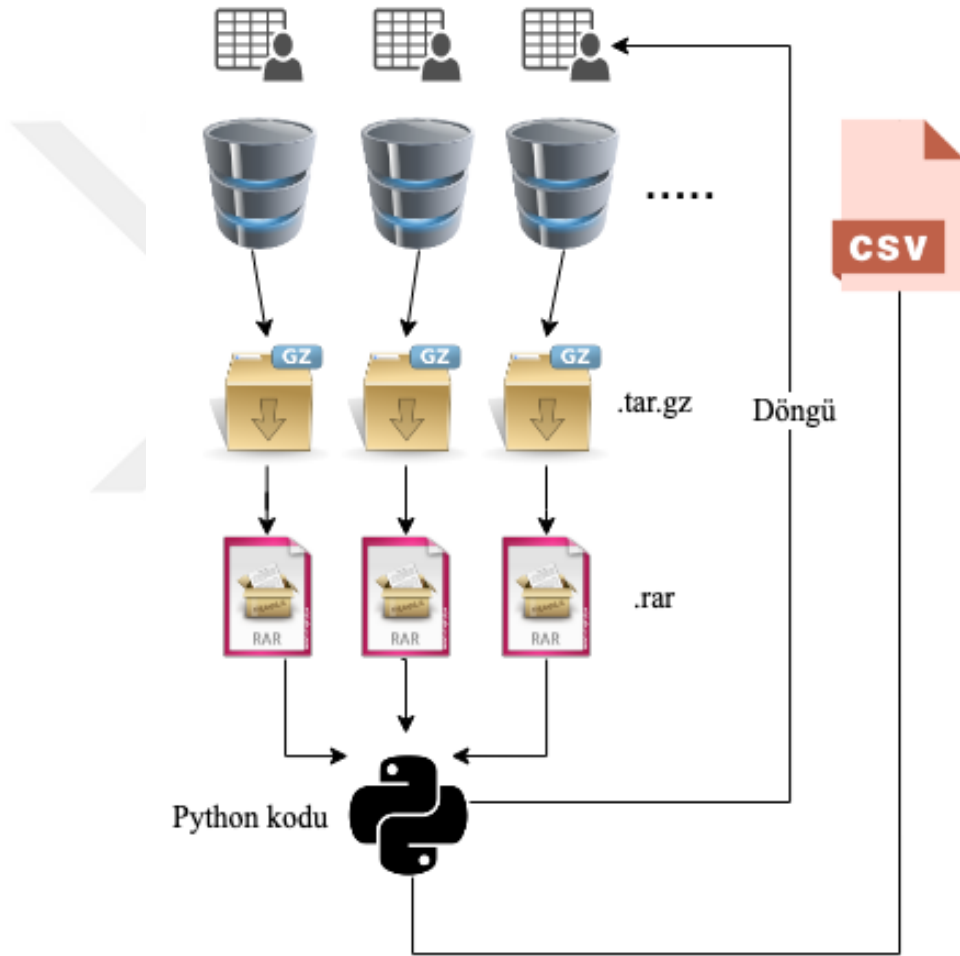
Tablo 3.5: Veri setlerinde yer alan özniteliklerin adları ve açıklamaları

Veri	Öznitelik Adı	Açıklaması	Veri Tipi
Oyuncu Kayıtları	UserID	Oyuncuya atanan benzersiz olduğunu gösteren etiket numarası	Benzersiz (Unique) - Nümerik
	Amount	Dolar bazında ödediği miktar	Nümerik
	Chips	Oyun içi alınan ödül miktarı	Nümerik
	Date	Satın alımın gerçekleştiği tarih etiketi	Tarih
	Channel	Ödeme yöntemi	Kategorik
Etkileşim Kayıtları	Sequence	Oyuncu kaydının an (sekans) etiketi	Benzersiz (Unique) - Nümerik
	UserID	Oyuncuya atanan benzersiz olduğunu gösteren etiket numarası	Benzersiz (Unique) - Nümerik
	GameID	Oyuncunun her oyun için atanan etiketidir. Her <u>GameID</u> etiketi, farklı türde oyunlarda maç başına farklı uzunluklara sahiptir.	Benzersiz (Unique) - Nümerik
	Level	Maç başına minimum 20 çipten maç başına maksimum 10 milyon çipe kadar oyun seviyesi	Nümerik
	WinNo	Her bir oturum için kazanılan maç sayısı	Nümerik
	DrawNo	Her bir oturum için berabere kalınan maç sayısı	Nümerik
	LostNo	Her bir oturum için kazanılan maç sayısı	Nümerik
	WinAmt	Her bir oturum için kazanılan toplam ödül miktarı	Nümerik
	LostAmt	Her bir oturum için kaybedilen toplam ödül miktarı	Nümerik
	Date	Oyunun tarih etiketi	Tarih
	Currency_Type1	Para birimi türü 1	Kategorik
	Currency_Type2	Para birimi türü 2	Kategorik

3.3. VERİNİN HAZIRLANMASI

Önceki bölümlerde belirtildiği üzere, verilerin modellemeye hazır hale getirilmesi için eksik verilerin tamamlanması, entegre edilmesi, temizlenmesi gibi işlemleri kapsayan veri ön işleme, CRISP-DM metodolojisinin model oluşturmadan önce ele alınması gereken önemli bir adımdır.

Blade & Soul veri seti için;



Şekil 3.2: Çok dosyalı ham verilerin tek bir .csv dosyasında işlenmesi süreci

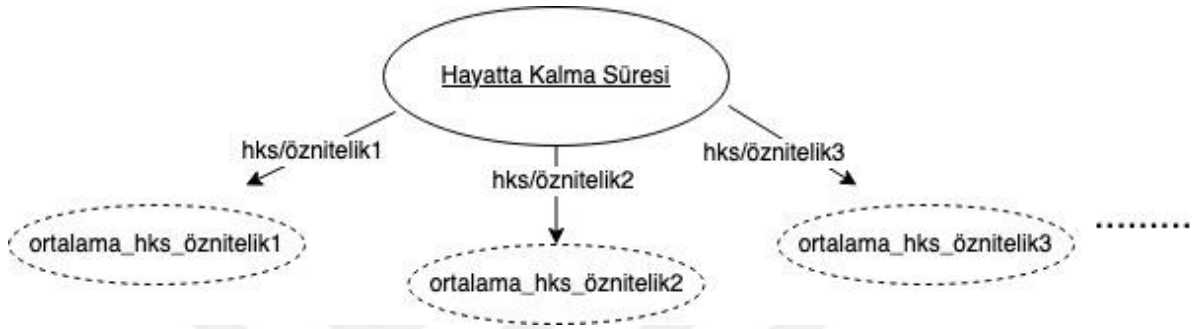
MMORPG oyun verisinde yer alan öznitelikler bir önceki bölümde verilmişti. Veriden anlamlı nümerik değerler elde edebilmek için oyuncu kayıt verisinde çok kodlu “LogID” özniteliğinde yer alan her bir öznitelik için işlemlerin saydırılması hedeflenmiştir. Bu yöntem en anlamlı sonuçları çıkarmak için oyun verilerinde sıklıkla kullanılmaktadır.

Ön işleme için adımlar aşağıda listelenmiştir:

1. Veriler yaklaşık 100 GB olduğu için çift sıkıştırma yöntemi (öncelikle .tar.gz ardından .rar) kullanılarak araştırmacılara sunulmuştur. Bu şekilde verilerin boyutu 10 GB'ye indirgenmiştir. Bu kısım veriyi sunan şirket tarafından oluşturulmuştur ancak araştırmacılar için birtakım problemleri beraberinde getirmektedir.
2. Veriler sıkıştırılmış dosyalardan tekli şekilde açılmak istendiğinde işlemcinin ve hafıza (RAM) kapasitesinin güçlü olduğu bilgisayarlarda bile “*yetersiz bellek (out-of-memory)*” hatası ile karşılaşılmıştır. Bu nedenle Şekil 3.2’de yer alan akış diyagramı uygulanarak veri ön işleme adımı .tar.gz dosyaları çıkartma işlemine tabi tutulmaksızın kod aracılığıyla ön işleme işlemi gerçekleştirilmiştir. 1000 farklı .csv dosyası ön işleme ile 10.000 satırlık tek bir CSV dosyasına dönüştürülmüştür.
3. Tamamen veya çok fazla eksik veri olan öznitelikler ön işlemede hariç tutulmuştur. Bu sütunlar Ek-1’te tablo şeklinde verilmiştir.
4. Oyun sektöründe oyuncular oyun içinde verilen bir görevi birçok yoldan gerçekleştirebilir. Bu sebepten ötürü oyuncu her işlemi kullanmak zorunda değildir. Bu durum da bazı kayıt işlemlerinin boş kalmasına sebep olmaktadır. Bu özniteliklerdeki kayıt değerleri veri setinden atılmadan 0 ile doldurulmuştur. Araştırmacı tarafından bu durum seyrek veri kavramı ile açıklanmaktadır. Yani oyuncunun eylemi yapma imkanı varken bunu yapmayı tercih etmemesinin bu duruma neden olduğu düşünülmektedir.
5. Oyun karakterinin sahip olduğu bazı kategorik özniteliklere (actor_job, actor_gender, actor_race vb.) one hot encoding uygulanmıştır.
6. Sınıflandırma probleminde verilere 0-1 arası normalizasyon, regresyon probleminde ise standardizasyon uygulanmıştır.
7. Bu kısımda öznitelik üretme ve performans değerlendirme kriterleri adlı başlıkta yer formüllere göre yeni özellikler üretilmiş ve performans değerlendirme için yeni bir ölçüte yer verilmiştir.

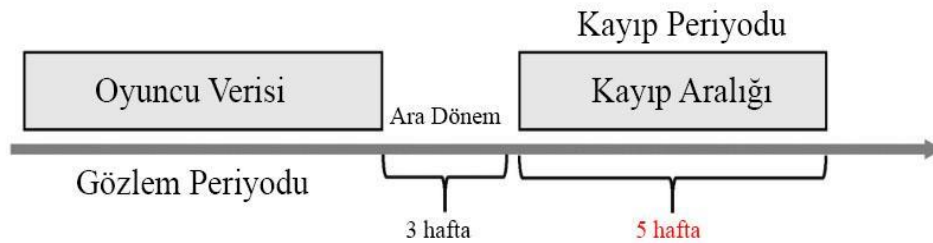
Hayatta kalma süresi değişkenini doğrudan kullanamadığımız için, veri setinde yeni öznitelikler oluşturarak bu öznitelik sınıflandırma probleminde dolaylı olarak kullanılmıştır. Hayatta kalma süresi, oyuncunun oyunda geçirdiği süreyi temsil ederken, son gün ve ilk gün, Formül 1'deki OBOGS (Oyuncu Başına Oynanan Gün Sayısı) olarak ilk giriş ve son giriş zaman damgalarını temsil etmektedir.

Oyuncunun ulaştığı toplam seviyeyi bulmak için seviye ve ustalık seviyesi değişkenleri tek bir öznitelik altında toplanmıştır. Formül 3'te gösterildiği gibi, oyuncunun ulaştığı toplam seviyeyi oyunda geçirilen süre özniteliğine bölerek seviye başına geçen süreyi oluşturulmuş ve ardından Her bir hayatta kalma süresi değişkenini kayıt numarası oluşturulan sütunlara bölerek oyuncu kayıt niteliklerinin her biri için 'ortalama_hayatta_kalma_suresi' değişkeni oluşturulmuştur. Bu işlemin detaylı açıklaması ve oluşturma basamakları Ek-1'de kod yığınları kısmında verilmiştir.



Şekil 3.3: Hayatta kalma süresi değişkeninin kayıt niteliklerine dağıtımının şeması

Çevrimiçi MMORPG oyunlarında gözlem süresi ağırlıklı olarak 4 ila 5 hafta olarak belirlenirken, mobil ve konsol oyunlarında bu süre değişkenlik göstermektedir. Bu doğrultuda, verileri sağlayan şirket tarafından hareketsizlik süresi 5 hafta olarak tanımlanmıştır. Dolayısıyla, oyuncunun kaybı gözlem (observation period) ve kayıp (churn period) periyodları olmak üzere iki farklı süreçte tanımlanmıştır. Gözlem süresi, oyuna son giriş tarihinden itibaren başlar ve eğer bir oyuncu beş haftalık hareketsizlik süresinin ardından oyuna girmezse churn etiketi atanır. Oyuncu bu beş hafta boyunca oyuna kısa bir süre giriş yapsa bile potansiyel müşteri olarak kalacaktır.



Şekil 3.4: Oyuncu kayıp tanımlanmasında gözlem ve kayıp periyodları

Şekil 3.4'te üretilen 86 yeni dağıtık özneliğin yanı sıra bir takım diğer özneliklerde veri setine eklenmiştir. Ön işlem basamağında üretilen bu özneliklerin formüllerine aşağıda yer verilmiştir.

$$NOPDP (NumberOfPlayedDaysPerPlayer) = survival\ time + (last\ log - first\ log) \quad (3.1)$$

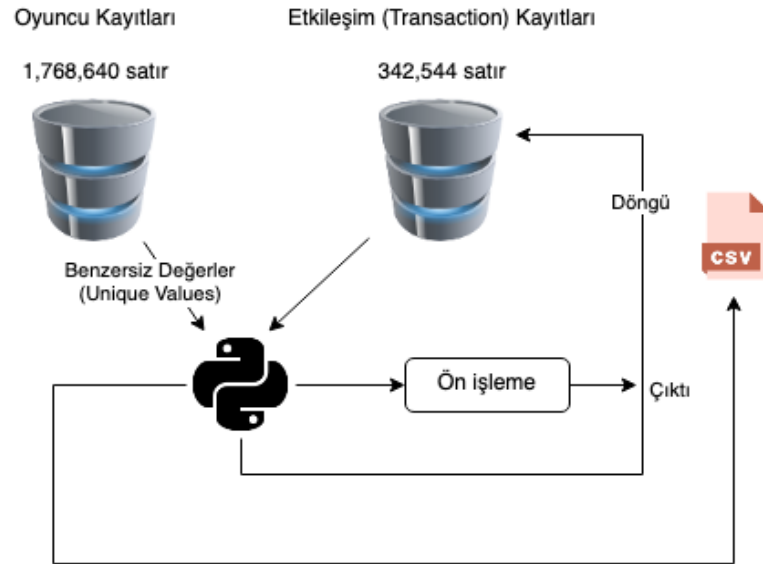
$$STPBD(SurvivalTimePerPlayerByDays) = \frac{survival\ time}{(last\ log - first\ log)} \quad (3.2)$$

$$STPAML(SurvivalTimePerPlayerMastery\ Level) = \frac{survival\ time}{actor\ mastery\ level} \quad (3.3)$$

$$STPAL(Survival\ Time\ Per\ Player\ Average\ Level) = \frac{survival\ time}{actor\ level} \quad (3.4)$$

Bu kısımda, NOPDP ile oyuncu başına toplam oyun oynama günü, STPBD ile oyuncuların gün başına düşen hayatta kalma süreleri, STPAML ve STPAL ile oyuncuların seviye başına düşen hayatta kalma sürelerinin ustalık seviyesine ve güncel seviyesine göre oluşturulmuştur.

Lengbear veri seti:



Şekil 3.5: Etkileşim ve kullanıcı veri setlerinden gelen ham verilerin işlenmesine dair akış diyagramı

Bu sınıflandırma probleminde LengBear adlı mobil oyundan elde edilen oyuncu kayıtları ve etkileşim kayıtları olmak üzere iki ayrı .csv dosyası bulunmaktadır. Oyuncu kayıtları verisi 1,768,640 kayıttan oluşurken, etkileşim verisi ise 342,544 kayıttan oluşmaktadır.

Oyuncu kayıt verisinde uygulanan ön işleme adımlarına aşağıda yer verilmiştir. Bunlar;

- UserID ve Date öznitelikleri ile her iki veri setinde oyuncuların yapmış olduğu işlemlerin tek bir veriye aktarılmasında benzersiz öznitelik olarak kullanılmıştır. Veriler yeni oluşturulan dosyaya aktarıldığında bazı değişkenler modelleme aşamasında sorun olmaması için çıkarılmıştır.
- Etkileşim kayıtlarının olduğu veri setinde GameID özniteliği one hot encoding uygulanarak veriye aktarılmıştır.
- Channel değişkeni ödeme yapılan mobil hat şirketini temsil ettiğinden ötürü verinin yapısına uygun olmadığı kanaati ile veri setinden çıkarılmıştır.
- DrawNo adlı öznitelik beraberlik sayısını temsil etmekte olup, tüm kolonlar 0 olarak veride yer almaktadır. Bir kumar oyununda oyunun berabere bitme şansı olmadığı için ve tüm kolonlar 0 olduğu için çıkarılmıştır.
- Currency_type1 ve Currency_type2 özniteliklerini şirket uzun vadede analiz yapmak için oluşturmuş ancak verinin toplama tarihlerinde bu kolonlara hiçbir değer aktarılmamıştır. Bu sebepten ötürü veriden çıkarılmıştır.
- Oyun içindeki her bir mini oyun için Tablo 3.4’de verilen oyun süreleri ve toplam oynanan oyun sayısı GameID özniteliğine göre eklenmiştir. Bu doğrultuda, aktif olarak geçirilen sürenin tespit edilmesi için bu değişkenler çarpılarak toplam aktif süre özniteliği oluşturulmuştur.
- Sınıflandırma probleminde verilere 0-1 arası normalizasyon uygulanmıştır.
- Veri setlerinde nümerik ve kategorik olan değişkenler modelleme basamağına kadar işlenmiş ve modellemeye hazır hale getirilmiştir.
- Kayıp (churn) etiketi araştırmacı tarafından 4-5-6 günlük aralıklarla fonksiyon oluşturarak etiketlenmiş ve bu etiketlere göre sonuçlar elde edilmiştir.

3.4. MODELLEME

Tez çalışması kapsamında her iki veri seti için sınıflandırma ve regresyon problemlerinin çözümünde aşağıda yer alan makine öğrenmesi algoritmaları kullanılmıştır. Bunlar;

- K-En Yakın Komşu (Sınıflandırma ve Regresyon),
- Karar Ağaçları (Sınıflandırma ve Regresyon),
- Rastgele Orman (Sınıflandırma ve Regresyon),
- Gradyan Artırma (Sınıflandırma ve Regresyon),
- Aşırı Gradyan Artırma (Sınıflandırma ve Regresyon),
- Destek Vektör Makineleri (Sınıflandırma ve Regresyon),
- Sade Bayes (Sınıflandırma),
- Lojistik Regresyon (Sınıflandırma),
- Çok Katmanlı Perseptron (Sınıflandırma ve Regresyon)

Her iki veri setinde veriler anlık olarak toplandığından ötürü eksik veri bulunmamaktadır. Şirketler tarafından veri oluşturma sürecinde tanımlanamayan eksik veriler ise 0 olarak nitelendirilmiştir.

Modellerin oluşturulma aşamasında veriler farklı dönemlerde toplandığından ötürü Blade & Soul ve LengBear veri setlerinde verilen oranlarda Dışarıda Tut (Hold-out) geçerleme yöntemi uygulanmıştır.

Blade & Soul veri setinde Naive Bayes algoritmasının aşım uyum gösterdiği ve bunun çözümüne alternatif olarak TBA (Temel Bileşenler Analizi) uygulanmış ve veri seti 75 özniteliğe düşürülmüştür. TBA analizi sonrası Naive Bayes için aşırı uyum sorunun ortadan kalkmış ve performansında artış gözlemlenmiştir.

Hiperparametre seçimi için Bölüm 2.10.3.4'te aşırı uyumun önüne geçmek için uygulanan bir prosedür olduğu belirtilmişti. Her algoritmada farklı olan hiperparametrelere her algoritmanın teorik kısmında yer verilmiştir.

3.5. MODEL PERFORMANS DEĞERLENDİRME

Model performans değerlendirme ölçütleri kısmında Bölüm 2.10.3.5'te verilen ölçütler kullanılmış ve Blade & Soul veri setinde ayrı veri setleri için daha anlaşılabilir olması açısından Test 1 ve Test 2 verilerinin harmonik ortalamasına yer verilmiştir. Bunun sebebi test verilerinin ayrı şekilde verilmesi ve modelin genellenebilirliğini gösterecek bir ölçüte ihtiyaç duyulmasıdır.

Bu kapsamda tez çalışmasında Doğruluk (Accuracy) ölçütünden ziyade ağırlıklı olarak Recall ve Precision ölçütlerine göre değerlendirme yapılmıştır. Bu noktada amaç özellikle yanlış tahmin yapılan sınıflar üzerindeki performans değerlendirmenin araştırmanın amacına uygun olmasıdır.



4. BULGULAR

Bu bölümde tez kapsamında kullanılan makine öğrenmesi algoritmalarının sınıflandırma ve regresyon problemlerinin çözümü için Blade & Soul ve LengBear veri setleri üzerinde gerçekleştirilen analizlerin bulguları paylaşılmıştır. Bu algoritmaların listesine Bölüm 3.4'te yer verilmiştir. Her veri seti için en iyi performans gösteren algoritmalar tablolastırılmıştır.

4.1. K-EN YAKIN KOMŞU ALGORİTMASINA DAİR BULGULAR

Tablo 4.1 ve Tablo 4.2'te K-En Yakın Komşu algoritmasının ikili sınıflandırma ve regresyon problemlerine dair bulgulara yer verilmiştir. En iyi performans gösteren hiperparametre seçimleri tabloların altında belirtilmiştir. Blade and Soul veri setinde dışarıda tutma (hold-out) yöntemi kullanılmasından dolayı verinin test ve eğitim olarak farklı yüzdelerde bölünmesi gerçekleştirilememiştir.

Tablo 4.1: K-En Yakın Komşu sınıflandırma probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri

Parametre	Veri Seti	Doğruluk	Hata	Kesinlik	Duyarlılık	F Ölçütü
k=3	Test 1	0.78	0.22	0.77	0.82	0.77
	Test 2	0.80	0.20	0.78	0.83	0.79
k=5	Test 1	0.77	0.23	0.77	0.82	0.76
	Test 2	0.80	0.20	0.78	0.83	0.79
k=7	Test 1	0.77	0.23	0.77	0.82	0.76
	Test 2	0.80	0.20	0.78	0.83	0.78
Performans Değ. Yöntemi*						
Hold-out	Test 1	0.78	0.22	0.77	0.82	0.77
	Test 2	0.80	0.20	0.78	0.83	0.79
10 Kat Çapraz	Test 1	0.78	0.22	0.68	0.95	0.72
Geçerleme	Test 2	0.78	0.22	0.68	0.95	0.72

*k=3 olarak alınmıştır.

Tablo 4.1'de farklı parametre değerlerine (k=3, k=5, k=7) sahip bir modelin iki farklı test veri kümesi (Test 1 ve Test 2) üzerindeki performansını çeşitli değerlendirme ve farklı metrikleri uygulanarak performans sonuçları elde edilmiştir.

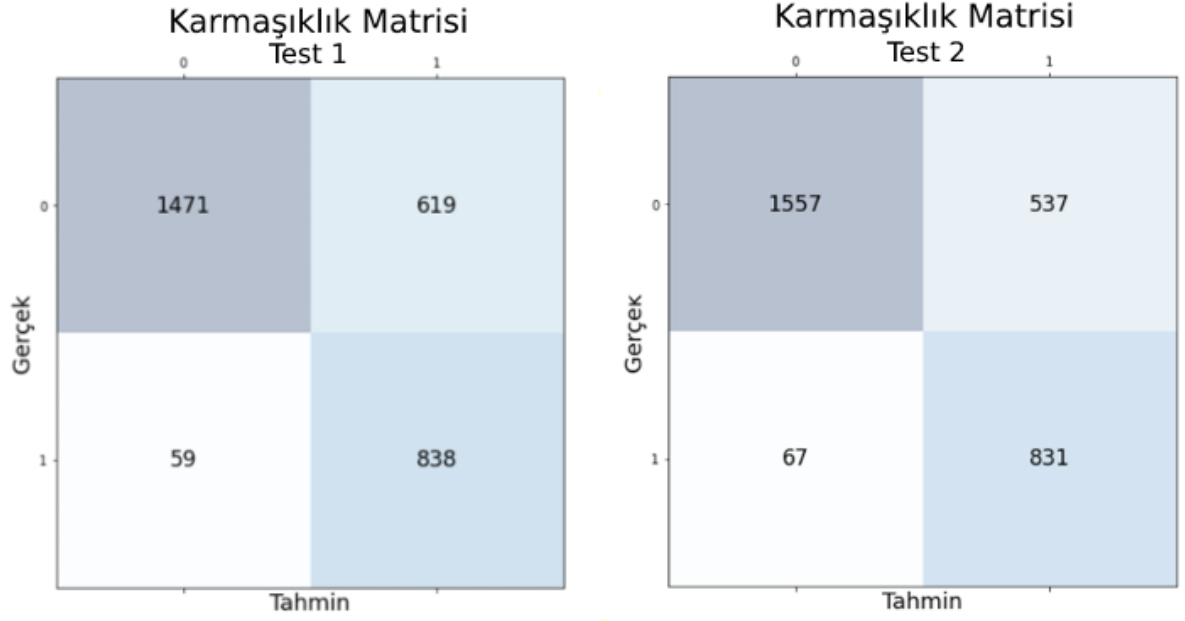
KNN algoritmasının sınıflandırma probleminde her iki test veri kümesinde de iyi performans gösterdiği söylenebilir. Kesinlik ve duyarlılık değerleri yüksek olduğundan dolayı, bu da modelin hem pozitif örnekleri hem de negatif örnekleri doğru sınıflandırmada iyi performans gösterdiği yorumu yapılabilir.

Ayrıca k parametresinin seçiminin modelin performansı üzerinde küçük bir etkisi olduğu görülmektedir. $k=3$ olduğunda, doğruluk her iki veri kümesinde de 0,78'dir. $k=5$ veya $k=7$ olarak seçildiğinde ise, doğruluk her iki veri kümesinde de 0,77 olarak bulunmuştur. Ancak, hata oranı, kesinlik, duyarlılık ve F-ölçümü k 'nın her üç değeri için birbirine yakındır.

Tablo 4.1'de görüldüğü üzere dışarıda tutma, 10 kat çapraz geçişleme sonuçlarına yer verilmiştir. Test 2 veri seti üzerinde daha iyi sonuç elde edildiği görülmektedir.

KNN algoritmasının Test 2'de Test 1'den daha iyi performans göstermesinin olası sebepleri şunlardır;

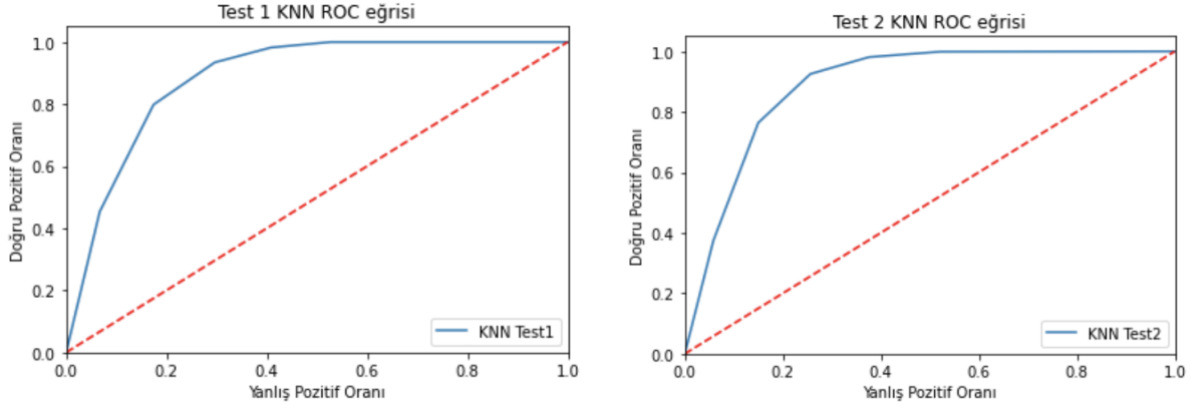
- Test 2'deki verilerin Test 1'deki verilere kıyasla daha doğrusal olarak ayrılabilir olabilir. Doğrusal ayrılabilirlik (linear separability) adı verilen, verilerin düz bir çizgi ile iki sınıfa ayrılıp ayrılamayacağını ifade eden bir veri özellik ile veriler doğrusal olarak ayrılabilirliği düşünüldüğünde, KNN algoritmasının bu konuda daha iyi performans gösterebilir.
- Bir başka olasılık ise Test 2'deki verilerin Test 1'deki verilerden daha dengeli olma durumudur. Dengeli bir veri kümesi, her sınıftaki örnek sayısının yaklaşık olarak eşit olduğu bir veri kümesidir. Veriler dengeliyse, sınıflandırıcının bir sınıfa veya diğerine karşı yanlı olma olasılığı daha düşüktür. Detaylı bilgilere Tablo 3.2'de yer verilmiştir.
- k -en yakın komşu algoritmasının bazı veri kümelerinde diğerlerine göre daha iyi performans göstermesi de mümkündür. Bunun nedeni KNN algoritmasının parametrik olmamasından kaynaklanmaktadır. Parametrik olmayan bir model, verilerin dağılımı hakkında herhangi bir varsayımda bulunmamakla birlikte çeşitli farklı dağılımlardan gelen verileri sınıflandırmada etkili olabileceği varsayımı da yapılabilir.



Şekil 4.1: Blade and Soul veri seti K-En Yakın Komşu algoritmasına dair karmaşıklık matrisi

Genel olarak karmaşıklık matrisine bakıldığında, k-en yakın komşu algoritması Test 1 1471 örneği doğru şekilde pozitif, 838 örneği doğru bir şekilde negatif olarak sınıflandırırken, gerçekte negatif olan 619 örneği pozitif olarak, gerçekte pozitif olan 59 örneği negatif olarak sınıflandırmıştır. Aynı şekilde Test 2 veri setinde ise 1557 örneği doğru şekilde pozitif, 831 örneği doğru bir şekilde negatif olarak sınıflandırırken, gerçekte negatif olan 537 örneği pozitif olarak, gerçekte pozitif olan 67 örneği negatif olarak sınıflandırmıştır.

Şekil 4.1’de karmaşıklık matrisinde yer alan durumda, modelin müşteri kayıp analizi açısından sorun oluşturabilecek potansiyele sahip olduğu söylenebilir. Çünkü, model ağırlıklı olarak kayıp olarak nitelendirilmeyen 619 oyuncuyu kayıp olarak nitelendirmiştir. Bu sebepten ötürü veri setinde eğitim verisinin artırılması bir çözüm olarak önerilebilir.



Şekil 4.2: Blade and Soul veri seti K-En Yakın Komşu algoritmasına dair ROC eğrileri

Test 1 ve Test 2 veri seti için ROC eğrilerine bakıldığında ise doğru pozitif oranında iyi seviyede bir başarı yakaladığını söylemek mümkündür.

Tablo 4.1: K-En Yakın Komşu regresyon probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri

Veri Seti	R^2	Düzeltilmiş - R^2	RMSE
Test 1	0.46	0.48	0.311
Test 2	0.37	0.39	0.331

Test 1 için R^2 değeri 0,46 olup regresyon modelinin bağımlı değişkenin yaklaşık %46'sını açıkladığını gösterirken, Test 2 için ise bu değer 0.39'dur. Test 1 için düzeltilmiş R^2 değeri 0,48 ve Test 2 için 0,41'dir. Bu modelde düşük miktarda olsa gereksiz bağımsız değişkenlerin olduğu yorumu yapılabilir.

RMSE, gözlenen değerlerin tahmin edilen değerlerden ortalama sapmasını ölçtüğü önceki bölümlerde bahsedilmişti. Test 1 ve Test 2 için elde edilen RMSE değerleri tahmin edilen değerlerin gözlenen değerlerden ortalama olarak yaklaşık 0,311-0,332 birim farklı olduğunu göstermektedir. Tahmin edilen ve gözlemlenen değerler arasında yüksek bir ortalama sapmaya işaret ettiği yorumu yapılabilir.

Genel olarak R^2 değerleri, regresyon modelinin hem Test 1 hem de Test 2 için bağımlı değişkendeki değişkenliğin orta seviyede açıklandığını göstermektedir. Bu kısımda, bağımsız değişkenlerin sayısı, veri miktarı ve verilerin dağılımı dahil olmak üzere bir modeli

etkileyebilecek birtakım unsurlardan bahsedilebilir. Bağımsız değişkenlerin sayısını ve eğitim verisi oranını arttırmak modelin daha iyi performans göstermesi açısından etkili olabilir.

4.2. KARAR AĞAÇLARI ALGORİTMASINA DAİR BULGULAR

Tablo 4.3'te farklı parametre değerlerine sahip bir modelin iki farklı test veri kümesi (Test 1 ve Test 2) üzerindeki performansını çeşitli değerlendirme metrikleri kullanarak sonuçlar paylaşılmıştır.

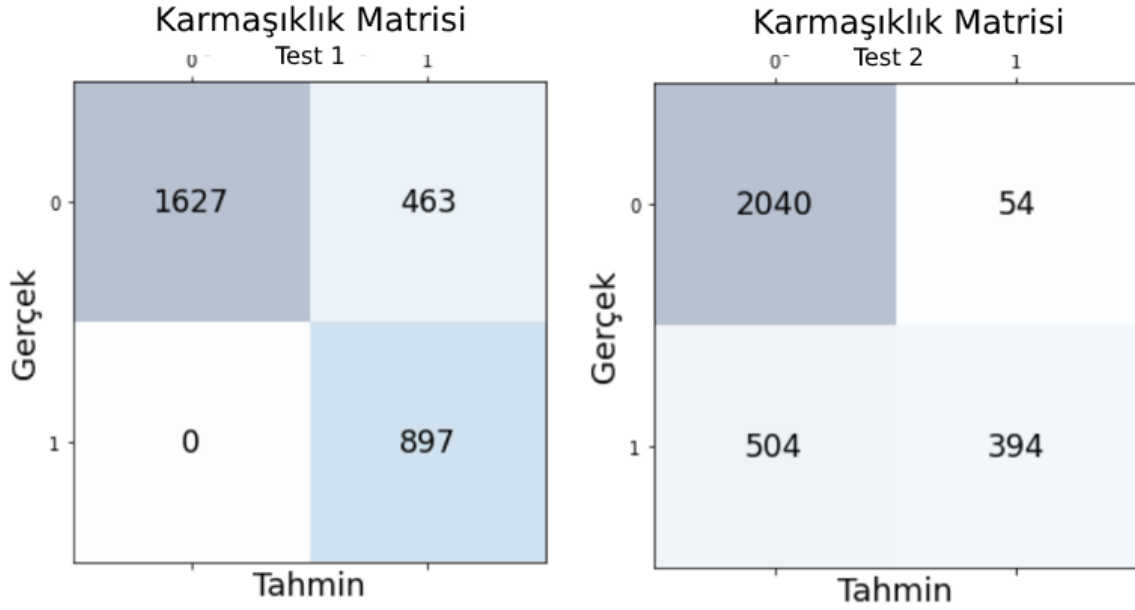
Test 1'den Test 2'ye hassasiyetteki düşüş eğitim veri setinde yer alan verinin az olmasından dolayı kaynaklanabilir. Veya verilerde, negatif örneklerle pozitif örneklerin oranı modelin pozitif örnekleri doğru bir şekilde tanımlama becerisini etkileyerek hassasiyetin azalmasına neden olmuş olabilir.

Test 1'de F-Ölçüsü 0,83'tür ve bu değer hem kesinlik hem de duyarlılığı dikkate alarak modelin performansının dengeli bir ölçümünü sağlar. Test 2'de F-Ölçüsü 0,73'e düşerek bu test setindeki duyarlılık ve kesinlik arasındaki daha düşük veri dengesi ortaya koymaktadır.

Tablo 4.2: Karar Ağaçları algoritması sınıflandırma probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri

Performans Değ. Yöntemi	Veri Seti	Doğruluk	Hata	Kesinlik	Duyarlılık	F Ölçütü
Hold-out	Test 1	0.84	0.16	0.83	0.88	0.83
	Test 2	0.81	0.19	0.84	0.71	0.73
10 Kat Çapraz Geçerleme	Test 1	0.87	0.13	0.74	0.89	0.80
	Test 2	0.83	0.17	0.70	0.80	0.73

Dışarıda tutma (hold-out) veri kümesinin 10 katlı çapraz doğrulamaya göre daha iyi performans gösterdiği söylenebilir. Bu, modelin daha az özellik kullanılarak, daha basit bir model kullanılarak veya daha fazla eğitim verisi elde edilerek geliştirilebileceğini göstermektedir.



Şekil 4.3: Blade and Soul veri seti Karar Ağaçları algoritmasına dair karmaşıklık matrisi

Genel olarak karmaşıklık matrisine bakıldığında, karar ağacı algoritması Test 1 1627 örneği doğru şekilde pozitif, 897 örneği doğru bir şekilde negatif olarak sınıflandırırken, aslında negatif olan 463 örneği pozitif olarak sınıflandırmıştır. Aslında yanlış negatif olarak sınıflandırmış örnek sayısı sıfırdır. Modelin bu konuda tahmin becerisi diğerlerine göre daha iyidir.

Aynı şekilde Test 2 veri setinde ise 2040 örneği doğru şekilde pozitif, 394 örneği doğru bir şekilde negatif olarak sınıflandırırken, aslında negatif olan 54 örneği pozitif olarak, aslında pozitif olan 504 örneği negatif olarak sınıflandırmıştır.

Karar ağacı modelinin KNN algoritmasıyla olan performansı değerlendirildiğinde karar ağaçlarının da iyi seviyede bir performans gösterdiği söylenebilir. Ancak model, müşteri kaybına neden olabilecek faktörleri ortaya çıkarma problemine çözüm olarak yaklaşıldığında ötürü çok iyi seviyede bir performans elde ettiği belirtilebilir. Genel anlamda, karar ağaçları modeli diğer algoritmalara göre daha düşük başarı seviyesine ve F-skoruna sahip olmuştur. Son olarak, farklı karar ağacı hiper parametreleri denenmiş ancak farklı bir başarı oranı elde edilememiştir.

Tablo 4.3: Karar Ağaçları regresyon probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri

Veri Seti	R^2	Düzeltilmiş - R^2	RMSE
Test 1	0.39	0.40	0.332
Test 2	0.47	0.49	0.304

Karar ağaçları regresyon modeli için iyi bir performans göstermemiştir. Düşük R-kare değerleri ve RMSE değerleri, modelin bağımlı değişkenin değerlerini doğru bir şekilde tahmin edemediğini göstermektedir. Bu kısımda modelin hayatta kalma süresini tahmin etmede etkili olmadığı düşünülmektedir.

4.3. RASTGELE ORMAN ALGORİTMASINA DAİR BULGULAR

Tablo 4.5'te Rastgele Orman algoritmasına ait farklı parametre değerlerine sahip bir modelin iki farklı test veri kümesi (Test 1 ve Test 2) üzerindeki performansı çeşitli değerlendirme metrikleri kullanılarak paylaşılmıştır.

Tablo 4.4: Rastgele Orman sınıflandırma probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri

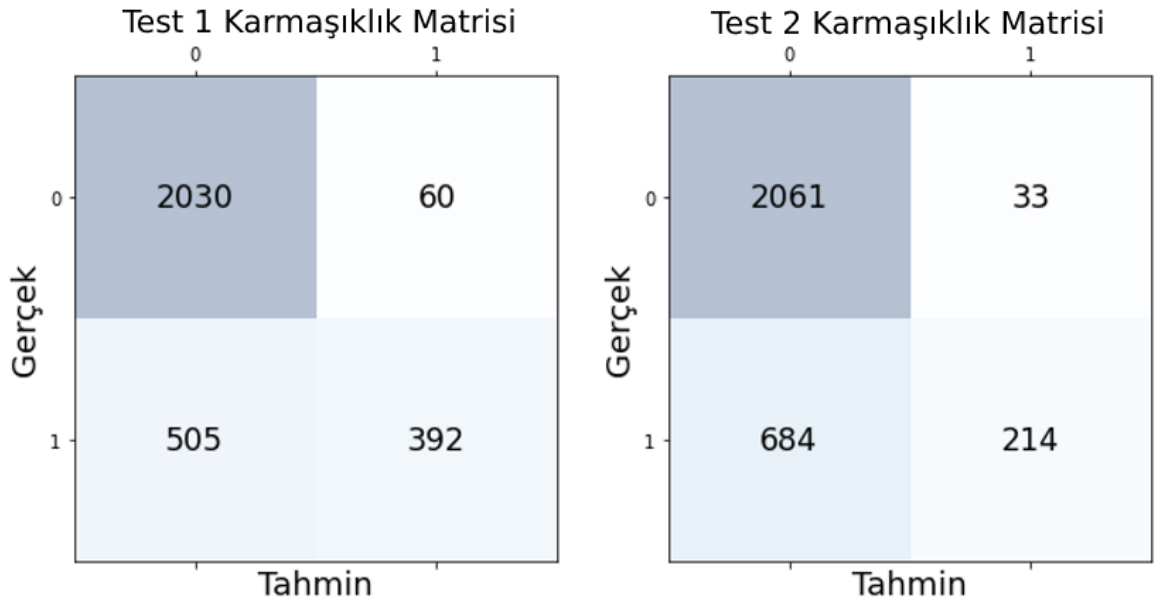
Performans Değ. Yöntemi	Veri Seti	Doğruluk	Hata	Kesinlik	Duyarlılık	F Ölçütü
Hold-out	Test 1	0.88	0.12	0.89	0.82	0.85
	Test 2	0.80	0.20	0.87	0.68	0.71
10 Kat	Test 1	0.80	0.20	0.87	0.40	0.54
Çapraz Geçerleme	Test 2	0.79	0.21	0.87	0.33	0.48

Test 1'de model, 0,88'lik doğruluk oranıyla güçlü bir performans sergilemiştir. Ayrıca, hassasiyet skoru modelin test setindeki gerçek pozitif örneklerin %82'sini etkili bir şekilde tahmin ettiğini göstermektedir. 0.85 olarak elde edilen F-ölçütü için ise, modelin Test 1'de kesinlik ve hassasiyet arasında uyumlu olduğuna dair yorum yapılabilir.

Ancak, modelin Test 2'deki performansı birden fazla ölçümde düşüş göstermektedir. Buna rağmen, kesinlik puanı 0,87 ile nispeten yüksek kalmıştır ve tahmin edilen pozitif örneklerin önemli bir kısmının gerçekten pozitif olduğunu göstermektedir. Buna karşılık, 0,68'lik

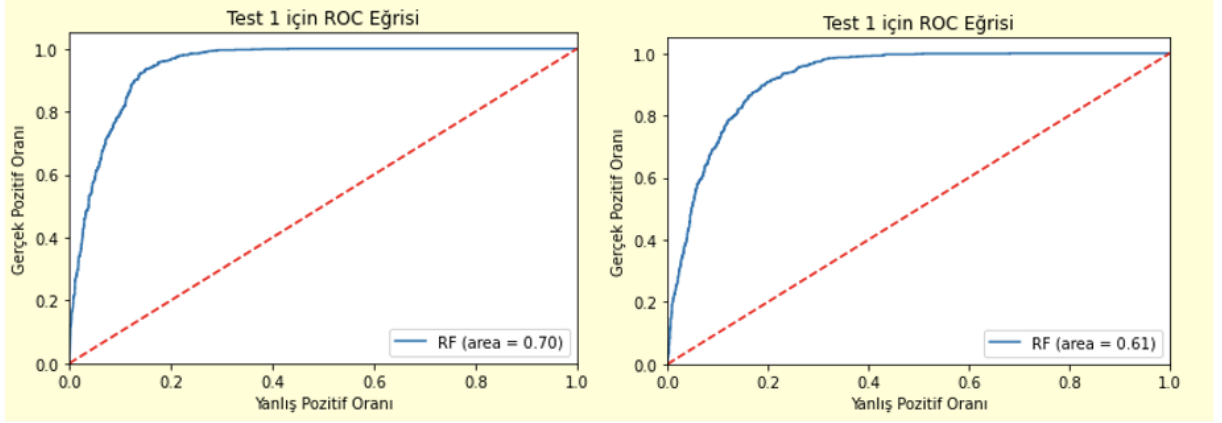
hassasiyet değeri, modelin Test 2'deki gerçek pozitif örneklerin önemli bir kısmını yakalayamadığını ortaya koymaktadır.

Bu bulgular, modelin Test 2'ye kıyasla Test 1'deki daha güçlü performansını ve genellenebilirliğini vurgulamaktadır. Test 2'deki düşük hassasiyet değeri ve F-ölçütü, modelin bu test setlerinde pozitif örnekleri doğru bir şekilde tanımlamakta zorlandığını göstermektedir. Bu performans farklılığına katkıda bulunan potansiyel faktörleri belirlemek ve modelin Test 2'deki performansını iyileştirilmesi gerekmektedir.



Şekil 4.4: Blade and Soul veri seti Rastgele Orman algoritmasına dair karmaşıklık matrisi

Bu sonuçlara dayanarak, rastgele orman modelinde 2030 adet pozitif örneğin ve 392 yanlış pozitif negatif örneğin doğru bir şekilde sınıflandırıldığı görülmektedir. Ancak, 60 adet ve 505 adet yanlış negatif bulunmaktadır. Hem Test 1 hem de Test 2 veri setleri üzerinde özellikle gerçekte kayıp olarak nitelendirilen müşterilerin modelin kayıp değil olarak nitelendirmesi sonuçların ileriye yönelik müşteri kaybına etki eden faktörleri belirleme konusunda yetersiz olduğu göstermektedir.



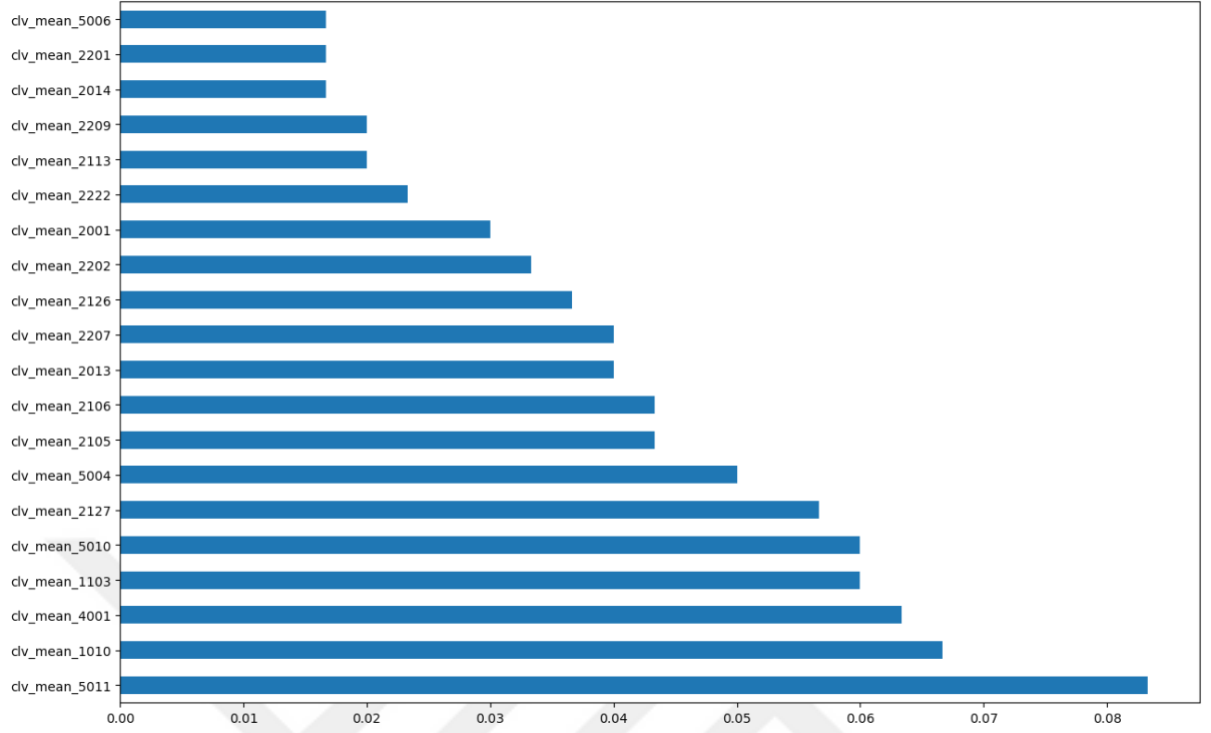
Şekil 4.5: Blade and Soul veri seti Rastgele Orman algoritmasına dair ROC eğrileri

ROC eğrisine bakıldığında ise modelde eğrinin k-en yakın komşu algoritmasında olduğu gibi tam sol yukarı çıkmadığı ve modelin nispeten iyi denilebilecek seviyede olduğu söylenebilir.

Tablo 4.5: Rastgele Orman regresyon probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri

Veri Seti	R^2	Düzeltilmiş - R^2	RMSE
Test 1	0.82	0.84	0.176
Test 2	0.73	0.74	0.216

Tablo 4.6’da performans değerleri Test 1 için daha iyi sonuçlar verdiğini ve çıktı özniteliğini açıklayıcılığının daha iyi olduğunu göstermektedir.



Şekil 4.6: Sınıflandırma probleminde Blade & Soul veri seti için Rastgele Orman algoritması öznelilik önemliliği sonuçları

Öznitelik önemi, belirli bir model için tüm girdi özellikleri için bir puan hesaplayan teknikleri ifade eder - puanlar basitçe her bir özneliliğin "önemini" temsil eder. Daha yüksek bir puan, belirli bir özneliliğin belirli bir çıktı değişkenini tahmin etmek için kullanılan model üzerinde daha büyük bir etkiye sahip olacağı anlamına gelir. Yukarıda yer alan özneliliklerin kodlarının açıklamalarına Ek-1'den ulaşılabilir.

Şekil 4.6'da oyuncunun eşya görevi almasını temsil eden 5006, eşya takaslama olan 2201 ve oyun içi gruba katılma başarı etki oranları olan 2014'ün önemliliğinin yaklaşık 0.01667 olduğu ve hedef değişkenin tahmini üzerinde nispeten düşük bir etkiye sahip olduğu görülmektedir.

2113 ve 5004 arasında yer alan kodların öznelilik önemliliği 0.02 ile 0.05 arasında değişmektedir. Bu durum, oyunculara ait hareketlerin hedef öznelilik tahmini üzerinde orta düzey etkiye sahip olduğunu belirtmektedir. Öznelilik önemliliği kodlarının açıklamalarına bakıldığında bu kısımda öneme sahip olanların ağırlıklı olarak eşya kazanma, eşya takaslama ve eşya yükseltme üzerine olduğu görülmektedir. Aynı zamanda oyuncu oyun içinde eşyasını farklı bir özel materyal birleştirerek yükseltmeye çalışmaktadır. Bu doğrultuda oyuncunun eşya yükselterek oyun içi görevleri de tamamlayarak oyunda daha etkin ve yüksek bir seviyeye

ulaşmaya çalıştığı yorumu yapılabilir. Özellikle, orta seviye öznitelik önemliliğinin eşya faktörü üzerinde yoğunlaşması da oyuncuların oyun içi eşya kullanımına daha fazla önem verdiğini göstermektedir.

Öznitelik önemliliği açısından en yüksek orana sahip olanlar ise 2127 kodu ile başlayıp, 5011 koduyla bitmektedir. Burada karşımıza çıkan özniteliklerin ağırlıklı olarak yetenek puanı artırma, oyun içinde başka bölgeye ışınlanma, oyun içi bireysel görevleri gerçekleştirmek için grup isteklerini reddetme ve günlük bonus olarak verilen hediye görevleri tamamlama üzerinde olduğu görülmektedir. Bu durum oyuncunun oyun içi harcama yapmadan oyunda günlük verilen ödüllerle oyunu oynama isteğinin yüksek olabileceği yorumu yapılabilir. Bu da müşteri kaybı açısından oldukça önemlidir. Bu doğrultuda, 5011 adlı “oyunda günlük görevleri tamamlama” öznitelikliğin en yüksek etkiye sahip olması, oyunculara verilen ödüllerin onları oyunda tutmada ne kadar büyük bir etkiye sahip olduğunu göstermektedir.

Bu kısımda günlük oyuncuya atanan görevlerde temel kurallardan biri olan basitten karmaşığa yöntemi uygulanabilir. En temel görevlerin yanında yan görevlerin de oyuncu envanterine eklenmesi oyun içinde sürekli oyuncuyu aktif olarak tutma açısından önemlidir. Öznitelik önemine dair oyun içinde sağlanabilecek öngörüler daha detaylı bir şekilde sonuç ve öneriler kısmında tartışılmıştır.

Tablo 4.6: LengBear veri seti için Rastgele Orman algoritması performans sonuçları

Yöntem	Doğruluk	Hata	Kesinlik	Duyarlılık	F Ölçütü
%53-%47 kayıp_günü=4	Aşırı Uyum gerçekleşmiştir. Eğitim başarı oranı: 0.97 Test başarı oranı: 0.48				
%57-%43 kayıp_günü=5*	0.83	0.17	0.82	0.81	0.82
%57-%43 kayıp_günü=6*	0.74	0.26	0.72	0.71	0.71

*'max_depth': 7, 'n_estimators': 75

Tablo 4.7’de Rastgele Orman algoritmasının LengBear veri seti üzerindeki performans tahmin sonuçlarına yer verilmiştir. Müşteri kayıp günü 4 gün olarak baz alındığında modelin aşırı uyum sorunuyla karşılaştığı görülmektedir. Ayrıca Izgara ve Rastgele arama yöntemlerinin

uygulanması sonucunda da aşırı uyum problemi etkisini korumaktadır. Bunun sebebi kayıp/kayıp değil olarak nitelendirilen oyuncuların dengesizliğinden, mobil oyunda oyuncuların davranışlarının kestirilmesinin zor oluşundan, verinin gerçek zamanlı akmasından ötürü her türlü oyuncu bilgisine modelde yer verilmesinden kaynaklanabilir. Ancak, veri boyutunun yeterli miktarda olmasına rağmen aşırı uyum problemi yaşanması müşteri kaybında baz alınan günün çok kısa olduğunu gösterebilir. Özellikle üzerinde çalışılan LengBear veri setinin bir sosyal kumar oyunu olması sebebiyle oyuncuların belirli aralıklarla belirli sebeplerle oyuna dönebilecekleri unutulmamalıdır.

Oyuncu kayıp süresi 5 gün olarak alındığında doğruluğun 0.83, kesinliğin 0.82, duyarlılığın ise 0.81 olduğu görülmektedir. F1-Skor ise 0.82 ile Çok Katmanlı Algılayıcı'dan çok ufak bir farklı daha iyi performans göstermiştir. Modelde kullanılan maksimum derinlik değerinin 7, tahminleyici sayısının ise 75 olarak seçilmesi verinin boyutu arttıkça işleme süresinin de artıracaklarını unutmamak gerekir. Tezin literatür kısmında bahsedildiği üzere oyuncu profilinde oyunu bırakmanın sebepleri MMO türünde daha kolay kestirilirken, mobil türünde bu tahmin oldukça zorlaşmaktadır. Çünkü, mobil cihazın temel amacı oyun oynamak değildir. Bu sebepten ötürü kayıp süresi değerlendirilirken oyuncuya biraz süre tanınması gerekir. Sonuç olarak, müşteri kayıp süresinin 5 gün seçilmesi sağlıklı bir tahmin yapabilmek ve şirketin müşteri kaybını önlemede aksiyon alması için daha etkili olacaktır.

4.4. AŞIRI GRADYAN ARTTIRMA ALGORİTMASINA DAİR BULGULAR

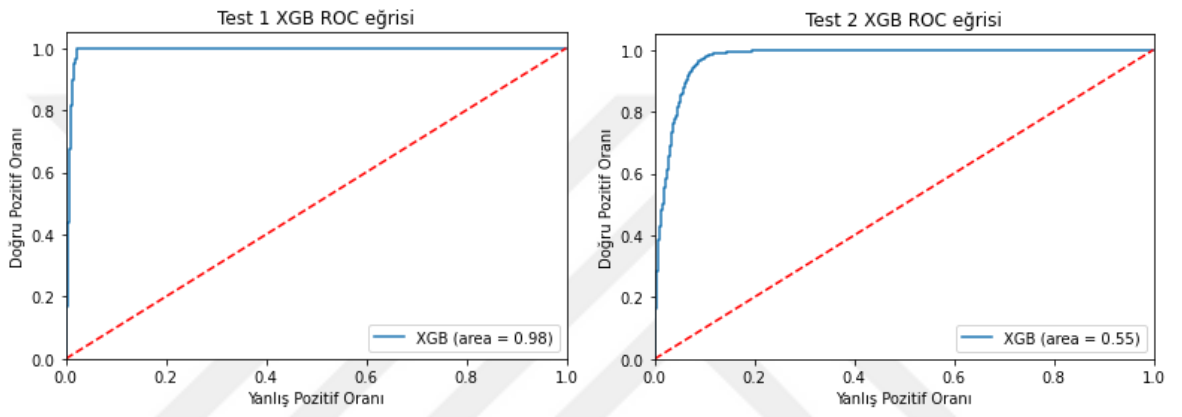
Tablo 4.8'de Aşırı Gradyan Arttırma algoritmasına ait farklı parametre değerlerine sahip bir modelin iki farklı test veri kümesi (Test 1 ve Test 2) üzerindeki performansını çeşitli değerlendirme metrikleri kullanarak sonuçlar paylaşılmıştır.

Tablo 4.7: Aşırı Gradyan Arttırma sınıflandırma probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri

Performans Değ.						
Yöntemi	Veri Seti	Doğruluk	Hata	Kesinlik	Duyarlılık	F Ölçütü
Hold-out	Test 1	0.97	0.03	0.95	0.84	0.96
	Test 2	0.93	0.07	0.95	0.84	0.92
10 Kat	Test 1	0.94	0.06	0.94	0.82	0.92
Çapraz Geçerleme	Test 2	0.90	0.10	0.92	0.82	0.91

Genel olarak model, Test 1'de Test 2'ye kıyasla daha yüksek doğruluk değeriyle, her iki veri kümesinde de çok iyi performans göstermektedir.

Aynı zamanda hem Test 1 hem de Test 2 veri setinde 0,95'lik bir kesinlik değerine ulaşarak doğru pozitif örneklerin yüksek bir yüzdesini iyi tahmin etmiştir. Duyarlılık oranında hem Test 1 hem de Test 2'de 0,84'lük hassasiyet değerine ulaşarak her iki veri kümesindeki gerçek pozitif örneklerin büyük bir kısmında da başarı gösterdiği sonucuna ulaşılabilir.



Şekil 4.7: Blade and Soul veri seti Aşırı Gradyan Artırma algoritmasına dair ROC eğrileri

Tablo 4.8: Aşırı Gradyan Artırma sınıflandırma probleminde Blade & Soul veri seti için ızgara ve rastgele arama sonuçları

Arama Yöntemi	Veri Seti	Doğruluk	Hata	Kesinlik	Duyarlılık	F Ölçütü
Izgara Arama*	Test 1	0.95	0.05	0.94	0.82	0.94
	Test 2	0.92	0.08	0.93	0.82	0.91
Rastgele Arama**	Test 1	0.94	0.06	0.91	0.83	0.92
	Test 2	0.91	0.09	0.90	0.82	0.91

*max_depth=6, n_estimators=45,cv=5

** max_depth=5, n_estimators=35,cv=5

Tablo 4.9'da ızgara arama yönteminin, rastgele arama yönteminden biraz daha yüksek doğruluğa sahip olduğu görülmektedir. Ancak rastgele arama yöntemi, hiperparametre değerlerinin tüm olası kombinasyonlarını denemek zorunda olmadığından daha verimlidir.

Rastgele orman algoritmasında maksimum derinliğin ve tahminci sayısının artırılması, çeşitli nedenlerle modelin performansını potansiyel olarak artırabilir:

- Maksimum derinliğin artırılması, rastgele ormandaki karar ağaçlarının daha derin büyümesine ve verilerdeki daha karmaşık örüntüleri yakalamasına olanak tanır. Bu, modelin karmaşık ilişkileri daha iyi anlamasına ve daha doğru tahminler yapmasına yardımcı olabilir.
- Değişken sayısını arttırarak, rastgele orman algoritmasında karar ağaçlarını oluştururken dikkate alması için daha büyük bir potansiyel tahmin ediciler havuzu sağlarsınız. Bu, topluluktaki ağaçların çeşitliliğini arttırarak daha iyi genel tahminlere yol açabilir.
- Ağaçların maksimum derinliğinin arttırılması modeldeki yanlılığı azaltabilir. Sığ ağaçlar verilere yetersiz uyum sağlayarak yüksek yanlılığa ve düşük tahmin performansına yol açabilir. Daha derin ağaçlar daha karmaşık ilişkileri yakalayıp yanlılığı azaltabilir ve potansiyel olarak model performansını artırabilir.

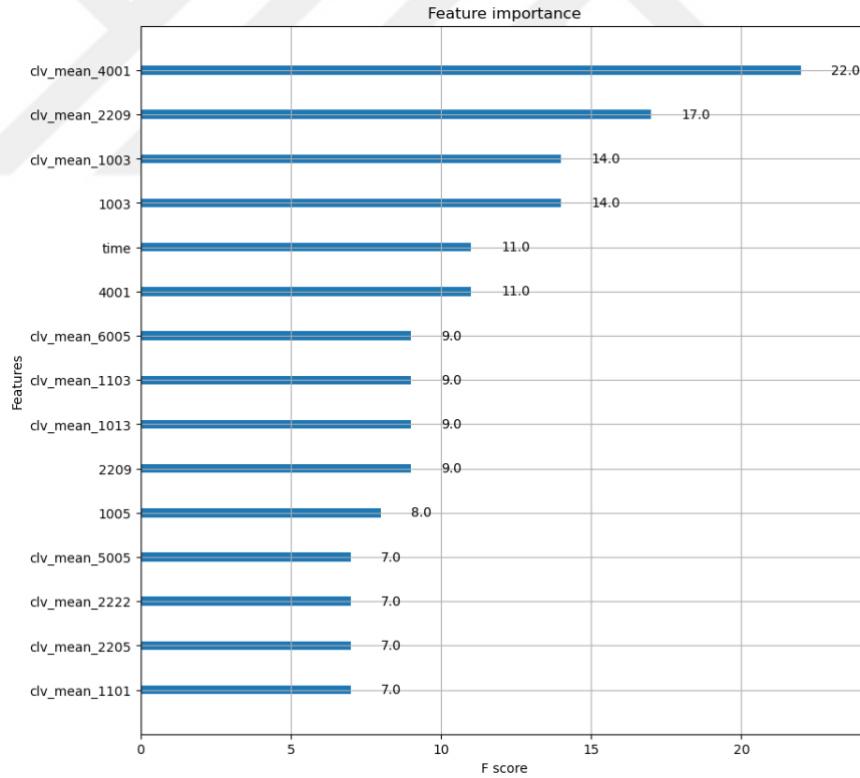
Ancak, maksimum derinliğin ve tahminci sayısının artırılmasının ödünleşim (trade-off) beraberinde getirebileceğini unutmamak önemlidir. Aşırı uyum ve işlem süresinin artması gibi etmenleri ortaya çıkarabilir. Sonuç olarak, rastgele orman algoritmasındaki maksimum derinliği ve tahminci sayısını arttırmak, daha karmaşık örüntüleri yakalayıp, özellik çeşitliliğini artırarak, genellemeyi iyileştirerek ve yanlılığı azaltarak performansı potansiyel olarak artırabilir.

Belirli bir problem için en iyi arama yöntemi, belirlenen probleme ve mevcut kaynaklara bağlıdır. Eğer model performans kriterimiz doğruluk (accuracy) ise ızgara arama yöntemi en iyi seçenek olarak değerlendirilebilir. Ancak, f-skor üzerinden bir değerlendirme yapılacaksa, rastgele arama yöntemi daha iyi bir seçim olabilir. Bu doğrultuda, Tablo 4.9’da ızgara arama ile rastgele arama arasındaki farkın çok küçük olduğu görülmektedir. Bu sebepten ötürü zaman ve performans açısından modelde rastgele arama kullanmak daha mantıklıdır. Ancak bu durum verinin çok olması ve değişken sayısının artması durumunda ızgara aramayı daha avantajlı hale getirecektir.

Tablo 4.9: Aşırı Gradyan Artırma regresyon probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri

Veri Seti	R^2	Düzeltilmiş - R^2	RMSE
Test 1	0.88	0.85	0.144
Test 2	0.79	0.77	0.189

Genel olarak R^2 değerleri, regresyon modelinin hem Test 1 hem de Test 2 için bağımlı değişkendeki değişkenliğin yüksek bir oranını açıkladığını göstermektedir. Düzeltilmiş R^2 değerleri, modeldeki tahminleyici sayısı dikkate alındığında biraz daha yüksektir. RMSE değerleri, modelin tahminlerinin gözlenen değerlerden nispeten düşük bir ortalama sapmaya sahip olduğunu ve Test 1'de daha iyi performans gösterdiğini göstermektedir. Bu sonuçlar, regresyon modelinin verilere iyi uyum sağladığını ve hem Test 1 hem de Test 2 için bağımlı değişkeni tahmin etmede iyi performans gösterdiğini ortaya koymaktadır.



Şekil 4.8: Sınıflandırma probleminde Blade & Soul veri seti için Aşırı Gradyan Artırma algoritması Öznitelik Önemliliği Sonuçları

Şekil 4.8’de Aşırı Gradyan Artırma algoritmasına dair öznitelik önemliliği sonuçları paylaşılmıştır. Bulgu değerlendirmede, “F-Skor” model performans değerlendirme

ölçütlerinden farklı olup öznitelik önemliliği açısından önemlidir. F-skor her bir özelliğin diğerlerinden bağımsız olarak ayırt edici gücünü ortaya koymaktadır. Her özellik için bir puan hesaplanır ve bu puanlara göre çıkarım oluşturmaya çalışılır. Özellikle, her bir özelliğin kaç kez bölündüğünü basitçe özetleyen bir metriktir (xgboost, 2023). Yukarıda yer alan özniteliklerin açıklamalarının uzun olması sebebiyle kodlar halinde verilmiş olup ve detaylı açıklamalarına Ek-1’den ulaşılabilir.

Genel itibariyle öznitelik önemliliğine bakıldığında, önem etkisi bakımından Rastgele Orman algoritmasında elde edilen özniteliklerle benzer olduğu görülmektedir. 4001 koduyla oyuncunun yetenek kazandığında oyun içinde ilerleme göstermesi anlaşılabilir. En yüksek puanın bu özelliğe sahip olduğu görülmektedir (F-Skor=22). Bir oyunda oyuncunun en temel hedeflerinden biri de oyun içinde yönettiği karakterin seviyesini yükseltmesidir. Çünkü, seviye yükselmeden oyun içinde topladığı eşyaları kullanamaz ve diğer yetenek setlerini aktif edemez. Bu sebepten ötürü oyuncunun yetenek kazandığını gösteren özniteliğin önemlilik açısından ön plana çıkması müşteri kaybında oldukça önemlidir. Bu doğrultuda, şirket oyun içinde oyuncu kaybını en aza indirmek için seviye atlamasını kolaylaştırabilir veya oyun için bonuslar ekleyerek oyunun sistemine entegre edebilir. Bu şekilde oyuncunun oyunda tutulduğu süre de artış gösterecektir.

Hemen ardından ise 2209 koduyla NPC’den eşya alması gelmektedir. F-Skor 11 ile 17 arasında değişmektedir. Şekil 4.8’e göre, oyuncunun oyun içinde en çok vakit harcadığı olayların en başında yetenek puanı kazanmanın ve oyun eşyası elde etmenin olduğu yorumu yapılabilir. Sonuçların devamı incelendiğinde ise 1013 (oyuncuya ait karakterin seviye atlaması), 1005 (oyuncunun haritaya giriş yapması), 2205 (oyuncunun mağazadan eşya satın alması) diğerlerine nazaran düşük etkiye sahip olduğu söylenebilir. Ancak, müşteri kayıp analizi kapsamında değerlendirilmesi faydalı olacaktır. Çünkü, düşük etki oranına sahip değişkenlerin oyun üzerindeki etkilerini kestirmek uygulama yapmadan pek mümkün değildir. Bu doğrultuda, 2205 kodu incelendiğinde oyuncu için mağazadan alınan eşyaların fiyatları düşürülebilir veya sık kullanılan eşyalar ile takas edilebilmesine imkan verilebilir. Ancak bu durum oyuncunun oyun içinde hedeflerine daha kolay ulaşmasına ve buna hızlı gelişim göstermesine sebep olacaktır. Bu noktada, MOBA türünde oyunlarda olduğu gibi bu tarz MMORPG oyun türlerinde de aylık eşya ve oyun içi güncelleme gerçekleştirerek çözülmesi önerilebilir.

Aşırı Gradyan Arttırma algoritmasının öznitelik önemliliği sonuçlarına göre, önemli olan özniteliklerin ağırlıklı olarak oyunda eşya ve geçirilen süreye göre şekillendiği söylenebilir. Müşteri kaybında kullanılabilecek olası önerilere geniş kapsamlı olarak sonuç bölümünde yer verilmiştir.

4.5. DESTEK VEKTÖR MAKİNELERİ ALGORİTMASINA DAİR BULGULAR

Tablo 4.11’de Destek Vektör Makineleri algoritmasına ait farklı parametre değerlerine sahip bir modelin iki farklı test veri kümesi (Test 1 ve Test 2) üzerindeki performansını çeşitli değerlendirme metrikleri kullanarak sonuçlar paylaşılmıştır.

Tablo 4.10: Destek Vektör Makineleri sınıflandırma probleminde Blade & Soul veri seti farklı çekirdekler için performans değerlendirme ölçü değerleri

Çekirdek	Veri Seti	Doğruluk	Hata	Kesinlik	Duyarlılık	F Ölçütü
linear	Test 1	0.95	0.05	0.94	0.95	0.94
	Test 2	0.88	0.12	0.90	0.81	0.84
rbf	Test 1	0.95	0.05	0.94	0.97	0.95
	Test 2	0.92	0.08	0.92	0.88	0.90
sigmoid	Test 1	0.94	0.06	0.93	0.96	0.94
	Test 2	0.92	0.08	0.93	0.91	0.92
poly	Test 1	0.81	0.19	0.76	0.69	0.71
	Test 2	0.79	0.21	0.78	0.69	0.71

RBF çekirdek hem Test 1 hem de Test 2’de en yüksek doğruluğa sahiptir. Diğer çekirdeklerle arasında tahmin etme becerisinde düşük miktar farklılıklar oluşmuştur.

RBF çekirdeğin diğerlerine göre daha iyi performans göstermesinin olası sebepleri;

- RBF çekirdeği, veri üzerinde doğrusal olmayan ilişkileri sınıflandırma konusunda daha iyidir. Bağımsız öznitelikleri daha yüksek boyutlu bir özellik uzayına eşleyerek (mapping) karmaşık desenleri ortaya koyabilir. Bu, DVM’nin doğrusal olmayan karar sınırlarını daha etkili bir şekilde modellemesine olanak tanır ve doğrusal veya diğer daha basit çekirdek işlevlerine kıyasla potansiyel olarak daha iyi performans sağlamaktadır.
- RBF çekirdeğinin önemli parametrelerinden biri gamma (γ) parametresidir. Bu parametre her bir eğitim örneğinin karar sınırları üzerindeki etkisini kontrol eder. Gama

parametresini ayarlayarak karar sınırının esnekliğini kontrol edilebilir. Tablo 4.14'te farklı gamma ve C değerlerine ait hiperparametrelerin sonuçlarına yer verilmiştir. Şekil 4.9'da ise C ve gamma değerlerinin farklı denemelerinden elde edilen sonuçların artış ve azalış değerleri paylaşılmıştır.

- RBF çekirdeği de dahil olmak üzere bir çekirdek işlevinin performansı veri kümesine ve eldeki soruna bağlı olarak değişebilir. Bu test verileri üzerinde RBF çekirdeğin daha iyi çalıştığını ancak doğrusal (linear) çekirdeği de aynı seviyede çalıştığını söylemek mümkündür. Çünkü, oyuncuların oyunu oynayıp oynamadığını belirlemede oynama süresi değişkeni ön plana çıkar. Bu da oyuncuların oyunu yükleyip kısa süre oynayıp oyunu silmesine ya da oyunu severek daha fazla vakit geçirmesiyle sonuçlanmaktadır. Bu da doğrusal ayrılabilirlik açısından sınıflandırma probleminde oldukça etkilidir. Bu doğrultuda, mevcut veri setleri için en iyi performansı gösteren modeli bulmak farklı çekirdek fonksiyonları ve hiperparametre seçimlerine dair sonuçlara Tablo 4.13'te yer verilmiştir.

Tablo 4.11: Destek Vektör Makineleri sınıflandırma probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri

Performans Değ.						
Yöntemi-rbf	Veri Seti	Doğruluk	Hata	Kesinlik	Duyarlılık	F Ölçütü
Hold-out	Test 1	0.95	0.05	0.94	0.97	0.95
	Test 2	0.92	0.08	0.92	0.88	0.90
10 Kat	Test 1	0.96	0.04	0.88	0.99	0.93
Çapraz Geçerleme	Test 2	0.95	0.05	0.87	0.97	0.92

Tablo 4.12'te elde edilen sonuçlar şu şekilde yorumlanabilir;

- Doğrusal çekirdek ve RBF çekirdek sınıflandırma görevi için iyi bir seçimdir.
- Sigmoid çekirdek sınıflandırma görevi için iyi bir seçimdir olmakla birlikte doğrusal çekirdek veya RBF çekirdeği kadar iyi değildir.
- Çok boyutlu (poly) çekirdek sınıflandırma görevi için iyi bir seçim değildir.

Sigmoid ve çok boyutlu çekirdekler içinse verilerdeki temel örüntülerin doğrusal olmaya daha yakın olduğunda veya RBF çekirdeği tarafından iyi bir şekilde yakalanabilecek doğrusal olmayan bir modele sahip olduğu sonucuna ulaşılabilir. Veriler doğrusal bir ayrılabilirliğe

sahipse veya doğrusal olarak ayrılabilir bir uzaya etkili bir şekilde dönüştürülebiliyorsa, doğrusal çekirdeğin iyi performans göstermesi muhtemeldir. Öte yandan, sigmoid ve çok boyutlu (poly) çekirdekler bu örüntüleri etkili bir şekilde yakalayamayabilir ve bu da daha düşük performansla sonuçlandığı anlamına gelir.

Tablo 4.12: Destek Vektör Makineleri regresyon probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri

Veri Seti	R^2	Düzeltilmiş - R^2	RMSE
Test 1	0.806	0.797	0.187
Test 2	0.784	0.774	0.194

RMSE ise düşük olmakla birlikte daha iyi tahmin doğruluğuna ve modelin tahminlerinin ortalama olarak gerçek değerlerden yaklaşık 0,187 birim saptığını göstermektedir. Test 1'e benzer şekilde, modelin tahminlerinin ortalama olarak gerçek değerlerden yaklaşık 0,194 birim saptığını göstermektedir.

Akademik bağlamda sonuçlar, R^2 değerlerinin de gösterdiği gibi, hem Test 1 hem de Test 2 modellerinin bağımsız ve bağımlı değişkenler arasında nispeten güçlü ilişkiler gösterdiğini ortaya koymaktadır. Düzeltilmiş R^2 değerleri, yordayıcıların sayısını hesaba katarak modellerin açıklayıcı gücünün daha güvenilir bir ölçüsünü sağlamaktadır. Ayrıca, RMSE değerleri ortalama tahmin hatasını gösterir ve daha düşük değerler daha iyi model doğruluğuna işaret eder.

Tablo 4.13: Destek Vektör Makineleri sınıflandırma probleminde Blade & Soul veri setine dair ızgara ve rastgele arama yöntemlerine göre performans değerlendirme ölçü değerleri

Arama Yöntemi	Veri Seti	Doğruluk	Hata	Kesinlik	Duyarlılık	F Ölçütü
Izgara Arama*	Test 1	0.95	0.05	0.94	0.97	0.95
	Test 2	0.92	0.08	0.92	0.88	0.90
Rastgele Arama**	Test 1	0.95	0.05	0.94	0.97	0.95
	Test 2	0.92	0.08	0.92	0.88	0.90

*C=100, gamma=0.01, kernel='rbf'

** C=100, gamma=0.01, kernel='rbf'

SVM algoritmasında, C parametresi düzenleme parametresini veya maliyet parametresini temsil eder. Düşük bir eğitim hatası elde etmek ile daha geniş bir marj veya hiper düzlem ayrımı sağlamak arasındaki dengeyi kontrol eder.

C değerinin artırılması, eğitim verilerindeki yanlış sınıflandırmalara daha yüksek bir ceza uygular. Sonuç olarak, DVM daha az yanlış sınıflandırmaya izin vererek eğitim hatasını en aza indirmeyi amaçlar. Bu genellikle sınıflar arasında daha dar bir marj veya hiper düzlem ayrımına yol açar. Model, eğitim verilerine yakından uymaya daha fazla odaklanır ve potansiyel olarak aşırı uyuma yol açar.

Daha yüksek bir C değeri SVM modelinin karmaşıklığını artırır. Algoritma verilere daha hassas bir şekilde uymaya çalışır, bu da potansiyel olarak daha fazla sayıda destek vektörüne veya daha karmaşık bir karar sınırına yol açar. Bu artan karmaşıklık, daha uzun eğitim sürelerine ve daha yüksek hesaplama gereksinimlerine neden olabilir.

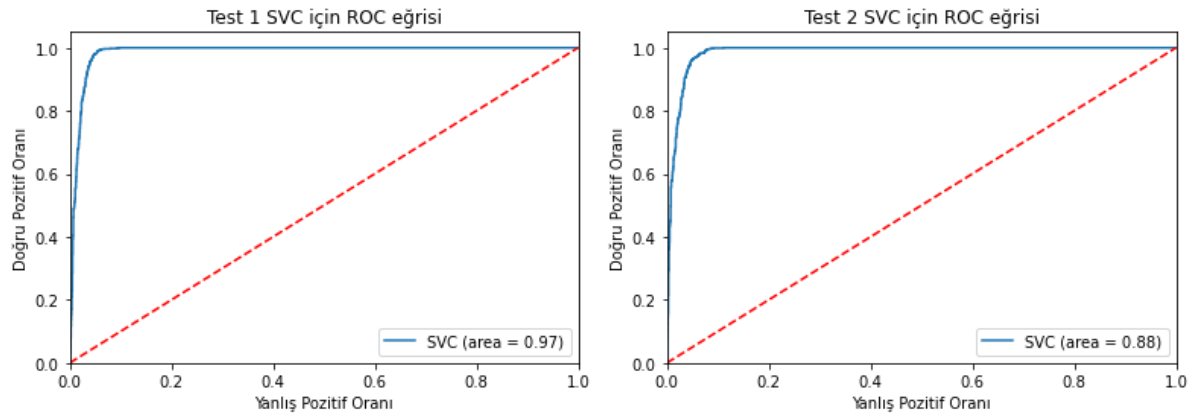
C değeri büyük olduğunda, DVM eğitim setindeki gürültülü veya yanlış etiketlenmiş veri noktalarına karşı daha hassas hale gelebilir. Verilerin altında yatan örüntüleri temsil etmeseler bile, bu aykırı değerlere uymaya öncelik verebilir. Sonuç olarak, modelin gürültüye karşı dayanıklılığı azalabilir ve yeni, görülmemiş veriler üzerinde iyi performans göstermeyebilir. C değeri çok büyükse SVM genelleme yerine eğitim hatasına öncelik verebilir. Model, eğitim verilerine çok yakından uymaya odaklandığı için görülmeyen verilere iyi genelleme yapmakta zorlanabilir. Bu da test veya doğrulama verilerinde düşük performansa neden olabilir.

C değeri	Gamma	Skor
0,10	1,00	0,911
0,10	0,10	0,91
0,10	0,01	0,91
0,10	0,00	0,91
1,00	1,00	0,951
1,00	0,10	0,942
1,00	0,01	0,942
1,00	0,00	0,942
10,00	1,00	0,952
10,00	0,10	0,95
10,00	0,01	0,951
10,00	0,00	0,95
100,00	1,00	0,955
100,00	0,10	0,95
100,00	0,01	0,948
100,00	0,00	0,948

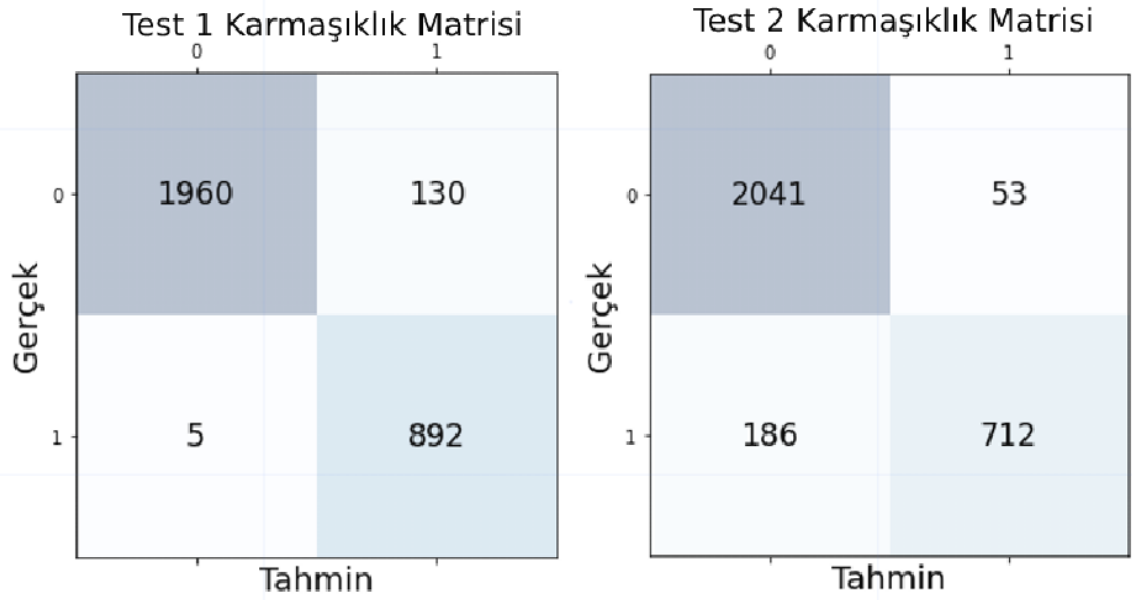
Şekil 4.9: Blade & Soul veri seti için DVM algoritmasında C ve Gamma'ya göre F-1 Skor sonuçları

Daha yüksek gama değerleri genellikle karmaşık veya doğrusal olmayan desenlere sahip veri kümeleri için seçilirken, daha yumuşak veya global desenlere sahip veri kümeleri için daha düşük gama değerleri tercih edilir.

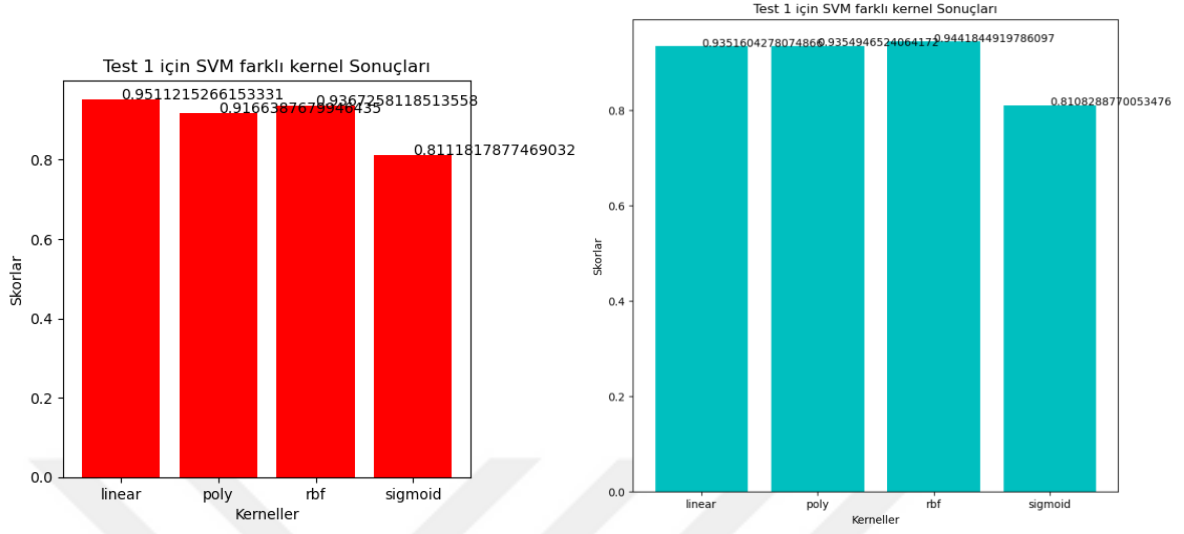
İyi bir performans elde etmek ve aşırı uyumu önlemek için C parametresi için uygun bir değer bulmak önemlidir. Bu değer genellikle belirli veri kümesine ve eldeki soruna bağlıdır. Çapraz doğrulama gibi teknikler, modelin farklı parametre değerleri için doğrulama verileri üzerindeki performansını değerlendirerek optimum C değerinin belirlenmesine yardımcı olabilir. Bu sebepten ötürü ızgara arama ve rastgele arama yöntemleri uygulanarak modelin en iyi performans gösterdiği parametrelere ulaşılmaya çalışılmıştır. Her iki arama yöntemine göre elde edilen parametreler ise Şekil 4.9'da verilmiştir.



Şekil 4.10: Blade and Soul veri seti Destek Vektör Makineleri algoritmasına dair ROC eğrileri



Şekil 4.11: Blade and Soul veri seti Destek Vektör Makineleri algoritmasına dair karmaşıklık matrisi



Şekil 4.12: Blade and Soul veri seti Destek Vektör Makineleri Algoritmasına Dair Kernel F1-skorları

Test 1 verisi doğrusal olarak daha iyi ayrılabilen bir yapıya sahiptir. 0.95 başarımlarına sahip olmakla birlikte Test 2 veri seti içinse RBF kernelin daha iyi performans sağladığı görülmektedir. Ancak aralarındaki hata oranı çok düşük olduğundan dolayı her iki veri setinde de doğrusal kernelin başarılı çalıştığı ve öngörü elde konusunda yardımcı olabileceği söylenebilir. Doğrusal çekirdeğin hem Test 1 hem de Test 2'de başarılı olması, veri setlerinin doğrusal olarak ayrılabilir olduğunu veya veri noktalarını doğru bir şekilde sınıflandırmak için doğrusal bir karar sınırının yeterli olduğunu gösterir.

4.6. SADE BAYES ALGORİTMASINA DAİR BULGULAR

Sade Bayes Algoritması'na dair bulgulara ve performans değerlendirme ölçülerinin aldığı en yüksek değerler 10 kat çapraz geçirelemeye göre sonuçlar Tablo 4.15'te yer verilmiştir.

Tablo 4.14: Sade Bayes sınıflandırma probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri

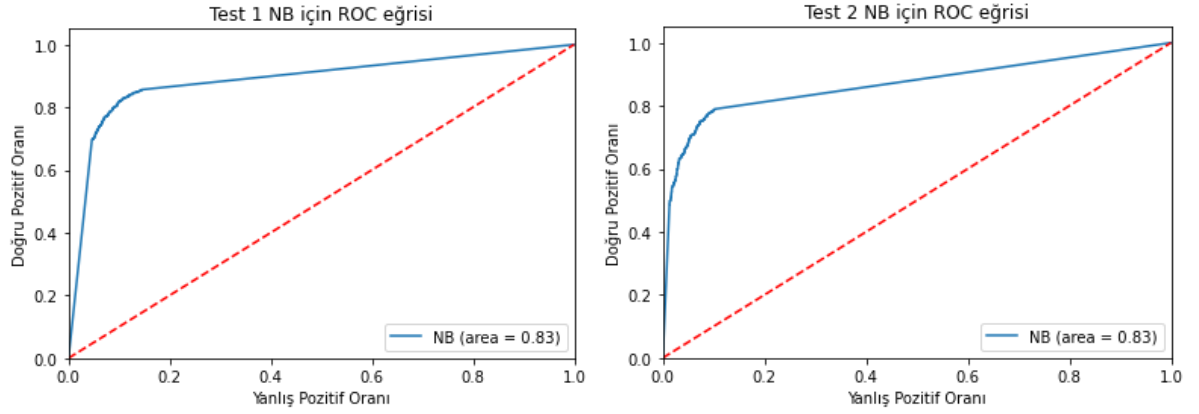
Yöntem	Veri Seti	Doğruluk	Hata	Kesinlik	Duyarlılık	F Ölçütü
Hold-out	Test 1	0.88	0.12	0.87	0.83	0.84
	Test 2	0.85	0.15	0.88	0.75	0.79

10 Kat Çapraz Geçerleme	Test 1	0.93	0.07	0.96	0.96	0.89
	Test 2	0.91	0.09	0.80	0.92	0.85

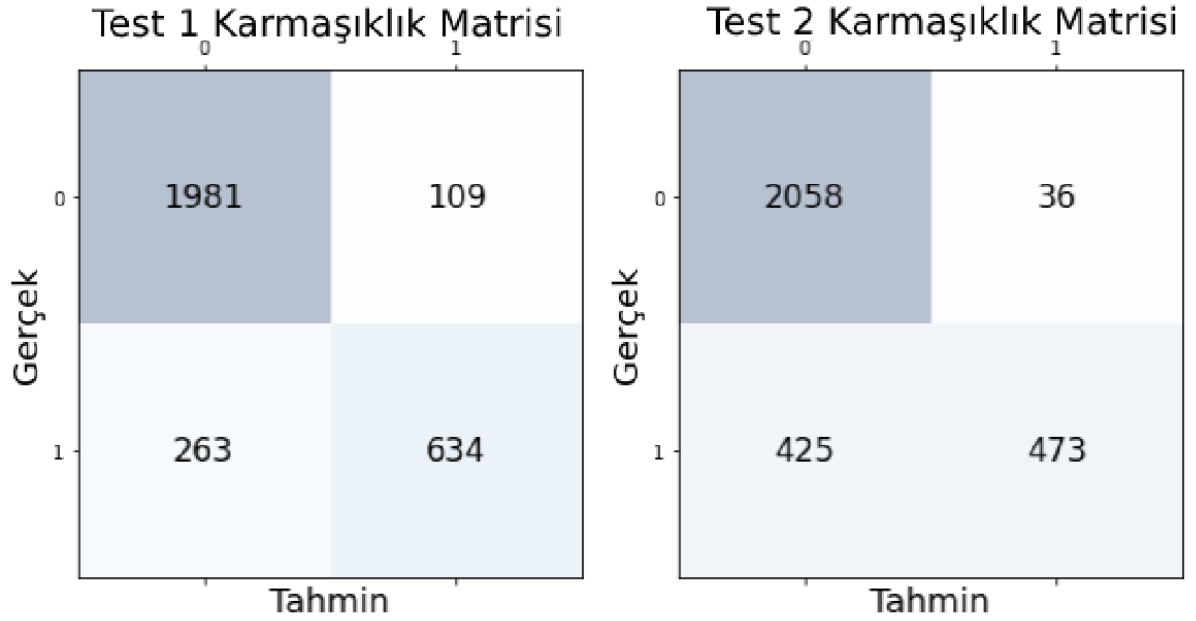
Test 1'de, önerilen yöntem 0,88'lik bir doğruluğa ulaşarak bu veri kümesindeki örneklerin yaklaşık %88'ini doğru bir şekilde sınıflandırdığını göstermektedir. Hata oranının 0,12 gibi düşük bir seviyede olması da yöntemin doğru sınıflandırma konusundaki etkinliğini teyit etmektedir. 0,87'lik kesinlik puanı, yöntemin tahmin edilen tüm pozitif örneklerin %87'sini doğru bir şekilde tanımladığını göstermektedir. 0,83'lük duyarlılık skoru, yöntemin Test 1'deki gerçek pozitif örneklerin %83'ünü etkili bir şekilde yakaladığını göstermektedir. 0,84'lük dengeli F-skor ise önerilen yöntemin bu veri kümesinde duyarlılık ve duyarlılık arasında uyumlu bir denge sağladığını göstermektedir.

Öte yandan, Test 2'de önerilen yöntem çeşitli ölçütlerde biraz daha düşük performans göstermektedir. Doğruluk 0,85'e düşerek Test 1'e kıyasla biraz daha düşük bir sınıflandırma doğruluğuna işaret etmektedir. 0,15'lik hata oranı, örneklerin küçük bir kısmının yanlış sınıflandırıldığını göstermektedir. Buna rağmen, kesinlik puanı 0,88 ile nispeten yüksek kalmaktadır, bu da tahmin edilen pozitif örneklerin önemli bir kısmının gerçekten pozitif olduğunu göstermektedir. Ancak, duyarlılık puanı 0,75'e düşerek yöntemin Test 2'deki gerçek pozitif örneklerin önemli bir kısmını yakalayamadığını gösterir. Sonuç olarak, F Ölçümü 0,79'a düşerek bu veri kümesinde duyarlılık ve duyarlılık arasında biraz dengesiz bir ödünleşim (trade-off) olduğunu yansıtır.

Bu karşılaştırmalı analizin sonuçları, önerilen yöntemin Test 1'de yüksek doğruluk, kesinlik, duyarlılık ve F-skor göstererek iyi performans gösterdiğini ortaya koymaktadır. Bununla birlikte, yöntemin performansı Test 2'de daha düşük duyarlılık ve F-skor puanları ile biraz düşmektedir. Bu tutarsızlık, iki veri kümesi arasındaki verilerin özellikleri veya dağılımındaki farklılıklara bağlanabilir.



Şekil 4.13: Blade and Soul veri seti Sade Bayes algoritmasına dair ROC eğrileri



Şekil 4.14: Blade and Soul veri seti Sade Bayes algoritmasına dair karmaşıklık matrisi

Doğru negatifler 1981 olarak sınıflandırılırken, doğru pozitif olarak tahmin edilen örnekler 634'tür. Yanlış pozitifler 109 iken yanlış negatifler 263'tür.

Karmaşıklık matrisi, modelin performansına genel bir bakış sağlayarak doğruluk, kesinlik, duyarlılık ve F1 puanı gibi çeşitli değerlendirme metriklerini hesaplamamıza olanak tanır. Bu metrikler, modelin pozitif ve negatif örnekleri doğru tahmin etme açısından ne kadar iyi performans gösterdiğine dair iç-görü (insight) sağlar.

Not: Karmaşıklık matrisine dayalı olarak modelin performansını tam olarak yorumlamak için sınıflandırma problemi ve pozitif ve negatif sınıfların önemi hakkında daha fazla bağlam veya bilgiye sahip olmak önemlidir.

Genel olarak bu bulgular, önerilen yöntemin sağlamlığını ve genelleştirilebilirliğini değerlendirmek için birden fazla veri kümesi üzerinde değerlendirilmesi gerektiğini vurgulamaktadır. Test 1 ve Test 2 arasında gözlemlenen farklılıklar, yöntemin değerlendirilmesi ve uygulanması sırasında farklı veri kümelerinin kendine has özelliklerinin ve ortaya çıkardığı zorlukların dikkate alınmasının önemini vurgulamaktadır. Farklı veri kümeleri ve gerçek dünya senaryolarındaki performansını artırmak için önerilen yöntemin daha fazla araştırılması ve iyileştirilmesi gerektiği yorumu yapılabilir.

Boyut indirgeme yöntemlerinden biri olan PCA Sade Bayes modeli üzerinde uygulanmış toplam öznitelik sayısı 0.95 eşik (threshold) değerinde 75 özniteliğe indirgenmiştir. Ancak sonuçlarda yükselmenin aksine bir düşüş olduğu gözlenmiştir.

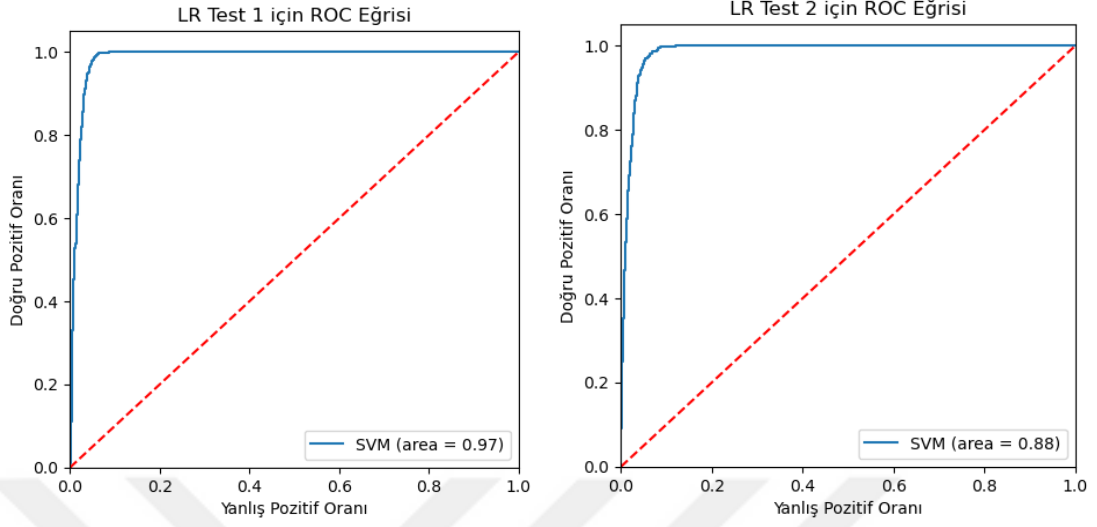
4.7. LOJİSTİK REGRESYON ALGORİTMASINA DAİR BULGULAR

Lojistik Regresyon algoritmasının Blade and Soul veri setine dair bulgulara ve performans değerlendirme ölçülerinin aldığı en yüksek değerler 10 kat çapraz geçermeye göre sonuçlara Tablo 4.16'da yer verilmiştir.

Tablo 4.15: Lojistik Regresyon sınıflandırma probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri

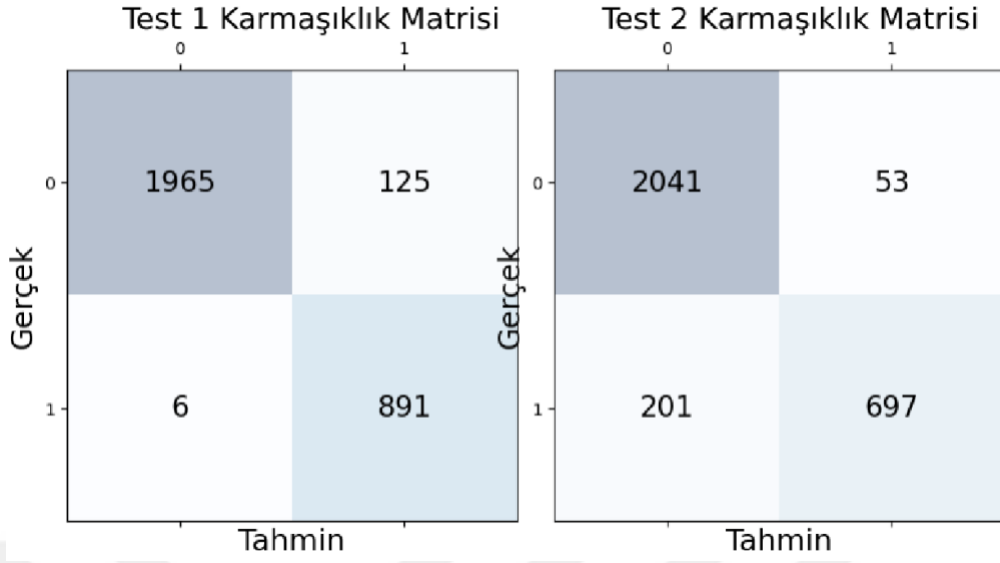
Yöntem	Veri Seti	Doğruluk	Hata	Kesinlik	Duyarlılık	F Ölçütü
Hold-out	Test 1	0.96	0.04	0.94	0.97	0.95
	Test 2	0.92	0.08	0.92	0.88	0.90
10 Kat Çapraz Geçerleme	Test 1	0.96	0.04	0.88	0.98	0.93
	Test 2	0.95	0.05	0.87	0.97	0.92

Model, her iki veri kümesinde de örneklerin çoğunu doğru sınıflandırabilmiştir. Yanlış sınıflandırılan örneklerin oranını temsil eden hata oranı Test 1 için %4 ve Test 2 için %8'dir. Daha düşük bir hata oranı, modelin sınıfları doğru bir şekilde tahmin etmede daha iyi bir performans sergilediğini gösterir.



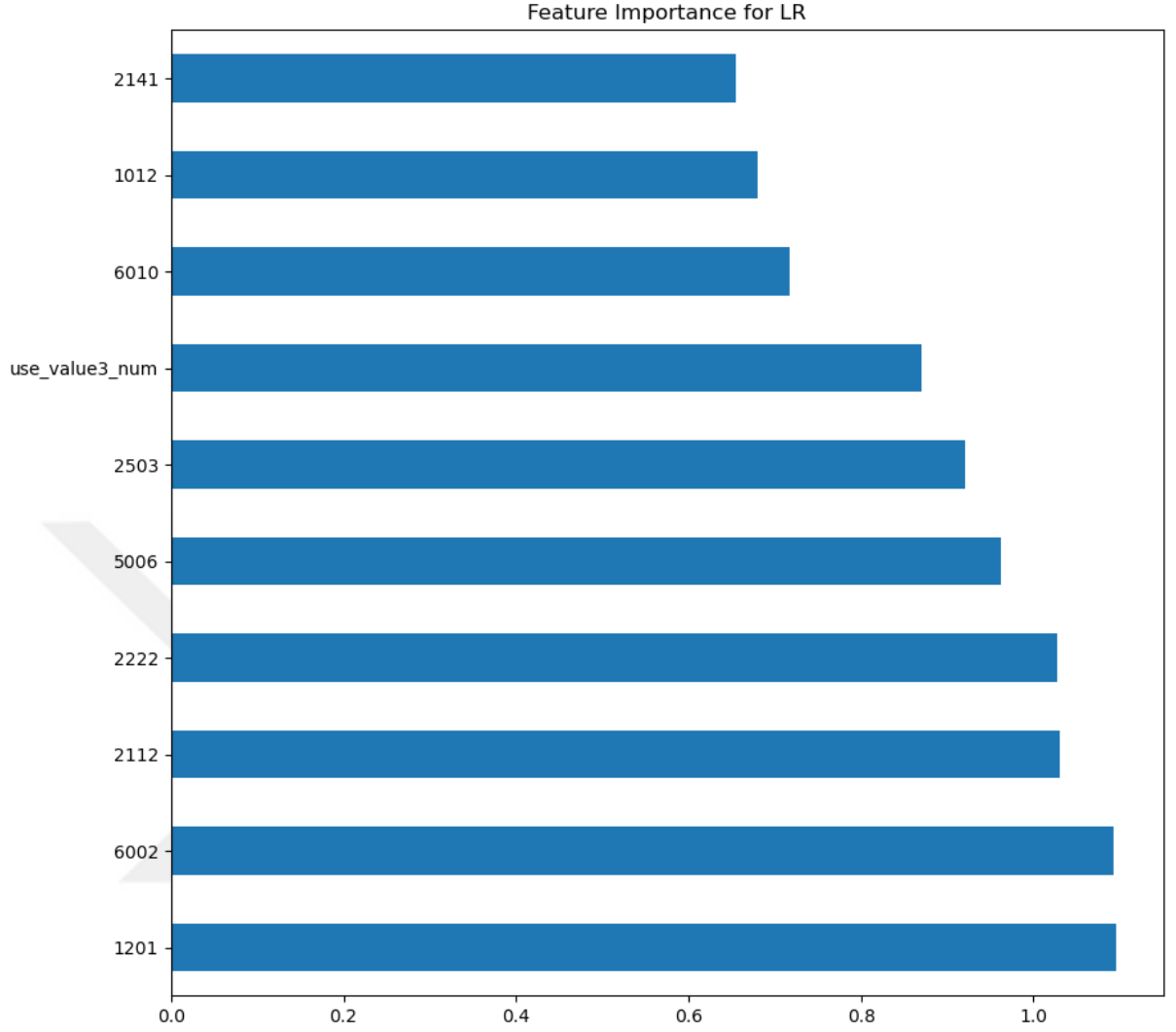
Şekil 4.15: Blade and Soul veri seti Lojistik Regresyon algoritmasına dair ROC eğrileri

ROC eğrisinin sol üst köşeye çok yakın olması, Lojistik Regresyon modelinin iyi performans gösterdiğini ve yüksek bir ayırım gücüne sahip olduğunu gösterir. Bu durum aynı zamanda, modelin düşük hata oranıyla pozitif ve negatif sınıfları doğru bir şekilde ayırt edebildiğinin de kanıtıdır. Sonuç olarak, modelin sınıflandırma görevinde başarılı olduğu ve pozitif ve negatif sınıfları iyi bir şekilde ayırt ettiği söylenebilir.



Şekil 4.16: Blade and Soul veri seti Lojistik Regresyon algoritmasına dair karmaşıklık matrisi

Şekil 4.16'daki karmaşıklık tablosuna bakıldığında modelin her iki test veri kümesi üzerinde mükemmel seviyede başarılı performans gösterdiğini söylemek mümkündür. Özellikle, gerçekte kayıp olarak nitelendirilmemiş ancak model tarafından kayıp olarak tahmin edilen sayının düşük olması, modelin performansının geleceğe yönelik öngörü sağlama ve şirket için karar verme aşamasında önemini vurgulamaktadır. Çünkü, gerçekte kayıp olmamış bir müşteriye kayıp demek, gerçekte kayıp olan ancak modelin müşteriye kayıp olarak nitelendirmesinden çok daha önemlidir.



Şekil 4.17: Blade & Soul veri seti için Lojistik Regresyon algoritması öznitelik önemliliği sonuçları

Lojistik Regresyon algoritmasının öznitelik önemliliği sonuçlarına Şekil 4.17’de yer verilmiştir. Yukarıda yer alan bulgularda en yüksek etki değerine sahip ilk 10 öznitelik paylaşılmıştır.

Öznitelik etkileri 0.65 ile 1.0 arasında değişmektedir. 0.80 üzerinde etkiye sahip 1201-oyuncunun 1v1 dövüşlerde ölmeye yakın olması diğer adıyla 1v1 dövüşlerde sersemlemesi, 6002 koduyla oyuncunun oyun grubundan ayrılması, 2112 koduyla oyuncunun kişisel envanter deposunu genişletmesi, 2222 oyuncunun depodan eşya alması, 5006 koduyla oyuncunun oyun içi görevi bitirmesiyle ödül eşya kazanması, 2503 koduyla özel oyuncu etkinliğini bitirmesi ve ardından oyun ödülü alması gibi özniteliklerin etki oranlarının yüksek seviyede olduğu söylenebilir.

Aynı zamanda ağaç tabanlı modellerin öznitelik etkilerine bakıldığında Lojistik Regresyonun çok daha farklı öznitelikleri ön plana çıkardığı görülmektedir. Rastgele Orman ve Aşırı Gradyan Arttırma algoritmalarında önemlilik etkilerinin oyuncunun kayıp tahmini üzerinde bütünsel bir etkiye sahip olduğu ve özniteliklerin birbiriyle uyumlu olduğu görülürken, Lojistik Regresyon algoritmasında bu özniteliklerin birbirinden oldukça farklılaştığı görülmektedir.

Ancak, genel bir değerlendirme ve öngörü elde edilmek istediğinde, oyuncunun 1v1 dövüşlere oyun içinde daha fazla vakit harcadığı ve özellikle eşya takasına veya depolanmasına önem verdiği de söylenebilir. Böyle bir etki oranına sahip tablonun diğer algoritmalarından gelen etki oranlarıyla karşılaştırılarak değerlendirilmesi daha faydalı olacaktır. Çünkü, ağaç tabanlı modellerle karşılaştırıldığında sadece 2222 ve 5006 kodlarının ortak olduğu görülmektedir. Bu kodlar, oyun içi eşya depolama ve görev eşyası kazanma kayıtlarına karşılık gelmektedir. Ancak, diğer modellerden gelen bilgilere göre uygulanacak olan öneriler lojistik regresyon ile farklı tutularak uygulanarak arasındaki farkı olup oluşmayacağı da belirlenebilir.

Lojistik Regresyon algoritmasının LengBear veri setine dair bulgulara ve performans değerlendirme ölçülerinin aldığı en yüksek değerlere göre sonuçlara Tablo 4.17’de yer verilmiştir.

Tablo 4.16: Lojistik Regresyon sınıflandırma probleminde LengBear veri seti için performans değerlendirme ölçü değerleri

Yöntem	Doğruluk	Hata	Kesinlik	Duyarlılık	F Ölçütü
%53-%47 kayıp_günü=4	0.75	0.25	0.53	0.52	0.52
%57-%43 kayıp_günü=5*	0.81	0.19	0.80	0.80	0.80
%57-%43 kayıp_günü=6*	0.72	0.28	0.54	0.52	0.53

*'max_iter': 7700, 'C': 0.01

Lojistik Regresyon algoritması istikrarlı bir şekilde çalışmasına rağmen, nihai skorda XGBoost'a kıyasla minimum düzeyde daha düşük bir nihai skora sahiptir. Lojistik regresyonda C parametresi bir düzenleme hiperparametresidir. Genel olarak, daha büyük bir C değeri modelin daha az düzenli olduğu ve aşırı uyuma daha yatkın olabileceği anlamına gelir. Daha

küçük bir C değeri, modelin daha düzenli olduğu ve yetersiz uyuma daha yatkın olabileceği anlamına gelir.

%57-%43 bölümlerde 5 günlük model nispeten daha iyi bir sınıflandırma gerçekleştirerek ve hata oranı %19 olarak daha düşük olduğu görülmüştür. Bu da daha az yanlış sınıflandırmalar yapıldığını gösterir. Kesinlik ve duyarlılık değerleri modelin hem yanlış pozitif hem de yanlış negatif oranlarının düşük olduğunu gösterir. Son olarak, F ölçütü de 0.80 olarak hesaplanmış olup sonuçlar, ikinci yöntemin daha iyi bir sınıflandırma performansı sunduğunu göstermektedir. Bu nedenle, ikinci yöntemin veri setinin sınıflandırılması için daha etkili olacağı savunulabilir.

4.8. ÇOK KATMANLI PERSEPTRON ALGORİTMASINA DAİR BULGULAR

Çok Katmanlı Perseptron algoritmasının Blade and Soul veri setine dair bulgulara ve performans değerlendirme ölçülerinin aldığı en yüksek değerler 10 kat çapraz geçiremeye göre sonuçlara Tablo 4.18’de yer verilmiştir.

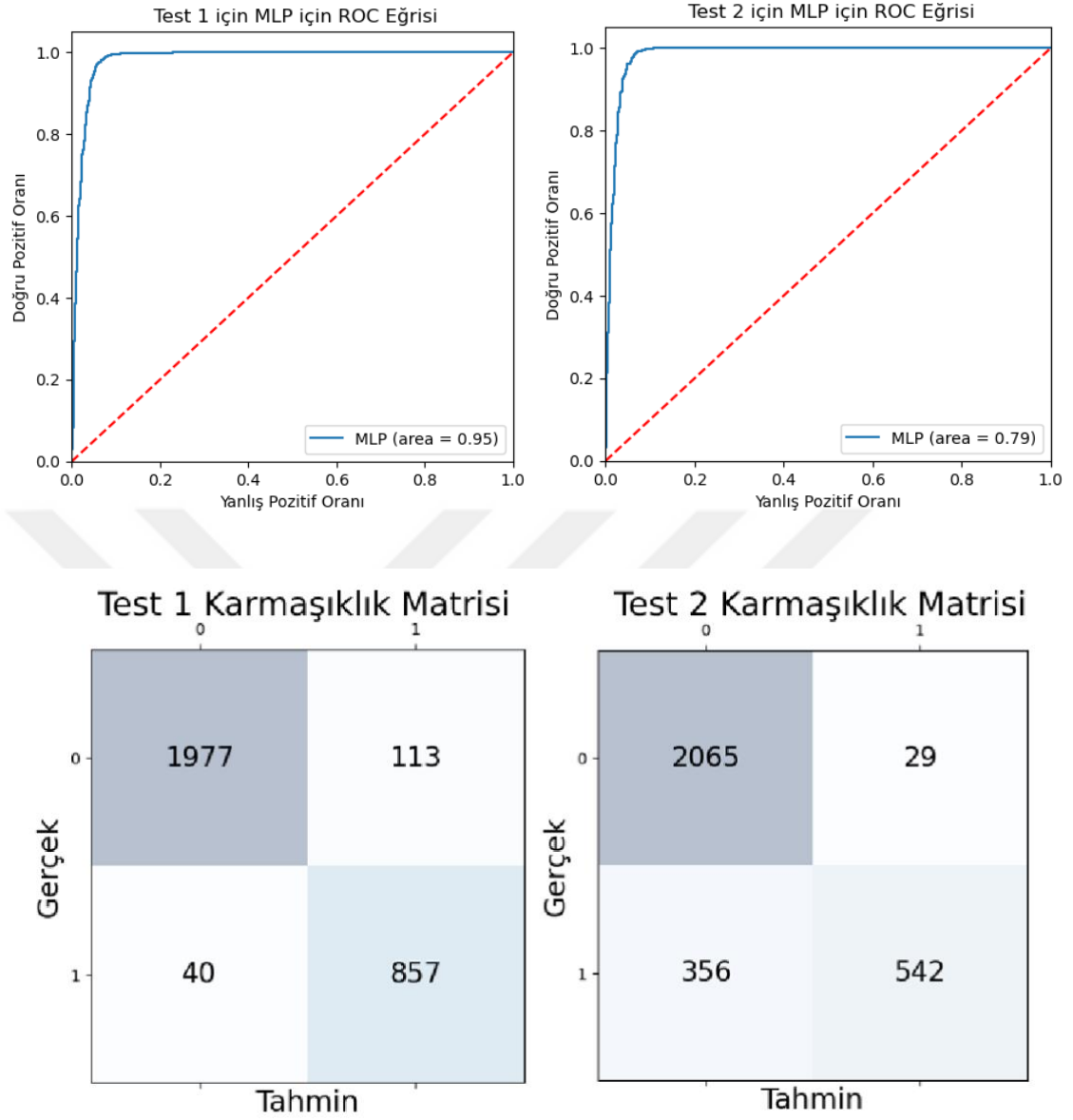
Tablo 4.17: Çok Katmanlı Perseptron sınıflandırma probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri

Yöntem	Veri Seti	Doğruluk	Hata	Kesinlik	Duyarlılık	F Ölçütü
Hold-out	Test 1	0.74	0.26	0.93	0.94	0.93
	Test 2	0.86	0.14	0.89	0.77	0.80
10 Kat Çapraz Geçerleme	Test 1	0.96	0.04	0.90	0.98	0.93
	Test 2	0.95	0.05	0.88	0.95	0.91

Test 2’de model 0.86 olan değeri daha yüksek bir doğruluk oranına ulaşarak Test 1’e kıyasla daha iyi performans göstermiştir. 0,89’luk duyarlılık, yüksek bir doğru pozitif tahmin yüzdesine işaret etmekle birlikte duyarlılık değerinin 0,77 olması, modelin gerçek pozitif örneklerin iyi seviyede yakaladığını göstermektedir. 0,80’lik F1 puanı ise modelin performansında dengeli bir değerlendirme ölçütü sağlar.

Modelde 10 kat çapraz geçerleme uygulandığında ise tahmin performansında artış olduğu görülmektedir. Özellikle, Test 1 ve Test 2 veri setlerinde doğruluk, duyarlılık ve hassasiyet değerlerinin mükemmel seviyeye geldiği görülmektedir. Modele çarpaz geçerleme uygulandığında performansının artış göstermesinin potansiyel sebepleri şu şekilde sıralanabilir;

- Çapraz doğrulama, modeli verilerin birden fazla alt kümesi üzerinde eğiterek ve değerlendirerek aşırı uyumu azaltmaya yardımcı olan bir tekniktir. Bu, modelin eğitim verilerini ezberleme riskini azaltır ve görülmeyen verilere genellemesini geliştirir. Ayrıca test veri kümesinde daha iyi performansa yol açan bir düzenleme etkisi sağlar.
- MLP modelleri başlangıç ağırlıklarına, önyargılara ve eğitim verilerinin rastgele karıştırılmasına karşı hassastır. Çapraz doğrulamanın birden fazla kez yapılmasıyla modelin performansı, verilerin farklı rastgele bölünmelerinin ortalaması alınarak rastgele dalgalanmaların etkisi azaltılır ve daha istikrarlı bir performans tahmini sağlanır.
- Çapraz doğrulama, birden fazla kat üzerinden sonuçların ortalamasını alarak modelin performansının daha kapsamlı bir değerlendirmesini sunar. Verilerin farklı alt kümeleri arasında modelin tutarlılığının ve güvenilirliğinin değerlendirilmesine olanak tanır. Çapraz doğrulama iterasyonlarının sayısının artırılması (5 veya 10 kez) modelin gerçek performansının daha iyi tahmin edilmesini sağlar ve tek bir eğitim-test ayrımında rastgele şansın etkisini azaltır.
- Ayrıca çapraz doğrulama hiperparametre ayarı için de kullanılabilir. Modelin performansını farklı hiperparametre ayarları arasında değerlendirerek, en iyi sonuçları veren optimum yapılandırmanın belirlenmesine yardımcı olur. Çapraz doğrulamanın yinelemeli yapısı, hiperparametre uzayının kapsamlı bir şekilde araştırılmasına olanak tanıyarak performansın iyileştirilmesini sağlar.



Şekil 4.18: Çok Katmanlı Algılayıcı Blade & Soul veri seti için ROC eğrileri ve karmaşıklık matrisleri

Özetle, model Test 2'de Test 1'e kıyasla daha yüksek doğruluk, daha düşük hata oranı ve biraz daha iyi duyarlılık ve F1 skoruyla nispeten daha iyi performans göstermiştir. Bununla birlikte, modelin pozitif örnekleri doğru bir şekilde tanımlama becerisi her iki test setinde de daha düşük olduğundan, özellikle duyarlılık değerinin biraz daha düşük kaldığı söylenebilir.

Test 1 üzerinde test edildiğinde MLP sınıflandırıcının doğruluğu 0.74'e düşmüştür. Bu, modelin eğitim verilerine aşırı uyum sağlamış olabileceğini ve yeni, görülmemiş verilere iyi genelleme yapamadığını göstermektedir. Ancak bu sorun, 10 çapraz geçişle giderilmiştir.

Öte yandan, Test 2 üzerinde test edildiğinde MLP sınıflandırma tahmini 0.86 doğruluk değeri elde etmiştir. Eğitim setindeki doğruluktan daha düşük olsa da Test 1'deki doğruluktan daha yüksektir. Bu, modelin Test 2'de nispeten daha iyi performans gösterdiğini ve Test 1'e kıyasla daha iyi genelleme yaptığını göstermektedir.

Sonuç olarak Şekil 4.19'da ki karmaşıklık matrisine bakıldığında 10 kat çapraz geçişleme sonuçları modelin iyi bir performans ortaya koyduğunu göstermektedir.

Tablo 4.18: Çok Katmanlı Algılayıcı regresyon probleminde Blade & Soul veri seti için performans değerlendirme ölçü değerleri

Veri Seti	R^2	Düzeltilmiş - R^2	RMSE
Test 1	0.85	0.87	0.162
Test 2	0.77	0.79	0.197

R^2 değerleri, regresyon modelinin Test 1 ($R^2 = 0.85$) için Test 2'ye ($R^2 = 0.77$) kıyasla bağımlı değişkendeki değişkenliğin daha yüksek bir oranını açıkladığını göstermektedir. Düzeltilmiş R^2 , modeldeki yordayıcı sayısını dikkate alır ve gereksiz yordayıcıların dahil edilmesini cezalandırır. Test 1'deki daha yüksek düzeltilmiş R^2 , modelin karmaşıklığını Test 1'de daha etkili bir şekilde hesaba kattığını göstermektedir. Test 1, Test 2'ye (0.197) kıyasla daha düşük bir RMSE'ye (0.162) sahiptir. Daha düşük bir RMSE, modelin tahminlerinin gözlenen değerlere daha yakın olduğunu gösterir ve Test 1 için daha iyi tahmin performansına sahip olduğunu göstermektedir.

Genel olarak sonuçlar, regresyon modelinin Test 1'de Test 2'ye kıyasla daha iyi performans gösterdiğini ortaya koymaktadır. Model, bağımlı öznelikteki değişkenliğin daha yüksek bir oranını açıklamakta, modelin karmaşıklığını daha etkili bir şekilde hesaba katmakta ve Test 1'de daha düşük ortalama tahmin hatalarına sahip olmaktadır. Bu bulgular, modelin genelleme yeteneklerini değerlendirmek için farklı veri kümeleri üzerindeki performansını değerlendirmenin önemini vurgulamaktadır.

Tablo 4.19: Çok Katmanlı Perseptron sınıflandırma probleminde Blade & Soul veri setine dair arama yöntemlerine göre performans değerlendirme ölçü değerleri

Arama Yöntemi	Veri Seti	Doğruluk	Hata	Kesinlik	Duyarlılık	F Ölçütü
---------------	-----------	----------	------	----------	------------	----------

Izgara Arama*	Test 1	0.95	0.05	0.94	0.96	0.95
	Test 2	0.87	0.18	0.90	0.79	0.82
Rastgele Arama**	Test 1	0.94	0.06	0.93	0.95	0.94
	Test 2	0.83	0.17	0.90	0.80	0.83

**hidden_layer_sizes=(10,30,10),shuffle=True,activation='relu',alpha=0.0001,learning_rate='constant', max_iter=50, solver = 'adam'*

***hidden_layer_sizes=(20,),shuffle=True,activation='tanh',alpha=0.0001,learning_rate='adaptive', max_iter=50, solver = 'adam'*

Her iki arama metodunda MLP algoritmasının Test 1 verisi üzerinde “aşırı uyum” sorunu ortaya koymuştur. Eğitim ve test veri setleri başarımları yüzdeleri 0.96-0.74 şeklindedir. Bu sonuç modelin ezbere daha yakın olduğu ve sisteme dahil edilecek yeni verilerden sonuç çıkarımı yapamaması anlamına gelmektedir.

Bu sonuçlara göre hem Izgara Arama hem de Rastgele Arama yöntemleri Test 1'de benzer performans gösterirken, Izgara Arama Test 2'de doğruluk, kesinlik, duyarlılık ve F-ölçümü açısından biraz daha iyi performans göstermektedir. Bu bulgular, Izgara Arama'nın verilen veri kümesi için modelin hiper-parametrelerini optimize etmede daha etkili olabileceğini ve görünmeyen veriler üzerinde daha iyi performansa yol açabileceğini göstermektedir. Ancak, kesin sonuçlara varmak ve gözlemlenen farklılıkların istatistiksel önemini değerlendirmek için daha fazla analiz ve deney yapılması gerekmektedir.

Tablo 4.20: Çok Katmanlı Perseptron sınıflandırma probleminde LengeBear veri seti için performans değerlendirme ölçü değerleri

Yöntem	Doğruluk	Hata	Kesinlik	Duyarlılık	F Ölçütü
%53-%47 kayıp_günü=4	0.78	0.22	0.69	0.70	0.70
%57-%43 kayıp_günü=5*	0.82	0.18	0.79	0.80	0.80
%57-%43 kayıp_günü=6	0.69	0.31	0.70	0.71	0.71

**activation='relu', alpha=0.0001, batch_size='auto', beta_1=0.9*

Mobil oyun verisi olan LengBear üzerinde gerçekleştirilen Çok Katmanlı Algılayıcı algoritmasının sonuçları Tablo 4.21’de verilmiştir. Mobil oyunlar MMORPG oyunlardan farklı değerlendirilmektedir. Mobil platformda yayınlanan oyunlar hızlı tüketildiğinden ve kompleks yapıda olmadığından dolayı müşteri kayıp süresi diğer platform türlerine göre daha kısadır. Literatür taraması bölümünde görüldüğü üzere mobil oyunda müşteri kayıp süresi ağırlıklı olarak 4 ile 7 gün arasında değişmektedir.

Mevcut veri üzerinde 4, 5 ve 6 günlük müşteri kayıp atamaları yapılarak 3 farklı sonuç elde edilmiştir. Tablo 4.21’e göre müşteri kayıp süresi 5 gün olarak değerlendirildiğinde F1 skor 0.80 ile iyi 4 günlük değerlendirmeye göre iyi bir artış göstermiştir. Bu da oyuncuların kayıp analizinde ön görüş sağlanması ve oyun içi değişiklik yapılması konusunda oldukça etkilidir.

Müşteri kayıp süresi 4 gün baz alındığında kesinlik oranı 0.69, 5 gün 0.79 ve 6 günde ise 0.70 F1 skoruyla olarak performans göstermiştir. Yüksek kesinlik oranları, pozitif tahminlerin genellikle doğru olduğunu göstermektedir. Ancak, gerçekleştirilen tahminlerin doğruluğu, hata oranı ve diğer performans ölçümleri kayıp süresinin 5 gün alındığı duruma kıyasla biraz daha düşüktür. Duyarlılık ve F1 skor bazında bakıldığında ise kesinlik gibi bu değerlerin de oranında benzer şekilde artış olduğu görülmektedir. Bu, tahmin modelinin doğruluğunun ve performansının belirli bir koşula bağlı olarak değişebileceğini göstermektedir.

5. TARTIŞMA VE SONUÇ

Müşteri kaybı analizi, oyun geliştiricilerinin ve yayıncılarının oyuncu kaybını anlamalarına ve yönetmelerine yardımcı olan oyun analitiğinin kritik bir yönüdür. Oyun şirketleri, oyuncu davranış kalıplarını inceleyerek ve müşteri kaybına katkıda bulunan faktörleri belirleyerek, müşteri kaybı oranlarını azaltmak ve oyuncuları elde tutmayı artırmak için bilinçli kararlar alabilir. Araştırma çalışmaları, oyun sektöründe müşteri kaybı analizinin önemini vurgulamaktadır. Wang ve Zhao (2018) müşteri kaybı analizinin oyun şirketlerinin oyuncu tercihlerini anlamasını, müşteri kaybına neden olan temel faktörleri belirlemesini ve etkili oyuncu tutma stratejileri uygulamasını sağladığını vurgulamaktadır. Ayrıca, Hadiji ve diğ. (2020), mevcut oyuncuları elde tutmanın yeni oyuncular edinmekten daha uygun maliyetli olması nedeniyle, oyuncu kaybının finansal etkilerini vurgulamaktadır. Oyun şirketleri, müşteri kaybı analizinden yararlanarak kaynaklarını optimize edebilir ve değerli oyuncuları elde tutmaya yönelik çabalarını tahsis edebilir.

Oyun bağlamında müşteri kaybı analizi yapmak için çeşitli teknikler ve veri kaynakları kullanılmaktadır. Akademik literatür, müşteri kaybı analizi için öngörücü modeller geliştirmek üzere oyun içi eylemler, etkileşim ölçümleri ve sosyal etkileşimler gibi oyuncu davranışsal verilerinin kullanımını özetlemektedir. Wang ve Zhao (2018), karar ağaçları ve lojistik regresyon gibi makine öğrenimi algoritmalarının geçmiş oyuncu verilerine dayanarak oyuncu kaybını tahmin etmek için kullanılabileceğini öne sürmektedir. Ayrıca, oyun şirketleri, oyuncu tercihleri ve oyuna yönelik duygular hakkında daha derin iç-görüler elde etmek için genellikle sosyal medya platformlarından oyuncu geri bildirimlerini ve duygu analizlerini dahil etmektedir. Oyun şirketleri, çeşitli veri kaynaklarını bütünleştirerek ve sofistike analitik yaklaşımlar kullanarak, oyuncu kaybını azaltmak ve oyuncuyu elde tutmayı iyileştirmek için eyleme geçirilebilir stratejileri yönlendiren değerli iç-görülerini ortaya çıkarabilir.

Sonuç olarak, müşteri kaybı analizi oyun endüstrisinde çok önemli bir rol oynamakta ve oyun şirketlerinin oyuncu davranışını anlamasını, oyuncu kaybına katkıda bulunan faktörleri belirlemesini ve etkili oyuncu tutma stratejileri uygulamasını sağlamaktadır. Oyuncu verilerinin analizi ve makine öğrenimi algoritmalarının uygulanması yoluyla, oyun şirketleri müşteri kaybını tahmin edebilir ve değerli oyuncuları elde tutmak için hedefli müdahaleler geliştirebilir. Oyun şirketleri, oyuncu kaybı analizinin önemini vurgulayarak ve gelişmiş analitik

tekniklerden yararlanarak çabalarını optimize edebilir, oyuncu memnuniyetini artırabilir ve finansal performansı iyileştirebilir.

Tablo 5.1 ve 5.2’de yer alan değerler, algoritmalar arasındaki ikili karşılaştırmalar için p-değerlerini temsil etmektedir. Anlamlılık seviyeleri aşağıdaki gibi gösterilmiştir: $p < 0,05$ (istatistiksel olarak anlamlı fark var) için * ve (istatistiksel olarak anlamlı fark yok) için ** kullanılmıştır.

Tablo 5.1: Test 1 veri seti için farklı makine öğrenimi algoritmalarının belirli metrikler üzerindeki performansları açısından bir karşılaştırması – anlamlılık testleri

	KNN	XGB	RF	SVM	LR	ÇKA
KNN	-----	0.000*	0.180**	0.000*	0.000*	0.000*
XGB		-----	0.0042*	0.143**	0.000*	0.146*
RF			-----	0.000*	0.000*	0.000*
SVM				-----	0.116**	0.127*
LR					-----	0.129*
ÇKA						-----

* İstatistiksel olarak anlamlı fark var

** İstatistiksel olarak anlamlı fark yok

Tablo 5.2: Test 2 veri seti için farklı makine öğrenimi algoritmalarının belirli metrikler üzerindeki performansları açısından bir karşılaştırması – anlamlılık testleri

	KNN	XGB	RO	DVM	LR	ÇKA
KNN	-----	0.000*	0.239**	0.000*	0.000*	0.000*
XGB		-----	0.065**	0.132**	0.319**	0.186*
RO			-----	0.000*	0.000*	0.000*
DVM				-----	0.078**	0.000*
LR					-----	0.154*

* İstatistiksel olarak anlamlı fark var

** İstatistiksel olarak anlamlı fark yok

Tablo 5.2’de Test 2 veri seti için farklı makine öğrenimi algoritmalarının (KNN, XGB, RF, SVM, LR) belirli metrikler üzerindeki performansları açısından bir karşılaştırması verilmiştir.

Blade and Soul veri seti için sonuçlar ve öneriler;

Sınıflandırma problemi için XGBoost, SVM (Destek Vektör Makineleri), LR (Lojistik Regresyon) Test 1 ve Test 2 verileri üzerinde F1 skor değerlerinin tahmininde birbirine yakın performans gösterdiğini göstermektedir. Algoritmalar arasında XGBoost en yüksek doğruluk oranına sahiptir ve Test 1 ile Test 2 setleri arasındaki farkı en aza indiren algoritma olarak karşımıza çıkmaktadır.

NCSOFT, sırasıyla Test 1 ve Test 2 veri kümelerini toplarken P2P (Pay to Play) pazarlama stratejisinden F2P (Free to Play) pazarlama stratejisine geçmiştir. Bu nedenle, bazı algoritmalarda (XGBoost, Lojistik Regresyon ve Destek Vektör Makinesi) farkın nispeten yüksek olduğu F2P modelinin Test 2 veri kümesindeki doğruluk oranları tutarlı sonuç vermemektedir. Bunun nedeni, P2P modelinde oyuncuların belirli bir bedel ödeyerek oyuna dahil olması, oyunlarda müşteri kaybı analizi için belirli faktörleri filtreleyerek oyunu seven ve oynayan oyuncuların olduğunu göstermektedir. Ancak F2P modelinde oyuncu oyunu denemek için indirmiş, bir kez oynamış ve muhtemelen bir daha giriş yapmamıştır. Bu nedenle, oyunculara kampanya uygulamak elde tutma oranının F2P modelindeki kadar düşük olmayacağından daha mantıklıdır. Buna göre F2P modelinde müşteri kaybı analizi yapmanın daha zor olduğu yorumu yapılabilir. Bu nedenle modeller arasında performans farkı ortaya çıkmaktadır.

Makine öğrenmesi perspektifinden bakıldığında;

Test 1 verisi için ÇKA, XGBoost ve Lojistik Regresyon ile anlamlı farklılık ortaya çıkmamakla birlikte diğer algoritmalarla arasında performans bakımından anlamlı farklılık ortaya çıkmıştır. Test 2 verisi için ÇKA, XGBoost ve Lojistik Regresyon ile anlamlı farklılık ortaya çıkmamakla birlikte diğer algoritmalarla arasında performans bakımından anlamlı farklılık ortaya çıkmıştır.

KNN algoritmasının performansı göz önünde bulundurulduğunda, k sayısındaki artış modelin performansında iyileştirme sağlamamıştır. Test 2 veri seti üzerinde herhangi bir değişiklik olmamakla birlikte Test 1 veri seti için $k=3$ baz alındığında çok ufak bir miktar artış görülmüştür. Bu durumun, k-en yakın komşu algoritmasının tembel öğrenen yapısından

kaynaklandığı düşünülmektedir. Ayrıca, yüksek boyutta yeni verilerle birlikte model oluşturulduğunda oluşacak olan örneklemelerin birbirine çok yakın olmasının k değerinin yüksek seçildiğinde modelin performansının düşmesine sebep olabileceği de ihtimaller arasındadır. Ancak, genel bir kestirim yapmak için k -en yakın komşu algoritmasının iyi bir seçenek olduğunu belirtmekte de fayda vardır. Özellikle gerçekte kayıp olarak nitelendirilmeyen ancak modelde kayıp olarak tahmin edilen sayısı da oldukça yüksektir. Bu sebep modelin güvenilirliğini nispeten düşük hale getirmektedir. Castro ve Tsuzuki (2015) çalışmasında bu tez çalışmasında olduğu gibi oturma kayıtlarını kullanarak gerçek zamanlı veriler üzerinde gerçekleştirdiği çalışmada KNN oldukça iyi performans göstermiştir. Yine bir diğer çalışmada Biçer (2021) mobil oyun üzerinde gerçekleştirdiği çalışmada KNN ile 0.88 oranında başarı oranı elde etmiştir.

Karar Ağaçları'na bakıldığında ağaç tabanlı modellerin Blade and Soul veri seti üzerinde iyi seviyede çalıştığını söylemek mümkündür. Ancak karar ağaçlarındaki asıl problem Test 2 veri seti üzerinde duyarlılık değerinin Test 1 veri setine göre çok daha düşük olmasıdır. Bu da k -en yakın komşu algoritmasında olduğu gibi modelin performansını etkileyerek tek bir yönde tahminin çok iyi olup diğer yönde çok kötü performans sergilemesine yol açmaktadır. Tahmin gerçekleştirirken asıl hedef hem kesinlik hem de duyarlılık üzerinde dengeli bir tahmin performansı ortaya koymasıdır. Bu bakımdan tez kapsamında kullanılan veriler üzerinde karar ağaçları da genel bir tahmin yürütmek için iyi bir seçim olabilir ancak daha iyi ve dengeli performans gösteren algoritmalar olduğundan ötürü detaylı iç görü oluşturmada yetersiz kalmıştır.

Rastgele Orman ve Aşırı Gradyan Arttırma algoritmalarına bakıldığında Rastgele Orman Karar Ağaçlarına benzer bir performans sergilerken, Aşırı Gradyan Arttırma mükemmel ve oldukça dengeli bir tahmin performansı ortaya koymuştur. XGBoost algoritmasının Rastgele Orman'a göre bir takım avantajlı yönleri olduğu literatür kısmında bahsedilmiştir. Verinin yapısının uygunluğu ve XGBoost'un kompleks veri setleri üzerinde daha iyi performans göstermesi sebepleri arasında gösterilebilir. Ayrıca, XGBoost her iterasyonda hataları en aza indirirken zayıf öğrenenleri iyileştirmeye odaklanır. Öte yandan, Random Forest bağımsız olarak birden fazla karar ağacı oluşturur ve tahminlerini birleştirmesi performans açısından oldukça etkilidir. Yine, XGBoost, öznitelikler arasındaki karmaşık ilişkileri ve etkileşimleri yakalama becerisinde oldukça iyidir. Sırayla zayıf öğreniciler (lazy learner) eklenerek modelin

performansını optimize etmek için gradyan artırma teknikleri kullanır. Bu, XGBoost'un doğrusal olmayan ilişkileri etkili bir şekilde ele almasını ve verilerdeki daha incelikli kalıpları yakalamasını sağlar. Rastgele Orman ise, bazı doğrusal olmayan ilişkileri yakalayabilmesine rağmen, karmaşık etkileşimleri etkili bir şekilde yakalamakta zorlanabilir. Bunlara ek olarak, makine öğrenmesi algoritmalarının performansı, veri kümesinin belirli özelliklerinden etkilenmektedir.

Veri seti analizi hem P2P hem de F2P modelleri için tutarlı bir model kurmayı ve veri setlerinde anlamlı sonuçlar üretebilmeyi amaçlamaktadır. Doğruluk metriği sadece pozitif gerçek ve tahmin edilen değerlere odaklandığı için bu durum araştırmaya dahil edilmiştir. Ancak, oyunlarda müşteri kaybı analizinde negatif tahmini ve gerçek değerler de bir o kadar önemlidir. Bu nedenle Test 1 için F1 skor en iyi 0,96 Test 2 veri seti içinse 0.93 ile XGBoost algoritmasının performansında elde edilmiştir.

XGBoost algoritmasında maksimum derinlik = 6 değerinin seçilmesinin model performansı açısından daha iyi performans gösterdiği görülmüştür. XGBoost bir topluluk algoritması olduğundan, tahminleyici sayısı ve ikili çözücü gibi diğer parametreler de model seçimi ve aşırı uyum açısından kritik öneme sahiptir. XGBoost, seyrekliğe duyarlı öğrenme, ağırlıklı kantil ve ağaç paralel öğrenme sisteminin yapısından dolayı daha verimli hale getiren bir dizi optimizasyon tekniğine sahiptir. Öğrenme oranı, ağaçların maksimum derinliği ve topluluktaki ağaç sayısı gibi performans artırmak için ayarlanabilen çok çeşitli hiper parametrelere sahiptir. Ayrıca, birçok gerçek dünya veri kümesinde büyük bir zorluk olabilen girdi verilerindeki eksik değerleri de işleyebilmektedir.

Destek Vektör Makineleri ve Çok Katmanlı Algılayıcı algoritmalarında ise model performans değerlerinin benzer olduğu görülmektedir. Özellikle her iki algoritma Test 1 ve Test 2 veri seti üzerinde kesinlik ve duyarlılık açısından mükemmel denilebilecek performansa sahiptir. Ancak, tasarlanan modellerde işleme süresi ve optimizasyon göz önünde bulundurulduğunda ÇKA özellikle hiper parametre seçiminin dikkatle yapılması gereken bir algoritma olduğundan ötürü işlem süresinde de uzamalar meydana gelebilir. ÇKA'da 10 kat çapraz geçişleme kullanılarak modelin performansında büyük bir gelişme meydana gelmiştir. Bu da verilerin karıştırılarak kullanıldığında ve hiper parametre ayarı iyi yapıldığında etkili sonuçlar verebileceğini ortaya koymaktadır. DVM için 'C' ve 'gamma' oranı minimize edildiğinde daha iyi sonuçlar elde edilmektedir. Öte yandan SVM, F1 skorda Lojistik Regresyon ile benzer

performans göstermiştir. Radyal Taban Fonksiyonu (RBF) çekirdeği, doğrusal olarak ayrılamayan verileri işleyebildiği ve uygulamada genellikle iyi performans sağladığı için destek vektör makineleri için yaygın bir seçimdir. Özellikle Test 2 veri seti üzerinde SVM algoritmasında iyi doğrusal modele göre nispeten daha iyi performans sağladığı görülmüştür. Aynı zamanda, aşırı uyum ve yetersiz uyumdan kaçınmak için C parametresinin değeri çok yüksek ayarlanmamıştır.

Son olarak Lojistik Regresyon algoritması istikrarlı bir şekilde çalışmasına rağmen, nihai skorda XGBoost'a kıyasla minimum düzeyde daha düşük bir F1 skora sahiptir. Lojistik regresyonda C parametresi, düzenleştirmenin gücünü kontrol eden bir düzenleştirme için kullanılan bir hiper parametre olup, genel olarak, daha büyük bir C değeri modelin daha az düzenli olduğu ve aşırı uyuma daha yatkın olabileceği söylenebilir. Tez kapsamında kullanılan veri setleri üzerinde iyi performans gösterdiği görülmektedir.

Veri Analitiği ve uygulamaları perspektifinden bakıldığında;

XGBoost algoritmasında öznitelik önemliliği tablosunda yer alan daha yüksek puanlar, kayıpları tahmin etmede daha büyük öneme işaret etmektedir. Daha yüksek önem puanlarına sahip özellikler, bir müşterinin ayrılma olasılığının olup olmadığını belirlemede daha etkilidir. Aynı zamanda, bu öznitelikler müşterilerin karakteristiğini veya davranışını temsil etmektedir. Bu özellikleri müşteri kaybı analizi bağlamında yorumlanması daha doğru olacaktır. Yorumlama yaparken Etki Yönü (Direction of Impact) göz önünde bulundurulduğunda, bazı öznitelikler müşteri kaybı tahmini üzerinde olumlu bir etkiye sahip olabilirken, diğerleri olumsuz bir etkiye sahip olabilir. Etkinin yönünün anlaşılması, hedeflenen müdahalelerin oluşturulmasına yardımcı olmaktadır. Bu doğrultuda, Blade and Soul oyununda en iyi performans gösteren XGBoost algoritmasının sağladığı öznitelik önemliliği sonuçları göz önünde bulundurarak kayıp analizinde şirkete öngörü sağlama konusunda bir analiz gerçekleştirildiğinde;

- 4001 koduna karşılık gelen “oyuncunun özel yetenek kazandığında (22.0)”, oyuncunun oyun içinde ilerlediğini ve yeni yetenekler kazandığını göstermektedir. Yüksek bir önem düzeyine sahip olması, oyuncunun oyunun ileriki aşamalarına daha fazla yatırım yaptığını ve dolayısıyla daha az kayıp olma olasılığının artırdığı düşünülebilir.

- 2209 koduna karşılık gelen “oyuncunun NPC (Non-Player Character) takas yoluyla eşya alması (17.0)”, oyuncunun oyun içinde ekonomik etkileşimlerde bulunduğunu ve kaynakları doğru bir şekilde kullanma eğiliminde olduğunu gösterebilir. Bu da oyuncunun oyuna daha bağlı ve aktif olarak oynadığı yorumu yapılabilir.
- 1003 koduna karşılık gelen “oyuncunun oyun sunucusuna giriş yapması (14.0)”, oyuncunun düzenli olarak oyun sunucusuna bağlandığını ve aktif olduğunu gösterir. Bu da aktif oyuncuların genellikle daha az kayıp eğiliminde olduğu bilinmektedir. Aynı zamanda bu öz nitelikte oyuncunun oyunda kalması açısından bir takım mikro-etkileşimler uygulanabilir.
- clv_1003 koduna karşılık gelen “oyunda geçirilen süre (gün bazında) (11.0)”, oyun içinde harcadığı zamanı göstermektedir. Genellikle, daha fazla zaman harcayan oyuncuların daha bağlı olduğu ve kayıp olarak nitelendirilme olasılığının daha düşük olduğu düşünülür.
- 4001 koduna karşılık gelen “oyuncunun yetenek kazanması (11.0)”, bir önceki öz niteliğe benzer şekilde, oyuncunun ilerlemesini ve yeni yetenekler kazanmasını gösterir. Ancak bu kısımda, oyuncu oyun içinde ne kadar fazla yetenek puanı kazanırsa, o kadar kayıp olma olasılığı azalır. Çünkü, oyunda geçirilen süre kayıp olma olasılığını azaltmaktadır. Bu noktada, oyun içinde tecrübe puanı kazanma şekillerinde bir güncelleme paketi yayınlayarak, oyuncuların tecrübe puanı aracılığıyla oyun içinde aktif katılım göstermesi sağlanabilir. Öte yandan, oyuncunun tecrübe puanı kazandırılarak oyunda tutulması, oyuna olan bağlılığına da artıracığı tahmin edilmektedir.
- 6005 koduna karşılık gelen “oyuncunun klana katılması (9.0)”, oyuncunun sosyal olarak oyun içinde etkin olduğunu gösterir. Öte yandan, genellikle oyunu bırakma olasılığını azalttığı yorumu yapılabilir çünkü sosyal bağlantılar (social interactions) oyuncuları oyunun içinde tutmada oldukça etkilidir. Çünkü, oyuncular arasında etkileşim oyun oynama süresinin de artmasını sağlayarak, oyuncuların topluluk duygusu oluşturarak, oyuncuların birbirleriyle etkileşime geçmelerini sağlayacaktır. Aynı şekilde, klan yöneticileri, üyeler arasında bir ara bağlantı görevi görerek, yeni oyunculara kılavuzluk etmek, sorularını yanıtlamak ve destek sağlamak, onların oyun içinde daha rahat hissetmelerini ve klanda daha uzun süre kalmalarını sağlayabilir. Yine bu doğrultuda, klan içindeki en aktif oyunculara özel ödüller veya klan içi rütbelere vererek, oyuncuların

rekabet etmelerini ve daha fazla zaman harcamalarını teşvik edebilirsiniz. Sonuç olarak, klana katılma konusunda oyun tasarımı aşamasında oyunculara farklı alternatifler sunulması ve içerikle bağlantılı şekilde oyun içi görevlere uyarlanması, kayıp oranını artırmada önemli sayılabilir.

- 1103 koduna karşılık gelen “oyuncunun grup davetini reddetmesi (9.0)”, oyuncuların oyun içi gruplarda bulunmasının kayıp oranını artırmaya olan etkisini göstermektedir. Bir diğer ifadeyle, oyuncunun sosyal etkileşimlere katılmadığını ve bu noktada oyun içi gruplardan ziyade bir klana ait olmanın oyunda daha bağlayıcı olduğu bir önceki analizde görülmektedir. Gruplar oyun içinde belirli görevleri gerçekleştirdikten sonra oyuncular tarafından kısa süre sonra silinir. Gruplar, klana göre daha kısa süreli ve hedefe yöneliktir. Bu sebepten ötürü oyunculara uzun vadeli klanlara dahil olmak daha cazip gelebilmektedir. Öneri olarak, grup davetlerinin içeriği gözden geçirilebilir. Ayrıca, davetlerin zamanlaması ve sıklığı da oyuncuların tercihleri doğrultusunda ayarlanabilir.
- 2209 koduna karşılık gelen “oyuncunun karakteri seviye atlaması (9.0)”, Bu öznelik, oyuncunun ilerlemesini ve karakterinin seviye atlamasını gösterir. Oyun içi ilerleme ağırlıklı olarak oyuncunun aktif katılım göstermesiyle ilişkilendirilir ve kayıp olma olasılığını azaltır. Karakter seviyesi genellikle oyun içinde ilerlemenin bir göstergesidir. Oyuncular karakterlerini yükseltmek için zaman ve çaba harcarlar ve bu süreçte oyunu daha derinlemesine deneyimler. Seviye atlamak, oyunculara daha fazla içerik, beceri ve ödüllerin kilidini açma fırsatı verir, bu da uzun vadeli motivasyonlarını artırır. MMORPG'ler ağırlıklı olarak çok oyunculu etkileşimlerin önemli olduğu oyunlardır ve daha yüksek seviyede olan oyuncular, düşük seviyedeki oyuncularla uyum sağlayabilir bununla birlikte daha zorlu görevleri gerçekleştirebilirler. Bu bakımdan, yüksek seviyeli oyuncu ile düşük seviyeli oyuncu bir arada tutarak oyunu oynatmanın kritik noktalarından biridir. Bu doğrultuda, karakterin seviye atlaması genellikle oyuncunun oyun içindeki katılımını artıracığından dolayı, şirketler seviye atlama konusunda oyuna yeni başlayan grup için özel birtakım içerikler hazırlayarak onları oyunda süreklilik sürecine dahil edebilir.
- 2222 koduna karşılık gelen “oyuncunun depodan ürün alması (7.0)”, oyuncunun oyun içi ekonomik etkileşimlerde bulunduğunu gösterir. Oyun içi görevleri bakımından, oyuncunun oyun içinde aktif kalmasını sağlar ve kayıp olma olasılığını düşürür.

Depodan ürün almak, genellikle oyuncunun oyun içinde aktif olduğunu ve kaynakları etkin bir şekilde kullanabildiğini, oyuncunun oyun içi ekonomiyle etkileşim halinde olduğunu ve oyunda ilerleme için kaynakları yönettiğine dair çıkarım yapılabilir. Yine bu doğrultuda, oyuncular depodan ürün alırken diğer oyuncularla etkileşime geçebilirler. Bu, oyun içi topluluklar oluşturmayı teşvik edebilir ve oyuncuların oyunu terk etme olasılığını azaltabilir. Sonuç olarak, oyuncu oyunda sürekli görev gerçekleştirmesi oyunda daha aktif oyuncu olma eğilimini artırır. Bu doğrultuda, oyuncunun depo ile iletişim kurması oyunda elde ettiği değerli eşyaları burada saklamasına da katkı sağlar. Bu kısımda, belirli görevleri gerçekleştiren oyunculara, depoda fazla eşya tutma hakkı verilebilir. Belirli koleksiyonlar veya eşya setleri oluşturarak oyuncuları depodan ürün almaya teşvik edebilir.

- 2205 koduna karşılık gelen “oyuncunun ürünü tekrar satın alması (7.0)”, genellikle oyuncunun o eşyadan memnun olduğunu ve oyun içindeki deneyiminden tatmin olduğunu gösterir. Tatmin olan oyuncuların ise oyunu terk etme eğilimi daha düşüktür. Aynı zamanda oyunda bir eşyayı tekrar satın almak, oyun içi ekonomiyle etkileşimde olduğunu ve oyun içinde ilerlemesini veya başarısını artırmak için kaynakları etkin bir şekilde kullandığı yorumu da yapılabilir. Özellikle belirli eşyaların daha yüksek tekrar satın alma oranlarına sahip olması, bu eşyaların oyuncular için özel bir öneme sahip olduğunu da gösterebilir. Oyuncunun kendine özel eşya oluşturma sürecinde aynı eşyayı satın alma durumunda, oyuncu için o eşyanın sağlanması oranı istatistiksel olarak artırılabilir. Aynı görevi birden fazla gerçekleştiren oyuncuların bu eşyaya sahip olma ihtimalleri artırılırsa, oyuna olan katılımı daha da artar denilebilir.

LengBear veri seti için sonuçlar ve öneriler;

Mobil oyun endüstrisinde müşteri kaybını analiz etmek ve önlemek, oyun geliştiricileri için stratejik bir öncelik haline gelmiştir. Geliştirilen mobil oyunların çoğu ücretsiz olarak sunulmakta ve gelir, oyuncuların oyun içinde harcadıkları zaman ve gerçek para harcamalarıyla elde edilmektedir. Dolayısıyla, oyuncuların oyunu terk etmesi veya aktifliğin azalması, hem kısa vadeli gelir kaybına yol açar hem de uzun vadede oyunun başarısını olumsuz etkileyebilir. Tez kapsamında LengBear veri setinde oyunun yapısı, içerdiği mini oyun süreleri, oyuncu katılım seviyeleri, oyunda gerçekleşen para harcama oranları ve galibiyet-mağlubiyet oranları

dahil olmak üzere müşteri kaybına katkıda bulunan çeşitli faktörlerin incelenmesini içermektedir.

Aynı şekilde mobil oyunlarda müşteri kaybını analiz ederken kayıp döneminin süresi, oyunun yapısı, oyuncuların ortalama oyun süresi veya katılım düzeyi ve oyun güncellemelerinin veya içerik sürümlerinin sıklığı dahil olmak üzere çeşitli faktörlere bağlı olarak değişebilir. Oyunun platformlarına ve cihazlarına göre farklılık gösterebileceğinden, kayıp döneminin gün kriterinin standart bir süresini olmadığını belirtmekte fayda vardır. Mobil oyunda müşteri kayıp süresi kriteri göz önünde bulundurulduğunda, LengBear veriseti için 4, 5 ve 6 günlük performans sonuçlarına yer verilmiştir. Sonuçlara göre, 5 gün kayıp süresi baz alındığında müşteri kayıp analizinde daha etkili sonuçlar ortaya konduğu görülmektedir (Shi ve diğ., 2017; Gao ve Bai, 2014; López-Martínez ve diğ., 2018). Mobil oyun alanında müşteri kaybı analizinde kritik bir faktör olarak kayıp süresinin dikkate alınmasının önemini vurgulamaktadır. Oyun geliştiricileri, 5 günlük bir müşteri kaybı dönemine odaklanarak, etkili elde tutma stratejileri geliştirmek ve uzun vadeli oyuncu bağlılığını teşvik etmek için eyleme geçirilebilir iç görülerden yararlanabilirler. Ayrıca, Lojistik Regresyon gibi gelişmiş sınıflandırma algoritmalarının benimsenmesi, verimli kayıp tahmin modellerinin geliştirilmesini kolaylaştırarak, pro-aktif kayıp yönetimi ve rekabetin yüksek olduğu mobil oyun pazarında sürdürülebilir oyuncu sadakatinin önünü açmaktadır.

Ayrıca, 5 günlük müşteri kaybı dönemine dayalı olarak müşteri kaybını tahmin etmedeki etkinliklerini değerlendirmek için çeşitli sınıflandırma algoritmalarının karşılaştırmalı bir analizini gerçekleştirdik. Özellikle hem Rastgele Orman hem de Lojistik Regresyon algoritmaları, yüksek doğruluk oranları ve sağlam tahmin yetenekleri sergileyerek sınıflandırma görevinde övgüye değer bir performans göstermiştir. Bununla birlikte, hesaplama verimliliği ve işlem hızı açısından Lojistik Regresyon diğer algoritmalarından daha iyi performans göstermiş ve bu da onu gerçek zamanlı kayıp tahmini ve karar verme için özellikle uygun hale getirmiştir. Bu doğrultuda, kayıp olasılığı yüksek olan oyuncuların listesi elde edilebilir ve bu oyunculara oyunda kalma sürelerini artırmak için birçok şekilde yaklaşım oluşturulabilir.

- Şirket kullanıcıları kayıp ihtimallerine göre birden fazla gruba ayırabilir. Örneğin: oyun içi kayıp başarı yüzdelere göre %0-20, %21-40, %41-60, %61-80 ve geri kalanlar olarak nitelendirilebilir.

- Şirket, kullanıcıların belirli promosyon paketlerine verdikleri tepkileri kontrol etmek ve şirket lehine işe yarayanları seçmek için müşteri grupları üzerinde A/B testi uygulayabilir.
- Oyun içi kazanma ve kaybetme oranına sahip kullanıcıların aynı oyunda yer almasını sağlayabilir bu şekilde kullanıcılar oyunda daha fazla süre geçirmiş olacak, böylece aynı becerilere sahip kişilerle zaman geçirme ihtimali artırabilecektir. Bu doğrultuda, oyunun yüksek puanlı oyuncular tarafından sürekli baskın hale getirilmesine de engel alınabilir.



KAYNAKLAR

- Abdi, H., Williams, L. J. (2010). Principal component analysis: Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(4), 433-459. <https://doi.org/10.1002/wics.101>
- Abt, C. C. (1970). *Serious game*. Viking Press.
- Ackoff, R. (1989). From Data to Wisdom. *Journal of Applied Systems Analysis*, 16, 3-9.
- Adams, E. (2014). *Fundamentals of game design.*, New Riders, London.
- Ahn, J., Hwang, J., Kim, D., Choi, H., Kang, S. (2020). A survey on churn analysis in various business domains. *IEEE Access*, 8 (1).
- Akbaş, A., Turhal, U., Babur, S., Avci, C. (2013). Performance Improvement with Combining Multiple Approaches to Diagnosis of Thyroid Cancer. *Engineering*, 5(10), 264-267. <https://doi.org/10.4236/eng.2013.510B055>
- Aloupis, G., Demaine, E. D., Guo, A., and Viglietta, G. (2015). Classic Nintendo games are (computationally) hard. *Fun with Algorithms*, 586, 135-160. <https://doi.org/10.1016/j.tcs.2015.02.037>
- Alpaydin, E. (2016). *Machine learning: The new AI*. MIT Press.
- Alpaydin, E. (2020). *Introduction to machine learning (Fourth edition)*. The MIT Press.
- Altman, N. S. (1992). An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression. *The American Statistician*, 46(3), 175-185. <https://doi.org/10.2307/2685209>
- Altunkaynak, B. (2019). *Veri madenciliği yöntemleri ve R uygulamaları* (2. bs). Seçkin Yayıncılık.
- American Marketing Association. (2017). What is Marketing? —The Definition of Marketing—AMA. <https://www.ama.org/the-definition-of-marketing-what-is-marketing/> [Ziyaret Tarihi: 16 Kasım 2020]
- Anderson, C. (2009). *Free: The future of a radical price* (1st ed). Hyperion.
- Anderson, T., Galley, S. (2004). The History of Zork. New Zork Times. <https://samizdat.co/shelf/documents/2004/05.27-historyOfZork/historyOfZork.pdf> [Ziyaret Tarihi: 5 Ekim 2020]
- Ang, C. S., Zaphiris, P. (2008). *Computer games and language learning*, Handbook of Research on Instructional Systems and Technology (T. T. Kidd H. Song, Ed.; 1. bs). IGI Global. <https://doi.org/10.4018/978-1-59904-865-9>
- Ant, M. (2003). *Oyun ve bugün*, Yapı Kredi Yayınları, İstanbul.
- Apperley, T. H. (2006). Genre and game studies: Toward a critical approach to video game genres. *Simulation Gaming*, 37(1), Art. 1. <https://doi.org/10.1177/1046878105282278>
- Arf, C. (1959). Makineler Düşünebilir mi ve Nasıl Düşünebilir? *Üniversite Çalışmalarını Muhite Yayma ve Halk Eğitimi Yayınları Konferanslar Serisi* 1, 1, 91-106.
- Arık, K., Gezer, M., Tayali, S. T. (2022). Bibliometric Analysis of Scientific Studies Published on Game Customer Churn Analysis Between 2008 and 2022. *Journal of Politics*, 5(1), 21.
- Arjoranta, J. (2019). How to Define Games and Why We Need, *The Computer Games Journal*, 8(3-4), Art. 3-4. <https://doi.org/10.1007/s40869-019-00080-6>
- artsandculture.google.com. (1983). Arcade game:Spy Hunter arcade game—Bally Midway. Google Arts Culture. <https://artsandculture.google.com/asset/arcade-game-spy-hunter-arcade-game-bally-midway/UAE6iGexnBy9gQ> [Ziyaret Tarihi: 7 Ocak 2021]
- Avedon, E. M., Sutton-Smith, B. (2015). *The study of games*, Ishi Press, USA.

- Awad, M., Khanna, R. (2015). *Machine Learning and Knowledge Discovery*. In Efficient Learning Machines (M. Awad R. Khanna), Apress, 19-38. https://doi.org/10.1007/978-1-4302-5990-9_2
- Baer, R. (1973). *Television gaming and training apparatus* (United States Patent Sy US3728480A). <https://patents.google.com/patent/US3728480A/en>
- Barnes, F. (1994). *Donkey Kong*. New Republic, 211(20), Art. 20. <https://www.elibrary.ru/item.asp?id=2201393>
- Barr, P., Noble, J., Biddle, R. (2007). Video game values: Human–computer interaction and games. *Interacting with Computers*, 19(2), Art. 2. <https://doi.org/10.1016/j.intcom.2006.08.008>
- Baydemir, M. B. (2014). *Lojistik regresyon analizi üzerine bir inceleme*, Yüksek Lisans, İnönü Üniversitesi. <https://tez.yok.gov.tr/UlusalTezMerkezi/tezSorguSonucYeni.jsp>
- Bellinger, G., Castro, D., Mills, A. (2004). *Data, information, knowledge, and wisdom*. <https://homepages.dcc.ufmg.br/~amendes/SistemasInformacaoTP/TextosBasicos/Data-Information-Knowledge.pdf>
- Bergstrom, K. (2019). Moving Beyond Churn: Barriers and Constraints to Playing a Social Network Game. *Games and Culture*, 14(2), 170-189. <https://doi.org/10.1177/1555412018791697>
- Bertens, P., Guitart, A., Chen, P. P., Perianez, A. (2018). A Machine-Learning Item Recommendation System for Video Games. 2018 *IEEE Conference on Computational Intelligence and Games (CIG)*, 1-4. <https://doi.org/10.1109/CIG.2018.8490456>
- Bertens, P., Guitart, A., Periañez, Á. (2017). Games and Big Data: A Scalable Multi-Dimensional Churn Prediction Model. 2017 *IEEE Conference on Computational Intelligence and Games (CIG)*, 33-36. <https://doi.org/10.1109/CIG.2017.8080412>
- Beuscher, D. (2014). *Game Boy—Overview—Allgame*. www.allgame.com. <https://web.archive.org/web/20141212112812/http://www.allgame.com/platform.php?id=20> [Ziyaret Tarihi: 24 Aralık 2021]
- Beyer, H. R., Holtzblatt, K. (1995). Apprenticing with the customer. *Communications of the ACM*, 38(5), 45-52.
- BİÇER, E. C. (2021). *Churn prediction in hyper-casual games*, Yüksek Lisans, Bahçeşehir Üniversitesi.
- Binark, M., Sütçü, G. B., Fidaner, I. B. (2009). *Dijital Oyun Rehberi: Oyun Tasarımı, Türler ve Oyuncu*. Kalkedon Yayınları, İstanbul.
- Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.
- Blazek, P. J. (2022). *Why we will never open deep learning's black box*. Medium. <https://towardsdatascience.com/why-we-will-never-open-deep-learning-black-box-4c27cd335118> [Ziyaret Tarihi: 2 Mart 2019]
- Blizzard. (1994). *Warcraft: Orcs Humans—WoW* [Resmi Web Sitesi]. <https://worldofwarcraft.com/en-us/story/timeline> [Ziyaret Tarihi: 9 Şubat 2019]
- Bonometti, V., Ringer, C., Hall, M., Wade, A. and Drachen, A. (2019). Modelling Early User-Game Interactions for Joint Estimation of Survival Time and Churn Probability, 2019 *IEEE Conferences on Games*, (p-8). <https://doi.org/10.1109/CIG.2019.8848038>
- Borbora, Z. H., Srivastava, J. (2012). User Behavior Modelling Approach for Churn Prediction in Online Games. 2012 International Conference on Privacy, Security, Risk and Trust and 2012 *International Confernece on Social Computing*, 51-60. <https://doi.org/10.1109/SocialCom-PASSAT.2012.84>
- Borbora, Z., Srivastava, J., Hsu, K.-W., Williams, D. (2011). Churn Prediction in MMORPGs Using Player Motivation Theories and an Ensemble Approach. 2011 *IEEE Third Int'l*

- Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third Int'l Conference on Social Computing*, 157-164. <https://doi.org/10.1109/PASSAT/SocialCom.2011.122>
- Borle, S., Singh, S. S., Jain, D. C. (2008). Customer Lifetime Value Measurement, *Management Science*, 54(1), 100-112. <https://doi.org/10.1287/mnsc.1070.0746>
- Bottou, L. (2012). *Stochastic Gradient Descent Tricks*. In G. Montavon, G. B. Orr, K.-R. Müller (Ed.), *Neural Networks: Tricks of the Trade* (C. 7700, ss. 421-436). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-35289-8_25
- Breiman, L. (1994). Bagging predictors. *Machine Learning*, 24(2), 123-140. <https://doi.org/10.1007/BF00058655>
- Breiman, L. (2000). Bias, Variance , And Arcing Classifiers. *Technical Report 460*, Statistics Department, University of California.
- Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1), 5-32. <https://doi.org/10.1023/A:1010933404324>
- Brookhaven National Lab. (2008). BNL | *History: The First Video Game?* [Ulusal Laboratuvar]. Brookhaven National Lab. <https://www.bnl.gov/about/history/firstvideo.php> [Ziyaret Tarihi: 9 Şubat 2019]
- Brownlee, J. (2021). *Machine Learning Mastery With Python: Understand Your Data, Create Accurate Models and Work Projects End-To-End*, Public Publishing.
- Burez, J., Van den Poel, D. (2009). Handling class imbalance in customer churn prediction. *Expert Systems with Applications*, 36(3), 4626-4636. <https://doi.org/10.1016/j.eswa.2008.05.027>
- Burkov, A. (2019). *The hundred-page machine learning book*, Public Publishing
- Buttle, F. Francis. (2006). *Customer relationship management: Concepts and tools*, Köln, Germany.
- Cailliois, R., Barash, M. (2001). *Man, play, and games*. University of Illinois Press, USA.
- Calders, T., Custers, B. (2013). *What is data mining and how does it work?* In B. Custers, T. Calders, B. Schermer, T. Zarsky (Ed.), *Discrimination and Privacy in the Information Society* (C. 3, ss. 27-42). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-30487-3_2
- Canning, G. (1982). Do a value analysis of your customer base, *Industrial Marketing Management*, 11(2), 89-93. [https://doi.org/10.1016/0019-8501\(82\)90001-3](https://doi.org/10.1016/0019-8501(82)90001-3)
- Carr, D. (2006). *Computer games: Text, narrative, and play*. Polity, USA.
- Casas, S., Portalés, C., García-Pereira, I., Fernández, M. (2017). On a First Evaluation of ROMOT—A RObotic 3D MOvie Theatre—For Driving Safety Awareness. *Multimodal Technologies and Interaction*, 1(2), Art. 2. <https://doi.org/10.3390/mti1020006>
- Castéran, H., Meyer-Waarden, L., Reinartz, W. (2017). *Modeling Customer Lifetime Value, Retention, and Churn*. In C. Homburg, M. Klarmann, A. Vomberg (Ed.), *Handbook of Market Research* (ss. 1-33). Springer International Publishing. https://doi.org/10.1007/978-3-319-05542-8_21-1
- Castro, E. G., Tsuzuki, M. S. G. (2015). Churn Prediction in Online Games Using Players' Login Records: A Frequency Analysis Approach. *IEEE Transactions on Computational Intelligence and AI in Games*, 7(3), 255-265. <https://doi.org/10.1109/TCIAIG.2015.2401979>
- Centor, R. M. (1991). Signal Detectability: The Use of ROC Curves and Their Analyses, *Medical Decision Making*, 11(2), 102-106. <https://doi.org/10.1177/0272989X9101100205>

- Chang, R., Chen, S., Lin, H., Yu, H. (2010). DUIRA: An interactive learning platform for mixed reality. *2010 IEEE International Conference on Multimedia and Expo*, 1152-1153. <https://doi.org/10.1109/ICME.2010.5583212>
- Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T. P., Shearer, C., Wirth, R. (2000). *CRISP-DM 1.0: Step-by-step data mining guide*.
- Charles, D., Cowley, B. U. (2020). *Behavlet Analytics for Player Profiling and Churn Prediction*. In C. Stephanidis, D. Harris, W.-C. Li, D. D. Schmorow, C. M. Fidopiastis, P. Zaphiris, A. Ioannou, X. Fang, R. A. Sottolare, J. Schwarz (Ed.), *HCI International 2020 – Late Breaking Papers: Cognition, Learning and Games* (C. 12425, ss. 631-643). Springer International Publishing. https://doi.org/10.1007/978-3-030-60128-7_46
- Chatterjee, S., Hadi, A. S. (2012). *Regression analysis by example (Fifth edition)*. Wiley.
- Chen, P. P., Guitart, A., del Rio, A. F., Perianez, A. (2018). Customer Lifetime Value in Video Games Using Deep Learning and Parametric Models. *2018 IEEE International Conference on Big Data (Big Data)*, 2134-2140. <https://doi.org/10.1109/BigData.2018.8622151>
- Chen, T., Guestrin, C. (2016). *XGBoost: A Scalable Tree Boosting System*. <https://doi.org/10.48550/ARXIV.1603.02754>
- Chikhani, R. (2015). The History Of Gaming: An Evolving Community. TechCrunch. <https://social.techcrunch.com/2015/10/31/the-history-of-gaming-an-evolving-community/>
- Chollet, F. (2021). *Deep learning with Python (Second edition)*. Manning Publications.
- Chomboon, K., Chujai, P., Teerarassammee, P., Kerdprasop, K., Kerdprasop, N. (2015). An Empirical Study of Distance Metrics for k-Nearest Neighbor Algorithm. *The Proceedings of the 2nd International Conference on Industrial Application Engineering 2015*, 280-285. <https://doi.org/10.12792/iciae2015.051>
- Clark, N., Scott, P. S. (2009). *Game addiction: The experience and the effects*. McFarland.
- Clarke, B., Fokoue, E., Zhang, H. H. (2009). *Principles and Theory for Data Mining and Machine Learning*. Springer New York. <https://doi.org/10.1007/978-0-387-98135-2>
- Cohen, D. S. (2019). *OXO aka Noughts and Crosses—The First Video Game*. Lifewire. <https://www.lifewire.com/oxo-aka-noughts-and-crosses-729624>
- Computer Hope. (2018). Game Definition Language. <http://games.stanford.edu/games/gdl.html>
- Cortes, C., Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3), 273-297. <https://doi.org/10.1007/BF00994018>
- Costikyan, G. (2002). I Have No Words I Must Design: Toward a Critical Vocabulary for Games. 25.
- Cover, T., Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1), 21-27. <https://doi.org/10.1109/TIT.1967.1053964>
- Cramer, J. S. (2003). The Origins of Logistic Regression. *SSRN Electronic Journal*. <https://doi.org/10.2139/ssrn.360300>
- Crawford, C. (2003). *Chris Crawford on game design*. New Riders.
- Cruz-Neira, C., Fernández, M., Portalés, C. (2018). Virtual Reality and Games. *Multimodal Technologies and Interaction*, 2(1), Art. 1. <https://doi.org/10.3390/mti2010008>
- Çelik, U., Akçetin, E., Gök, M. (2017). *Rapidminer ile Uygulamalı Veri Madenciliği* (1. bs). Pusula Yayıncılık.
- Demchenko, Y., De Laat, C., Membrey, P. (2014). Defining architecture components of the Big Data Ecosystem. *2014 International conference on collaboration technologies and systems (CTS)*, 104-112.

- Demediuk, S., Murrin, A., Bulger, D., Hitchens, M., Drachen, A., Raffae, W. L., Tamassia, M. (2018). Player retention in league of legends: A study using survival analysis. *Proceedings of the Australasian Computer Science Week Multiconference*, 1-9. <https://doi.org/10.1145/3167918.3167937>
- DeWinter, J. (2015). *Shigeru Miyamoto: Super Mario Bros., Donkey Kong, the Legend of Zelda*. Bloomsbury Academic.
- Ding, J., Gao, D., Chen, X. (2015). Alone in the Game: Dynamic Spread of Churn Behavior in a Large Social Network a Longitudinal Study in MMORPG. *International Journal of Smart Home*, 9(3), 35-44.
- Ding, Z., Huang, Y., Yuan, H., Dong, H. (2020). *Introduction to Reinforcement Learning*. In H. Dong, Z. Ding, S. Zhang (Ed.), *Deep Reinforcement Learning* (ss. 47-123). Springer Singapore. https://doi.org/10.1007/978-981-15-4095-0_2
- Drachen, A., Green, J., Gray, C., Harik, E., Lu, P., Sifa, R., Klabjan, D. (2016). Guns and guardians: Comparative cluster analysis and behavioral profiling in destiny. *2016 IEEE Conference on Computational Intelligence and Games (CIG)*, 1-8. <https://doi.org/10.1109/CIG.2016.7860423>
- Drachen, A., Lundquist, E. T., Kung, Y., Rao, P. S., Klabjan, D., Sifa, R., Runge, J. (2016). Rapid Prediction of Player Retention in Free-to-Play Mobile Games. <https://doi.org/10.48550/ARXIV.1607.03202>
- Drachen, A., Pastor, M., Liu, A., Fontaine, D. J., Chang, Y., Runge, J., Sifa, R., Klabjan, D. (2018). To be or not to be...social: Incorporating simple social features in mobile game customer lifetime value predictions. *Proceedings of the Australasian Computer Science Week Multiconference*, 1-10. <https://doi.org/10.1145/3167918.3167925>
- Efe, M. O., Abadoglu, E., Kaynak, O. (1999). A novel analysis and design of a neural network assisted nonlinear controller for a bioreactor. *International Journal of Robust and Nonlinear Control*, 9(11), 799-815. [https://doi.org/10.1002/\(SICI\)1099-1239\(199909\)9:11<799::AID-RNC441>3.0.CO;2-U](https://doi.org/10.1002/(SICI)1099-1239(199909)9:11<799::AID-RNC441>3.0.CO;2-U)
- Efe, M. O., Kaynak, O. (1999). A comparative study of neural network structures in identification of nonlinear systems. *Mechatronics*, 9(3), 287-300. [https://doi.org/10.1016/S0957-4158\(98\)00047-6](https://doi.org/10.1016/S0957-4158(98)00047-6)
- El-Nasr, M. S. (2013). *Game analytics: Maximizing the value of player data*. Springer.
- Encyclopædia Britannica. (2023). *Artificial intelligence* | Definition, Examples, Types, Applications, Companies, Facts | Britannica [Ansiklopedi]. Britannica. <https://www.britannica.com/technology/artificial-intelligence> [Ziyaret Tarihi: 20 Ekim 2021]
- Evers, E., van de Ven, N., Weeda, D. (2015). The Hidden Cost of Microtransactions: Buying In-Game Advantages in Online Games Decreases a Player's Status, *International Journal of Internet Science*, 10.
- Fatma Önay Koçoğlu. (2017). *Müşteri Kayıp Analizi Probleminin Çözümünde Analitik Yaklaşımlar* [Doktora]. İstanbul Üniversitesi Fen Bilimleri Enstitüsü.
- Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8), 861-874. <https://doi.org/10.1016/j.patrec.2005.10.010>
- Fayyad, U. (2001). *Knowledge Discovery in Databases: An Overview*. In S. Džeroski N. Lavrač (Ed.), *Relational Data Mining* (ss. 28-47). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-662-04599-2_2
- Fayyad, U. (2005). Knowledge discovery in databases: An overview. *Inductive Logic Programming: 7th International Workshop, ILP-97 Prague, Czech Republic September 17-20, 1997 Proceedings*, 1-16.

- Fetzer, J. H. (1990). *What is Artificial Intelligence?* In *Artificial Intelligence: Its Scope and Limits* (C. 4, ss. 3-27). Springer Netherlands. https://doi.org/10.1007/978-94-009-1900-6_1
- Fix, E., Hodges, J. (1951). *Discriminatory Analysis, Nonparametric Discrimination: Consistency Properties*. [Teknik Rapor]. Randolph Field.
- Fix, E., Hodges, J. L. (1989). Discriminatory Analysis. Nonparametric Discrimination: Consistency Properties. *International Statistical Review / Revue Internationale de Statistique*, 57(3), 238-247. <https://doi.org/10.2307/1403797>
- Fjelland, R. (2020). Why general artificial intelligence will not be realized. *Humanities and Social Sciences Communications*, 7(1), 1-9.
- Forhad, N., Hussain, Md. S., Rahman, R. M. (2014). Churn analysis: Predicting churners. *Ninth International Conference on Digital Information Management (ICDIM 2014)*, 237-241. <https://doi.org/10.1109/ICDIM.2014.6991433>
- Frasca, G. (2001). *Videogames of the Oppressed: Videogames as a Means for Critical Thinking and Debate* [PhD Thesis]. Georgia Institute of Technology.
- Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of statistics*, 1189-1232.
- Galehantomo, G. (2015). Platform Comparison Between Games Console, Mobile Games And PC Games. *SISFORMA*, 2, 23. <https://doi.org/10.24167/sisforma.v2i1.407>
- Gallagher, M., Ryan, A. (2003). The 2003 Congress on Evolutionary Computation, CEC 2003: Proceedings: Canberra, Australia, 8-12 December, *Congress on Evolutionary Computation*, Piscataway, NJ. IEEE.
- Gamble, J. E. (2008). *Competition in Video Game Consoles: Sony, Microsoft, and Nintendo Battle for Supremacy*. In *Strategic Management* (ss. 493-507). McGraw Hill.
- GaminginTurkey. (2019). *Türkiye Oyun Sektörü 2019 Raporu* (s. 138). Gaming in Turkey. <https://www.gaminginturkey.com/turkiye-oyun-sektoru-raporu-2019.pdf>
- Garavaglia, S., Sharma, A. (1998). A smart guide to dummy variables: Four applications and a macro. *Proceedings of the northeast SAS users group conference*, 43.
- García, S., Luengo, J., Herrera, F. (2015). *Data Preparation Basic Models*. In S. García, J. Luengo, F. Herrera (Ed.), *Data Preprocessing in Data Mining* (ss. 39-57). Springer International Publishing. https://doi.org/10.1007/978-3-319-10247-4_3
- Gashler, M., Giraud-Carrier, C., Martinez, T. (2008). Decision tree ensemble: Small heterogeneous is better than large homogeneous. *2008 Seventh international conference on machine learning and applications*, 900-905.
- Géron, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems (Second edition)*. O'Reilly Media, Inc.
- Giudici, P., Figini, S. (2009). *Applied data mining for business and industry* (2nd ed). Wiley.
- Gold, C., Tzuo, T. (2020). Fighting churn with data: The science and strategy of customer retention. Manning.
- Goodfellow, I., Bengio, Y., Courville, A. (2016). *Deep learning*. The MIT Press.
- Gregersen, A. (2011). Genre, technology and embodied interaction: The evolution of digital game genres and motion gaming. *MedieKultur Journal of Media and Communication Research*, 51, 94-109.
- Grüsser, S. M., Thalemann, R., Griffiths, M. D. (2007). Excessive Computer Game Playing: Evidence for Addiction and Aggression? *CyberPsychology Behavior*, 10(2), 290-292. <https://doi.org/10.1089/cpb.2006.9956>

- Guitart, A., Chen, P., Periañez, Á. (2018). The Winning Solution to the IEEE CIG 2017 Game Data Mining Competition. *Machine Learning and Knowledge Extraction*, 1(1), Art. 1. <https://doi.org/10.3390/make1010016>
- Gupta, S., Zeithaml, V. (2006). Customer Metrics and Their Impact on Financial Performance. *Marketing Science*, 25(6), 718-739. <https://doi.org/10.1287/mksc.1060.0221>
- Gupta, U. G., Gupta, A. (2016). Vision: A missing key dimension in the 5V Big Data framework. *Journal of International Business Research and Marketing*, 1(3), 50-56.
- Gülseçen, S. (2014). Bir Değer Olarak Bilgi ve Bilginin Yönetimi, *Türk Kütüphaneciliği*, 28(1), 62-68.
- Gülseçen, S. (2015). *Bilgi Yönetimi Bilgi Türeticileri, Büyük Veri İnovasyon ve Kurumsal Zeka* Papatya Yayıncılık, İstanbul.
- Gürsakil, N. (2021). *Makine Öğrenmesi*, Dora Yayınları, İstanbul.
- Hadden, J., Tiwari, A., Roy, R., Ruta, D. (2007). Computer assisted customer churn management: State-of-the-art and future trends. *Computers Operations Research*, 34(10), 2902-2917. <https://doi.org/10.1016/j.cor.2005.11.007>
- Hadiji, F., Sifa, R., Drachen, A., Thureau, C., Kersting, K., Bauckhage, C. (2014). Predicting player churn in the wild. *2014 IEEE Conference on Computational Intelligence and Games*, 1-8. <https://doi.org/10.1109/CIG.2014.6932876>
- Haldun Akpınar. (2018). *Data—Veri Madenciliği Veri Analizi*, Papatya Bilim Kitabevi.
- Halliburton, R., Pearson, J. (2002). *Solitaire game played over the internet with features to extend play* (United States Patent Sy US20020052229A1). <https://patents.google.com/patent/US20020052229A1/en>
- Hamasuna, Y., Endo, Y., Miyamoto, S. (2008). Support Vector Machine for data with tolerance based on Hard-margin and Soft-Margin. *2008 IEEE International Conference on Fuzzy Systems (IEEE World Congress on Computational Intelligence)*, 750-755.
- Hamzaçebi, C. (2021). *Matlab Uygulamalı Yapay Sinir Ağları*. Seçkin Yayıncılık.
- Han, J., Kamber, M., Pei, J. (2012). *Data mining: Concepts and techniques*, Elsevier.
- Hand, D. J., Mannila, H., Smyth, P. (2001). *Principles of data mining*. MIT Press.
- Hansen, D. (2016). *Game on! Video game history from Pong and Pac-man to Mario, Minecraft, and more*, Feiwei Friends.
- Health IT. (2020). How big is the internet, and how do we measure it? Health IT. <https://healthit.com.au/how-big-is-the-internet-and-how-do-we-measure-it/> [Ziyaret Tarihi: 15 Temmuz 2020]
- Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(6), 417-441. <https://doi.org/10.1037/h0071325>
- Huang, H.-C. (2016). Freemium business model: Construct development and measurement validation, *Internet Research*, 26(3), Art. 3. <https://doi.org/10.1108/IntR-03-2014-0064>
- Huizinga, J. (2013). *Homo Ludens: Onurun Kültür İçindeki Yeri Üzerine Bir İnceleme*, Ayrıntı Yayıncılık.
- IBM. (t.y.). *What is Machine Learning?* | IBM. Geliş tarihi 19 Şubat 2023, gönderen <https://www.ibm.com/topics/machine-learning>
- James, G., Witten, D., Hastie, T., Tibshirani, R. (Ed.). (2013). *An introduction to statistical learning: With applications in R*. Springer.
- Jang, K., Kim, J., Yu, B. (2019). Vector-based Churn Prediction using Neural Networks in Mobile Games. *2019 IEEE International Conference on Big Data (Big Data)*, 6081-6083. <https://doi.org/10.1109/BigData47090.2019.9006052>
- Jeon, J., Yoon, D., Yang, S., Kim, K. (2017). Extracting gamers' cognitive psychological features and improving performance of churn prediction from mobile games. *2017 IEEE*

- Conference on Computational Intelligence and Games (CIG)*, 150-153. <https://doi.org/10.1109/CIG.2017.8080428>
- Jin Ha Lee, Natascha Karlova, Rachel Ivy Clarke, Katherine Thornton, Andrew Perti. (2014, Mart 1). Facet Analysis of Video Game Genres. *ICConference 2014 Proceedings. iConference 2014 Proceedings: Breaking Down Walls. Culture - Context - Computing*. <https://doi.org/10.9776/14057>
- Joiner, I. A. (2018). *Artificial Intelligence*. In *Emerging Library Technologies* (ss. 1-22). Elsevier. <https://doi.org/10.1016/B978-0-08-102253-5.00002-2>
- Jolliffe, I. T. (1986). *Principal component analysis*. Springer New York. <https://doi.org/10.1007/978-1-4757-1904-8>
- Juul, J. (2001). Games telling stories? A brief note on games and narratives. *Game studies*, 1(1), Art. 1.
- Kaplan, A. (2014). *The epic evolution of video games*.
- Karenberg, A. (2013). The world of gods and the body of man: Mythological origins of modern anatomical terms. *Anatomy (International Journal of Experimental and Clinical Anatomy)*, 6-7, 7-22. <https://doi.org/10.2399/ana.11.142>
- Karmakar, B., Liu, P., Mukherjee, G., Che, H., Dutta, S. (2021). Improved retention analysis in freemium role-playing games by jointly modelling players' motivation, progression and churn. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 185. <https://doi.org/10.1111/rssa.12730>
- Kartal, E., Balaban, E. (2015). *Veri madenciliği ve makine öğrenmesi*, Çağlayan Yayınevi.
- Kartal, E., Balaban, M. E. (2018). Machine learning techniques in cardiac risk assessment. *Turkish Journal of Thoracic and Cardiovascular Surgery*, 26(3), Art. 3. <https://doi.org/10.5606/tgkdc.dergisi.2018.15559>
- Kastalli, I. V., Van Looy, B. (2013). Servitization: Disentangling the impact of service business model innovation on manufacturing firm performance. *Journal of Operations Management*, 31(4), Art. 4. <https://doi.org/10.1016/j.jom.2013.02.001>
- Kawale, J., Pal, A., Srivastava, J. (2009). Churn Prediction in MMORPGs: A Social Influence Based Approach. *2009 International Conference on Computational Science and Engineering*, 423-428. <https://doi.org/10.1109/CSE.2009.80>
- Kearns, M. (1988). *Thoughts on Hypothesis Boosting*. University of Pennsylvania. <https://www.cis.upenn.edu/~mkearns/papers/boostnote.pdf>
- Kearns, M., Valiant, L. (1994). Cryptographic limitations on learning Boolean formulae and finite automata. *Journal of the ACM*, 41(1), 67-95. <https://doi.org/10.1145/174644.174647>
- Kelleher, J. D., Tierney, B. (2022). *Veri Bilimi* (3. bs). Tellekt Yayınları.
- Kent, S. L. (2001). *The ultimate history of video games: From Pong to Pokémon and beyond: the story behind the craze that touched our lives and changed the world (1st ed)*. Prima Pub.
- Kılıç, S. (2013). ROC analysis in clinical decision making. *Journal of PBS*, 3(3), 135-140.
- Kilimci, Z. H., Yörük, H., Akyokus, S. (2020). Sentiment Analysis Based Churn Prediction in Mobile Games using Word Embedding Models and Deep Learning Algorithms. *2020 International Conference on INnovations in Intelligent SysTems and Applications (INISTA)*, 1-7. <https://doi.org/10.1109/INISTA49547.2020.9194624>
- Kim, K.-J., Cho, S.-B. (2013). Game AI Competitions: An Open Platform for Computational Intelligence Education [Educational Forum]. *IEEE Computational Intelligence Magazine*, 8(3), Art. 3. <https://doi.org/10.1109/MCI.2013.2264568>

- Kim, S., Choi, D., Lee, E., Rhee, W. (2017). Churn prediction of mobile and online casual games using play log data, *PLOS ONE*, 12(7), <https://doi.org/10.1371/journal.pone.0180735>
- King, L. (2002). *Game on: The history and culture of videogames*. Universe Pub. : Distributed to the U.S. trade by St. Martin's Press.
- Kingsford, C., Salzberg, S. L. (2008). What are decision trees?, *Nature biotechnology*, 26(9), 1011-1013.
- Kiseleva, E. M., Nekrasova, M. L., Mayorova, M. A., Rudenko, M. N., Kankhva, V. S. (2016). The Theory and Practice of Customer Loyalty Management and Customer Focus in the Enterprise Activity, *International Review of Management and Marketing*, 6(S6), 95-103.
- Klein, M. (1984). The Bite of Pac-Man. *The Journal of Psychohistory*, 11(3), Art. 3.
- Kolko, B. E. (2006). *Definitions about digital games*. <http://faculty.washington.edu/bkolko/games/index.shtml>
- Kolster, R. (2013). *Theory of Fun for Game Design*, O'REILLY Publishing, USA.
- Kotler, P., Armstrong, G. (2018). *Principles of marketing (Seventeenth edition)*. Pearson Higher Education.
- Kowert, R., Quandt, T. (Ed.). (2016). *The video game debate: Unravelling the physical, social, and psychological effects of digital games*. Routledge.
- Koza, J. R., Bennett, F. H., Andre, D., Keane, M. A. (1996). *Automated Design of Both the Topology and Sizing of Analog Electrical Circuits Using Genetic Programming*. In J. S. Gero F. Sudweeks (Ed.), *Artificial Intelligence in Design '96* (ss. 151-170). Springer Netherlands. https://doi.org/10.1007/978-94-009-0279-4_9
- Köklü, N., Büyüköztürk, Ş., Çoluk, Ö. (2021). *Sosyal Bilimler için İstatistik* (25. bs). Pegem Akademi Yayıncılık.
- Koroğlu, Y. (2017). *Yapay Zeka 'nın Teorik ve Pratik Sınırları*. Boğaziçi Üniversitesi. <https://www.cmpe.boun.edu.tr/~yavuz.koroglu/publications/EBES17.pdf> [Ziyaret Tarihi: 5 Kasım 2022]
- Kristensen, J. T., Burelli, P. (2019). Combining Sequential and Aggregated Data for Churn Prediction in Casual Freemium Games. *2019 IEEE Conference on Games (CoG)*, 1-8. <https://doi.org/10.1109/CIG.2019.8848106>
- Kristensen, J. T., Guckelsberger, C., Burelli, P., Hämäläinen, P. (2022). Personalized Game Difficulty Prediction Using Factorization Machines. *The 35th Annual ACM Symposium on User Interface Software and Technology*, 1-13. <https://doi.org/10.1145/3526113.3545624>
- Krzywinska, T., Brown, D. (2015). *Online Games and Genre*. In P. H. Ang R. Mansell (Ed.), *The International Encyclopedia of Digital Communication and Society*, ss. 1-4. Wiley.
- Kulkarni, A., Patil, A., Patil, M., Bhoite, S. (2019). Customer Churn Analysis and Prediction. *International Journal of Computer Applications Technology and Research*, 8(9), 363-366. <https://doi.org/10.7753/IJCATR0809.1005>
- Kurtz, B. (2004). *The encyclopedia of arcade video games*. Schiffer Pub.
- Kwon, H., Jeong, W., Kim, D.-W., Yang, S.-I. (2018). Clustering Player Behavioral Data and Improving Performance of Churn Prediction from Mobile Game. *2018 International Conference on Information and Communication Technology Convergence (ICTC)*, 1252-1254. <https://doi.org/10.1109/ICTC.2018.8539514>
- Larose, D. T., Larose, C. D. (2014). *Discovering Knowledge in Data: An Introduction to Data Mining*. John Wiley Sons, Inc. <https://doi.org/10.1002/9781118874059>
- LaValley, M. P. (2008). Logistic Regression, *Circulation*, 117(18), 2395-2399. <https://doi.org/10.1161/Circulationaha.106.682658>

- Lee, E., Jang, Y., Yoon, D.-M., Jeon, J., Yang, S., Lee, S.-K., Kim, D.-W., Chen, P. P., Guitart, A., Bertens, P., Perianez, A., Hadiji, F., Muller, M., Joo, Y., Lee, J., Hwang, I., Kim, K.-J. (2019). Game Data Mining Competition on Churn Prediction and Survival Analysis Using Commercial Game Log Data. *IEEE Transactions on Games*, 11(3), 215-226. <https://doi.org/10.1109/TG.2018.2888863>
- Lee, E., Kim, B., Kang, S., Kang, B., Jang, Y., Kim, H. K. (2020). Profit Optimizing Churn Prediction for Long-Term Loyal Customers in Online Games. *IEEE Transactions on Games*, 12(1), 41-53. <https://doi.org/10.1109/TG.2018.2871215>
- Lejeune, M. A. P. M. (2001). Measuring the impact of data mining on churn management. *Internet Research*, 11(5), 375-387. <https://doi.org/10.1108/10662240110410183>
- Lessard, J. (2015). *Early Computer Game Genre Preferences (1980-1984)*. 2015 Bildiriler Kitabı, 12.
- Levent, G. C. (2022). *Determining player lifetime value in mobile puzzle game using machine learning techniques*, Yüksek Lisans. Bahçeşehir Üniversitesi.
- Lewis, C., Wardrip-Fruin, N. (2010). Mining game statistics from web services: A World of Warcraft armory case study. *Proceedings of the Fifth International Conference on the Foundations of Digital Games - FDG '10*, 100-107.
- Liashchynskiy, P., Liashchynskiy, P. (2019). Grid search, random search, genetic algorithm: A big comparison for NAS. *arXiv preprint arXiv:1912.06059*.
- Liu, D.-R., Shih, Y.-Y. (2005). Integrating AHP and data mining for product recommendation based on customer lifetime value. *Information Management*, 42(3), 387-400. <https://doi.org/10.1016/j.im.2004.01.008>
- Loh, W.-Y. (2014). Fifty years of classification and regression trees: fifty years of classification and regression trees. *International Statistical Review*, 82(3), 329-348. <https://doi.org/10.1111/insr.12016>
- Loria, E., Marconi, A. (2021). Exploiting limited players' behavioral data to predict churn in gamification. *Electronic Commerce Research and Applications*, 47, 101057. <https://doi.org/10.1016/j.elerap.2021.101057>
- Lovato, N. (2015). *16 Reasons Why Players Are Leaving Your Game*. GameAnalytics. <https://gameanalytics.com/blog/16-reasons-players-leaving-game/> [Ziyaret Tarihi : 1 Mart 2018]
- Lowood, H. (2009). Videogames in Computer Space: The Complex History of Pong. *IEEE Annals of the History of Computing*, 31(3), Art. 3.
- Lyons, K., Messinger, P. R., Niu, R. H., Stroulia, E. (2012). A tale of two pricing systems for services. *Information Systems and E-Business Management*, 10(1), 19-42. <https://doi.org/10.1007/s10257-010-0151-3>
- Manjunatha, K., Shivalingaiah, D. (2004). Customer's Perception of Service Quality in Libraries. *Annals of Library and Information Studies*, 51(4), Art. 4.
- Maroney, K. (2001). My Entire Waking Life [Kişisel Web Sitesi]. *thegamesjournal.com*. <http://www.thegamesjournal.com/articles/MyEntireWakingLife.shtml>
- Martins Kummer, L. B., Cesar Nievola, J., Paraiso, E. C. (2018). Applying Commitment to Churn and Remaining Players Lifetime Prediction. *2018 IEEE Conference on Computational Intelligence and Games (CIG)*, 1-8.
- Massaro, A., Vitti, V., Lisco, P., Galiano, A., Savino, N. (2019). A business intelligence platform Implemented in a big data system embedding data mining: A case of study. *International Journal of Data Mining Knowledge Management Process (IJDMP)*, 9(1), 1-20.

- McCarthy, J. (1989). *What is AI? / Basic Questions*. <http://jmc.stanford.edu/>, [Ziyaret Tarihi: 11 Şubat 2020]
- Merriam-Webster Sözlüğü. (2020). *Definition of Digital Game*. Merriam-Webster. <https://www.merriam-webster.com/dictionary/video+game>
- Metlek, S., Kayaalp, K. (2020). *Makine öğrenmesinde, teoriden örnek matlab uygulamalarına kadar destek vektör makineleri* (1. bs). İksad Yayınevi.
- Meyer, C., Schwager, A. (2007). Understanding customer experience. *Harvard business review*, 85(2), 116.
- Milošević, M., Živić, N., Andjelković, I. (2017). Early churn prediction with personalized targeting in mobile social games. *Expert Systems with Applications*, 83, 326-332. <https://doi.org/10.1016/j.eswa.2017.04.056>
- Mitchell, V.-W., Walsh, G., Yamin, M. (2005). Towards a conceptual model of consumer confusion. *ACR North American Advances.*, pp. 37-49.
- Mohamed, W. N. H. W., Salleh, M. N. M., Omar, A. H. (2012). A comparative study of reduced error pruning method in decision tree algorithms. *2012 IEEE International conference on control system, computing and engineering*, 392-397.
- Molnar, C. (2022). *Interpretable machine learning: A guide for making black box models explainable (Second edition)*. Christoph Molnar.
- Moore, D. H. (1987). *Classification and regression trees*, Leo Breiman, Jerome H. Friedman, Richard A. Olshen, and Charles J. Stone. Brooks/Cole Publishing, Monterey, 1984, 358 pages, \$27.95. *Cytometry*, 8(5), 534-535. <https://doi.org/10.1002/cyto.990080516>
- Morristown, N. J. (1981). Year 121 – 1981: “*The Origin of Spacewar*” by J. Martin Graetz, in: *Creative Computing*, August 1981 | 150 Years in the Stacks [MIT Üniversitesi Kütüphanesi]. <https://libraries.mit.edu/150books/2011/05/07/1981/>
- Murray, J., Bogost, I., Mateas, M., Nitsche, M. (2006). Game design education: Integrating computation and culture. *Computer*, 39(6), Art. 6. <https://doi.org/10.1109/MC.2006.195>
- MuseumofPlay. (2016, Mart 24). Video Game History Timeline [İnteraktif Oyun Müzesi]. Thestrong-National Museum of Play. <https://www.museumofplay.org/about/icheg/video-game-history/timeline>
- Mustać, K., Bačić, K., Skorin-Kapov, L., Sužnjević, M. (2022). Predicting Player Churn of a Free-to-Play Mobile Video Game Using Supervised Machine Learning. *Applied Sciences*, 12(6), 2795. <https://doi.org/10.3390/app12062795>
- Mutanen, T., Nousiainen, S., Ahola, J. (2010). *Customer churn prediction—A case study in retail banking*. 218. <https://doi.org/10.3233/978-1-60750-633-1-77>
- Müller, A. C., Guido, S. (2016). *Introduction to machine learning with Python: A guide for data scientists* (First edition). O'Reilly Media, Inc.
- Myers, B. A. (1998). A brief history of human-computer interaction technology. *Interactions*, 5(2), 44-54. <https://doi.org/10.1145/274430.274436>
- Nabiyev, V. (2012). *Yapay Zeka* (6. bs). Seçkin Yayıncılık.
- Narendra, K. S., Parthasarathy, K. (1990). Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks*, 1(1), 4-27. <https://doi.org/10.1109/72.80202>
- Nayyar, A., Puri, V. (2017). Comprehensive Analysis Performance Comparison of Clustering Algorithms for Big Data. *Review Of Computer Engineering Research*, 4, 54-80. <https://doi.org/10.18488/journal.76.2017.42.54.80>
- Newbold, P., Carlson, W. L., Thorne, B. M. (2020). *Statistics for business and economics* (Ninth edition, global edition). Pearson.
- Newman, J. (2013). *Video Games* (2nd bs). Routledge Publishing.

- Ng, A. (2017). *Machine Learning Yearning Book (1. bs)*. DeepLearning.AI. <https://info.deeplearning.ai/machine-learning-yearning-book>
- Nilsson, N. J. (2010). *The quest for artificial intelligence: A history of ideas and achievements*. Cambridge University Press.
- Nisbet, R., Miner, G., Yale, K., Elder, J. F., Peterson, A. F. (2018). *Handbook of statistical analysis and data mining applications (Second edition)*. Academic Press.
- O'Dea, S. (2020). *Smartphone users worldwide 2016-2021*. Statista. <https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/> [Ziyaret Tarihi: 18 Temmuz 2020]
- Oh, G., Ryu, T. (2007). Game Design on Item-selling Based Payment Model in Korean Online Games. *DiGRA Conference*, 650-657.
- Oliver, R. L. (1999). Whence Consumer Loyalty? *Journal of Marketing*, 63(4_suppl1), 33-44. <https://doi.org/10.1177/00222429990634s105>
- Olson, D. L., Delen, D. (2008). *Advanced Data Mining Techniques*. Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-540-76917-0>
- Opitz, D., Maclin, R. (1999). Popular Ensemble Methods: An Empirical Study, *Journal of Artificial Intelligence Research*, 11, 169-198. <https://doi.org/10.1613/jair.614>
- Oracle. (2020, Eylül 23). Predicting Your Business's Future. *Oracle NetSuite*. <https://www.netsuite.com/portal/resource/articles/financial-management/predictive-modeling.shtml>
- Ozay, M., Vural, F. T. Y. (2012). *A New Fuzzy Stacked Generalization Technique and Analysis of its Performance*. <https://doi.org/10.48550/ARXIV.1204.0171>
- Özdil, T., Oğuzhan, A., Çakıcı, M. (2015). *İstatistik (1. bs)*. Ekin Basım Yayın.
- Özen, Z. (2016). *Kimlik doğrulaması için tuş vuruş dinamiklerine dayalı bir güvenlik sisteminin yapay sinir ağları ile geliştirilmesi* [Doktora, İstanbul Üniversitesi]. <https://tez.yok.gov.tr/UlusalTezMerkezi/tezSorguSonucYeni.jsp>
- Öztemel, E. (2012). *Yapay Sinir Ağları (3. bs)*. Papatya Bilim Kitabevi.
- Pafka, S. (2015). *Data Science Los Angeles » Benchmarking Random Forest Implementations* <http://datascience.la/benchmarking-random-forest-implementations/>
- Pandey, A., Jain, A. (2017). Comparative analysis of KNN algorithm using various normalization techniques. *International Journal of Computer Network and Information Security*, 11(11), 36.
- Parisi, T. (2015). *Learning virtual reality: Developing immersive experiences and applications for desktop, web, and mobile (First edition)*. O'Reilly Media, Inc.
- Parlett, D. (1982). *Botticelli and Beyond: Over 100 of the World's Best Word Games*. Pantheon Books.
- Pedersen, C., Togelius, J., Yannakakis, G. N. (2010). Modeling Player Experience for Content Creation. *IEEE Transactions on Computational Intelligence and AI in Games*, 2(1), 54-67. <https://doi.org/10.1109/TCIAIG.2010.2043950>
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, É. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12(85), 2825-2830.
- Pirim, H. (2006). Yapay Zeka. *Journal of Yaşar University*, 1(1), 81-93.
- Piryonesi, s. M., El-Diraby, T. (2020). Data Analytics in Asset Management: Cost-Effective Prediction of the Pavement Condition. *Journal of Infrastructure Systems*, 26. [https://doi.org/10.1061/\(ASCE\)IS.1943-555X.0000512](https://doi.org/10.1061/(ASCE)IS.1943-555X.0000512)

- Plant, M. (2018, Aralık 21). Top 10 best-selling videogame consoles [Dünya Rekorları]. Guinness World Records. <https://www.guinnessworldrecords.com/news/2018/12/top-10-best-selling-videogame-consoles-551938>
- Polikar, R. (2006). Ensemble based systems in decision making. *IEEE Circuits and Systems Magazine*, 6(3), 21-45. <https://doi.org/10.1109/MCAS.2006.1688199>
- Prensky, M. (2001). Digital Natives, Digital Immigrants. *On the Horizon*, 9(5), 1-6.
- Pruett, C. (2003). The Evolution of Video Games, *Journal of Game Art*, 3(7)
- Przybylski, A. K., Weinstein, N. (2016). How we see electronic games. *PeerJ*, 4, e1931. <https://doi.org/10.7717/peerj.1931>
- Quandt, T., Looy, J. V., Vogelgesang, J., Elson, M., Ivory, J. D. ;, Consalvo, M., Mayra, F. (2015). Digital Games Research: A Survey Study on an Emerging Field and Its Prevalent Debates. *Journal of Communication*, 65(6), 975-996.
- Rätsch, G., Onoda, T., Müller, K.-R. (2001). Soft margins for AdaBoost. *Machine learning*, 42, 287-320.
- Rawlings, J. O., Pantula, S. G., Dickey, D. A. (2016). *Uygulamalı Regresyon Analizi* (Y. Akdi A. Şahin, Çev.; 1. bs). Nobel Akademi Yayınları.
- Reime, E. V. (2011). *Exploring the Freemium Business Model* [Yüksek Lisans]. University of Oslo.
- Rogers, B. (2018). *A Simple Guide to Churn Analysis* (1. bs). Everage.
- Roiger, R. J. (2017). *Data Mining: A Tutorial-Based Primer* (2. bs). Chapman and Hall/CRC. <https://doi.org/10.1201/9781315382586>
- Rokach, L. (2010). Ensemble-based classifiers. *Artificial Intelligence Review*, 33(1-2), 1-39. <https://doi.org/10.1007/s10462-009-9124-7>
- Rokach, L., Maimon, O. (2005). *Decision trees*. In Data mining and knowledge discovery handbook, 165-192.
- Roohi, S., Relas, A., Takatalo, J., Heiskanen, H., Hämäläinen, P. (2020). Predicting Game Difficulty and Churn Without Players. *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, 585-593.
- Room, A. (2010). *Dictionary of sports and games terminology*. McFarland Co.
- Rothenbuehler, P., Runge, J., Garcin, F., Faltings, B. (2015). Hidden Markov models for churn prediction. *2015 SAI Intelligent Systems Conference (IntelliSys)*, 723-730. <https://doi.org/10.1109/IntelliSys.2015.7361220>
- Rothmeier, K., Pflanzl, N., Hullmann, J. A., Preuss, M. (2021). Prediction of Player Churn and Disengagement Based on User Activity Data of a Freemium Online Strategy Game. *IEEE Transactions on Games*, 13(1), 78-88. <https://doi.org/10.1109/TG.2020.2992282>
- Rowe, M. W. (1992). The Definition of 'Game'. *Philosophy*, 67(262), Art. 262. <https://doi.org/10.1017/S0031819100040663>
- Rowley, J. (2007). The wisdom hierarchy: Representations of the DIKW hierarchy. *Journal of Information Science*, 33(2), 163-180. <https://doi.org/10.1177/0165551506070706>
- Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5), 206-215. <https://doi.org/10.1038/s42256-019-0048-x>
- Runge, J., Gao, P., Garcin, F., Faltings, B. (2014). Churn prediction for high-value players in casual social games. *2014 IEEE Conference on Computational Intelligence and Games*, 1-8. <https://doi.org/10.1109/CIG.2014.6932875>
- Russell, S. J., Norvig, P., Davis, E., Edwards, D. (2016). *Artificial intelligence: A modern approach* (Third edition, Global edition). Pearson.

- Saas, A., Guitart, A., Perianez, A. (2016). Discovering playing patterns: Time series clustering of free-to-play game data. *2016 IEEE Conference on Computational Intelligence and Games (CIG)*, 1-8. <https://doi.org/10.1109/CIG.2016.7860442>
- Sardone, N., Devlin-Scherer, R., Martinelli, J. (2008). *Game-based instruction in a college classroom*. <https://doi.org/10.4018/9781599048086.ch082>
- Schapire, R. E. (1990). The strength of weak learnability. *Machine Learning*, 5(2), 197-227. <https://doi.org/10.1007/BF00116037>
- Sellers, J. (2001). *Arcade fever: The fan's guide to the golden age of video games*. Running Press.
- Selvi, Ö. (2014). Müşteri İlişkileri Yönetimi ve Perakende Sektörüne Yönelik Bir Uygulama. *Erciyes Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 1(36), Art. 36.
- Shaaban, E., Helmy, Y., Khedr, A., Nasr, M. (2012). A Proposed Churn Prediction Model. *International Journal of Engineering Research and Applications (IJERA)*, 2, 693-697.
- Shabanah, S. (2011). *Computer games for algorithm learning*." Handbook of Research on Improving Learning and Motivation through Educational Games: Multidisciplinary Approaches. İçinde P. Felicia (Ed.), Handbook of Research on Improving Learning and Motivation through Educational Games: Multidisciplinary Approaches (ss. 1036-1063). IGI Global. <https://doi.org/10.4018/978-1-60960-495-0>
- Shaver, M. (2017). *What Makes Tetris so Incredibly Addictive?* Tetris. <https://tetris.com/article/44/what-makes-tetris-so-incredibly-addictive>
- Sınıksaran, E. (2011a). *İstatistiksel Yöntemler: Teori ve Uygulamalarıyla Çözümlü 407 Problem (3. bs)*. Türkmen Kitabevi.
- Sınıksaran, E. (2011b). *Şansın Matematiği (1. bs)*. Türkmen Kitabevi.
- Sifa, R., Hadiji, F., Runge, J., Drachen, A., Kersting, K., Bauckhage, C. (2015). Predicting Purchase Decisions in Mobile Free-to-Play Games. *AIIDE. Conference on Artificial Intelligence and Interactive Digital Entertainment*.
- Silahtaroglu, G. (2016). *Veri Madenciliği: Kavram ve Algoritmaları (3. bs)*. Papatya Bilim.
- Simon Kemp. (2022). *Digital 2022: Global Overview Report [Analysis]*. <https://datareportal.com/reports/digital-2022-global-overview-report>
- Sohrabi, B., Khanlari, A. (2007). Customer lifetime value (CLV) measurement based on RFM model, *Iranian Accounting & Auditing Review*, 14(47), pp 7- 20.
- Statista. (2020). F2P games revenues [Rapor]. *Free-to-Play (F2P) Games Market Revenue in North America from 2018 to 2023, by Category*. <https://www.statista.com/statistics/240993/share-of-f2p-mmo-spending-by-region/>
- Su, Y., Backlund, P., Engström, H. (2021). Comprehensive review and classification of game analytics. *Service Oriented Computing and Applications*, 15(2), 141-156. <https://doi.org/10.1007/s11761-020-00303-z>
- Suits, B. (1967). What Is a Game? *Philosophy of Science*, 34(2), Art. 2. JSTOR.
- Swain, P. H., Hauska, H. (1977). The decision tree classifier: Design and potential. *IEEE Transactions on Geoscience Electronics*, 15(3), 142-147. <https://doi.org/10.1109/TGE.1977.6498972>
- Şeker, S. E. (2016). Müşteri Kayıp Analizi. *YBS Ansiklopedi*, 3(1), Art. 1.
- Tabachnick, B. G., Fidell, L. S. (2013). *Using multivariate statistics (6th ed)*. Pearson Education.
- Taha, H. A. (2017). *Yöneylem Araştırması (Ş. A. Baray Ş. Esnaf, Çev.; 10. bs)*. Literatür Yayıncılık.
- Tamassia, M., Raffae, W., Sifa, R., Drachen, A., Zambetta, F., Hitchens, M. (2016). Predicting player churn in destiny: A Hidden Markov models approach to predicting player

- departure in a major online game. *2016 IEEE Conference on Computational Intelligence and Games (CIG)*, 1-8. <https://doi.org/10.1109/CIG.2016.7860431>
- Tan, P.-N., Steinbach, M., Karpatne, A., Kumar, V. (2019). *Introduction to data mining (Second edition)*. Pearson.
- Tavinor, G. (2008, Ocak 14). “*Definition of Videogames*”. Petrilankoski.Com. <https://petrilankoski.com/2008/01/14/definition-of-videogames-by-grant-tavinor/> [Ziyaret Tarihi: 14 Ocak 2020]
- TDK. (2018). *TDK Sözlüğü*. Dil Derneği Yayınları.
- Tekinbaş, K. S., Zimmerman, E. (2003). *Rules of play: Game design fundamentals*. MIT Press.
- Tezel, Y. (2016). *Bilgisayar oyunları tarihi*. Sokak Kitapları Yayıncılık.
- Tolun, S. (2008). *Destek vektör makineleri: Banka başarısızlığının tahmini üzerine bir uygulama* [Doktora Tezi]. İstanbul Üniversitesi.
- Tomić, N. Z. (2019). Economic model of microtransactions in video games. *Journal of Economic Science Research*, 1(1). <https://doi.org/10.30564/jesr.v1i1.439>
- Tony Smith. (2000). *Sony PS One sales rocket as PS Two famine continues*. www.theregister.co.uk.
http://www.theregister.co.uk/2000/12/06/sony_ps_one_sales_rocket/
- Tsymbalov, E. (2016). *Churn Prediction for Game Industry Based on Cohort Classification Ensemble*, Third Workshop on Experimental Economics and Machine Learning, Moscow, Russia.
- Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, LIX(236), 433-460. <https://doi.org/10.1093/mind/LIX.236.433>
- Uğuz, S. (2020). *Makine Öğrenmesi: Teorik Yönleri ve Python Uygulamaları ile Bir Yapay Zeka Ekolü* (2. bs). Nobel Akademi Yayınları.
- Valiant, L. G. (1984). A theory of the learnable. *Communications of the ACM*, 27(11), 1134-1142. <https://doi.org/10.1145/1968.1972>
- Van den Poel, D., Larivière, B. (2004). Customer attrition analysis for financial services using proportional hazard models. *European Journal of Operational Research*, 157(1), 196-217. [https://doi.org/10.1016/S0377-2217\(03\)00069-9](https://doi.org/10.1016/S0377-2217(03)00069-9)
- Vo, N. N. Y., Liu, S., Li, X., Xu, G. (2021). Leveraging unstructured call log data for customer churn prediction. *Knowledge-Based Systems*, 212, 106586. <https://doi.org/10.1016/j.knosys.2020.106586>
- Walker, T. (2006). EverQuest II announced. *GameSpot*. <https://www.gamespot.com/articles/everquest-ii-announced/1100-2863644/>
- Wang, H., Ma, C., Zhou, L. (2009). A Brief Review of Machine Learning and Its Application. *2009 International Conference on Information Engineering and Computer Science*, 1-4. <https://doi.org/10.1109/ICIECS.2009.5362936>
- Wang, Q.-F., Xu, M., Hussain, A. (2019). Large-scale Ensemble Model for Customer Churn Prediction in Search Ads. *Cognitive Computation*, 11(2), 262-270. <https://doi.org/10.1007/s12559-018-9608-3>
- Weller, D., Lobão, A. S., Hatton, E. (2004). *Spacewar!* İçinde D. Weller, A. S. Lobão, E. Hatton (Ed.), *Beginning .NET Game Programming in VB .NET* (ss. 245-269). Apress. https://doi.org/10.1007/978-1-4302-0724-5_5
- Wesley, D. T. A., Barczak, G. (2010). *Innovation and marketing in the video game industry: Avoiding the performance trap*. Gower ; Ashgate.
- What is the difference between the R gbm (gradient boosting machine) and xgboost (extreme gradient boosting)? - Quora. (2017). [Forum]. <https://www.quora.com/What-is-the->

difference-between-the-R-gbm-gradient-boosting-machine-and-xgboost-extreme-gradient-boosting

- Whitton, N. (2014). *Learning with Digital Games: Research and Theory*. Routledge Publishing.
- Wickramasinghe, I., Kalutarage, H. (2021). Naive Bayes: Applications, variations and vulnerabilities: a review of literature with code snippets for implementation. *Soft Computing*, 25(3), 2277-2293. <https://doi.org/10.1007/s00500-020-05297-6>
- Wijman, T. (2018). Global Games Market Report 2018. *Computer Hope*. <https://newzoo.com/insights/articles/global-games-market-reaches-137-9-billion-in-2018-mobile-games-take-half/>
- Williams, G., Williams, G. (2011). Descriptive and predictive analytics. *Data Mining with Rattle and R: The Art of Excavating, Data for Knowledge Discovery*, 171-177.
- Wilson, F. (2006, Mart 23). The Freemium Business Model. AVC. https://avc.com/2006/03/the_freemium_bu/
- Winget, M. A. (2011). Videogame preservation and massively multiplayer online role-playing games: A review of the literature. *Journal of the American Society for Information Science and Technology*, 62(10), Art. 10. <https://doi.org/10.1002/asi.21530>
- Witell, L., Löfgren, M. (2013). From service for free to service for fee: Business model innovation in manufacturing firms. *Journal of Service Management*, 24(5), Art. 5. <https://doi.org/10.1108/JOSM-04-2013-0103>
- Woh, S.-M., Lee, J.-H. (2018). Game State Prediction with Ensemble of Machine Learning Techniques. *2018 Joint 10th International Conference on Soft Computing and Intelligent Systems (SCIS) and 19th International Symposium on Advanced Intelligent Systems (ISIS)*, 89-92. <https://doi.org/10.1109/SCIS-ISIS.2018.00025>
- Wolf, M. J. P. (Ed.). (2002). *The medium of the video game (1st ed)*. University of Texas Press.
- Wolf, M. J. P. ;, Perron, B. (2003). *The Video Game Theory Reader*. Routledge Publishing.
- Wright, R. E. (1995). *Logistic regression*. İçinde Reading and understanding multivariate statistics (ss. 217-244). American Psychological Association.
- Xu, M., Watanachaturaporn, P., Varshney, P., Arora, M. (2005). Decision tree regression for soft classification of remote sensing data. *Remote Sensing of Environment*, 97(3), 322-336. <https://doi.org/10.1016/j.rse.2005.05.008>
- Xue, S., Wu, M., Kolen, J., Aghdaie, N., Zaman, K. A. (2017). Dynamic Difficulty Adjustment for Maximized Engagement in Digital Games. *Proceedings of the 26th International Conference on World Wide Web Companion - WWW '17 Companion*, 465-471. <https://doi.org/10.1145/3041021.3054170>
- Yang, Z., Pan, Z., Wang, Y., Cai, D., Liu, X., Shi, S., Huang, S.-L. (2021). Interpretable Real-Time Win Prediction for Honor of Kings, a Popular Mobile MOBA Esport (arXiv:2008.06313). *arXiv*. <https://doi.org/10.48550/arXiv.2008.06313>
- Yeshwanth, V., Raj, V., Mohan, S. (2011). Evolutionary Churn Prediction in Mobile Networks Using Hybrid Learning. *Proceedings of the 24th International Florida Artificial Intelligence Research Society, FLAIRS - 24*.
- Yin, C. (2019). *Stochastic ordering of Gini indexes for multivariate elliptical random variables*. <https://doi.org/10.48550/ARXIV.1908.01943>
- Zackariasson, P., Dymek, M. (2016). *Video Game Marketing: A student Textbook (1. bs)*. Routledge. <https://doi.org/10.4324/9781315748900>
- Zhang, H., Su, J. (2004). *Naive Bayesian Classifiers for Ranking*. İçinde J.-F. Boulicaut, F. Esposito, F. Giannotti, D. Pedreschi (Ed.), *Machine Learning: ECML 2004 (C. 3201, ss. 501-512)*. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-30115-8_46

- Zhang, S., Li, X., Zong, M., Zhu, X., Cheng, D. (2017). Learning k for kNN Classification. *ACM Transactions on Intelligent Systems and Technology*, 8(3), 1-19. <https://doi.org/10.1145/2990508>
- Zhao, S., Wu, R., Tao, J., Qu, M., Zhao, M., Fan, C., Zhao, H. (2022). perCLTV: A General System for Personalized Customer Lifetime Value Prediction in Online Games. *ACM Transactions on Information Systems*. <https://doi.org/10.1145/3530012>
- Zheng, A., Chen, L., Xie, F., Tao, J., Fan, C., Zheng, Z. (2020). *Keep You from Leaving: Churn Prediction in Online Games*. İçinde Y. Nah, B. Cui, S.-W. Lee, J. X. Yu, Y.-S. Moon, S. E. Whang (Ed.), *Database Systems for Advanced Applications* (ss. 263-279). Springer International Publishing. https://doi.org/10.1007/978-3-030-59416-9_16
- Zhou, Z.-H. (2012). *Ensemble Methods: Foundations and Algorithms*, Chapman and Hall/CRC. <https://doi.org/10.1201/b12207>
- Ziegler, A., König, I. R. (2014). Mining data with random forests: Current options for real-world applications. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 4(1), 55-63.
- Zikopoulos, P., deRoos, D., Parasuraman, K., Deutsch, T., Giles, J., Corrigan, D. (2012). Harness the Power of Big Data The IBM Big Data Platform. *McGraw-Hill Osborne Media*. <https://doi.org/10.1036/9780071808187>
- Zou, K. H., O'Malley, A. J., Mauri, L. (2007). Receiver-Operating Characteristic Analysis for Evaluating Diagnostic Tests and Predictive Models, *Circulation*, 115(5), 654-657.

EKLER

EK-1

Kategori	Oyuncu Kayıt Öznitelikleri	Oyuncu Kayıt Niteliği	Açıklama	Veri Tipi
Bağlantı (Connection)	1003	EnterWorld	Oyuncu oyun sunucusuna girdiğinde	Nümerik
	1004	LeaveWorld	Oyuncu oyun sunucusundan ayrıldığında	Nümerik
	1005	EnterZone	Oyuncu bölgeye girdiğinde	Nümerik
	1006	LeaveZone	Oyuncu bölgeden ayrıldığında	Nümerik
	1010	Teleport	Oyuncu başka bir bölgeye ışınlандığında	Nümerik
	1012	DeletePC	Oyuncu karakteri silmeyi veya tamamlamayı talep ettiğinde	Nümerik
Karakter (Character)	1013	PCLevelUp	Oyuncu karakteri seviye atladığında	Nümerik
	1016	GetExperience	Oyuncu tecrübe puanı kazandığında	Nümerik
	1017	GetMoney	Envanterdeki oyun parası arttığında	Nümerik
	1018	SpendMoney	Envanterdeki oyun parası azaldığında	Nümerik
	1022	GetItem	Envanterdeki öğeler arttığında	Nümerik
	1023	Loseltem	Envanterdeki öğeler azaldığında	Nümerik
	1101	InviteParty	Oyuncu bir hedefi partiye davet ettiğinde	Nümerik
	1102	JoinParty	Oyuncu gruba katıldığında	Nümerik
	1103	RefuseParty	Oyuncu grup davetini reddettiğinde	Nümerik
	1104	DismissParty	Oyuncu gruptan ayrıldığında	Nümerik
	1105	KickPartyMember	Grup yetkilisi oyuncuyu gruptan kovduğunda	Nümerik
	1201	Exhaustion	Oyuncunun ölmeğe yakın olması - sersemlemesi	Nümerik
	1202	Die	Oyuncu öldüğünde	Nümerik
	1203	Resurrect	Oyuncu dirildiğinde	Nümerik
	1208	KillNPC	Oyuncu oynanmayan karakteri (Non Player Character) öldürdüğünde	Nümerik
	1209	KillPC	Oyuncu herhangi bir karşı oyuncuyu öldürdüğünde	Nümerik
	1404	DuelEnd(PC)	1 v 1 diello yaptığında	Nümerik
	1406	DuelEnd(Team)	Düello bittiğinde	Nümerik

	1407	MoveToArena	Oyuncu düello veya zindan (dungeon) sunucusuna taşındığında	Nümerik
	1422	PartyBattleEnd(Team)	Takım savaşı bittiğinde	Nümerik
	1424	PartyBattleResulet(PC)	Oyuncu savaşı bittiğinde	Nümerik
	1425	OccupyBase	Oyuncu üssü işgal ettiğinde	Nümerik
	2001	LootItem	Oyuncu bir eşyayı yağmaladığında	Nümerik
	2002	UseItem	Oyuncu destek eşyası kullandığında	Nümerik
	2004	DestroyItem	Eşya inha edildiğinde	Nümerik
	2006	GetPartyLootMoney	Oyuncu partiyi yağmalayarak para kazandığında	Nümerik
	2011	PartyAuctionStart	Ganimet dağıtımına başladığında	Nümerik
	2013	BidPartyAuction	Oyuncu ganimet için bir teklif verdiğinde	Nümerik
	2014	PartyAuctionSuccess	Oyuncu ganimeti satın aldığıda	Nümerik
	2016	DistributeAuctionMoney	Kazanan teklif dağıtıldığında	Nümerik
	2102	UnEquipItem	Oyuncu bir eşyayı teçhizatın çıkardığında	Nümerik
	2103	SaveEquipInfo	Oyuncu bir bölgeye girdiğinde ve eşya bilgisi kayıt edildiğinde	Nümerik
	2105	RevealItem	Oyuncu Ruh Kalkanı (Soul Shield) birleştirdiğinde	Nümerik
	2106	ConsumeGemByReveal	Oyuncu Ruh Kalkanını eriterek kullanılan malzeme kaydedildiğinde	Nümerik
	2109	GetItemByDecomposition	Oyuncu bir eşyayı ayrıştırdığında	Nümerik
	2112	ExpandWarehouse	Oyuncu depoyu genişlettiğinde	Nümerik
	2113	RepairItem	Oyuncu öğeyi onardığında	Nümerik
	2121	GrowUpItem	Oyuncu eşyayı başka bir eşyaya dönüştürmeye çalıştığında	Nümerik
	2126	ResultOfTransform	Oyuncu dönüştürme sonucunu gördüğünde	Nümerik
	2127	ExceedItemLimit	Oyuncu eşya limitini aşmaya çalıştığında	Nümerik
	2141	ChangeItemLook	Oyuncu öğenin görünümünü değiştirdiğinde veya kaldırdığında	Nümerik
	2145	FeedingResult	Oyuncu rastgele eşya ile desteklendiğinde	Nümerik
	2201	TradeGiveItem	Oyuncu takas penceresinden ürün verdiğinde	Nümerik
	2202	TradeGetItem	Oyuncu takas penceresinden eşyayı aldığıda	Nümerik
	2204	SellItem	Oyuncu Mağaza aracılığıyla ürün sattığında	Nümerik
	2205	BuyMyItem	Oyuncu ürünü mağaza aracılığıyla yeniden satın aldığıda	Nümerik

Eşya (Item)

	2206	TradeGiveMoney	Oyuncu dükkân penceresinden para verdiğinde	Nümerik
	2207	TradeGetMoney	Oyuncu para bozma bürosundan para aldığında	Nümerik
	2209	GetItemFromNPC	Oyuncu NPC'den takas yoluyla eşya aldığında	Nümerik
	2221	DepositItem	Oyuncu ürünü depoya yatırdığında	Nümerik
	2222	RetriveItem	Oyuncu depodan ürün aldığında	Nümerik
	2301	PutMainAuction	Oyuncu eşyayı açık artırmaya koyduğunda	Nümerik
	2307	BuyItemNowMainAuction	Oyuncu açık artırmadan eşya satın aldığında hemen	Nümerik
	2405	UseGatheringItem	Oyuncu toplanan bir eşya kullandığında	Nümerik
	2407	GetGatheringItem	Oyuncu zanaat/toplama eşyası kullanarak eşya aldığında	Nümerik
	2503	ExpireEventItem	Özel etkinliğin süresi dolduğunda	Nümerik
Beceri (Skill)	4001	AcquireSkill	Oyuncu yetenek kazandığında	Nümerik
	4002	SkillLevelUp	Oyuncunun yetenek seviyesi yükseldiğinde	Nümerik
	4006	LearnTraining	Oyuncu antreman modunda puan kullandığında	Nümerik
	5001	AcquireQuest	Oyuncu bir görev aldığında	Nümerik
Görev (Quest)	5004	CompleteQuest	Oyuncu görevi tamamladığında	Nümerik
	5005	DisposeQuest	Oyuncu görevi bıraktığında	Nümerik
	5006	GetQuestItem	Oyuncu görev eşyası kazandığında	Nümerik
	5008	GetQuestSkill	Oyuncu görev sonrası yetenek kazandığında	Nümerik
	5010	GetChallengeTodayItem	Oyuncu günlük görevde ödüllendirildiğinde	Nümerik
	5011	CompleteChallengeToday	Oyuncu günlük yarışmayı tamamladığında	Nümerik
Lonca (Guild)	5014	GetChallengeWeekItem	Oyuncu haftalık yarışma ödülü aldığında	Nümerik
	5015	CompleteChallengeWeek	Oyuncu haftalık yarışmayı tamamladığında	Nümerik
	6001	CreateGuild	Oyuncu ücretini ödeyerek loncayı kurduğunda	Nümerik
	6002	DestoryGuild	Oyuncu loncayı kaldırdığında	Nümerik
	6003	GuildLevelUp	Lonca gereksinimi karşılayarak seviye atladığında	Nümerik
	6004	InviteGuild	Oyuncu bir diğer oyuncuyu loncaya davet ettiğinde	Nümerik
	6005	JoinGuild	Bir oyuncu loncaya katıldığında	Nümerik
	6008	RefuseGuildInvite	Bir oyuncu lonca davetini reddettiğinde	Nümerik
	6009	DismissGuild	Oyuncu loncadan ayrıldığında	Nümerik
	6010	KickGuildMember	Yöneticinin bir oyuncuyu loncadan attığında	Nümerik

DİĞER ÖZNEMLİKLER			Seq	Günlük kayıt sırası kaydı	Benzersiz (unique) oluşturulan kod
			Time	Oyuna girişlerde tutulan farklı zaman kayıtları	Tarih veri türü
			Actor Level	Oyuncun oyun içi seviyesi	Nümerik - 1-55 arası olabilir.
			Actor Race	Oyuncunun yönettiği karakterin ırkı	Kategorik – 9 farklı türde
			Actor Job	Oyuncunun yönettiği karakterin işi	Kategorik – 10 farklı türde
			Actor Mastery Level	55 seviyeden sonra oyuncuya sunulan yeni seviye paketi – 25'e kadar olabilir	Nümerik
			Actor Gender	Oyuncunun yönettiği karakterin cinsiyeti	Kategorik – İkili
			Old Value3	Oyuncunun oyun parası değişiminden önceki miktarı	Nümerik
			Use Value2	Oyuncunun oyun parası değişim miktarı	Nümerik
			survival_time	Oyuncunun oyuna başladığı günden itibaren şirketle geçirdiği süre	Nümerik
			Churn	Kayıp olma durumu – kayıp veya kayıp değil	Kategorik – İkili
			Total_level	Normal seviye ve ustalık seviyelerinin toplamı	Nümerik
			distributed-attribute1		Nümerik
			distributed-attribute2		Nümerik
			NOPDP (NumberOfPlayedDaysPer Player)	Oyuncu başına oynanan gün sayısı	Nümerik
			STPBD (SurvivalTimePerPlayer ByDays)	Gün Bazında Oyuncu Başına Hayatta Kalma Süresi	Nümerik
			STPAML (Survival Time Per Player Average Mastery Level)	Oyuncu ortalama ustalık seviyesi başına düşen hayatta kalma süresi	Nümerik
			STPAL (Survival Time Per Player Average Level)	Oyuncu ortalama seviyesi başına düşen hayatta kalma süresi	Nümerik

Veri	Öznitelik Adı	Açıklaması	Veri Tipi
Oyuncu Kayıtları	UserID	Oyuncuya atanan benzersiz olduğunu gösteren etiket numarası	Benzersiz (Unique) - Nümerik
	Amount	Dolar bazında ödediği miktar	Nümerik
	Chips	Oyun içi alınan ödül miktarı	Nümerik
	Date	Satın alımın gerçekleştiği tarih etiketi	Tarih
	Channel	Ödeme yöntemi	Kategorik
Etkileşim Kayıtları	Sequence	Oyuncu kaydının an (sekans) etiketi	Benzersiz (Unique) - Nümerik
	UserID	Oyuncuya atanan benzersiz olduğunu gösteren etiket numarası	Benzersiz (Unique) - Nümerik
	GameID	Oyuncunun her oyun için atanan etiketidir. Her <u>GameID</u> etiketi, farklı türde oyunlarda maç başına farklı uzunluklara sahiptir.	Benzersiz (Unique) - Nümerik
	Level	Maç başına minimum 20 çipten maç başına maksimum 10 milyon çipe kadar oyun seviyesi	Nümerik
	WinNo	Her bir oturum için kazanılan maç sayısı	Nümerik
	DrawNo	Her bir oturum için berabere kalınan maç sayısı	Nümerik

	LostNo	Her bir oturum için kazanılan maç sayısı	Nümerik
	WinAmt	Her bir oturum için kazanılan toplam ödül miktarı	Nümerik
	LostAmt	Her bir oturum için kaybedilen toplam ödül miktarı	Nümerik
	Date	Oyunun tarih etiketi	Tarih
	Currency_Type1	Para birimi türü 1	Kategorik
	Currency_Type2	Para birimi türü 2	Kategorik



Mini Oyun Adı	Kodu	Süresi
SLOT_100_LINE	1008	20 Saniye
BINH	9001	80 Saniye
POKDENG	9002	80 Saniye
XOCDIA	9003	20 Saniye
SHOW	9004	80 Saniye
BAUCUA	9005	20 Saniye
SIKU	9006	80 Saniye
SLOT_20_LINE	9008	20 Saniye

ÖZGEÇMİŞ

Kişisel Bilgiler	
Adı Soyadı	KAAN ARIK
Doğum Yeri	
Doğum Tarihi	Tarih girmek için tıklayın veya dokununuz.
Uyruğu	☒ T.C. ☐ Diğer:
E-Posta Adresi	
Web Adresi	

Eğitim Bilgileri	
Lisans	
Üniversite	SAKARYA ÜNİVERSİTESİ
Fakülte	EĞİTİM FAKÜLTESİ
Bölümü	BİLGİSAYAR VE ÖĞRETİM TEK. EĞİTİMİ ÖĞR.
Mezuniyet Yılı	28.07.2015

Yüksek Lisans	
Üniversite	SAKARYA ÜNİVERSİTESİ
Enstitü Adı	EĞİTİM BİLİMLERİ ENS.
Anabilim Dalı	BİLGİSAYAR VE ÖĞR. TEK. EĞİTİMİ ABD
Programı	BİLGİSAYAR VE ÖĞR. TEK. EĞİTİMİ

Doktora	
Üniversite	İstanbul Üniversitesi
Enstitü Adı	Fen Bilimleri Enstitüsü
Anabilim Dalı	Enformatik Anabilim Dalı
Programı	Enformatik Programı

Makale ve Bildiriler	
Arık, K. (2016). <i>Zihinsel Engelliler Öğrencilerin Öğretim Ortam Tasarımlarının İncelenmesi</i>. SEAK - Sakarya'da Eğitim Araştırmaları Kongresi (Tam Metin Bildiri/Sözlü Sunum) Bektaş, M., Arık, K., İşbulan, O. (2017). <i>The Effect Of Cloud Computing Supported Learning Activities On Academic Success</i>. 26. Uluslararası Eğitim Bilimleri Kongresi, 60-62. (Özet Bildiri/Sözlü Sunum) Arık, K. (2016). <i>Zihinsel Engelliler Öğrencilerin Öğretim Ortam Tasarımlarının İncelenmesi</i>. Seak - Sakarya'da Eğitim Araştırmaları Kongresi (Tam Metin Bildiri/Sözlü Sunum) Arık, K. ,Bektaş M. (2016). <i>Level Of Public Education Center Students' Digital Literacy: An Example Of Duzce</i>. Iclcl - International Conference Lifelong Education And Leadership, 635-641. (Tam Metin Bildiri/Sözlü Sunum)	

- Arık, K.** (2019). A Usability Research On Task Prediction For Experienced Users With Cogtool. *The Journal Of Cognitive Systems*, 4(2), 46-53.
- Arık, K., Kızılcı M.** (2019). Identifying Digital Literacy Of High School Students: A Case Of Hendek. *Sakarya University Journal Of Education*, 9(1), 47-68.
- Gerçek, S., **Arık K.** (2020). Dijital Oyunlarda Şiddet Unsuruna Oyuncuların Yüklediği Anlam: Üniversite Öğrencilerine Yönelik Bir Metafor Analizi. *Beykoz Akademi Dergisi*, 8(2), 251-269., Doi: 10.14514/Byk.M.26515393.2020.8/2.251-269
- Akkaya, A., **Arık, K.**, (2020) *İnsan Bilgisayar Etkileşimi Araştırma ve Uygulamalar, Bölüm Adı:(Eğitimcilerin Gözünden Scratch 3 Görsel Programlama Ortamının Arayüz İncelemesi)*, İstanbul University Press, Editör:Prof. Dr. Sevinç Gülseçen, Prof. Dr. Kerem Rızvanoğlu, Doç. Dr. Nilgün Tosun, Dr. Emre Akadal, Basım Sayısı:1, Sayfa Sayısı 316, Isbn:978-605-07-0726-7, Türkçe(Bilimsel Kitap)
- Arık, K.**, Gezer M., Tolun, S. (2022). *The Study Of Indicators Affecting Customer Churnin Mmorpg Games With Machine Learning Models.* The Upravenets, 13(6), 70-85., doi: 10.29141/2218-5003-2022-13-6-6
- Arık, K.** (2022). Social Media Content Review Of Mmorpg Games: Reddit Comment Scraping And Sentiment Analysis. *Journal Of Emerging Computer Technologies*, 1(1), 13-21. (Yayın No: 7690460)
- Arık K.**, Gezer M., Tayalı S.T. (2022). Bibliometric Analysis Of Scientific Studies Published On Game Customer Churn Analysis Between 2008 And 2022. *Journal Of Politics Economy And Management*, 5(1), 56-75.
- Ağdaş, M.T., **Arık, K.** (2022). *Classification Of Alzheimer's Disease Using Visual Based Vit (Vision Transformers) Model.* 2 Nd International Conference On Engineering And Applied Natural Sciences (Özet Bildiri/Sözlü Sunum)
- Arık, K.** (2022), *Dijital Çağda Uygulamalı Veri Yönetimi, Bölüm Adı:(Bilişsel Bilim Bağlamında İnsan Bilgisayar Etkileşimi: Cogtool Yazılımında Örnek Bir Uygulama)* Korkmaz, A., Büyükgöze, S., Paradigma Akademi Yayınları, Basım Sayısı:1, Sayfa Sayısı 134, Isbn:978-605-72190-6-0, Türkçe (Bilimsel Kitap)
- Meral, P.S, İpek, M., **Arık, K.** (2022). Bitcoin Kullanımının Ülke Marka Bilinirliğine Etkisi: El Salvador Örneği. *Yeni Yüzyıl'da İletişim Kongresi* (Tam Metin Bildiri/Sözlü Sunum)
- Meral, P.S., **Arık, K.**, İpek, M. (2023). Sentiment Analiziyle Bitcoin Kullanımının Ülke Marka Bilinirliğine Etkisi: El Salvador Örneği. *YeniYüzyıl'da İletişim Araştırmaları Dergisi*, 2(6), 8-31.
- Arık, K.** (2023). Machine Learning Models on MOBA Gaming: League of Legends Winner Prediction. *Acta Infologica*, 7(1), 52-60.
- Kırksekiz, A., **Arık, K.** (2023). *Türkiye'de Dijital Oyun Sektöründe İstihdam Edilebilecek Ara Eleman Niteliklerinin Belirlenmesi.* ERPA International Congresses on Education in 2023, Cunda, Balıkesir.