

**KASKO SİGORTASINDA MAKİNE ÖĞRENMESİ
YÖNTEMLERİ İLE HASARLI/HASARSIZ DURUM TAHMİNİ**

**ESTIMATION OF CLAIM/NO-CLAIM AUTOMOBILE
INSURNACE WITH MACHINE LEARNING METHODS**

SEDA KİLİSLİ

PROF. DR KASIRGA YILDIRAK

Tez Danışmanı

Hacettepe Üniversitesi

Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin

Aktüerya Bilimleri Anabilim Dalı İçin Öngördüğü

YÜKSEK LİSANS TEZİ olarak hazırlanmıştır.

2019

Seda Kilisli'nin hazırladığı “Kasko Sigortasında Makine Öğrenmesi Yöntemleri ile Hasarlı/Hasarsız Durum Tahmini” adlı bu çalışma aşağıdaki jüri tarafından AKTÜERYA BİLİMLERİ ANABİLİM DALI'nda YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

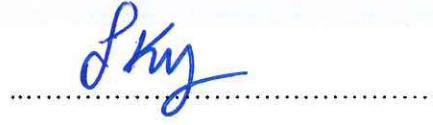
Doç. Dr. Çağdaş Hakan ALADAĞ

Başkan



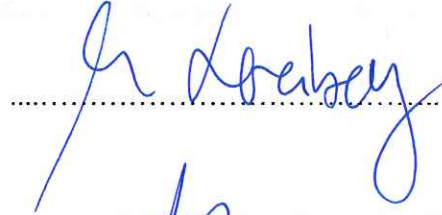
Prof. Dr. Kasırga YILDIRAK

Danışman



Dr. Öğr. Üyesi Uğur KARABEY

Üye



Dr. Öğr. Üyesi Yasemin GENÇTÜRK

Üye



Doç.Dr. Özge Sezgin ALP

Üye



Bu tez Hacettepe Üniversitesi Fen Bilimleri Enstitüsü tarafından YÜKSEK LİSANS TEZİ olarak //..... tarihinde onaylanmıştır.

Prof. Dr. Menemşe GÜMÜŞDERELİOĞLU

Fen Bilimleri Enstitüsü Müdürü



*Sevgili Annem, Babam,
Ablalarım ve Yeğenlerime İthafen,*

ETİK

Hacettepe Üniversitesi Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmada,

- tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- ve bu tezin herhangi bir bölümünü bu üniversitede veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

11 / 09 / 2013

SEDA KİLİSLİ

YAYINLANMA FİKRİ MÜLKİYET HAKKLARI BEYANI

Enstitü tarafından onaylanan lisansüstü tezimin/raporumun tamamını veya herhangi bir kısmını, basılı (kağıt) ve elektronik formatta arşivleme ve aşağıda verilen koşullarla kullanıma açma iznini Hacettepe üniversitesine verdiğimi bildiririm. Bu izinle Üniversiteye verilen kullanım hakları dışındaki tüm fikri mülkiyet haklarım bende kalacak, tezimin tamamının ya da bir bölümünün gelecekteki çalışmalarda (makale, kitap, lisans ve patent vb.) kullanım hakları bana ait olacaktır.

Tezin kendi orijinal çalışmam olduğunu, başkalarının haklarını ihlal etmediğimi ve tezimin tek yetkili sahibi olduğumu beyan ve taahhüt ederim. Tezimde yer alan telif hakkı bulunan ve sahiplerinden yazılı izin alınarak kullanması zorunlu metinlerin yazılı izin alarak kullandığımı ve istenildiğinde suretlerini Üniversiteye teslim etmeyi taahhüt ederim.

Yükseköğretim Kurulu tarafından yayınlanan “*Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge*” kapsamında tezim aşağıda belirtilen koşullar haricince YÖK Ulusal Tez Merkezi / H. Ü. Kütüphaneleri Açık Erişim Sisteminde erişime açılır.

- ☐ Enstitü / Fakülte yönetim kurulu kararı ile tezimin erişime açılması mezuniyet tarihimden itibaren 2 yıl ertelenmiştir.
- ☐ Enstitü / Fakülte yönetim kurulu gerekçeli kararı ile tezimin erişime açılması mezuniyet tarihimden itibaren ay ertelenmiştir.
- ☐ Tezim ile ilgili gizlilik kararı verilmiştir.

11.03.2018


SEDA KİLİSLİ

ÖZET

KASKO SİGORTASINDA MAKİNE ÖĞRENMESİ

YÖNTEMLERİ İLE HASARLI/HASARSIZ

DURUM TAHMİNİ

Seda KİLİSLİ

Yüksek Lisans, Aktüerya Bilimleri Bölümü
Tez Danışmanı: Prof. Dr. Kasırga YILDIRAK
Eylül 2019, 60 sayfa

Bu çalışmanın amacı sigorta şirketine yeni katılacak sigortalıların hasar getirip getirmeme durumunu makine öğrenmesi yöntemleri ile tahmin etmektir. Trafikte ki araç sayısının her geçen yıl artması olası kaza riskinin artmasını da beraberinde getirmektedir. Artan kaza sayıları sigorta şirketlerinin maliyetlerini arttırmakta dolayısıyla bu artış kasko primlerine de yansımaktadır. Ancak sigorta şirketlerindeki fiyat rekabetinden dolayı optimal primin altında yapılan satışlar şirketlerin karlılık hedeflerinin gerisinde kalmalarına sebep olmaktadır. Sigorta şirketlerinin karlı olarak büyüebilmeleri için hasarsız müşteri profilinin portföye katılması çok önemlidir. Karlı müşterilerin portföye katılması için bireysel davranışları yakalayacak modellere ihtiyaç duyulmaktadır. Bu nedenle son zamanlarda birçok sektörde kullanılan makine öğrenmesi algoritmalarından faydalanarak en iyi hasarlı/hasarsız sınıf tahmini yapan model belirlenecek ve sigorta sektöründe yaygın olarak kullanılan lojistik regresyon modeli ile karşılaştırılacaktır.

Anahtar Kelimeler: Makine Öğrenmesi, Kasko Sigortası, Hasarlı/Hasarsız Tahmin Modeli, Naive Bayes, Sinir Ağı, Karar Ağacı, Rastgele Ormanlar, Topluluk Modelleri, Lojistik Regresyon, KNN

ABSTRACT

ESTIMATION OF CLAIM/NO-CLAIM AUTOMOBILE INSURANCE WITH MACHINE LEARNING METHODS

Seda KİLİSLİ

Master of Science, Department of Actuarial Sciences

Supervisor: Prof. Dr. Kasırga YILDIRAK

September 2019, 60 pages

The aim of this study is to estimate whether new insured who will join insurance company have claim by using machine learning methods. The increase in the number of vehicles in traffic each year brings with it an increase in the risk of accidents. Increasing number of accidents increases the costs of insurance companies and this increase is also reflected in the insurance premiums. However, due to price competition in insurance companies, sales below the optimal premium cause companies to fall behind their profitability targets. In order for insurance companies to maintain their profitability, it is very important to include the profitable insured profile in the portfolio. In order to include the profitable insured in the portfolio, models that simulate individual behaviors are needed. Therefore, by using machine learning algorithms used in many sectors recently, the model that makes the best estimate of claimed policies in the portfolio is determined and compared with the logistic regression model commonly used in the insurance sector.

Keywords: Machine Learning, Casco Insurance, Claim Predictive Model, Naive Bayes, Neural Network, Decision Tree, Random Forest, Boosting, Logistic Regression, KNN

TEŞEKKÜR

Bu çalışmam boyunca bana yol gösteren ve destekleyen tez danışmanım sayın Prof. Dr. Kasırğa YILDIRAK'a,

Tez savunma jürisinde yer alan değerli düşünce ve görüşleriyle tezime katkı sağlayan Doç. Dr. Hakan Çağdaş Aladağ, Dr. Öğr. Üyesi Uğur KARABEY, Dr. Öğr. Üyesi Yasemin GENÇ-TÜRK, Doç. Dr. Özge Sezgin ALP'e,

Yüksek lisans eğitimim boyunca desteğini ve bilgisini benden esirgemeyen hocalarım Doç. Dr. Şule ŞAHİN, Dr. Öğr. Üy. Başak BULUT KARAGEYİK'e,

Bu süreç boyunca her konuda motivasyon ve desteği sağlayan, sevgili arkadaşım Mustafa Asım ÖZALP'e,

Tez çalışmam boyunca benden hoşgörüsünü ve desteğini esirgemeyen sevgili arkadaşlarım, Hande YEŞİLMADEN, Cem YAVRUM, Zehra Ece GÜLŞEN, Merve SARINÇ, Çisem SURAY, Merve TUNÇBİLEK, Aynur COSKUN, Aslı ÖZTÜRK başta olmak üzere tüm arkadaşlarıma,

Tez çalışmam boyunca beni destekleyen tüm iş arkadaşlarıma,

Tez çalışmam boyunca destek olan, Yücel AYDIN'a, eğitim hayatım boyunca beni her konuda destekleyen eniştem Alp Cevat BOZBIYIK ve ablalarım Nazlı Canan KİLİSLİ, Fatma Sinem BOZBIYIK, Betül AYDIN'a, verdikleri enerjiyle tüm dertleri unutturan sevgili yeğenlerim Alp Yiğit BOZBIYIK ve Deniz Mert BOZBIYIK'a, tez yazma motivasyonum olan ve beni bugünlere getiren sevgili babam Sait KİLİSLİ ve sevgili annem Zeynep KİLİSLİ'ye, sonsuz teşekkürlerimi sunarım.

İÇİNDEKİLER

ÖZET	ii
ABSTARCT	iii
TEŞEKKÜR	iv
İÇİNDEKİLER	vi
ŞEKİLLER	vii
ÇİZELGELER	viii
SİMGELER VE KISALTMALAR	ix
1 GİRİŞ	1
2 GENEL BİLGİLER	6
2.1 Danışmanlı Makine Öğrenmesi Yöntemleri	6
2.1.1 K En Yakın Komşu (K Nearets Neighbor) Yöntemi	6
2.1.2 Karar Ağacı (Decision Trees) Yöntemleri	7
2.1.2.1 Regresyon Ağaçları	8
2.1.2.2 Sınıflandırma Ağaçları	10
2.1.3 Topluluk Yöntemleri (Ensemble Methods)	11
2.1.3.1 Bagging	11
2.1.3.2 Boosting-Ada Boost	12
2.1.3.3 Rastgele Orman (Random Forests)	14
2.1.4 Destek Vektör Makineleri (Support Vektor Machines) Yöntemi	15
2.1.5 Naive Bayes Yöntemi	18
2.1.6 Sinir Ağları (Neural Network) Yöntemi	20
2.2 Lojistik Regresyon	23
3 UYGULAMA	24
3.1 Verinin Niteliği	24
3.2 Hasarlı/Hasarsız (0/1) Tahmin Modelleri	27
3.2.1 K En Yakın Komşu (kNN) Model Seçimi	27
3.2.2 Karar Ağaçları Model Seçimi	30
3.2.3 AdaBoost Model Seçimi	33
3.2.4 Rastgele Orman AdaBoost Model Seçimi	34
3.2.5 Destek Vektör Makineleri Model Seçimi	34
3.2.6 Sinir Ağları Model Seçimi	35
3.2.6.1 Optimizasyon Yöntemi Seçimi	36

3.2.6.2 Tek Katmanlı Yapıda Nöron Sayısı Seçimi	38
3.2.6.3 Çift Katmanlı Yapıda Nöron Sayısı Seçimi	43
3.2.7 Lojistik Regresyon Model Seçimi	49
3.3 Seçilen Modellerin Karşılaştırılması	50
4 SONUÇLAR	55
EKLER	56
KAYNAKÇA	57



ŞEKİLLER

2.1 kNN-Sınıflandırma	7
2.2 Maksimum Marjin Destek Vektörleri	16
2.3 Katmanlı Sinir Ağı Yapısı	21
3.1 Eğitim Veri Seti Hasarlı/Hasarsız (0/1) Frekans	27
3.2 Test Veri Seti Hasarlı/Hasarsız (0/1) Frekans	27
3.3 Tree %90 – 4 Modeli Karar Ağacı	32
3.4 Seçilen Modeller-0 Sınıf Calibration Plot	53
3.5 Seçilen Modeller-1 Sınıf Calibration Plot	54

ÇİZELGELER

3.1	Simülasyon Değişken, Dağılım ve Parametreleri	24
3.2	Veri Seti ve Türetilen Değişkenler	25
3.3	Model Değişkenleri	26
3.4	Değişkenler Arasındaki Korelasyon	26
3.5	k=99, kNN Modelleri ve Sonuçları	28
3.6	kNN Eucildean Distance k Değeri Belirleme Sonuçları	29
3.7	Karar Ağaçları Model Sonuçları	30
3.8	Karar Ağaçları Budama Model Sonuçları	31
3.9	Karar Ağaçları Modelleri Boğum-Yaprak Sayısı	32
3.10	AdaBoost Modelleri	33
3.11	AdaBoost Model Sonuçları	33
3.12	Rastgele Orman Model ve Sonuçları	34
3.13	SVM Model Sonuçları	35
3.14	Sinir Ağları Optimizasyon Yöntemi Seçim Modelleri	36
3.15	Sinir Ağları Optimizasyon Yöntemi Seçim Model Sonuçları	37
3.16	Aktivasyon Fonk. Optimizasyon Yöntemi-Nöron Sayısı Kombinasyonları	39
3.17	Sinir Ağları Tek Katmanlı Nöron Sayısı Belirleme Sonuçları	43
3.18	Sinir Ağları Çift Katman Modelleri	45
3.19	Sinir Ağları Çift Katmanlı Nöron Sayısı Belirleme Sonuçları	48
3.20	Lojistik Regresyon Model Sonuçları	50
3.21	Seçilen Modellerin Sonuçları	51
3.22	Seçilen Modellerin Karmaşıklık Matrisleri	52

SİMGELER VE KISALTMALAR

IRIS: Insurance Regulatory Information System

kNN: k En Yakın Komşu (k Nearest Neighborhood)

NN: Sinir Ağları

SGD: Stochastic Gradient Descent

SVM: Destek Vektör Makineleri (Support Vektör Machines)

RSS: Artık Kareler Toplam (Residual Sum of Squares)

Akt. F.: Aktivasyon Fonksiyonu

Opt: Optimisasyon

G.K: Gizli Katman

N.S.: Nöron Sayısı

1. GİRİŞ

Sigorta sektöründe veri analizi ve portföy yönetimi oldukça önemlidir. Çünkü sigortanın temeli belirsiz riskleri teminat altına alma üzerine kuruludur ve şirketlerin devamlılığı için risklerin iyi yönetilmesi gerekmektedir. Risklerin sınıflandırılması riski yönetmenin temelini oluşturmaktadır. Riskin değerlendirilmesi, fiyatlandırılması için sınıflandırılması gerekmektedir.

Son zamanlarda makine öğrenmesi yöntemleri ile sigortacılıkta sahteciliğin tespit edilmesi, müşteri kaybı analizleri, fiyat optimizasyonu, rezerv, ölümlülük oranları, müşteri davranışları gibi riski sınıflandırmaya yönelik birçok çalışma yapılmaktadır. Ancak makine öğrenmesi yöntemlerinin temelleri geçmişe dayanmakta olup son zamanlarda popüler olmasının sebebi gelişen bilgisayar teknolojileri ile uygulanmasının kolaylaşmasıdır.

Verinin işlenmesi ve anlamlı sonuç çıkarılması için veri madenciliği üzerine geçmişten günümüze birçok çalışma yapılmıştır. Bu çalışmalar klasik istatistiksel yöntemlerin aksine bağımsız ve bağımlı değişkenler arasındaki doğrusal olmayan ilişki durumlarında da veriden anlamlı sonuçlar elde edilebileceğini göstermektedir.

Makine öğrenmesi yöntemlerinden en uzun geçmişe sahip olan sinir ağları ilk kez 1959'da Rosenblatt ve 1960'ta Windrow ve Haff tarafından keşfedilmiş olup 1988'te Collins ve arkadaşları ipotek sigortasının değerlendirme sürecinde verilen kararları sinir ağları yapısı ile sınıflandırmış ve yeni gelen taleplerin değerlendirilmesinde algoritmanın, riski değerlendiren kişilerden daha tutarlı kararlar verdiğini görmüşlerdir (Shapiro ve Jain ,2003). 1996 yılında Van Wezel ve arkadaşları, sigorta şirketinin veri tabanında yer alan müşteri bilgilerini kullanarak sinir ağları tabanlı bir yöntem inşa edip müşteri davranışlarını etkileyen boyutları belirlemeyi denemişlerdir. Bu yöntem boyutları be-

lirlemede kullanılabilecek bir alternatif sunmuştur ancak gerçek veriler üzerinde yapılan testlerin sonucu beklenildiğinden kötü çık-mıştır. Sorunun sebeplerini tartışarak geliştirdikleri yöntemin daha az gürültülü verilerde daha iyi sonuçlar vereceğini ileri sürmüşlerdir (Van Wezel ve ark.,1996). 1997 Vaughn ve arkadaşları sinir ağlarının özel bir durumu olan çok katmanlı algılayıcı öğrenme yöntemini (“Multilayer Perceptron Network, MLP”) kullanarak hayat sigortası yaptıran kişilerin riskini standart ve standart dışı olarak sınıflandırmışlardır. Yöntemin performansını ölçmek için yaptıkları testler sonucunda bu yöntemin riski sınıflandırmada iyi sonuçlar verdiğini gösterilmişleridir (Vaughn ve ark., 1997). 1998’de Brockett ve arkadaşları araç sigortasından gelen bedensel yaralanma hasarlarını geri yayılım algoritması ile sahtecilik derecesine göre sınıflandırmışlar ve yaptıkları testler ile algoritmanın sigortada sahtecilik araştırması yapan kişilerden daha tutarlı ve güvenilir olduğunu ortaya koymuşlardır (Brockett ve ark., 1998). 1999 Magge simülasyonla oluşturulan ve sektörden alınan kaza sigorta verilerini kullanarak sinir ağları, zincir merdiven, en küçük kareler regresyonu ve additive yöntemleri ile rezerv hesaplamış ve her iki veriden elde edilen sonuçları karşılaştırmıştır. Yapılan karşılaştırma sonucunda sinir ağlarının hem mutlak değer hatası hem de yanlışlıkta en iyi performansa sahip olduğunu göstermiştir (Shapiro ve Jain ,2003).

Sınıflandırma yapısı gruplar içindeki maksimum homojenliği ve gruplar arasındaki maksimum heterojenliği sağlamak için tasarlanmıştır. Her kümedeki nesneler bir bakıma birbirine benzer olma eğiliminde ve farklı kümelerdeki nesneler de birbirinden farklı olma eğiliminde olduğundan kümeleme algoritmalarıyla da sınıflandırma çalışmaları yapılmıştır (Anderberg, 1973). 1994 yılında Huang ve arkadaşları sigorta şirketlerindeki finansal bozulmaları tahmin etmek için IRIS oranlarını sinir ağları, k en yakın komşu(kNN), diskriminant analizi ve logit yöntemleri ile modellemişlerdir. Modellerin performanslarını incelendiklerinde kNN algoritmasının örneklemin öngörülmesinde iyi bir performans gösterdiğini ancak örneklem dışı tahminlerde başarısız olduğunu ve en iyi performansa sahip modelin sinir ağları olduğunu görmüşlerdir

(Shapiro ve Jain ,2003).

1990 yıllarında sinir ağılarına alternatif arayışında olan Vapnik ve arkadaşları 1960'lı yıllarının ortalarında iki temel unsuru bilinen (hiper düzlemin destek vektörleri ve kernel çekirdekleri kullanarak hiper düzlemin genişletilmesi) destek vektör makinelerinin (SVM'nin) genel yapısını oluşturmuşlardır (Vapnik ve ark. 1992). 1995 yılında ise Vapnik ve Cortes ayrışmayan durumlar için genişletilmiş SVM tekniğini elde etmiş ve geçmişte kNN, karar ağacı ve sinir ağıları ile yapılmış birkaç uygulamayı SVM algoritması ile denemişlerdir. Bu çalışma neticesinde SVM algoritmasının iyi bir performansa sahip olduğunu göstermişlerdir (Vapnik,1995). Bu algoritmanın geliştirilmesiyle birlikte öğrenme algoritmalarının genelleme yeteneği bağımsız değişkenler yerine marjın kavramıyla ele alınmıştır.1997'de Barlett sinir ağılarını, Freund ve arkadaşları AdaBoost yöntemlerini marj kavramıyla ele almışlardır (Vapnik, 1998).

2009 yılında Caruna ve Mizil son yıllarda gelişen (SVM, sinir ağıları, lojistik regresyon, naive bayes, hafıza temelli, rastgele orman, bagged trees, boosted trees, boosted stumps) danışmanlı öğrenme algoritmalarının performansını ve kalibre yöntemlerinin algoritmaların performansı üzerindeki etkilerini incelemiştir. Kalibre ile boosted trees, SVM, boosted stumps ve naive bayes algoritmalarının performanslarının önemli ölçüde arttığını, rastgele orman algoritmasının performansında ise küçük ama önemli bir gelişme gözlemlemişlerdir. Bu çalışma sonucunda bazı algoritmaların genel olarak daha iyi bazıların ise genel olarak daha kötü olduğunu, ancak çözüm aranan problem ve ölçüme göre bu durumun değişiklik gösterdiğini ileri sürülmüşlerdir (Caruna ve Mizil, 2009).

Gelişen yeni algoritmalar aktüeryal hesaplamalara önemli katkılar sağlamıştır. Bhowmik karar ağacı ve naive bayes algoritmalarını, Gepp ve arkadaşları ise karar ağacı ve sinir ağılarını kullanarak araç sigortalarındaki dolandırıcılığı tespit etme üzerine çalışmışlardır (Bhowmik, 2011; Gepp ve arkadaşları

2012). Pozzolo makine öğrenmesi algoritmaları ile (karar ağacı, rastgele ormanlar, naive bayes, kNN, sinir ağları, SVM) sigortalı kişinin özelliklerine bağlı olarak hasar frekansı için tahmin modeli kurmuş ve algoritmaların performansını karşılaştırmıştır (Pozzolo,2011). Frempong ve arkadaşları karar ağacı ile araç sigortalarındaki hasarlar için tahmin modeli oluşturmuşlar ve böylelikle gelecekteki riskleri tahmin ederek önceden önlem alınabileceğini ileri sürmüşlerdir (Frempong, 2017). Spedicto ve arkadaşları artan fiyat rekabetinden dolayı fiyat optimizasyonunda müşteri davranışlarının daha önemli hale geldiğini ileri sürmüşlerdir. Boosted trees algoritması ve GLM ile fiyat optimizasyon modellerini kurup yöntemlerin avantajlarını ve dezavantajlarını incelemişlerdir (Spedicto,2018).

Son bir yılda sigortacılık alanında yapılan çalışmalar, danışmanlı makine öğrenmesi algoritmaları ile hayat sigortasında risk değerlendirme sürecinin otomatik hale getirilmesi (Boodhun ve Manoj Jayabalan,2018), bireysel kredi sigortalarının rezerv hesabında SVM, sinir ağları ve GLM kullanılarak makine öğrenmesi algoritmalarının rezerv hesaplamalarına alternatif oluşturup oluşturamayacağını incelenmesi (Ticconi,2018), boosted ağaçları ve rastgele ormanlar algoritmaları ile hasar frekans ve şiddet modeli kurarak sigorta tarife planlarının oluşturulmasıdır (Henckaerts ve arkadaşları, 2019).

Uluslararası Aktüerya Derneği (International Actuarial Association) tarafından hayat dışı alanında yapılan çalışmalara istinaden yayınlanan 2017 ve 2018 raporlarında klasik rezerv yöntemleri ile makine öğrenmesi yöntemlerinin bir arada kullanılması ele alınmıştır (ASTIN Report, 2017;2018).

Kasko branşı hayat dışı sigorta şirketlerinin portföyünün büyük bir kısmını oluşturmaktadır. TUİK verileri incelendiğinde 2008 itibari ile son 10 yılda otomobil kullanımının ortalama %7 arttığı görülmektedir. Artan otomobil kullanımıyla birlikte kaza sayıları ve sigorta yaptırma bilinci de artmaktadır. Bu nedenle sigortalanmadan önce müşterinin özelliklerine göre hasar getirip getirmeyeceğini tahmin edebileceğimiz modeller ile ters seçim riskinin

minimize edilmesi amaçlanmaktadır.

Modellemede, Makine Öğrenmesi Yöntemlerinin kullanılmasının sebebi büyük bir data ile çalışıldığında sınıflandırma sorununu ortadan kaldırmaları ve yinelemeli metotlar ile bireysel özellikleri yakalamada başarılı olmalarıdır.

Bu çalışmada makine öğrenmesi algoritmaları ile müşterinin hasarlı/hasarsız sınıf tahmini için modeller oluşturulmuş ve modellerin sonuçları karşılaştırılmıştır. İkinci bölümde kullanılacak algoritmalar açıklanmıştır. Üçüncü bölümde kullanılacak verinin yapısı, veride yapılan değişiklikler, uygulanacak yöntemlerin parametreleri ve çıktılarına yer verilmiştir. Dördüncü bölümde ise yapılan uygulama sonuçlarının genel değerlendirmesi yapılmıştır.

2. GENEL BİLGİLER

Bu bölümde hasarlı/hasarsız sınıf tahmininde, kullanılacak yöntemlerden bahsedilecektir.

Günümüzde birçok alanda yaygın olarak kullanılan makine öğrenmesi algoritmaları ve sigorta sektöründe kategorik verilerin sınıflandırılmasında yaygın olarak kullanılan lojistik regresyon yöntemi ile hasarlı/hasarsız sınıf tahmini yapılacaktır.

Makine öğrenmesi veriyi gruplayarak anlamlı bilgiler elde edilmesini, veriyi belirli yöntemler ile işleyerek veriden çıkan bilgi doğrultusunda sınıf tahmini yapılmasını sağlayan algoritmaların bütünü olup; Danışmanlı Öğrenme (Supervised Learning), Danışmansız Öğrenme (Unsupervised Learning) ve Takviyeli Öğrenme (Reinforcement Learning) olarak 3 gruptan oluşmaktadır.

Sınıf tahmini için kurulacak modellerde danışmanlı öğrenme algoritmaları kullanılacak olup danışmanlı öğrenme yöntemleri; eğitim için alınan örneklemdeki belirli özellikleri kullanarak farklılıklara göre sınıflandırma yapılmasını sağlayan algoritmalarlardır. Test için ayrılan örneklem sınıf tahminlerinde doğruluk oranını ölçmek için kullanılır. Kullanılacak makine öğrenmesi yöntemleri ve açıklamaları aşağıda yer almaktadır.

2.1. Danışmanlı Makine Öğrenmesi Yöntemleri

2.1.1. K En Yakın Komşu (K Nearest Neighbor) Yöntemi

Basit bir parametrik olmayan yöntem olan kNN, k tane en yakın komşusuna göre dataların gruplandırılmasında ve sınıflandırılmasında kullanılmaktadır.

$K(x)$, p boyutlu gerçel bir uzayda herhangi bir x 'in en yakın k noktasını gösteren indeks olarak tanımlanmaktadır. Sınıflandırma x 'e en yakın k nok-

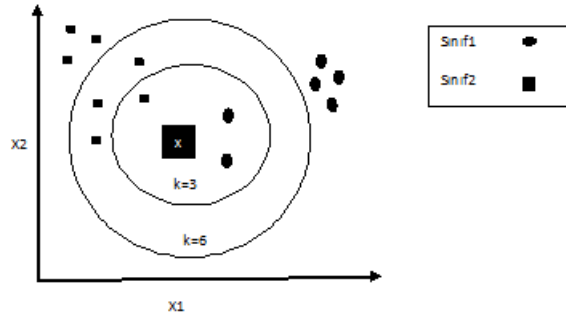
talarına karşılık gelenlerin ortalamasına göre yapılmaktadır (Chen,2013).

$$f(x) = \frac{1}{K} \sum_{i \in K(x)} y_i$$

Girdiler $\binom{n}{k}$ parçadan oluşmakta ve her biri bir k veri kümesine karşılık gelmektedir. Noktalar en yakın oldukları veri kümesine göre konumlanmaktadır. Örneğin i kümesine ait veriler diğer kümelerden ziyade i kümesine yakın noktalardır.

k küçüldükçe, model büyür, varyans yükselir ve yanlılık düşer. Modelin performansı için k değerinin doğru belirlenmesi önemlidir. En iyi k değerini belirlemek için Çapraz Doğrulama (Cross Validation) yöntemi yaygın olarak kullanılmaktadır (Chen,2013).

x değerine karşılık gelen değerın sınıflandırılması Şekil 2.1’de görüldüğü gibi en yakın k komşusunun çoğunluğuna göre belirlenmektedir.



Şekil 2.1: kNN-Sınıflandırma

2.1.2. Karar Ağacı (Decision Trees) Yöntemleri

Karar ağaçları anlaması ve kullanılması kolay doğrusal olmayan popüler yöntemlerden olmasının yanı sıra sınıflandırma problemleri için uygun bir yöntemdir. Sınıflandırma ağaçları ve regresyon ağaçları olarak ikiye ayrılmaktadır. Sınıflandırma ağaçlarında sayısal tahmin yerine evet/hayır gibi nitelik tahmin edilmektedir. Verilerin tamamı yinelemeli olarak farklı küçük grup-

lara ayrılmakta ve bu işlem belirli bir hedefe ulaşılan kadar tekrarlanmaktadır. Ağaç değişkenlere göre tahmin uzayını birkaç basit bölgeye ayırmamızı sağlamaktadır.

2.1.2.1 Regresyon Ağaçları

$X_1, X_2, \dots, X_p$ için olası değerler olan tahmin uzayı, j farklı ve birbiri ile örtüşmeyen $(R_1, R_2, \dots, R_j)$ bölgeye ayrılmaktadır. R_j bölgesine düşen eğitim setindeki cevapların ortalaması ilgili bölgenin tahmin değerini oluşturmaktadır (Gareth ve ark., 2013).

Her bir bölge oluşturulmasında aşağıdaki 3 soruya cevap aranmaktadır.

1. Bölme hangi değişkene göre ve hangi noktadan yapılmalıdır?
2. En iyi bölme noktası ve değişkeni nasıl tanımlanır?
3. Bölme işlemini durdurma kararı nasıl verilir?

Bölgeler, Regresyon ağaçlarında RSS'nin minimize edilmesi ile belirlenmektedir (Gareth ve ark., 2013).

Ancak bu yöntem ile bölgelere ayırmak pek mümkün olmadığından özyinelemeli ikili bölünme olarak bilinen aşağıdan yukarı doğru bölme işleminin gerçekleştirildiği greedy yaklaşımı kullanılmaktadır. Özyinelemeli ikili bölünme yöntemi yukarıdan aşağı doğrudur çünkü ağacın tepesinden başlar ve bu nokta da tüm gözlemler aynı bölgede yer almaktadır. Daha sonra art arda tahmin uzayı bölünmekte ve her bölünme ağaç üzerinde aşağı doğru iki dala ayrılmaktadır. Greedy yaklaşımında tüm adımlar bütün olarak ele alınıp en iyi tahmin sonucunu verecek bölünme noktalarını belirlemek yerine bulunan adımın en iyi sonucuna göre bölme işlemi gerçekleştirilmektedir.

Tüm tahmin ediciler $X_1, X_2, \dots, X_p$ ve tahmin edicilerin olası kesim noktalarından, ağaçta en düşük RSS değerine sahip tahmin edici ve kesim noktasından ikili bölme işlemi gerçekleştirilmektedir (Gareth ve ark., 2013). Bu

bölme işlemi bölgeler tamamlanana kadar tekrarlanmaktadır.

$$R_1(j, s) = \{X|X_j < s\} \text{ ve } R_2(j, s) = \{X|X_j \leq s\}$$

j tahmin edicisi ve s kesim noktası,

$$\sum_{i:x_i \in R_1(j,s)} (y_i - \hat{y}_{R_1})^2 + \sum_{i:x_i \in R_2(j,s)} (y_i - \hat{y}_{R_2})^2$$

ifadesinin minimize edilmesi ile belirlenmektedir (Gareth ve ark., 2013).

$\hat{y}_{R_1}$, eğitim setinde yer alan gözlemlere $R_1(j, s)$ bölgesinde verilen ortalama cevaplar

$\hat{y}_{R_2}$, eğitim setinde yer alan gözlemlere $R_2(j, s)$ bölgesinde verilen ortalama cevaplar

Ağacın, yaprak düğümleri en düşük kirliliğe ulaşana kadar büyütülmeye devam edilirse eğitim seti ile aşırı uyum sorunu oluşmaktadır. Eğer bölünme işlemi çok erken durdurulursa, eğitim verilerindeki hata yeterince düşük olmaz ve bu durum tahmin edicinin performansının düşük olmasına sebep olmaktadır.

Karar ağacı algoritmalarının stratejisi ağacı oluşturmak ve sonra budama işlemi yapmaktır. Bunun nedeni karmaşık bir ağaç düşük test performansına sebep olabilmektedir. Karmaşık bir ağaç yerine daha düşük varyanslı ve biraz yanlış daha küçük bir ağaç ile ilerlemek yorumlamamızı kolaylaştırabilmektedir. Budama işlemini yapmamız için bölünmenin değerli olup olmadığını anlamamız gerekmektedir. Bölünmenin değerliliği, budama işlemi yapıldıktan sonraki RSS ile budamadan önceki RSS arasında oluşan farka göre belirlenebilmektedir. Ancak bu yöntemin sıkıntısı anlamsız bir bölünmenin ardından anlamlı bir bölünme gerçekleşmiş olabilmesidir. Bizim hedefimiz büyük ağacın içerisinde budama ile en düşük test hatasına sahip ağacı belirlemektir. Bunun için çapraz doğrulama yöntemi ile büyük ağaçtan alınan alt ağaçların test hata oranları karşılaştırılarak en iyi ağacı belirlememize yardımcı olacak

en zayıf halka budaması olarak bilinen Maliyet Karmaşıklığı Budama yöntemi kullanılmaktadır. Bu yöntemde her alt ağacı düşünmek yerine negatif olmayan ağaç dizininden oluşan ayar parametresi olan α kullanılacaktır.

T_0 oluşturulan büyük ağaç olup, her α değeri bir $T \subset T_0$ alt ağacına karşılık gelmekte olup

$$\sum_{m=1}^T \sum_{i: x_i \in R_m} (y_i - \hat{y}_{R_m})^2 - a|T|$$

ifadesinin olası en küçük değeri aranmaktadır (Gareth ve ark., 2013).

$|T|$, T ağacının düğüm sayısını göstermektedir.

R_m , m . düğüme karşılık gelen tahmin uzayıdır.

$\hat{y}_{R_m}$, R_m 'deki eğitim gözlemlerinin ortalamasıdır.

2.1.2.2 Sınıflandırma Ağaçları

Sınıflandırma ağaçlarında regresyon ağaçları ile aynı akış takip edilmekte olup RSS yerine daha uygun hata ölçüm yöntemleri kullanılmaktadır. Sınıflandırma ağaçlarında kullanılabilecek saflık ölçümleri aşağıda yer almaktadır.

$\hat{p}_k$, R bölgesindeki k sınıfında yer alan gözlemlerin yüzdesini göstermektedir (Chen,2013).

1. Sınıflandırma Hata Oranı (R bölgesi için)

$$E = 1 - \max_k \hat{p}_k$$

2. Gini İndeksi

$$G = \sum_{k=1}^K \hat{p}_k (1 - \hat{p}_k)$$

3. Çapraz Entropi

$$D = - \sum_{k=1}^K \hat{p}_k \log(\hat{p}_k)$$

Eğer R bölgesi saflığa yakınsa ve gözlemlerin çoğu bir sınıftan geliyorsa Gini İndeksi ve Çapraz Entropi yöntemlerinin hata değerleri, Sınıflandırma Hata Oranı yöntemindeki hatadan daha düşük olmaktadır. Gini İndeksi ve Çapraz Entropi yöntemleri düğüm saflığına daha duyarlı olduğundan bölünmenin kalitesini ölçmek için yaygın olarak kullanılmaktadırlar. Bu üç yöntemden biri ağacın budanmasında da kullanılabilmektedir. Eğer budanmış ağacın tahminlerinin doğruluğu kontrol edilecekse Sınıflandırma Hata Oranı yöntemi tercih edilmektedir (Chen, 2013).

2.1.3. Topluluk Yöntemleri (Ensemble Methods)

Topluluk yöntemlerinde, eğitim verisinde örneklemeler alınarak temel öğrenme olarak adlandırılan öğrenme grupları oluşturulmakta ve daha sonra bu gruplar birleştirilerek nihai öğrenmeye ulaşılmaktadır. Bu yöntemin amacı zayıf olan öğrenmeleri güçlü hale getirmektir. Bu nedenle zayıf öğrenmelerden oluşan nihai öğrenmenin güçlü olması beklenmektedir. Genellikle topluluk yöntemlerinde homojen temel öğrenme grupları oluşturulmak istenildiğinden tek bir öğrenme algoritması kullanılmaktadır.

2.1.3.1 Bagging

Bagging, eğitim setindeki verilerden yerine koyma yöntemi ile örneklemeler alarak tek bir öğrenme yöntemi ile alınan örneklemelerle öğrenme işlemi gerçekleştiren bir topluluk modelidir. Örneklem alma ve öğrenme işlemi tekrarlanmakta ve her bir tekrarlama işlemi bir sınıflandırıcı ile sonuçlanmaktadır. Oluşturulan sınıflandırıcıların tahminleri dikkate alınarak nihai tahmin edici oluşturulmaktadır (Maglogiannis ve ark.,2007).

$\hat{f}$, eğitim setindeki verilere $D = (y_1, x_1), \dots, (y_n, x_n)$ dayanılarak elde edilen tahmin edici olmaktadır. Eğitim setinden bootstrap yöntemi ile alınan bir örneklem D_1 ve bu örneklemde elde edilen tahmin edici $\hat{f}_1$ olarak göste-

ilmektedir. β kez bu işlemin gerçekleştirilerek $\hat{f}_1, \dots, \hat{f}_\beta$ tahmin edicileri elde edilmektedir. Elde edilen tüm tahmin ediciler sonucunda ulaşılan nihai tahmin ediciye

$$\hat{f}_{bagging}(x) = \frac{1}{\beta} \sum_{j=1}^{\beta} \hat{f}_j^*(x)$$

eşitliği ile ulaşılmaktadır (Chen, 2013).

1 veya -1 gibi iki sınıflı sınıflandırma durumlarında her bir $\hat{f}_j$ 1 veya -1 değerini almaktadır. Gözlemler $\hat{f}_j, j = 1, \dots, \beta$ tahmin edicilerinden en çok hangi sınıf tahminini alırsa o sınıfa dahil edilmektedir. Test tahmin hatası

$$\hat{f}_j^*(x_j) = \frac{\sum_{j=1}^{\beta} \hat{f}_j^*(x) I((y_i, x_i) \notin D_j^*)}{\sum_{j=1}^{\beta} I((y_i, x_i) \notin D_j^*)}$$

eşitliği ile yapılmaktadır (Chen, 2013).

(y_i, x_i) , i. data noktalarını içermeyen örneklemelere ait tahmin edicilerin ortalaması olmaktadır. OBB (out-of-bag) ortalama kare hatası,

$$\sum_{i=1}^n (y_i - \hat{f}^*(x_i))^2$$

olarak ifade edilmekte ve bu test ortalama kare hatasının tahmin edicisi olmaktadır (Chen, 2013).

OBB sınıflarından en yüksek oyu alan sınıf $\hat{f}^*(X_i)$ 'dir ve OBB sınıflandırma hatası,

$$\frac{1}{n} \sum_{i=1}^n (y_i \neq \hat{f}^*(x_i))$$

olarak ifade edilmektedir. Bu da sınıflandırma hatasının tahmin edicisi olmaktadır (Chen, 2013).

2.1.3.2 Boosting-Ada Boost

Bu yöntem Bagging yöntemiyle benzer olmakla beraber öğrenme algoritmasının performansını dikkate alarak doğru öğrenilmemiş datalar üzerine yo-

ğunlaşmaktadır. Başlangıçta seçilen rastgele örneklemden oluşturulan ilk sınıflandırıcı ile yanlış tahmin edilen gözlemlere, bir sonraki örneklem seçimlerinde öncelik verilmektedir. Her adımda, mevcut modelden tahmin edilen değerler ile gerçek değerler arasındaki fark olan artıkları veya uygun kayıp fonksiyonun negatif gradyanlarını tahmin etmek amaçlanmaktadır (Maglogiannis ve ark.,2007).

$\hat{f}$, eğitim setindeki verilere $D = (y_i, x_i), i = 1, \dots, n$ dayanılarak elde edilen tahmin edici olmaktadır. Başlangıçta $\hat{f} = 0, r_i = y_i$ olmakta ve aşağıdaki algoritma yinelemeli olarak takip edilmektedir (Chen, 2013).

- $(x_i, y_i), i = 1, \dots, n$ verilerine istinaden elde edilen tahmin edici $\hat{g}$
- $\hat{f} = \hat{f} + \lambda \hat{g}$ olarak revize edilir
- $r_i = r_i - \lambda \hat{g}(x_i)$ olarak revize edilir.

Bu işlem yineleme duruncaya kadar devam ettirilir ve $\hat{f}$ tahmin edicileri oluşturulur. λ öğrenme oranı olmaktadır. Öğrenme yönteminin kayıp fonksiyonu olan L 'yi minimize etmek için Eşitlik (2.1) kullanılmaktadır. Başlangıçta $\hat{f}_0(x) = \gamma$ ve $\gamma, \sum_{i=1}^n L(y_i, \gamma)$ 'yi minimize eden bir sabit olmaktadır (Chen, 2013).

$$\begin{aligned} \hat{f}_k(x) &= \hat{f}_{k-1}(x) + \lambda \hat{g}(x), \quad k = 1, \dots, K \\ (\hat{\lambda}, \hat{g}(x)) &= \arg \min_{g \in G, \lambda} \sum_{i=1}^n L(y_i, \hat{f}_{k-1}(x) + \lambda g(x)) \end{aligned} \quad (2.1)$$

G , algoritmadan oluşturulan öğrenme topluluklarının bütününü ifade etmektedir.

Boosting algoritmalarında; küçük öğrenme oranının kullanılması, aşırı uyumu önlemek için erken durma ve rastgele örneklemeler seçerek temel öğrenme sınıfları oluşturma olarak önemli üç unsur vardır. Küçük öğrenme oranının kullanılması yinelemeyi daha kararlı hale getirmektedir. Yinelemelerin hangi aş-

mada durdurulacağı ise test hatasının doğrulama veya çapraz doğrulama yoluyla kontrol edilmesi ile belirlenmektedir. Rastgele örneklem ile oluşturulan sınıflar arasındaki korelasyonu düşürmek amaçlanmaktadır (Chen, 2013).

Boosting algoritmasının özel bir durumu olan AdaBoost algoritması ikili sınıflandırmaya göre çalışmakta olup örneklem seçiminde, bir önceki sınıfta yanlış tahmin edilen değerlere bir sonraki örneklem seçiminde öncelik verilmektedir (Jong ve Heller, 2008).

2.1.3.3 Rastgele Orman (Random Forests)

Rastgele Orman, bagging yoluyla bağımsız ve aynı dağılımlı rastgele vektör örneklemelerinden elde edilen tahmin edici ağaçların bütününden oluşmaktadır. Rastgele Orman oluşturma adımları aşağıda yer almaktadır (Bremian, 2001)

- $k = 1, \dots, \beta$ için bootstrap yöntemi ile $D_k^* = \{(x_1^*, y_1^*), \dots, (x_n^*, y_n^*)\}$ örnekleme seçilmektedir.
- Seçilen her bir D_k^* örnekleme için T_k ağacı oluşturulmaktadır. Ağaç oluşturulurken her bölünmeden önce p değişkenden rastgele m değişken seçilmekte ve bu m değişken arasından en iyi değişken ve kesme noktası seçilerek bölme işlemi gerçekleştirilmektedir.
- Nihai tahmin edici $\hat{f}(x) = \frac{1}{n} \sum_{k=1}^{\beta} T_k^*$ olmaktadır.

Sınıflandırma problemi için her x girdisinin rastgele ormandaki her T_k ağacında bir sınıf tahmini oluşmakta ve T_k sınıf tahminlerinden en çok oyu alan sınıf x 'in nihai tahmin edicisi olmaktadır. Bootstrap yoluyla alt küme oluşturulmasının sebebi daha az korelasyona sahip ağaçlar oluşturularak bagging yönteminin varyans azaltmada daha etkili olmasını sağlamaktır. Yaygın olarak m değişken seçimi $\sqrt{p}$ eşit veya küçük olarak yapılmakta olup minimum 1 olabilmektedir (Chen, 2013).

2.1.4. Destek Vektör Makineleri (Support Vektor Machines) Yöntemi

SVM regresyon uygulaması olmasına rağmen yaygın olarak sınıflandırma problemlerinde kullanılmaktadır. Bu yöntemle doğrusal olan bir sınıflandırıcı doğrusal olmayan karar sınırları üreten bir yapıya dönüşmektedir.

p boyutlu bir vektörü $b = (b_1, \dots, b_p)$ düşünelim, tüm p vektörleri $x = (x_1, \dots, x_p)$, z 'nin b 'ye dik olduğu yerlerde $ab + z$ olarak ifade edilmektedir (Chen, 2013).

Doğrusal fonksiyon,

$$g(x) = \langle b, x \rangle = b_1x_1 + \dots + b_px_p$$

$g(x) = 0$ iken $p-1$ boyutlu orjin boyunca ve vektöre $b = (b_1, \dots, b_p)^T$ dik olan A_0 ile gösterilen bir hiperdüzlem tanımlanmaktadır. $g(x) = c$ 'yi sağlayan tüm x noktaları $p-1$ boyutlu $A_c = cb/||b||^2 + A_0$ olarak ifade edilen bir hiperdüzlem oluşturmaktadır. $g(x) - c > 0$ 'yi sağlayan x noktaları b yönü boyunca A_c 'nin bir tarafında, $g(x) - c < 0$ 'ı sağlayanlar ise diğer tarafında olmaktadır. $(g(x) - c)/||b||$, x ve A_c arasındaki mesafeyi göstermektedir (Chen, 2013).

$D = \{(y_i, x_i), i = 1, \dots, n\}$ verilerinin $y_i = \pm 1$ olan bir hiperdüzlemle ayrıldığı durumu düşünelim

$$f(x) = \beta_0 + \beta_1x_1 + \dots + \beta_px_p,$$

1 sınıfında olan tüm i noktalarında $f(x) > 0$ olup bu i noktaları hiperdüzlemin bir tarafında, -1 sınıfında olan tüm i noktalarında ise $f(x) < 0$ olup bu i noktaları ise hiperdüzlemin diğer tarafında yer almaktadır. Bu durum,

$$y_if(x_i) > 0, \quad i = 1, \dots, n.$$

eşitsizliği ile ifade edilebilmektedir (Chen, 2013).

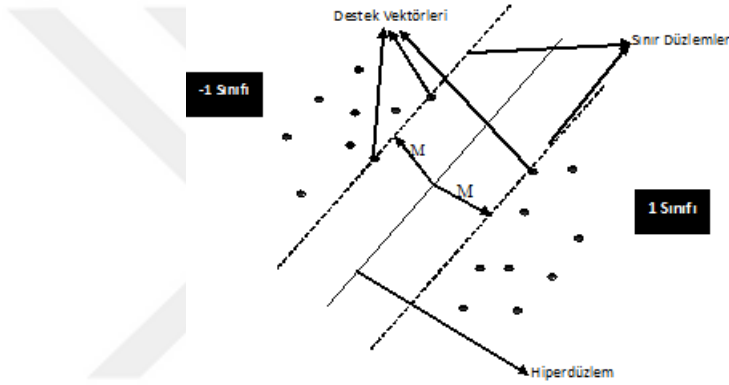
Tüm noktaların ayırma hiperdüzlemine uzaklığının minimum olduğu, maksimi-

mum marjin sınıflandırıcılı bir hiperdüzlem bulmaya çalışmaktadır. $\max_{\beta_0, \beta_1, \dots, \beta_p} M$ koşuluyla

$$\sum_{i=1}^p \beta_i = 1$$

tüm i değerleri için $y_i(\beta_0 + \beta_1 x_{i1} + \dots + \beta_p x_{ip}) \geq M$

M veri noktalarını iki sınıfa ayıran şeridin yarısıdır ve şerit $|f(x)| < M$ olarak tanımlanmaktadır. Sınır düzlemlerinin üzerinde veya sınırında olan noktalar destek vektörleri olarak adlandırılmakta olup Şekil 2.2 destek vektörlerinin yapısı gösterilmektedir (Chen, 2013).



Şekil 2.2: Maksimum Marjin Destek Vektörleri

Genellikle iki sınıfın hiperdüzlemle ayrıldığı durumlarda maksimum marjin yüksek varyanslı olduğundan ve buna bağlı olarak aşırı uyum oluşmasından dolayı iki sınıfın ayrılması durumların da bu yöntem çok tercih edilmemektedir. Maksimum marjin sınıflandırıcısı ile sınıfların ayrılamadığı durumlarda, maksimum marjin yerine soft marjin yöntemi kullanılmaktadır. Bu yöntem bireysel gözlemlerde daha fazla sağlamlık ve eğitim gözlemlerinde daha iyi sınıflandırma sağlamaktadır. Soft marjin yönteminde sınır düzlemlerinin yanlış tarafında veya bazen hiperdüzlemin yanlış tarafında olan noktalarla karşılaşmak mümkündür. Bu durumu ortadan kaldırmak için aşağıdaki süreç izlenmektedir (Chen, 2013).

$\max_{\beta_0, \beta_1, \dots, \beta_p} M$ koşuluyla

$$\sum_{j=1}^p \beta_j^2 = 1,$$

Tüm i değerleri için $y_i(\beta_0 + \beta_1 x_{i1} + \dots + \beta_p x_{ip}) \geq M(1 - \epsilon_i)$, $\epsilon_i > 0$,

$$\sum_{i=1}^n \epsilon_i \leq C$$

C negatif olmayan ayarlama parametresi, ϵ_i ise dolgu değişken olmaktadır. Sınıflandırma kuralına istinaden x gözleminin sınıfı $f(x) > 0$ ise 1, değilse -1 olmaktadır.

$\epsilon_i = 0 \Leftrightarrow i$. gözlem sınırın doğru tarafında, $\epsilon_i > 0 \Leftrightarrow i$. gözlem sınırın yanlış tarafında, $\epsilon_i > 1 \Leftrightarrow i$. gözlem hiper düzlemin yanlış tarafında olmaktadır. C ayar parametresi n gözlemden sınırı ihlal edenleri göstermektedir. $C = 0 \Leftrightarrow$ tüm i gözlemleri için sonuç olmaktadır (Chen, 2013).

Doğrudan sınırın üzerinde olan veya sınırın yanlış tarafında olan gözlemler destek vektörleri olarak adlandırılmaktadır. C büyüdükçe ihlal artmakta buna bağlı olarak destek vektörü de artmaktadır. Destek vektörü arttıkça varyans küçülmekte ve sınıflandırma daha sağlam olmaktadır.

Doğrusal destek vektör sınıfı,

$$f(x) = \beta_0 + \sum_{i=1}^n \alpha_i < x_i, x >$$

eşitliği ile ifade edilmektedir.

$\alpha_i, < x_i, x >$ göre hesaplanmaktadır ve tüm destek vektörleri için $\alpha_i \neq 0$ olmaktadır. Doğrusal destek vektör makine sınıflandırıcısının hesaplanmasında yalnızca özellik uzayının iç çarpımı dikkate alınır ve doğrusal bir sınırı vardır. Ancak pratikte doğrusal bir sınır ile ayıramayacağımız durumlar mevcuttur. Bu durumda özellik uzayının genişletilmesini sağlayan çekirdek numarası

(kernel trick) yaklaşımı kullanılmaktadır. Destek vektör makinesinin mevcut özellik uzayı çekirdek fonksiyon uzayına genişletilir (Chen, 2013).

$$x_i \rightarrow K(., x_i)$$

Negatif olmayan kesinlik (nonnegative definiteness) özelliğini sağlayan iki değişkenli çekirdek fonksiyonu,

$$\sum_{i,j} \alpha_i \alpha_j K(x_i, x_j) \geq 0$$

eşitsizliği p boyutlu özellik uzayını, sonsuz boyutta olan çekirdek fonksiyon uzayına genişletmektedir. Destek vektör makine sınıflandırıcısı için eğitim verisindeki tüm x_i, x_j $K(x_i, x_j)$ 'ler için değeri hesaplanmaktadır.

Destek vektör makine sınıflandırıcısı fonksiyonu,

$$f(x) = \beta_0 + \sum_{i=1}^n \alpha_i K(x_i, x_j)$$

eşitliği ile gösterilmektedir (Chen, 2013).

Yaygın olarak kullanılan çekirdek fonksiyonları (Chen, 2013)

- Doğrusal çekirdeği, $K(x_i, x_j) = \langle x_i, x_j \rangle = x_i^T x_j$
- d derece polinomal çekirdeği, $K(x_i, x_j) = (1 + \langle x_i, x_j \rangle)^d$
- Gauss radyal çekirdeği, $K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$, $\gamma > 0$

2.1.5. Naive Bayes Yöntemi

Bağımsızlık varsayımı ile Bayes Teorimi'ne dayalı sınıflandırma yapılmasını sağlayan olasılık tabanlı bir makine öğrenmesi yöntemidir. Bayes sınıflandırmasının koşullu bağımsızlık yaklaşımıyla ele alındığı Naive Bayes yönteminde sınıftaki herhangi bir özelliğin varlığının diğer özelliklerin varlığına bağlı olmadığı varsayılmaktadır. x gözlemleri ve x gözlemlerine karşılık ge-

len C sınıflarından oluşan bileşik model formu

$$p(x, c) = p(c) \prod_{i=1}^D p(x_i|c)$$

eşitliği ile ifade edilmektedir (Barber,2012).

D , özellik sayısı olup x_i , C sınıfında yer alan bağımsız özellik vektörü olmaktadır. Datanın aynı dağılımlı ve birbirinden bağımsız özelliklerden oluştuğu varsayılarak Maksimum Olabilirlik yöntemi ile $p(c)$ ve $p(x_i|c)$ parametreleri belirlenmektedir. Tüm parametre tahminleri tamamlandıktan sonra x^* özellik vektörüne ait sınıf tahmini,

$$p(c|x_*) = \frac{p(x^*|c)p(c)}{p(x^*)} = \frac{p(x^*|c)p(c)}{\sum_c p(x^*|c)p(c)}$$

eşitliği ile yapılmaktadır (Barber,2012).

İki özelliğe bağlı sınıflandırılma yapılması durumunda, $\{x^n, n = 1, \dots, N\}$ veri kümesi için $x_i^n \in \{0, 1\}, i = 1, \dots, D$ olup her x^n veri noktası bir c^* sınıfına ait olmaktadır. Veri noktalarının $c = 0$ olması durumu n_0 , $c = 1$ olması durumu n_1 ile gösterilmektedir. İki sınıftaki her bir özellik için $p(x_i = 1|c) = \theta_i^c$ olasılığının tahmin edilmesi gerekmektedir. İki sınıf olduğundan $p(x_i = 0|c) = \theta_i^c = 1 - p(x_i = 1|c) = \theta_i^c = 1 - \theta_i^c$ olarak ifade edilebilmektedir. Naive Bayes koşullu bağımsızlık varsayımına dayanarak, bir x vektörünün gözlemlenme olasılığı,

$$p(x|c) = \prod_{i=1}^D p(x_i|c) = \prod_{i=1}^D (\theta_i^c)^{x_i} (1 - \theta_i^c)^{1-x_i}$$

eşitliği ile hesaplanmaktadır (Barber,2012).

Eğitim verisinin aynı dağılımlı ve bağımsız olduğu varsayımıyla sınıf ve öze-

liklerin log-olabilirliği Eşitlik (2.2)'de yer almaktadır.

$$\begin{aligned}
 L &= \sum_n \log p(x^n, c^n) = \sum_n p(c^n) \prod_i p(x_i, c^n) \\
 &= \sum_{i,n} x_i^n \log \theta_i^{c^n} + (1 - x_i^n) \log(1 - \theta_i^{c^n}) + n_0 \log p(c = 0) + n_1 \log p(c = 1)
 \end{aligned}
 \tag{2.2}$$

Eşitlik (1.2) $p(c)$ ve $\theta_i^{c^n}$ 'ye göre düzenlenerek parametrelerin maksimum olabilirlik değerleri elde edilebilmektedir (Barber,2012).

$$\begin{aligned}
 \theta_i^{c^n} &= p(x_i|c) = \frac{c \text{ sınıfındaki } x_i = 1 \text{ sayısı}}{c \text{ sınıfındaki toplam gözlem sayısı}} \\
 p(c) &= \frac{c \text{ sınıf olanların sayısı}}{\text{toplam gözlem sayısı}}
 \end{aligned}$$

Bir x^* vektörünün sınıfının 1 olabilmesi için $p(c = 1|x^*) > p(c = |x^*)$ eşitsizliğinin sağlanması gerekmektedir. Bu eşitsizliğin açık hali,

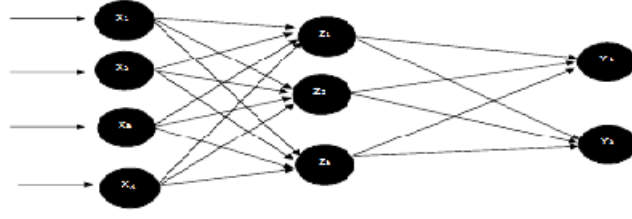
$$\begin{aligned}
 &\sum_i \{x_i^* \log \theta_i^1 + (1 - x_i^*) \log(1 - \theta_i^1)\} + \log p(c = 1) \\
 &\sum_i \{x_i^* \log \theta_i^0 + (1 - x_i^*) \log(1 - \theta_i^0)\} + \log p(c = 0)
 \end{aligned}$$

olmaktadır (Barber,2012).

2.1.6. Sinir Ağları (Neural Network) Yöntemi

Son yıllarda derin öğrenmedeki performansı ile en başarılı makine öğrenim yöntemlerinden biri olan sinir ağları, X girdilerine karşılık gelen Y çıktısına doğru bilgi akışı sağlayan çok katmanlı hiyerarşik bir sistemden oluşmaktadır. Bir önceki katman çıktısının doğrusal kombinasyonları doğrusal olmayan bir fonksiyon ile işlenmektedir. Şekil 2.3'de üç katmanlı sinir ağı yapısı yer almaktadır. Katmanlar soldan sağa doğru olarak giriş, gizli ve çıkış olarak adlandırılmaktadır (Chen, 2013).

Geri yayılım olarak adlandırılan optimizasyon algoritması ile eğitim datasın-



Şekil 2.3: Katmanlı Sinir Ağı Yapısı

dan öğrenme geriye doğru akışla yapılmakta olup çıktı tahminlerinde ileri doğru bilgi akışı yapılmaktadır. Tek gizli katmandan oluşan “vanilya” sinir ağında geri yayılım ile öğrenme incelenecektir.

Eğitim setindeki girdiler ve çıktılar $(x_i, y_i); i = 1, \dots, n$ olup girdiler $x_i = (1, x_{i1}, \dots, x_{ip})$ olarak gösterilmektedir. σ aktivasyon fonksiyonu ve gizli katman $m + 1$ boyutlu $h_i = (1, h_{i1})$ vektörü,

$$h_{ij} = \sigma(w_j^T, x_i); \quad j = 1, \dots, m$$

eşitliğinden elde edilmektedir. m gizli katmandaki birimleri ifade etmektedir (Chen, 2013).

$w_j, mx(p \pm 1)$ olan W matrisinin j . satırı ve $\hat{y}_i = g(\beta^T, h_i)$ aktivasyon fonksiyonu olmaktadır. $\theta(\beta, W)$ parametrelerinden oluşup $m + 1 + m(p + 1)$ boyutludur.

$$\sum_{i=1}^n (y_i - \hat{y}_i)^2 = \sum_{i=1}^n (y_i - g(\beta^T h_i))^2 = \sum_{i=1}^n R_i(\theta)$$

eşitliğindeki hata karelerinin en küçük değeri dikkate alınmaktadır (Chen, 2013).

Geri yayımlı olarak hataların yayılması çıktı katmanından giriş katmanına doğru olmaktadır. Gradyan temelli kayıp fonksiyonun küçültülmesi geriye doğru her katmanda kayıp fonksiyonun türevlenmesi ile yapılmakta olup süreç aşağıda yer almaktadır (Chen, 2013).

Çıkış Katmanı;

$$\frac{\partial R_i(\theta)}{\partial \beta} = -2(y_i - \hat{y}_i)g(\beta^T h_i)h_i = \delta_i h_i$$

Gizli Katman;

$$\begin{aligned}\frac{\partial R_i(\theta)}{\partial \beta} &= -2(y_i - \hat{y}_i)g(\beta^T h_i)\beta_j \frac{\partial h_{ij}}{\partial w_j} \\ &= -2(y_i - \hat{y}_i)g(\beta^T h_i)\beta_j \sigma(w_j^T x_i)x_i = s_{ji}x_i\end{aligned}$$

δ_i, s_{ji} mevcut modeldeki çıktı ve gizli katman birimlerindeki hatalar olup,

$$s_{ji} = \sigma(w_j^T x_i)\delta_i\beta_j$$

eşitliğinden elde edilmektedir (Chen, 2013).

Seçilen öğrenme γ oranı ile parametrelerin güncellenmesi,

$$\begin{aligned}\beta^{yeni} &= \beta - \gamma \sum_{i=1}^n \frac{\partial R_i(\theta)}{\partial \beta} = \beta_j - \gamma \sum_{i=1}^n \delta_i h_i \\ w_j^{yeni} &= w_j - \gamma \sum_{i=1}^n \frac{\partial R_i(\theta)}{\partial w_j} = w_j - \gamma \sum_i s_{ji} x_i\end{aligned}$$

eşitlikleri ile yapılmaktadır (Chen, 2013).

Geriye doğru yayılma algoritması iki aşamalı bir sisteme sahiptir. İlk aşamada γ tahminlerini hesaplamak için iler doğru bir akış gerçekleştirilir. İkinci aşamada çıkış katmanındaki δ_i hatası hesaplanır ve bir sonraki katmandaki s_{ji} hatası hesaplanarak bu işlem geriye doğru akış ile katmanlar sonlanana kadar sürdürülür. Hesaplanan tüm hatalar uygun öğrenme oranı γ ile parametre güncellemesinde kullanılır. Parametre güncelleme işlemini tüm eğitim verilerine uygulamak yerine eğitim verilerinden alınan bir alt örnekleme uygulayabiliriz. Bu durumda yaygın olarak kullanılan algoritma SGD (Stochastic Gradient Descent)'dir (Chen, 2013).

2.2. Lojistik Regresyon

Lojistik regresyon analizi bir veya birden fazla bağımsız değişken ile kategorik olan bağımlı değişken arasındaki ilişkiyi modellemek için kullanılmaktadır. Regresyon modellemesinde kategorik değişkenler açıklayıcı değişkenler olarak kullanılırken, lojistik regresyonda kategorik değişkeni tahmin etmek için modelleme yapılmaktadır.

Modellenecek bağımlı değişkenin hasarlı/hasarsız, ölü/hayatta gibi iki kategoriye sahip olduğu durumlarda hasar/ölüm gibi olayların gerçekleştiği durumlar 1, diğer durum ise 0 ile gösterilmektedir. π , $y = 1$ olmasının olasılığı olup $y \sim B(1, \pi)$, $g(\pi) = x'\beta$ olmaktadır. Lineer regresyon modelinde $E(Y|x) = \alpha + \beta x$ olmakta ve koşullu beklenen değer $-\infty$ ve ∞ arasında değerler alabilmektedir. Ancak bizim modellemek istediğimiz bağımlı değişken 0 ve 1 değerini almaktadır. Bu durumda odds olarak adlandırılan olayın gerçekleşme olasılığının gerçekleşmeme olasılığına bölümü olan $\pi/(1 - \pi)$ dönüşümü ile bağımlı değişken sınırları $(0, \infty)$ olarak genişletilir. Bağımlı değişken sınırlarını $(-\infty, \infty)$ olarak genişletmek içinde log dönüşümü uygulanmış olup dönüşümler

$$g(\mu) = \ln \frac{\pi}{1 - \pi} = x'\beta \Rightarrow \pi = \frac{e^{x'\beta}}{1 + e^{x'\beta}}$$

olmaktadır.

Tek değişkene bağlı olarak modellenecek bağımlı değişken için regresyon eşitliği,

$$\ln \frac{\pi}{1 - \pi} = \alpha + \beta x$$

birden fazla (r tane) değişkene bağlı olarak modellenecek bağımlı değişken için regresyon eşitliği

$$\ln \frac{\pi}{1 - \pi} = \beta_0 + \beta_1 x_1 + \dots + \beta_{r-1} x_{r-1}$$

olmaktadır (Jong ve Heller,2008).

3. UYGULAMA

Çalışmanın bu bölümünde sektörden alınan Kasko Sigortası verilerinden elde edilen dağılım ve parametreler ile simüle edilen veri kullanılmıştır. Simülasyon ile oluşturulan değişkenler, kullanılan dağılım ve parametreler Çizelge 3.1’de yer almaktadır.

DEĞİŞKENLER	ORTALAMA	VARYANS	DAĞILIM	PARAMETRE
UYG._KADEME_NO	1.7492	2.69881	N. Binom	$n=3945210, p=0.99999$
PLAKA_IL_KODU	30.5442	2,801.84	N. Binom	$n=3945210, p=0.9999$
TOP_HASAR_SAYISI	1.2329	0.2915	Binom	$n=3945210, p=3.1249E-07$
POLBASYIL	2,008.38	2.01145	Binom	$n=3945210, p=0.0005$
TOP_SIGORTALIYIL	3.0584	1.01584	Binom	$n=3945210, p=7.7522-07$
MODEL_YILI	2,004.29	11.78369	Binom	$n=3945210, p=0.0005$
EXPOSURE	0.5	0.07952	Beta	$\alpha=1.072, \beta=1.072$
TOP_HASAR_MEBL	2,013.38	18,753,152.40	Lognormal	$\mu=6.9207, \sigma=1.172$

Çizelge 3.1: Simülasyon Değişken, Dağılım ve Parametreleri

Kasko, herhangi bir kaza anında karşı tarafın canına veya malına gelebilecek zararların teminat altına alınması amacıyla Trafik Kanunu gereğince yaptırılması zorunlu olan Kara yolları motorlu araçlar zorunlu mali sorumluluk (Trafik) sigortasının aksine kişinin kendi aracına gelebilecek zararları teminat altına alan özel bir sigortadır.

3.1. Verinin Niteliği

Veri seti 2006-2010 yılları Kasko sigortasında oluşmakta olup simülasyonla elde edilmiştir. Veri setindeki değişkenler ve veri setindeki değişkenlerden türetilen değişkenler Çizelge 3.2’de yer almaktadır.

Poliçenin sorumluluğu başlangıç ve bitiş tarihi süresince devam ettiğinden hasar gelme riski de sigortacı için bu süre boyunca devam etmektedir. Bu nedenle poliçenin başlangıç yılı ve bitiş yılları arasındaki riske maruz değerleri bizim için önemli olmaktadır. Riske maruz değer olarak adlandırılan exposure poliçe başlangıç ve bitiş yıllarına göre dönemlere ayrılarak Eşitlik

Veri Seti Değişkenleri	Türetilen Değişkenler
Kii ID	ARACYAS
MODEL_YILI	HASAR_VAR/YOK
PLAKA_IL_KODU	SIGORTALIYIL
UYGULANAN_KADEME_NO	
TOP_SIGORTALIYIL	
POLBASYIL	
POLBITYIL	
EXPOSURE	
HASAR_DONEM(EXPOSURE_DONEM)	
TOP_HASAR_MEBLAG	
TOP_HASAR_SAYISI	

Çizelge 3.2: Veri Seti ve Türetilen Değişkenler

(3.1) ile hesaplanmaktadır. Yıllık poliçeleri dikkate aldığımızdan exposure dönemi 0 ve 1 değerlerinden oluşmaktadır. Hasarda söz konusu exposure dönemlerinde oluşabileceğinden hasar dönem ve exposure dönemi aynı olarak düşünülmüştür.

$$\begin{aligned}
 \text{exposure 0.dönem} &= \frac{\text{Poliçeni Bitiş Yılıının İlk Gün Tarihi-Poliçe Başlangıç Tarihi}}{365} \\
 \text{exposure 1.dönem} &= \frac{\text{Poliçeni Bitiş Tarihi-Poliçe Bitiş Yılıının İlk Gün Tarihi}}{365}
 \end{aligned}
 \tag{3.1}$$

Araç yaş, poliçe başlangıç yılından araç model yılının çıkartılması ile elde edilmiştir.

Sigortalı yıl, ilgili kişinin kasko yaptırılan yıl sayısını göstermektedir. Örneğin sisteme 2006 yılında ilk defa girmiş olan bir müşteri için bu değer 1 olmakta ancak 2005'te sigorta yaptırmış olması durumunda bu değer 2 olmaktadır. Toplam sigortalı yıl, 2006-2010 döneminde kişinin kasko yaptırdığı toplam yıl sayısını göstermektedir. Örneğin her yıl poliçe yaptırmış biri için bu değer 5, herhangi yıllardan birinde poliçe yaptırmamış kişi için 4 olmaktadır. Her yıl poliçe yaptırmış kişi için 2010 yılında sigortalı yıl ile toplam sigortalı yıl değeri eşit olmaktadır.

Hasar olup olmadığı durumu, toplam hasar meblağı sıfırdan büyük olanlar için “1”, sıfır veya boş olduğu durumlar “0” olarak düzenlenmiştir.

Hasarlı- hasarsız sınıf tahmin modellemesinde Orange programı kullanılmış olup maksimum 500.000 veri kapasitesi sebebi ile kümül veriden örneklem alınarak modelleme yapılmıştır. 2006-2010 yıllı içinden en az 1 hasar getirmiş olan müşteriler seçilmiş ve bu müşterilere ait 2006-2010 dönemdeki tüm verileri dikkate alınmıştır. 2006-2009 dönemi verileri eğitim için, 2010 dönem verileri test için kullanılmıştır. Modellemeye alınan değişkenler çizelge 3.3’de yer almaktadır.

SIGORTALI YIL
TOPLAM SIGORTALI YIL
PLAKA_IL_KODU
UYGULANAN_KADEME_NO
EXPOSURE
HASAR_DONEM (EXPOSURE DONEM)
ARACYAS
HASAR_VAR/YOK

Çizelge 3.3: Model Değişkenleri

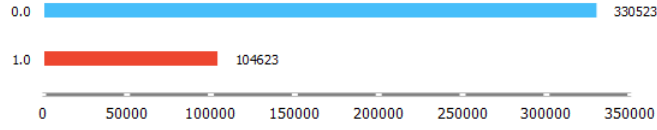
Modellemeye alınan değişkenler arasındaki korelasyon değerleri Çizelge 3.4’te yer almaktadır.

DEĞİŞKENELR	SPEARMAN KOR. DEĞER	PEARSON KOR.DEĞER
SIGORTALIYIL-TOP_SIGORTALIYIL	0.477	0.505
ARACYAS-SIGORTALIYIL	0.26	0.227
ARACYAS-TOP_SIGORTALIYIL	0.13	0.111
ARACYAS-EXPOSURE	-	-
EXPOSURE-TOP_SIGORTALIYIL	-	-
EXPOSURE-SIGORTALIYIL	-	-

Çizelge 3.4: Değişkenler Arasındaki Korelasyon

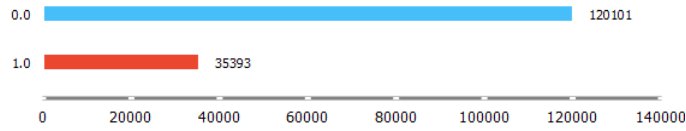
2006-2009 dönemine ait eğitim veri setindeki hasarlı/hasarsız (0/1) frekans-

larını gösteren box-plot grafiği Şekil 3.1’de yer almaktadır.



Şekil 3.1: Eğitim Veri Seti Hasarlı/Hasarsız (0/1) Frekans

2010 yılına ait test verilerinde hasarlı/hasarsız (0/1) frekanslarını gösteren box-plot grafiği şekil 3.2’de yer almaktadır.



Şekil 3.2: Test Veri Seti Hasarlı/Hasarsız (0/1) Frekans

3.2. Hasarlı/Hasarsız (0/1) Tahmin Modelleri

Bu bölümde Orange programı kullanılarak eğitim verileri ile kurulan tahmin modellerinin test performansları karşılaştırılacaktır. Makine öğrenmesi yöntemleri tek tek incelenerek model seçimi yapılacak ve tüm yöntemlerin seçilen modelleri karşılaştırılacaktır.

3.2.1. K En Yakın Komşu (kNN) Model Seçimi

K en yakın komşu algoritması ile kurulacak modellerde uzaklığın hesaplanacağı ölçüm yönteminin ve bu yöntemde uzaklık hesaplanırken komşular arasındaki etkinin ağırlıklandırılmasında kullanılacak yöntemin seçilmesi için modeller kurulmuştur. Kurulan modeller ve sonuçları Çizelge 3.5’de yer almaktadır.

	Sınıflar Arası Ortalama					0 Sınıf Tahminleri			1 Sınıf Tahminleri		
Model kNN	AUC	CA	F1	Pre.	Recall	F1	Precision	Recall	F1	Pre.	Recall
Euclidean-Dist.	0.728	0.797	0.752	0.778	0.797	0.881	0.805	0.973	0.312	0.685	0.20
Euclidean-Uni.	0.727	0.797	0.752	0.778	0.797	0.881	0.805	0.973	0.312	0.685	0.20
Manhattan-Dist.	0.728	0.797	0.752	0.778	0.797	0.881	0.805	0.972	0.313	0.684	0.20
Manhattan-Uni.	0.728	0.797	0.752	0.778	0.797	0.881	0.805	0.972	0.313	0.684	0.20
Chebyshev-Dis.	0.631	0.776	0.738	0.738	0.776	0.867	0.803	0.942	0.303	0.521	0.21
Chebyshev-Uni.	0.631	0.776	0.738	0.738	0.776	0.867	0.803	0.942	0.303	0.521	0.21

Model kNN	Kar Zarar	Kazanç
Euclidean-Dist.	-4,790,423.91	10,902,542.78
Euclidean-Uni.	-4,790,423.91	10,902,542.78
Manhattan-Dist.	-4,787,416.30	10,905,550.39
Manhattan-Uni.	-4,787,826.60	10,905,140.09
Chebyshev-Dis.	-3,785,985.04	11,906,981.65
Chebyshev-Uni.	-3,785,985.04	11,906,981.65
Gerçekleşen	-15,692,966.69	

Çizelge 3.5: k=99, kNN Modelleri ve Sonuçları

Sonuçlar incelendiğinde Euclidean ve Manhattan ölçüm yöntemleri ile kurulan modellerin AUC değerlerinin birbirine çok yakın olduğu, Chebyshev ölçüm yönteminin ise AUC değerinin düşük olduğu görülmüştür. Euclidean ve Manhattan ölçüm yöntemleri ile kurulan modellerin sonuçları birbirine çok yakın olduğundan bu modellerden kNN-Euclidean-Distance modeli en iyi model olarak seçilmiştir.

Model kNN	Sınıflar Arası Ortalama					0 Sınıf Tahminleri			1 Sınıf Tahminleri		
	AUC	CA	F1	Pre.	Recall	F1	Pre.	Recall	F1	Pre.	Recall
Euclidean-Dist.-3	0.637	0.758	0.737	0.728	0.758	0.851	0.810	0.896	0.352	0.450	0.280
Euclidean-Dist.-5	0.662	0.769	0.743	0.737	0.769	0.860	0.810	0.917	0.347	0.488	0.269
Euclidean-Dist.-7	0.676	0.776	0.747	0.743	0.776	0.865	0.810	0.928	0.346	0.516	0.261
Euclidean-Dist.-9	0.686	0.781	0.749	0.749	0.781	0.869	0.809	0.938	0.342	0.543	0.249
Euclidean-Dist.-19	0.704	0.793	0.753	0.766	0.793	0.877	0.808	0.960	0.330	0.625	0.224
Euclidean-Dist.-29	0.711	0.795	0.753	0.771	0.795	0.879	0.807	0.965	0.324	0.647	0.216
Euclidean-Dist.-39	0.718	0.796	0.753	0.772	0.796	0.880	0.807	0.967	0.323	0.655	0.214
Euclidean-Dist.-49	0.722	0.796	0.753	0.774	0.796	0.880	0.806	0.969	0.320	0.666	0.210
Euclidean-Dist.-59	0.724	0.796	0.752	0.774	0.796	0.880	0.806	0.969	0.319	0.666	0.210
Euclidean-Dist.-69	0.726	0.797	0.752	0.776	0.797	0.881	0.806	0.970	0.317	0.673	0.207
Euclidean-Dist.-79	0.727	0.797	0.752	0.776	0.797	0.881	0.806	0.971	0.316	0.675	0.207
Euclidean-Dist.-89	0.728	0.797	0.752	0.778	0.797	0.881	0.806	0.972	0.314	0.684	0.204
Euclidean-Dist.-99	0.728	0.797	0.752	0.778	0.797	0.881	0.805	0.973	0.312	0.685	0.202
Model kNN	Kar Zarar		Kazanç								
Euclidean-Dist.-3	-5,055,645.46		10,637,321.27								
Euclidean-Dist.-5	-4,997,665.18		10,695,301.55								
Euclidean-Dist.-7	-4,692,379.50		11,000,587.22								
Euclidean-Dist.-9	-4,769,138.45		10,923,828.28								
Euclidean-Dist.-19	-4,437,341.35		11,255,625.38								
Euclidean-Dist.-29	-4,550,146.42		11,142,820.31								
Euclidean-Dist.-39	-4,594,291.38		11,098,675.34								
Euclidean-Dist.-49	-4,640,274.41		11,052,692.31								
Euclidean-Dist.-59	-4,758,538.14		10,934,428.58								
Euclidean-Dist.-69	-4,880,616.41		10,812,350.32								
Euclidean-Dist.-79	-4,749,434.84		10,943,531.89								
Euclidean-Dist.-89	-4,744,225.21		10,948,741.52								
Euclidean-Dist.-99	-4,790,423.91		10,902,542.82								
Gerçekleşen	-15,692,966.73										

Çizelge 3.6: kNN Eucildean Distance k Değeri Belirleme Sonuçları

Euclidean-Distance çifti ile en iyi sonucu veren k değerinin belirlenmesi için kurulan modeller ve sonuçları Çizelge 3.6'da yer almaktadır.

kNN ile kurulan modeller incelendiğinde en çok kazanç getiren model $k = 19$ 'dur. Ancak $k = 19$ için AUC değeri 0.704'dir ve en yüksek AUC değeri ile arasındaki fark %2.4'tür. En yüksek AUC değerine sahip $k = 89$ ve $k = 99$ ile kurulan modellerin de kazanç getirileri yüksek olduğundan bu iki mo-

del arasından en yüksek kazanç getirisine sahip kNN-Euclidean-Distance-89 modeli en iyi model olarak seçilmiştir.

3.2.2. Karar Ağaçları Model Seçimi

Karar ağaçlarında belirtilen bir çoğunluğa ulaşıldığında bölme işlemi sonlandırılmaktadır. Çizelge 3.7’de farklı çoğunluk değerleri için kurulan modeller ve sonuçlar yer almaktadır.

Model	Sınıflar Arası Ortalama					0 Sınıf Tahminleri			1 Sınıf Tahminleri		
	AUC	CA	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall
Tree_%100	0.731	0.797	0.745	0.785	0.797	0.882	0.801	0.981	0.278	0.731	0.172
Tree_%90	0.731	0.797	0.745	0.785	0.797	0.882	0.801	0.981	0.278	0.731	0.172
Tree_%80	0.710	0.797	0.745	0.785	0.797	0.882	0.801	0.981	0.278	0.731	0.172
Tree_%70	0.500	0.772	0.673	0.597	0.772	0.872	0.772	1.000	-	-	-

Model	Kar/Zarar	Kazanç
Tree_%100	-6,487,321.71	9,205,645.01
Tree_%90	-6,487,321.71	9,205,645.01
Tree_%80	-6,487,321.71	9,205,645.01
Tree_%70	-15,692,966.73	
Gerçekleşen	-15,692,966.73	

Çizelge 3.7: Karar Ağaçları Model Sonuçları

Sonuçlar incelendiğinde en yüksek AUC değerine sahip, %90 ve %100 çoğunluk ile büyütülen ağaçların tüm değerlerinin aynı olduğu görülmektedir. Bu nedenle en iyi model olarak Tree %90 modeli seçilmiştir. Ancak bu model ile oluşturulan ağaç 181,804 boğum ve 93,972 yapraktan oluşmaktadır ve bu ağacın incelenmesi ve yorumlanması çok zor olduğundan farklı derinlik değerleri ile daha küçük ağaçlar elde etmek için kurulan modeller ve sonuçları Çizelge 3.8’de yer almaktadır. %90 büyütülen ağacın derinliği bilinmediğinden seçilebilen maksimum derinlik olan 65’ten başlayarak modeller oluşturulmuştur.

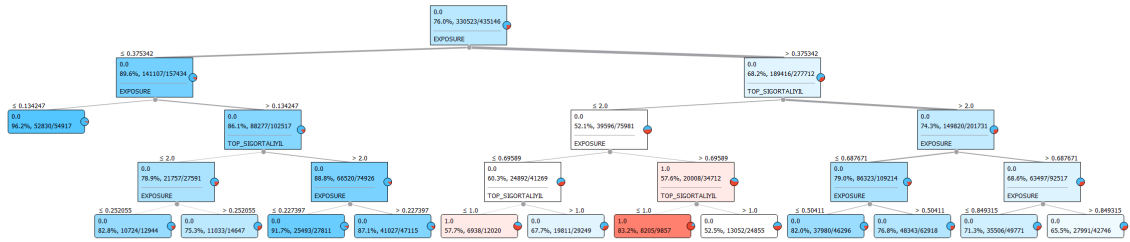
	Sınıflar Arası Ortalama					0 Sınıf Tahminleri			1 Sınıf Tahminleri		
Model	AUC	CA	F1	Pre.	Recall	F1	Pre.	Recall	F1	Pre.	Recall
Tree_%90	0.731	0.797	0.745	0.785	0.797	0.882	0.801	0.981	0.278	0.731	0.172
Tree_%90_65	0.731	0.797	0.745	0.785	0.797	0.882	0.801	0.981	0.278	0.731	0.172
Tree_%90_55	0.731	0.797	0.745	0.785	0.797	0.882	0.801	0.981	0.278	0.731	0.172
Tree_%90_45	0.731	0.797	0.745	0.785	0.797	0.882	0.801	0.981	0.278	0.731	0.172
Tree_%90_35	0.731	0.797	0.745	0.785	0.797	0.882	0.801	0.981	0.278	0.731	0.172
Tree_%90_25	0.731	0.797	0.745	0.785	0.797	0.882	0.801	0.981	0.278	0.731	0.172
Tree_%90_15	0.731	0.797	0.745	0.785	0.797	0.882	0.801	0.981	0.278	0.731	0.172
Tree_%90_10	0.731	0.797	0.745	0.785	0.797	0.882	0.801	0.981	0.278	0.731	0.172
Tree_%90_9	0.731	0.797	0.745	0.785	0.797	0.882	0.801	0.981	0.278	0.731	0.172
Tree_%90_8	0.731	0.797	0.745	0.785	0.797	0.882	0.801	0.981	0.278	0.731	0.172
Tree_%90_7	0.731	0.797	0.745	0.785	0.797	0.882	0.801	0.981	0.278	0.731	0.172
Tree_%90_6	0.731	0.797	0.745	0.785	0.797	0.882	0.801	0.981	0.278	0.731	0.172
Tree_%90_5	0.731	0.797	0.745	0.785	0.797	0.882	0.801	0.981	0.278	0.731	0.172
Tree_%90_4	0.724	0.797	0.753	0.775	0.797	0.880	0.807	0.969	0.323	0.668	0.213
Tree_%90_3	0.714	0.780	0.742	0.745	0.780	0.869	0.804	0.946	0.311	0.544	0.217
Tree_%90_2	0.696	0.772	0.673	0.597	0.772	0.872	0.772	1.000	-	-	-
Model	Kar/Zarar		Kazanç								
Tree_%90	-6,487,321.71		9,205,645.01								
Tree_%90_65	-6,487,321.71		9,205,645.01								
Tree_%90_55	-6,487,321.71		9,205,645.01								
Tree_%90_45	-6,487,321.71		9,205,645.01								
Tree_%90_35	-6,487,321.71		9,205,645.01								
Tree_%90_25	-6,487,321.71		9,205,645.01								
Tree_%90_15	-6,487,321.71		9,205,645.01								
Tree_%90_10	-6,487,321.71		9,205,645.01								
Tree_%90_9	-6,487,321.71		9,205,645.01								
Tree_%90_8	-6,487,321.71		9,205,645.01								
Tree_%90_7	-6,487,321.71		9,205,645.01								
Tree_%90_6	-6,487,321.71		9,205,645.01								
Tree_%90_5	-6,487,321.71		9,205,645.01								
Tree_%90_4	-4,064,295.32		11,628,671.40								
Tree_%90_3	-6,967,335.16		8,725,631.57								
Tree_%90_2	-15,692,966.73										
Gerçekleşen	-15,692,966.73										

Çizelge 3.8: Karar Ağaçları Budama Model Sonuçları

Model	Boğum Sayısı	Yaprak Sayısı
Tree_%90	181804	93972
Tree_%90_65	181804	93972
Tree_%90_55	181804	93972
Tree_%90_45	181804	93972
Tree_%90_35	181804	93972
Tree_%90_25	178908	92524
Tree_%90_15	93230	49684
Tree_%90_10	33430	19612
Tree_%90_9	23524	14471
Tree_%90_8	14692	9748
Tree_%90_7	7255	5544
Tree_%90_6	2192	1962
Tree_%90_5	283	260
Tree_%90_4	25	13
Tree_%90_3	13	7
Tree_%90_2	7	4

Çizelge 3.9: Karar Ağaçları Modelleri Boğum-Yaprak Sayısı

Çizelge 3.8’deki sonuçlar incelendiğinde derinlik 65-5 arasında kurulan modellerin sonuçlarının aynı olduğu görülmektedir. Bu da bize 5. derinlikten sonraki bölünmelerin önemsiz olduğunu göstermektedir. Derinlik 5-2 arasındaki modelleri incelediğimizde derinlik 4 ile kurulan modelin AUC değeri ile en yüksek AUC değeri arasındaki farkın %0.7 olduğu görülmektedir. AUC değerleri arasındaki farkın küçük olması, ağaç büyüklüğünün incelenbilir büyüklükte olması ve en yüksek kazanç değerine sahip olduğundan Tree %90 – 4 modeli en iyi model olarak seçilmiştir. Tree %90 – 4 modelinden oluşan karar ağacı Şekil 3.3’de yer almaktadır.



Şekil 3.3: Tree %90 – 4 Modeli Karar Ağacı

3.2.3. AdaBoost Model Seçimi

AdaBoost algoritmasıyla; temel tahmin edicilerin ağırlıklarının belirlenmesinde kullanılacak algoritmalar ve regresyon kayıp fonksiyonlarından en iyi sonucu veren çiftin belirlenmesi için kurulan modellerin detayları Çizelge 3.10’da, sonuçları ise Çizelge 3.11’de yer almaktadır.

Model	Parametre/Koşulu		
SameR	Temel Tah. Edici: Tree	Temel Tah. Edici:Tree	Temel Tah. Edici:Tree
	Tahmin Ed. Sayısı:100	Tahmin Ed. Sayısı:100	Tahmin Ed. Sayısı:100
	Sınıf. Alg.: Samme R	Sınıf. Alg.: Samme R	Sınıf. Alg.: Samme R
	Kayıp Reg.:Exp.	Kayıp Regr.:Linear	Kayıp Reg.:Square
Same	Temel Tah. Edici: Tree	Temel Tah. Edici:Tree	Temel Tah. Edici: Tree
	Tah. Edici Sayısı:100	Tah. Edici Sayısı:100	Tah. Edici Sayısı:100
	Sınıf. Alg.: Samme R	Sınıf. Alg.: Samme R	Sınıf. Alg.: Samme R
	Kayıp Reg.:Exp.	Kayıp Reg.:Linear	Kayıp Reg.:Square

Çizelge 3.10: AdaBoost Modelleri

Model	Sınıflar Arası Ortalama					0 Sınıf Tahminleri			1 Sınıf Tahminleri		
	AUC	CA	F1	Pre.	Recall	F1	Pre.	Recall	F1	Pre.	Recall
AdaBoost											
Same.R_Exp.	0.706	0.755	0.735	0.726	0.755	0.849	0.809	0.894	0.347	0.441	0.286
Same.R_Linear	0.706	0.755	0.735	0.726	0.755	0.849	0.809	0.894	0.347	0.441	0.286
Same.R_Square	0.704	0.757	0.735	0.726	0.757	0.851	0.809	0.897	0.344	0.445	0.281
Same_Exp.	0.696	0.783	0.750	0.751	0.783	0.870	0.809	0.941	0.343	0.553	0.248
Same_Linear	0.696	0.783	0.750	0.751	0.783	0.870	0.809	0.941	0.343	0.553	0.248
Same_Square	0.696	0.783	0.750	0.751	0.783	0.870	0.809	0.941	0.343	0.553	0.248

Model	Kar Zarar	Kazanç
kNN		
Same.R_Exp.	-5,495,151.01	10,197,815.72
Same.R_Linear	-5,495,151.01	10,197,815.72
Same.R_Square	-5,315,116.05	10,377,850.68
Same_Exp.	-4,579,840,46	11,113,126.27
Same_Linear	-4,579,840,46	11,113,126.27
Same_Square	-4,579,840.46	11,113,126.27
Gerçekleşen	-15,692,966.73	

Çizelge 3.11: AdaBoost Model Sonuçları

Sonuçlar incelendiğinde de Same algoritması ile farklı regresyon kayıp fonksiyonları ile kurulan modellerin aynı sonucu verdiği görülmektedir. Same.R algoritması ile kurulan modellerde ise Exponential ve Linear kayıp fonksiyonları ile kurulan modellerin aynı sonucu verdiği görülmektedir. Tüm modellerin AUC değerleri dikkate alındığında en yüksek AUC değeri ile en düşük AUC değeri arasındaki fark %1’dir. Ancak Same algoritması ile kurulan

modellerin AUC deęerleri 0.7'nin altında olduęundan Same.R algoritması ile kurulan modellerinde 1 sınıf precision deęeri 0.5'in altında olduęundan Ada Boost yöntemi ile kurulan modellerden iyi bir model seçilememiştir.

3.2.4. Rastgele Orman AdaBoost Model Seçimi

Rastgele orman yönteminde toplam deęişkenler arasından seçilen m adet deęişken ile ağaçlar oluşturulmaktadır. Çizelge 3.12'da farklı m deęişkenleri ile kurulan modeller ve sonuçları yer almaktadır.

Model	Sınıflar Arası Ortalama					0 Sınıf Tahminleri			1 Sınıf Tahminleri		
	AUC	CA	F1	Pre.	Recall	F1	Pre.	Recall	F1	Pre.	Recall
Ran. For.	0.733	0.796	0.754	0.774	0.796	0.880	0.807	0.960	0.324	0.663	0.214
Ran. For. 2	0.736	0.798	0.751	0.780	0.798	0.882	0.804	0.975	0.306	0.699	0.196
Ran. For. 3	0.734	0.797	0.750	0.778	0.797	0.881	0.804	0.974	0.304	0.688	0.195
Ran. Fore. 4	0.735	0.797	0.752	0.778	0.797	0.881	0.805	0.972	0.313	0.685	0.203
Ran. For. 5	0.735	0.797	0.753	0.776	0.797	0.881	0.806	0.970	0.318	0.675	0.208
Ran. For. 6	0.734	0.796	0.753	0.775	0.796	0.880	0.806	0.969	0.319	0.669	0.209
Ran. For. 7	0.734	0.797	0.753	0.776	0.797	0.881	0.806	0.971	0.318	0.676	0.208

Model	Kar-Zarar	Kazanç
Ran. For.	-4,782,416.15	10,910,550.58
Ran. For. 2	-5,434,691.02	10,258,275.71
Ran. For. 3	-5,428,447.38	10,264,519.35
Ran. Forest_4	-4,976,500.48	10,716,466.25
Ran. For. 5	-4,794,469.86	10,898,496.87
Ran. For. 6	-4,929,413.15	10,763,553.58
Ran. For. 7	-4,868,855.87	10,824,110.85
Gerçekleşen	-15,692,966.73	

Çizelge 3.12: Rastgele Orman Model ve Sonuçları

Sonuçlar incelendiğinde rastgele orman ile kurulan modellerin AUC deęerlerinin 0.733-0.736 arasında olduęu görölmektedir. AUC deęerleri arasındaki fark küçük olduęundan en iyi model olarak en çok kazanç getiren Random Forest modeli seçilmiştir.

3.2.5. Destek Vektör Makineleri Model Seçimi

Destek vektör makineleri yönteminde farklı Kernel çekirdek fonksiyonları ve farklı C (ayar parametresi) deęerleri ile modeller kurulabilmektedir. Çizelge 3.13'de kullanılabilecek çekirdek fonksiyonları ve C deęerleri ile kombine edilmiş modeller ve sonuçlar yer almaktadır.

	Sınıflar Arası Ortalama					0 Sınıf Tahminleri			1 Sınıf Tahminleri		
Model	AUC	CA	F1	Pre.	Recall	F1	Pre.	Recall	F1	Pre.	Recall
Linear_1	0.654	0.739	0.710	0.696	0.739	0.842	0.793	0.897	0.262	0.369	0.203
Linear_10	0.679	0.459	0.473	0.768	0.459	0.486	0.911	0.331	0.428	0.282	0.891
Linear_50	0.658	0.391	0.383	0.756	0.391	0.377	0.902	0.238	0.406	0.261	0.912
Linear_100	0.316	0.567	0.600	0.750	0.567	0.646	0.879	0.511	0.445	0.314	0.760
Polinomal_1	0.646	0.228	0.084	0.052	0.228	-	-	-	0.371	0.228	1.000
Polinomal_10	0.354	0.228	0.085	0.824	0.228	-	1.000	-	0.371	0.228	1.000
Polinomal_50	0.395	0.772	0.673	0.597	0.772	0.872	0.772	1.000	-	-	-
Polinomal_100	0.375	0.772	0.673	0.597	0.772	0.872	0.772	1.000	-	-	-
RBF_1	0.286	0.502	0.528	0.753	0.502	0.557	0.889	0.406	0.431	0.291	0.829
RBF_10	0.641	0.230	0.091	0.714	0.230	0.008	0.857	0.004	0.371	0.228	0.998
RBF_50	0.345	0.696	0.696	0.696	0.696	0.803	0.804	0.802	0.334	0.333	0.335
RBF_100	0.480	0.772	0.673	0.597	0.772	0.872	0.772	1.000	-	-	-
Sigmoid_1	0.713	0.351	0.316	0.780	0.351	0.290	0.935	0.172	0.402	0.255	0.960
Sigmoid_10	0.725	0.599	0.630	0.760	0.599	0.680	0.886	0.551	0.463	0.333	0.759
Sigmoid_50	0.610	0.386	0.384	0.720	0.386	0.382	0.858	0.246	0.390	0.252	0.862
Sigmoid_100	0.401	0.383	0.376	0.734	0.383	0.370	0.875	0.234	0.395	0.254	0.886
Model	Kar-Zarar		Kazanç								
Linear_1	-12,526,333.53		3,166,633.20								
Linear_10	-584,885.27		15,108,081.46								
Linear_50	-312,150.20		15,380,816.53								
Linear_100	-2,097,082.06		13,595,884.67								
Polinomal_1	-15,692,966.73		0.00								
Polinomal_10	3,120.00		15,696,086.73								
Polinomal_50	-15,692,966.73		0.00								
Polinomal_100	-15,692,966.73		0.00								
RBF_1	1,776,172.24		17,469,138.97								
RBF_10	80,374.77		15,773,341.49								
RBF_50	-8,993,133.38		6,699,833.35								
RBF_100	-15,692,966.73		0.00								
Sigmoid_1	435,493.15		16,128,459.88								
Sigmoid_10	313,046.93		16,006,013.66								
Sigmoid_50	-473,735.18		15,219,231.54								
Sigmoid_100	-1,124,683.74		14,568,282.99								
Gerçekleşen	-15,692,966.73										

Çizelge 3.13: SVM Model Sonuçları

Sonuçlar incelendiğinde modellerden yalnızca iki tanesinin AUC değerinin 0.7'nin üzerinde olduğu görülmektedir. Ancak bu modellerin tahmin edilenlerin doğruluğunu gösteren CA değeri 0.7'nin altında olduğundan SVM yöntemi ile kurulan modellerden iyi bir model seçilememiştir.

3.2.6. Sinir Ağları Model Seçimi

Sinir ağları, çok katmanlı hiyerarşik bir yapıdan oluştuğundan, bir den fazla aktivasyon ve optimizasyon yöntemi ile kurgulanabildiğinden senaryolar aşamalı olarak genişletilmiştir.

3.2.6.1 Optimizasyon Yöntemi Seçimi

İlgili aktivasyon fonksiyonu ile aynı şartlarda en iyi sonucu veren optimizasyon seçimi için Çizelge 3.14’te yer alan modeller kurulmuştur.

Model Sinir Ağları	Parametre/Koşulu		
Lojistik	Gizli Katman:1	Gizli Katman:1	Gizli Katman:1
	Nöron Sayısı:10	Nöron Sayısı:10	Nöron Sayısı:10
	Aktivasyon Fonksiyonu: Lojistik	Aktivasyon Fonksiyonu: Lojistik	Aktivasyon Fonksiyonu: Lojistik
	Optimizasyon: Adam	Optimizasyon: L-BFGS-B	Optimizasyon: SGD
	Yineleme:1000	Yineleme:1000	Yineleme:1000
Identity	Gizli Katman:1	Gizli Katman:1	Gizli Katman:1
	Nöron Sayısı:10	Nöron Sayısı:10	Nöron Sayısı:10
	Aktivasyon Fonksiyonu: Identity	Aktivasyon Fonksiyonu: Identity	Aktivasyon Fonksiyonu: Identity
	Optimizasyon: Adam	Optimizasyon: L-BFGS-B	Optimizasyon: SGD
	Yineleme:1000	Yineleme:1000	Yineleme:1000
tanh	Gizli Katman:1	Gizli Katman:1	Gizli Katman:1
	Nöron Sayısı:10	Nöron Sayısı:10	Nöron Sayısı:10
	Aktivasyon Fonksiyonu: tanh	Aktivasyon Fonksiyonu: tanh	Aktivasyon Fonksiyonu: tanh
	Optimizasyon: Adam	Optimizasyon: L-BFGS-B	Optimizasyon: SGD
	Yineleme:1000	Yineleme:1000	Yineleme:1000
Relu	Gizli Katman:1	Gizli Katman:1	Gizli Katman:1
	Nöron Sayısı:10	Nöron Sayısı:10	Nöron Sayısı:10
	Aktivasyon Fonksiyonu: Relu	Aktivasyon Fonksiyonu: Relu	Aktivasyon Fonksiyonu: Relu
	Optimizasyon: Adam	Optimizasyon: L-BFGS-B	Optimizasyon: SGD
	Yineleme:1000	Yineleme:1000	Yineleme:1000

Çizelge 3.14: Sinir Ağları Optimizasyon Yöntemi Seçim Modelleri

Aynı eğitim seti için çalıştırılmış olan modellerin aynı test seti üzerindeki sonuçlar Çizelge 3.15’te yer almaktadır.

Model	Sınıflar Arası Ortalama					0 Sınıf Tahminleri			1 Sınıf Tahminleri		
	AUC	CA	F1	Pre.	Recall	F1	Pre.	Recall	F1	Pre.	Recall
Identity_Adam_10	0.727	0.789	0.746	0.760	0.789	0.876	0.804	0.963	0.303	0.613	0.202
Identity_L-BFGS-B_10	0.727	0.789	0.735	0.765	0.789	0.877	0.797	0.976	0.250	0.657	0.155
Identity_SGD_10	0.727	0.788	0.730	0.766	0.788	0.877	0.794	0.980	0.229	0.671	0.138
Logistic_Adam_10	0.739	0.796	0.760	0.772	0.796	0.879	0.812	0.958	0.354	0.635	0.246
Logistic_L-BFGS-B_10	0.739	0.797	0.754	0.776	0.797	0.881	0.807	0.970	0.323	0.673	0.213
Logistic_SGD_10	0.727	0.789	0.749	0.760	0.789	0.875	0.806	0.958	0.320	0.603	0.218
Relu_Adam_10	0.737	0.797	0.746	0.785	0.797	0.882	0.802	0.980	0.284	0.727	0.177
Relu_L-BFGS-B_10	0.733	0.796	0.759	0.772	0.796	0.879	0.811	0.960	0.350	0.638	0.242
Relu_SGD_10	0.732	0.790	0.764	0.763	0.790	0.873	0.819	0.936	0.392	0.576	0.297
tanh_Adam_10	0.730	0.793	0.756	0.767	0.793	0.877	0.810	0.956	0.346	0.618	0.240
tanh_LBFGS-B_10	0.734	0.789	0.764	0.763	0.789	0.872	0.820	0.932	0.398	0.569	0.306
tanh_SGD_10	0.735	0.795	0.754	0.770	0.795	0.879	0.808	0.963	0.331	0.642	0.223

Model	Kar-Zarar	Kazanç
Identity_Adam_10	-6,799,467.85	8,893,498.84
Identity_L-BFGS-B_10	-8,646,245.84	7,046,720.85
Identity_SGD_10	-9,163,513.10	6,529,453.59
Logistic_Adam_10	-3,386,344.00	12,306,622.69
Logistic_L-BFGS-B_10	-4,436,941.47	11,256,025.22
Logistic_SGD_10	-6,159,710.26	9,533,256.43
Relu_Adam_10	-6,043,591.11	9,649,375.58
Relu_L-BFGS-B_10	-3,324,963.34	12,368,003.35
Relu_SGD_10	-1,933,280.21	13,759,686.48
tanh_Adam_10	-4,440,455.80	11,252,510.89
tanh_LBFGSB_10	-1,454,675.59	14,238,291.10
tanh_SGD_10	-5,153,313.30	10,539,653.39
Gerçekleşen	-15,692,966.69	

Çizelge 3.15: Sinir Ağları Optimizasyon Yöntemi Seçim Model Sonuçları

Sonuçlar incelendiğinde;

Identity aktivasyon fonksiyonu için tüm optimizasyon yöntemleri ile kurulan modellerin AUC değeri 0.727'dir. Bu modeller arasında en çok kazanç getiren Adam optimizasyon yöntemi ile kurulan model en iyi model olarak seçilmiştir.

Lojistik aktivasyon fonksiyonu için Adam ve L-BFGS-B optimizasyon yöntemleri ile kurulan modellerin AUC değerleri 0.739'dur. Bu iki model arasında en çok kazanç getiren Adam optimizasyon yöntemi ile kurulan model en iyi model olarak seçilmiştir.

Relu aktivasyon fonksiyonu ile kurulan modellerin en küçük AUC değeri ile en yüksek AUC değeri arasındaki fark %0.5'tir. AUC değerleri arasındaki fark küçük olduğundan en çok kazanç getiren SGD optimizasyon yöntemi ile kurulan model en iyi model olarak seçilmiştir.

Tanh aktivasyon fonksiyonu ile kurulan modeller incelendiğinde en yüksek kazanç getiren modelin L-BFGS-B optimizasyon ile kurulan model olduğu görülmektedir. L-BFGS-B optimizasyonu ile kurulan modelin AUC değeri ile en yüksek AUC değeri arasındaki fark %0.1 olduğundan en iyi model olarak L-BFGS-B optimizasyon yöntemi ile kurulan model en iyi model olarak

seçilmiştir.

3.2.6.2 Tek Katmanlı Yapıda Nöron Sayısı Seçimi

Aktivasyon fonksiyonları için seçilen uygun optimizasyon yöntemleri ile Çizelge 3.16'daki tek katmanlı modeller kurularak her aktivasyon fonksiyonu için en iyi sonucu veren aktivasyon-optimizasyon-nöron kombinasyonu seçilmiştir.

N.S.	Parametre/Koşul			
	identity_Adam	Logistic_Adam	Relu_SGD	tanh_L-BFGS-B
1	G.K.:1	G.K.:1	G.K.:1	G.K.:1
	N.S.:1	N.S.:1	N.S.:1	N.S.:1
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:100	Yineleme:100	Yineleme:100	Yineleme:100
2	G.K.:1	G.K.:1	G.K.:1	G.K.:1
	N.S.:2	N.S.:2	N.S.:2	N.S.:2
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
3	G.K.:1	G.K.:1	G.K.:1	G.K.:1
	N.S.:3	N.S.:3	N.S.:3	N.S.:3
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
4	G.K.:1	G.K.:1	G.K.:1	G.K.:1
	N.S.:4	N.S.:4	N.S.:4	N.S.:4
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
5	G.K.:1	G.K.:1	G.K.:1	G.K.:1
	N.S.:5	N.S.:5	N.S.:5	N.S.:5
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
6	G.K.:1	G.K.:1	G.K.:1	G.K.:1
	N.S.:6	N.S.:6	N.S.:6	N.S.:6
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
7	G.K.:1	G.K.:1	G.K.:1	G.K.:1
	N.S.:7	N.S.:7	N.S.:7	N.S.:7
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
8	G.K.:1	G.K.:1	G.K.:1	G.K.:1
	N.S.:8	N.S.:8	N.S.:8	N.S.:8
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
9	G.K.:1	G.K.:1	G.K.:1	G.K.:1
	N.S.:9	N.S.:9	N.S.:9	N.S.:9
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh

	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
10	G.K.:1	G.K.:1	G.K.:1	G.K.:1
	N.S.:10	N.S.:10	N.S.:10	N.S.:10
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
20	G.K.:1	G.K.:1	G.K.:1	G.K.:1
	N.S.:20	N.S.:20	N.S.:20	N.S.:20
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
30	G.K.:1	G.K.:1	G.K.:1	G.K.:1
	N.S.:30	N.S.:30	N.S.:30	N.S.:30
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
40	G.K.:1	G.K.:1	G.K.:1	G.K.:1
	N.S.:40	N.S.:40	N.S.:40	N.S.:40
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
50	G.K.:1	G.K.:1	G.K.:1	G.K.:1
	N.S.:50	N.S.:50	N.S.:50	N.S.:50
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
60	G.K.:1	G.K.:1	G.K.:1	G.K.:1
	N.S.:60	N.S.:60	N.S.:60	N.S.:60
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
70	G.K.:1	G.K.:1	G.K.:1	G.K.:1
	N.S.:70	N.S.:70	N.S.:70	N.S.:70
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
80	G.K.:1	G.K.:1	G.K.:1	G.K.:1
	N.S.:80	N.S.:80	N.S.:80	N.S.:80
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
90	G.K.:1	G.K.:1	G.K.:1	G.K.:1
	N.S.:90	N.S.:90	N.S.:90	N.S.:90
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
100	G.K.:1	G.K.:1	G.K.:1	G.K.:1
	N.S.:100	N.S.:100	N.S.:100	N.S.:100
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000

Çizelge 3.16: Aktivasyon Fonk. Optimizasyon Yöntemi-Nöron Sayısı Kombinasyonları

Aynı eğitim seti için çalıştırılmış olan modellerin aynı test seti üzerindeki

sonuçları Çizelge 3.17’de yer almaktadır.

Model	Sınıflar Arası Ortalama					0 Sınıf Tahminleri			1 Sınıf Tahminleri		
	AUC	CA	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall
Identity_Adam_1	0.726	0.788	0.742	0.759	0.788	0.876	0.802	0.948	0.290	0.614	0.157
Identity_Adam_2	0.727	0.790	0.739	0.763	0.789	0.877	0.799	0.954	0.269	0.643	0.170
Identity_Adam_3	0.727	0.788	0.749	0.757	0.789	0.874	0.807	0.952	0.325	0.590	0.179
Identity_Adam_4	0.727	0.789	0.745	0.760	0.790	0.876	0.803	0.956	0.302	0.614	0.190
Identity_Adam_5	0.727	0.789	0.735	0.763	0.788	0.877	0.797	0.960	0.253	0.650	0.200
Identity_Adam_6	0.727	0.789	0.745	0.760	0.788	0.876	0.803	0.958	0.303	0.614	0.201
Identity_Adam_7	0.727	0.790	0.748	0.760	0.789	0.876	0.805	0.959	0.313	0.609	0.202
Identity_Adam_8	0.727	0.789	0.741	0.762	0.789	0.877	0.800	0.961	0.279	0.632	0.204
Identity_Adam_9	0.726	0.789	0.747	0.758	0.787	0.875	0.805	0.961	0.313	0.601	0.209
Identity_Adam_10	0.727	0.789	0.746	0.760	0.789	0.876	0.804	0.963	0.303	0.613	0.211
Identity_Adam_20	0.727	0.788	0.749	0.758	0.786	0.875	0.806	0.962	0.323	0.595	0.209
Identity_Adam_30	0.727	0.789	0.746	0.760	0.789	0.876	0.804	0.961	0.306	0.612	0.213
Identity_Adam_40	0.726	0.787	0.750	0.757	0.790	0.874	0.807	0.961	0.330	0.584	0.207
Identity_Adam_50	0.727	0.786	0.751	0.755	0.790	0.873	0.808	0.963	0.336	0.575	0.205
Identity_Adam_60	0.726	0.790	0.748	0.760	0.790	0.876	0.805	0.963	0.316	0.607	0.212
Identity_Adam_70	0.727	0.790	0.747	0.760	0.789	0.876	0.805	0.965	0.312	0.610	0.222
Identity_Adam_80	0.727	0.789	0.747	0.760	0.789	0.876	0.805	0.969	0.311	0.609	0.224
Identity_Adam_90	0.726	0.789	0.746	0.759	0.789	0.875	0.804	0.972	0.306	0.606	0.230
Identity_Adam_100	0.726	0.789	0.747	0.760	0.789	0.876	0.804	0.975	0.309	0.607	0.230

Model	Kar/Zarar	Kazanç
Identity_Adam_1	-7,424,888.22	8,268,078.50
Identity_Adam_2	-8,160,304.17	7,532,662.56
Identity_Adam_3	-6,116,113.40	9,576,853.32
Identity_Adam_4	-6,894,957.54	8,798,009.19
Identity_Adam_5	-8,660,248.56	7,032,718.17
Identity_Adam_6	-6,881,152.71	8,811,814.02
Identity_Adam_7	-6,375,527.58	9,317,439.15
Identity_Adam_8	-7,696,728.56	7,996,238.16
Identity_Adam_9	-6,448,319.07	9,244,647.66
Identity_Adam_10	-6,799,467.85	8,893,498.87
Identity_Adam_20	-6,142,927.63	9,550,039.10
Identity_Adam_30	-6,651,404.97	9,041,561.75
Identity_Adam_40	-5,877,870.03	9,815,096.69
Identity_Adam_50	-5,925,854.58	9,767,112.15
Identity_Adam_60	-6,337,841.60	9,355,125.13
Identity_Adam_70	-6,434,944.09	9,258,022.63
Identity_Adam_80	-6,489,826.63	9,203,140.10
Identity_Adam_90	-6,735,091.04	8,957,875.69
Identity_Adam_100	-6,602,694.97	9,090,271.75
Gerçekleşen	-15,692,966.73	

Model	Sınıflar Arası Ortalama					0 Sınıf Tahminleri			1 Sınıf Tahminleri		
	AUC	CA	F1	Pre.	Recall	F1	Pre.	Recall	F1	Pre.	Recall
Logistic_Adam_1	0.726	0.783	0.752	0.752	0.783	0.869	0.812	0.936	0.355	0.547	0.263
Logistic_Adam_2	0.735	0.797	0.753	0.778	0.797	0.881	0.806	0.971	0.318	0.680	0.207
Logistic_Adam_3	0.735	0.798	0.754	0.777	0.798	0.881	0.807	0.970	0.324	0.677	0.213
Logistic_Adam_4	0.737	0.797	0.755	0.776	0.797	0.881	0.807	0.969	0.327	0.669	0.216
Logistic_Adam_5	0.736	0.797	0.756	0.775	0.797	0.880	0.808	0.966	0.333	0.662	0.223
Logistic_Adam_6	0.737	0.797	0.752	0.778	0.797	0.881	0.806	0.972	0.316	0.683	0.206
Logistic_Adam_7	0.738	0.795	0.760	0.770	0.795	0.878	0.813	0.955	0.361	0.625	0.254
Logistic_Adam_8	0.737	0.796	0.758	0.772	0.796	0.879	0.811	0.961	0.347	0.642	0.238
Logistic_Adam_9	0.738	0.796	0.758	0.771	0.796	0.879	0.810	0.961	0.346	0.640	0.237

Logistic_Adam_10	0.739	0.796	0.760	0.772	0.796	0.879	0.812	0.958	0.354	0.635	0.246
Logistic_Adam_20	0.739	0.793	0.764	0.767	0.793	0.876	0.818	0.943	0.387	0.595	0.287
Logistic_Adam_30	0.737	0.794	0.764	0.767	0.794	0.876	0.817	0.944	0.383	0.599	0.282
Logistic_Adam_40	0.738	0.795	0.763	0.769	0.795	0.878	0.815	0.951	0.373	0.615	0.268
Logistic_Adam_50	0.738	0.794	0.765	0.768	0.794	0.876	0.818	0.944	0.386	0.599	0.285
Logistic_Adam_60	0.739	0.793	0.765	0.767	0.793	0.876	0.818	0.942	0.388	0.594	0.288
Logistic_Adam_70	0.736	0.796	0.762	0.770	0.796	0.878	0.814	0.953	0.367	0.622	0.260
Logistic_Adam_80	0.737	0.794	0.763	0.768	0.794	0.877	0.816	0.948	0.377	0.608	0.273
Logistic_Adam_90	0.735	0.792	0.765	0.766	0.792	0.875	0.819	0.939	0.392	0.587	0.295
Logistic_Adam_100	0.737	0.793	0.765	0.766	0.793	0.875	0.818	0.940	0.391	0.590	0.292

Model	Kar/Zarar	Kazanç
Logistic_Adam_1	-5,333,146.32	10,359,820.41
Logistic_Adam_2	-4,653,324.52	11,039,642.20
Logistic_Adam_3	-4,268,945.09	11,424,021.63
Logistic_Adam_4	-4,388,259.69	11,304,707.04
Logistic_Adam_5	-4,083,284.28	11,609,682.45
Logistic_Adam_6	-4,912,411.60	10,780,555.13
Logistic_Adam_7	-3,247,002.03	12,445,964.69
Logistic_Adam_8	-3,619,653.72	12,073,313.00
Logistic_Adam_9	-3,855,300.21	11,837,666.52
Logistic_Adam_10	-3,386,344.00	12,306,622.72
Logistic_Adam_20	-1,957,672.28	13,735,294.45
Logistic_Adam_30	-2,102,486.70	13,590,480.03
Logistic_Adam_40	-2,406,158.87	13,286,807.86
Logistic_Adam_50	-1,786,637.83	13,906,328.90
Logistic_Adam_60	-1,965,423.11	13,727,543.61
Logistic_Adam_70	-2,657,232.73	13,035,734.00
Logistic_Adam_80	-2,106,790.66	13,586,176.07
Logistic_Adam_90	-1,296,527.70	14,396,439.03
Logistic_Adam_100	-1,592,834.97	14,100,131.76
Gerçekleşen	-15,692,966.73	0.00

	Sınıflar Arası Ortalama					0 Sınıf Tahminleri			1 Sınıf Tahminleri		
Model	AUC	CA	F1	Pre.	Recall	F1	Pre.	Recall	F1	Pre.	Recall
Relu_SGD_1	0.687	0.772	0.673	0.597	0.772	0.872	0.772	1.000	-	-	-
Relu_SGD_2	0.699	0.657	0.683	0.752	0.657	0.747	0.868	0.656	0.467	0.361	0.661
Relu_SGD_3	0.732	0.795	0.760	0.769	0.795	0.878	0.812	0.954	0.359	0.620	0.253
Relu_SGD_4	0.732	0.798	0.750	0.780	0.798	0.881	0.804	0.975	0.305	0.697	0.196
Relu_SGD_5	0.734	0.797	0.754	0.774	0.797	0.880	0.808	0.967	0.328	0.662	0.218
Relu_SGD_6	0.735	0.795	0.759	0.769	0.795	0.878	0.812	0.956	0.354	0.623	0.248
Relu_SGD_7	0.734	0.795	0.758	0.769	0.795	0.878	0.811	0.957	0.351	0.627	0.244
Relu_SGD_8	0.734	0.797	0.753	0.778	0.797	0.881	0.806	0.972	0.317	0.682	0.206
Relu_SGD_9	0.737	0.784	0.765	0.760	0.784	0.868	0.824	0.915	0.417	0.541	0.339
Relu_SGD_10	0.732	0.790	0.764	0.763	0.790	0.873	0.819	0.936	0.392	0.576	0.297
Relu_SGD_20	0.738	0.796	0.758	0.771	0.796	0.879	0.811	0.960	0.349	0.637	0.240
Relu_SGD_30	0.736	0.788	0.764	0.761	0.788	0.871	0.820	0.928	0.400	0.561	0.310
Relu_SGD_40	0.737	0.777	0.763	0.756	0.777	0.862	0.827	0.900	0.425	0.516	0.361
Relu_SGD_50	0.738	0.788	0.765	0.762	0.788	0.871	0.821	0.927	0.404	0.561	0.316
Relu_SGD_60	0.738	0.775	0.762	0.756	0.775	0.860	0.829	0.892	0.432	0.507	0.376
Relu_SGD_70	0.737	0.792	0.763	0.765	0.792	0.875	0.817	0.942	0.382	0.590	0.282
Relu_SGD_80	0.735	0.791	0.759	0.763	0.791	0.875	0.814	0.946	0.366	0.592	0.265
Relu_SGD_90	0.738	0.774	0.762	0.756	0.774	0.858	0.830	0.889	0.436	0.504	0.384
Relu_SGD_100	0.739	0.783	0.764	0.759	0.783	0.867	0.824	0.914	0.416	0.538	0.339

Model	Kar/Zarar	Kazanç
Relu_SGD_1	-15,692,966.73	
Relu_SGD_2	-278,575.84	15,414,390.88
Relu_SGD_3	-2,919,992.06	12,772,974.67

Relu_SGD_4	-5,005,263.84	10,687,702.89
Relu_SGD_5	-4,723,590.84	10,969,375.89
Relu_SGD_6	-3,812,424.11	11,880,542.62
Relu_SGD_7	-3,782,787.81	11,910,178.91
Relu_SGD_8	-4,806,807.57	10,886,159.16
Relu_SGD_9	-600,446.21	15,092,520.51
Relu_SGD_10	-1,933,280.21	13,759,686.52
Relu_SGD_20	-3,687,815.95	12,005,150.78
Relu_SGD_30	-1,611,934.64	14,081,032.08
Relu_SGD_40	-263,075.95	15,429,890.78
Relu_SGD_50	-1,282,231.91	14,410,734.81
Relu_SGD_60	422,262.14	16,115,228.86
Relu_SGD_70	-2,265,632.86	13,427,333.86
Relu_SGD_80	-3,586,759.24	12,106,207.49
Relu_SGD_90	892,714.19	16,585,680.92
Relu_SGD_100	-956,956.35	14,736,010.37
Gerçekleşen	-15,692,966.73	

Model	Sınıflar Arası Ortalama					0 Sınıf Tahminleri			1 Sınıf Tahminleri		
	AUC	CA	F1	Pre.	Recall	F1	Pre.	Recall	F1	Pre.	Recall
tanh_LBFGSB_1	0.726	0.785	0.751	0.753	0.785	0.871	0.810	0.943	0.343	0.562	0.247
tanh_LBFGSB_2	0.728	0.756	0.752	0.749	0.756	0.844	0.834	0.854	0.441	0.460	0.424
tanh_LBFGSB_3	0.736	0.787	0.764	0.761	0.787	0.870	0.822	0.925	0.405	0.555	0.318
tanh_LBFGSB_4	0.736	0.771	0.761	0.755	0.771	0.856	0.831	0.882	0.438	0.496	0.393
tanh_LBFGSB_5	0.739	0.795	0.762	0.769	0.795	0.877	0.815	0.951	0.371	0.614	0.265
tanh_LBFGSB_6	0.738	0.789	0.764	0.763	0.789	0.872	0.820	0.932	0.396	0.570	0.304
tanh_LBFGSB_7	0.735	0.731	0.739	0.751	0.731	0.820	0.847	0.795	0.465	0.424	0.513
tanh_LBFGSB_8	0.736	0.787	0.765	0.762	0.787	0.871	0.822	0.926	0.405	0.558	0.318
tanh_LBFGSB_9	0.734	0.764	0.756	0.750	0.764	0.851	0.831	0.872	0.433	0.478	0.397
tanh_LBFGSB_10	0.734	0.789	0.764	0.763	0.789	0.872	0.820	0.932	0.398	0.569	0.306
tanh_LBFGSB_20	0.737	0.786	0.766	0.762	0.786	0.869	0.825	0.917	0.419	0.547	0.340
tanh_LBFGSB_30	0.736	0.787	0.766	0.762	0.787	0.870	0.823	0.922	0.412	0.553	0.329
tanh_LBFGSB_40	0.734	0.777	0.762	0.756	0.777	0.862	0.827	0.898	0.426	0.514	0.364
tanh_LBFGSB_50	0.734	0.793	0.764	0.767	0.793	0.876	0.817	0.943	0.386	0.596	0.285
tanh_LBFGSB_60	0.721	0.789	0.762	0.762	0.789	0.873	0.818	0.936	0.388	0.573	0.293
tanh_LBFGSB_70	0.727	0.796	0.754	0.773	0.796	0.880	0.808	0.966	0.328	0.657	0.218
tanh_LBFGSB_80	0.701	0.789	0.751	0.760	0.789	0.875	0.808	0.954	0.331	0.596	0.229
tanh_LBFGSB_90	0.731	0.754	0.750	0.747	0.754	0.843	0.832	0.853	0.435	0.456	0.417
tanh_LBFGSB_100	0.687	0.797	0.753	0.776	0.797	0.881	0.806	0.971	0.318	0.675	0.208

Model	Kar/Zarar	Kazanç
tanh_LBFGSB_1	-5,789,634.44	9,903,332.29
tanh_LBFGSB_2	-1,637,316.11	14,055,650.62
tanh_LBFGSB_3	-722,449.11	14,970,517.62
tanh_LBFGSB_4	179,118.33	15,872,085.06
tanh_LBFGSB_5	-2,469,629.86	13,223,336.87
tanh_LBFGSB_6	-1,173,155.24	14,519,811.49
tanh_LBFGSB_7	2,002,966.05	17,695,932.78
tanh_LBFGSB_8	-797,358.06	14,895,608.66
tanh_LBFGSB_9	-2,072,418.69	13,620,548.04
tanh_LBFGSB_10	-1,454,675.59	14,238,291.14
tanh_LBFGSB_20	-408,296.84	15,284,669.88
tanh_LBFGSB_30	-1,279,787.62	14,413,179.11
tanh_LBFGSB_40	-348,388.14	15,344,578.59
tanh_LBFGSB_50	-2,035,960.72	13,657,006.00
tanh_LBFGSB_60	-1,214,964.53	14,478,002.20
tanh_LBFGSB_70	-4,878,482.14	10,814,484.59
tanh_LBFGSB_80	-4,888,322.34	10,804,644.38

tanh_LBFGS_90	-487,981.19	15,204,985.54
tanh_LBFGS_100	-4,381,958.96	11,311,007.77
Gerçekleşen	-15,692,966.73	

Çizelge 3.17: Sinir Ağları Tek Katmanlı Nöron Sayısı Belirleme Sonuçları

Sonuçlar incelendiğinde;

Identity aktivasyon fonksiyonu ve Adam optimizasyon yöntemi ile kurulan modeller de AUC'nin 0.726 veya 0.727 değerlerini aldığı görülmüştür. AUC değerleri birbirine çok yakın olduğundan en yüksek kazanç getiren Identity-Adam-40 modeli en iyi model olarak seçilmiştir.

Lojistik aktivasyon fonksiyonu ve Adam optimizasyon yöntemi ile kurulan modeller de AUC'nin 0.735 ile 0.739 arasında değerler aldığı görülmüştür. AUC değerleri arasındaki fark düşük olduğundan en yüksek kazanç getiren Logistic-Adam-90 modeli en iyi model olarak seçilmiştir.

Relu aktivasyon fonksiyonu ve SGD optimizasyon yöntemi ile kurulan modeller de en yüksek AUC değeri 0.739'dur. Yüksek kazanç getiren ve AUC değeri yüksek olan modeller arasından 1 sınıf tahminleri en iyi olan Relu-SGD-50 en iyi model olarak seçilmiştir.

Tanh aktivasyon fonksiyonu ve L-BFGS-B optimizasyon yöntemi ile kurulan modeller de en yüksek AUC değeri 0.739'dur. Yüksek kazanç getiren ve AUC değeri yüksek olan modeller arasından 1 sınıf tahminleri en iyi olan tanh-L-BFGS-B-20 modeli en iyi model olarak seçilmiştir.

3.2.6.3 Çift Katmanlı Yapıda Nöron Sayısı Seçimi

Bir önceki bölümde tek katman için belirlediğimiz nöron sayısı ile ikinci katmanda en iyi sonucu veren nöron sayısını belirlemek için Çizelge 3.18'deki modeller kurulmuştur.

	Parametre/Koşulu			
N.S.	identity_Adam	Logistic_Adam	Relu_SGD	tanh_L-BFGS-B
	Gizli Katman:1	Gizli Katman:1	Gizli Katman:1	Gizli Katman:1
	N.S.40;1	N.S.90;1	N.S.50;1	N.S.20;1

	N.S.40;40	N.S.90;40	N.S.50;40	N.S.20;40
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
50	Gizli Katman:1	Gizli Katman:1	Gizli Katman:1	Gizli Katman:1
	N.S.40;50	N.S.90;50	N.S.50;50	N.S.20;50
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
60	Gizli Katman:1	Gizli Katman:1	Gizli Katman:1	Gizli Katman:1
	N.S.40;60	N.S.90;60	N.S.50;60	N.S.20;60
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
70	Gizli Katman:1	Gizli Katman:1	Gizli Katman:1	Gizli Katman:1
	N.S.40;70	N.S.90;70	N.S.50;70	N.S.20;70
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
80	Gizli Katman:1	Gizli Katman:1	Gizli Katman:1	Gizli Katman:1
	N.S.40;80	N.S.90;80	N.S.50;80	N.S.20;80
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
90	Gizli Katman:1	Gizli Katman:1	Gizli Katman:1	Gizli Katman:1
	N.S.40;90	N.S.90;90	N.S.50;90	N.S.20;90
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000
100	Gizli Katman:1	Gizli Katman:1	Gizli Katman:1	Gizli Katman:1
	N.S.40;100	N.S.90;100	N.S.50;100	N.S.20;100
	Akt. F.: identity	Akt. F.: Lojistik	Akt. F.: Relu	Akt. F.: tanh
	Opt.: Adam	Opt.: Adam	Opt.: SGD	Opt.: L-BFGS-B
	Yineleme:1000	Yineleme:1000	Yineleme:1000	Yineleme:1000

Çizelge 3.18: Sınır Ağları Çift Katman Modelleri

Aynı eğitim seti için çalıştırılmış olan modellerin aynı test seti üzerindeki sonuçları Çizelge 3.19’da yer almaktadır.

Model	Sınıflar Arası Ortalama					0 Sınıf Tahminleri			1 Sınıf Tahminleri		
	AUC	CA	F1	Pre.	Recall	F1	Pre.	Recall	F1	Pre.	Recall
Identity_Adam_40	0.726	0.787	0.750	0.757	0.789	0.874	0.807	0.957	0.330	0.584	0.220
Identity_Adam_40,1	0.727	0.789	0.747	0.760	0.790	0.876	0.804	0.963	0.309	0.608	0.202
Identity_Adam_40,2	0.727	0.790	0.747	0.761	0.789	0.876	0.804	0.960	0.308	0.612	0.211
Identity_Adam_40,3	0.727	0.789	0.743	0.761	0.789	0.876	0.802	0.962	0.291	0.623	0.201
Identity_Adam_40,4	0.727	0.790	0.748	0.760	0.785	0.876	0.805	0.941	0.313	0.609	0.255
Identity_Adam_40,5	0.726	0.789	0.745	0.760	0.787	0.876	0.803	0.950	0.303	0.611	0.233
Identity_Adam_40,6	0.726	0.789	0.742	0.760	0.789	0.876	0.802	0.952	0.289	0.620	0.235
Identity_Adam_40,7	0.726	0.787	0.750	0.756	0.789	0.873	0.808	0.966	0.333	0.581	0.188
Identity_Adam_40,8	0.727	0.789	0.745	0.760	0.789	0.876	0.803	0.966	0.303	0.612	0.189
Identity_Adam_40,9	0.727	0.790	0.746	0.761	0.789	0.876	0.804	0.962	0.304	0.616	0.201
Identity_Adam_40,10	0.726	0.789	0.743	0.761	0.789	0.876	0.802	0.962	0.293	0.620	0.203
Identity_Adam_40,20	0.726	0.789	0.748	0.759	0.790	0.875	0.806	0.960	0.317	0.601	0.211
Identity_Adam_40,30	0.727	0.789	0.745	0.760	0.789	0.876	0.803	0.963	0.302	0.613	0.200

Identity_Adam_40,40	0.727	0.789	0.746	0.760	0.789	0.876	0.804	0.966	0.305	0.612	0.190
Identity_Adam_40,50	0.726	0.789	0.743	0.761	0.789	0.876	0.802	0.958	0.290	0.622	0.215
Identity_Adam_40,60	0.727	0.789	0.752	0.759	0.790	0.874	0.809	0.962	0.336	0.590	0.206
Identity_Adam_40,70	0.727	0.785	0.752	0.754	0.787	0.871	0.811	0.954	0.350	0.560	0.219
Identity_Adam_40,80	0.726	0.789	0.747	0.760	0.789	0.875	0.805	0.965	0.313	0.606	0.192
Identity_Adam_40,90	0.727	0.789	0.749	0.759	0.789	0.875	0.806	0.961	0.322	0.600	0.207

Model	Kar/Zarar	Kazanç
Identity_Adam_40	-5,877,870.03	9,815,096.69
Identity_Adam_40,1	-6,581,523.97	9,111,442.76
Identity_Adam_40,2	-6,585,528.87	9,107,437.85
Identity_Adam_40,3	-7,318,063.96	8,374,902.76
Identity_Adam_40,4	-6,391,014.62	9,301,952.11
Identity_Adam_40,5	-6,858,206.91	8,834,759.82
Identity_Adam_40,6	-7,413,854.60	8,279,112.12
Identity_Adam_40,7	-5,881,237.80	9,811,728.92
Identity_Adam_40,8	-6,844,513.17	8,848,453.56
Identity_Adam_40,9	-6,711,362.92	8,981,603.81
Identity_Adam_40,10	-7,228,537.06	8,464,429.66
Identity_Adam_40,20	-6,248,011.11	9,444,955.61
Identity_Adam_40,30	-6,871,367.79	8,821,598.94
Identity_Adam_40,40	-6,754,602.03	8,938,364.70
Identity_Adam_40,50	-7,319,614.25	8,373,352.47
Identity_Adam_40,60	-5,620,800.15	10,072,166.57
Identity_Adam_40,70	-5,547,661.48	10,145,305.25
Identity_Adam_40,80	-6,474,858.97	9,218,107.75
Identity_Adam_40,90	-6,104,032.19	9,588,934.54
Identity_Adam_40,100	-6,449,038.49	9,243,928.23
Gerçekleşen	-15,692,966.73	0.00

Model	Sınıflar Arası Ortalama					0 Sınıf Tahminleri			1 Sınıf Tahminleri		
	AUC	CA	F1	Pre.	Recall	F1	Pre.	Recall	F1	Pre.	Recall
Logistic_Adam_90	0.735	0.792	0.765	0.766	0.792	0.875	0.819	0.939	0.392	0.587	0.295
Logistic_Adam_90,1	0.731	0.775	0.763	0.756	0.775	0.859	0.830	0.891	0.434	0.507	0.380
Logistic_Adam_90,2	0.726	0.795	0.762	0.770	0.795	0.878	0.815	0.951	0.371	0.617	0.265
Logistic_Adam_90,3	0.724	0.794	0.765	0.767	0.794	0.876	0.818	0.942	0.389	0.596	0.288
Logistic_Adam_90,4	0.728	0.788	0.765	0.762	0.788	0.871	0.822	0.927	0.405	0.560	0.317
Logistic_Adam_90,5	0.734	0.795	0.761	0.769	0.795	0.877	0.814	0.952	0.365	0.616	0.260
Logistic_Adam_90,6	0.724	0.785	0.764	0.760	0.785	0.868	0.823	0.918	0.412	0.544	0.332
Logistic_Adam_90,7	0.730	0.788	0.765	0.762	0.788	0.871	0.822	0.925	0.408	0.558	0.322
Logistic_Adam_90,8	0.728	0.788	0.766	0.763	0.788	0.871	0.823	0.925	0.410	0.560	0.323
Logistic_Adam_90,9	0.727	0.768	0.758	0.752	0.768	0.854	0.830	0.880	0.432	0.487	0.388
Logistic_Adam_90,10	0.727	0.745	0.745	0.746	0.745	0.834	0.836	0.833	0.443	0.440	0.446
Logistic_Adam_90,20	0.722	0.788	0.764	0.762	0.788	0.871	0.820	0.929	0.400	0.563	0.310
Logistic_Adam_90,30	0.727	0.790	0.766	0.764	0.790	0.872	0.822	0.930	0.406	0.570	0.316
Logistic_Adam_90,40	0.718	0.785	0.765	0.761	0.785	0.869	0.824	0.918	0.415	0.547	0.334
Logistic_Adam_90,50	0.733	0.785	0.764	0.759	0.785	0.869	0.822	0.920	0.408	0.546	0.326
Logistic_Adam_90,60	0.715	0.775	0.762	0.756	0.775	0.860	0.829	0.894	0.431	0.509	0.373
Logistic_Adam_90,70	0.729	0.788	0.767	0.763	0.788	0.870	0.824	0.923	0.415	0.557	0.330
Logistic_Adam_90,80	0.729	0.792	0.764	0.766	0.792	0.875	0.818	0.940	0.389	0.587	0.291
Logistic_Adam_90,90	0.700	0.758	0.751	0.745	0.758	0.847	0.828	0.868	0.422	0.463	0.387

Model	Kar/Zarar	Kazanç
Logistic_Adam_90	-1,296,527.70	14,396,439.03
Logistic_Adam_90,1	1,142,725.28	16,835,692.00
Logistic_Adam_90,2	-2,741,485.36	12,951,481.36
Logistic_Adam_90,3	-1,378,422.58	14,314,544.15
Logistic_Adam_90,4	107,135.58	15,800,102.31
Logistic_Adam_90,5	-1,822,060.60	13,870,906.13

Logistic_Adam_90,6	286,436.91	15,979,403.63
Logistic_Adam_90,7	-10,428.30	15,682,538.43
Logistic_Adam_90,8	-292,550.47	15,400,416.26
Logistic_Adam_90,9	2,427,431.32	18,120,398.04
Logistic_Adam_90,10	1,145,883.57	16,838,850.29
Logistic_Adam_90,20	-369,013.21	15,323,953.52
Logistic_Adam_90,30	-231,586.48	15,461,380.25
Logistic_Adam_90,40	371,624.79	16,064,591.52
Logistic_Adam_90,50	216,014.82	15,908,981.55
Logistic_Adam_90,60	1,414,277.41	17,107,244.14
Logistic_Adam_90,70	85,193.51	15,778,160.23
Logistic_Adam_90,80	-1,184,019.75	14,508,946.97
Logistic_Adam_90,90	323,777.47	16,016,744.19
Logistic_Adam_90,100	95,796.18	15,788,762.90
Gerçekleşen	-15,692,966.73	0.00

	Sınıflar Arası Ortalama					0 Sınıf Tahminleri			1 Sınıf Tahminleri		
Model	AUC	CA	F1	Pre.	Recall	F1	Pre.	Recall	F1	Pre.	Recall
Relu_SGD_50	0.738	0.788	0.765	0.762	0.788	0.871	0.821	0.927	0.404	0.561	0.316
Relu_SGD_50,1	0.738	0.776	0.763	0.757	0.776	0.860	0.829	0.894	0.433	0.510	0.375
Relu_SGD_50,2	0.733	0.798	0.750	0.782	0.798	0.882	0.804	0.976	0.303	0.705	0.193
Relu_SGD_50,3	0.738	0.798	0.752	0.780	0.798	0.882	0.805	0.974	0.313	0.692	0.202
Relu_SGD_50,4	0.738	0.792	0.764	0.766	0.792	0.875	0.818	0.940	0.390	0.588	0.292
Relu_SGD_50,5	0.738	0.796	0.760	0.771	0.796	0.879	0.812	0.956	0.358	0.629	0.251
Relu_SGD_50,6	0.737	0.794	0.763	0.767	0.794	0.876	0.816	0.947	0.377	0.604	0.274
Relu_SGD_50,7	0.737	0.780	0.765	0.759	0.780	0.864	0.828	0.903	0.430	0.524	0.364
Relu_SGD_50,8	0.737	0.777	0.763	0.757	0.777	0.861	0.829	0.896	0.431	0.513	0.372
Relu_SGD_50,9	0.736	0.788	0.766	0.762	0.788	0.870	0.823	0.924	0.410	0.557	0.325
Relu_SGD_50,10	0.738	0.795	0.762	0.768	0.795	0.877	0.814	0.951	0.369	0.613	0.264
Relu_SGD_50,20	0.739	0.783	0.765	0.760	0.783	0.866	0.826	0.909	0.424	0.534	0.352
Relu_SGD_50,30	0.735	0.795	0.761	0.769	0.795	0.878	0.813	0.954	0.364	0.620	0.257
Relu_SGD_50,40	0.738	0.795	0.762	0.769	0.795	0.878	0.814	0.951	0.370	0.616	0.265
Relu_SGD_50,50	0.738	0.791	0.765	0.764	0.791	0.873	0.820	0.935	0.396	0.577	0.301
Relu_SGD_50,60	0.733	0.797	0.755	0.776	0.797	0.881	0.808	0.968	0.328	0.668	0.217
Relu_SGD_50,70	0.738	0.796	0.760	0.771	0.796	0.879	0.812	0.958	0.357	0.633	0.249
Relu_SGD_50,80	0.738	0.795	0.761	0.769	0.795	0.878	0.814	0.952	0.367	0.616	0.261
Relu_SGD_50,90	0.738	0.789	0.764	0.763	0.789	0.872	0.820	0.932	0.398	0.569	0.306

Model	Kar/Zarar	Kazanç
Relu_SGD_50	-1,282,231.91	14,410,734.81
Relu_SGD_50,1	24,250.65	15,717,217.38
Relu_SGD_50,2	-4,946,128.49	10,746,838.24
Relu_SGD_50,3	-4,591,627.86	11,101,338.87
Relu_SGD_50,4	-1,886,031.56	13,806,935.17
Relu_SGD_50,5	-3,283,356.45	12,409,610.27
Relu_SGD_50,6	-2,289,172.10	13,403,794.63
Relu_SGD_50,7	733,717.24	16,426,683.97
Relu_SGD_50,8	465,491.66	16,158,458.39
Relu_SGD_50,9	-830,338.91	14,862,627.82
Relu_SGD_50,10	-2,956,266.99	12,736,699.74
Relu_SGD_50,20	10,379.93	15,703,346.66
Relu_SGD_50,30	-3,248,580.48	12,444,386.24
Relu_SGD_50,40	-2,362,942.17	13,330,024.55
Relu_SGD_50,50	-1,571,383.58	14,121,583.15
Relu_SGD_50,60	-4,317,331.25	11,375,635.48
Relu_SGD_50,70	-3,131,331.56	12,561,635.16
Relu_SGD_50,80	-2,980,855.98	12,712,110.75
Relu_SGD_50,90	-1,580,815.25	14,112,151.47

Relu_SGD_50,100	-3,337,077.84	12,355,888.89
Gerçekleşen	-15,692,966.73	0.00

	Sınıflar Arası Ortalama					0 Sınıf Tahminleri			1 Sınıf Tahminleri		
Model	AUC	CA	F1	Pre.	Recall	F1	Pre.	Recall	F1	Pre.	Recall
tanh_LBFGSB_20	0.737	0.786	0.766	0.762	0.786	0.869	0.825	0.917	0.419	0.547	0.340
tanh_LBFGSB_20,1	0.736	0.780	0.764	0.758	0.780	0.864	0.827	0.903	0.427	0.524	0.361
tanh_LBFGSB_20,2	0.739	0.777	0.764	0.758	0.777	0.861	0.830	0.895	0.435	0.513	0.377
tanh_LBFGSB_20,3	0.736	0.736	0.743	0.752	0.736	0.825	0.846	0.805	0.464	0.431	0.502
tanh_LBFGSB_20,4	0.738	0.791	0.764	0.764	0.791	0.874	0.818	0.938	0.389	0.581	0.293
tanh_LBFGSB_20,5	0.739	0.787	0.766	0.762	0.787	0.870	0.823	0.923	0.412	0.555	0.327
tanh_LBFGSB_20,6	0.739	0.784	0.766	0.761	0.784	0.867	0.826	0.913	0.422	0.540	0.346
tanh_LBFGSB_20,7	0.735	0.790	0.765	0.764	0.790	0.873	0.820	0.932	0.400	0.572	0.308
tanh_LBFGSB_20,8	0.740	0.797	0.755	0.776	0.797	0.881	0.808	0.968	0.330	0.667	0.219
tanh_LBFGSB_20,9	0.737	0.796	0.762	0.771	0.796	0.878	0.814	0.954	0.365	0.625	0.258
tanh_LBFGSB_20,10	0.738	0.791	0.765	0.765	0.791	0.874	0.820	0.935	0.396	0.578	0.302
tanh_LBFGSB_20,20	0.733	0.796	0.759	0.770	0.796	0.879	0.812	0.957	0.355	0.630	0.247
tanh_LBFGSB_20,30	0.733	0.795	0.760	0.769	0.795	0.878	0.813	0.955	0.361	0.622	0.254
tanh_LBFGSB_20,40	0.736	0.794	0.757	0.768	0.794	0.878	0.811	0.956	0.349	0.621	0.243
tanh_LBFGSB_20,50	0.731	0.795	0.760	0.769	0.795	0.878	0.813	0.954	0.362	0.620	0.256
tanh_LBFGSB_20,60	0.735	0.793	0.761	0.766	0.793	0.876	0.814	0.949	0.368	0.605	0.264
tanh_LBFGSB_20,70	0.736	0.794	0.763	0.768	0.794	0.877	0.815	0.948	0.376	0.607	0.272
tanh_LBFGSB_20,80	0.733	0.796	0.762	0.771	0.796	0.878	0.814	0.954	0.365	0.625	0.258
tanh_LBFGSB_20,90	0.724	0.794	0.759	0.768	0.794	0.877	0.812	0.954	0.357	0.618	0.251

Model	Kar/Zarar	Kazanç
tanh_LBFGSB_20	-408,296.84	15,284,669.88
tanh_LBFGSB_20,1	786,269.45	16,479,236.17
tanh_LBFGSB_20,2	1,137,273.81	16,830,240.54
tanh_LBFGSB_20,3	2,270,705.56	17,963,672.29
tanh_LBFGSB_20,4	-1,306,730.10	14,386,236.63
tanh_LBFGSB_20,5	-689,480.57	15,003,486.15
tanh_LBFGSB_20,6	7,440.07	15,700,406.80
tanh_LBFGSB_20,7	-1,186,642.36	14,506,324.37
tanh_LBFGSB_20,8	-3,558,519.62	12,134,447.11
tanh_LBFGSB_20,9	-2,683,493.82	13,009,472.91
tanh_LBFGSB_20,10	-1,722,352.06	13,970,614.67
tanh_LBFGSB_20,20	-3,583,273.25	12,109,693.48
tanh_LBFGSB_20,30	-3,161,803.02	12,531,163.70
tanh_LBFGSB_20,40	-3,334,444.87	12,358,521.86
tanh_LBFGSB_20,50	-3,234,942.26	12,458,024.46
tanh_LBFGSB_20,60	-2,739,507.64	12,953,459.09
tanh_LBFGSB_20,70	-2,386,328.59	13,306,638.14
tanh_LBFGSB_20,80	-2,717,364.97	12,975,601.75
tanh_LBFGSB_20,90	-3,153,577.55	12,539,389.18
tanh_LBFGSB_20,100	-754,338.52	14,938,628.20
Gerçekleşen	-15,692,966.73	0.00

Çizelge 3.19: Sinir Ağları Çift Katmanlı Nöron Sayısı Belirleme Sonuçları

Sonuçlar incelendiğinde;

Identity aktivasyon fonksiyonu, Adam optimizasyon yöntemi ve birinci katmandaki nöron sayısı 40 ile kurulan modeller de AUC'nin 0.726 veya 0.727

değerlerini aldığı görülmüştür. AUC değerleri birbirine çok yakın olduğundan en yüksek kazanç getiren Identity-Adam-40,70 modeli en iyi model olarak seçilmiştir.

Lojistik aktivasyon fonksiyonu, Adam optimizasyon yöntemi ve birinci katmandaki nöron sayısı 90 ile kurulan modellerden en yüksek kazanç getiren Logistic-Adam-90,9 ve Logistic-Adam-90,10 modellerinin 1 sınıf tahminlerinin 0.5'in altında olduğu görülmektedir. AUC, kazanç ve 1 sınıf tahminleri dikkate alındığında en yüksek AUC değeri ile arasında %0.4 fark olan, 1 sınıf tahminleri 0.5'in üstünde olan ve yüksek kazanç getiren Logistic-Adam-90,1 modeli en iyi model olarak seçilmiştir.

Relu aktivasyon fonksiyonu, SGD optimizasyon yöntemi ve birinci katmandaki nöron sayısı 50 ile kurulan modeller de en yüksek AUC değeri %0.739 olup Relu-SGD-50,20'ne aittir. En yüksek kazanç getiren diğer modellerden daha yüksek CA ve 1 sınıf precision değerine sahip olduğundan en iyi model olarak Relu-SGD-50,20 modeli seçilmiştir.

Tanh aktivasyon fonksiyonu, L-BFGS-B birinci katmandaki nöron sayısı 20 ile kurulan modeller de en yüksek AUC değerine ve yüksek kazanç getiren diğer modellerden daha yüksek 1 sınıf precision değerine sahip tanh-LBFGSB-20,2 modeli en iyi model olarak seçilmiştir.

3.2.7. Lojistik Regresyon Model Seçimi

Lojistik regresyonda ceza terimi Lasso1 ve Ridge2 yöntemleri düzenlenebilmektedir. Bu yöntemlerde kullanılan farklı sabit katsayı değerleri ile kurulan modeller ve sonuçları Çizelge 3.20'de yer almaktadır.

Model Log. Reg.	Sınıflar Arası Ortalama					0 Sınıf Tahminleri			1 Sınıf Tahminleri		
	AUC	CA	F1	Pre.	Recall	F1	Pre.	Recall	F1	Pre.	Recall
Ridge_0.001	0.727	0.786	0.752	0.755	0.786	0.876	0.807	0.959	0.325	0.613	0.221
Ridge_0.01	0.727	0.786	0.752	0.755	0.786	0.873	0.809	0.949	0.340	0.580	0.240
Ridge_0.1	0.727	0.786	0.752	0.755	0.786	0.872	0.810	0.945	0.343	0.570	0.246
Ridge_1	0.727	0.786	0.752	0.755	0.786	0.872	0.810	0.944	0.346	0.567	0.249
Ridge_10	0.727	0.786	0.752	0.755	0.786	0.872	0.810	0.944	0.346	0.566	0.249
Ridge_100	0.727	0.787	0.752	0.757	0.787	0.872	0.810	0.943	0.347	0.566	0.250
Ridge_1000	0.725	0.791	0.751	0.763	0.791	0.872	0.810	0.943	0.347	0.566	0.250
Lasso_0.001	0.726	0.790	0.749	0.761	0.790	0.876	0.806	0.960	0.317	0.612	0.214
Lasso_0.01	0.727	0.789	0.747	0.760	0.789	0.876	0.805	0.960	0.312	0.608	0.210
Lasso_0.1	0.727	0.789	0.748	0.760	0.789	0.875	0.806	0.958	0.317	0.604	0.215
Lasso_1	0.727	0.767	0.756	0.749	0.767	0.854	0.827	0.883	0.422	0.485	0.374
Lasso_10	0.727	0.744	0.746	0.748	0.744	0.833	0.838	0.828	0.449	0.439	0.459
Lasso_100	0.727	0.726	0.735	0.747	0.726	0.817	0.844	0.792	0.457	0.417	0.505
Lasso_1000	0.727	0.715	0.728	0.748	0.715	0.806	0.849	0.768	0.461	0.405	0.535

Model	Kar-Zarar	Kazanç
Ridge_0.001	-5,726,694.67	9,966,272.06
Ridge_0.01	-5,495,213.52	10,197,753.21
Ridge_0.1	-5,603,631.72	10,089,335.01
Ridge_1	-5,486,949.12	10,206,017.61
Ridge_10	-5,492,970.21	10,199,996.52
Ridge_100	-5,491,707.02	10,201,259.71
Ridge_1000	-5,491,707.02	10,201,259.71
Lasso_0.001	-6,279,278.40	9,413,688.32
Lasso_0.01	-6,453,539.38	9,239,427.34
Lasso_0.1	-6,296,315.07	9,396,651.65
Lasso_1	-2,032,153.83	13,660,812.90
Lasso_10	658,462.20	16,351,428.92
Lasso_100	-1,580,628.76	14,112,337.97
Lasso_1000	-1,649,505.81	14,043,460.92
Gerçekleşen	-15,692,966.73	

Çizelge 3.20: Lojistik Regresyon Model Sonuçları

Sonuçlar incelendiğinde modellerin AUC değerlerinin 0.726 veya 0.727 olduğu görülmektedir. En çok kazanç getiren modellerin 1 sınıf tahminlerinin precision değerlerinin 0.5'in altında olduğu görülmektedir. Bu nedenle precision değeri 0.5'in üzerinde olup en yüksek kazanç getiren Logistic Regression-Ridge-1 modeli en iyi model olarak seçilmiştir.

3.3. Seçilen Modellerin Karşılaştırılması

Seçilen modellerin Çizelge 3.21'deki sonuçları incelendiğinde en yüksek kazanç getiren modellerin Sinir Ağları ile kurulan modeller olduğu görülmektedir. NN-Relu-SGD-50,20 ve NN-tanh-LBFGSB-20,2 modellerinin AUC de-

ğerleri 0.739 olup diğer modellerin AUC değerinden yüksek olmaktadır.

Model	Sınıflar Arası Ortalama					0 Sınıf Tahminleri			1 Sınıf Tahminleri		
	AUC	CA	F1	Pre.	Recall	F1	Pre.	Recall	F1	Pre.	Recall
kNN_Euc.-Dis._89	0.728	0.797	0.752	0.778	0.797	0.881	0.806	0.972	0.314	0.684	0.204
Tree_%90_4	0.724	0.797	0.753	0.775	0.797	0.880	0.807	0.969	0.323	0.668	0.213
Ran. For.	0.733	0.796	0.754	0.774	0.796	0.880	0.807	0.960	0.324	0.663	0.214
NN Iden. Adam 40,70	0.727	0.785	0.752	0.754	0.787	0.871	0.811	0.954	0.350	0.560	0.219
NN Log. Adam 90,1	0.731	0.775	0.763	0.756	0.775	0.859	0.830	0.891	0.434	0.507	0.380
NN Relu SGD 50,20	0.739	0.783	0.765	0.760	0.783	0.866	0.826	0.909	0.424	0.534	0.352
NN tanh LBFGSB20,2	0.739	0.777	0.764	0.758	0.777	0.861	0.830	0.895	0.435	0.513	0.377
Naive Bayes	0.721	0.788	0.758	0.759	0.788	0.873	0.814	0.940	0.369	0.572	0.273
Log. Reg.Ridge1	0.727	0.786	0.752	0.755	0.786	0.872	0.810	0.944	0.346	0.567	0.249

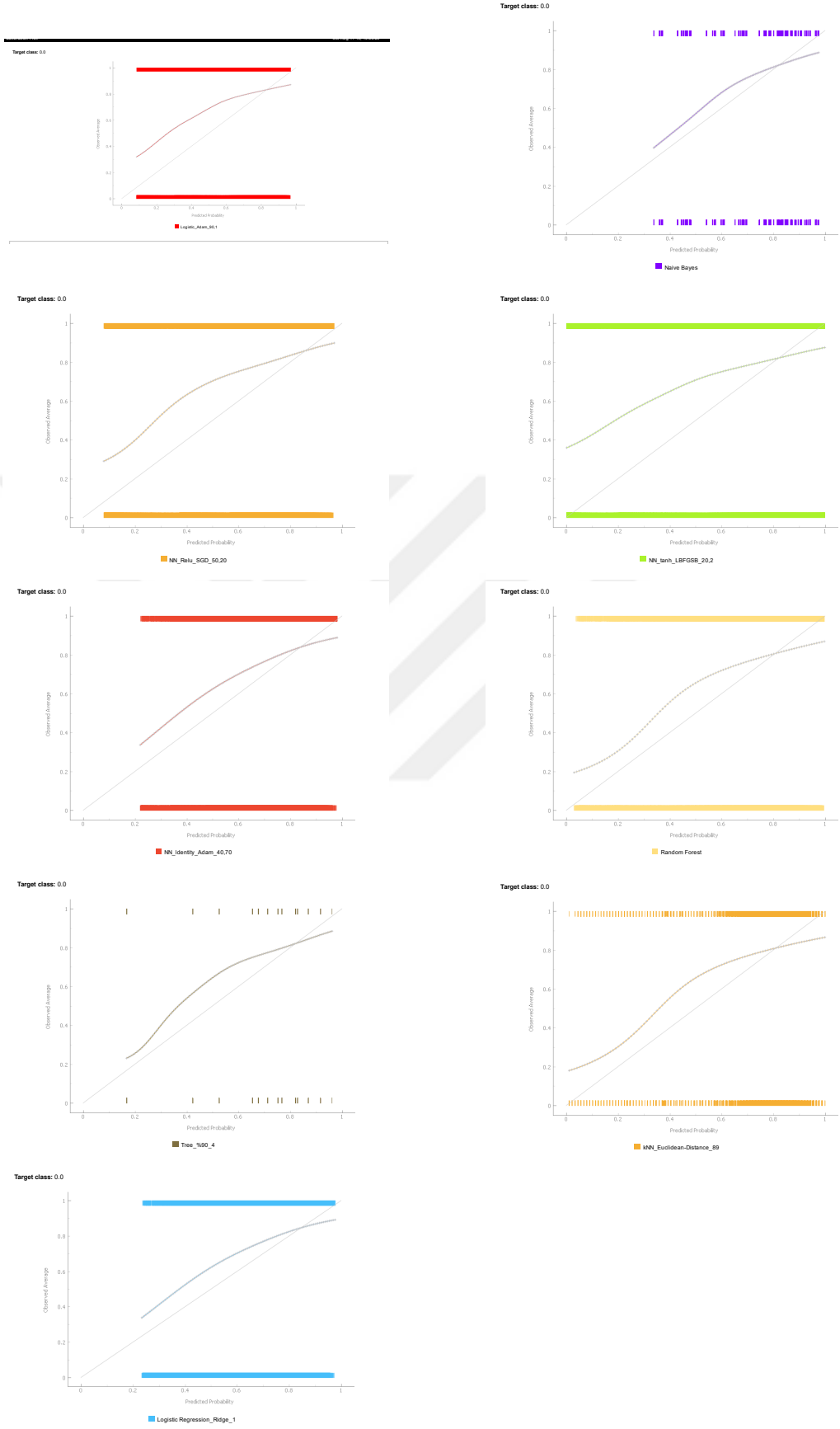
Model	Kar-Zarar	Kazanç
kNN Euc. Dis. 89	-4,744,225.21	10,948,741.52
Tree_%90_4	-4,064,295.32	11,628,671.40
Ran. For.	-4,782,416.15	10,910,550.58
NN Iden. Adam 40,70	-5,547,661.48	10,145,305.25
NN Log. Adam 90,1	1,142,725.28	16,835,692.00
NN Relu SGD 50,20	10,379.93	15,703,346.66
NN tanh LBFGSB 20,2	1,137,273.81	16,830,240.54
Naive Bayes	-3,682,680.51	12,010,286.22
Log. Reg. Ridge 1	-5,486,949.12	10,206,017.61
Gerçekleşen	-15,692,966.73	

Çizelge 3.21: Seçilen Modellerin Sonuçları

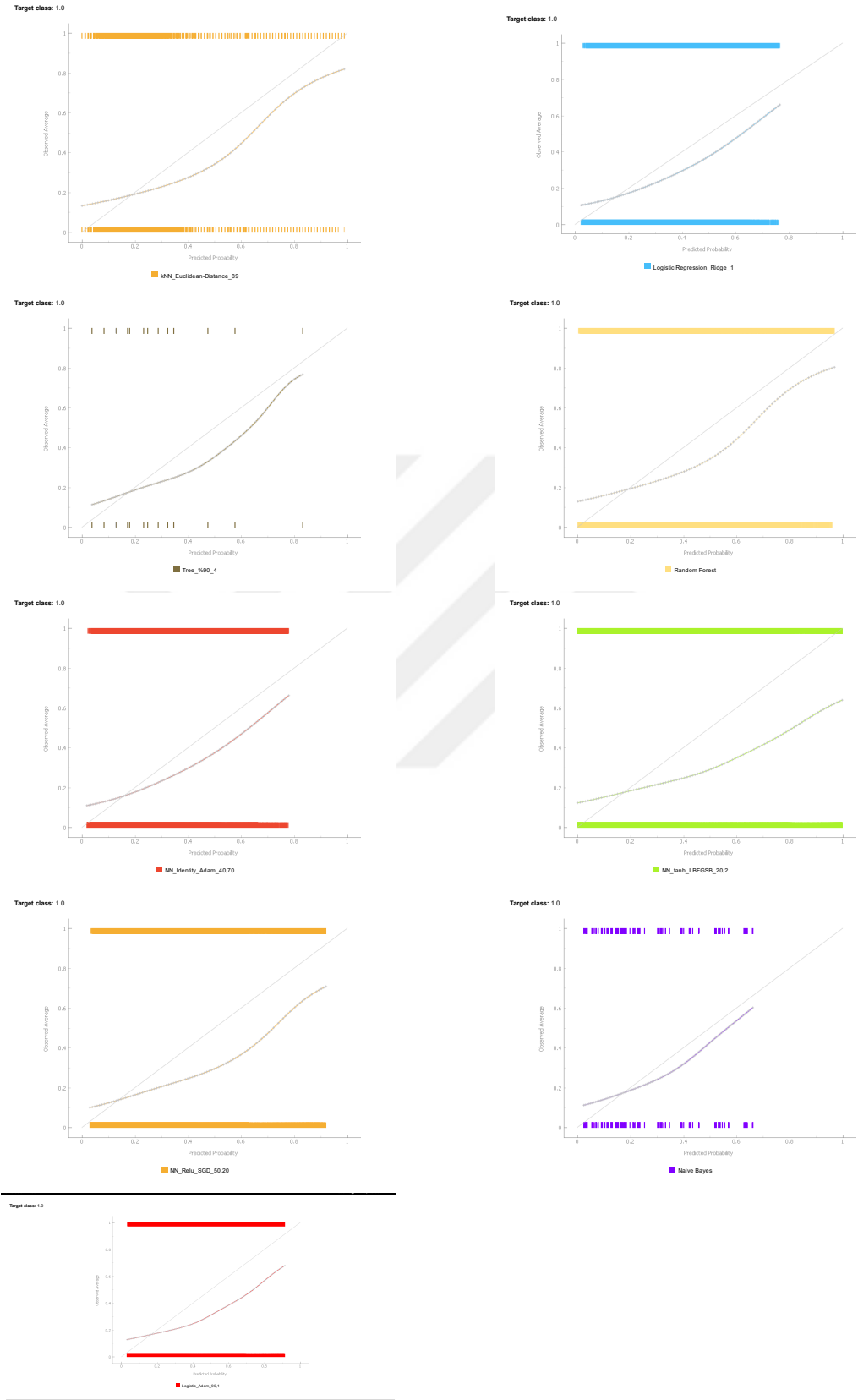
Çizelge 3.22'deki karmaşıklık matrisini incelediğimizde 1 sınıf tahminleri yüksek olan kNN-Euclidean-Distance-89, Tree %90 – 4, Random Forest* modellerinin kazanç getirilerinin düşük olduğu görülmektedir. Bunun nedeni verideki hasarlı müşterilerin %69.9'nun zararda olmasıdır. Hasar getiren müşterilerin %31.1'i karlı müşteriler olmaktadır. Sinir ağları ile kurulan modellerin daha düşük 1 sınıf tahminine rağmen daha karlı olduğu görülmektedir. Bu sonuçlar neticesinde sinir ağları ile kurulan modellerin riskli grubu yakalamadaki başarısının diğer modellerden daha iyi olduğunu söyleyebiliriz.

kNN_Euclidean-Distance_89				Tree_%90_4				Random Forest			
		Predicted				Predicted				Predicted	
		0	1			0	1			0	1
Actual	0	80.60%	31.60%	Actual	0	80.70%	33.20%	Actual	0	80.70%	33.70%
	1	19.40%	68.40%		1	19.30%	66.80%		1	19.30%	66.30%
NN_Logistic_Adam_90,1				NN_Logistic_Adam_90,1				NN_Relu_SGD_50,20			
		Predicted				Predicted				Predicted	
		0	1			0	1			0	1
Actual	0	81.10%	44.00%	Actual	0	83.00%	49.30%	Actual	0	82.60%	46.60%
	1	18.90%	56.00%		1	17.00%	50.70%		1	17.40%	53.40%
NN_tanh_LBFGSB_20,2				Naive Bayes				Logistic Regression_Ridge_1			
		Predicted				Predicted				Predicted	
		0	1			0	1			0	1
Actual	0	83.00%	48.70%	Actual	0	81.40%	42.80%	Actual	0	81.00%	43.30%
	1	17.00%	51.30%		1	18.60%	57.20%		1	19.00%	56.70%

Çizelge 3.22: Seçilen Modellerin Karmaşıklık Matrisleri



Şekil 3.4: Seçilen Modeller-0 Sınıf Calibration Plot



Şekil 3.5: Seçilen Modeller-1 Sınıf Calibration Plot

4. SONUÇLAR

Bu çalışmada makine öğrenmesi yöntemleri ile sigorta şirketine yeni katılacak kasko müşterilerinin hasar getirip getirmeme durumu tahmin edilerek sigorta şirketinin karlı büyümesine katkı sağlanması amaçlanmıştır.

Simülasyon ile oluşturulan veri kullanılarak her bir makine öğrenmesi yöntemi ile kurulan modellerden en iyi sonuç veren modeller seçilmiş ve performansları karşılaştırılmıştır. SVM ve AdaBoost yöntemi ile kurulan modellerin performansları düşük olduğundan bu yöntemlerden iyi bir model seçilememiştir. Karar ağaçları, kNN, Naive Bayes, Sinir Ağları, Rastgele Orman ve Lojistik Regresyon yöntemleri ile kurulan modeller karşılaştırıldığında; 1 sınıf tahminleri yüksek olan kNN_Euclidean-Distance_89, Tree_%90_4, Random Forest* modellerinin kazanç getirilerinin düşük olduğu görülmüştür. Sinir ağları ile kurulan modellerin daha düşük 1 sınıf tahminine rağmen kazanç getirilerinin daha yüksek olduğu görülmüştür. Bu sonuçlar ışığında riskli sınıfı tahmin etmede Sinir Ağlarının diğer modellerden daha iyi olduğu ve seçilen modellerin hepsinin sigorta şirketinin kazancını arttırdığı görülmüştür.

Verinin niteliği düşünüldüğünde, çok fazla açıklayıcı değişken bulunmamasına rağmen makine öğrenmesi yöntemleri ile kurulan modellerde sigorta şirketlerinin kar/zarar durumunu önemli ölçüde etkileyecek sonuçlar alınmıştır. Daha fazla veri ve daha fazla açıklayıcı değişken ile makine öğrenmesi yöntemleriyle kurulacak modeller ile müşterilerin riskini sigortalanmadan önce tahmin ederek sigorta şirketinin karlı büyümesine katkı sağlanabilir. Ayrıca kim yeniler modeli kurularak sigortasını yenileme tercihi düşük olan karlı müşteriler önceden tahmin edilerek kişiye özel yapılan kampanyalar ile karlı müşterinin portföyde kalması sağlanabilir.

EKLER

EK:1 Karmaşıklık Matrisi ve Test Sonuçları

		Predicted	
		0	1
Actual	0	TP	FN
	1	FP	TN
Karmaşıklık Matrisi			

TP: Pozitif tahmin edilenlerden doğru olanları

TN: Negatif tahmin edilenlerden doğru olanları

FP: Pozitif tahmin edilenlerden yanlış olanları (1.Tip Hata)

FN: Negatif tahmin edilenlerden yanlış olanları (2.Tip Hata)

Test sonuçlarında yer alan değerlerin açıklamaları;

AUC: ROC eğrisinin altında kalan alan olmaktadır. ROC eğriside duyarlılık (sensitivity) ve seçicilik (specificity) arasındaki ilişkinin grafiksel gösterimi olmaktadır.

CA: Yapılan tahminlerden doğru olanların oranını göstermekedir. $(TP+TN) / (TP+TN+FP+FN)$

Precision: Pozitif tahmin edilenlerden doğru olanlarının yüzdesi $TP / (TP+FP)$

Recall: Test datasındaki pozitiflerin doğru tahmin edilenlerinin yüzdesi $TP / (TP+FN)$

F1: Recall ve Precision değerlerinin harmonic ortalamasıdır.

KAYNAKLAR

- Anderberg, M. R. (2016). Cluster Analysis for Applications. *Cluster Analysis for Applications*, 1. <https://doi.org/10.1016/c2013-0-06161-0>
- Barber, D. (2012). Bayesian Reasoning and Machine Learning. *Bayesian Reasoning and Machine Learning*. <https://doi.org/10.1017/cbo9780511804779>
- Bhowmik, R. (2011). Detecting Auto Insurance Fraud by Data Mining Techniques. *Emerging Trends in Computing and Information Sciences*, 2(4), 156–162.
- Breiman, L. (2001). *RANDOM FORESTS*. 1–33.
- Chen,K.(2013).LearningMethods.
<https://www.math.ust.hk/~makchen/MAFS5140/MAFS5140.Htm>, 801–809.
https://doi.org/10.1007/978-2-8178-0016-5_6
- Cortes, C., & Vapnik, V. (1995). In Silico Log P Prediction for a Large Data Set with Support Vector Machines, Radial Basis Neural Networks and Multiple Linear Regression. *Chemical Biology & Drug Design*, 20, 273–297. <https://doi.org/10.1111/j.1747-0285.2009.00840.x>
- Dal Pozzolo, A. (2011). *Comparison of Data Mining Techniques for Insurance Claim Prediction*.Retrievedfrom
http://www.ulb.ac.be/di/map/adalpozz/pdf/Claim_prediction.pdf
- Frempong, N. K., Nicholas, N., & Boateng, M. A. (2017). Decision Tree as a Predictive Modeling Tool for Auto Insurance Claims. *International Journal of Statistics and Applications*, 7(2), 117–120. <https://doi.org/10.5923/j.statistics.20170702.07>
- Gepp, A., Wilson, J. H., Kumar, K., & Bhattacharya, S. (2012). A Comparative Analysis of Decision Trees Vis a-vis Other Computational Data Mining Techniques in Automotive Insurance Fraud Detection. *Journal of Data Science*, 10, 537–561.
- Harej, B., Gächter, R., Jamal, S., Schaeper, A., Boumezoued, A., Mcguinness, A., ... Magatti, V. (2017). Individual Claim Development with Machine Learning. *ASTIN*.
- Henckaerts, R., & C, M. (2019). *Boosting insights in insurance tariff plans with tree-based machine learning*. (1972), 1–33.
- Jamal, S., Canto, S., Fernwood, R., Giancaterino, C., Hiabu, M., Invernizzi, L., ... Shen, H. (2018). Machine Learning and Traditional Methods Synergy in Non-Life Reserving. *ASTIN*.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An Introduction to Statistical Learning* (Vol. 103). <https://doi.org/10.1007/978-1-4614-7138-7>
- Jong, P., Heller, G.Z. (2008). Generalize Linear Models for Insurance Data, .
- Maglogiannis, I., Karpousiz, K., Wallace, M., Soldatos, J. (2007) Emerging Artificial Intelligence Applications In Computer Engineering. The authors and IOS Press.

- Patrick L. Brockett, Xiaohua Xia, & Derrig, R. A. (1998). Using Kohonen's Self-Organizing Feature Map to Uncover Automobile Bodily Injury Claims Fraud. *The Journal of Risk and Insurance*, 65(2), 245–274.
- Rich Caruana, & Niculescu-Mizil Alexandru. (2009). *An Empirical Comparison of Supervised Learning Algorithms*. 161–168. <https://doi.org/10.1145/1143844.1143865>
- Shapiro, A. F., & Jain, L. C. (2003). Intelligent and Other Computational Techniques in Insurance: Theory and Applications (Series on Innovative Intelligence, 6). In *Intelligence* (Vol. 6).
- Spedicato, G., Dutang, C., & Petrini, L. (2017). Machine Learning Methods to Perform Pricing Optimization. A Comparison with Standard GLMs. *Cas*, (Pozzolo 2011), 1–21.
- Ticconi, D. (2018). *Individual claims reserving in Credit insurance using GLM and Machine Learning*. (December), 0–14. <https://doi.org/10.13140/RG.2.2.13118.33600>
- van Wezel, M. C., Kok, J. N., & Sere, K. (2002). *Determining the number of dimensions underlying customer-choices with a competitive neural network*. 484–489. <https://doi.org/10.1109/icnn.1996.548941>
- Vapnik, V. (1998). *Statistical Learning Theory*. John Wiley & Sons.
- Vaughn, M. L., Ong, E., & Cavill, S. J. (1997). Interpretation and knowledge discovery from a Multilayer Perceptron network that performs whole life assurance risk assessment. *Neural Computing and Applications*, 6(4), 201–213. <https://doi.org/10.1007/BF01501507>



HACETTEPE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
YÜKSEK LİSANS/~~DOKTORA~~-TEZ ÇALIŞMASI ORJİNALLİK RAPORU

HACETTEPE ÜNİVERSİTESİ
FEN BİLİMLER ENSTİTÜSÜ
AKTÜERYA BİLİMLERİ ANABİLİM DALI BAŞKANLIĞI'NA

Tarih: 10/09/2019

Tez Başlığı / Konusu: Kasko Sigortasında Makine Öğrenmesi Yöntemleri ile Hasarlı/Hasarsız Durum Tahmini

Yukarıda başlığı/konusu gösterilen tez çalışmamın a) Kapak sayfası, b) Giriş, c) Ana bölümler d) Sonuç kısımlarından oluşan toplam 55 sayfalık kısmına ilişkin, 10/09/2019 tarihinde ~~şahsen~~/tez danışmanım tarafından Turnitin adlı intihal tespit programından aşağıda belirtilen filtrelemeler uygulanarak alınmış olan orijinallik raporuna göre, tezimin benzerlik oranı %5 'tir.

Uygulanan filtrelemeler:

- 1- Kaynakça hariç
- 2- Alıntılar hariç/~~dâhil~~
- 3- 5 kelimedenden daha az örtüşme içeren metin kısımları hariç

Hacettepe Üniversitesi Fen Bilimleri Enstitüsü Tez Çalışması Orijinallik Raporu Alınması ve Kullanılması Uygulama Esasları'nı inceledim ve bu Uygulama Esasları'nda belirtilen azami benzerlik oranlarına göre tez çalışmamın herhangi bir intihal içermediğini; aksinin tespit edileceği muhtemel durumda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Gereğini saygılarımla arz ederim.

10/09/19
Tarih ve İmza

Adı Soyadı: SEDA KİLİSLİ

Öğrenci No: N13129364

Anabilim Dalı: AKTÜERYA BİLİMLERİ

Programı: AKTÜERYA BİLİMLERİ

Statüsü: ☒ Y.Lisans ☐ Doktora ☐ Bütünleşik Dr.

DANIŞMAN ONAYI

UYGUNDUR.

PROF. DR. KASIRGA YILDIRAK

(Unvan, Ad Soyad, İmza)

ÖZGEÇMİŞ

Adı Soyadı : Seda Kilisli

Doğum Yeri : Mersin

Medeni Hali : Bekar

E-posta : seda.klsl@gmail.com

Adres : Mecidiyeköy mah. Ak sokak. Şimşek Apt. Kat:7 No:9 Şişli/İstanbul

Eğitim

Lise : MTSO Anadolu Lisesi, 2005-2009

Lisans : Hacettepe Üniversitesi, Aktüerya Bilimleri,2009-2013

Yüksek Lisans : Hacettepe Üniversitesi, Aktüerya Bilimleri,2013-

Yabancı Dil ve Düzeyi

İngilizce, Orta

İş Deneyimi

Deneyim Alanları