T.C.
İSTANBUL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

MAKİNE ÖĞRENMESİYLE MÜŞTERİ KAYIPLARININ TAHMİNİ

Melike GÜNAY

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

DANIŞMA
Dr. Öğr. Üyesi Tolga ENSARİ

Haziran, 2018

İSTANBUL

Bu çalışma, 25.06.2018 tarihinde aşağıdaki jüri tarafından Bilgisayar Mühendisliği Anabilim Dalı, Bilgisayar Mühendisliği Programında Yüksek Lisans tezi olarak kabul edilmiştir.

Tez Jürisi

Dr. Öğr. Üyesi Tolga ENSARİ(Danışman)
İstanbul Üniversitesi
Mühendislik Fakültesi

Prof. Dr. Ahmet SERTBAŞ
İstanbul Üniversitesi
Mühendislik Fakültesi

Doç. Dr. Zeynep ORMAN
İstanbul Üniversitesi
Mühendislik Fakültesi

Dr. Öğr. Üyesi Pelin GÖRGEL
İstanbul Üniversitesi
Mühendislik Fakültesi

Dr. Öğr. Üyesi Yusuf YASLAN
İstanbul Teknik Üniversitesi
Bilgisayar ve Bilişim Fakültesi

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

ÖNSÖZ

Öncelikle yükseköğrenimim boyunca ve bu çalışmayı hazırlarken benden desteğini ve yardımını esirgemeyen, motivasyonumu her zaman üst seviyede tutmama yardımcı olan saygı değer hocam Sayın Dr. Tolga ENSARİ'ye, yoğun iş ve eğitim dönemim boyunca bana destek olan aileme ve iş arkadaşlarım Öznur Şengel, Semira Bener ve Özge Günaydın'a en içten teşekkürlerimi sunarım.

Haziran 2018 Melike GÜNAY

İÇİNDEKİLER

Sayfa No

ŞEKİL LİSTESİ

Sayfa No

TABLO LİSTESİ

Sayfa No

SİMGE VE KISALTMA LİSTESİ

Kısaltmalar	Açıklama
BFO	: Bacterial Foraging Optimization
ÇOY	: Çarpımsal Olasılıklar Yöntemi
DVM	: Destek Vektör Makineleri
LR	: Lojistik Regresyon
NB	: Naive Bayes
PSO	: Particle Swarm Optimization
SFLA	: Shuddled Frog Leaping Algorithm
TBA	: Temel Bileşen Analizi
YSA	: Yapay Sinir Ağları

ÖZET

YÜKSEK LİSANS TEZİ

MAKİNE ÖĞRENMESİYLE MÜŞTERİ KAYIPLARININ TAHMİNİ

Melike GÜNAY

İstanbul Üniversitesi

Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

Danışman : Dr. Öğr. Üyesi Tolga EBNSARİ

Kayıp müşteri analizi, son zamanlarda üyelik tabanlı çalışan firmalar için önem kazanmaya başlamıştır. Çünkü bu firmalar, kaybetmek üzere oldukları müşterileri tespit edip onları üye olmaya devam etmeleri için ikna kampanyaları düzenlemek istemektedirler. Böylelikle yeni üye elde etmektense var olan üyeyi elde tutmanın karından yararlanacaklardır. Sektörün bu ihtiyacı sebebi ile bu çalışmada makine öğrenmesi teknikleri ile telekomünikasyon sektöründe ki müşterilerin üyelikten çıkıp çıkmayacağı tahmin edilmeye çalışılmıştır. Konu ile ilgili geçmişte yapılan çalışmalar incelendiğinde en çok kullanılan yöntemler bulunmuş ve algoritmaların performansı ölçülmüştür. Bununla birlikte Lojistik Regresyon ve Naif Bayes yöntemleri kullanılarak yeni bir yöntem önerilmiştir. Bu yeni yöntemin tahmin başarısının Naif Bayes ve Lojistik Regresyon yöntemlerinden daha yüksek olduğu görülmüştür.

Haziran 2018, 57 sayfa.

Anahtar kelimeler: Makine öğrenmesi, kayıp müşteri analizi, yapay öğrenme

SUMMARY

M.Sc. THESIS

CUSTOMER CHURN PREDICTION BY MACHINE LEARNING

Melike GÜNAY

İstanbul University

Institute of Graduate Studies in Science and Engineering

Department of Computer Engineering

Supervisor : Assist. Prof. Dr. Tolga ENSARİ

Customer churn analysis is getting important for the firms that are working with memberships. Because most of the companies want to know which of the customers want to cancel the contract and convince them to continue using services from the company with new offers. Thus, they can use the financial opportunity of working with old customers rather than finding new ones. We analyze well-known machine learning methods that are logistic regression, Naïve Bayes, support vector machines, artificial neural networks and propose new prediction method. Because of the request from the companies, we tried to predict customer churn in telecommunication sector with machine learning techniques. When we searched about the customer churn analysis and found most popular machine learning methods. In this study, we calculated the performance of the methods. In addition to this, we proposed new approach to predict churn analysis by using Logistic Regression and Naive Bayes methods. As a result, we get better results than two methods separately.

June 2018, 57 pages.

Keywords: Machine learning, customer churn analysis, artificial intelligence

1. GİRİŞ

Son zamanlarda kayıp müşteri analizi, üyelik ya da abonelik yoluyla kazanç sağlayan firmalar için önemini artırmıştır. Özellikle sigortacılık, bankacılık ve telekomünikasyon sektörleri gibi alanlarda kayıp müşteri analizi ile üyelikten ayrılmak üzere olan müşterilerin önceden tahmin edilmesi firmalara bu müşteriler için yeni kampanyalar yaparak onları ayrılmadan önce ikna etmelerine olanak sağlayacağından bu alanda ki çalışmalar üzerinde yoğunluk artmıştır. Firmalar için yeni müşteri kazanmanın veya kaybolan müşterileri geri döndürmenin maliyeti üyelikten vazgeçmek üzere olan müşterileri firmanın üyesi olmaya devam etmesi için ikna etme çalışmalarının maliyetinden en az 9 kat fazla olduğu bilinmektedir. Pazar payının korunması için mevcut müşterilerin elde tutulması gerekliliği, kayıp müşteri analizi için çeşitli veri madenciliği ve makine öğrenmesi tekniklerinin geliştirilmesi ihtiyacının artmasına sebep olmuştur. Firmaların sektörde ki baskınlığı devamlı kullanıcıların sayısı ile orantılı olduğundan, kaybetme potansiyeli yüksek müşterilerin firmanın üyesi olmaya devam etmesi müşteri ilişkileri yönetiminin üzerinde çalıştığı önemli konulardan biri olmuştur. Kayıp müşteri analizinin ihtiyaç duyulduğu en önemli sektörler telekomünikasyon, bankacılık, perakendecilik, sigorta, oyun ve eğlence sektörü gibi alanlardır.

Müşteri kaybının engellenmesi için iyi bir sınıflandırma yöntemine ihtiyaç vardır. Bu metodun geliştirilmesi için geçmişte yapılan çalışmalarda Destek Vektör Makineleri, Lojistik Regresyon, Karar Ağaçları, Yapay Sinir Ağları, K-En Yakın Komşuluk, Naif Bayes, Rastgele Ormanlar gibi algoritmalar kullanılmıştır.

Bu tez kapsamında var olan tekniklerin analizini yaparak bu yöntemleri geliştirmeyi hedeflemekteyiz. Önereceğimiz yeni akıllı yöntem ile kayıp müşterilerinin tahmininde başarıyı yükselteceğimizi ön görmekteyiz.

2. GENEL KISIMLAR

Bu bölümde kayıp müşteri analizi ile ilgili daha önce yapılmış çalışmalar incelenmiştir. Son dönemde yapılan çalışmalara bakıldığında en çok kullanılan ve yüksek başarı gösteren yöntemler Lojistik Regresyon, Destek Vektör Makineleri, Naive Bayes, Yapay Sinir Ağları olarak belirlenmiştir.

2.1. YAPILAN ÇALIŞMALAR

Müşteri kaybı analizi ile ilgili yapılan son çalışmalardan biri 2017 yılının Mayıs ayında 'Makine Öğrenmesi Yöntemleriyle Müşteri Kaybı Analizi' başlığıyla geliştirilmiş ve Destek Vektör Makineleri, Naif Bayes ve Çok Katmanlı Yapay Sinir Ağları kullanılarak 3 model elde edilmiştir (Kaynar, Tuna, Görmez, & Deveci, 2017). Çalışma, firmanın ürünlerini kullanan ve kullanmaktan vazgeçen müşteriler olmak üzere 2 adet sınıftan ve 21 adet öznitelik ile 4667 adet müşteri bilgisi barındıran veri seti ile yapılmıştır. Eğitim ve test veri kümeleri %75 ve %25 oranlar ile rastgele oluşturulmuştur. Belirtilen 3 yöntem ile oluşturulan modellerden tahmin başarısı en yüksek olan %92.35 ile Yapay Sinir Ağlarıdır. İkinci ve üçüncü sırada ise sırası ile %87.15 ile Naif Bayes ve %77.89 ile Destek Vektör Makineleri bulunmaktadır. Ayrıca hassasiyet değeri en yüksek uygulama Naif Bayes ile elde edilmiştir. Sonuç olarak Yapay Sinir Ağları ve Naif Bayes beklenen başarılı sonucu verirken, Destek Vektör Makineleri beklenenden düşük sonuç vermiştir. Bu algoritmanın başarısızlığının veri setinde ki bazı öznitelikler ve örnek sayısının yetersizliği sebebiyle olduğu ön görülmüştür.

Telekomünikasyon sektöründe yapılmış bir çalışmada kayıp müşteri analizine başlamadan önce kullanılan veri seti üzerinde yapılan işlemlere dikkat çekilmiştir (Coussement, Lessmann, & Verstraeten, 2017). Çalışmada veri setinde yer alan devamlı ve ayrık değişkenlerin uygun formata çevrilmesinin ve yapılacak analize uygun olarak gösterilmesinin kayıp müşteri analizi sonuçları üzerinde ki etkisi vurgulanmıştır. Analizi yapılacak müşterilerin ve müşterilere ait özelliklerin doğru seçilmesi analizin performansını artırmaktadır. Kayıp müşteri analizi karmaşık bir işlem olarak ele alınmaktadır. Bu karmaşık yapı CRISP-DM tarafından işi anlama, veriyi anlama, veri ön işleme, modelleme, değerlendirme ve geliştirme olmak üzere 6 aşamaya ayrılmıştır. Bu çalışmada ise tamamlanması zaman gerektiren veri önişleme aşamasına odaklanılmıştır. Veri ön işleme işlemleri (VÖİ) veri indirgeme ve veri hazırlama aşaması olmak üzere ikiye ayrılmıştır. Veri indirgeme yöntemleri analizi etkileyecek veya etkilemeyecek

özelliklerin saptanarak elenmesi ve verinin boyutunun azaltılmasını hedef almaktadır. Veri hazırlama metotları ise orijinal değişkenlerin uygun formatlara dönüştürülmesi için kullanılmaktadır. VÖİ değer dönüşümü ve sunumu olarak iki adımda incelenmiştir. Değer dönüşümü bağımsız verilerin ayrık verilere dönüştürülmesi olarak tanımlanmıştır. Değer dönüşümü yöntemleri var ise eksik değerleri ayrı bir kategori olarak değerlendirmekte ve analizin performansını artırmaktadır. Çalışmanın eksik veriler ile ilgili kısmı Lojistik Regresyon gibi bu konu ile ilgili herhangi bir işlem bulundurmayan yöntemler için tercih sebebi olarak görülmektedir. Değer sunumu aşaması ise ilk aşamada elde edilen ayrık değişkenlerin formatlarının kayıp tahmini için uygunluğunu sağlamak için uygulanmaktadır. Değer dönüşümü yöntemleri kategorik / ayrık değişkenler ve devamlı / sürekli değişkenler için ayrıca uygulanmıştır. Kategorik değişkenler için karar ağaçları yöntemleri ile tahmin edilecek bağımsız değişkene uygun olarak dönüşüm gerçekleştirilmiştir. Sürekli değişkenler ise bilinen 3 ayrıştırma yöntemi kullanılarak ayrık verilere çevrilmiştir. İlk olarak eşit frekans ayırımı yöntemi ile sürekli değişkenler kategorik değişkenlere çevrilmiştir. İkinci aşamada eşit genişlik ayrımı yöntemi ile sürekli değişkenler eşit genişliğe sahip gruplara bölünmüştür. Son olarak ise karar ağaçları yöntemi uygulanarak sürekli değişkenler ayrık değişkenlere dönüştürülerek tekrar etiketlenmiştir. VÖİ adımlarının ikinci aşaması olan değerlerin sunumu aşamasının amacı ise değer dönüşümünde elde edilen değişkenlerin kayıp analizi için doğru formatta olduğundan emin olmaktır. Bu aşamada Kukla Kodlama (Dummy Coding) ve insidans değişimi teknikleri ile değişkenler analiz için uygun formata dönüştürülmüştür. Çalışmada kullanılan veri seti 30,104 müşterinin 156 kategorik 800 sürekli değişkeninden oluşmaktadır. Veri seti çalışmada eğitim, seçim ve test seti olmak üzere sırayla veri setinin %50,%20 ve %30'luk parçaları alınarak kullanılmıştır. Lojistik Regresyon modelinin kayıp müşteri analizinde ki başarısı veri ön işleme işlemlerinin etkisiyle birlikte ölçülmüştür. Sonuç olarak veri ön işleme işlemlerinin tahmin başarısını %34'e kadar çıkarabildiği gözlenmiştir. Bu aşamalarla birlikte uygulanan lojistik regresyon algoritması için klasik olarak kullanılan Yapay Sinir Ağları ve Destek Vektör Makinesi gibi metotlara nazaran daha hızlı olduğu görülmüştür.

Kayıp müşteri tahmininde kullanılmak üzere geliştirilmiş bir başka teknik ise 2017 yılında sunulmuştur (Amina, et al., 2017). Bu çalışmada kaba kümeler teorisi kullanılarak akıllı kural tabanlı karar verme mekanizması tekniği geliştirilmiştir. Bu teknik ile kayıp olan ve olmayan müşterilere karar vermede ilişkili olan kuralların başarı çıkarımı hedef alınmıştır. Tekniğin ölçümünde yorucu/ayrıntılı algoritma, genetik algoritma, örtme algoritması ve LEM2

algoritması kullanılmış ve kaba kümeler teorisi tabanlı yaklaşımın en iyi sonucu genetik algoritma ile birlikte verdiği görülmüştür. Genel olarak 4 algoritmanın sonucuna bakıldığında ise önerilen tekniğin ideal sonuçlar verdiği vurgulanmıştır.

Son zamanlarda kayıp müşteri tahmininde derin öğrenme teknikleri de uygulanmaya başlamıştır. P.Spanoudes ve T.Nguyen çalışmalarında farklı şirketlerde veya veri setlerinde genel olarak kullanılabilecek bağımsız soyut özellik vektörleri ile gözetimsiz öğrenme tekniği üzerinde durarak derin yapay ağlar algoritmasını kullanmışlardır (Spanoudes & Nguyen, 2017) . Tahmin sonuçlarına bakıldığında veri setinin boyutlarının küçültülmesinin örüntü tanıma aşamasında ne kadar önemli olduğu ispatlanmıştır. Sunulan veri soyut veri gösterimi popüler olarak kullanılan diğer mühendislik metotlarıyla karşılaştırılmış ve onlardan daha düşük bir performansa sahip olmadığı görülmüştür. Çalışmada derin öğrenme tekniği de uygulanmış ancak geliştirilmesi gerektiği vurgulanmıştır.

'Müşteri Davranışlarını Öz yineli Sinir Ağları (TSA) ile Anlamak' başlıklı çalışmada ise sinir ağlarının müşteri davranışlarını tahmin etmede ve modellemede uygun bir yöntem olduğu vurgulanmıştır (Lang & Rettenmeier, 2017). Öz yineli sinir ağları yönteminin daha önce uygulanmış diğer yöntemlere göre avantajları olduğu söylenmektedir. Bu yöntem daha önce denenmiş olan lojistik regresyon gibi vektör tabanlı algoritmalarla aynı veya daha iyi sonuçlar verdiği görülmüştür. Yöntemin diğer bir avantajı ise uygulamadan önce geniş bir veri önişleme aşamasına ihtiyaç duymamasıdır. Çalışmada çevrimiçi moda platformunu kullanan kullanıcıların verilerinden oluşan veri seti kullanılmıştır. Müşteri davranışları, web marketi ile etkileşimlerinden oluşan müşteri geçmişinin işlenmemiş halinden elde edilmiştir. Etkileşimleri müşterilerin marketi kullanım zamanları ve seçmiş oldukları ürün bilgileri oluşturmaktadır. Birçok makine öğrenmesi yöntemi sabit uzunluklu özellik vektörleri kullanarak lojistik regresyon, sinir ağları ve rastgele orman gibi modelleri kullanmaktadır. Bu modelleri gerçekleyerek müşteri davranışlarını tahmin edebilmek için müşteri geçmişlerinden sabit uzunluklu özellik setleri oluşturmak gerekmektedir. Ancak bu özellikleri doğru şekilde oluşturmak ve tahminde bulunmak zor ve uzun bir süreç gerektirebilmektedir. Öz yineli sinir ağları ile yapılan çalışmada bu zorlukların üstesinden gelindiği belirtilmiştir. Sinir ağları birçok farklı uzunlukta olan müşteri geçmişleri ile direk olarak çalışabilmektedir. Bu sebeple özellik çıkarımı / dönüşümü ve buna benzer zaman alan işlemlere ihtiyaç duyulmamaktadır. Geliştirilen yöntemde olay akışı ve zaman akışı olmak üzere iki farklı yaklaşım ile tahminlerde

bulunulmuştur. Sonuç olarak geliştirilen yöntem lojistik regresyon gibi yöntemlere nazaran daha başarılı tahmin sonuçları vermekle birlikte özellik çıkarımı için gereken ön işlemlere gerek duyulmadığından işlem zamanını da kısaltmıştır. Olay akışı tabanlı yapılan doğru tahminlerin başarısının zaman tabanlı yapılan analize göre daha düşük olduğu da görülmüştür.

Telekom sektöründe mobil reklamcılık ile ilgili yapılan diğer çalışmada da büyük veri analizi tabanlı bir sistem yapısı üzerinde Lojistik Regresyon algoritması kullanılmıştır (Zhang, Cheng, Yuan, Xu, Cheng, & Chao, 2016). Çalışmada çevrimiçi reklamcılık ile ilgili büyük veri analizinde karşılaşılan 2 güçlükten bahsedilmiş ve sunulan modelin bu güçlüklerin üstesinden geleceği söylenmiştir. Sözü edilen zorluklardan ilki kullanılan veri setlerinin yetersizliği diğeri ise kullanılacak makine öğrenmesi tekniğinin uygun şekilde seçilip uygulanmasıdır. Geliştirdikleri mimaride DSP, DMP, Adx ve SSP isimlerinde 4 önemli bölüm bulunmaktadır. Bu bölümler talep platformu (DSP), destek platformu (SSP), veri yönetimi platformu (DMP) ve reklam dönüşümü/basımı platformu (Adx) olarak görev almaktadır. Tasarladıkları model mobil kullanıcının cihazından bir uygulama açması ile başlamaktadır. Açılan uygulama reklam alanına sahip bir uygulama ise destek platformuna reklam için istek yollanmaktadır. İstek mesajını alan destek platformu reklam basımı platformu ile kullanıcı bilgilerini paylaşarak kullanıcıya uygun reklam bilgilerini elde etmektedir. Reklam basım platformu bu işlemi talep platformu ile iletişime geçerek tamamlayacaktır. Burada DSP bölümü bu işlem ile ilgili sadece konum, zaman, görüntüleme şekli ve mobil cihazın işletim sistemi gibi bilgiler ile çalışmaktadır. DSP bu bilgiler ile isteği aldığında kullanıcı kimliğini veri setinde ki bilgiler ile eşleştirmekte ve daha detaylı bilgiler elde etmeye çalışmaktadır. Daha sonra ise bu kullanıcıya ait en uygun reklam seçilmekte ve Adx platformu üzerinde SSP platformuna bu bilgi iletilmektedir. SSP ise bu reklamın kullanıcının cihazında görüntülenmesini sağlamaktadır. Bu yapıda anahtar bölüm DMP olarak görülmektedir, çünkü veri yönetimi kısmında birçok kullanıcıdan gelen veriler saklanmakta ve işlenmektedir. Bu bölümde üzerinde çalışılması gereken iki nokta bulunmaktadır. Birinci nokta yüksek kalitede verinin toplanması ve eşleştirilmesi, ikinci nokta ise bu verinin veri madenciliği yöntemleri ile işlenmesidir. Önerilen mimaride sunulan model çevrimiçi ve çevrimdışı olmak üzere 2 adımda gerçekleştirilmektedir. Çevrimdışı model uygun sınıflandırma algoritmasını geçmiş kullanıcı verileri üzerinde çalışarak bulmayı hedeflemektedir. Uygun algoritma bulunduğunda çevrimiçi olarak dinamik olarak gerçek zamanlı gelen veriler üzerinde uygulanmaktadır. Sınıflandırma algoritmaları kullanılmadan önce veri ön işleme aşamasında 1 aylık veri setinin yeterli kalitede olmadığı

gözlenmiştir. Modelde geçmiş çalışmalarda başarılı sonuç veren lojistik regresyon algoritması kullanılmış ve sonuç olarak kullanıcıların %85'inin kendilerine uygun olarak seçilen reklamlara tıkladığı ve hedeflenen kullanıcıya erişim başarısının %0,5 arttığı görülmüştür.

Telekom sektöründe müşteri sınıflandırması ile ilgili bir diğer çalışmada ise k-ortalamalar algoritması ile önerilen MQSFLA-k algoritması kullanılmıştır (Cheng & Cheng, 2016). MSQFLA-k algoritmasının performansı 3 farklı fonksiyon kullanılarak SFLA, BFO ve PSO gibi diğer algoritmalarla karşılaştırılarak ölçülmüştür. Müşteri sınıflandırması ile ilgili sunulan MSQFLA-k algoritmasının k-ortalamalar ve diğer algoritmalara nazaran avantajlı olduğu bilgisi verilmiştir. Bir diğer çalışmada ise müşteri kaybı tahmininde makine öğrenmesi tekniklerinin karşılaştırılması yapılmıştır (Vafeiadis, Diamantaras, Sarigiannidis, & Chatzisavvas, 2015). Bu çalışmada Yapay Sinir Ağları, Destek Vektör Makineleri, Karar Ağaçları, Naif Bayes ve regresyon analizi algoritmaları uygulanmış ve karşılaştırılmıştır. Analiz de 2 aşama bulunmaktadır. İlk olarak algoritmalar direk olarak veri seti üzerinde uygulanmıştır, daha sonra 2. aşama da ise aynı algoritmalara performansı yükseltici (boosting algorithm) bir başka yöntem uygulanarak analiz edilmiştir. Sonuç olarak f-skor sonuçlarına bakarsak en başarılı tahmini yükseltici yöntem uygulanmadan önce yapay sinir ağları, yükseltici yöntem ile destekçi vektör makineleri algoritması vermiştir.

2015 yılında müşteri kaybı analizi ile ilgili yapılmış bir tez çalışmasında ise bu alanda çokça kullanılan Karar Ağaçları ve Lojistik Regresyon gibi yöntemler anlatılmış ve özel bir bankanın müşteri bilgilerinden oluşan veri seti üzerinde Karar Ağaçları ve Lojistik Regresyon yöntemleri kullanılarak geliştirilen yöntem uygulanarak sonuçları analiz edilmiştir (Karaağaç, 2015). Belirtilen yöntemler uygulanmadan önce veri seçimi, verilerin birleştirilmesi, temizlenmesi ve dönüştürülmesi gibi veri ön işleme aşamalarından geçirilmiştir. Geliştirilen modeli test etmek için 'Back testing' yöntemi kullanılmıştır. Karışıklık matrisi ise doğru tahmin başarısını ölçmek adına sunulmuş ve sonuçlara göre yöntemin test veri seti üzerinde elde ettiği doğruluk oranı %89 olarak görülmektedir.

Bir başka tez çalışmasında ise kayıp müşteri tahmini Temel Bileşen Analizi ve Lojistik Regresyon yöntemi kullanılarak gerçeklemeye çalışılmıştır (Kanar, 2014).

3. MALZEME VE YÖNTEM

3.1. VERİ KÜMESİ

Bu tezde 2 adet veri kümesi kullanılmıştır. İlk veri kümesi daha önce yapılmış olan çalışmalarda en çok kullanılan yöntemleri denemek ve yeni bir tahmin yöntemi geliştirmek için kullanılmıştır. İkinci veri kümesi ise yeni tahmin yöntemini test etmek amacı ile çalışmaya dâhil edilmiştir. İki veri kümesi de telekomünikasyon firmalarına üye olan / kullanan müşterilerin verilerinden oluşmaktadır. Her iki veri kümesinde de sınıf etiketi kayıp var ya da kayıp yok yani 0 ya da 1 olarak belirlenmiştir. Veri kümelerinin %45 öğrenme kümesi için % 55'i ise test için kullanılmıştır.

3.1.1. Veri Kümesi-I

Bu veri kümesi 3333 adet müşterinin 20 adet özniteliğinden oluşmaktadır (bigml, 2017). Bu özniteliklerden 4 tanesi yazılı, 16 tanesi sayısal olarak bulunmaktadır. Özniteliklerin hiçbirinde eksik veri bulunmamaktadır. Veri kümesinde bulunan özelliklerin listesi ve türleri Tablo 3.1'de verilmiştir.

Tablo 3.1: Veri kümesi-I 'de bulunan öznitelikler ve türleri.

Öznitelik	**Türü**
State	Yazılı
Account	Sayısal
Area Code	Yazılı
Intl Plan	Yazılı
Vmail Plan	Sayısal
Nmb Vmaıl Mssg	Sayısal
Ttl Day Mın	Sayısal
Ttl Day Calls	Sayısal
Ttl Day Charge	Sayısal

Tablo 3.1 (devam) : Veri kümesi-I ’de bulunan öznitelikler ve türleri.

Ttl Eve Mın	Sayısal
Ttl Eve Calls	Sayısal
Ttl Eve Charg	Sayısal
Ttl Nght Mın	Sayısal
Ttl Nght Calls	Sayısal
Ttl Nght Charge	Sayısal
Ttl Intl Mın	Sayısal
Ttl Intl Calls	Sayısal
Ttl Intl Charge	Sayısal
Cust Serv Calls	Sayısal
Churn	Yazılı

3.1.2. Veri Kümesi-II

Bu veri kümesi 7043 adet müşterinin 20 adet özniteliğinden oluşmaktadır (IBM Watson Analytics, 2017). Veri kümesinde eksik veri bulunmamaktadır. Veri kümesinde bulunan özellikler ve türleri Tablo 3.2 de bulunmaktadır.

Tablo 3.2: Veri kümesi-II’de bulunan öznitelikler ve türleri.

Öznitelik	**Türü**
Gender	Yazılı
Partner	Yazılı
Dependents	Yazılı
Phone Service	Yazılı
Multiple Lines	Yazılı
Internet Service	Yazılı
Online Security	Yazılı
Online Backup	Yazılı
Device Protection	Yazılı

Tablo 3.2 (devam): Veri kümesi-II'de bulunan öznitelikler ve türleri.

Tech Support	Yazılı
Streaming TV	Yazılı
Streaming Movies	Yazılı
Contract	Yazılı
Paperless Billing	Yazılı
Payment Method	Yazılı
Churn	Yazılı

3.2. VERİ DÖNÜŞÜMÜ

Veri kümelerinin algoritmalarda kullanılabilmesi için özniteliklerin türlerinin uygun formatta olması gerekmektedir. Bu sebeple iki veri kümesinde de bulunan sayısal olmayan öznitelikleri sayısallaştırdık. Veri kümesi-I de bulunan sayısal olmayan özniteliklerden hesap numarası (account) ve alan kodu (area code) bilgileri analizle ilgisi olmadığından veri kümesinden çıkarılmıştır. Sayısal olmayan özniteliklerden 'churn' isimli öznitelik sınıf etiketini ifade etmektedir. Bu öznitelik evet/hayır gibi değer alarak müşteri kaybının olduğunu veya olmadığını göstermektedir. Bu öznitelik sayısal olarak 1 veya 0 olarak dönüştürerek veri kümesi –I i analize hazır hale getirdik.

Veri kümesi-II de bulunan özniteliklerin türü sayısal olarak tutulmamaktadır. Bu sebeple her özniteliği ayrık sayısal verilere Tablo 3.3 de gösterildiği gibi dönüştürülmüştür.

Tablo 3.3: Veri kümesi-II için veri dönüşümü tablosu.

Öznitelik	**Veri Tipi**	**Dönüşüm**
Gender	Yazılı	Female: 0 Male:1
Partner	Yazılı	No: 0 Yes: 1
Dependents	Yazılı	No: 0 Yes: 1

Tablo 3.3 (devam): Veri kümesi-II için veri dönüşümü tablosu.

Phone Service	Yazılı	No: 0 Yes: 1
Multiple Lines	Yazılı	No: 0 Yes: 1 No phone service: 2
Internet Service	Yazılı	No: 0 Fiber Optic: 1 DSL: 2
Online Security	Yazılı	No: 0 Yes: 1 No internet service: 2
Online Backup	Yazılı	No: 0 Yes: 1 No internet service: 2
Device Protection	Yazılı	No: 0 Yes: 1 No internet service: 2
Tech Support	Yazılı	No: 0 Yes: 1 No internet service: 2
Streaming TV	Yazılı	No: 0 Yes: 1 No internet service: 2
Streaming Movies	Yazılı	No: 0 Yes: 1 No internet service: 2
Contract	Yazılı	Month-to-month : 0 One year: 1 Two year : 2
Paperless Billing	Yazılı	No: 0 Yes: 1

Tablo 3.3 (devam): Veri kümesi-II için veri dönüşümü tablosu.

Payment Method	Yazılı	Mailed Check: 0 Electronic Check: 1 Bank Transfer(Auto.): 2 Credit Card(Auto) : 3
Churn	Yazılı	No: 0 Yes: 1

3.3. VERİ ÖNİŞLEME

Veri önişleme aşaması her sınıflandırma probleminde olduğu gibi kayıp müşteri analizi içinde önem taşımaktadır. Bu aşama seçilen veri kümesinde bulunan ve analize etkisi olmayacak veya negatif etkisi olacak özniteliklerin elenerek veri kümesinin boyutunun azaltılması ve veri kümesinde bulunan özniteliklerin yapılacak analizler için doğru formata çevrilmesi işlemlerinden oluşmaktadır. Veri kümesi-I ve veri kümesi-II' de yazılı olarak bulunan özniteliklerin sayısal veri türüne çevrilmesi gerekmektedir. Bununla birlikte veri kümesi – I de bulunan özniteliklerden en çok hangilerinin müşteri kaybı tahminine etkisi olduğu belirlenerek veri kümesinin boyutunu azaltma hedef edinilmiştir. Burada amaç zaman ve yer kısıtı olan çalışmalar için bu şekilde bir boyut azaltmanın kullanılıp kullanılamayacağının analizini yapmak ve boyut azaltmanın tahmin başarısında ki etkisini görmektir. Çoğu zaman d boyutlu bir veri kümesini k<d olacak şekilde k boyutlu bir veri kümesine indirgemek, bellek ve zaman açısından yarar sağlamakta, sistemi anlamayı kolaylaştırmaktadır. Öte yandan boyut azaltmak bazı durumlar için doğru tahmin oranını negatif yönde etkileyebilmektedir.

3.3.1. Pearson Katsayıları ile Öznitelik Seçimi

Çalışmanın bu bölümünde veri kümesinde bulunan özniteliklerden sınıf etiketini belirlemede en çok etkisi olan öznitelikleri saptayarak, etkisi olmayan öznitelikleri kayıp müşteri analizinde kullanmayarak zamandan ve yerden tasarruf edinmeyi amaçladık. Öznitelik sayısının azalmasının tahmin başarısında ki etkisi ve gerekliliği de analiz edilmesi gereken diğer konular olarak belirlenmiştir.

Veri kümelerinde bulunan ‘churn’ isimli öznitelik sınıf etiketini ifade etmekte ‘yes/no’ olarak iki değer almaktadır. Etiketin ‘yes’ olarak belirlenmesi müşteri kaybının olduğunu, ‘no’ olarak belirlenmesi müşteri kaybının olmadığını ifade etmektedir. WEKA kullanarak tamamladığımız bu bölümde var olan 19 adet özniteliğin sınıf etiketini temsil eden öznitelik ile olan ilişkisini ölçerek, en ilintili 10 özniteliği kullanmayı hedefledik. WEKA’da bulanan ‘CorrelationAttributeEval’ yöntemini kullandık. Bu yöntemle Pearson katsayıları ile sınıf etiketi ve diğer tüm öznitelikler ile arasında ki bağıntıyı ölçtük. Pearson katsayılarının hesaplanmasında kullanılan formül 3.2’de gösterilmektedir.

$$r = \frac{n(\sum_{i=1}^{n} x_i\, y_i) - (\sum_{i=1}^{n} x_i)(\sum_{i=1}^{n} y_i)}{\sqrt{\left[n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2\right]\left[n\sum_{i=1}^{n} y_i^2 - \left(\sum_{i=1}^{n} y_i\right)^2\right]}} \tag{3.2}$$

Bu formülde x ve y iki farklı özniteliği, n ise örnek sayısını temsil etmektedir (Statistics How To, 2017). Bu yöntem ile elde edilen r katsayılarının 1 ile -1 arasında değerler alması beklenmektedir. Katsayının 1’e yakın çıkması iki öznitelik arasında pozitif yönde uyum olduğunu, -1’e yakın çıkması ise tersi yönde bir uyum olduğunu göstermektedir. Katsayının 0 ya da 0’a yakın bir değer çıkması ise özniteliğin diğer öznitelik üzerinde pozitif ya da negatif bir etkisi olmadığını göstermektedir. Katsayıların derecelendirildiği aralıklar Tablo 3.4’de gösterilmektedir.

Tablo 3.4: Pearson katsayılarının dereceleri.

Derece	**Değer Aralığı (d)**
Mükemmel	d = ~1
Yüksek Derece	0.5 < d < 1
Orta Derece	0.3 < d < 0.49
Düşük Derece	d < 0.29
Bağıntı Yok	d = 0

Değeri en yüksek katsayılara göre seçilmiş 10 öznitelik, en yüksek katsayıya sahip olandan en düşüğüne göre Tablo 3.5 de gösterilmiştir.

Tablo 3.5 : Pearson katsayıları ile seçilen 10 öznitelik.

Pearson Katsayısı	Öznitelik
0.25985	International Plan
0.20875	Customer Service Calls
0.20515	Total Day Minutes
0.20515	Total Day Charge
0.10215	Voice Mail Plan
0.0928	Total Eve Minutes
0.09279	Total Eve Charge
0.08973	Number Vmail Messages
0.06826	Total Intl Charge
0.06824	Total Intl Minutes

3.3.2. Temel Bileşen Analizi (TBA) ile Öznitelik Çıkarımı

Çalışmalarda temel bileşen çözümlemesi veya analizi olarak adı geçen analiz genellikle çok boyutlu uzaylarda boyut azaltmak için kullanılmaktadır. Bu analizde amaç veri kümesindeki değişintiyi temsil eden daha az sayıda vektör ile veri kümesini temsil etmektir. Bu vektörler iki özellik arasında ki bağlantıyı da barındırmaktadır. Özellikler arasında ki bağlantının tutulması boyutu azaltılmış veri kümesinden geriye dönülerek orijinal veri kümesinin elde edilmesini de sağlamaktadır. Temel bileşen analizi gözetimsiz bir yöntem olarak bilinmektedir çünkü yöntem sınıf etiketi kullanarak çalışmamaktadır.

Temel bileşen analizinde amaç aralarında korelasyon bulunan verilerden aralarında korelasyon bulunmayan vektörler oluşturarak veriyi daha az boyutlarda ifade etmektir. Bu amaçla veri kümesinin doğrusal bileşenlerini elde etmek için kovaryans matrisi kullanılır. Kovaryans matrisinin öz değerleri ve öz vektörleri kullanılır. Kovaryans matrisi çok boyutlu bir matrisi genelleştirmek için kullanılmaktadır. Burada amaç veri boyutu düşürürken, bilgi kaybını engellemek için değişintiyi (varyans) yüksek tutmaktır.

Veri kümesi-I için veri boyutunu azaltmak için Matlab kullanılarak veri temel bileşenlerine ayrılmıştır. Veri kümesini en iyi açıklayan 10 bileşen kullanılmış ve verinin boyutu 10x3333 boyutuna düşürülmüştür.

3.4. YÖNTEMLER

Çalışmanın bu bölümünde tüm veri kümesi üzerinde ve öznitelik çıkarımı ve öznitelik seçimi aşamalarından sonra elde ettiğimiz veri kümeleri üzerinde Lojistik Regresyon, Naive Bayes, Destek Vektör Makineleri, Yapay Sinir Ağları gibi kayıp müşteri analizinde sıkça tercih edilen gözetimli öğrenme yöntemleri kullanarak müşteri kaybını tahmin etmeye çalıştık. Bu bölümde sözü geçen yöntemlerin doğru tahmin başarıları, performans ve hız ölçümleri gerçekleştirilmiştir. Ayrıca son bölümde yeni bir yaklaşım olarak geliştirdiğimiz çarpımsal olasılık yönteminin de karşılaştırmalı sonuçları verilecektir.

3.4.1. Lojistik Regresyon Yöntemi

İlk yöntem olarak geçmişte en çok kullanılan lojistik regresyon yöntemini denedik. Lojistik regresyon sınıf özelliğinin kategorik ve ikili veya çoklu durumlarda diğer özellikler ile arasında ki ilişkiye bağlı olarak matematiksel bağıntıyı kullanan bir analizdir. Bu analizde oluşan modele göre sınıf tahmini yapmaya çalışılmaktadır. Oluşan modelde bağımsız değişkenler ve bağımlı değişken olan sınıf kategorisi kullanılmakta ve aralarında ki ilişkiyi tanımlayan bir model oluşturulmaktadır. Lojistik regresyon iki değer alabilen bağımlı değişkene sahiptir. Bu değişken evet/hayır, olumlu/olumsuz gibi yani 0 ya da 1 gibi değerler alabilen bir değişkendir. Lojistik regresyon ve doğrusal regresyon arasında ki fark bağımlı değişkenin türüdür. Doğrusal regresyonda bağımlı değişken sürekli olabilirken lojistik regresyonda bağımlı değişken kategoriktir. Doğrusal regresyonda bağımlı değişken ile bağımsız değişkenler arasında ki ilişkinin doğrusal olduğu bilinmekte ve amaç bağımlı değişkeni tahmin etmektir. Lojistik regresyon ise bağımlı değişken ve bağımsız değişkenler arasında ki ilişkinin doğrusal olmadığı durumlarda kullanılmakta ve hedef bağımlı değişkenin olasılığını hesaplamaktır. Lojistik regresyon ikili ve çoklu olmak üzere iki farklı şekilde uygulanabilmektedir. İkili lojistik regresyon bir bağımsız değişken ve bir bağımlı değişken kullanılarak, çoklu lojistik regresyon birden çok bağımsız değişken ve bir bağımlı değişken ile çalışmaktadır.

Bu analiz denklem 3.3 de gösterildiği gibi olasılık teoremi üzerinden yapılmaktadır (Bayrak).

$$Logit[p(x)] = \frac{1}{1 + e^{-(\mathrm{b0+\ b1x1++\ b2x2+\cdots+\ bixi})}} \tag{3.3}$$

Veri kümesinde önişleme aşamasında seçtiğimiz bağımsız 10 özellik ve bağımlı sınıf özelliği Matlab kullanılarak lojistik regresyon analizinden geçirilmiştir. Matlab'da bulunan 'glmfit' fonksiyonu ile 10 bağımsız değişken ve 1 bağımlı değişken için toplam 11 adet katsayı (b_i) üretilmiştir. Glmfit fonksiyonu binom dağılımını kullanmaktadır. Binom dağılımı ayrık olasılık dağılımı olarak bilinmektedir (WikiZer, 2017). Binom dağılımı kullandığımız bağımlı değişken ikili değerlerden oluştuğu için uygun görülmüştür. Katsayılar elde edildikten sonra denklem 3.3 de görülen formül Matlab kullanılarak gerçeklenmiştir.

Veri kümesinde, bağımlı sınıf değişkeni müşteri kaybı varsa 1, yok ise 0 değerini almaktadır. Çalışmada Logit fonksiyonu sonucunda oluşan değerler bağımlı değişkene ait olasılıklar olduğundan 0.5 değerinden küçük ise müşteri kayıp yok yani 0, 0.5 değerinden büyük ise müşteri kaybı var yani sınıf değişkeni 1 olarak tanımlanmıştır. B_n katsayıları veri kümesi-I' den ilk 1500 adet müşterinin verileri kullanılarak üretilmiş ve kalan 1833 adet müşteri verisinde kullanılarak sınıf etiketi belirlenerek test edilmiştir. Analiz sonucunda 1833 adet müşterinin üyeliklerini devam edip etmeyeceği tahmini %84.34 oranla doğru yapılmıştır. Tablo 3.6 de algoritmaya ait karışıklık matrisi verilmiştir.

Tablo 3.6: Veri kümesi-I' de özellik seçiminden sonra Lojistik Regresyon yönteminin karışıklık matrisi.

	Öngörülen Sınıf		
Gerçek Sınıf		Müşteri Kaybı Var	Müşteri Kaybı Yok
	Müşteri Kaybı Var	34	259
	Müşteri Kaybı Yok	27	1512

Bu yöntemi veri kümesi-I kullanılarak temel bileşen çözümlemesi ile özellik çıkarımından(TBA) sonra tekrar uygulayarak tahmin başarısını ölçtük. Bu durumda doğru tahmin başarısının %83.73 olduğunu saptadık. Bu analizin karışıklık matrisi tablo 3.7'da verilmektedir. Özellik seçimi ve çıkarımı aşamalarından sonra uygulanan yöntemin doğru tahmin oranlarına baktığımızda çok büyük bir fark bulunmamaktadır. Ayrıca veri kümesi-I ile boyut azatlımı yapmadan, orijinal hali ile Lojistik Regresyon yöntemi kullanıldığında doğru tahmin başarısının %84.50 olarak belirlenmiştir.

Tablo 3.7 : Veri kümesi-I' de özellik çıkarımından sonra Lojistik Regresyon yönteminin karışıklık matrisi.

	Öngörülen Sınıf		
Gerçek Sınıf		Müşteri Kaybı Var	Müşteri Kaybı Yok
	Müşteri Kaybı Var	22	271
	Müşteri Kaybı Yok	12	1527

3.4.2. Naif Bayes Yöntemi

Naif Bayes sınıflandırma problemlerinde oldukça sık kullanılan olasılık tabanlı bir gözetimli öğrenme algoritmasıdır. Naif Bayes ile sınıflandırma yönteminde kullanılacak her özelliğin birbirinden bağımsız olma şartı vardır. Bayes karar teoremi verinin hangi sınıfa ait olabileceği olasılık değerleri karşılaştırılarak oluşturulmuştur. Örneğin veri1'in 1 numaralı sınıfa ait olma olasılığı 2 numaralı sınıfa ait olma olasılığından büyük ise veri1 in 1 sınıfa ait olduğu kararı verilir. Denklem 3.4 de bu karşılaştırma işlemini anlatan formül verilmiştir (Çayıroğlu). S: Ayrıştırılacak sınıflar kümesini, x: nicelik vektörünü temsil etmektedir.

$$P(S_i)\prod_{k=1}^{L} P(x_k|S_i) > P(S_j)\prod_{k=1}^{L} P(x_k|S_j) \tag{3.4}$$

Denklem 3.4'de verilen ifade doğru ise veri i sınıfına, yanlış ise j sınıfına aittir sonucuna varılır. En çok kullanılan sınıflandırma algoritmalarından olan Naif Bayes algoritması, önce ki analizlerde kullanılan öğrenme kümesi ve test kümesi ile %63.99 oranında doğru tahminde bulunmuştur. Tablo 3. 8 de Naif Bayes sınıflandırıcısına ait karışıklık tablosu verilmiştir.

Tablo 3.8 : Veri Kümesi-I ile özellik seçimi sonrası Naive Bayes yönteminin karışıklık matrisi.

	Öngörülen Sınıf		
Gerçek Sınıf		Müşteri Kaybı Var	Müşteri Kaybı Yok
	Müşteri Kaybı Var	269	24
	Müşteri Kaybı Yok	635	904

Naif Bayes yöntemi ikinci olarak temel bileşen analizi sonrası özellik çıkarımından sonra tekrar kullanılmıştır. Bu analize ait karışıklık matrisini Tablo 3.9'de bulabilirsiniz.

Tablo 3.9 : Veri Kümesi-I ile özellik çıkarımı sonrası Naif Bayes yönteminin karışıklık matrisi.

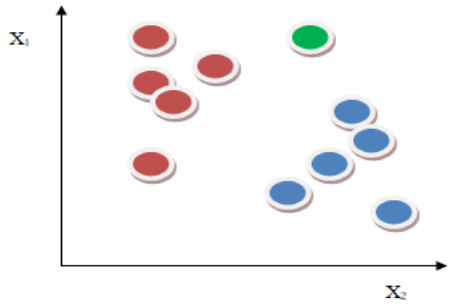
	Öngörülen Sınıf		
Gerçek Sınıf		Müşteri Kaybı Var	Müşteri Kaybı Yok
	Müşteri Kaybı Var	124	169
	Müşteri Kaybı Yok	97	1442

Özellik çıkarımı ile elde ettiğimiz %64 lük doğru tahmin başarısının % 84.99 olduğunu gördük.

3.4.3. Destek Vektör Makineleri

Destek vektör makineleri gözetimli öğrenme tekniklerinden biridir. Bu yöntemin yüksek boyutlu uzaylarda başarılı sonuçlar verdiği bilinmektedir. Bu algoritma ile eğitim kümesi kullanılarak oluşturulan model test kümesinde denenerek algoritmanın başarısı ölçülmektedir. Destek vektör makineleri istatistiksel öğrenme teorisine bağlı olarak çalışmaktadır (Ayhan & Erdoğmuş, 204). İstatiksel öğrenme teorisi Vapnik Chervonenkis(VC) teorisi üzerine geliştirilmiştir. Teoriye göre değişkenler arasında ki bağıntı bilinmemektedir ve amaç en uygun modeli bulmaktır. Teoride sözü geçen VC boyutu algoritmanın öğrenme kapasitesi üzerinde etkili bir özelliktir. Destek vektör makineleri olasılık tabanlı tahmin değil evet/hayır gibi nokta tahmini yapmaktadır.

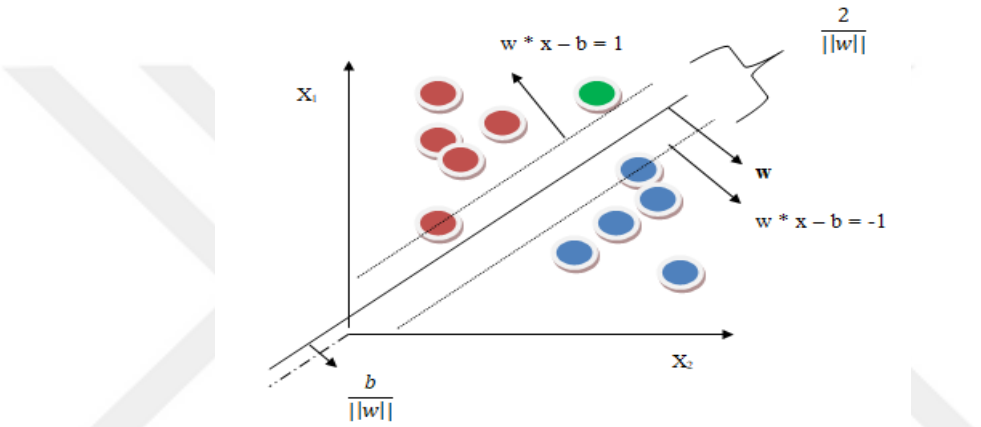
Destek vektör makinelerinin çalışma prensibini genel anlamda açıklayalım. Şekil 3.1 de gösterilen örnek uzayda bulunan X_1 ve X_2 iki farklı özniteliği temsil etmektedir. Düzlemde bulunan mavi ve kırmızı veriler sınıf etiketleri belli ve öbeklenmiş verilerdir.



Şekil 3.1: Örnek düzlemde iki özniteliğe göre öbeklenmiş veri kümesi**.**

Burada yeşil renkte gösterilen verinin sınıf etiketi henüz belirlenmemiştir. Destek vektör makineleri ile kırmızı ve mavi sınıfa ait veriler kullanılarak iki sınıfı birbirinden ayıracak, hata

oranı en düşük olacak şekilde tanımlayabilecek bir vektör ya da daha çok sınıf etiketi barındıran veri kümeleri için birden çok vektör oluşturmaktadır. Önemli olan bu vektörün sınıflara ayrılmış verilere olan uzaklığının en uygun şekilde belirlenebilmesidir. Vektörün oluşturduğu üst düzleme göre sınıflandırılması gereken verilerin sınıf tahmini yapılmaktadır. Örnekte verdiğimiz iki sınıfı birbirinden ayıran vektör w olsun. w vektörünün her sınıfın bu vektöre en yakın elemanına olan uzaklığının eşit veya hatanın en az olduğu noktaya göre belirlenmesi önemli olan diğer konudur. Şekil 3.2'de destek vektör makineleri yöntemi ile belirlenmiş vektör ve sınıf belirlemede kullanılan diğer parametreler gösterilmiştir.



Şekil 3.2: Destek vektör makinelerinde kullanılan terimlerin oluşan üst düzlemde gösterimi.

Şekil 3.2 de oluşan w vektörü, iki sınıfa ait en yakın verilerden geçen iki paralel vektörün tam ortasında konumlanmaktadır. Oluşan vektörün düzlemin orta noktasına olan uzaklığı ise $\frac{b}{||w||}$ olarak tanımlanmaktadır. Bu durumda sınıf etiketi belirlenmesi istenen veri denklem 3.5 de verilen eşitsizliğe göre tahmin edilebilir. Burada sınıf etiketi y_i olarak gösterilmektedir.

$$\vec{w} * \vec{x} - b \geq 1, \qquad eğer\ y_i = 1 \tag{3.5}$$

$$\vec{w} * \vec{x} - b < -1, \qquad eğer\ y_i = -1$$

Veri kümesi-1 ile öznitelik seçiminden sonra elde ettiğimiz eğitim kümesi ve öğrenme kümesi kullanılarak Destek Vektör Makineleri yöntemi kullanıldığında doğru tahmin başarısı %77.25 olarak belirlemiştir. Aynı veri kümesi ile öznitelik çıkarımından sonra elde ettiğimiz başarı ise %70.31 olarak belirlenmiştir. İlgili karışıklık matrislerine Tablo 3.10 ile Tablo 3.11'de görebilirsiniz.

Tablo 3.10: Veri Kümesi-I ile özellik seçimi sonrası Destek Vektör Makineleri yönteminin karışıklık matrisi.

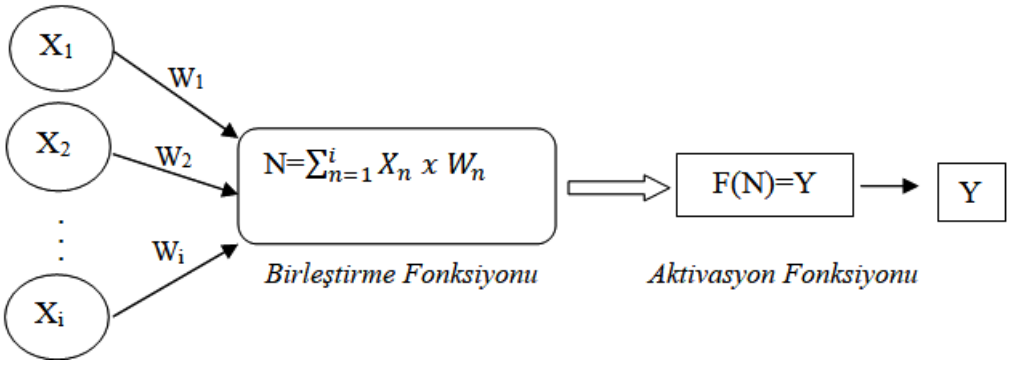
	Öngörülen Sınıf		
Gerçek Sınıf		Müşteri Kaybı Var	Müşteri Kaybı Yok
	Müşteri Kaybı Var	224	69
	Müşteri Kaybı Yok	347	1192

Tablo 3.11: Veri Kümesi-I ile özellik çıkarımı sonrası Destek Vektör Makineleri yönteminin karışıklık matrisi.

	Öngörülen Sınıf		
Gerçek Sınıf		Müşteri Kaybı Var	Müşteri Kaybı Yok
	Müşteri Kaybı Var	207	86
	Müşteri Kaybı Yok	457	1081

3.4.4. Yapay Sinir Ağları

Yapay sinir ağları ilhamını beyinin çalışma mekanizmasından almıştır. İnsan beyni bilgisayardan farklılıklar göstermektedir. Bilgisayarlar tek bir işlemciye sahipken, insan beyninin birden çok işlemcinin yerini tutabilecek sinir hücrelerine sahip olduğu ve bu hücrelerin paralel olarak çalıştığı bilinmektedir. Bununla birlikte beynin yapısında bulunan işlemci birimleri bilgisayarda ki işlemciden daha yavaş çalışmaktadır ancak insan beyninin çalışma yapısını farklı kılan hızı değil işlemci birimlerin yüksek bağlantı sayılarıdır. Bir sinir hücresinin diğer sinir hücreleri ile yaklaşık 10^4 adet bağlantısı bulunmaktadır. Bilgisayar sistemlerinde işlemci aktif, hafıza ayrı ve pasif olarak bulunmaktadır. Diğer taraftan beyinde ise işlemci ve hafıza birlikte ağ üzerinde dağıtılmış olarak kullanıldığı kabul edilmektedir. Sinyaller bir sinir hücresine sinir uçları aracılığı ile gelerek hücrenin çekirdeğinde toplanır. Toplanan sinyaller aksonlara iletilir ve burada işlenerek bağlantıları aracılığı ile diğer sinir hücrelerine aktarılırlar. Bu yapıdan yola çıkarak tek komutlu birden çok işlemciye ve her bir işlemciye ait farklı parametrelerin bulunduğu belleklerin kullanıldığı sistemler geliştirilmeye başlanmış ve yapay sinir ağları adı verilmiştir. Yapay sinir hücreleri girdiler, bu girdilerin ağırlıkları, birleştirme fonksiyonu, aktivasyon fonksiyonu ve bu fonksiyonun çıktılarından oluşmaktadır.



Şekil 3.3 : Yapay sinir hücresinin yapısı.

Şekil 3.3'de yapay bir sinir hücresi matematiksel olarak gösterilmiştir. Şekilde başlangıç girdisi olarak i adet x özelliği gösterilmektedir. Birleştirme fonksiyonu her X girdisinin kendisine ait W ağırlığı ile çarpıp toplayarak N toplamını oluşturmaktadır. Birleştirme fonksiyonu toplam şeklinde alınabileceği gibi Tablo 3.12 de gösterildiği gibi farklı yöntemlerle de kullanılabilmektedir.

Tablo 3.12 : Toplama fonksiyonu için kullanılan popüler yöntemler.

Toplam	$N=\sum_{n=1}^{i} X_n * W_n$
Çarpım	$N=\prod_{n=1}^{i} X_n * W_n$
Maksimum	$N=Max(X_n * W_n)$
Minimum	$N=Min(X_n * W_n)$
Çoğunluk	$N=\sum_{n=1}^{i} Sgn(X_n * W_n)$
Kümülâtif Toplam	$N= N(eski) + \sum_{n=1}^{i} X_n * W_n$

Birleştirme fonksiyonu sonucunda oluşan toplam N, aktivasyon fonksiyonuna girdi olarak verilmektedir. Aktivasyon fonksiyonu bu girdinin hücreye karşılık olarak vereceği çıktıyı hesaplamaktadır. Bu fonksiyonun özelliği doğrusal olmaması ve türevinin kolay alınabilir olmasıdır. Özellikle geri beslemeli yapar sinir ağlarında türevi zor alınan aktivasyon fonksiyonlarının alınması sistemi yavaşlatmaktadır. Aktivasyon fonksiyonu için genel olarak Sigmoid fonksiyonu kullanılırken, doğrusal fonksiyon, adım fonksiyonu, tanjant hiperbolik fonksiyonu, eşik değer fonksiyonu ve sinüs fonksiyonları da tercih edilmektedir. Sigmoid fonksiyonu denklem 3.6 de gösterildiği gibi doğrusal olmayan, türevi alınabilir sürekli bir fonksiyondur ve 0 ile 1 arasında değer almaktadır

$$F(N) = \frac{1}{1 + e^{-N}} \tag{3.6}$$

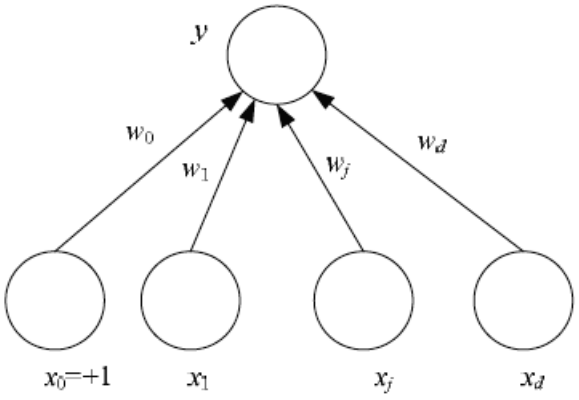
Aktivasyon fonksiyonundan çıkan her değer, o sinir hücresinin çıktısı olarak değerlendirilmektedir.

Yapay sinir ağlarında her işlemci bir sinir hücresine, yerel parametreler bağlantı ağırlıklarına denk gelmekte ve bu yapıya sinir ağı denilmektedir. Makine öğrenmesinin başladığı noktada işlemlerin işlemciler arasında paylaştırılması ve yerel parametrelerin ağırlığının saptanması yer almaktadır.

Sinir ağlarında temel işlemci elemana algılayıcı adı verilmektedir. Her algılayıcı dışarıdan veya başka bir algılayıcının çıktısından oluşan girdiler almaktadır. Her girdi X_j ve her algılayıcı çıktısı Y ile gösterilirse X_j ve Y arasındaki bağıntı bağlantı ağırlıkları W_j cinsinden denklem 3.7' de gösterilmiştir.

$$Y = (\sum_{j=1}^{d} W_j * X_j) + w_0 \quad w_j \in R \;\; ve \; x_j \in R \,, \; j = 1, \dots, d \tag{3.7}$$

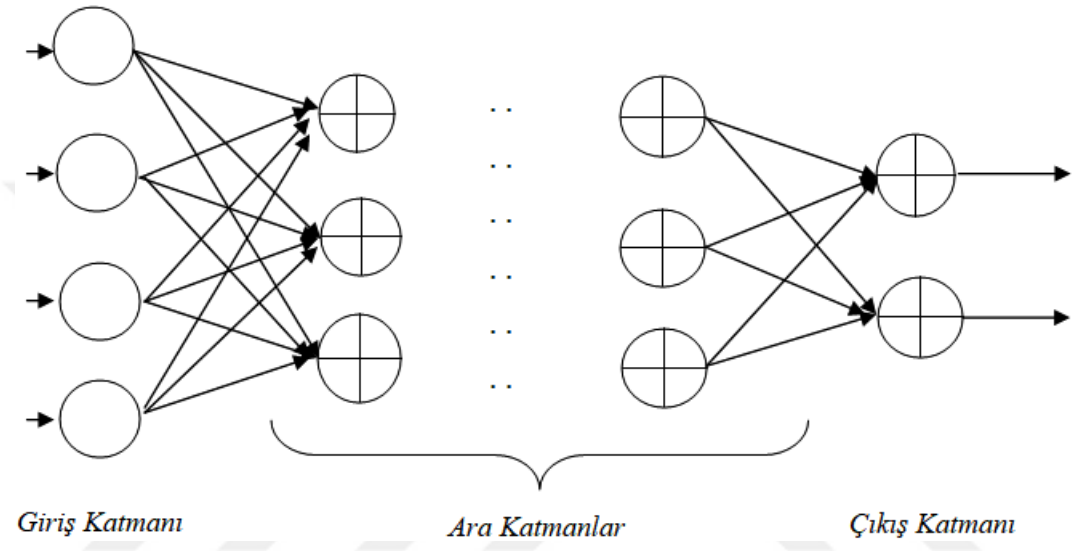
Denklem 3.7'de gösterilen w_0 modeli genelleştirmek için eklenmiş bir kesme değerdir ve w_0 genellikle 1 olarak alınmaktadır. Sinir ağlarında kullanılan bir algılayıcının genel yapısı Şekil 3.4 de gösterilmiştir (Alpaydın, Çok Katmanlı Algılayıcılar, 2013). Bir yapay sinir ağı birden çok algılayıcı ve katman bulundurmaktadır.



Şekil 3.4 : Yapay sinir ağlarında kullanılan algılayıcının gösterimi.

Tek sinir hücresinin yapısından ağın yapısına geçiş yaptığımızda ise ağın giriş katmanı, gizli katmanlar ve çıkış katmanı olarak 3 yapıdan oluştuğunu görmekteyiz. Giriş katmanı dışarıdan gelen girdilerin yapay sinir ağına giriş yaptığı katmandır. Bu katmana gelen girdi sayısı kadar

sinir hücresi oluşturulur ve genelde hiçbir işleme tabii tutulmadan alt katmanlara iletilir. Giriş katmanından iletilen veriler gizli katman adı verilen ara katmanlara gelir. Sinir ağının yapısına göre tek katmanlı olabildiği gibi birden fazla ara katmanı da olabilmektedir. Ara katmanlarda bulunan hücre sayısı da farklılık gösterebilmektedir. Ara katmanlardan gelen bilgiyi son olarak işleyerek sonucu üreten katman ise çıkış katmanı olarak adlandırılmıştır. Yapay sinir ağlarının genel yapısı bahsedilen üç katman ile birlikte Şekil 3.5' de gösterilmektedir.



Şekil 3.5: Yapay sinir ağının yapısı.

3.4.4.1. Yapay Sinir Ağlarının Çeşitleri

Yapay sinir ağlarında modeli oluştururken herhangi bir standart bulunmamaktadır. Bir ağ, burada bulunan algılayıcıların dizilimi, algılayıcıların ağırlıkları hesaplanırken kullanılan yöntemler ve öğrenme zamanına göre sınıflandırılmaktadır (Çayıroğlu).

Ağın içerdiği nöronların birbiri ile olan ilişkisine göre yapay sinir ağları ileri ve geri beslemeli olmak üzere 2 sınıfa ayrılmaktadır. İleri beslemeli ağlarda bir katmanın çıktısı ancak kendinden sonra ki katmanda bulunan bir başka algılayıcıya girdi olabilmektedir. Ağa gelen bilgiler sırasıyla tüm katmanlardan geçerek son çıktıyı oluşturmaktadır. Geri beslemeli ağlarda ise bir algılayıcının çıktısı, yalnız kendinden sonra ki katmanlara değil, kendi katmanına veya önce ki katmanlara girdi olarak kullanılabilmektedir. Bu sebeple yapay sinir ağları dinamik bir yapı oluşturmaktadır.

Öğrenme yöntemlerine göre de 3'e ayrılan Yapay Sinir Ağları gözetimli, gözetimsiz ve destekleyici öğrenme olarak değerlendirilmektedir. Gözetimli öğrenme uygulanan ağlarda, sisteme girdiler ile birlikte olması gereken çıktılar yani beklenen değerler de verilir. Sistemin ürettiği çıktılar ile beklenen değerler arasında ki hata hesaplanarak sistemin ağırlık katsayıları güncellenir. Gözetimsiz öğrenmede ise ağa sadece girdiler verilir ve benzerlik gösteren veriler kendi aralarında sınıflandırılır. Ağırlık katsayıları aynı özellikte olan verileri sınıflandırabilecek şekilde güncellenir. Destekleyici öğrenme yöntemi ise ağın her kullanıldığında elde edilen çıktıya bakarak sonucun iyi ya da kötü olduğu bilgisi üretilir. Bu bilgiye göre ağda güncelleme yapılarak model iyileştirilmeye çalışılır.

Yapay sinir ağlarında öğrenme zamanı önemli bir kıstas olarak belirlenmiştir. Modelin statik veya dinamik olarak öğrenmesi kullanım şeklini etkilemektedir. Statik öğrenme yönteminde sistem bir miktar veri ile eğitilir, eğitim aşaması bittikten sonra ise yaratılan model kullanılabilir. Eğitim aşaması bittikten sonra ağda kullanılan ağırlıklarda bir güncelleme yapılmaz. Dinamik öğrenme yönteminde ise ağın eğitilmesi kullanıldığı sürece devam eder. Her yeni veri ile ağırlıklarda güncelleme yapılarak öğrenme başarısı artırılmaktadır.

3.4.4.2. Yapay Sinir Ağlarının Eğitilmesi

Yapay sinir ağlarında daha önce de belirttiğimiz gibi sisteme giren veriler ara katmanlardan geçirilerek birleştirme ve aktivasyon fonksiyonları ile hücrenin çıktısını oluşturur. Bu yapıda kullanılan katsayılar başlangıçta rastgele atanan değerler olduğundan, beklenen çıktı ile elde edilen çıktı arasında ki hata yok sayılabilecek kadar az olana kadar bu katsayıların güncellenmesi gerekmektedir. Ağın eğitilmesi ve öğrenme bu katsayıların iyileştirilmesi ile mümkündür. Yapay sinir ağı bir miktar veri ile eğitildikten sonra, daha önce kullanılmayan veriler ile test edilmektedir. Öğrenme ve test aşamalarında kullanılacak verinin boyutu için sabit bir oran ya da sayı olmamak ile birlikte genellikle verinin % 80'i eğitim %20'si test için kullanılmaktadır. %60- %40 veya %50-%50 oranlarda bölünerek eğitim ve test kümelerinin oluşturulduğu deneylerde bulunmaktadır. Yapay sinir ağlarının eğitilmesi şekil 3.6 de gösterildiği gibi 9 bölüme ayrılarak incelenebilir. Şekilde gösterildiği gibi ilk olarak kullanılacak veri kümesi belirlenmeli, örneklerin toplanması ile ilgili araştırma yapılmalıdır. Kullanılacak veri setine göre ağın topolojik yapısı, girdi sayısı, ara katman sayısı ve her katmanda oluşturulacak algılayıcı sayısına karar verilebilir. Yapıya karar verildikten sonra öğrenmenin gerçekleştirilmesi için gerekli öğrenme katsayısı, birleştirme ve aktivasyon

fonksiyonları ve momentum katsayısı gibi sistem parametreleri belirlenir. Öğrenme ve momentum katsayıları ağırlık katsayılarının güncellenmesi için kullanılan önemli parametrelerdir. Algılayıcının eğitilmesi için yeni ağırlık katsayılarının oluşturulduğu formül denklem 3.8' de gösterilmiştir.

$$w_j^{k+1} = w_j^k + \lambda(y_i - \hat{y}_i^k)x_{ij} \tag{3.8}$$

Denklemde j.algılayıcı için yeni katsayı w_j^{k+1} , eski katsayı w_j^k, öğrenme katsayısı λ, algılayıcıya verilen i.girdi x_{ij}, ve beklenen çıktı ve elde edilen çıktı arasındaki fark yani hata kullanılmaktadır. Öğrenme katsayısı genellikle 0 ile 1 arasında, çoğunlukla 0.2 çevresinde alınmaktadır. Bu katsayı gerçekleşen değişikliğin büyüklüğünü temsil eder ve hata azaldığında katsayının değeri sabit olarak artırılır. Aynı şekilde hata artarsa geometrik olarak azaltılır (Alpaydın, Öğrenme Yordamları, 2013). Öğrenme oranı ya da öğrenme çarpanı olarak bilinen λ değeri çok küçükse beklenen değere olan yakınsama için çok sayıda öğrenme katmanına gereksinim olacaktır. Ağırlık değerleri ek olarak girdiye bağlı olarak da değişebilmektedir. Eğer girdinin değeri sıfıra yakınsa ağırlık katsayısına etkisi de küçük olacaktır. Girdinin değeri büyüdükçe ağırlık değerine etkisinin daha büyük olması beklenmektedir.

Yapay sinir ağlarının eğitilmesi için daha önce toplanan veri kümesinden eğitim kümesi elde edilir. Elde edilen eğitim kümesinde ki her girdi o sisteme verilir, elde edilen çıktılar kullanılarak ağırlık değerleri güncellenir. Eğitim kümesi kullanılarak karar verilen katsayılar ile modelin son hali oluşturularak, test kümesinde ki veriler ile sistemin doğru tahmin başarısı ölçülür. Sonuç olarak elde edilen başarı oranı yeterli değil ise sistem baştan çalıştırılarak rastgele atanan başlangıç ağırlıkları veya öğrenme katsayısı değiştirilerek sistemin başarısı yükseltilebilir. Ayrıca zaman açısından sistemin performansının yükselmesi için başlangıç ağırlıklarının lojistik regresyon yöntemi ile belirlenebileceği bilinmektedir.

Algılayıcıdan elde edilen y değerinin, pozitif veya negatif olmasına bağlı olarak denklem 3.9'de gösterildiği gibi sınıf etiketleri belirlenebilmekte veya 0 yerine farklı sınır değerler kullanılarak etiketler oluşturulabilmektedir.

$$\begin{aligned} &\text{Sınıf}_1 \quad \text{eğer} \quad y > 0 \\ &\text{Sınıf}_2 \qquad \text{değilse} \end{aligned} \tag{3.9}$$

Şekil 3.6 : Yapay sinir ağlarının eğitilmesi.

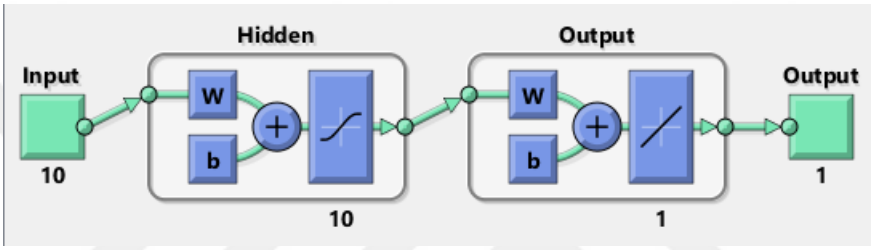
Yapay sinir ağları yöntemi ile öğrenme, daha önce bahsettiğimiz her girdi için oluşan ağırlık değerlerinin en iyi sonuç alınana kadar eğitilmesi ile gerçekleşmektedir.

Yapay sinir ağlarının diğer yöntemlerden en önemli farkı, diğer yöntemlerde eğitim kümesi var olan kurallardan geçirilmekteyken, yapay sinir ağlarında bu kurallar/parametreler her seferinde güncellenmektedir. Hesaplamalar topluca ve eşzamansız olarak yapılmakta ve bellek ağa yayılmış durumdadır. Yapay sinir ağları hata oranı hesaplanarak oluşturulmakta ve donanıma bağlı olarak çalışmaktadır. Donanıma bağımlı olmaları dezavantaj olarak yorumlanmaktadır

çünkü bu ağların en önemli özelliği paralel olarak çalışmaları dolayısı ile paralel çalışan işlemcilere gereksinim duymalarıdır.

3.4.4.3. Kayıp Müşteri Analizinde Yapay Sinir Ağları

Birçok sınıflandırma probleminde başarılı sonuçlar veren yapay sinir ağları yöntemini kayıp müşterilerin tahmininde de kullandık. Yöntemi kullanırken veri kümesinde bulunan 3333 verinin %45 inin eğitim kümesi için %55 i test kümesi için ayrılmıştır. Model oluşturulurken her katmanda üretilecek algılayıcı sayısı 10 olarak belirlenmiştir. İleri beslemeli olarak tasarlanan ağ Şekil 3.7'de gösterildiği gibi giriş, gizli/ara katman ve çıkıştan oluşmaktadır.



Şekil 3.7: Kayıp müşteri analizi için kullanılan yapay sinir ağı modeli.

Ağa verilen her verinin 10 özelliği bulunmaktadır. 1500 müşterinin verisi eğitim seti için kullanılırken 1833 adet veri ise test için kullanılmıştır. Yapay sinir ağlarının kayıp müşterilerin tahmininde ki başarısı %91 olarak belirlenmiştir. Modelin ara katman sayısı 10 olarak belirlenmiştir. Katman sayısı artırıldığında veya azaltıldığı tahmin başarısında artış görülmemektedir. Yapay sinir ağları yönteminin özellik seçimi ve çıkarımından sonra elde edilen karışıklık matrisi Tablo 3.13'de ve 3.14 de verilmiştir.

Tablo 3.13: Veri Kümesi-I ile özellik seçimi sonrası Yapay Sinir Ağları yönteminin karışıklık matrisi.

	Öngörülen Sınıf		
Gerçek Sınıf		Müşteri Kaybı Var	Müşteri Kaybı Yok
	Müşteri Kaybı Var	168	125
	Müşteri Kaybı Yok	34	1506

Tablo 3.14: Veri Kümesi-I ile özellik çıkarımı sonrası Yapay Sinir Ağları yönteminin karışıklık matrisi.

	Öngörülen Sınıf		
Gerçek Sınıf		Müşteri Kaybı Var	Müşteri Kaybı Yok
	Müşteri Kaybı Var	157	136
	Müşteri Kaybı Yok	21	1519

Bu yöntem ile müşteri kaybının olduğu 293 veriden 168 doğru sınıflandırılırken, müşteri kaybının olmadığı 1540 adet verinin 1506 tanesi doğru sınıflandırılmıştır. Yapay sinir ağları beklendiği gibi en başarılı sonucu vermiştir. Modelin başarısını ölçmek için kullandığımız doğruluk ve hata oranları ölçüm sonuçları ile kesinlik ve duyarlılık sonuçları da Tablo 3.15'da gösterilmiştir.

Tablo 3.15: Yapay sinir ağları yönteminin başarı ölçümleri.

Doğruluk : $\frac{TP+TN}{TP+FP+FN+TN}$	0.9133
Hata Oranı: $\frac{FP+FN}{TP+FP+FN+TN}$	0.0867
Kesinlik : $\frac{TP}{TP+FP}$	0.8317
Duyarlılık : $\frac{TP}{TP+FN}$	0.5734
F-Ölçütü : $\frac{2*\text{Duyarlılık}*\text{Kesinlik}}{\text{Duyarlılık}+\text{Kesinlik}}$	0.6788

Yapay sinir ağları yöntemi Matlab kullanılarak ileri beslemeli olarak gerçekleştirilmiştir. Ağda kullanılan ağırlıklar 'Gradient descent' yöntemi ile güncellenmiştir. Bu yöntem öğrenme katsayısı olarak 0.01, momentum sabiti olarak ise 0.9 kullanmaktadır. Aktivasyon fonksiyonu için ise varsayılan fonksiyon sigmoid kullanılmıştır.

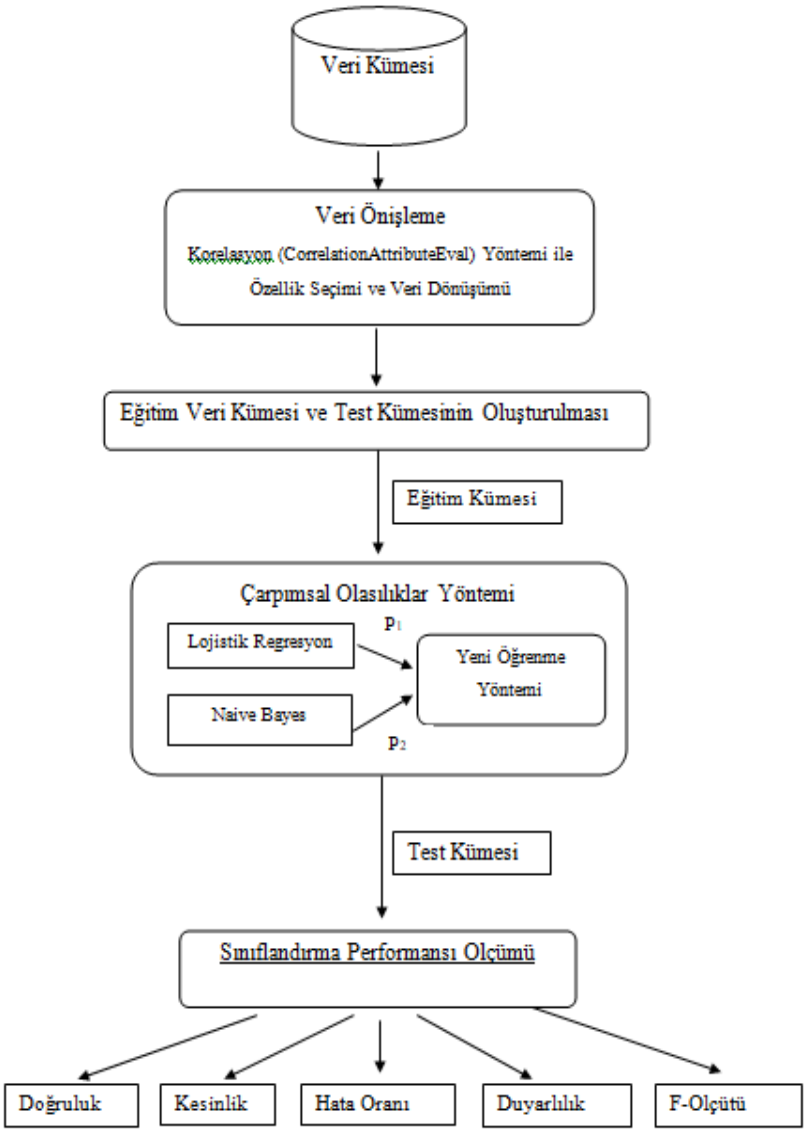
3.4.5. Yeni Yaklaşım: Çarpımsal Olasılıklar Yöntemi

Daha önce yapılmış çalışmalar incelendiğinde sınıflandırma problemlerine hibrit yöntemler ile de çözüm bulunmaya çalışıldığını görmekteyiz. A. Chaudhary, S. Kolhe ve R. Kamal geliştirdikleri hibrit modelde ilk olarak sınıflandırmada etkili olabilecek en önemli özellikleri kullanabilmek adına kazanç oranı özellik seçimi (gain ratio feature selection) yöntemini kullanmışlardır (Chaudhary, Kolhe, & Kamal, 2016). Çalışma da sınıflandırma başarısını yükseltmek için Lojistik Regresyon ve Naif Bayes yöntemlerini bir arada kullanarak bir model

oluşturulmuştur. Her veri için Lojistik Regresyon ve Naive Bayes yöntemlerinden elde edilen olasılık değerlerinin ortalaması göz önüne alınarak yapılan sınıflandırmada başarının %94.73'e çıktığı görülmüştür.

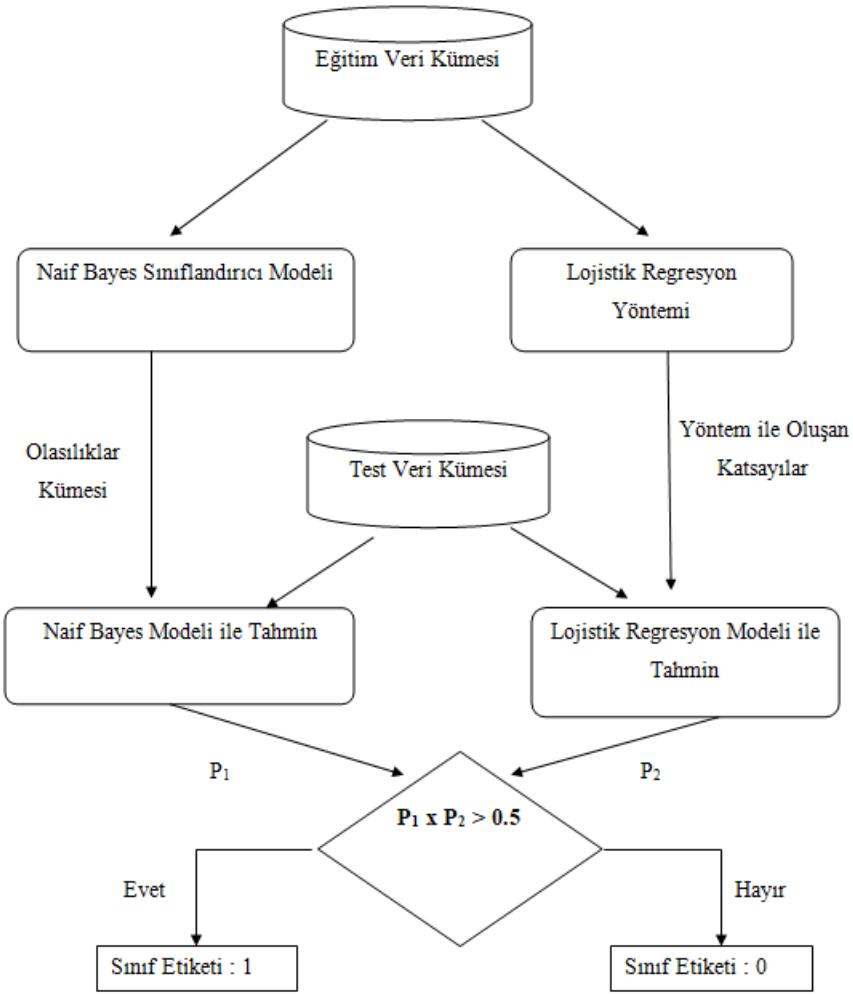
Yapılan başka bir çalışmada ise konvolüsyonel sinir ağı ve aşırı öğrenen yöntemleri tercih edilmiştir (Duan, Li, Yang, & Li, 2017). Bu çalışmalar ışığında, tahmin başarısını artırmak için veri setimize Lojistik Regresyon ve Naif Bayes kullanarak birleşik yöntem uyguladık. Uyguladığımız yöntemin akış şeması Şekil 3.8 de gösterilmektedir. Şekilde gösterildiği gibi veri setimizde gerçekleştirdiğimiz veri ön işleme aşamalarından sonra eğitim ve test veri kümeleri oluşturulmuştur. İlk olarak modelin oluşturulması ve eğitilmesi için eğitim kümesi kullanılmaktadır. Modelde kullanılan eğitim setinde 1500 adet müşterinin 10 farklı özelliği kullanılmıştır. Bu veriler kullanılarak Lojistik Regresyon ve Naif Bayes sınıflandırıcıları ile elde edilen 2 farklı olasılık değeri yeni yaklaşımımız için kullanılmıştır. Bu yöntemde Şekil 3.8 de gösterilen P_1 ve P2 çıktıları sırasıyla Lojistik Regresyon ve Naif Bayes yöntemlerinden elde edilmiştir. Elde edilen değerlerin çarpımı her veri için hesaplanmış ve değerlendirmeye alınmıştır. Elde edilen değer 0 ve 1 arasında değişiklik gösterdiğinden, orta değer olan 0.5 e göre sınıf etiketi tahmin edilmiştir.

Şekil 3.9 da hibrit yöntemde sınıf etiketini belirlemede kullandığımız karar yapısı verilmiştir. Bu yapıya göre hibrit sistemin çıktısı 0,5 den büyük ise sınıf etiketi 1,yani müşteri kaybı var; 0,5'den küçük ise sınıf etiketi 0, yani müşteri kaybı yok olarak tahmin edilmektedir.



Şekil 3.8 : Çarpımsal olasılıklar yönteminin oluşumu.

Şekil 3.9'da yöntemin sınıf etiketini belirlemede kullandığımız karar yapısı verilmiştir. Eğitim veri kümesi ile oluşturulan olasılıklar kümesi ve katsayılar Naive Bayes ve Lojistik Regresyon yöntemleri için kullanılacak modeli oluşturmaktadır. Oluşan modeller test veri kümesinde ki her veri için ayrıca kullanılmıştır. Test veri kümesinde ki her veri için Naive Nayes modelinden P_1, Lojistik Regresyon modelinden P_2 olasılıkları üretilmiştir. Sınıf etiketinin tahmini bu olasılıkların çarpımına göre belirlenmektedir. Bu yapıya göre sistemin çıktısı 0.5'den büyük ise sınıf etiketi 1,yani müşteri kaybı var; 0.5'den küçük ise sınıf etiketi 0, yani müşteri kaybı yok olarak tahmin edilmektedir.



Şekil 3.9 : Çarpımsal olasılıklar yönteminin çalışma prensibi.

Bu yeni yaklaşım için ise test veri kümesi için yine eğitim kümesinde kullanılmayan 1833 kişinin verileri kullanılmıştır. Bu yöntem için özellik seçimi ve çıkarımı sonrası oluşan karışıklık matrisi de Tablo 3.16 ve 3.17'de gösterilmiştir. Özellik seçimi sonrası sonuçlarına bakıldığında yeni sınıflandırıcının özellik seçiminden sonra doğru tahmin başarısının % 84.51 olduğunu görmekteyiz. Temel bileşen analizi ile özellik çıkarımı sonrası ise doğru tahmin başarısının %84.72 olduğu görülmüştür. Bu sonuç daha önce Naif Bayes (%64) ve Lojistik Regresyon (%84) yöntemlerinden aldığımız sonuçların üzerinde bir başarıdır. Bu anlamda Naif Bayes yönteminin lojistik regresyon yöntemini desteklediği, tahmin başarısını artırdığı sonucuna varılabilmektedir.

Tablo 3.16 : Veri Kümesi-I ile özellik seçimi sonrası Çarpımsal Olasılıklar yönteminin karışıklık matrisi.

	Öngörülen Sınıf		
Gerçek Sınıf		Müşteri Kaybı Var	Müşteri Kaybı Yok
	Müşteri Kaybı Var	32	261
	Müşteri Kaybı Yok	23	1517

Tablo 3.17: Veri Kümesi-I ile özellik çıkarımı sonrası Çarpımsal Olasılıklar yönteminin karışıklık matrisi.

	Öngörülen Sınıf		
Gerçek Sınıf		Müşteri Kaybı Var	Müşteri Kaybı Yok
	Müşteri Kaybı Var	8	285
	Müşteri Kaybı Yok	9	1531

Yöntemin sınıflandırma performansı ölçümleri ise Tablo 3.18'de gösterilmektedir. Bu tabloda ölçümleri verilen analiz veri kümesi-I ile özellik seçimi sonrası elde edilen sonuçlar ile tamamlanmıştır. Bu oran Lojistik Regresyon ve Naif Bayes'in tahmin başarısının üzerinde gözükse de, hata oranı, kesinlik gibi diğer performans ölçümlerine bakıldığında geliştirilmesi gerektiği gözlenmektedir. Geçmiş çalışmalarına bakıldığında birleşik sistemlerde bir ya da birden çok yöntem bir arada kullanılırken sonuçların ortalamaları veya ağırlık ortalamalarının alınarak birleştirildiği görülmüştür.

Tablo 3.18: Çarpımsal olasılıklar yönteminin veri seti-I için özellik seçimi sonrası performans ölçümleri.

Doğruluk : $\frac{TP+TN}{TP+FP+FN+TN}$	0.8451
Hata Oranı: $\frac{FP+FN}{TP+FP+FN+TN}$	0.1549
Kesinlik : $\frac{TP}{TP+FP}$	0.5818
Duyarlılık : $\frac{TP}{TP+FN}$	0.1092
F-Ölçütü : $\frac{2*Duyarlılık*Kesinlik}{Duyarlılık+Kesinlik}$	0.1839

Öğrenme algoritmalarının başarı ölçümü yapılırken kullanılan en genel ölçüt doğruluktur. Pozitif veya negatif olarak doğru sınıflandırılmış veri sayısının tüm veri sayısına oranı ile elde edilir. Hata oranı ise pozitif veya negatif olarak yanlış sınıflandırılmış veri sayısının tüm veri sayısına oranı hesaplanmaktadır. Bu ölçütlerin yanı sıra, çalışmanın başarısı hakkında daha çok bilgi veren kesinlik, duyarlılık ve f-ölçütü gibi ölçüm türleri bulunmaktadır. Kesinliği ve duyarlılığı daha iyi olan öğrenme yönteminin daha başarılı olduğu yorumu yapılabilir. Bu iki ölçütün harmonik ortalaması kullanılarak elde edilen F-ölçütü daha genel ve doğru bilgi verebilmektedir.

3.4.5.1. Çarpımsal Olasılıklar Yönteminin Test Edilmesi

Çarpımsal olasılıklar yöntemi yeni bir yaklaşım olduğundan yöntemi başka bir veri kümesi ile deneyip test ettik. Bölüm 3.1.2'de açıklanan veri set-II ile yöntemi tekrar denedik. Veri seti-II'de bulunan verilerden 3169 tanesi eğitim için, 3874 tanesi test için kullanılmıştır. Veri kümesinde bulunan 20 özniteliğin hepsi kullanılmıştır. Karşılaştırma yapabilmek için bu veri seti ile daha önce uyguladığımız Naif Bayes, Lojistik Regresyon, Destek Vektör Makineleri, Yapay Sinir Ağları yöntemleri de uygulanmıştır. Bu veri kümesi ile yeni yöntem %79.43 oranında, Lojistik Regresyon %77.77, Naif Bayes %70.91, Destek Vektör Makineleri %73,85, Yapay Sinir Ağları %79.19 doğru tahmin yapabilmektedir. Veri kümesi-I de olduğu gibi veri kümesi-II ile de yeni yöntem ayrı ayrı Naif Bayes ve Lojistik Regresyon yöntemlerinin sonuçlarından daha yüksek sonuç vermiştir. Yöntemin bu veri seti ile elde ettiğimiz karışıklık matrisi Tablo 3.19 da bulunmaktadır.

Tablo 3.19 : Veri Kümesi-II ile Çarpımsal Olasılıklar yönteminin karışıklık matrisi

	Öngörülen Sınıf		
Gerçek Sınıf		Müşteri Kaybı Var	Müşteri Kaybı Yok
	Müşteri Kaybı Var	562	477
	Müşteri Kaybı Yok	32	2515

Yöntemin sonuçlarına ait doğruluk, hata oranı, kesinlik, duyarlılık, f-ölçütü gibi performans ölçümleri ise Tablo 3.20'de verildiği gibidir.

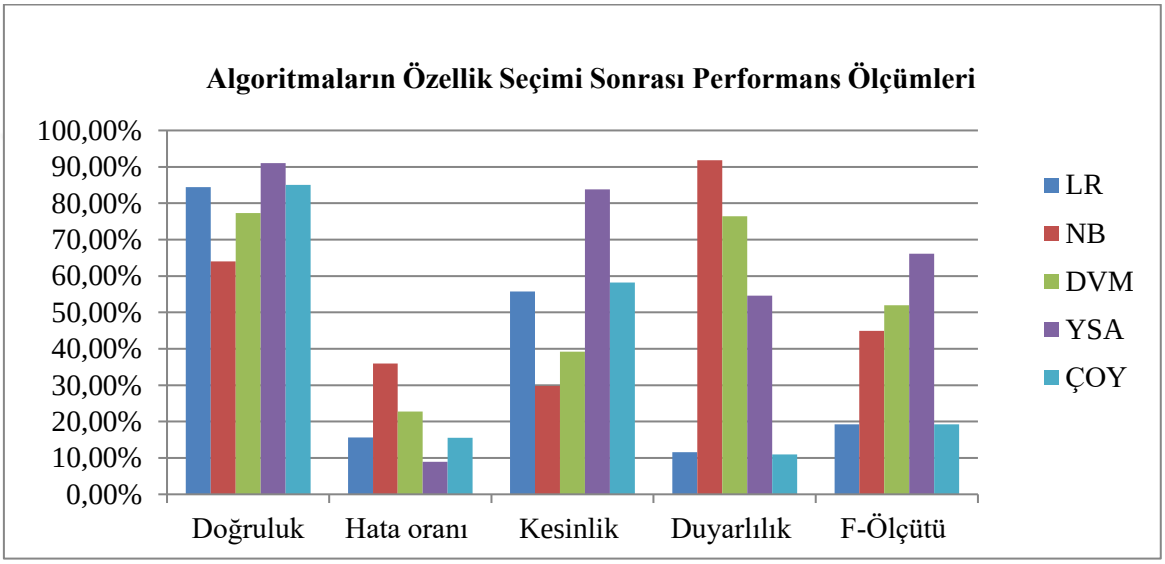
Tablo 3.20 : Çarpımsal olasılıklar yönteminin veri seti-II için performans ölçümleri.

Doğruluk : $\frac{TP+TN}{TP+FP+FN+TN}$	0.7943
Hata Oranı: $\frac{FP+FN}{TP+FP+FN+TN}$	0.2057
Kesinlik : $\frac{TP}{TP+FP}$	0.6372
Duyarlılık : $\frac{TP}{TP+FN}$	0.54009
F-Ölçütü : $\frac{2*Duyarlılık*Kesinlik}{Duyarlılık+Kesinlik}$	0.5851

Veri seti-I için yöntemi öznitelik seçimi ve çıkarımı olmadan toplamda 17 öznitelik ile denediğimizde de tahmin başarısının, bu yöntemi oluştururken kullandığımız iki yöntemin tahmin başarısından yüksek ve %85 olduğu görülmüştür.

4. BULGULAR

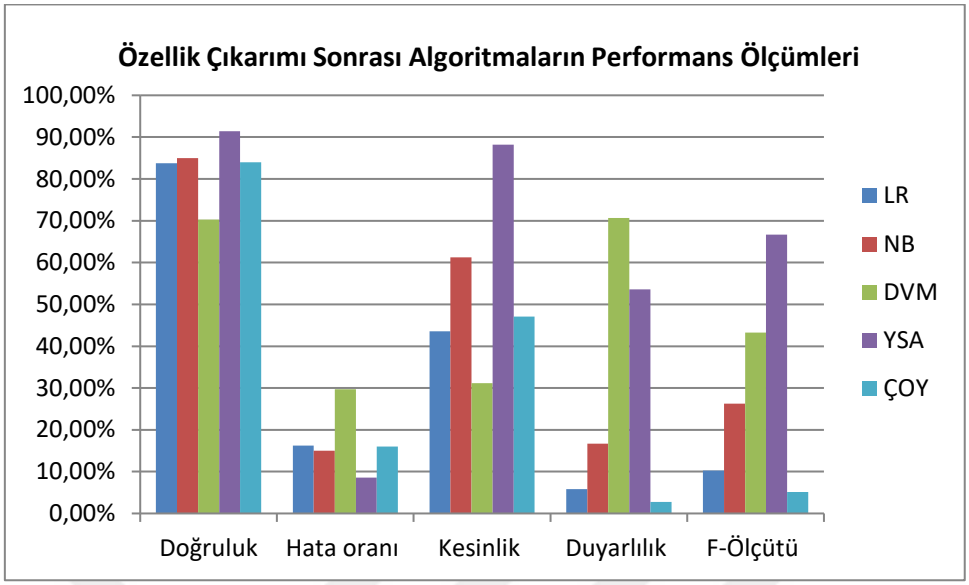
Çalışmamız 1. Veri kümesi ile 3 bölümden oluşmaktadır. 1.bölüm Naive Bayes, Lojistik Regresyon, Destek Vektör Makineleri ve Yapay Sinir Ağları yöntemlerini özellik çıkarımından sonra elde edilen 10x3333 boyutlarına indirilen veri kümesi ile tamamlanmıştır. Ayrıca bu veri kümesi ile yeni geliştirmekte olduğumuz çarpımsal olasılıklar yöntemi (ÇOY) de denenmiştir. Bu veri kümesi ile ilgili performans ölçümleri Şekil 4.1 de görülmektedir.



Şekil 4.1: Veri kümesi-I ile özellik seçimi algoritmaların performans ölçümleri.

Şekil 4.1 de verilen grafikte görüldüğü gibi en yüksek doğru tahmin oranı %91.05 ile YSA algoritmasına ait. YSA algoritmasını %85 ile ÇOY, %84.39 ile LR yöntemleri takip etmektedir. En düşük tahmin başarısını NB gösterirken beklendiği gibi en yüksek hata oranına sahip yöntem olarak grafikte yer almaktadır. Algoritmaların kesinlik ölçümlerinde de yine YSA ilk sırada yer almaktadır. Kesinlik ölçümlerinde en düşük oranı ise NB yöntemi vermektedir. Hata oranları en düşük 3 yöntem sırasıyla YSA, LR ve ÇOY olarak belirlenmiştir. Duyarlılık ve kesinlik ölçümlerinin ortalaması alınarak elde edilen F-ölçütü için ise YSA beklendiği gibi ilk sırada yer almaktadır.

Çalışmamızda analiz ettiğimiz 2. konu ise TBA ile boyut azaltımı sonrası algoritmaların performansının değişimi idi. TBA sonrası elde edilen sonuçlar Şekil 4.2 de verilmektedir.

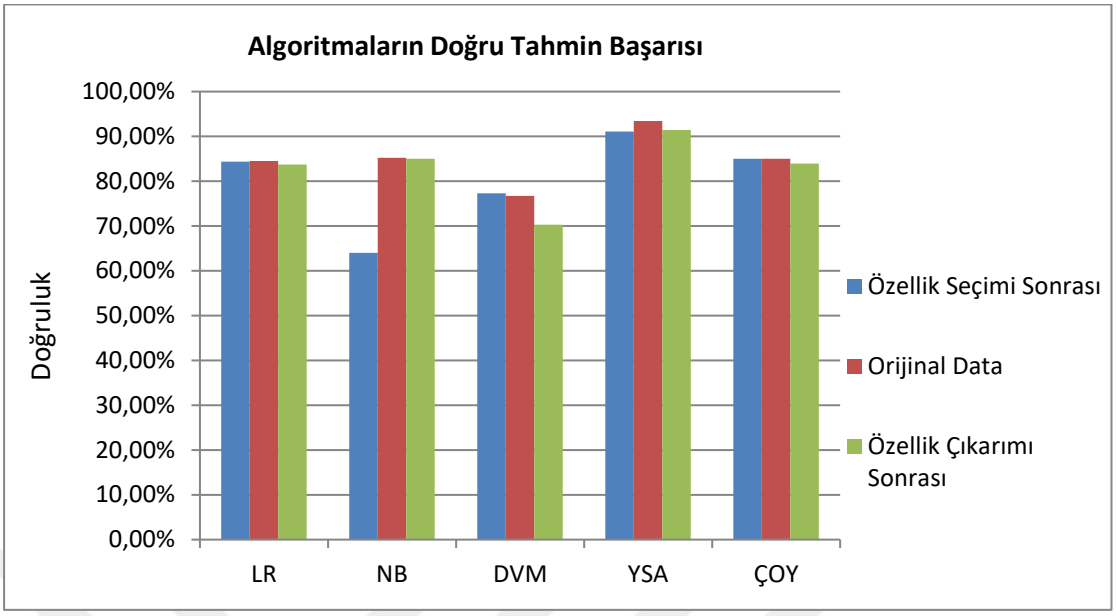


Şekil 4.2: Veri kümesi-I ile özellik çıkarımı sonrası algoritmaların performans ölçümleri.

Temel bileşen analizi sonrası algoritmaların performanslarını değerlendirdiğimizde yine ilk 3 sırada sırasıyla YSA (%91.43), NB(%84,99) ve ÇOY(%83,96) yöntemleri bulunmaktadır. En düşük hata oranı ise yine YSA, NB ve ÇOY ilk 3 sırayı almaktadır. Özellik seçimi ve özellik çıkarımı aşamalarından sonra algoritmaların başarısını değerlendirdiğimizde ilk 3 sırada NB ve LR yöntemleri arasında değişiklik olduğunu görüyoruz. Özellik seçimi ile boyutu azalmış veri setinde LR yönteminin, özellik çıkarımı ile boyutu azalmış veri setinde NB yönteminin daha başarılı olduğunu analiz ettik.

F-ölçütü sonuçlarına baktığımızda ise doğruluk ve hata ölçümlerinde ilk 3 sırada bulunmayan DVM yöntemini ikinci sırada bulmaktayız. DVM yönteminin duyarlılık oranı diğerlerinden daha yüksek olduğundan F-ölçütü sıralamasında bu algoritmayı üst sıralara taşımıştır.

İki farklı boyut azaltma işlemleri sonrası yapılan karşılaştırmadan sonra boyut azaltma işlemlerinin performansı nasıl etkilediğini, veri kaybının büyüklüğünü ölçmek için veri setinde boyut azaltmadan algoritmaların performansını tekrar ölçtük ve karşılaştırdık.

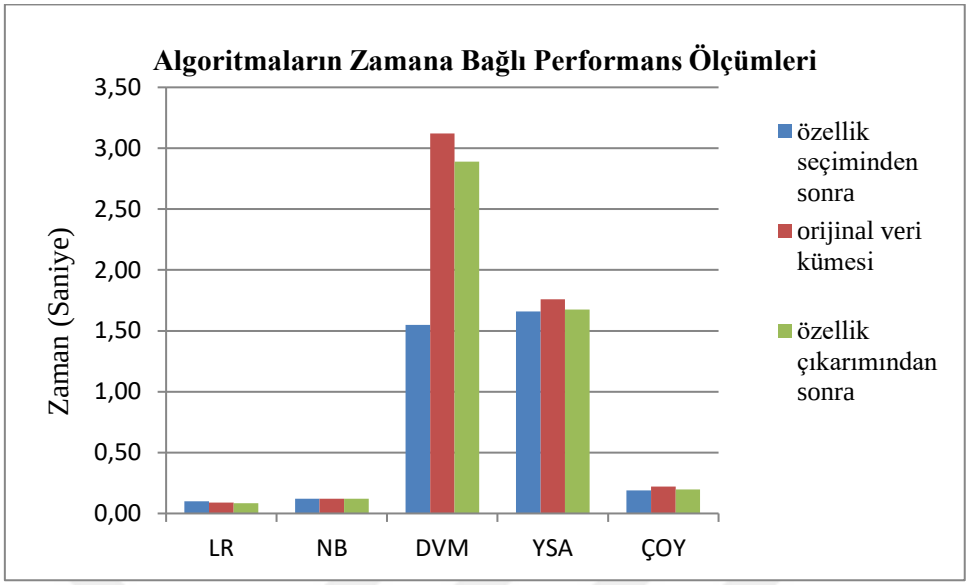


Şekil 4.3: Veri Kümesi –I ile boyut azalttıktan önce ve sonra doğru tahmin başarılarının karşılaştırılması.

Şekil 4.3 de görüldüğü gibi veri kümesinin boyutunun azalması en çok NB yöntemini etkilemektedir. Özellik seçimi yaparak veri kümesinin boyutunu azaltmak doğru tahmin oranını düşürmektedir. Boyut azaltmadan kullandığımız veri kümesinde başarı NB yöntemi ile %85.25 iken, özellik seçimi sonrası %64'e düşmüştür. Özellik çıkarımından sonra ise orijinal veri kümesi ile elde ettiğimize sonuca çok yakın bir sonuç olan %84.99 başarı oranını elde ediyoruz.

YSA yöntemi en yüksek sonucu orijinal veri kümesinde elde etmektedir. LR ve ÇOY yöntemleri kullanılmadan önce veri kümesinin boyutu azaldığında, orijinal veri kümesine göre doğru tahmin başarısında büyük farklar görülmemektedir. Ancak DVM yönteminde temel bileşen analizi ile boyut azaltılması algoritmanın başarısında farka sebep olmaktadır. DVM yöntemi özellik seçimi sonrası %77.29, orijinal veri ile %76.69 ve özellik çıkarımı sonrası %70.31 başarı göstermektedir.

Veri kümesi ile boyut azaltmanın doğru tahmin başarısı üzerinde ki etkisinin yanında bu algoritmaların eğitim ve test aşamalarının tamamlanma sürelerini de ölçtük. Boyut azaltmanın zaman kısıtlaması üzerinde ki etkisine baktığımız da ise Şekil 4.4'de ki grafiği elde ettik.



Şekil 4.4: Veri Kümesi –I ile algoritmaların zamana bağlı performans ölçümleri.

Veri kümesinin eğitilmesi ve test edilmesi için en çok zamana ihtiyacı olan yöntem DVM olarak belirlenmiştir. Bunun ile birlikte özellik seçimi ile birlikte veri kümesinin eğitim ve test sürelerinin oldukça azaldığını da görmekteyiz. NB sınıflandırıcısı için veri boyutunun azalması eğitim ve test sürelerinde dikkat çekici bir değişime sebep olmamıştır. YSA yöntemi için de veri kümesinin boyutunda ki değişiklik eğitim ve test sürelerinde az miktarda değişikliğe sebep olurken, zaman açısından LR için özellik çıkarımı, ÇOY için ise özellik seçimi yolu ile boyut azaltmanın daha doğru olacağını görmekteyiz.

5. TARTIŞMA VE SONUÇ

Yaptığımız analizler sonucunda beklendiği gibi sınıflandırma başarısı en yüksek yöntem YSA olarak belirlenmiştir. Özellik seçimi ile boyut azaltmanın en çok etkilediği yöntem NB olarak bulunmuştur. NB yönteminin doğru tahmin başarısı orijinal veri kümesi ile yapılan çalışmaya nazaran %21 azalmıştır. Bununla birlikte boyut azaltmanın NB yöntemi için zaman açısından bir artısı olmadığı da görülmektedir. Bu sebeple NB kullanılacak çalışmalar için özellik seçimi yöntemi ile boyut azaltmak önerilmemektedir. Temel bileşen analizi ile boyut azalttıktan sonra elde edilen doğru tahmin başarısı ise orijinal veri seti ile aldığımız başarısı arasında %0.27'lik gibi küçük bir fark bulunmaktadır. Bu sebeple de NB için özellik çıkarımı işlemi önemli değildir.

Boyut azaltmanın en büyük etkisini DVM yönteminde görmekteyiz. Özellik seçiminden sonra eğittiğimiz ve test ettiğimiz veri kümesi için DVM yöntemi, orijinal veri kümesine nazaran %50 daha hızlı sonuçlanmaktadır. Bununla birlikte özellik seçimi sonrası doğru tahmin başarısının da orijinal veri kümesi ile yaptığımız çalışmanın doğru tahmin başarısından daha yüksek olduğu görülmüştür. Bu sonuçlara göre DVM kullanılacak çalışmalar için özellik seçimi ile boyut azaltmak kesinlikle önerilmektedir.

Bu çalışma içerisinde geliştirdiğimiz ÇOY yöntemi ise çarpıcı sonuçlar vermektedir. Bu yöntem LR ve NB yöntemleri kullanılarak geliştirilmiş ve bu iki yöntemden daha yüksek tahmin oranına sahip olduğu görülmüştür. Bu yöntem için özellik seçimi yolu ile boyut azaltmak, orijinal veriye göre tahmin başarısını azaltmamaktadır. Ayrıca boyutu azaltılmış veri kümesi ile çalışmak 0.03 saniye hız kazandırmıştır. ÇOY yöntemi özellik seçimi ile boyutu azaltılan veri setinde YSA yönteminden sonra en yüksek doğru tahmin başarısını elde etmiştir. Ancak f-ölçütü ve hata oranı gibi performans ölçümleri göz önüne alındığında, yöntemin geliştirilmesi gerektiği sonucuna varılmıştır.

Kayıp müşteri analizi yapılan önceki çalışmalarda da olduğu gibi, bu çalışmada da en önemli etkenlerden biri kullanılan veri kümesidir. Kullanılan verilerin gerçek veri olması veya suni olarak üretilmiş olması analizi etkilemektedir. Gerçek veri ile yapılan çalışmalarda veri kümesini oluşturan müşterilerin sayısının yetersizliği göze çarpmaktadır. Firmaların toplanan müşteri verilerinin toplanması ve kullanılması konusunda kısıtlamaları bulunmaktadır. Öte yandan elde edilen veri kümesinde bulunan dengesizliklerde kullanılan yöntemlerin

sonuçlarının sağlıklı olmamasına sebep olmaktadır. Örneğin kullandığımız veri kümesinde bulunan 3333 müşteriden 483 tanesi müşteri kaybı var olarak etiketlenmiş, 2850 tanesi kayıp yok olarak etiketlenmiştir. Dolayısı ile iki sınıf etiketine ait veri sayılarında dengesizlik bulunmaktadır. Bu durum kullandığımız öğrenme algoritmalarının, öğrenme veri kümesini kullanarak doğru tahminde bulunması zorlaşmaktadır. Veri kümesinde ki dengeyi sağlamak için kayıp yok olarak etiketlenmiş veri sayısını azaltarak sayıyı 483'e yaklaştırmak bir çözüm olarak görülse de, bu sefer de toplam veri sayısında ki azalma sebebi ile doğru tahmin başarısının güvenirliliği azalmaktadır. Kullanılan yöntemleri test etmek amacı ile kullanılan ikinci veri kümesi ise yine aynı dengesizliği görmekteyiz. Veri kümesi-II de bulunan 7043 adet müşteri verisinden sadece 1869 tanesi müşteri kabı var olarak etiketlenmiştir. Daha gerçekçi ve güvenilir sonuçlar için gerçek kullanıcı verileri ile dengeli etiketlenmiş veri kümelerinin kullanılması önerilmektedir.

KAYNAKLAR

Alpaydın, E. (2013). Çok Katmanlı Algılayıcılar. E. Alpaydın içinde, *Yapay Öğrenme* (s. 198-200). İstanbul: Boğaziçi Üniversitesi Yayınevi.

Alpaydın, E. (2013). Öğrenme Yordamları. E. Alpaydın içinde, *Yapay Öğrenme* (s. 217). İstanbul: Boğaziçi Üniversitesi Yayınevi.

Amina, A., Anwara, S., Adnana, A., Nawaza, M., Alawfib, K., Hussainc, A., et al. (2017). Customer churn prediction in the telecommunication sector using a rough. *Neurocomputing* , 242-254.

Ayhan, S., & Erdoğmuş, Ş. (204). Destek Vektör Makineleriyle Sınıflandırma Problemlerinin Çözümü İçin Çekirdek Fonksiyonu Seçimi. *Eskişehir Osmangazi Üniversitesi İktisadi ve İdari Bilimler Dergisi* , 175-198.

Bavaş, E. (2017, 07). *Erdoğan Bavaş*. http://erdoganb.com/2017/07/naive-bayes-ile-siniflandirma/ adresinden alınmıştır.

Bayrak, P. (tarih yok). 2017 tarihinde Lojistik Regresyon, Teori ve SPSS Çözümleri: http://www.academia.edu/11479607/Lojistik_Regresyon adresinden alındı.

Chaudhary, A., Kolhe, S., & Kamal, R. (2016). A Hybrid Ensemble for Classification in mutliclass datasets: An applicationto Oilseed Diasease Dataset. *Computers and Electronics in Agriculture* , 65-72.

Cheng, C., & Cheng, X. (2016). Anovel Cluster Algorithm for Telecom Customer Segmentation. *Anovel Cluster Algorithm for Telecom Customer Segmentation.* Qingdao, China: 16th International Symposium on Communications and Information Technologies (ISCIT).

Coussement, K., Lessmann, S., & Verstraeten, G. (2017). A comparative analysis of data preparation algorithms for customer churn prediction: A case study in the telecommunication industry. *Decision Support Systems* , 27-36.

Çayıroğlu, İ. (tarih yok). *İleri Algoritma Analizi-5* . 2018 tarihinde http://www.ibrahimcayiroglu.com/dokumanlar/ilerialgoritmaanalizi/ilerialgoritmaanalizi- 5.hafta-yapaysiniraglari.pdf adresinden alındı

Duan, M., Li, K., Yang, C., & Li, K. (2017). A Hybrid Deep Learning CNN-ELM for Age and Gender Classification. *Neurocomputing* , 448-461.

Kanar, Ç. (2014). Predıctıng Lapsıng Customers Wıth Logıstıc Regressıon Approach In Retaıl, (Yüksek Lisans Tezi). *Predıctıng Lapsıng Customers Wıth Logıstıc Regressıon Approach In Retaıl* . Yıldız Teknik Üniversitesi / Fen Bilimleri Enstitüsü / İstatistik Anabilim Dalı / İstatistik Bilim Dalı.

Karaağaç, Ş. S. (2015). Churn Analysıs And Churn Predıctıon In A Prıvate Bank (Yüksek Lisans Tezi). *Churn Analysıs And Churn Predıctıon In A Prıvate Bank* . İstanbul: Marmara Üniversitesi / Fen Bilimleri Enstitüsü / Endüstri Mühendisliği Anabilim Dalı.

Kaynar, O., Tuna, M. F., Görmez, Y., & Deveci, M. A. (2017). Makine öğrenmesi yöntemleriyle müşteri kaybı analizi. *C.Ü. İktisadi ve İdari Bilimler Dergisi* .

Lang, T., & Rettenmeier, M. (2017). Understanding Consumer Behavior with Recurrent Neural Networks. *Understanding Consumer Behavior with Recurrent Neural Networks.* Int. Workshop on Machine Learning Methods for Recommender Systems.

Spanoudes, P., & Nguyen, T. (2017, 03 10). *Deep Learning in Customer Churn Prediction: Unsupervised Feature Learning on Abstract Company Independent Feature Vectors.* 2017 tarihinde Cornell University Library: https://arxiv.org/abs/1703.03869 adresinden alındı

Statistics How To. (2017). 2017 tarihinde http://www.statisticshowto.com/probability-and-statistics/correlation-coefficient-formula/#Pearson adresinden alındı

Vafeiadis, T., Diamantaras, K., Sarigiannidis, G., & Chatzisavvas, K. (2015). A comparison of machine learning techniques for customer churn prediction. *Simulation Modelling Practice and Theory* .

Zhang, T., Cheng, X., Yuan, M., Xu, L., Cheng, C., & Chao, K. (2016). Mining Target Users for Mobile Advertising Based on Telecom Big Data. *Mining Target Users for Mobile Advertising Based on Telecom Big Data.* Qingdao: 2016 16th International Symposium on Communications and Information Technologies (ISCIT).

(2017). 2017 tarihinde bigml: https://bigml.com/user/bigml/gallery/dataset/4f89bff4155268645c000030 adresinden alındı.

(2017). 2018 tarihinde IBM Watson Analytics: https://www.ibm.com/communities/analytics/watson-analytics-blog/predictive-insights-in-the-telco-customer-churn-data-set/ adresinden alındı.

(2017). WikiZer:

https://www.wikizero.com/tr/Binom_da%C4%9F%C4%B1l%C4%B1m%C4%B1

adresinden alınmıştır.

EKLER

EK 1. Özellik seçimi ile boyutu azaltılmış veri kümesi kullanılarak elde edilmiş algoritmaların performans ölçümleri.

	Özellik Seçimi ile Boyutu Azaltılmış Veri Kümesi-I				
	Doğruluk	**Hata oranı**	**Kesinlik**	**Duyarlılık**	**F-Ölçütü**
LR	84,39%	15,61%	55,74%	11,60%	19,21%
NB	64,00%	35,97%	29,76%	91,81%	44,95%
SVM	77,29%	22,71%	39,23%	76,45%	52,00%
YSA	91,05%	8,95%	83,77%	54,61%	66,12%
ÇOY	85,00%	15,49%	58,18%	10,92%	19,24%

EK 2. Orijinal boyutlarda ki veri kümesi kullanılarak elde edilmiş performans ölçümleri.

	Orijinal Veri Kümesi-I				
	Doğruluk	**Hata oranı**	**Kesinlik**	**Duyarlılık**	**F-Ölçütü**
LR	84,50%	15,50%	56,16%	13,99%	22,40%
NB	85,26%	14,74%	54,55%	47,10%	50,55%
SVM	76,69%	23,31%	38,61%	77,47%	51,53%
YSA	93,45%	6,55%	86,19%	70,31%	77,44%
ÇOY	85,00%	15,11%	66,00%	11,26%	19,24%

EK 3. TBA ile boyutu azaltılmış veri kümesi kullanılarak elde edilmiş performans ölçümleri.

	PCA ile Boyutu Azaltılmış Veri Kümesi-I				
	Doğruluk	**Hata oranı**	**Kesinlik**	**Duyarlılık**	**F-Ölçütü**
LR	83,73%	16,27%	43,59%	5,80%	10,24%
NB	84,99%	15,01%	61,25%	16,72%	26,27%
SVM	70,31%	29,69%	31,13%	70,65%	43,22%
YSA	91,43%	8,57%	88,20%	53,58%	66,67%
ÇOY	83,96%	16,04%	47,06%	2,73%	5,16%

EK 4. Veri kümesi- II ile test edilen algoritmaların performans ölçümleri.

	Veri Kümesi -II				
	Doğruluk	**Hata oranı**	**Kesinlik**	**Duyarlılık**	**F-Ölçütü**
LR	77,77%	22,23%	77,13%	24,35%	37,02%
NB	70,91%	29,09%	47,49%	80,27%	59,68%
SVM	73,85%	26,15%	50,82%	77,96%	61,53%
YSA	79,19%	20,81%	66,39%	45,62%	54,08%
ÇOY	79,43%	20,57%	63,72%	54,09%	58,51%

ÖZGEÇMİŞ

Kişisel Bilgiler	
Adı Soyadı	Melike Günay
Doğum Yeri	Kırıkkale
Doğum Tarihi	09.06.1988
Uyruğu	❍ T.C. ❒ Diğer:
Telefon	0505 540 1003
E-Posta Adresi	melike_gny@hotmail.com
Web Adresi	

Eğitim Bilgileri	
Lisans	
Üniversite	İstanbul Kültür Üniversitesi
Fakülte	Mühendislik Fakültesi
Bölümü	Bilgisayar Mühendisliği
Mezuniyet Yılı	01.09.2012

Yüksek Lisans	
Üniversite	İstanbul Üniversitesi
Enstitü Adı	Fen Bilimleri Enstitüsü
Anabilim Dalı	Bilgisayar Mühendisliği
Programı	Bilgisayar Mühendisliği

Makale ve Bildiriler

Günay, M., & Ensari, T. (2017). Diabet Diagnosis with Machine Learning. *International Conference on Artificial and Soft Computing.* Amsterdam, Netherlands.

Günay, M., & Ensari, T. (2017). Yüz Tanıma Algoritmalarının Karşılaştırılması. *25. Sinyal İşleme ve İletişim* Uygulamaları *Kurultayı (SIU 2017).* Antalya, Türkiye.

Günay, M., & Ensari, T. (2018). EEG Signal Analysis of Patients with Epilepsy Disorder Using Machine Learning Techniques. *The Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science in 2018 (EBBT'2018)* . İstanbul,Turkey.

Günay, M., & Ensari, T. (2018). Makine Öğrenmesi Yöntemleri ile Kayıp Müsteri Analizi. *26. Sinyal İşleme ve İletişim Uygulamaları* Kurultayı *(SIU-2018).* İzmir,Türkiye.

Günay, M., & Ensari, T. (2018). New Approach for Predictive Churn Analysis in Telecom. *The 2018 International Conference on Applied Mathematics and Computational Methods in Engineering (AMCME 2018)).* Venice, Italy.

Şengel, Ö., & Günay, M. (2014). Assistant Assignment to Final Exams(using Genetic Algorithms). *20th Conference of the International Federation of Operational Research Societies(IFOR 2014).* Barcelona, Spain.