T.C.
SÜLEYMAN DEMİREL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

E-TİCARET SİSTEMLERİNDE SAHTECİLİK TESPİT SİSTEMİ TASARIMI

Fatma Serap ÖRENLİ

Danışman
Dr. Öğr. Üyesi Ufuk ÖZKAYA

YÜKSEK LİSANS TEZİ
ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI
ISPARTA 2019

TEZ ONAYI

Fatma Serap ÖRENLİ tarafından hazırlanan **"E-Ticaret Sistemlerinde Sahtecilik Tespit Sistemi Tasarımı"** adlı tez çalışması aşağıdaki jüri üyeleri önünde Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü **Elektrik Elektronik Mühendisliği Anabilim Dalı**'nda **YÜKSEK LİSANS TEZİ** olarak başarı ile savunulmuştur.

Danışman	**Dr. Öğr. Üyesi Ufuk ÖZKAYA** Süleyman Demirel Üniversitesi	.....
Jüri Üyesi	**Prof. Dr. Selçuk ÇÖMLEKÇİ** Süleyman Demirel Üniversitesi	.....
Jüri Üyesi	**Doç. Dr. Övünç POLAT** Akdeniz Üniversitesi	.....

Enstitü Müdürü **Doç. Dr. Şule Sultan UĞUR** .....

TAAHHÜTNAME

Bu tezin akademik ve etik kurallara uygun olarak yazıldığını ve kullanılan tüm literatür bilgilerinin referans gösterilerek tezde yer aldığını beyan ederim.

Fatma Serap ÖRENLİ

İÇİNDEKİLER

Sayfa

İÇİNDEKİLER i
ÖZET iii
ABSTRACT iv
TEŞEKKÜR v
ŞEKİLLER DİZİNİ vi
ÇİZELGELER DİZİNİ vii
SİMGELER VE KISALTMALAR DİZİNİ ix
1. GİRİŞ 1
1.1. Elektronik Ticaret 1
1.1.2. E-Ticaret Modelleri 2
1.1.2.1. İşletmeden İşletmeye (Business To Business) 2
1.1.2.2. İşletmeden Tüketiciye (Business To Customer) 2
1.1.3. Örnek Sistemler 3
1.1.3.1. Hazır E-Ticaret Sistemleri 3
1.1.3.2. Özel Yazılım Sistemleri 3
1.1.4. Önerilen Sistem 4
1.1.4.1. Teknik Altyapı 4
1.1.4.2. Yönetim Paneli 6
1.1.4.3. Ürün Yönetimi 6
1.1.4.4. Seçeneksiz Ürün İşlemleri 7
1.1.4.5. Seçenekli Ürün İşlemleri 8
1.1.4.6. Kategori Yönetimi 10
1.1.4.7. Arama Motoru Optimizasyonu Yönetimi 13
1.1.5. Güvenlik 13
1.1.5.1. Veri Tabanı Güvenliği 13
1.1.5.2. Web Ödeme Entegrasyonları ve Temel Kavramlar 14
1.1.5.3. Sahtecilik Türleri ve Önleme Yöntemleri 16
1.1.6. Problemin Tanımı 17
2. LİTERATÜR ÖZETLERİ 19
3. MATERYAL & METOD 23
3.1. Kullanılan Yöntemler 23
3.1.1. Naive Bayes Algoritması 24
3.1.2. K-En Yakın Komşu Algoritması 27
3.1.3. Rassal Orman Yöntemi 30
3.1.4. Genelleştirilmiş Regresyon Sinir Ağı Yöntemi 32
3.1.5. Lojistik Regresyon Yöntemi 34
3.1.6. Destek Vektör Makineleri Yöntemi 38
3.2. Kullanılan Veri Setleri 40
3.3. Yaklaşımlar 42
3.3.1. Temel Bileşenler Analizi 42
3.3.2. Tüm Veri Seti Yaklaşımı 42
3.3.3. Özellik Çıkarımı Yaklaşımı 42
3.3.4. Under-sampling Yaklaşımı 44
3.3.5. Karışıklık Matrisi (Confusion Matrix) 44
4. ARAŞTIRMA BULGULARI VE TARTIŞMA 49
4.1. Veri Seti-I için elde edilen sonuçlar 50

4.1.1. Naive Bayes Uygulaması 50
4.1.2. K- En Yakın Komşu Uygulaması 51
4.1.3. Lojistik Regresyon Uygulaması 52
4.1.4. RF Uygulaması 53
4.1.5. GRNN Uygulaması 54
4.1.6. SVM Uygulaması 54
4.2. Veri Seti–II için elde edilen sonuçlar 57
4.2.1. Lojistik Regresyon Uygulaması 58
4.2.2. KNN Uygulaması 59
4.2.3. Naive Bayes Uygulaması 60
4.2.4. RF Uygulaması 60
4.2.5. SVM Uygulaması 61
5. SONUÇ VE ÖNERİLER 64
KAYNAKLAR 66
ÖZGEÇMİŞ 68

ÖZET

Yüksek Lisans Tezi

E-TİCARET SİSTEMLERİNDE SAHTECİLİK TESPİT SİSTEMİ TASARIMI

Fatma Serap ÖRENLİ

Süleyman Demirel Üniversitesi
Fen Bilimleri Enstitüsü
Elektrik Elektronik Mühendisliği Anabilim Dalı

Danışman: Dr. Ögr. Üyesi Ufuk ÖZKAYA

E-Ticaret sektörü ülkemizde ve dünyada hızla büyüyerek potansiyel varlığını ve gelişimini sürdürmektedir. Buna bağlı olarak mevcut işlerini e-ticaret ortamına taşımak isteyen firma sahiplerinin sayısı da hızla artış göstermektedir. E-Ticaret alanında taleplerin yüksek olması nedeniyle çok sayıda e-ticaret yazılımlarının bulunmasına karşın bu yazılımlardaki altyapı eksiklikleri, müşteri taleplerine kolay cevap verememesi, karmaşık yapıda tasarlanmış olmaları ve en önemlisi de verilen siparişlerde oluşabilecek sahtecilik kontrollerinin yapılmadığı ya da gözardı edildiği görülmektedir. Bu sebeple e-ticaret sektörüne katılacak olan yeni yazılımların geliştirilmesinde var olan bu eksikliklerin giderilmesine odaklanılmalı, oluşabilecek sahtecilik işlemleri de göz önünde bulundurularak önlemler alınmaya çalışılmalıdır.

Bu çalışmada mevcut sistemlerin incelenmesi ile alternatif bir e-ticaret altyapı yazılımının sunulması ve bununla birlikte tasarlanacak olan sistemin, siparişlerde oluşabilecek sahteciliklerin (fraud) tespit edilmesine yönelik, literatürdeki veri setlerinden yararlanılarak veri madenciliği sınıflandırma algoritmaları aracılığıyla tahmin sistemi önerilmesi ve performans karşılaştırmalarının yapılması amaçlanmaktadır.

Çalışma iki farklı veri seti için gerçekleştirilmiştir. Birinci çalışmada Eylül 2013 yılına ait 284.807 kayıttan oluşan kredi kartı verileri üzerinde özellik çıkarım yaklaşımları uygulanarak K-En Yakın Komşu Algoritması, Naive Bayes, Lojistik Regresyon, Destek Vektör Makineleri, Rassal Orman, Genelleştirilmiş Regresyon Sinir Ağı sınıflandırıcılarıyla başarım sonuçları değerlendirilmiştir. Belirtilen veri seti üzerinde kullanılan yöntemlerden Lojistik Regresyon yönteminin diğer yöntemlere göre daha yüksek başarı sağladığı tespit edilmiştir. İkinci çalışmada ise oluşturulan 700 adet kayıt içeren veri seti için yöntemlerin başarısı incelenmiştir. Kullanılan yöntemler arasında yine en iyi performans Lojistik Regresyon yöntemi ile elde edilmiştir.

Anahtar Kelimeler: E-Ticaret, Yazılım, Veri Madenciliği, Sahtecilik

2019, 68 sayfa

ABSTRACT

M.Sc. Thesis

FRAUD DETECTİON SYSTEM DESIGN IN E-COMMERCE SYSTEMS

Fatma Serap ÖRENLİ

Süleyman Demirel University
Graduate School of Natural and Applied Sciences
Department of Electric Electronic Engineering

Supervisor: Asst. Prof. Dr. Ufuk ÖZKAYA

As e-commerce area is rapidly growing all over the world, it's potential is also taking place and continuing it's development in our country. Companies who want to carry their work into online commerce websites are growing accordingly. Although there are many e-commerce softwares due to the high demands in the field of e-commerce, it is seen that the infrastructure deficiencies in these softwares, not being able to easily respond to customer demands, being designed in a complex structure and most importantly, the fraud controls that may occur in the orders placed are being ignored. Therefore, it should be focused on eliminating these deficiencies in the development of new software that will participate in the e-commerce sector and measures should be taken by taking fraud transactions into consideration.

In this study, it is aimed to present an alternative e-commerce infrastructure software by examining the existing systems and to propose a prediction system by using data mining classification algorithms and to make performance comparisons by using data sets in the literature in order to detect frauds of the system to be designed.

The study was carried out for two different data sets. In the first study, the performance results were evaluated with K-Nearest Neighbor Algorithm, Naive Bayes, Logistic Regression, Support Vector Machines, Random Forest and Generalized Regression Neural Network classifiers by applying feature selection approaches on credit card data consisting of 284.807 records of September 2013. Logistic Regression method, which is one of the methods used on the data set, was found to have higher success compared to the other methods. In the second study, the success of the methods for the data set consisting of 700 records was examined. Among the methods used, the best performance was obtained by Logistic Regression method.

Keywords: E-Commerce, Software, Data Mining, Fraud

2019, 68 pages

TEŞEKKÜR

Bu araştırmada beni yönlendiren, bilgi ve tecrübesi ile çalışmalarımda yardımcı olan değerli Danışman Hocam Dr. Öğr. Üyesi Ufuk ÖZKAYA'ya, teşekkürlerimi sunarım. Tez çalışmalarım sırasında beni destekleyen aileme sonsuz sevgilerimi ve saygılarımı sunarım.

Fatma Serap ÖRENLİ
ISPARTA, 2019

ŞEKİLLER DİZİNİ

Sayfa

Şekil 1.1. En popüler teknolojiler 2018 5
Şekil 1.2. Yönetici paneli dashboard görünümü 7
Şekil 1.3. Ürün tablosu veri tabanı yapısı 8
Şekil 1.4. Ürün varyasyon (seçenek) veri tabanı yapısı 9
Şekil 1.5. Seçenekli ürün site görünümü 10
Şekil 1.6. Kategori yapısı veri tabanı görünümü 11
Şekil 1.7. Kategori ağaç yapısı hiyerarşik gösterim 12
Şekil 1.8. Yönetici panelinden kategori ağaç yapısı 12
Şekil 3.1. Veri madenciliği süreçleri 23
Şekil 3.2. KNN model grafiği (1) 28
Şekil 3.3. KNN model grafiği (2) 28
Şekil 3.4. Rassal Orman yöntemi örnek gösterimi 31
Şekil 3.5. Genelleştirilmiş Regresyon Sinir Ağı 33
Şekil 3.6. Lineer Regresyon örnek gösterim (1) 35
Şekil 3.7. Lineer Regresyon örnek gösterim (2) 35
Şekil 3.8. Lineer Regresyon örnek gösterim (3) 36
Şekil 3.9. Lojistik Regresyon örnek gösterim (1) 36
Şekil 3.10. Lojistik Regresyon örnek gösterim (2) 37
Şekil 3.11. Lojistik Regresyon örnek gösterim (3) 38
Şekil 3.12. SVM örnek gösterim 39
Şekil 3.13. SVM kernel trick 40
Şekil 3.14. Veri seti-I özniteliklerin histogram gösterimi 43
Şekil 3.15. Karışıklık matrisi (confusion matrix) 45
Şekil 3.16. Karışıklık matrisi tahmin sonuç değeri gösterimi 45
Şekil 3.17. Algoritmaların “under-sampling” yöntemi ile elde edilen ROC eğrilerinin gösterimi 57
Şekil 3.18. Yapay veri setinin normalizasyon ve PCA dönüşümlerinin yapılarak elde edilen ROC eğrilerinin gösterimi 63

ÇİZELGELER DİZİNİ

Sayfa

Çizelge 1.1. Kategori ağaç yapısı örnek gösterim 11
Çizelge 3.1. Naive Bayes örnek eğitim kümesi 25
Çizelge 3.2. KNN örnek veriler 30
Çizelge 3.3. KNN örnek sonuçları 30
Çizelge 3.4. Karışıklık matrisi örnek tahminleme 46
Çizelge 3.5. Çalışmada kullanılan metrikler ve formülleri 48
Çizelge 4.1. Veri seti özellikleri 50
Çizelge 4.2. Naive Bayes uygulaması doğruluk sonuçları 50
Çizelge 4.3. Veri seti özellikleri 50
Çizelge 4.4. Naive Bayes uygulaması doğruluk sonuçları 50
Çizelge 4.5. Veri seti özellikleri 51
Çizelge 4.6. Naive Bayes uygulaması doğruluk sonuçları 51
Çizelge 4.7. Veri seti özellikleri 51
Çizelge 4.8. K-En Yakın Komşu uygulaması doğruluk sonuçları 51
Çizelge 4.9. Veri seti özellikleri 51
Çizelge 4.10. K-En Yakın Komşu uygulaması doğruluk sonuçları 52
Çizelge 4.11. Veri seti özellikleri 52
Çizelge 4.12. K-En Yakın Komşu uygulaması doğruluk sonuçları 52
Çizelge 4.13. Veri seti özellikleri 52
Çizelge 4.14. Lojistik Regresyon uygulaması doğruluk sonuçları 52
Çizelge 4.15. Veri seti özellikleri 52
Çizelge 4.16. Lojistik Regresyon uygulaması doğruluk sonuçları 53
Çizelge 4.17. Veri seti özellikleri 53
Çizelge 4.18. Lojistik Regresyon uygulaması doğruluk sonuçları 53
Çizelge 4.19. Veri seti özellikleri 53
Çizelge 4.20. RF uygulaması doğruluk sonuçları 53
Çizelge 4.21. Veri seti özellikleri 53
Çizelge 4.22. RF uygulaması doğruluk sonuçları 54
Çizelge 4.23. Veri Seti özellikleri 54
Çizelge 4.24. RF uygulaması doğruluk sonuçları 54
Çizelge 4.25. Veri seti özellikleri 54
Çizelge 4.26. GRNN uygulaması doğruluk sonuçları 54
Çizelge 4.27. Veri seti özellikleri 55
Çizelge 4.28. SVM uygulaması doğruluk sonuçları 55
Çizelge 4.29. Veri seti özellikleri 55
Çizelge 4.30. SVM uygulaması doğruluk sonuçları 55
Çizelge 4.31. Veri seti özellikleri 55
Çizelge 4.32. SVM uygulaması doğruluk sonuçları 55
Çizelge 4.33. Özellik seçimlerine göre algoritmaların başarı sonuçları 56
Çizelge 4.34. Yapay veri seti öznitelikleri 58
Çizelge 4.35. Özniteliklere göre oluşturulan kurallar 58
Çizelge 4.36. Veri seti özellikleri 59
Çizelge 4.37. Lojistik Regresyon uygulaması doğruluk sonuçları 59
Çizelge 4.38. Lojistik Regresyon uygulaması doğruluk sonuçları 59
Çizelge 4.39. Veri seti özellikleri 59
Çizelge 4.40. KNN uygulaması doğruluk sonuçları 59

Çizelge 4.41. KNN uygulaması doğruluk sonuçları 60
Çizelge 4.42. Veri seti özellikleri 60
Çizelge 4.43. Naive Bayes uygulaması doğruluk sonuçları 60
Çizelge 4.44. Naive Bayes uygulaması doğruluk sonuçları 60
Çizelge 4.45. Veri seti özellikleri 61
Çizelge 4.46. RF uygulaması doğruluk sonuçları 61
Çizelge 4.47. RF uygulaması doğruluk sonuçları 61
Çizelge 4.48. Veri seti özellikleri 61
Çizelge 4.49. SVM uygulaması doğruluk sonuçları 62
Çizelge 4.50. SVM uygulaması doğruluk sonuçları 62
Çizelge 4.51. Algoritmaların başarı sonuçları 62

SİMGELER VE KISALTMALAR DİZİNİ

AUC	ROC eğrisi altında kalan alan (Area under curve)
FN	Yanlış tahmin edilen negatif (False negative)
FP	Yanlış tahmin edilen pozitif (False positive)
GRNN	Genel regresyon sinir ağı (General regression neural network)
KNN	K en yakın komşu algoritması (K nearest neighbor)
LR	Lojistik regresyon (Logistic regression)
NB	Naive bayes sınıflandırıcısı
ROC	Alıcı çalışma karakteristiği (Receiver operating charactristics)
SVM	Destek vektör makineleri (Support vector machines)
TN	Doğru tahmin edilen negatif (True negative)
TP	Doğru tahmin edilen pozitif (True positive)
PCA	Temel bileşen analizi (Principal component analysis)

1. GİRİŞ

Günümüzde elektronik ticaret, mobil kullanımlardaki büyük artış ile yer ve zamandan bağımsız olmasının getirdiği kullanım kolaylığı sebebiyle hızlı bir büyüme sağlamıştır. E-ticaret siteleri üzerinden yapılan alışverişlerde en büyük problem ise güvenlik ile ilgilidir. Üçüncü kişiler ile paylaşılmaması gereken bilgilerin çalınması durumunda web üzerinden yapılan alışverişler sahteciliğe açık hale gelmektedir. Bunun önüne geçmek için site sahibi firmaların sahteciliğin tespitine yönelik ek önlemler almaları gerekmektedir. Bankaların kendi bünyelerinde kredi kartı sahteciliğini önleme sistemleri bulunsa da e-ticaret üzerinden yapılan bir alışverişin sahtecilik ile ilişkili olup olmadığının tespitinin yapılması site sahibi kurumlar tarafından yönetilebilir bir sistem ile yeni sipariş oluşturulduktan kısa bir süre sonra bilgilendirilerek durumdan haberdar olunması ve önlem alınması, oluşabilecek büyük zararları önlemede yardımcı olacaktır.

Bu tez çalışmasında, mevcut e-ticaret altyapı yazılımlarına alternatif, temel teknik özelliklerini içeren bir altyapı yazılımının önerilmesi ve bununla birlikte veri madenciliğinde makine öğrenmesi sınıflandırma algoritmaları kullanılarak internet üzerinden verilen bir siparişin sahte olup olmadığının tahminine dayalı bir sistem önerilmesi amaçlanmıştır. Tahmine dayalı sistem için seçilen veri setinin ve algoritmaların literatürde bu konuda yapılan diğer çalışmalar incelenerek modelin performans ölçümleri projede sunulmuştur.

Takip eden bölümlerde sırasıyla e-ticaret sistemlerinin yapısal özellikleri hakkında detaylı bilgi verilmiş, bu sistemlerde oluşabilecek sahteciliklerin tespiti için kullanılan yöntemler sunulmuştur.

1.1. Elektronik Ticaret

İşletmelerin ürün ya da hizmetlerinin satışlarının tüketicilere sunularak internet üzerinden elektronik olarak gerçekleştirilen her türlü satış işlemine e-ticaret denmektedir ve elektronik ticaret kelimelerinin kısaltılmış halidir. Perakende ticaretin yanı sıra, seyahat harcamaları, dijital uygulamaların

indirilmesi, tüketiciler ya da işletmeler arası platformlarda gerçekleşen alışverişler, kısacası sanal ortamda satın alma işlemlerinin tümü e-ticaret olarak değerlendirilir. E-Ticaret özellikle küçük ve orta ölçekli şirketler için uygun bir ticaret şeklidir. E-Ticaret çalışma şekillerine göre kendi içinde modellere ayrılmaktadır.

1.1.2. E-Ticaret Modelleri

Ticari bir kuruluşun tüketicilerine bir elektronik ağ aracılığıyla ürün ya da hizmetlerinin alım satım ve her türlü ticari işlemlerinin yapılmasına elektronik ticaret (e-ticaret) denmektedir. E-ticaret faaliyetleri, alışverişi gerçekleştiren tarafların niteliğine göre iş modellerine ayrılmaktadır.

1.1.2.1. İşletmeden İşletmeye (Business To Business)

Ürün ya da hizmet satışının işletmeden işletmeye (B2B) gerçekleştiği modeldir. Bu model, üretici firmaların ürünlerini internet üzerinden toptancılara satış yapması, bununla birlikte toptancıların perakendecilere internet üzerinden satış gerçekleştirmesidir. Genellikle büyük firmaların oluşturduğu bayilik sistemlerinde bu altyapı görülmektedir. Maliyet ve verimlilik açısından B2B modeli önemli bir yere sahiptir.

1.1.2.2. İşletmeden Tüketiciye (Business To Customer)

B2C modeli, firmaların ürün ya da hizmetlerini elektronik ortamdan müşterilerine (son kullanıcılara) sundukları alışveriş modelidir. Burada hizmeti alan taraf yalnızca son kullanıcılardır. Amaç, firma sahiplerinin mevcut ve potansiyel müşterilerine daha kolay ya da daha düşük maliyetle internet üzerinden satış imkanı sunmaktadır. Müşteri siparişlerle birlikte, ödeme için kredi kartı bilgilerini elektronik ortamda göndermektedir. Bu model, işletmeler arası gerçekleşen ticaret modeline göre hacim olarak daha küçüktür.

Yukarıda bahsedilen temel e-ticaret iş modellerine ek olarak günümüzde son kullanıcıların kendi aralarında gerçekleştirdikleri alışveriş türü olan (letgo,

gittigidiyor.com gibi) tüketiciden tüketiciye (C2C) modeli; devlet kurum ya da firmalara yönelik gerçekleştirilen işlemlerin yapıldığı yönetimden işletmeye (G2B) modeli gibi farklı e-ticaret iş modelleri ile de hizmet verilmektedir. Bu çalışma kapsamında, yukarıda bahsedilen elektronik ticaret modellerinden günümüzde çok tercih edilen B2C (Business to Customer) modeli üzerine bir yapı oluşturulmuştur.

1.1.3. Örnek Sistemler

Günümüzde e-ticaret alanında yıllık ya da aylık ücretlendirme ile hizmet veren e-ticaret paket yazılımlar ve bu yazılımlara göre daha maliyetli özel yazılımlar geliştirilmektedir. Bu sistemlerin kendi içlerinde avantajları olsa da her iki tarafta da dezavantajlar barındırmaktadır.

1.1.3.1. Hazır E-Ticaret Sistemleri

Ülkemizde bu tür yazılımlara genel talep olarak bakıldığında Tsoft ve Ticimax şirketleri örnek olarak verilebilir. Yaygın olarak kullanılan bu tür hazır e-ticaret yazılımlarının sunduğu avantajlar; kurulum, teknik destek, hazır tema seçeneklerinin bulunması yanı sıra karmaşıklık seviyelerinin yüksek olması, son kullanıcı açısından karmaşık bir yapıyı yönetmesi beklenmektedir. Maliyet açısından düşünüldüğünde paket sistem ile çalışan yazılımların maliyetleri özel yazılımlara nispeten daha düşüktür.

1.1.3.2. Özel Yazılım Sistemleri

Firmaya özel yapılan e-ticaret yazılım hizmetleri, firmanın iş süreçleri ve iş akışına göre planlanıp, bulunduğu sektöre ve tamamen söz konusu firmanın ihtiyaçlarına özgü hazırlanan projeleri kapsar. Maliyet olarak paket sistemlere göre yüksektir ancak firmaya özel olduğu için iş süreçlerinin takip edilmesi ve daha verimli kullanılması açısından avantajlı olduğu düşünülebilir.

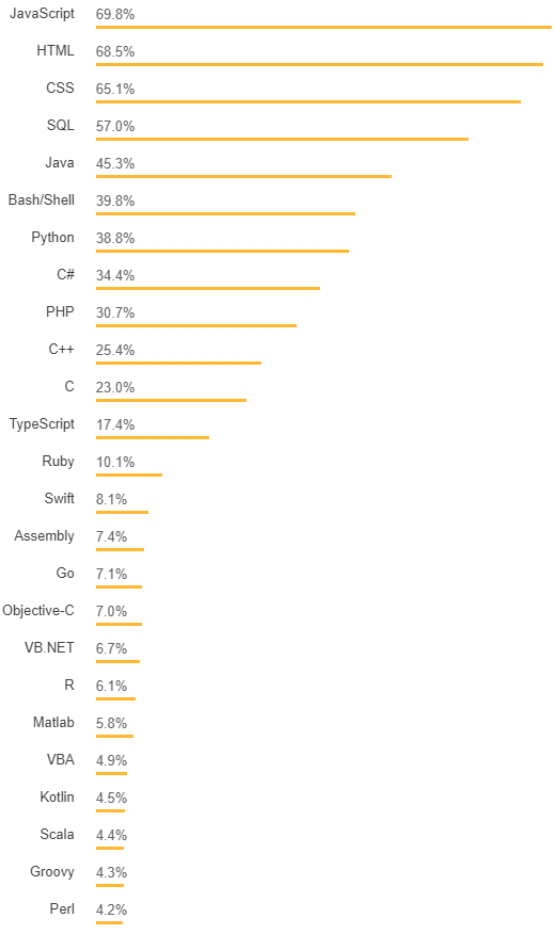
1.1.4. Önerilen Sistem

1.1.4.1. Teknik Altyapı

Bir e-ticaret yazılımının altyapısındaki en önemli özelliklerden birisi modüler olarak tasarlanmasıdır. Yazılımın karmaşıklık seviyesi arttıkça daha fazla ihtiyaç haline gelen modülerlik, gelişen e-ticaret süreçlerine uyum sağlanması, müşteri taleplerinin hızlı bir şekilde hayata geçirilebilmesi için sistemin anlaşılabilir küçük parçalar (modüller) halinde tasarlanması, geliştirici açısından büyük bir öneme sahiptir.

Modülerlik, bir bilgisayar programının alt programlara ayrılarak soyutlanması ve yeniden kullanılabilirlik gibi yazılım mühendisliğinin temel kavramlarının hayata geçirilmesidir. Modüler tasarımla bir modülün çıkarılması, bakımının yapılması ya da yerine yeni bir modülün eklenmesi gerektiğinde, sistemin diğer bileşenlerinin bu gibi işlemlerden etkilenmemesi amaçlanmaktadır.

E-ticaret altyapısında seçilen programlama dili ve teknolojiler, geliştirme süreçlerini doğrudan etkileyeceği için oldukça önemlidir. Şekil 1.1'de 2018 yılında dünyada en çok tercih edilen programlama dilleri gösterilmektedir (Stack overflow, 2019). Bu bilgilere göre son 6 yılda en çok tercih edilen programlama dili JavaScript; Python dilinin ise C# programlama dilinin önüne geçtiği görülmektedir. Python en hızlı büyüyen programlama dili seçilirken C# ise yine en çok tercih edilen diller arasında yer almaktadır.



Şekil 1.1. En popüler teknolojiler 2018 (Stack overflow, 2019)

Uygulama Programlama Dili

Bu çalışmada yazılım altyapısının hazırlanmasında Microsoft teknolojilerinden Asp.Net MVC ortamında C# programlama dilinin uygulanması tercih edilmiştir. MVC (Model-View-Controller) yapısında mantıksal bir gruplama yaklaşımıyla modüler geliştirmeye olanak sağlayan Areas (Code Project, 2019) kullanılmıştır. Veri tabanı yönetim sistemi olarak Microsoft SQL 2014 kullanılmıştır. Sahtecilik tespit işlemleri için belirlenen sınıflandırma algoritmalarının uygulamaları yine Visual Studio 2017 ortamında; Python programlama dili kullanılarak gerçekleştirilmiştir.

Arayüz Tasarımı

Arayüz tasarımı, sunucu tarafında çalışan yazılımın işlevselliğinin son kullanıcıya ön yüzde görsel olarak sunulmasıdır. Arayüz tasarımında isteğe bağlı olarak farklı araçlar tercih edilebilir. Burada kullanıcı deneyimi (UX) ve arayüz tasarımı (UI) için ön yüz geliştirme araçlarının doğru kullanılması, görünümün yanı sıra SEO açısından da önemlidir. Bu açıdan bir çok sistem HTML5 ve JavaScript araçlarını kullanmaktadır. Bu çalışmada arayüz geliştirmede CSS desteği ile HTML5, arayüz tarafından sunucu taraflı kodlarla iletişimde de JQuery kullanılmıştır.

Mobil Uyumluluk

Akıllı telefonlar, tabletler ve mobil uygulamaların yaygınlaşmasıyla mobil üzerinden e-ticaret sitelerine erişimde büyük bir artış olduğundan, sistemin farklı cihazlardan erişimlerinin dikkate alınarak tasarlanması öngörülmüştür. Bu anlamda diğer sistemler tarafından hazır temalar yaygın olarak kullanılmaktadır. Bu çalışmada ise mobil kullanım kolaylığı ile birlikte Bootstrap tasarım aracı desteği kullanılarak hem yönetim panelinde hem de müşterinin satın alma işlemlerini gerçekleştirdiği e-ticaret sitesi tarafında sıfırdan bir arayüz oluşturularak tüm cihazlar ile uyumlu hale getirilmiştir.

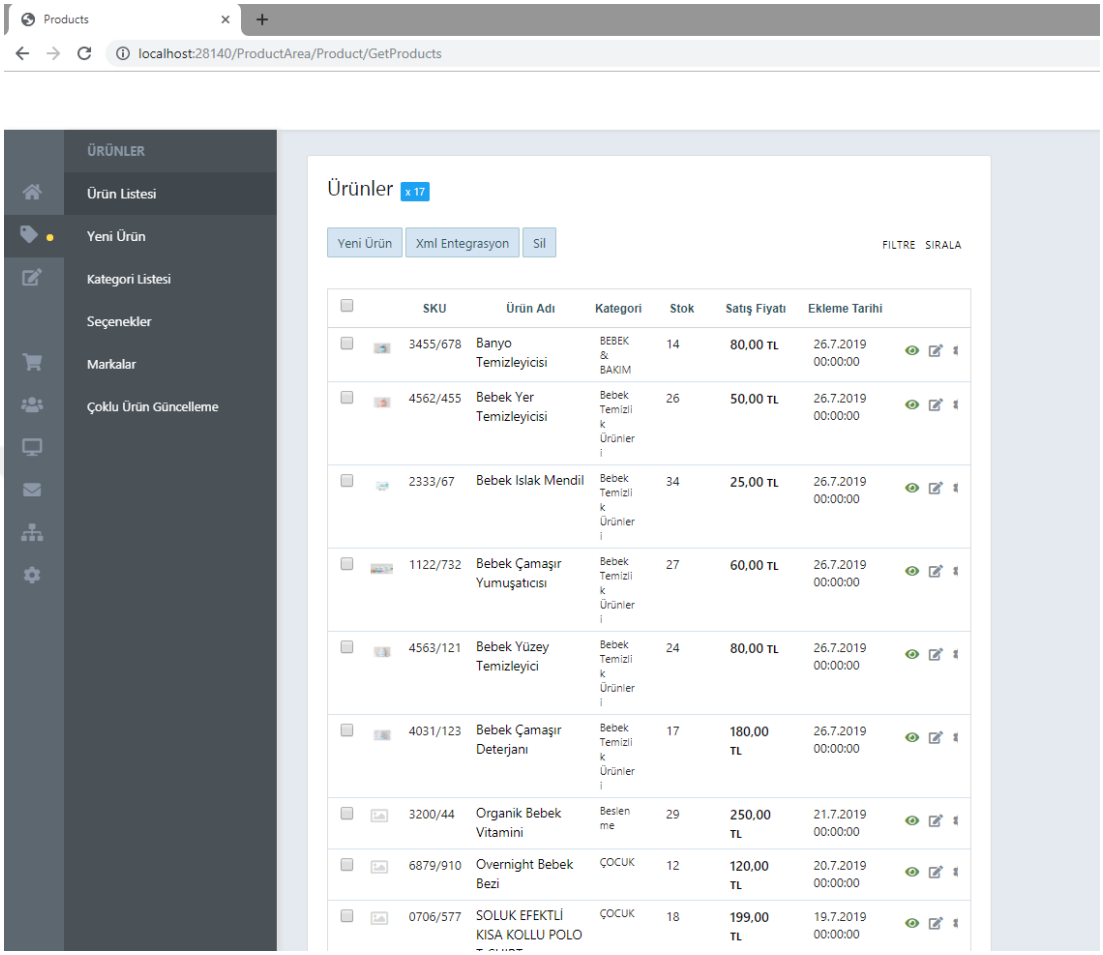
1.1.4.2. Yönetim Paneli

Yönetim paneli, bir e-ticaret sitesinin temel gereksinimleri düşünülerek oluşturulmuştur. Site sahibi firma tarafından sitede bulunan tüm özelliklerinin yönetildiği, düzenlenebildiği ve e-ticaret süreçlerinin takip edilebildiği bir "dashboard" görünümündedir. Bu bölümde çalışmada oluşturulan e-ticaret yönetim panelinin temel özellikleri tanıtılmaktadır.

1.1.4.3. Ürün Yönetimi

Ürün yönetimi, ürünlerin teker teker ya da bir xml entegrasyon ile toplu olarak girişinin yapıldığı bölümdür. Oluşturulan ürün ve içerik yapısı, temel bir ürün girişinin yapılarak, ürüne ait ek seçenekler varsa ürüne özel ek alanların

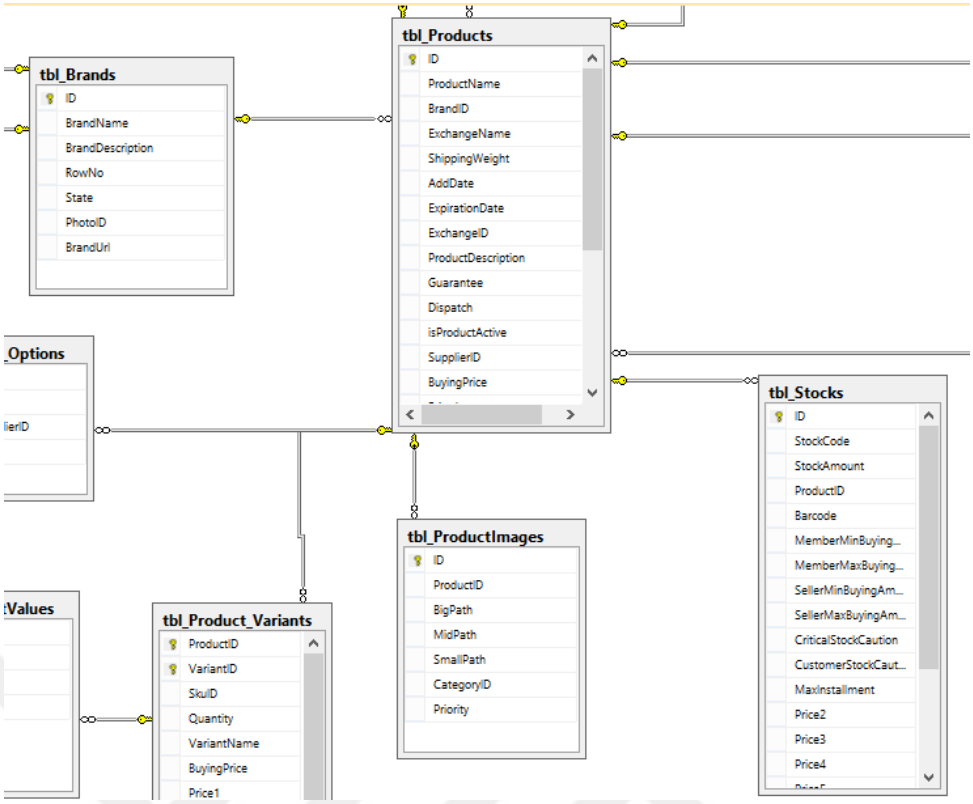
oluşturulmasını; böylece ürünlerin çeşitlendirilmesini sağlamaktadır. Yönetici paneli ürün yönetimi görünümü Şekil 1.2'de gösterilmektedir.



Şekil 1.2. Yönetici paneli dashboard görünümü

1.1.4.4. Seçeneksiz Ürün İşlemleri

Ürünün tek tipte bir ürün olduğu durumlarda temel ürün şablon girişinin yapılarak herhangi bir seçenek değeri eklenmeden ürün girişinin yapılmasıdır. Şekil 1.3'te temel bir ürün tablosu veri tabanı diyagramı gösterilmektedir.



Şekil 1.3. Ürün tablosu veri tabanı yapısı

Şekil 1.3'te;

tbl_Products : Ana ürün tablosu,

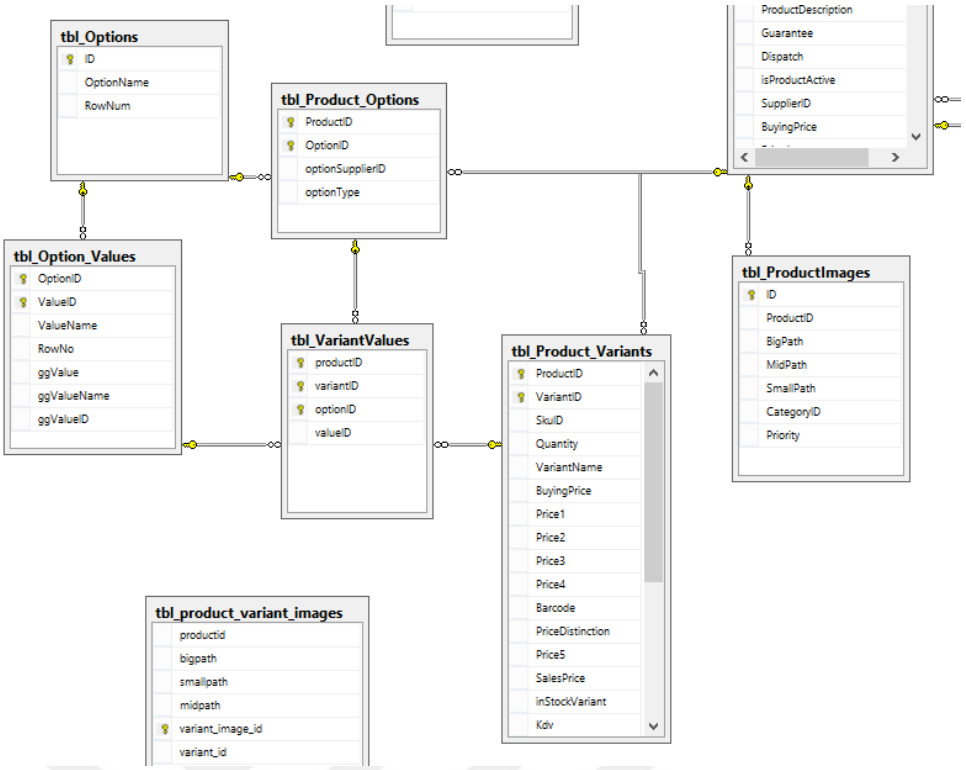
tbl_Brands : Markalar tablosu,

tbl_Product_Images : Ürün resimlerinin tablosu,

tbl_Stocks : Ürün stok tablosudur.

1.1.4.5. Seçenekli Ürün İşlemleri

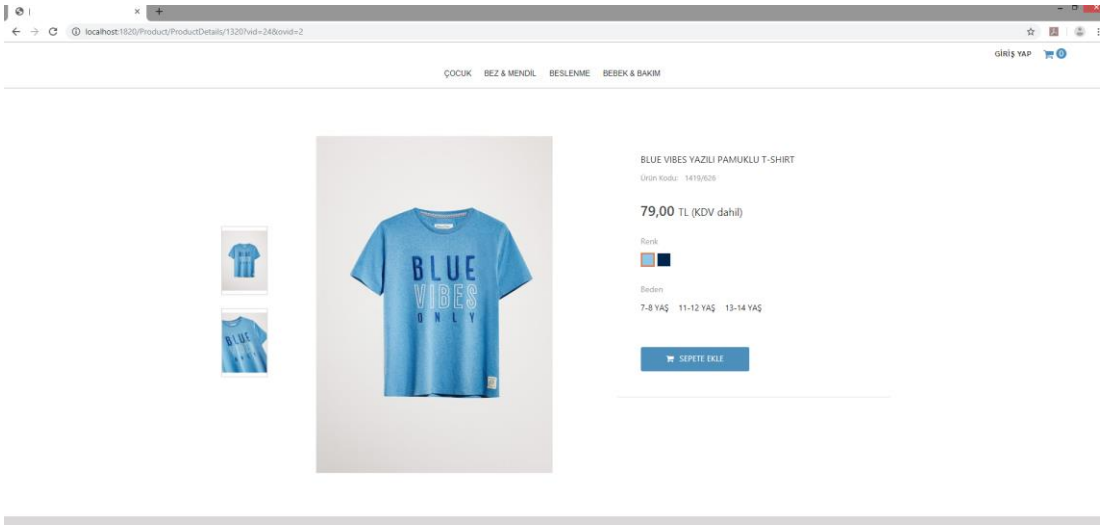
Ürüne ait farklı seçeneklerin bulunması durumunda (renk, beden, … gibi) sisteme önceden tanımlanmış seçenek tipleri ve değerleri ürüne atanmaktadır. Ürüne atanan seçenek tipleri üzerinden ürüne özel seçenekler seçilerek kayıt tamamlanır, böylece ürün seçenekleri oluşturulmuş olur. Ürün ve seçenekli ürün tabloları Şekil 1.4'te gösterilmektedir.



Şekil 1.4. Ürün varyasyon (seçenek) veri tabanı yapısı

Şekil 1.4'te; tbl_Options : Ürün seçenek tipi,
tbl_Option_Values : Ürün seçenek değerleri,
tbl_Product_Options : Ürüne atanan seçeneklerin tutulduğu tablo,
tbl_Product_Variants : Seçenekli ürün varyasyon tablosu,
tbl_VariantValues : Seçenekli ürüne ait seçenek tipi ve seçenek değerlerini tutan tablodur.

Böyle bir veri tabanı yapısı aracılığıyla tıpkı ana üründe olduğu gibi fiyat, stok miktarı, stok kodu, barkod kodu, indirim tutarı, ürün ismi, ürün resmi gibi ürüne özel bilgiler, ürünün tüm seçenek kombinasyonlarının bulunduğu "tbl_Product_Variants" tablosunda tutularak firma sahibinin isteğine bağlı olarak her ürün varyasyonuna özel olarak tanımlanabilmektedir. Şekil 1.5'te site tarafında seçenekli ürün detayının görünümü yer almaktadır.

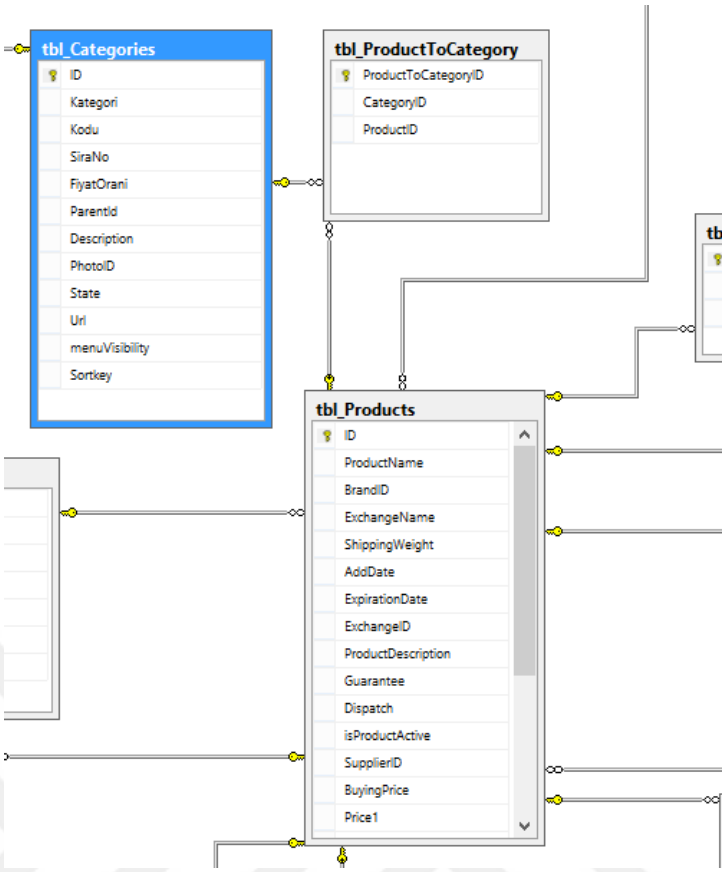


Şekil 1.5. Seçenekli ürün site görünümü

1.1.4.6. Kategori Yönetimi

Bir e-ticaret sitesinin en önemli bileşenlerinden birisi de kategorilerdir. Bir ürünün birden fazla ya da sınırsız sayıda kategori altında olabileceği düşünülerek tasarlanması gerekmektedir. Böylece ürünler site üzerinden yapılan filtrelemelerde birden fazla kategori altında listelenebilmekte ve görüntülenebilmektedir.

Çalışmada oluşturulan kategori veri tabanı yapısı Sekil 1.6'da görüldüğü gibidir. Veri tabanında ana kategori ve alt alta kategoriler halinde "parent-child" mantığında üst alt ilişkisi ile tasarlanmıştır.



Şekil 1.6. Kategori yapısı veri tabanı görünümü

Çizelge 1.1'de;

tbl_Categories : Kategori tablosunu,

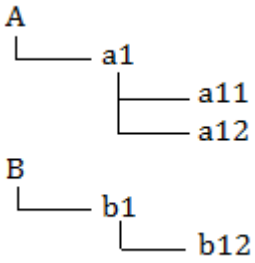
tbl_ProductToCategories : Ürün ve kategori bilgilerinin tutulduğu ortak tabloyu,

tbl_Products : Ürün tablosunu

temsil etmektedir. Tablolar arasında bire çok ilişki bulunmaktadır.

Çizelge 1.1. Kategori ağaç yapısı örnek gösterim

Id	text	parentid
1	A	null
2	B	null
5	a1	1
6	b1	2
9	a11	5
10	a12	5
11	b12	6



Şekil 1.7. Kategori ağaç yapısı hiyerarşik gösterimi

Yönetim paneli üzerinden kategoriler menüsünde tut-sürükle-bırak yöntemi ile kategori alt üst ilişkisi değiştirilebilmektedir. Ayrıca belirli bir kategori altında bulunan ürünlere gidilebilmekte, kategori üzerine gelerek düzenlemeler yapılabilmektedir. Şekil 1.7 ve Şekil 1.8'de yönetim panelinden kategori ağaç yapıları örnek olarak gösterilmektedir.

Şekil 1.8. Yönetici paneli kategori ağaç yapısı

Yönetim panelinde mevcut tanımlanmış kategori, marka, içerik sayfası ya da özel bağlantılarının (link) e-ticaret sitesinde bulunan menü içeriğinde görüntülenebilmesi için menü tipi seçilerek menünün tanımlanması gerekmektedir. Tanımlanan menü aktif olarak işaretli olduğu sürece web sitesi tarafında görüntülenebilecektir. Kategori alt üst yapısına benzer şekilde, tut-

sürükle-bırak yöntemiyle menü bağlantılarının yerleri değiştirilebilir, alt menü eklenebilir ya da silinebilmektedir.

1.1.4.7. Arama Motoru Optimizasyonu Yönetimi

Arama motoru optimizasyonu (SEO), bir internet sitesinin teknik düzenlemelerle arama motorlarında üst sıralara çıkarılması amacıyla yapılan çalışmaları kapsamaktadır. Doğru bir SEO yönetimi satış yapan firmayı aramalarda daha geniş kitlelere ulaştırmakla birlikte satışlarını da buna bağlı olarak artıracaktır. Mevcut sistemlerin hemen hepsinde bu modüle yer verildiği görülmektedir. Bu çalışmada mevcut sistemlerdekine benzer şekilde, toplu meta girişleri, ürüne özel anahtar kelimelerin tanımlanması, genel SEO ayarlarından tüm ürünlere otomatik tanımlanan anahtar kelimelerin girilmesi; yine ürün sayfasından ürüne özel olarak düzenlenebilmektedir.

1.1.5. Güvenlik

Bu başlıkta veri tabanı güvenliğine yönelik önlemler, web ödeme sistemi entegrasyonuna ait temel kavramlar, sahtecilik türleri ve önleme yöntemleri açıklanmıştır.

1.1.5.1. Veri Tabanı Güvenliği

Veri tabanına yönelik sızma saldırıları (SQL Injection) günümüzde de halen bir tehdit olarak yerini korumaktadır (Code curmudgeon, 2019). Sızma, bir veri tabanı sistemine yönelik istenmeyen sorguların çalıştırılarak zarar vermeyi amaçlayan bir saldırı tekniğidir. Saldırıyı yapan kişi, uygulamadan sızabileceği yerleri saldırıya geçmeden önce belirleyerek veri tabanına sızarak zarar verebilmektedir. Örnek olarak bu tip saldırılar, ‘select * from products’ SQL cümlelerinde araya yerleştirilen bir meta karakter ile sızmaya sebep olabilmektedir. Bu tür saldırılar çeşitli yöntemler ile önlenebilmektedir. Uygulamanın veri tabanı bağlantılarının tutulduğu “WebConfig” dosyasında bulunan “connectionString” bölümünün şifrelenmesi, hata mesajlarının kullanıcıya gösterilmeden kayıt günlüğünün (log) tutulması, SQL sorgularında

saklı yordam (stored procedure) kullanarak parametreli sorgular çalıştırılması, “drop database” gibi veri tabanını yok etmeye yönelik ifadelerin engellenmesi, geliştirici yetkisine göre kullanıcı yetkilendirmesinde en az düzeyde yetki verilerek erişimin sınırlandırılması gibi önlemler sızma saldırılarına önemli ölçüde fayda sağlamaktadır.

1.1.5.2. Web Ödeme Entegrasyonları ve Temel Kavramlar

Bu aşama site sahibi firmanın tercihine göre belirlenmektedir. Ödeme işlemleri sanal pos entegrasyonu yapan firmalar ile anlaşılarak ya da bankaların ayrı ayrı sanal pos entegrasyonlarının yapılması ile tercih edilebilmektedir. Ayrı bankalardan alınacak sanal pos entegrasyonları maliyet olarak yüksek olduğundan İyzico, Payu gibi tek seferde tüm bankalar ile alışveriş yapılmasına olanak sağlayan hizmetler tercih edilebilmektedir.

Güvenli Giriş Katmanı (SSL)

Sunucu ile alıcı arasındaki verilerin şifrelenerek yapılmasını sağlayan bir protokoldür. Web sitelerinde özel bilgi gönderiminde güvenli bağlantı kurulmasını sağlayan, standart bir algoritmadan oluşan SSL, sunucu tarafında bir anahtar ve alıcı tarafında bir sertifika ile çalışmaktadır. Özellikle alışveriş sitelerinde bulunması zorunludur. SSL sertifikasına sahip sitelere girildiğinde adres satırının başında bulunan “https://” yazısının yeşil renkte olması o sitenin güvenlik sertifikasının bulunduğunu belirtmektedir.

Güvenli Alışveriş (3D Secure)

Banka kartı ile internetten alışveriş yapılabilmesi için bir zorunluluktur. Hizmeti satın alacak müşteri firmanın internet sitesinden kart bilgilerini girdikten ve bu işlemi onayladıktan sonra müşteri bilgileri şifrelenerek ilgili bankaya gönderilmektedir. Banka, gelen bilgileri deşifre ettikten sonra müşterinin kayıtlı cep telefonu numarasına şifre göndererek, müşterinin bu şifreyi yönlendirildiği bankanın ekranına girmesiyle doğrulama yapılarak alışveriş tamamlanmaktadır. Güvenli alıiveriş ile müşteri ve banka arasında

telefon yoluyla doğrulama yapıldığından çalıntı bir kredi kartı ile oluşabilecek saldırıların önüne geçilmiş olmaktadır.

Alıcı Banka (Acquirer Bank)

Ticari firmaların ürün ya da hizmetleri karşılığında ödeme almalarına aracı olan finans kuruluşları için kullanılan bir terimdir. POS cihazı üzerinden müşterinin kredi kartından ödeme alarak anlaşmaları doğrultusunda satıcıya aktarır.

Issuer Bank

Bu terim, kredi kartını müşteriye veren banka için kullanılan bir terimdir.

POS (Point of Sales Terminal) ve Sanal POS (VPOS)

POS, Müşterinin ödeme noktalarında nakit kullanmadan ödeme yapmasını sağlamaktadır. Ödemede kullanılan kredi kartının üzerinde bulunan çip ile bilgileri okuyarak kartın gerçek olup olmadığı, kart hesabının uygunluk durumu, hangi üye işyeri üzerinden, hangi acquier banka tarafından, hangi issuer kart ile yapıldığı bilgilerini ilgili merkeze ileten, tutarı karttan tahsil ederek ödeme işlemini tamamlayan elektronik cihazlardır. Sanal POS ise, POS cihazlarının internet üzerinden yapılan alışverişlerde kullanılmak üzere uyarlanmış yazılım halidir. Bu şekilde müşterinin yaptığı ödemeler üye işyeri hizmetlerinden ve VPOS sisteminden yararlanılan bankadaki firma hesabına geçmektedir.

Şifre (PIN)

Şifre, banka kartının alışverişlerde kullanılan kimlik parola numarasıdır. Bu şifre tuşlanarak ödemeler onaylanmaktadır.

1.1.5.3. Sahtecilik Türleri ve Önleme Yöntemleri

Sahtecilik (Fraud) tespiti, kredi kartı sahibinin kart bilgilerinin kullanılarak yapılan harcama işlemlerinin izinsiz olarak başkaları tarafından yapılıp yapılmadığının tespit edilmesidir. Genel olarak iki türde sahtecilik tespiti bulunmaktadır. Bunlar suistimal tespiti (misuse detection) ve anomali tespitidir (anomaly detection). Anomali tespitinde kart sahibinin geçmiş harcama işlemlerinin verilerine ihtiyaç bulunmaktadır. Bu geçmiş veriler ile gelen işlem verisi karşılaştırılarak normalin dışında seyreden bir durumun olup olmadığı tespit edilmeye çalışılmaktadır. Buna anomali tespiti adı verilmektedir. Bu yöntemde kart sahibine ait kart hareketlerinin; birbirini izleyen işlem veri örnekleri, kart sahibinin normal işlem davranışının karakterize edilebilmesi için yeterli miktarda bulunmasını gerektirmektedir. Suistimal tespiti ise, sınıflandırma yöntemleri kullanılarak gelen bir işlemin sahte (fraud) ya da normal (genuine) olup olmadığının tespit edilmesine denir. Bu yaklaşım genellikle, modelin farklı sahte örüntüleri tanıyabilmesi için var olan sahte işlem tiplerinden öğrenmeye ihtiyaç duymaktadır.

Ticarette kredi kartı sahteciliği teşebbüsleri kartın fiziksel olarak mevcut olduğu (card present fraud) ya da sanal olarak (card not present) yapılan türleri olmak üzere ikiye ayrılmaktadır. Fiziksel kredi kartı sahteciliğinde kart fiziksel olarak ele geçirilerek ya da banka kartı bilgileri kopyalanarak yapılmaktadır, sanal ortamda ise yine izinsiz ele geçirilen kart bilgileri gerçek kart sahibi gibi girilerek yapılmaktadır. E-ticaret siteleri üzerinden yapılan sahtecilikler sanal ortamda yapılan sahtecilik türü kategorisindedir. Bu sahtecilikler, firmaları büyük maddi zararların yanı sıra itibar kaybına da yol açmaktadır. Bu tez çalışması kapsamında sanal ortamda yapılan sahteciliklerde suistimal tespitine yönelik bir uygulama önerilmiştir.

Sahteciliği Önleme Yöntemleri

Kredi kartı sahtecilik tespitinde örnekler sahte ya da gerçek olarak ikili (binary) sistemde etiketlenerek sınıflandırılmaktadır. E-ticaret sitelerinde oluşabilecek sahtecilikler, web sitesi üzerinden oluşturulan bir sipariş kaydının özelliklerinin

incelenerek sahte olup olmadığı tespit edilebilmektedir. Bu da müşteri tarafından oluşturulan siparişin belirli karakteristik özelliklerine bakılarak yapılmaktadır. Bunlar sahtecilik işlemini gerçekleştiren kişilerin yapmaya eğilimli oldukları davranış türleri düşünülerek incelenebilir.

- Sipariş verilen kargo adresinin fatura adresinden farklı olması riskli olarak değerlendirilir.
- Teslimat adresi, fatura adresi ve IP adresinin tümünün birbiriyle yakın olması daha güvenlidir. Bu adresler arasındaki mesafelerin çok uzak olması sahtecilik ihtimalini artırmaktadır. Bunun yanında fatura ve teslimat adreslerinin farklı olması da siparişin risk değerini artıran bir faktördür.
- Sahteciler genellikle, kendi IP adreslerini gizleyerek hareket edebilecekleri için IP adresinin yasal olup olmadığının belirlenmesi gereklidir.
- İlk defa alışveriş yapan müşteriler risklidir.
- Hızlı gönderi tercih edilmesi, tek işlemde normalden fazla miktarda yapılan harcama girişimleri, aynı üründen fazla miktarda sipariş verilmesi yine riskli olarak değerlendirilir.
- Normal olmayan uzun e-posta adreslerinin kullanılması, kolayca oluşturulabilen ve tek kullanım için ideal olan ücretsiz e-posta hizmetlerinden alınmış e-postalar çok tercih edilmektedir. Bu sebeple gmail, yahoo gibi isim uzantısına sahip adresler, iş alan adı uzantılarına sahip e-posta adreslerine göre daha risklidir.
- Düşük fiyatlı ürünler satın alınarak kart bilgilerinin doğrulanması amacıyla kartın test edilmeye çalışılması gibi özellikler sahtecilerin sıklıkla başvurduğu yöntemler olarak bilinmektedir ve bu özellikleri taşıyan işlemler riskli olarak değerlendirilerek sipariş oluşturulduktan sonra incelenerek kontrol altına alınmalıdır.

1.1.6. Problemin Tanımı

E-ticaret altyapı sistemlerinin yaygınlaşması günümüzde çok sayıda firmanın ürün ya da hizmetlerinin satışını internet üzerinden yapabilecekleri bu programların kullanımı yaygınlaşmıştır. Ülkemizde özel yazılım ve paket

yazılım çözümleri, site sahibi firmalara hizmet vermektedirler. E-ticaret süreçlerinin takip edilmesi, yönetilmesi, her hizmete cevap verecek nitelikte olması gerektiğinden son kullanıcılar son derece karmaşık sistemleri kullanmaktalardır. Çok sayıda alternatifi bulunan bu sistemlerin belirli durumlarda yetersiz kalması ise kullanıcıları kendilerine daha uygun çözümlerin arayışına itmektedir.

E-ticaret sitelerinden yapılan alışverişlerde oluşabilecek suistimaller (misuse) ile de günümüzde yaygın olarak karşılaşılabilmektedir. Bu sistemlerin kendi içlerinde sahtecilik önleme mekanizmaları barındırmaları, oluşabilecek güvenlik problemlerini önlemede yardımcı olacaktır. Fakat böyle sistemlerin geliştirilmesi, hassas veriler ile çalışılmasını gerektirdiğinden bu alanda yapılan çalışmalar yaygın olarak yapılamamaktadır. Eğer sahtecilik önleme mekanizmaları böyle sistemlerde uygulanabilirse, oluşabilecek güvenlik sorunlarına karşı tedbirler artırılmış olacaktır.

Bu çalışmanın amacı piyasada mevcut kullanılan e-ticaret sistemlerinin incelenmesi ile temel teknik özellikleri barındıran alternatif bir e-ticaret sistemin önerilmesi, bununla birlikte internet üzerinden yapılan alışveriş verileri üzerinde sahteciliklerin tespitine yönelik bir sistem önerilmesidir. Önerilen sistemin, literatürde yapılmış çalışmalar incelenerek; veri madenciliği sınıflandırma algoritmalarından Naive Bayes, KNN, Lojistik Regresyon, SVM, Rassal Orman, GRNN sınıflandırıcıları kullanılarak performansları python ortamında uygulanarak değerlendirilmiştir.

2. LİTERATÜR ÖZETLERİ

Zareapoor (2015), Bagging Ensemle sınıflandırma yöntemini kullanarak kredi kartı verileri üzerinde anormal işlemlerin tespitine yönelik bir uygulama önerilmektedir. Bu yöntemin Naive Bayes, KNN, SVM yöntemleri ile performansları karşılaştırılmıştır. Sınıflandırıcılar Kalifornia Üniversitesi ve FICO iş birliği ile düzenlenen yarışmadan elde edilen gerçek e-ticaret kredi kartı verileri üzerinde denenmiştir. 98 gün içerisinden 100.000 kayıttan oluşan veri seti 20 alan içermektedir ve banka tarafından 1/0 olarak etiketlenmiş verilerdir. Veri setinde %2,8 (2293/100.000 kayıt) sahte işlem, %97,2 oranında yasal işlem olduğu belirtilmektedir. Veri seti dengesiz yapıda olduğu için eğitim aşamasında 4 parçaya (D1, D2, D3, D4) ayırılmıştır. Verilerin sahtecilik oranları sırasıyla yaklaşık %20, %15, %10, %3 oranındadır. Sınıflandırma başarısı "Fraud Catching Rate", "False Alarm Rate", "Balanced Classification Rate" ve "Mathews Corelation Coefficient" metrikleriyle ölçümlenmiştir. Sonuçlar, sahtecilik tespitinde karar ağacına dayalı Bagging Ensemble tekniğinin diğer yöntemlere göre daha başarılı olduğu; bu sonucun elde edilmesinde Bagging Ensemle tekniğinin veri özelliklerinden bağımsız çalışması ve dengesiz veri setleri üzerinde başarılı bir model olmasından dolayı sahtecilik problemleri üzerinde daha etkili olduğu sonucuna varıldığı belirtilmiştir.

Sharma (2018), öznitelik seçimi ve birliktelik (Ensemble) tekniğine dayalı kredi kartı sahtecilik tespiti için Naive Bayes, KNN, Rassal Orman, Karar Ağacı (J48) sınıflandırıcıları ile, Eylül 2013 yılına ait Avrupalı kredi kartı sahiplerinin işlem bilgilerini içeren bir veri seti kullanılarak bir yaklaşım sunulmaktadır. Veri setindeki gürültülü veriler Genetik Algoritma ve Yapay Arı Koloni algoritmaları uygulanarak öznitelik seçimi yapılmış ve Ensemble teknikler uygulanmıştır. Oldukça dengesiz olan bu veri seti üzerinde yapılan öznitelik seçiminin sınıflandırma algoritmaları üzerindeki etkisi analiz edilmiştir. Yapılan analiz sonucunda üç modelle de en iyi çalışan tekniğin "Bagging Ensemle" olduğu sonucuna varılmıştır. Önerilen yöntemin veri seti üzerinde tahminlemede daha fazla doğrululuk (accuracy), model oluşturma ve hesaplama zamanı olarak da daha az zaman aldığı belirtilmektedir.

Yee (2018), kredi kartı sahtecilik tespiti alanında popüler Bayes sınıflandırıcılarından K2, Tree Augmented Naive Bayes (TAN), Naive Bayes, Lojistik Regresyon ve J48 sınıflandırma modellerinin performansları k katlamalı çapraz doğrulama yöntemi uygulanarak değerlendirilmiştir. Çalışmada kullanılan veri seti, kredi kartı sahteciliği ile ilişkilendirilebilecek belirli karakteristik özellikler olan kredi kartı numarası, referans numarası, terminal id, asıl pin, girilen pin, işlem tutarı, lokasyon, işlem gün ve zamanı, fatura adresi, teslimat adresi gibi alanlar düşünülerek yapay veri seti, "generatedata.com" web adresi üzerinden otomatik olarak üretilmiştir. Üretilen yapay verilerin model üzerinde etkin olabilmesi için WEKA veri işleme araçları kullanılarak PCA dönüşümü yapılmıştır. Araştırmada oluşturulan yapay veri setinin ham hali ile işleme aşamasından sonra elde edilen veriler değerlendirildiğinde; yapay bir veri setinin veri madenciliği sınıflandırma problemlerinde etkili olabileceği, fakat ham veri setinden ziyade ön işleme alınmış verilerin modellerin performanslarını belirgin ölçüde arttığı gözlemlenmiştir. Bu modellerden Tree Augmented Naive Bayes (TAN) yöntemi, işlenmemiş veriler üzerinde %84,0 doğruluk oranı ile en yüksek, işlenmiş veriler üzerinde ise %99,7 en yüksek doğruluk oranıyla diğerler modellere göre daha etkili olduğu belirtilmektedir.

Sudha (2017), KNN algoritmasının modellenerek, sınıflandırma performansının kredi kartı doğrulaması yapan bir uygulama üzerinde ve k değerinin farklı değerler ile denenmesiyle KNN sınıflandırma oranının artırılması üzerinde bir çalışma önerilmiştir. Önerilen yöntemin sahtecilik içeren işlemlerin azalmasında ve yanlış alarm sayısının da en aza indirilmesinde etkili olduğu belirtilmektedir. Kullanıcıların geçmiş harcama işlemlerinin; kredi kartı numarası, işlem tutarı ve son satın alma işleminden bu yana işlem zamanı gibi karakteristik bilgiler kullanılmıştır.

Alam ve Pachauri (2017), J48 Karar Ağacı, Naive Bayes Network ve One-R sınıflandırıcılarını kullanarak Weka platformunda kredi kartı veri seti üzerinde sahtecilik tespitine yönelik sınıflandırma yapılmıştır. Bu üç sınıflandırıcının farklı Weka parametreleri üzerinden verimlilikleri de karşılaştırılmıştır. Çalışmada Professor Dr. Hans Hofmann tarafından önerilen 1000 örnek içeren

Alman kredi kartı sahtecilik verileri kullanılmıştır. Veri dosyası düzenlenerek birkaç kategorik özellik tamsayı olarak değiştirilmiştir; böylece kategorik değişkenlerle çalışmayan algoritmalar ile uyumlu hale gelmesi sağlanmıştır. Sonuçlar, J48 yönteminin model oluşturmada daha az zaman alması, örnekleri doğru sınıflandırması ve tahmin doğruluğunun da diğer 2 yönteme kıyasla daha yüksek olmasından dolayı J48 yönteminin kredi kartı veri seti üzerinde daha iyi performansa sahip olduğu belirtilmiştir.

Xuan (2018), RF-I ve RF-II olmak üzere iki Rassal Orman modeli, üç deneyde uygulanarak sınıflandırma performansları değerlendirilmiştir. Çalışma Çin'de bir B2C e-ticaret firmasından elde edilen 30.000.000 kayıttan oluşan gerçek veriler ile uygulanmıştır. 62 öznitelikten oluşan veri setinde zaman, tutar, lokasyon gibi bilgiler bulunmaktadır. 82.000 işlem sahte olmak üzere tüm verilerde %27 oranındadır ve veri setinde sahte/yasal oranı olarak oldukça dengesizdir. İlk deneyde, iki modelin etkinliğinin ölçülmesi amacıyla Ocak 2017 yılına ait tüm sahte işlemler ve 150.000 yasal işlem, tüm yasal işlemlerden rasgele seçilerek veri seti örnekleri dengelenmiştir. 1. deneyde RF-II performansı daha iyi olarak belirlenmiştir. 2. deneyde, RF-II modeli under-sampling yöntemi ile veriler dengelenerek uygulanmıştır. 3. deneyde, tüm veri setine yakın sayıda örnek kullanılarak ilk deneye benzer şekilde tüm sahte işlemler ile 5:1 oranında sahte/yasal olarak rastgele seçilmiştir. Test veri seti 4.7 milyon yasal 30.000 sahte işlemden oluşmaktadır. Sonuç olarak RF algoritmasının performansının küçük sayıda veriler üzerinde iyi olduğu fakat dengesiz veri setleri üzerinde hala problemli olduğu ve algoritmanın kendisinin geliştirilmesi gerektiği belirtilmiştir.

Lima (2015), "PagSeguro" ismindeki bir web ödeme sistemine ait işlemlerin bulunduğu gerçek bir veri seti üzerinde, Gain Ratio, CFS ve Relief özellik çıkarım teknikleri ile Bayes Ağları, Lojistik Regresyon, Karar Ağacı-J48 ve SVM sınıflandırıcıları uygulanmıştır. Performansların değerlendirilmesinde F-ölçütü ve Ekonomik Etkinlik (Economic Efficiency-EE) metriklerine bakılmıştır. Sonuçlar, veri setindeki sınıf dağılımındaki dengesizliğin özellik seçimi tekniklerinin etkinliğini düşürdüğünü, "under-sampling" stratejisi

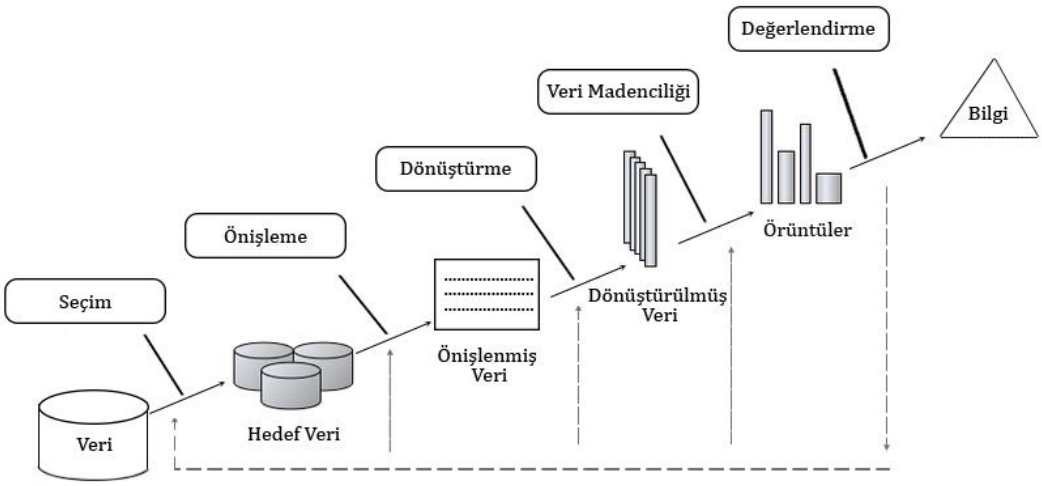
uygulandığında ise sonuçların olumlu etkilendiği belirtilmiştir. Çalışma sonuçlarında en yüksek başarı oranına “Bayes Relief” yönteminin sahip olduğu belirtilmektedir.

Belirtilen çalışmalar göz önünde bulundurularak bu çalışmada iki farklı veri seti üzerinde Lojistik Regresyon, Naive Bayes, Genelleştirilmiş Regresyon Sinir Ağı, K-En yakın komşu, Destek vektör makineleri ve Rassal Orman yöntemleri uygulanmıştır. Birinci veri seti literatürde 2013 yılına ait Banka kredi kartı verilerinin yer aldığı veri seti, ikinci veri seti ise çalışma kapsamında manuel olarak oluşturulan 700 kayıt içeren yapay veri setidir. Belirtilen yöntemler veri kümeleri üzerinde python dilinde uygulanarak performansları değerlendirilmiştir.

3. MATERYAL & METOD

3.1. Kullanılan Yöntemler

Tez çalışmasında sahtecilik tespiti için literatürde bulunan veri madenciliği ve makine öğrenmesi algoritmaları kullanılmıştır. Veri madenciliği, mevcut ham veriden yararlı bilgiye ulaşılmasında çeşitli istatistiksel yöntemleri, makine öğrenme algoritmalarını ve yapay zeka gibi disiplinleri iç içe kullanan bir analiz bilimidir. Veri madenciliği ile büyük ölçekli veriler arasından nitelikli bilgiye ulaşılması amacıyla veri belirli aşamalardan geçirilerek anlamlı bilgi haline getirilmektedir (Şekil 3.1).



Şekil 3.1. Veri madenciliği süreçleri

Veri madenciliğinin hangi yöntemler ile uygulanacağı, problemin hedef çıktılarına bağlı olarak farklı amaçlara sahip olabileceğinden, istenen sonucun başarılı bir şekilde sağlanabilmesi için probleme göre düşünülerek karar verilmelidir. Bu yöntemlerden sınıflandırma kavramı, bir veri kümesi üzerinde tanımlanmış sınıfların küme içerisindeki verilere dağıtılmasıdır. Sınıflandırma algoritmaları, veri kümesinde bulunan sınıfların dağılımına göre bu mevcut örnek veriler ile eğitilirler ve bu sayede daha önce hiç karşılaşmadıkları yeni veriler üzerinde sınıflandırma yeteneği kazanırlar. Bu bölümde tez

çalışmamızda tahminleme amacıyla uygulanan makine öğrenmesi sınıflandırma algoritmaları tanıtılmaktadır.

3.1.1. Naive Bayes Algoritması

Makine öğrenmesi sınıflandırma modellerinden Naive Bayes, olasılık ilkelerine göre tanımlanmış formüller ile sunulan verilerin sınıfını tespit etmeye çalışır. Burada amaç, verilen bilgilerden bir eğitim kümesi oluşturularak bu eğitim sonucunda modelin daha önce karşılaşmadığı verilerin doğru bir şekilde sınıflandırılmasıdır.

Bir sınıflandırma problemi, bir çok özellikten ve bir hedef (sonuç) değişkeninden oluşmaktadır. F veri kümesi (F_1, F_2, …, F_n) verildiğinde, bunun sonucunda oluşabilecek olayın olasılığının tespit edilmesi Denklem 3.1'de belirtildiği gibi hesaplanmaktadır.

$$P(C \mid F_1, \ldots, F_n) = P(C)p(F_1, \ldots, F_n \mid C) \,/\, p(F_1, \ldots, F_n) \qquad (3.1)$$

Denklemde C, verilen hedefi, F ise özellikleri temsil etmektedir. Buradaki temel düşünce, F setine P(C | F) olasılığı maksimum olan sınıfı atamaktır. Naive Bayes sınıflandırıcısı bütün koşullu olasılıkların çarpımıdır. Aslında Naive Bayes koşul ile sorulan sorunun yerini değiştirme temeline dayanır. Örnek olarak, belirli bir ürünü "20 yaşın altında kaç kişi satın almıştır" sorusu yerine; "satın alanların kaçı 20 yaşın altındadır" sorusunu sorarak değiştirme yapar ve bu şekilde hepsini hesaplayarak devam eder.

Adımlar:
Adım 1. Veri seti frekans tablosuna dönüştürülür.
Adım 2. Olasılık değerleri hesaplanarak, bir olabilirlik (likelihood) tablosu elde edilir.
Adım 3. Naive Bayes denklemini kullanarak her bir sınıf için sonsal (posterior) olasılık hesaplanır. En yüksek sonsal olasılığa sahip sınıf, tahmin çıktısı olarak kabul edilir.

Naive Bayes algoritmasının sınıflandırma için kullanım şekline bir örnek olarak mevcut yaş, gelir, öğrencilik durumu, kredi skoru gibi bilgilerin tamamının eğitim verisi olarak bulunduğunu düşünürsek, bu bilgilere bakılarak hangi gruptaki insanların bilgisayar satın alıp almayacağını tahmin edilmesi istenen bir problemin olduğunu düşünelim.

Çizelge 3.1. Naive Bayes örnek eğitim kümesi

nitelikler				sinif
yas	Gelir	ogrenci_mi	kredi_skoru	bilgisayar_alimi
genç	Yüksek	hayır	orta	hayır
genç	Yüksek	hayır	mükemmel	hayır
orta yaşlı	Yüksek	hayır	orta	evet
kıdemli	Orta	hayır	orta	evet
kıdemli	Düşük	evet	orta	evet
kıdemli	Düşük	evet	mükemmel	hayır
orta yaşlı	Düşük	evet	mükemmel	evet
genç	Orta	hayır	orta	hayır
genç	Düşük	evet	orta	evet
kıdemli	Orta	evet	orta	evet
genç	Orta	evet	mükemmel	evet
orta yaşlı	Orta	hayır	mükemmel	evet
orta yaşlı	Yüksek	evet	orta	evet
kıdemli	Orta	hayır	mükemmel	hayır

Sınıf değerlerini bulmak için tüm olasılık değerlerinin hesaplanması:

* Herhangi bir kişinin satın alıp almama ihtimali;

P(C): P(bilgisayar_alimi = ”evet”) = 9/14 = 0,643

P(C): P(bilgisayar_alimi = ”hayır”) = 5/14 = 0,357

* $P(X \mid C_i)$ her sınıf için olasılıkların hesaplanması:

Yaşa göre:

P(yas=”genç” | bilgisayar_alimi = ”evet”) = 2/9 = 0,22

P(yas=”genç” | bilgisayar_alimi = ”hayır”) = 3/5 = 0,6

Gelir düzeyine göre;

P(gelir = “orta” | bilgisayar_alimi = ”evet”) = 4/9 = 0,444

P(gelir = “orta” | bilgisayar_alimi = ”hayır”) = 2/5 = 0,4

Öğrenci olup olmama durumuna göre:
P(ogrenci = “evet” | bilgisayar_alimi=”evet”) = 6/9 = 0,617
P(ogrenci = “evet” | bilgisayar_alimi=”hayır”) = 1/5 = 0,2

Kredi skoruna göre:
P(kredi_skoru = “orta” | bilgisayar_alimi=”evet”) = 6/9 = 0,667
P(kredi_skoru = “evet” | bilgisayar_alimi=”hayır”) = 2/5 = 0,4

tüm olasılıklar hesaplanır. Bu durumda iki sınıf (“evet”/”hayır”) olasılık değerleri için hesaplandığında, verilen koşullarda (X) her bir koşul için alınma durumunun istatistiğini verecektir;

X = (yas = “genç”, gelir = “orta”, ogrenci_mi = “evet”, kredi_skoru = “orta”)
koşulları verildiğinde, bu özelliklerdeki bir kişi için;
$P(C_i)$ = P(X| bilgisayar_alimi = ”evet”) = 0,222 * 0,444 * 0,667 * 0,667 = 0,044
$P(C_i)$ = P(X| bilgisayar_alimi = ”hayır”) = 0,6 * 0,4 * 0,2 * 0,4 = 0,019

Son olarak X koşulunun gerçekleşmesi durumunda $P(C_i)$ ‘nin gerçekleşmesi;
$P(X|C_i) * P(C_i)$:
P(X| bilgisayar_alimi = ”evet”) * P(bilgisayar_alimi = ”evet”) = 0,028
P(X| bilgisayar_alimi = ”hayır”) * P(bilgisayar_alimi = ”hayır”) = 0,007

Sonuçlar 0,028 ihtimalini daha yüksek göstermektedir. Bu durumda alma olasılığı “evet” sınıfı daha yüksektir.

Naive Bayes algoritması uygulanması kolay bir yöntemdir. Fakat bu algoritmada öznitelikler birbirinden bağımsız olarak kabul edilmektedir; oysaki gerçek yaşamda nitelikler birbiriyle bağımsız olmayabilir; bazı durumlarda koşullu bağlılık durumları olabilmektedir. Bu sebeple bu modelde nitelikler (değişkenler) arasındaki ilişki modellenememektedir. Bu da Naive Bayes yaklaşımının zayıf yönlerinden biri olarak düşünülebilir.

3.1.2. K-En Yakın Komşu Algoritması

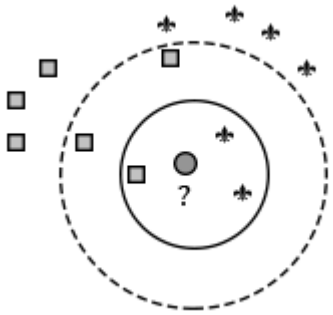
Makine öğrenme algoritmalarından gözetimli öğrenme kategorisinde bulunan K En Yakın Komşu algoritması, sınıfları belli olan bir örnek kümesindeki verilerden yararlanılarak, örnek veri setine katılacak olan yeni verinin, mevcut verilere olan uzaklığının hesaplanıp; k sayıda en yakın mesafelerine bakılarak yapılan bir hesaplamadır. Uzaklık hesapları için genellikle 3 tip uzaklık fonksiyonu kullanılmaktadır;

1) Öklid Uzaklığı (Euclidean Distance)
2) Manhattan Uzaklığı (Manhattan Distance)
3) Minkowski Uzaklığı (Minkowski Distance)

KNN algoritması, sınıflandırma sırasında çıkarılan özelliklerden, sınıflandırılmak istenen yeni verinin daha önce sınıflandırılmış verilerden k tanesine olan yakınlıklarına bakılarak, yeni k adet bireylerin oluşturulması ve en yakın sınıfın bulunmasını amaçlamaktadır. Böylece verinin ait olduğu sınıf tahmin edilmektedir.

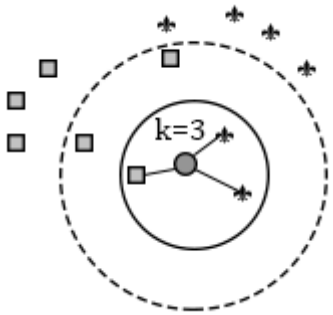
Algoritmadaki uzaklık, her bir test verisi ile öğrenilen verinin farkı ve bu çıkan sonuçların formül ile uzaklık hesabının yapılıp burada bulunan en küçük değerlerin kontrolü ile yönetilir. Algoritmada bulunan k, k adet komşu demektir ve test verisinin koordinat düzleminde bulunduğu noktaya en yakın olan komşular anlamını taşımaktadır.

Bir sınıflandırma problemi için k=3 verildiğini düşünürsek bu, eski sınıflandırılmış üyelerden 3 tanesinin alınacağı anlamına gelmektedir. Bu üyeler hangi sınıfa dahil ise yeni üye de o sınıfa dahil edilmektedir. k yakınlığını ölçmek için bir çok farklı uzaklık formülleri (Öklid, Manhattan, Gauss, gibi) içerisinden genellikle öklid yöntemiyle uzaklık mesafesi hesaplanmaktadır.



Şekil 3.2. KNN model grafiği (1)

Buradaki temel düşünce nesnelerin birbiri arasında yakınlık ilişkilerine göre kümeleme işlemi yapılmasıdır.



Şekil 3.3. KNN model grafiği (2)

Şekil 3.3'teki gibi yeni bir üye eklendiğinde yeni üyenin öklid mesafesi ile k=3 adet en yakın komşu uzaklıklarına bakılarak yeni üyenin sınıflandırılması yapılmaktadır.

KNN adımları:

Adım 1. k değeri belirlenir.

Adım 2. Örnek veri setine katılacak olan yeni verinin, mevcut verilere göre uzaklığı seçilen uzaklık fonksiyonu ile tek tek hesaplanır.

Adım 3. Uzaklıklar sıralanır ve minimum uzaklıklara bakılarak en yakın komşular bulunur. İlgili uzaklıklardan en yakın k komşu ele alınır. Öznitelik değerlerine göre k komşu veya komşuların sınıfına atanır.

Adım 4. En yakın komşu kategorileri toplanır. Seçilen sınıf, tahmin edilmesi

beklenen gözlem değerinin sınıfı olarak kabul edilir. Yani yeni veri etiketlenmiş olur.
Adım 5. En uygun komşu kategorisi seçilir.

Örnekleme:
1. k = 3
2. Yeni gelen veri ile diğer nesneler arasındaki uzaklıkları hesapla.
3. Uzaklıkları sırala, en yakın k adet nesneyi seç:
En yakın 3 uzaklık sırasıyla: Nesne 4, Nesne 3 ve Nesne 1.
4. Sınıf kategori sayılarını belirle: 2 adet "Sınıf_2 ", 1 adet "Sınıf_1".
5. Tahmin sonucu: "Sınıf_2".

KNN algoritmasının etkili olabilmesi için eğitim kümesinin büyük olması ve k değerinin uygun seçilmesi önemlidir. Algoritmayı probleme uygularken eğitim verisi artırılmalıdır. k değeri ise genellikle 1-3 arasında seçilir. Fakat bazı durumlarda bunun dışına çıkılması gerekebilir. k değeri artırılarak sırasıyla her bir k değeri için elde ettiğimiz tüm doğruluk oranlarına bakılmalıdır; en yüksek doğruluk değerine ulaştıran k değeri optimum olarak kabul edilebilmektedir ve uygulamada bu değer kullanılmalıdır.

Bu yöntemde dezavantaj olarak her ekleme işleminde uzaklıkların hesaplanması ve her durumun depolanması büyük veriler için düşünüldüğünde çok sayıda bellek alanına gereksinim oluşturmasıdır. Bu da maliyetleri artırmaktadır. Örnek sayısı arttığında yapmamız gereken karşılaştırma işlemlerinin sayısı da doğrusal olarak artmaktadır ve bu da ağır bir işlem yükü getirmektedir. Ayrıca bu yöntem, gürültülü veriler üzerinde iyi performans göstermediği bilinmektedir; ancak veriye uygun filtreleme yöntemleri ile modelin performansını artırmak mümkündür. Özellikle boyut sayısı ve k sayısı arttığında kNN algoritması "overfitting" problemi ile karşı karşıya kalmaktadır.
X, Y gözlem değerleri ve Z sınıf değerlerinin oluşturduğu verileri düşünürsek, bu gözlem değerlerinden yola çıkarak yeni gelen değerlerin hangi sınıfa ait olduğunun bulunması için;

Çizelge 3.2. KNN örnek veriler

X	Y	Z	Uzaklık
1	3	-	6
2	5	+	5.39
2	3	+	5
3	9	-	7.21
4	7	-	5
5	2	+	2.24
6	8	+	5.10
8	6	-	3.16
10	6	-	4.24
11	1	-	2.83

Yeni gözlem değeri X=7, Y=3.

Adım 1. (7, 3) noktası için k=4 seçilir,

Adım 2. Öklid mesafesi denklemine ile, her bir gözlem değeri için mesafeler hesaplanır,

Adım 3. En kısa 4 mesafenin belirlenmesi için sıralama yapılır,

Çizelge 3.3. KNN örnek sonuçları

X	Y	Uzaklık	Sıralama
5	2	2.24	1
8	6	3.16	3
10	6	4.24	4
9	1	2.83	2

Adım 4. Belirlenen değerler içinde en baskın değere karar verilir.

Sonuç, 1 pozitif ve 3 negatif değer içerisinden (7, 3) noktasının sınıfı negatif olarak belirlenir.

3.1.3. Rassal Orman Yöntemi

Rassal Orman (Random Forest) temel olarak 'Bagging' ve 'Random Subspace' yöntemlerine dayanan bu yöntem, Leo Brieman tarafından 2001 yılında geliştirilmiştir. Bu yöntemde mevcut veri setindeki değerler rastgele kullanılarak N adet rastgele karar ağaçlarının meydana getirdiği bir koleksiyon oluşturulmaktadır.

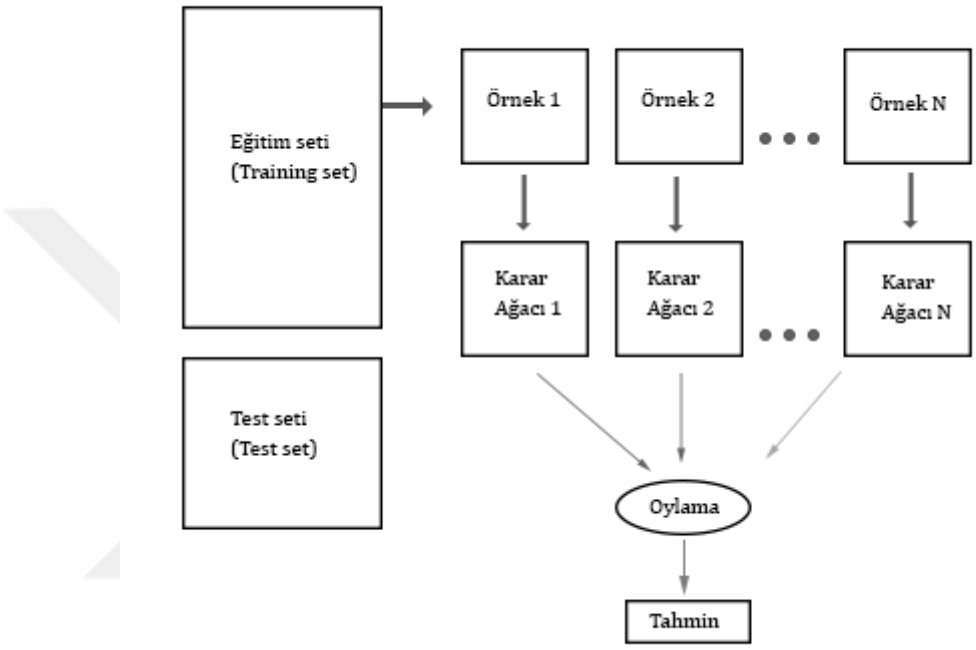
RF adımları:

1. Veri setinden örnekler rastgele seçilir,

2. Her bir örnek için karar ağacı oluşturulur ve her karar ağacından bir tahmin üretilir,

3. Her tahmin sonucu için bir oylama yapılır,

4. En fazla oya sahip olan tahmin sonucu son tahmin olarak belirlenir.



Şekil 3.4. Rassal Orman yöntemi örnek gösterimi

Bu yöntemin karar ağaçları ile temel farkı, işleme dahil edilen ağaç sayısının fazla olmasından dolayı doğruluk oranı yüksek, güçlü bir yöntem olmasıdır. Karar ağaçlarındaki aşırı uyum (overfitting) problemi bu yöntemde de karşımıza çıkmaktadır. Bunu çözmek için genellikle 'k-fold cross validation' gibi çapraz doğrulama metodları ile parametre ayarı yapılması tercih edilmektedir. N parametresi artırıldıkça doğruluk oranı artmaktadır. Mümkünse veri sayısı artırılarak da overfitting oluşma ihtimali azaltılır.

Bu yöntemin dezavantajı olarak, koleksiyonda çok sayıda ağaç olduğu için her tahmin aşamasında verilen bir girdi için tüm ağaçların tahmin üretmesi beklenmesidir, bu sebeple karar ağaçlarına nispeten daha yavaş çalışan bir

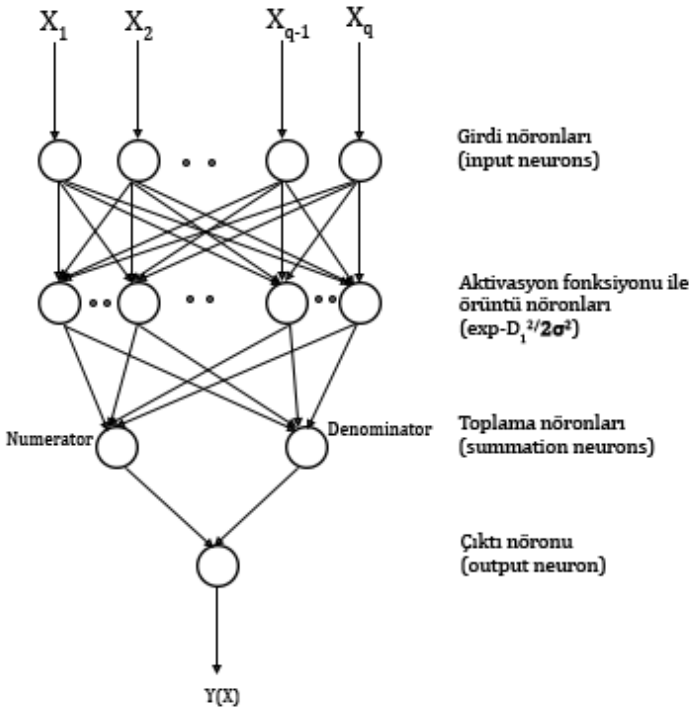
yöntemdir. Karar ağaçlarında ise ağaç üzerinde giden yol kolaylıkla takip edilerek karar alınır ve hesapsal olarak daha hızlıdır. Sonuç olarak RF yönteminin yorumlanması nispeten daha zor olmaktadır.

Avantaj olarak bu yöntem, hem sınıflandırma hem de regresyon problemlerinde kullanılabilmektedir. Destek Vektör Makinelerine (SVM) göre daha keskin tahmin sonuçları üretmektedir. Bunun nedeni, mümkün olduğunca birbirinden farklı ağaçlar oluşturularak düşük kolerasyon yapısında bir ağaç topluluğu elde edilmesidir.

3.1.4. Genelleştirilmiş Regresyon Sinir Ağı Yöntemi

Yapay sinir ağlarının temel prensibinde olduğu gibi Genelleştirilmiş Regresyon Sinir Ağı (GRNN) yönteminde de ağın eğitimi için eğitim verisine gereksinim vardır. Verilen bir eğitim veri seti ile ağ eğitilir ve ağ daha önce karşılaşmadığı test veri seti ile beslenir, bunun sonucunda da bir tahmin çıktısı üretecektir.

GRNN yönteminde ise çıktı, eğitim verisi çıktılarının ağırlıklı ortalaması kullanılarak hesaplanmaktadır. Burada ağırlık; eğitim verisi ile test verisi arasındaki mesafe öklid uzaklığı kullanılarak hesaplanmaktadır.



Şekil 3.5. Genelleştirilmiş Regresyon Sinir Ağı (GRNN)

GRNN ağının mimarisinde temel olarak 4 katman bulunmaktadır. Bunlar; girdi katmanı, örüntü katmanı, toplama katmanı ve çıktı katmanıdır. Girdi katmanı, girdiyi bir sonraki katmana verir. Örüntü katmanında, öklid uzaklığı ve aktivasyon fonksiyonu hesaplanır. Toplama katmanı 'numerator' ve 'denominator' olarak iki alt kısıma ayrılmaktadır; 'numerator', eğitim çıktılarının çarpımı ve aktivasyon fonksiyonunun toplamıdır; 'denominator' ise tüm aktivasyon fonksiyonlarının toplamıdır. Denominator ve numerator bir sonraki katmanı besler. Çıktı katmanı, bir önceki toplama katmanındaki nominatorü denominatöre bölerek bir çıktı nöronu oluşturmaktadır. Bu tek nöron da çıkış katmanındadır.

Denklem 3.2'de GRNN yönteminin matematiksel ifadesi verilmektedir.

$$Y(x) = \sum Y_i e^{-(d_i^2/2\sigma 2)} / \sum e^{-(d_i^2/2\sigma 2)} \quad (3.2)$$

Burada $d_i^2 = (x-x_i)^T (x-x_i)$ olmak üzere, X girdi örneğini, X_i ise eğitim örneğini ifade etmektedir. Y_i ise, girdi örneği i' nin çıktısını ifade etmektedir. d_i^2, X ve X_i

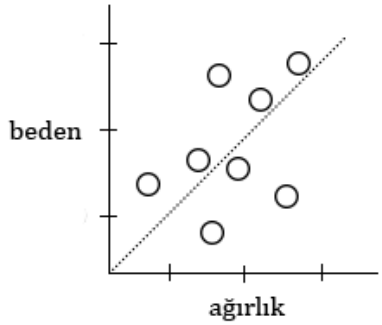
arasındaki öklid uzaklığıdır. $e^{-(d_i^2/2\sigma 2)}$ aktivasyon fonksiyonudur. Bu aktivasyon fonksiyonu, girdi için ağırlıktır (weight). Eğitim örneğinin ne kadar ağırlık katacağını belirlemektedir. d_i^2 değeri, eğitim örneğinin belirli bir test örneğinin çıktısına ne kadar katkıda bulunacağını belirtmektedir. Eğer d_i^2 küçük bir değerse çıktıya daha fazla değer katacağı anlamına gelmektedir, eğer büyük bir değerse çıktıya az değer katacağı anlamına gelmektedir. d_i^2 küçük bir değer ise $e^{-(d_i^2/2\sigma 2)}$ aktivasyon ifadesi nispeten büyük bir değer döndürür. Büyük bir değerse nispeten küçük bir değer döndürür. Eğer d_i^2 sıfırsa $e^{-(d_i^2/2\sigma 2)}$ ifadesi 1 döndürür. Bu da test verisinin eğitim verisine eşit olduğu anlamına gelmektedir ve test verisinin çıktısı eğitim örneğinin çıktısı olacaktır.

Denklemde bilinmeyen parametre olarak yalnızca sigma (σ) bulunmaktadır. Bu parametre, ağın eğitim aşamasında hatanın en az olduğu parametre gözlemlenerek optimum değere ayarlanabilir. Bunu bulmak için, MSE (Mean Squared Error) değerinin minimum olduğu konum bulunarak optimum değere ulaşılabilmektedir. Öncelikle tüm veri seti eğitim ve test olmak üzere iki kısıma ayırılarak ağ eğitim verisi ile eğitildikten sonra, test verisine GRNN uygulanarak farklı sigma değerleri için MSE hesaplanır. Minimum MSE değerine tekabül eden sigma değeri seçilir.

GRNN yönteminin sağladığı temel avantaj, eğitim aşamasını hızlandırması ve dolayısıyla ağın daha hızlı öğrenmesini sağlamaktadır.

3.1.5. Lojistik Regresyon Yöntemi

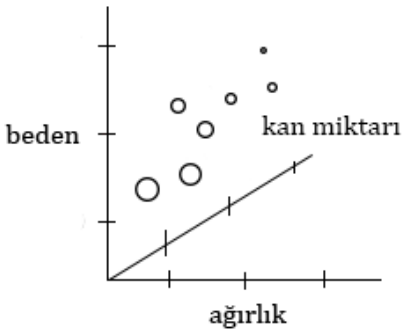
Lojistik regresyon (LR), kategorik tipte sınıflandırma problemleri için uygun bir sınıflandırıcıdır. Lineer regresyon ile benzerlikleri bulunsa da farklı sonuçlar üretirler. Lineer regresyona örnek olarak, verilen bir ağırlık değerine karşılık beden tahmini yapılacağını düşündüğümüzde, lineer regresyon gösterimi Şekil 3.6'daki gibidir.



Şekil 3.6. Lineer Regresyon örnek gösterim (1)

Lineer regresyon normal lineer regresyon ve çoklu lineer regresyon olmak üzere iki türde ele alınmaktadır. Normal lineer regresyonda;

1. R^2 hesaplanır ve şekildeki ağırlık ve beden özellikleri arasında bir korelasyon olup olmadığı belirlenir. Büyük değerler büyük etki gösterir.
2. Bir p değeri hesaplanarak R^2 değerinin istatistiksel önemi belirlenir.
3. Çizgiyi kullanarak verilen bir ağırlık değeri için beğen tahmin edilebilmektedir.



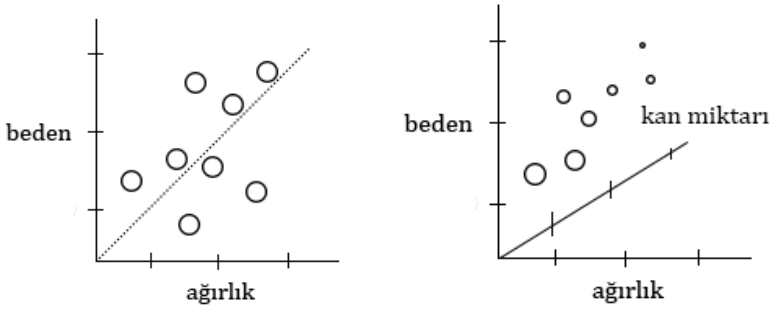
Şekil 3.7. Lineer Regresyon örnek gösterim (2)

Şekil 3.7'de ise normal regresyondan farklı olarak ağırlık ve kan miktarı özellikleri kullanılarak beden tahmini yapıldığını düşünürsek, yine normal lineer regresyondaki gibi çoklu lineer regresyonda da;

1. R^2 hesaplanır

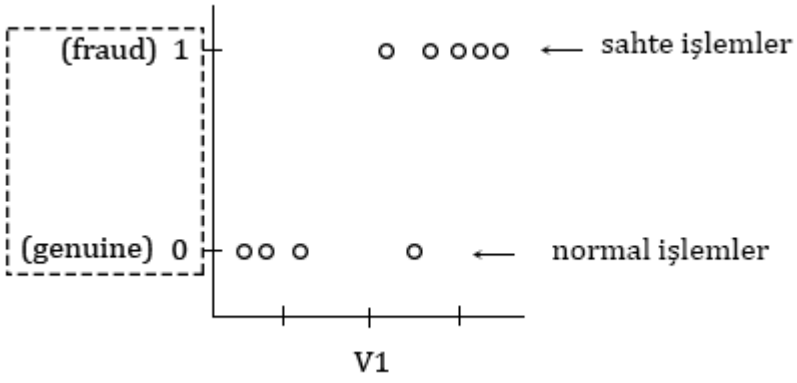
2. P değeri hesaplanır

3. Verilen ağırlık ve kan miktarı değişkenleri ile beden tahmini yapılır. Buradan genetik özellik de tahmin edilebilir. Ayrıca lineer regresyonda modeller karşılaştırılabilmektedir (Şekil 3.8).



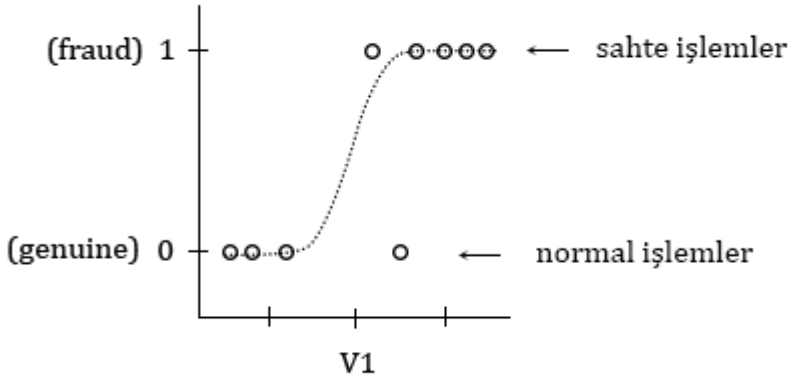
Şekil 3.8. Lineer Regresyon örnek gösterim (3)

Bu iki modelin (normal ve çoklu lineer regresyon) karşılaştırılması, beden tahmininde sadece ağırlık özelliği kullanılarak mı ya da ağırlık ve kan miktari özellikleri birlikte kullanılarak mı daha doğru bir tahmin elde edilebileceğini anlatır. Lojistik Regresyonda ise, lineer regresyonda olduğu gibi, -∞ ile +∞ arasında sürekli (continuous) bir tahmin değeri vermek yerine; doğru/yanlış, 1/0 gibi ikili bir tahmin sonucu üretilmektedir (Şekil 3.9).



Şekil 3.9. Lojistik Regresyon örnek gösterim (1)

Bir diğer farkı, verilerin doğrusal bir çizgi yerine S şeklinde bir lojistik fonksiyonuna (sigmoid fonksiyonu) sığdırılmasıdır. Şekil 3.10'da görüldüğü gibi eğim 0'dan 1'e doğru gitmektedir. Dolayısıyla [0, 1] arasında bir olasılık değeri üretilmektedir. Buradaki eğim, işlemin V1 özelliğinin baz alınarak sahte olma olasılığını vermektedir.

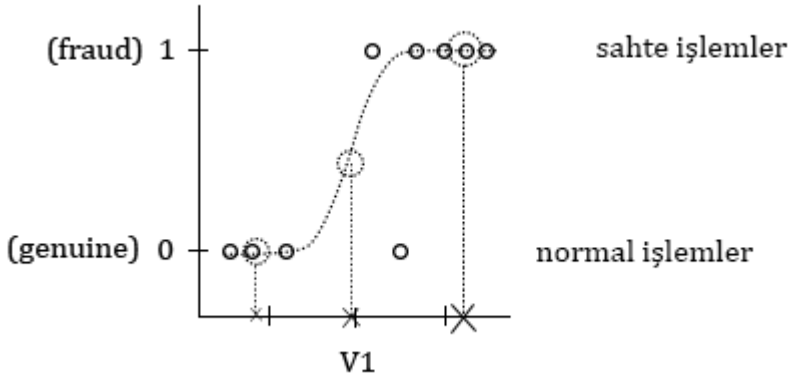


Şekil 3.10. Lojistik Regresyon örnek gösterim (2)

Sigmoid aktivasyon fonksiyonu Denklem 3.3'te belirtilmektedir.

$$f(x) = \frac{1}{1+e^{-x}} \tag{3.3}$$

Şekil 3.11'de görüleceği gibi V1 yüksek bir değer ise işlem büyük olasılıkla sahtedir, ortalarda bir yerde ise %50 civarındadır, daha küçük bir değerse sahte olma olasılığı çok düşüktür. Sonuç değeri > %50 ise sahte, sonuç değeri < %50 ise normal olarak sınıflandırılmaktadır.

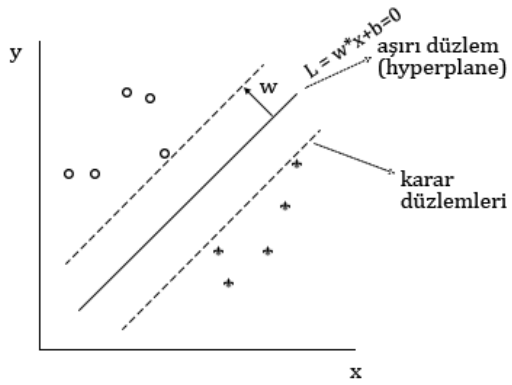


Şekil 3.11. Lojistik Regresyon örnek gösterim (3)

Lojistik regresyonda lineer regresyondaki gibi modeller kıyaslanamaz. Bunun yerine bir değişkenin tahmin üzerindeki etkisinin 0'dan önemli ölçüde farklı olup olmadığına bakılmaktadır. Eğer değilse değişkenin tahmine bir etkisinin olmadığı kabul edilmektedir (Bunun için "Wald Test" kullanılır).

3.1.6. Destek Vektör Makineleri Yöntemi

Destek vektör makineleri (Support Vector Machines), Lojistik Regresyon ile benzer bir sınıflandırıcıdır. Her iki yöntemde de iki sınıfı ayıran çizgi hesaplanmaktadır. Düzlem üzerinde iki veri grubunu ayırabilecek sonsuz sayıda çizgi çizilebilir ancak SVM, iki sınıfa ait verileri en iyi çizgiyi belirleyerek sınıflandırma yapmaya çalışır; bu çizginin yeri iki grubun da verilerine en uzak yer olmalıdır. Bunun için karar sınırları ve bir aşırı düzlem (hyperplane) belirlenmektedir.



Şekil 3.12. SVM örnek gösterim

Şekil 3.12'deki gibi 2 boyutlu bir düzlem üzerinde iki sınıfa ait verilerin olduğunu düşünürsek "confidence" formülü Denklem 3.4'te verilmektedir.

$$\text{"confidence"} = (w*x + b)\, y \tag{3.4}$$

Verilen bir i veri noktası için mesafe (margin/γ) formülü Denklem 3.5'te belirtildiği gibi hesaplanmaktadır.

$$\gamma_i = (w*x_i + b)\, Y_i \tag{3.5}$$

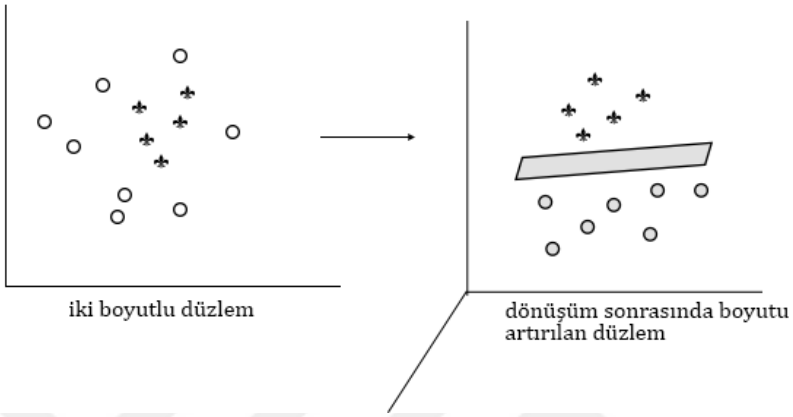
Burada γ_i mesafeyi (margin), Y_i tahminin sınıfını, $w*x_i + b$ ise "confidence" temsil etmektedir. Amaç maximum ayıran çizgiyi (maximum w) mümkün olan en geniş mesafe (margin) ile bulmak olduğundan maximum γ Denklem 3.6'da belirtildiği gibidir.

$$(w*x_i + b) >= \gamma \tag{3.6}$$

Burada her bir eğitim örneğinin mesafesi (margin) en azından γ'ye eşit ya da daha büyük olmalıdır. Maximum γ değerini bularak dolayısıyla w değeri bulunmuş olmaktadır.

SVM, verileri doğrusal olarak sınıflandırmaya çalışır ancak bazı durumlarda bu doğrusal ayrımın yapılması mümkün değildir. Buna çözüm olarak çekirdek

hilesi (Kernel Trick) yöntemi kullanılmaktadır. Bu yöntemle yeni bir boyut oluşturularak doğrusal olarak sınıflandırma yapılabilmektedir. Şekil 3.13'teki düzlemde yuvarlak noktalar hafif yukarı kaldırılıp z eksenine taşınır, böylece üçüncü bir boyut oluşur. Bu yöntem uygulanarak SVM ile doğrusal bir çizgi oluşturabilmektedir.



Şekil 3.13. SVM kernel trick

Kernel parametresini RBF, lineer ya da sigmoid olarak değiştirmek başarı oranını değiştirebilmektedir. Bu çalışmada SVM uygulamasında kernel parametresi olarak RBF kullanılmıştır.

3.2. Kullanılan Veri Setleri

E-Ticaret sitesine gelen bir siparişin sahte ya da gerçek olup olmadığının tespit edilebilmesi için veri madenciliği makine öğrenmesi sınıflandırma algoritmalarından yararlanılması mümkündür. Ancak sınıflandırma algoritmalarının eğitiminde kullanılacak veri seti de oldukça önemlidir.

Ülkemizde 6698 sayılı "Kişisel verilerin korunması kanunu" yasa metni kapsamında (Mevzuat Bilgi Sistemi, 2019), veri sorumluları haricinde bu verilere erişilmesi ve işlenmesi kanuna uygun olmadığından; ayrıca literatürde yapılan çalışmalar da göz önünde bulundurulduğunda, sahtecilik tespitine yönelik çalışmalarda veri erişiminin gizlilik sebepleriyle mümkün olmadığı, dolayısıyla bu alanda yapılan çalışmalarda veri kaynağı bakımından genel bir

sınırlamayla karşı karşıya kalındığı tespit edilmiştir. Bununla birlikte literatürde sahtecilik tespitine yönelik çalışmalarda kullanılabilecek veri setleri de mevcuttur.

Çalışmada iki farklı veri seti üzerinde sınıflandırıcıların performansları değerlendirilmiştir. Kullanılan birinci veri seti (Kaggle, 2019; Pozzolo, 2014; Pozzolo 2015; Pozzolo, 2018; Carcillo vd, 2018a; Carcillo vd, 2018b; Carcillo vd, 2019; Bertrand vd, 2019), Eylül 2013 yılına ait Avrupalı kredi kartı sahiplerinin iki gün içerisinde oluşan harcama işlemlerinin yer aldığı gerçek verilerdir. Bu veriler içerisinde 492 sahte işlem olarak, 284.807 işlem ise gerçek işlem olarak etiketlenmiştir. Veri setinde tüm işlemlerde %0,172 oranında pozitif sınıf (sahte) olduğundan, oldukça dengesiz bir veri setidir (imbalanced dataset). Verilerin banka tarafından temel bileşen analizi (PCA-Principal Component Analysis) yapıldığından dolayı veri setinde yalnızca sayısal değişkenler bulunmaktadır. Verilerin gizliliği sebebiyle 'V1, V2, …, V28' aralığındaki sütunlarda yer alan özelliklerin isimleri bilinmemekle birlikte, veriler üzerinde yapılan PCA dönüşümünün sonucunda verilerde bir eksiklik bulunmamaktadır. Bunların dışında veri kümesinde 'Time', 'Amount' ve 'Class' sütunlarındaki özellikleri belirtilmektedir. 'Time' özelliği her işlem ile veri kümesindeki ilk işlem arasında geçen saniyeleri içermektedir. 'Amount', işlem tutarını, 'Class' özelliği ise etiket değeridir ve 1 sahte işlemi (fraud transaction); 0 gerçek işlemi (normal transaction) ifade etmektedir.

Kullanılan ikinci veri seti, 700 kayıt içeren, 420 sahte işlem verisi ve 280 yasal veri olarak etiketlenmiş yapay olarak oluşturulan verilerdir. Veri setindeki öznitelik değerleri, e-posta adresi uzunluğu, teslimat-fatura adresinin aynı olup olmaması, IP adresleri, tutar bilgisi, üyelik bilgisi gibi bilgilerin tutulduğu 7 öznitelik içermektedir.

3.3. Yaklaşımlar

3.3.1. Temel Bileşenler Analizi

Temel bileşenler analizi (PCA), bir boyut değiştirme işlemidir. Çok boyutlu veri kümelerinin daha az boyutlara indirgenerek temel vektör bileşenlerinin elde edilmesi için kullanılan doğrusal bir yöntemdir. PCA, sınıflandırıcıların başarı oranlarının artırılması için gereklidir.
Bir veri setindeki bilgilerin orijinal boyutlarına eşit ya da daha düşük boyutlara dönüştürülmesi için, boyut üzerinde PCA için uygun olan dereceye döndürülmektedir. Bu işlem yapılırken veri kaybının yaşanmaması gerekmektedir. Yeni bulunan boyutlar ise daha iyi analiz yapılmasını sağlayan, verilerin daha iyi işlenmesini sağlayan veriler olmaktadır.

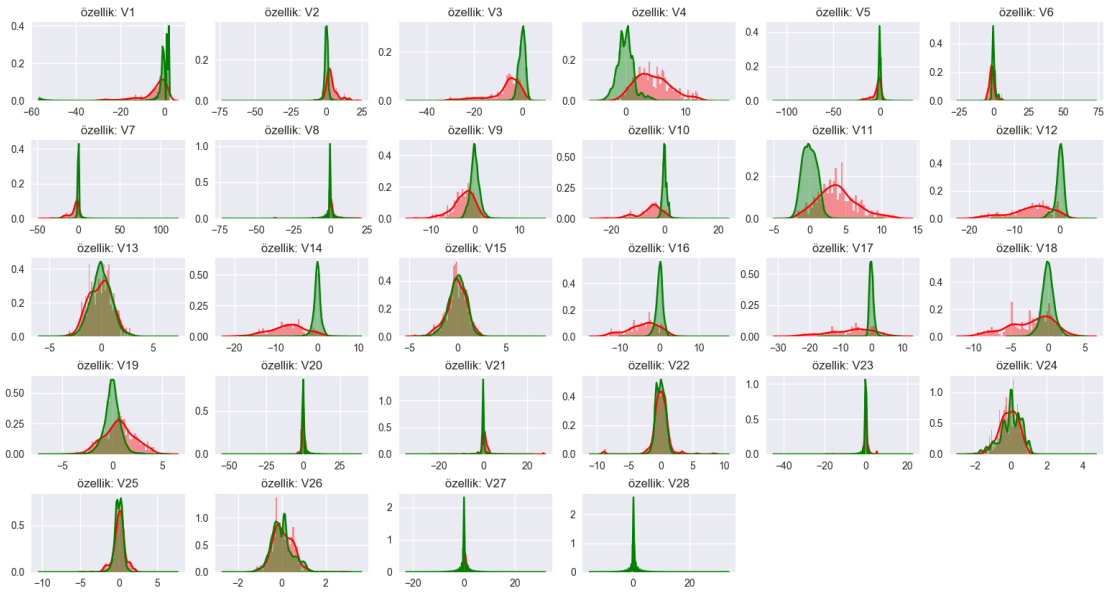
Bu tez çalışmasında kullanılan veri setinin PCA dönüşümü banka tarafından önceden yapılmış ve verilerde herhangi bir eksiklik bulunmamaktadır.

3.3.2. Tüm Veri Seti Yaklaşımı

Belirlenen algoritmalar üzerinde ilk deney olarak veri seti özelliklerinin yöntemler üzerindeki etkisinin belirlenmesi amacıyla veri setindeki tüm özellikler, herhangi bir özellik çıkarımı yapılmadan uygulanmıştır. Birinci veri setindeki veriler için {V1, ..., V28, Time, Amount} özellikleri olmak üzere toplam 30 özellik bulunmaktadır. Yalnızca 'Amount' (işlem tutarı) özelliği, veri setindeki diğer özelliklere göre yüksek değerler içerdiğinden özellik ölçekleme yapılarak bu alandaki veriler [-1,1] değerleri arasına getirilmiştir böylece özelliklerin standart bir normal dağılıma sahip olması sağlanmıştır.

3.3.3. Özellik Çıkarımı Yaklaşımı

Veri setinde sahte ve gerçek sınıf dağılımlarına bakılarak hangi özelliklerin sınıflandırmada verimli olacağı gözlemlenebilir. Şekil 3.14'te I. verilere ait sahte/gerçek grafik dağılımını içeren verilerin histogram gösterimi yer almaktadır. (Kırmızı: sahte, yeşil normal veri dağılımlarını temsil etmektedir.)



Şekil 3.14. Veri seti-I özniteliklerin histogram gösterimi

Şekil 3.14'te özelliklerin sınıf dağılımlarına bakıldığında veri seti içerisindeki özelliklerin belirli bir kısmı için ('V8', 'V13', 'V15', 'V20', 'V22', 'V23', 'V24', 'V35', 'V26', 'V28', 'V27') iki sınıfın da benzer dağılımlara sahip olduğu, görülmektedir. Bu sebeple bu yaklaşımda ilk veri kümesi içerisinden sınıflandırmaya katkısının olmayacağı düşünülen belirtilen özelliklerin çıkarılarak çalışmada kullanılan sınıflandırma yöntemleri üzerinde uygulaması yapılarak öznitelik çıkarımının yöntemlerin başarı oranları üzerindeki etkisi test edilmiştir.

Sahtecilik tespit sistemlerinde kullanılan veri setlerinde sınıf dağılımlarına bakıldığında sahte olmayan işlemler sahte işlemlere oranla çok fazla sayıda olduğunda, bu tür verilere dengesiz veri setleri denmektedir (unbalanced datasets). Bu durum, genellikle sınıflandırmada uygulanan modellerin performanslarına olumsuz etki edebilmektedir. Örneğin, bir veri setinin 10,000 kayıttan oluştuğunu ve bunlardan 10 tanesinin sahte, diğer verilerin normal sınıfa ait olduğunu düşünürsek, bu durumda sınıflandırıcı sahte işlemleri normal işlem olarak sınıflandırmaya meyilli olacaktır. Bunu çözmek için veri setini dengelemeye yönelik çeşitli yaklaşımlar uygulanmaktadır. Bunlar "over-sampling", "under-sampling" yöntemleridir. Bu yöntemlerde eşit sınıf dağılımına getirilen veri seti üzerinde performansa önemli ölçüde iyileştirme

etkisi yaratmaktadır. Bunun yanında her iki yöntemin de kendi içinde dezavatajları bulunmaktadır.

3.3.4. Under-sampling Yaklaşımı

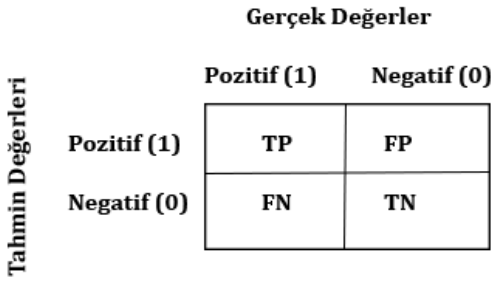
Veri madenciliği makine öğrenmesinde en büyük problemlerden biri olan dengesiz veri setlerinde karşılaşılan sınıf dağılımından kaynaklanan aradaki aşırı farkın iyileştirilmesi için kullanılan bir yöntemdir. Bu yöntem; sayıca çok olan çoğunluk sınıfına ait verilerin, sayıca az olan sınıf etiketindeki verilere yaklaştırılmasıdır. Bu yaklaşım, dengesiz veri seti dengelendiği için eşit sayıda sınıf dağılımı olacağından sınıflandırma başarısına önemli ölçüde katkı sağlayabilmektedir. Bu yöntemin dezavantajı ise, çoğunluk sınıf örneklerinden eksiltilerek örnek kümesi oluşturulduğu için, normal "0" sınıfı daha iyi temsil edebilecek veri örnekleri kaybolabileceğinden, sınıflandırmada yararlı olabilecek bilgiler gözden kaçırılmış olabilmektedir.

Alternatif olarak "over-sampling" yaklaşımı da kısaca; azınlık sınıf örneklerini sayıca diğer sınıf örneklerine yaklaştırmaktadır. Azınlık sınıfı kopyalanarak az sayıda örnek türünden çok sayıda görüleceği için sınıflandırma modelinde ezber (overfitting) problemi ortaya çıkacaktır. Bu da bu yöntemin dezavantajıdır. Bu tez çalışmasında veri seti-I üzerinde "under-sampling" yöntemi ile sınıflandırma algoritmalarının uygulama başarıları belirlenen ölçüm metrikleri ile test edilmiştir.

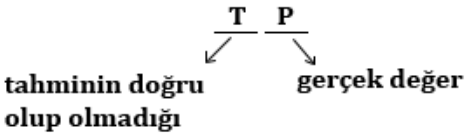
3.3.5. Karışıklık Matrisi (Confusion Matrix)

Karışıklık matrisi, tahmin edilen değerler ile gerçek değerlerden oluşan 4 farklı kombinasyonu bulunan bir tablodur (Towards Data Science, 2019) (Şekil 3.14). Bu matriste yer alan TP (True Positive), FP (False Positive), FN (False Negative), TN (True Negative) değerleri (Şekil 3.15) ile başarı ölçütlerinin hesaplaması yapılmaktadır.

Sahtecilik tespitinde kullanılan veri setlerinin yapısı gereği sahte ve gerçek işlem oranları arasındaki farkın çok fazla olmasından dolayı seçilen sınıflandırma modelinin başarısının yalnızca doğruluk (accuracy) ile ölçülmesi yeterli olmamaktadır. Doğruluk oranı yüksek çıksa da asıl başarı ölçütleri olarak Karışıklık Matrisine bakılır. Buradan hesaplanan True Positive Rate, False Positive Rate, Precision, Recall, F1-Measure, Accuracy metrikleri seçilen modelin performansını ölçmede gereklidir.



Şekil 3.15. Karışıklık matrisi (confusion matrix)



Şekil 3.16. Karışıklık matrisi tahmin sonuç değeri gösterimi

y_pred : tahmin sonuçları,

y_test : gerçek sonuçları,

sonuç : modelin tahmininin hata matrisindeki karşılığını temsil ettiğini düşünürsek örnek tahmin sonuçları Çizelge 3.4'teki gibidir.

Çizelge 3.4. Karışıklık matrisi örnek tahminleme

y_pred	y_test	sonuç
1	1	TP
1	0	FN
1	1	TP
0	1	FP
0	0	TN
1	0	FN
0	1	FP
1	0	FN
0	0	TN
0	0	TN
1	1	TP

Doğruluk (Accuracy)

Seçilen modelin ne kadar doğru sonuç verdiğinin ifade edilmesidir; doğru sınıflandırmanın (doğru tahminlerin) toplama oranıdır. Çoğu durumda doğruluk, %99,9 oranını verebilir ancak bu bazı problemlerde başarı ölçütü olarak yeterli görülmez. Bu sebeple karışıklık matrisinin de ölçülmesi daha sağlıklı sonuçlar vermektedir. Denklem 3.7'de doğruluk formülü belirtilmektedir.

$$\text{Doğruluk} = TN + TP \,/\, \text{Toplam} \tag{3.7}$$

Hata Oranı (Error Rate)

Yanlış sınıflandırma oranı, yanlışların toplama oranını ifade eder. Bu değer aynı zamanda 1'den doğruluk oranının çıkarılmasıyla da elde edilir. Denklem 3.8'de hata oranı formülü belirtilmektedir.

$$\text{Hata Oranı} = FN + FP \,/\, \text{Toplam} = 10 + 20 \,/\, 330 = 0{,}09$$

ya da

$$\text{Hata Oranı} = 1 - \text{Doğruluk} = 1 - 0{,}91 = 0{,}09 \tag{3.8}$$

Doğru Pozitif Oranı (True Positive Rate - Sensitivity)

Doğru olarak tahmin edilen pozitiflerin (TP) gerçek pozitiflere oranıdır. Modelin doğruları tahmin etme konusundaki ne kadar etkin olduğunun ölçülmesidir. Denklem 3.9'da doğru pozitif oranı formülü belirtilmektedir.

$$\text{Duyarlılık} = TP / (TP+FP) = 200/(200+10) = 0,95 \quad (3.9)$$

Yanlış Pozitif Oranı (False Positive Rate)

Gerçekte var olmayan ancak var olarak tahmin edilenlerin, gerçekte var olmayanlara oranıdır. FP/Gerçek Yok Toplamı. Denklem 3.10'da yanlış pozitif oranı formülü belirtilmektedir.

$$\text{Yanlış Pozitif Oranı} = FP / \text{Gerçek varlar} = 20/100+20 = 0,17 \quad (3.10)$$

Doğru Pozitif oranı ve Yanlış pozitif oranı Receiver Operating Charasteristic (ROC) grafiği çizimi ve Area Unver Curve (AUC) hesaplamada kullanılmaktadır.

Geri Çağırma (Recall)

Tüm pozitif sınıflardan, ne kadar doğru tahmin ettiğimizin ölçüsüdür. Bu değer olabildiğince yüksek olmalıdır. Denklem 3.11'de geri çağırma formülü belirtilmektedir.

$$\text{Recall} = TP/(TP+FN) \quad (3.11)$$

Hassasiyet (Precision)

Doğru olarak var tahmin edilenlerin, toplam var tahminlere oranıdır. Denklem 3.12'de hassasiyet formülü belirtilmektedir.

$$\text{Precision} = TP/(FP+TP) = 200/220 = 0,91 \quad (3.12)$$

F-Ölçütü (F-Measure)

F-ölçütü keskinlik ve geri çağırma ölçütlerinin ağırlıklandırılmış harmonik ortalamasıdır. Sınıflandırıcının ne kadar iyi performans gösterdiğinin bir ölçüsüdür ve sınıflandırıcıları karşılaştırmada kullanılır. Denklem 3.13'te f-ölçütü formülü belirtilmektedir.

F-Measure = 2*Recall*Precision/Recall + Precision (3.13)

Bu çalışmada sınıflandırma modellerinin değerlendirilmesinde Çizelge 3.5'te belirtilen metrikler kullanılmıştır.

ROC – AUC

ROC (Receiver Operating Characheristic) eğrisi, sonuç değişkeninin iki olasılıklı olduğu sınıflandırmalarda, ayrım eşik değerinin farklılık gösterdiği durumlarda, hassasiyetin kesinliğe olan oranıyla ortaya çıkmaktadır. Doğru pozitiflerin yanlış pozitiflere olan oranı olarak da ifade edilebilmektedir. Bir ROC eğrisi, farklı eşik değerleri için dikey eksen üzerinde doğru pozitifler (duyarlılık) ve yatay eksen üzerinde yanlış pozitiflerin (1-özgüllük) oranlarının yer aldığı bir eğridir. Bu eğri üzerindeki her bir nokta farklı eşik değerlerine karşılık gelen duyarlılık ve 1-özgüllük değerlerini belirtmektedir. Düşük yanlış pozitiflik oranlarını veren eşik değerleri, genellikle düşük doğru pozitif oranına da sahip olduğu anlamına gelmektedir. Doğru pozitif oranının yüksek, yanlış pozitif oranının düşük olduğu sonuçlar başarılı sonuçlar olarak adlandırılmaktadır (Netcad Yazılım, 2019).

Çizelge 3.5. Çalışmada kullanılan metrikler ve formülleri

Metrik	Formül ve Açıklama
True Positive Rates (TPR)	TP/(TP+FN)
False Positive Rates (FPR)	FP/(FP+TN)
Precision	TP/(TP+FP)
Recall	TP/(TP+FN)
F1-Measure	2*Precision*Recall/(Precision+Recall)
Accuracy	(TP+TN)/(TP+TN+FP+FN)

4. ARAŞTIRMA BULGULARI VE TARTIŞMA

Bu çalışmada iki farklı veri seti üzerinde sınıflandırma yöntemlerinin performansları değerlendirilmiştir. Birinci veri seti ile yapılan uygulamada kullanılan veriler (Credit Card Fraud Detection, 2019), Eylül 2013 yılına ait Avrupalı kredi kartı sahiplerinin 2 gün içerisinde yapılan işlemlerinin yer aldığı 284.807 adet kayıttan oluşmaktadır. Bu verilerden 492 adet işlem verisi, banka tarafından sahte olarak etiketlenmiştir ve tüm veriler içerisinde %0,172 oranında bulunmaktadır. Sınıf dağılımı olarak gerçek olarak işaretlenmiş verilerin, sahte olarak etiketlenmiş verilere oranla sayıca fazla olması nedeniyle dağılımının oldukça dengesiz olduğu bir veri kümesidir.

Bu verilerin (veri seti-I) PCA dönüşümü önceden yapılmış olmakla birlikte sadece nümerik değerlerden oluşmaktadır. Güvenlik sebebiyle özniteliklerin asıl değerleri hakkında detaylı bilgi elde edilememektedir. Veri kümesindeki 'V1, ..., V28' alanları PCA dönüşümü neticesinde gizlenmiş temel bileşenlerdir; yalnızca 'Time' ve 'Amount' niteliklerinin PCA dönüşümü yapılmamıştır. Burada 'time' zaman, 'amount' ise yapılan işlem tutarı anlamındadır. 'Class' özelliği hedef değişken olup, verinin sahte (1) ve gerçek (0) etiketi anlamında ikili (binary) değer taşımaktadır.

Çalışmada kullanılan ikinci veri seti (veri seti-II), 700 adet kayıttan oluşan yapay olarak oluşturulmuş verilerdir. Bu veri kümesinde 420 sahte işlem ve 280 yasal olarak etiketlenen verilerde sınıf dağılımları ilk veri setine göre daha dengeli olduğu için burada ilk yaklaşım olarak ham veriler üzerinde, ikinci yaklaşım olarak normalizasyon ve PCA uygulanarak elde edilen sonuçlar değerlendirilmiştir.

Sahte/gerçek sınıf oranlarındaki dengesizlikler sebebiyle bazı veri setlerinde sadece doğruluk (accuracy) ile performans ölçümü anlamlı olmamaktadır. Bu çalışma kapsamında bizim için en önemli metrikler duyarlılık (TP/Gerçek Pozitif), kesinlik (TP/Tahmini pozitif) ve F1-ölçütü metrikleridir.

Aşağıda belirtilen sınıflandırma yöntemlerinden Naive Bayes, KNN, RF, SVM, Lojistik Regresyon yöntemleri PYTHON programlama dilinde, GRNN yöntemi MATLAB ortamında uygulanmıştır. Her yöntem 10 kez çalıştırılmıştır ve her yöntem için ortalama başarı sonuçları elde edilmiştir.

4.1. Veri Seti-I İçin Elde Edilen Sonuçlar

4.1.1. Naive Bayes Uygulaması

Veri kümesi özellikleri ve algoritmanın uygulama sonuçları sırasıyla Çizelge 4.2, 4.4 ve 4.6'da gösterilmiştir.

i) Tüm değişkenlerin kullanılarak algoritmanın uygulanması

Çizelge 4.1. Veri seti özellikleri

Örnek veri kümesi: 284.807	%80 eğitim, %20 test
Özellikler: 30 (V1, ..., V28, Time, Amount)	

Çizelge 4.2. Naive Bayes uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	1,00	0,98	0,99	-
1	0,06	0,82	0,11	-
avg/total	0,53	0,90	0,55	0,96

ii) Veri kümesinden bazı temel bileşenlerin ('V28', 'V27', 'V26', 'V25', 'V24', 'V23', 'V22', 'V20', 'V15', 'V13', 'V8') özellik çıkarımı yapılarak algoritmanın uygulanması

Çizelge 4.3. Veri seti özellikleri

Örnek veri kümesi: 284.807	%80 eğitim, %20 test
Özellikler: 19	

Çizelge 4.4. Naive Bayes uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	1,00	0,98	0,99	-
1	0,08	0,85	0,16	-
avg/total	0,54	0,92	0,58	0,96

iii) Veri kümesinden bazı temel bileşenlerin özellik çıkarımı yapılarak ve "under-sampling" yöntemi ile algoritmanın uygulanması

Çizelge 4.5. Veri seti özellikleri

Örnek veri kümesi: 492	%80 eğitim, %20 test
Özellikler: 19	

Çizelge 4.6. Naive Bayes uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	0,88	0,98	0,92	-
1	0,98	0,86	0,91	-
avg/total	0,91	0,91	0,91	0,96

4.1.2. K- En Yakın Komşu Uygulaması

KNN sınıflandırıcısı için en yakın komşu değeri k=3 olarak belirlenmiştir.

i. Tüm değişkenler kullanılaak algoritmanın uygulanması

Çizelge 4.7. Veri seti özellikleri

Örnek veri kümesi: 284.807	%80 eğitim, %20 test
Özellikler: 30 (V1, …, V28, Time, Amount)	

Çizelge 4.8. K-En Yakın Komşu uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	1,00	1,00	1,00	-
1	0,94	0,76	0,84	-
avg/total	0,97	0,88	0,92	0,92

ii. Temel bileşenlerin özellik çıkarımı yapılarak algoritmanın uygulanması

Çizelge 4.9. Veri seti özellikleri

Örnek veri kümesi: 284.807	%80 eğitim, %20 test
Özellikler: 19	

Çizelge 4.10. K-En Yakın Komşu uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	1,00	1,00	1,00	-
1	0,95	0,79	0,86	-
avg/total	0,97	0,89	0,93	0,92

iii) Temel bileşenlerin özellik çıkarımı yapılarak ve "under-sampling" yaklaşımı ile algoritmanın uygulanması

Çizelge 4.11. Veri seti özellikleri

Örnek veri kümesi: 492	80% eğitim, %20 test
Özellikler: 19	

Çizelge 4.12. K-En Yakın Komşu uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	0,90	0,98	0,93	-
1	0,98	0,89	0,92	-
avg/total	0,94	0,93	0,92	0,97

4.1.3. Lojistik Regresyon Uygulaması

i) Tüm değişkenlerin kullanılarak algoritmanın uygulanması

Çizelge 4.13. Veri seti özellikleri

Örnek veri kümesi: 284.807	%80 eğitim, %20 test
Özellikler: 30 (V1, …, V28, Time, Amount)	

Çizelge 4.14. Lojistik regresyon uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	1,00	1,00	1,00	-
1	0,89	0,60	0,72	-
avg/total	0,94	0,80	0,86	0,97

ii) Temel bileşenlerin özellik çıkarımı yapılarak algoritmanın uygulanması

Çizelge 4.15. Veri seti özellikleri

Örnek veri kümesi: 284.807	%80 eğitim, %20 test
Özellikler: 19	

Çizelge 4.16. Lojistik Regresyon uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	1,00	1,00	1,00	-
1	0,88	0,60	0,71	-
avg/total	0,94	0,80	0,85	0,97

iii) Temel bileşenlerin özellik çıkarımı yapılarak ve "under-sampling" yaklaşımı ile algoritmanın uygulanması

Çizelge 4.17. Veri seti özellikleri

Örnek veri kümesi: 492	%80 eğitim, %20 test
Özellikler: 19	

Çizelge 4.18. Lojistik Regresyon uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	0,92	0,97	0,94	-
1	0,97	0,91	0,94	-
avg/total	0,95	0,94	0,94	0,98

4.1.4. RF Uygulaması

i) Tüm değişkenlerin kullanılarak algoritmanın uygulanması

Çizelge 4.19. Veri seti özellikleri

Örnek veri kümesi: 284.807	%80 eğitim, %20 test
Özellikler: 30 (V1, …, V28, Time, Amount)	

Çizelge 4.20. RF uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	1,00	1,00	1,00	-
1	0,95	0,77	0,85	-
avg/total	0,98	0,89	0,93	0,93

ii) Temel bileşenlerin özellik çıkarımı yapılarak algoritmanın uygulanması

Çizelge 4.21. Veri seti özellikleri

Örnek veri kümesi: 284.807	%80 eğitim, %20 test
Özellikler: 19	

Çizelge 4.22. RF uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	1,00	1,00	1,00	-
1	0,94	0,78	0,85	-
avg/total	0,97	0,89	0,93	0,92

iii) Temel bileşenlerin özellik çıkarımı yapılarak ve "under-sampling" yaklaşımı ile algoritmanın uygulanması

Çizelge 4.23. Veri seti özellikleri

Örnek veri kümesi: 492	%80 eğitim, %20 test
Özellikler: 19	

Çizelge 4.24. RF uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	0,91	0,97	0,94	-
1	0,97	0,90	0,94	-
avg/total	0,94	0,94	0,94	0,97

4.1.5. GRNN Uygulaması

iii) Temel bileşenlerin özellik çıkarımı yapılarak ve "under-sampling" yaklaşımı ile algoritmanın uygulanması. Bu yöntem MATLAB ile uygulanmıştır.

Çizelge 4.25. Veri seti özellikleri

Örnek veri kümesi: 492	%80 eğitim, %20 test
Özellikler: 19	

Çizelge 4.26. GRNN uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	0,91	0,95	0,93	-
1	0,95	0,90	0,92	-
avg/total	0,93	0,93	0,93	0,94

4.1.6. SVM Uygulaması

i) Tüm değişkenlerin kullanılarak algoritmanın uygulanması

Çizelge 4.27. Veri Seti Özellikleri

Örnek veri kümesi: 284.807	%80 eğitim, %20 test
Özellikler: 30 (V1, …, V28, Time, Amount)	

Çizelge 4.28. SVM uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	1,00	1,00	1,00	-
1	0,97	0,60	0,74	-
avg/total	0,98	0,80	0,87	0,95

ii) Temel bileşenlerin özellik çıkarımı yapılarak algoritmanın uygulanması

Çizelge 4.29. Veri seti özellikleri

Örnek veri kümesi: 284.807	%80 eğitim, %20 test
Özellikler: 19	

Çizelge 4.30. SVM uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	1,00	1,00	1,00	-
1	0,95	0,64	0,76	-
avg/total	0,98	0,82	0,94	0,94

iii) Temel bileşenlerin özellik çıkarımı yapılarak ve "under-sampling" yaklaşımı ile algoritmanın uygulanması

Çizelge 4.31. Veri seti özellikleri

Örnek veri kümesi: 492	%80 eğitim, %20 test
Özellikler: 19	

Çizelge 4.32. SVM uygulaması doğruluk sonuçları

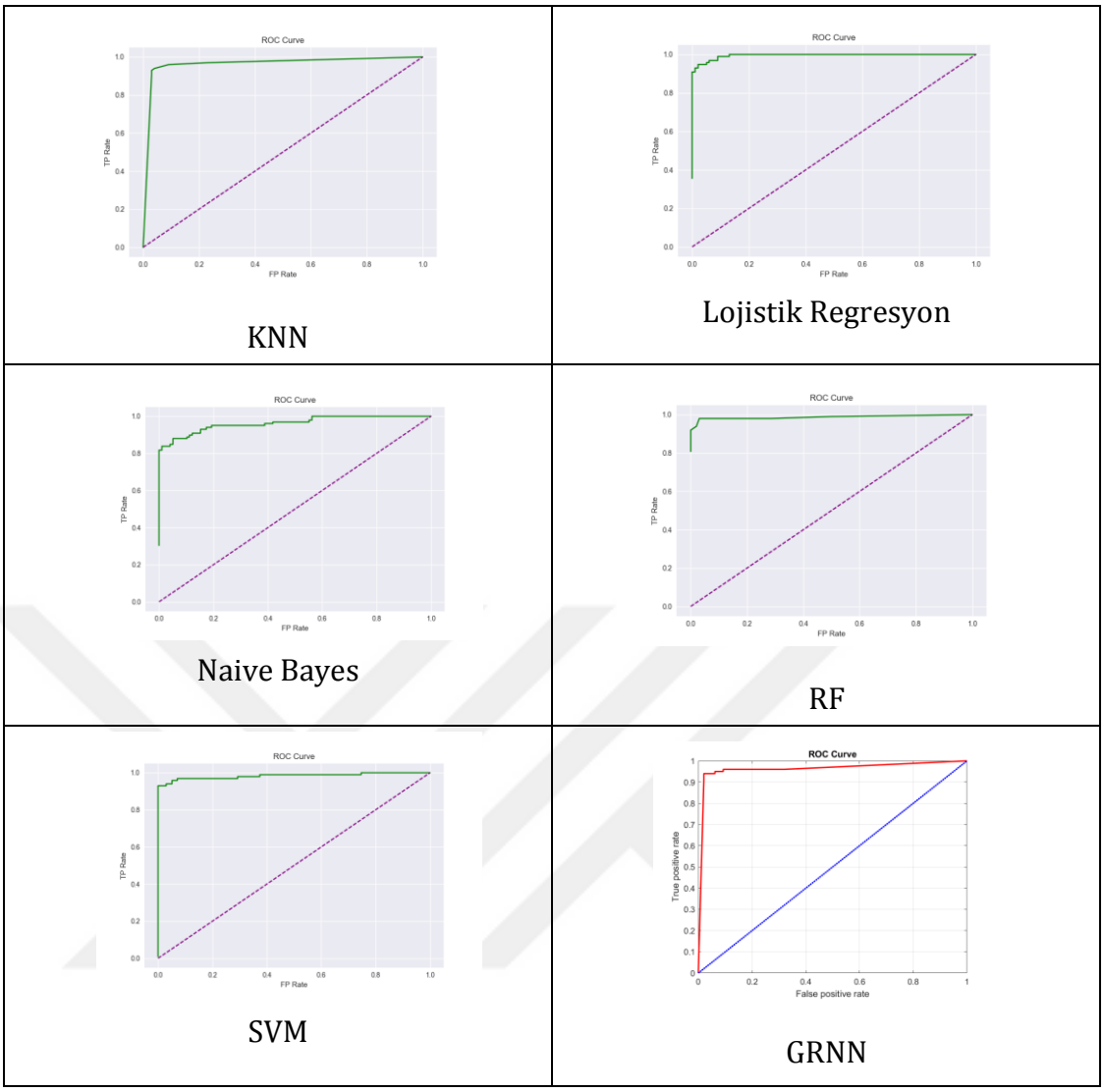
Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	0,90	0,96	0,93	-
1	0,96	0,90	0,93	-
avg/total	0,93	0,93	0,93	0,97

Yukarıda belirtilen uygulamalardan Naive Bayes, Lojistik Regresyon, KNN, RF ve SVM sınıflandırıcıları PYTHON programlama dilinde, GRNN yöntemi de MATLAB üzerinde uygulanmıştır. Her algoritmada üç farklı yaklaşım ile yapılan

uygulamalar sonucunda sınıflandırıcıların başarı oranları ölçümlenmiştir. Birinci deneyde tüm veri kümesi ile özellik çıkarımı yapılmadan; özelliklerin sınıflandırma üzerindeki etkisinin belirlenmesi amacıyla performans sonuçları elde edilmiştir. İkinci deneyde, veri setindeki özelliklerin sınıf dağılımları incelenerek sınıflandırmaya etki etmeyeceği düşünülen ('V28', 'V27', 'V26', 'V25', 'V24', 'V23', 'V22', 'V20', 'V15', 'V13', 'V8') özelliklerinin çıkarılması ile performans sonuçlarına ulaşılmıştır. Üçüncü durumda ise belirlenen bu özellik çıkarımlarına ek olarak "under-sampling" yöntemi de uygulanarak tüm algoritmaların sonuçları Çizelge 4.33'te gösterildiği gibi elde edilmiştir.

Çizelge 4.33. Öznitelik seçimlerine göre algoritmaların başarı sonuçları

Sınıflandırıcı	**Öznitelik seçim durumu**	**Keskinlik**	**Duyarlılık**	**F1-Ölçütü**	**ROC Oranı**
Naive Bayes	i	0,53	0,50	0,55	0,96
	ii	0,54	0,92	0,58	0,96
	iii	0,91	0,91	0,91	0,96
KNN	i	0,97	0,88	0,92	0,92
	ii	0,97	0,99	0,93	0,92
	iii	0,94	0,93	0,92	0,97
Lojistik Regresyon	i	0,94	0,80	0,86	0,97
	ii	0,94	0,80	0,85	0,97
	iii	**0,95**	**0,94**	**0,94**	**0,98**
RF	i	0,98	0,89	0,93	0,93
	ii	0,97	0,89	0,93	0,92
	iii	0,94	0,94	0,94	0,97
GRNN	iii	0,93	0,93	0,93	0,94
SVM	i	0,98	0,80	0,87	0,95
	ii	0,98	0,82	0,94	0,94
	iii	0,93	0,93	0,93	0,97



Şekil 3.17. Tüm algoritmaların "under-sampling" yöntemi ile elde edilen ROC eğrilerinin gösterimi

4.2. Veri Seti–II İçin Elde Edilen Sonuçlar

Bu kısımda kullanılan veri seti için öncelikle sahteciliğin tanınmasında değerlendirilen başlıca özellikler temel alınarak Çizelge 4.35'te belirtilen kurallar oluşturulmuştur. Bu kurallara göre 700 kayıt içeren bir yapay veri seti manuel olarak üretilmiştir. Nümerik ve ondalıklı değerlerden oluşan bu veri seti 420 adet sahte işlem ve 280 adet yasal işlem içermektedir. Belirtilen veri seti ile Lojistik Regresyon, Naive Bayes, RF, SVM ve KNN algoritmaları, iki yaklaşım ile uygulanmıştır. İlk yaklaşımda ham veri seti ile, ikinci yaklaşımda verilerin PCA dönüşümlerinin ve normalizasyonunun yapılarak algoritmaların başarıları

ölçülmüştür. Oluşturulan yapay veri setinin öznitelikleri Çizelge 4.34'te belirtildiği gibidir.

Çizelge 4.34. Yapay veri seti öznitelikleri

Öznitelik	**Öznitelik açıklaması**
X1	E-posta uzunluğu (nümerik): uzun (T)/normal(F)
X2	IP adresi (nümerik): kara listedeki ülkeler(T)/normal(F)
X3	Fatura adresi ve Teslimat adresinin farklı olma durumu (nümerik): farklı (T)/aynı(F)
X4	Fatura adresi, teslimat adresi ve IP adreslerinin yakınlık durumu (nümerik): IP, fatura, teslimat adresleri birbirinden uzak(T)/normal(F)
X5	Üyelik durumu (nümerik): yeni üye (T)/yeni üye değil(F)
X6	Tutar bilgisi (ondalıklı): çok yüksek ya da çok az tutar (T)/normal (F)
X7	Zaman aralığı (nümerik): Sahteciliğin yoğun olduğu zaman dilimi (T)/normal(F)
Sınıf	İşlemin sınıfı (nümerik): (sahte(T)/yasal(F))

Çizelge 4.35. Özniteliklere göre oluşturulan kurallar

Durum	X1	X2	X3	X4	X5	X6	X7	Sınıf
1	T	T	T	T	T	T	T	T
2	T	T	F	T	T	T	T	T
3	T	T	F	T	T	T	F	T
4	T	T	F	T	T	F	F	T
5	T	T	F	T	T	F	F	T
6	T	F	T	T	T	T	T	T
7	T	F	F	T	T	T	F	T
8	T	F	F	T	T	T	T	T
9	F	F	F	F	T	F	F	F
10	F	F	F	F	F	F	F	F
11	T	F	T	T	T	F	F	T

4.2.1. Lojistik Regresyon Uygulaması

Lojistik Regresyon uygulamasında Çizelge 4.36'da belirtildiği gibi tüm veri seti kullanılarak; ilk yaklaşımda ham veri seti için (Çizelge 4.37), ikinci yaklaşımda verilere PCA ve normalizasyon uygulanarak doğruluk sonuçlarına bakılmıştır (Çizelge 4.38).

i) Ham veri seti kullanılarak algoritmanın uygulanması

Çizelge 4.36. Veri seti özellikleri

Örnek veri kümesi: 700	%80 eğitim, %20 test
Özellikler: 7	

Çizelge 4.37. Lojistik Regresyon uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	0,92	0,96	0,94	-
1	0,97	0,95	0,96	-
avg/total	0,94	0,95	0,95	0,98

ii) Normalizasyon ve PCA dönüşümü yapılarak algoritmanın uygulanması

Çizelge 4.38. Lojistik Regresyon uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	0,96	0,96	0,96	-
1	0,98	0,98	0,97	-
avg/total	0,97	0,97	0,97	0,99

4.2.2. KNN Uygulaması

KNN uygulamasında Çizelge 4.39'da belirtildiği gibi tüm veri seti kullanılarak; ilk yaklaşımda ham veri seti için (Çizelge 4.40), ikinci yaklaşımda verilere PCA ve normalizasyon uygulanarak doğruluk sonuçlarına bakılmıştır (Çizelge 4.41).

i) Ham veri seti kullanılarak algoritmanın uygulanması

Çizelge 4.39. Veri seti özellikleri

Örnek veri kümesi: 700	%80 eğitim, %20 test
Özellikler: 7	

Çizelge 4.40. KNN uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	0,90	0,87	0,88	-
1	0,92	0,93	0,92	-
avg/total	0,91	0,90	0,90	0,96

ii) Normalizasyon ve PCA dönüşümü yapılarak algoritmanın uygulanması

Çizelge 4.41. KNN uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	0,92	0,97	0,95	-
1	0,98	0,95	0,96	-
avg/total	0,95	0,96	0,95	0,98

4.2.3. Naive Bayes Uygulaması

Naive Bayes uygulamasında Çizelge 4.42'de belirtildiği gibi tüm veri seti kullanılarak; ilk yaklaşımda ham veri seti için (Çizelge 4.43), ikinci yaklaşımda verilere PCA ve normalizasyon uygulanarak doğruluk sonuçlarına bakılmıştır (Çizelge 4.44).

i) Ham veri seti kullanılarak algoritmanın uygulanması

Çizelge 4.42. Veri seti özellikleri

Örnek veri kümesi: 700	%80 eğitim, %20 test
Özellikler: 7	

Çizelge 4.43. Naive Bayes uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	0,94	0,96	0,95	-
1	0,98	0,96	0,97	-
avg/total	0,96	0,96	0,96	0,98

ii) Normalizasyon ve PCA dönüşümü yapılarak algoritmanın uygulanması

Çizelge 4.44. Naive Bayes uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	0,92	0,95	0,93	-
1	0,96	0,94	0,95	-
avg/total	0,94	0,95	0,94	0,98

4.2.4. RF Uygulaması

Rassal Orman uygulamasında Çizelge 4.45'te belirtildiği gibi tüm veri seti kullanılarak, ilk yaklaşımda ham veri seti için (Çizelge 4.46), ikinci yaklaşımda

verilere PCA ve normalizasyon uygulanarak doğruluk sonuçlarına bakılmıştır (Çizelge 4.47).

i) Ham veri seti kullanılarak algoritmanın uygulanması

Çizelge 4.45. Veri seti özellikleri

Örnek veri kümesi: 700	%80 eğitim, %20 test
Özellikler: 7	

Çizelge 4.46. RF uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	0,91	0,96	0,93	-
1	0,97	0,94	0,95	-
avg/total	0,94	0,95	0,94	0,98

ii) Normalizasyon ve PCA dönüşümü yapılarak algoritmanın uygulanması

Çizelge 4.47. RF uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	0,96	0,96	0,96	-
1	0,97	0,98	0,97	-
avg/total	0,96	0,97	0,96	0,98

4.2.5. SVM Uygulaması

Lojistik Regresyon uygulamasında Çizelge 4.48'de belirtildiği gibi tüm veri seti kullanılarak, ilk yaklaşımda ham veri seti için (Çizelge 4.49), ikinci yaklaşımda verilere PCA ve normalizasyon uygulanarak doğruluk sonuçlarına bakılmıştır (Çizelge 4.50).

i) Ham veri seti kullanılarak algoritmanın uygulanması

Çizelge 4.48. Veri Seti Özellikleri

Örnek veri kümesi: 700	%80 eğitim, %20 test
Özellikler: 7	

Çizelge 4.49. SVM uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	0,93	0,89	0,91	-
1	0,93	0,96	0,94	-
avg/total	0,93	0,92	0,93	0,98

ii) Normalizasyon ve PCA dönüşümü yapılarak algoritmanın uygulanması

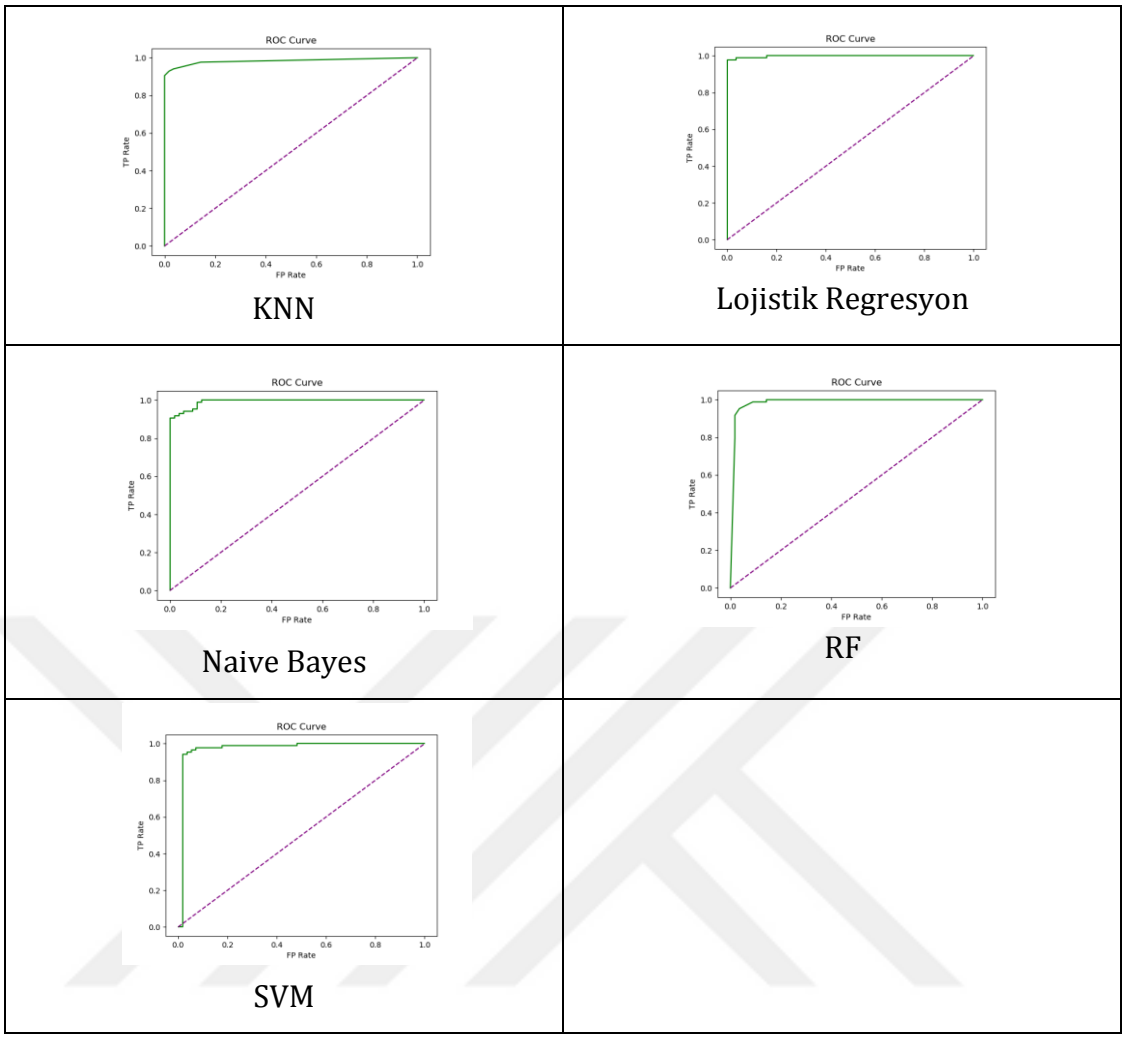
Çizelge 4.50. SVM uygulaması doğruluk sonuçları

Sınıf	Keskinlik	Duyarlılık	F1-Ölçütü	ROC Oranı
0	0,93	0,96	0,95	-
1	0,97	0,95	0,96	-
avg/total	0,95	0,96	0,95	0,98

Yukarıda belirtilen Naive Bayes, Lojistik Regresyon, KNN, RF ve SVM sınıflandırıcıları PYTHON programlama dilinde yapay veri seti üzerinde ilk yöntemde tüm veri kümesi ile; ikinci yöntemde veri kümesine PCA dönüşümü ve normalizasyon yapılarak uygulanmıştır. Tüm algoritmaların sonuçları Çizelge 4.51'de gösterildiği gibi elde edilmiştir.

Çizelge 4.51. Algoritmaların başarı sonuçları

Sınıflandırıcı	**Durum**	**Keskinlik**	**Duyarlılık**	**F1-Ölçütü**	**ROC Oranı**
Naive Bayes	i	0,96	0,96	0,96	0,98
	ii	0,94	0,95	0,94	0,98
KNN	i	0,91	0,90	0,90	0,96
	ii	0,95	0,96	0,95	0,98
Lojistik Regresyon	i	0,94	0,95	0,95	0,98
	ii	**0,97**	**0,97**	**0,97**	**0,99**
RF	i	0,94	0,95	0,94	0,98
	ii	0,96	0,97	0,96	0,98
SVM	i	0,93	0,92	0,93	0,98
	ii	0,95	0,96	0,95	0,98



Şekil 3.18. Yapay veri setinin normalizasyon ve PCA dönüşümlerinin yapılarak elde edilen ROC eğrilerinin gösterimi

5. SONUÇ VE ÖNERİLER

Bu tez çalışmasında temel bir e-ticaret altyapı yazılımının tasarlanarak e-ticaret sistemlerinde oluşabilecek sahteciliklerin araştırılması ve bu sahteciliklerin tespitine yönelik tahmine dayalı bir sistem önerilmiştir. Mevcut e-ticaret yazılımları göz önünde bulundurularak yapılan araştırmalar sonucunda, e-ticaret yazılım altyapılarının birbirlerine benzer yapıda ve eksikliklerinin de ortak yönlerde olduğu gözlemlenmiştir. Bu eksikliklerden yola çıkılarak temel bir e-ticaret sistemi altyapısı ortaya çıkarılmıştır. Ortaya çıkarılan yapı ile bir e-ticaret site altyapısı kullanılabilir hale getirilmiştir.

Günümüzde sanal ortamdan yapılan alışverişlerde oluşabilecek önemli güvenlik risklerinden birisi de e-ticaret siteleri üzerinden yapılan kredi kartı sahteciliğidir. Bu da tez çalışmamızın odak noktasını oluşturmuştur. Araştırmamızın ilk etaplarında sahtecilik tespitinde kullanılması amacıyla veri kaynağının bulunmasında bazı kısıtlamalarla karşılaşılmıştır. Ülkemizde e-ticaret üzerinden gelen verilerin "kişisel verilerin korunması kanunu" sebebiyle paylaşılması yasal olmadığından bu alanda yapılan çalışmalardaki en büyük kısıtlama gerçek veriler ile çalışmanın çoğunlukla mümkün olamamasıdır. Literatürde sahtecilik tespit sistemine yönelik yapılan çalışmalar da incelendiğinde araştırmacıların karşılaştığı en büyük problemin güvenlik sebebiyle gerçek veriye erişim zorluğundan dolayı bu alanda yapılan çalışmaları sınırladığı görülmektedir. Bu verilere alternatif olarak literatür veri setlerinin yanı sıra gerçek verilere benzer yapay verilerin oluşturularak uygulamaların yapılması da mümkündür.

Bu tez çalışmasında sahtecilik tespitine yönelik iki farklı veri seti kullanılarak tahmin yöntemlerinin sonuçları elde edilmiştir. İlk veri seti olarak, 2013 yılına ait Avrupa'da kredi kartı sahiplerinin işlemlerinin yer aldığı bir veri seti kullanılmıştır. Veri seti sınıf dağılımı olarak oldukça dengesiz (imbalanced dataset) bir yapıya sahiptir. Öngörülen sınıflandırma algoritmaları veri seti-I kullanılarak üç farklı yaklaşım ile değerlendirilmesi sonucunda uygulanan tüm yöntemler üzerinde en etkili yaklaşımın verilerdeki sınıf dağılımını dengelemesi

sebebiyle “under-sampling” yaklaşımının performansı önemli ölçüde olumlu etkilediği görülmüştür. Veriler üzerinde tüm algoritmalar içerisinde en düşük başarı oranında Naive Bayes sınıflandırıcısı olmuştur. KNN yöntemi, Naive Bayes yöntemine göre sınıflandırmada daha başarılı ancak uygulanan her üç deneyde de performans olarak daha yavaş çalıştığı ve “under-sampling” yaklaşımı ile nispeten bu durumun düzeldiği gözlemlenmiştir. Tüm algoritmalar içerisinde Lojistik Regresyon yöntemi diğer yöntemlere göre %94 F1 ölçütü ve %97 ROC AUC alanı başarı oranlarıyla daha etkili bir sınıflandırıcı olarak ölçümlenmiştir.

İkinci veri seti için 700 adet kayıttan oluşan yapay bir veri seti oluşturulmuştur. Oluşturulan bu veri setinde veri dağılımı daha dengeli olduğu için “under-sampling” yöntemi uygulamaya katılmadan, birinci yaklaşımda ham veri seti kullanılarak sınıflandırma üzerindeki performanslarına bakılmıştır. İkinci yaklaşımda ise veri kümesine PCA dönüşümü ve normalizasyon uygulanarak %97 F1 ölçütü ve %99 ROC değeriyle yine Lojistik Regresyon yöntemi bu veri seti üzerinde de daha etkili bir sınıflandırıcı olarak ölçümlenmiştir. Bu da önerilen yöntemler ile yapay veri seti kullanımının e-ticaret sitelerinde kredi kartı sahteciliğini tanımada etkili olduğunu göstermektedir. Çalışmada önerilen sistem farklı veri setlerine uyarlanarak kullanılabilir.

KAYNAKLAR

Alam, F., Pachauri, S., 2017. Comperative Study of J48, Naive Bayes and One-R Classification Techinique for Credit Card Fraud Detection using Weka, 1731-1743

Bertrand Lebichot, Yann-Ael Le Borgne, Liyun He, Frederic Oblé, Gianluca Bontempi, 2019. Deep-Learning Domain Adaptation Techniques for Credit Cards Fraud Detection. 10.1007/978-3-030-16841-4_8.

Carcillo, Fabrizio; Dal Pozzolo, Andrea; Le Borgne, Yann-Aël; Caelen, Olivier; Mazzer, Yannis; Bontempi, Gianluca., 2018a. Scarff: a scalable framework for streaming credit card fraud detection with Spark, Information fusion,41, 182-194, Elsevier

Carcillo, Fabrizio; Le Borgne, Yann-Aël; Caelen, Olivier; Bontempi, Gianluca., 2018b. Streaming active learning strategies for real-life credit card fraud detection: assessment and visualization, International Journal of Data Science and Analytics, 5,4,285-300, Springer International Publishing

Code Project, 2019. Erişim Tarihi: 25.05.2019. Simple Modularity through Areas in ASP.NET MVC. https://www.codeproject.com/Articles/874286/Simple-Modularity-through-Areas-in-ASP-NET-MVC

Code curmudgeon, 2019. SQLi Hall-of-Shame. Erişim Tarihi: 10.03.2019. https://codecurmudgeon.com/wp/sql-injection-hall-of-shame/

C# Corner, 2019. Simple Modularity through Areas in MVC. Erişim Tarihi: 07.03.2019. https://www.c-sharpcorner.com/article/simple-modularity-through-areas-in-asp-net-mvc/

Carcillo, Fabrizio & Le Borgne, Yann-Aël & Caelen, Olivier & Kessaci, Yacine & Oblé, Frédéric & Bontempi, Gianluca, 2019. Combining Unsupervised and Supervised Learning in Credit Card Fraud Detection Information Sciences, 10.1016/j.ins.2019.05.042.

Kaggle, 2019. Credit Card Fraud Detection. Erişim Tarihi: 21.02.2019. https://www.kaggle.com/mlg-ulb/creditcardfraud

Lima, R., 2015. A Fraud detection model based on feature selection and undersampling applied to Web payment systems, IEEE, 978-1-4673-9618-9/15

Mevzuat Bilgi Sistemi, 2019. Kişisel verilerin Korunması Kanunu. Erişim Tarihi: 16.06.2019. https://www.mevzuat.gov.tr/MevzuatMetin/1.5.6698.pdf

Netcad Yazılım, 2019. ROC (Receiver Operating Characteristic) Analizi. Erişim Tarihi:22.03.2019. https://portal.netcad.com.tr/display/HELP/ROC+Analizi

Pozzolo, D., 2015. Adaptive Machine learning for credit card fraud detection ULB MLG PhD thesis (supervised by G. Bontempi)

Pozzolo, D., 2014. Learned lessons in credit card fraud detection from a practitioner perspective, Expert systems with applications, 41, 10, 4915-4928

Pozzolo, D., 2015. Calibrating Probability with Undersampling for Unbalanced Classification. In Symposium on Computational Intelligence and Data Mining (CIDM), IEEE

Pozzolo, D., 2018. Credit card fraud detection: a realistic modeling and a novel learning strategy, IEEE transactions on neural networks and learning systems, 29, 8, 3784-3797

Sharma, S., 2018. An Approach to Detect Credit Card Frauds using Attribute Selection and Ensemble Techniques, International Journal of Computer Applications, 0975-8887

Stack overflow, 2019. Most Popular Technologies. Erişim Tarihi: 17.04.2019. https://insights.stackoverflow.com/survey/2018#most-popular-technologies

Sudha, C., 2017. Credit Card Fraud Detection in Internet using K-Nearest Neighbor Algorithm, ISSN, 2321-5992

Towards Data Science, 2019. Understanding Confisuon Matrix. Erişim Tarihi: 05.04.2019. https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

Xuan, S., 2018. Random Forest for Credit Card Fraud Detection, IEEE, 5386-5053

Yee, O., 2018. Credit Card Fraud Detection Using Machine Learning As Data Mining Technique, e-ISSN, 2289-8131

Zarepoor, M., 2015. Application of Credit Card Fraud Detection: Based on Bagging Ensemble Classifier, Elsevier, 679-686

ÖZGEÇMİŞ

Adı Soyadı : Fatma Serap ÖRENLİ

Doğum Yeri ve Yılı : Antalya, 1983

Medeni Hali : Bekar

Yabancı Dil : İngilizce

E-Posta : serap.orenli@gmail.com

Eğitim Durumu

Lise : Akdeniz Koleji, 2000

Üniversite : Maltepe Üniversitesi, Mühendislik ve Doğa Bilimleri Fakültesi, Bilgisayar Mühendisliği

EF Oxford/İngiltere 2007

Mesleki Deneyim

Türk Telekom Antalya İl Müdürlüğü, Bilgi İşlem Staj, (07.2005-08.2005)

Anadolu Hastanesi, Yazılım Geliştirme, 12.2010-03.2011

Cosoft Yazılım & Mühendislik, Yazılım Geliştirme 2015-… (halen)