T.C.
DOKUZ EYLÜL ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
EKONOMETRİ ANABİLİM DALI
EKONOMETRİ PROGRAMI
YÜKSEK LİSANS TEZİ

MAKİNE ÖĞRENMESİ TEKNİKLERİYLE MOBİL ÖDEMEDE SAHTEKARLIK TESPİTİ

Özlem GÜVEN

Danışman
Dr. Öğretim Üyesi Serkan ARAS

İZMİR – 2021

YEMİN METNİ

Yüksek Lisans Tezi olarak sunduğum “Makine Öğrenmesi Teknikleriyle Mobil Ödemede Sahtekarlık Tespiti” adlı çalışmanın, tarafımdan, akademik kurallara ve etik değerlere uygun olarak yazıldığını ve yararlandığım eserlerin kaynakçada gösterilenlerden oluştuğunu, bunlara atıf yapılarak yararlanılmış olduğunu belirtir ve bunu onurumla doğrularım.

26/01/2021
Özlem GÜVEN

ÖZET

Yüksek Lisans Tezi

MAKİNE ÖĞRENMESİ TEKNİKLERİYLE MOBİL ÖDEMEDE SAHTEKARLIK TESPİTİ

Özlem GÜVEN

Dokuz Eylül Üniversitesi

Sosyal Bilimler Enstitüsü

Ekonometri Anabilim Dalı

Ekonometri Programı

Mobil çağın gelişmesi ile birlikte ödeme dünyası mobil cihazlarda hayat bulmuştur. Gelişen teknolojide mobil ödeme hemen hemen her sektörde olduğu gibi toplu taşıma sektöründe de sıklıkla tercih edilmektedir. Mobil biletleme sistemi toplu taşıma yolcularına zaman, maliyet, kullanım kolaylığı açısından birçok fayda sağlamaktadır. Mobil ödemenin diğer ödeme türlerine göre daha çok tercih edilmesi, birçok güvenlik ve gizlilik kaygısını da beraberinde getirmiştir. Toplu taşıma gibi çok kullanıcılı ve aktif kullanılan sistemlerde, mobil uygulamayı çalıntı kredi kartı veya çalıntı hesaplarla kullanmaya çalışan ve kullanım amacı dışına çıkarak sistem zafiyetlerini arayan riskli kullanıcıların bulunması kaçınılmazdır. Söz konusu sahtekar kullanıcıların belirlenebilmesi hem uygulama sahibinin itibarı ve karlılığı hem de müşteri memnuniyeti ve devamlılığı açısından hayati önem taşır.

Bu çalışmanın amacı, ulaşım sektöründe faaliyet gösteren bir şirketin sisteminde var olan sahtekar kullanıcıların tespit edilmesidir. Çalışma akıllı ulaşım sistemlerinde öncü bir firmanın gerçek bir sistemi üzerinde gerçekleştirilmiştir. ABD 'de yer alan Kentkart mobil uygulamasını kullanan kullanıcılar ve kullanıcıların geçmiş hareketleri temel alınarak çalışma yürütülmüştür. Sahteci kullanıcıları belirleyebilmek için bağlam duyarlı yaklaşım ile kullanıcı hareketleri belirlenmiş, sistemde risk analizi yapılmış ve

ardından makine öğrenmesi teknikleri kullanılarak kullanıcıların hareketlerine göre kara liste ve beyaz liste şeklinde kullanıcı sınıflandırmaları sağlanmıştır. Çalışmada sınıflandırma problemlerinde sıklıkla kullanılan yöntemlerden olan Rassal Orman, Destek Vektör Makineleri, Lojistik Regresyon, K-En Yakın Komşu ve Naif Bayes makine öğrenme teknikleri tercih edilmiştir. Mobil uygulama kullanıcıları üzerinde elde edilen sonuçlar basit topluluk öğrenme yöntemleriyle birleştirilmiştir. Uygulamada topluluk öğrenme tekniklerinden Yumuşak Oylama ve Basit Çoğunluk Oylama yöntemleri kullanılmıştır. En yüksek performanslı çalışan sistemin belirlenebilmesi için beş farklı senaryo hazırlanmış ve tüm senaryolarda topluluk öğrenme yöntemlerinin, sınıflandırma algoritmalarından daha başarılı sonuçlar elde ettiği görülmüştür. Bu çalışma ile toplu taşıma firmasının müşterileri arasında sahtekar kullanıcıları kara listeye otomatik olarak alan bir sistem tasarlanmış, bu kullanıcıların firmada sebep oldukları sahtekarlık maliyetleri önlenmiştir.

Anahtar Kelimeler: Makine Öğrenmesi, Sınıflandırma Algoritmaları, Topluluk Öğrenme, İhlal Analizi, Risk Analizi, Kara Liste Yönetimi.

ABSTRACT

Master's Thesis

FRAUD DETECTION IN MOBILE PAYMENT WITH MACHINE LEARNING METHODS

Özlem GÜVEN

Dokuz Eylül University

Graduate School of Social Sciences

Department of Econometrics

Econometrics Program

With the deandlopment of the mobile age, the world of payment has come to life on mobile devices. In the deandloping technology, mobile payment is frequently preferred in the public transportation sector as in almost eandry sector. Mobile ticketing system provides many benefits to public transport passengers in terms of time, cost and ease of use. The preference of mobile payments oandr other payment types has brought many security and privacy concerns. In multi-user and actiandly used systems such as public transportation, it is inevitable to find risky users who try to use the mobile application with stolen credit cards or stolen accounts and seek system vulnerabilities by going beyond the intended use. Identifying the fraudulent users in question is vital for both the reputation and profitability of the application owner, as well as customer satisfaction and continuity.

The purpose of this study is to identify fraudulent users in the system of a company operating in the transportation industry. The study was carried out on a real system of a leading company in smart transportation systems. The study was conducted based on the past moandments of users and users using the Kentkart mobile application in the USA. In order to identify fraudulent users, user actions were determined with a context awareness approach, risk analysis

was performed in the system, and then user classifications in the form of a black list and a white list were provided using machine learning techniques. In the study, Random Forest, Support Andctor Machines, Logistic Regression, K-Nearest Neighbor and Naiand Bayesian machine learning techniques, which are among the methods frequently used in classification problems, were preferred. The results obtained on mobile application users are combined with ensemble learning methods. Soft Voting and Simple Majority (Hard) Voting methods were used in the practice. In order to determine the highest performing system, fiand different scenarios were prepared and it was obserandd that the ensemble learning methods achieandd more successful results than the classification algorithms in all scenarios. With this study, a system that automatically blacklists fraudulent users among the customers of the public transportation company was designed, and fraud costs caused by these users in the company were prevented.

Keywords: Machine Learning, Classification Algorithms, Ensemble Learning, Fraud Analysis, Risk Analysis, Blacklist Management.

MAKİNE ÖĞRENMESİ TEKNİKLERİYLE MOBİL ÖDEMEDE SAHTEKARLIK TESPİTİ

İÇİNDEKİLER LİSTESİ

İKİNCİ BÖLÜM

MAKİNE ÖĞRENMESİ YÖNTEMLERİ

KISALTMALAR

CART	Sınıflandırma ve Regresyon Ağacı
FN	Yanlış Negatif
FP	Yanlış Pozitif
KNN	K-En Yakın Komşu
LASSO	En Küçük Mutlak Küçültme ve Seçim Operatörü
LR	Lojistik Regresyon
NB	Naif Bayes
NFC	Yakın Alan İletişimi
PCA	Temel Bileşenler Analizi
PDA	Kişisel Dijital Asistan
RF	Rassal Orman
RFE	Özyinelemeli Özellik Seçimi
RFID	Radyo Frekansı ile Tanımlama
SMS	Kısa Mesaj Servisi
SVM	Destek Vektör Makinesi
TN	Doğru Negatif
TP	Doğru Pozitif

TABLOLAR LİSTESİ

ŞEKİLLER LİSTESİ

GİRİŞ

Gelişen teknoloji mobil ödeme kavramını da hızla hayatımıza katmıştır. Mobil ödeme sisteminde ödeme, alınan bir malın veya hizmetin karşılığını mobil ağ bağlantısı olan herhangi bir elektronik cihaz üzerinden satın alınması ile gerçekleşir. Mobil tipte ödeme sistemi toplu taşıma sektöründe de benimsenmiş, mobil cihazlar üzerinden bilet satışı, mobil biletleme olarak adlandırılmış ve zamanla sıklıkla kullanılan bir yöntem haline gelmiştir. Mobil biletleme sistemi ile bilet basımı, dağıtımı, korunması, saklanması, taşınması gibi süreçlerin önüne geçilebilirken, kullanıcılar da günlük hayatlarının bir parçası olan mobil cihazlar ile herhangi bir zamanda veya herhangi bir yerde bilet alabilmekte, almış olduğu biletleri cihazında taşıyabilmekte ve bunları mobil cihazı aracılığı ile kolayca kullanabilmektedir.

Ödeme sistemlerinin tümünde olduğu gibi mobil biletleme sisteminde de güvenlik, gizlilik, işlevsellik gereksinimleri bulunur. Bu gereksinimlerin sağlanması sistemin devamlılığı, müşteri memnuniyeti, işletmenin karlılığı ve güvenilirliği açısından önemlidir. Sistem gereksinimleri güvenlik duvarları, çok katmanlı mimari programlama, güvenli yazılımlar, casus yazılım önleyiciler gibi çeşitli yöntemlerle korunmasına karşın kriptografik yöntemlerin kullanılamadığı ya da çözüm sağlayamadığı durumlar için ortaya çıkabilecek güvenlik zafiyetlerini önlemek için yöntem geliştirme ihtiyacı oluşmuştur. Bu amaçla mobil uygulamada üretilen ve/veya saklanan hassas verilerin cihazdan çalınması, kopyalanması veya sızdırılması, kişisel verilerin erişimi, sistem açıklarından yararlanılması, çalıntı kart ya da hesapla sistemin kullanılması gibi işletmeyi çeşitli şekillerde zarar verebilecek kullanıcıları tespit eden ve engelleyen bir yöntem geliştirilmiştir. Uygulama akıllı ulaştırma sistemlerinde öncü, hem yurtiçi hem yurtdışı birçok şehirde toplu taşıma sistemleri sağlayan Kentkart firmasından sağlanmıştır. Çalışmada firmanın ABD ‘de aktif olarak kullanılmakta olan akıllı ulaştırma mobil uygulamasına ait kullanım verilerinden faydalanılmıştır.

Geliştirilen yöntemin amacı basitçe, riskli kullanıcıların (uygulamayı kopyalama, uygulama özelliklerini ve promosyonlarını kötüye kullanma, bilet sistemine nüfuz ederek sahte bilet üretme, bunları dağıtma veya satma işlemi yapan gibi) belirlenerek bu kullanıcıların otomatik olarak kara listeye alınması ve böylece sisteme girişlerinin engellenmesidir. Söz konusu riskli kullanıcıların alınacak önlemlere adapte olarak davranışlarını zaman içerisinde değiştirmesi durumu dikkate alınarak, kara liste yapısı kural tabanlı değil öğrenme tabanlı algoritma ile oluşturulmuştur. Bu amaçla işletme verilerine göre en elverişli yöntemi bulmak için 5

farklı senaryo geliştirilmiştir. Oluşturulan senaryolar uygulamada değişken seçimi ve değişken indirgeme ihtiyacına göre farklılık göstermektedir. Senaryolar ile birlikte değişkenler belirlenmiş, sınıflandırma algoritmaları çalıştırılmış ve ardından topluluk öğrenme yöntemleriyle nihai çıktı belirlenmiştir. Sınıflandırma yapmak amacıyla makine öğrenmesi yöntemlerinden Rassal Orman, Destek Vektör Makineleri, Lojistik Regresyon, K-En Yakın Komşu ve Naif Bayes algoritmaları kullanılmıştır. Algoritmalar riskli kullanıcıların belirlenmesi için kullanıcı geçmişlerinin oluşturulduğu bağlam verileri ile eğitilerek kullanıcıları kara listede ve beyaz listede olacak şekilde sınıflandırmıştır. Sınıflandırma teknikleriyle modeller kurulduktan sonra, sınıflandırma teknikleri arasından seçim yapmak yerine tüm teknikler üzerinden birleştirme yapılarak daha iyi bir sınıflandırma performansına ulaşmak hedeflenmiştir. Bu amaca yönelik olarak, topluluk öğrenmesi için Yumuşak Oylama ve Basit Çoğunluk Oylama yöntemleri kullanılmıştır. İki oylama yönteminden en iyi sonucu veren Yumuşak Oylama yöntemi ile nihai sınıflandırma yapılmış ve ardından sisteme verilen kullanıcının kara listede olma olasılığını gösteren olasılıklar uygulamada tehdit skorları olarak nitelendirilmiştir. Böylece otomatik olarak kara liste ve beyaz liste kararı verilen kullanıcılar üzerinde tehdit skorlarına göre gözle kontrol etme kolaylığı da sunulmuştur. Söz konusu yöntem ile mevcutta kara listede olan kullanıcıların atak seviyeleri ve sisteme genel davranış biçimleri öğrenilerek, bu çıkarım doğrultusunda diğer kullanıcıların hareketleri incelenmiş ve kullanıcıların riskli olup olmadığına karar verilmiştir. Belli bir tehdit skoru seviyesini aşan kullanıcılar otomatik olarak kara listeye alınmış, böylece gözle kontrole harcanan zaman büyük ölçüde azaltılmıştır.

Geliştirilen öğrenme tabanlı algoritmanın, sistemin yeni risklerini ve sisteme yönelik saldırıları gerçekleştiren kişileri en kısa zamanda, en etkin şekilde ve minimum sapma ile belirlemesi amaçlanmıştır. Bu sistem binlerce farklı olası risk kural kombinasyonu tanımlama yükünden kaçındığı gibi, gerçekleştirilen otomasyon sistemini kullanan firmaya hem zaman, hem de maliyet optimizasyonu sağlamıştır. Bu çalışma, gerçekleştirilen firmadaki bir mobil ya da web hesaba sahip tüm kullanıcıları makine öğrenme teknikleri ile otomatik olarak sınayarak kullanıcılarının risk skorlarını gösteren ve böylece güvenilir olan ve olmayan kullanıcıların belirlenebildiği bir yapı oluşturmuş, gerçek bir ihtiyacı firma itibarı ve karlılığı açısından tatmin edici bir seviyede karşılamıştır.

Çalışmanın birinci bölümünde temel olarak ödeme sistemlerinden, mobil ödeme ve mobil biletleme yöntemlerinden ve yöntemlerin güvenlik, gizlilik ihtiyaçlarından, güvenlik ve gizliliğin ihlal edilmesinden doğan mobil sistemlerdeki sahtekarlıklardan bahsedilmiştir. Ardından çalışmanın temel konusunu oluşturan mobil sistemlerde sahtekarlık tespiti ve önlenmesi konularına değinilmiştir. İlk bölümde son olarak sistemdeki sahtekarlığın boyutunun anlaşılabilmesi için gerekli olan risk analizinden ve sergilenen risklerin belirlenebilmesini sağlayan bağlam duyarlı yaklaşım ele alınmıştır.

Çalışmanın ikinci bölümünde ise genel olarak makine öğrenmesi yaklaşımından bahsedilmiş ve ardından makine öğrenmesinde boyut indirgeme ve değişken seçimi, sınıflandırma ve topluluk öğrenme teknikleri ele alınmıştır. Makine öğrenmesinde sonuçların değerlendirilmesine yardımcı olacak metriklerden bahsedilerek ikinci bölüm sonlandırılmıştır.

Çalışmanın son bölümünde, uygulamaya ait detaylar sunularak faydalanılan makine öğrenmesi tekniklerinin sonuçları verilmiş ve sonuçlar değerlendirilerek en iyi senaryo seçilmiştir. Böylece kullanılan gerçek bir sistemde otomatik bir şekilde kara liste yönetimi gerçekleştirilmiş ve işletmeye faydaları tartışılmıştır.

BİRİNCİ BÖLÜM

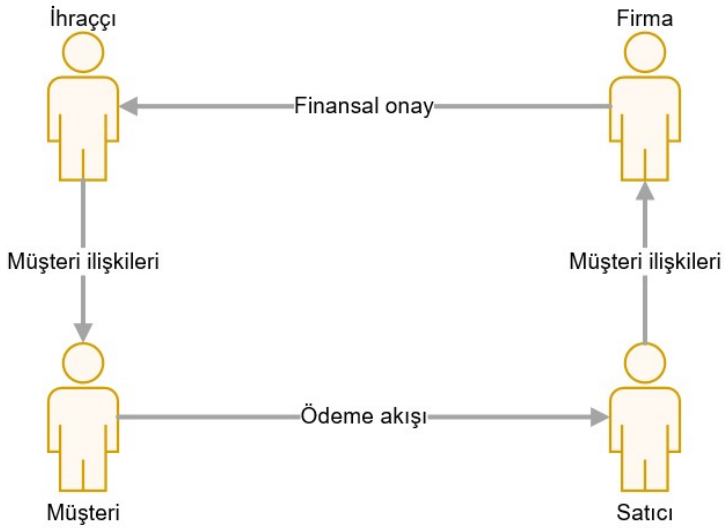
MOBİL SİSTEMLER

1.1. ÖDEME SİSTEMLERİ

Ödeme, bir mal veya hizmet alımı karşılığında belirlenen bedeli karşılama eylemidir. Ödeme sistemleri temel ödeme teorisi ile yönetilir. Ödeme teorisi aslında oldukça basittir. Bu teori, bir yükümlülüğü yerine getiren iki taraf arasındaki finansal değerin takasında kullanılan prosedürleri, kuralları, standartları ve araçları tanımlar. Amaç oldukça basit olmasına rağmen, ödeme sistemi zamanla değişikliğe maruz kalmıştır. Her ülkenin içinde bulunduğu koşullar (nüfus, yasal sistem, iş uygulaması, iletişim, altyapı, araçların ve kuruşların gelişme aşamaları vb.), sahip olduğu ödeme sistemini eşsiz kılar. Bu sebeple gelişmiş bir ülkenin ödeme sistemini gelişmekte olan ülkelere olduğu gibi kopyalayarak tek bir model olarak kullanmak mümkün değildir (Listfield ve Montes, 1994).

Çoğu ödeme işlemi üç temel aşamadan oluşur. İlk olarak tüketici istediği ürünü alışveriş yaparak seçer. Alışveriş aşamasından sonra müşteri satıcı tarafından faturalandırılır. Son olarak, müşteri satıcıya mal için ödeme yapar (Eze ve diğerleri, 2008: 224-231). Ödeme işlemi bittikten sonra, ödeme aracının geçerli olup olmadığı bir yetkili tarafından değerlendirilir. Ödeme aracı geçerli, sistem tarafından tanınan bir araçsa ödeme onaylanır. Şekil 1 ‘de görülen müşteri; ödemeyi yapan taraf, satıcı; ödemeyi kabul eden taraf, firma; tüccar ile ilişkisi olan ve tüccar ile etkileşime giren üçüncü taraf ve ihraççı diğer bir deyişle sağlayıcı; müşteriyle ilişkisi olan ve müşteriyle etkileşime giren bir üçüncü taraftır (Karnouskos, 2004: 44-66).

Şekil 1: Ödeme Döngüsü



Kaynak: Karnouskos, 2004: 44-66.

Ödeme çeşitli kanallardan yapılabilir. Zamanla ödeme eyleminin gerçekleştirilebileceği birçok sistem geliştirilmiştir. En yaygın olan ödeme sistemleri; nakit, kredi kartı, hesap kartı, mobil ödeme, elektronik cüzdan ile ödeme olarak bilinmektedir.

1.1.1. Nakit Ödeme

Oxford sözlüğüne göre nakit, "madeni para veya banknot cinsinden para" anlamına gelir (Oxford University Press, https://www.oxfordlearnersdictionaries.com/, (09.01.2021)). Alınan mal veya hizmet karşılığında fiziki bir değer ile ödeme yapılıyorsa ödeme nakit olarak gerçekleştirilmiş demektir. Madeni paralar, banknotlar, cari hesaplardaki paralar ve kısa vadeli mevduatlar, kullanılmayan banka kredili mevduatlar, dövizler ve hızlı bir şekilde para biriminize dönüştürülebilen mevduatları içerir (Amoako, 2011).

1.1.2. Kredi Kartı

Banka tarafından kullanıcıya tahsis edilmiş kredi ile ilişkili fiziksel bir karttır. Kredi kartında kullanıcının kredi geçmişini temel alan bir harcama limiti vardır, bir kullanıcı tüm kredi kartı bakiyesini ödeyebilir veya her bir fatura dönemi için minimum tutarı ödeyebilir (Upadhayaya, 2012: 37-41). Kredi kartının temel işlevi ekonomik alışverişi kolaylaştırmaktır. Bu açıdan bakıldığında, kredi kartları, her bir kart türü (örneğin banka kartı, mağaza tarafından verilen kart, seyahat ve eğlence kartı) farklı

bir ödeme sistemini temsil eden kolaylaştırıcı sistemler olarak incelenebilir (Hirschman, 1979: 58-66).

1.1.3. Hesap (Debit) kartı

Kullanıcının banka hesabını temsil eden fiziksel bir karttır. Ödeme bankanın onayıyla gerçekleşir. Kart sahibinin banka hesabında ödemeyi karşılayacak tutarı mevcutsa, banka ödemeyi onaylar. Satış tutarını kart sahibinin banka hesabından kaldırır ve satıcının banka hesabına aktarır (Upadhayaya, 2012: 37-41).

1.1.4. Mobil Ödeme

Mobil ödeme, parasal bir işlemin başarılı bir şekilde tamamlanmasını sağlayan, mobil ağa bağlantısı olan bir elektronik cihaz içeren her türlü bireysel veya ticari faaliyet olarak tanımlanabilir. Bu nedenle, temel olarak iki taraf arasındaki ödemelerin ve işlemlerin bir mobil cihaz kullanılarak her zaman ve her yerde hızlı, rahat, güvenli ve basit bir şekilde tamamlanmasını içerir (Liébana-Cabanillas ve diğerleri, 2014: 464-478). Bu ödeme türü müşterilerin cep telefonları aracılığıyla mal veya hizmet satın almalarını ve ödeme yapmalarını sağlar. Burada, her cep telefonu uzaktan satış ile bağlantılı olarak kişisel ödeme aracı olarak kullanılır. Telefon tabanlı ödeme sistemi, cep telefonunun hem fiziksel kartı hem de kart terminalinin yerini alması bakımından geleneksel kart tabanlı ödeme türüne göre avantaj sağlar (Gao ve diğerleri, 2009: 320-329). Mobil ödeme, mallar, hizmetler ve faturalar için alternatif bir ödeme yöntemidir. Mobil cihazları (cep telefonu, akıllı telefon veya Kişisel Dijital Asistan gibi) ve kablosuz iletişim teknolojilerini (mobil telekomünikasyon ağları veya yakınlık teknolojileri gibi) kullanır (Kim ve diğerleri, 2010: 310-322). Popüler inancın aksine, mobil ödemeler kendilerini cep telefonu üzerinden ödemelerle sınırlamaz, aşağıdakiler de dahil olmak üzere neredeyse tüm mobil cihazlarda ödeme yapmak mümkündür (Karnouskos, 2004: 44-66):

- Tablet (genellikle bir kişi tarafından kullanılan sınırlı hareket kabiliyetine sahip tam işlevli bir kişisel bilgisayardır).
- PDA (ing. Personal Digital Assistant) (multimedya ve bağlantı özelliklerine sahip gerçek bir kişisel dijital asistan olan bir mobil cihazdır).
- Akıllı telefon (PDA 'ların ve eski cep telefonlarının birleştirilmesi).
- Bir ödemeyi başlatma, etkinleştirme ve / veya onaylama yeteneğine sahip tüm mobil ödeme terminalleri veya cihazları (yerleşik güvenliği olan satıcı tarafından işletilen terminaller).

1.1.5. Elektronik Cüzdan

Elektronik cüzdan, doğrudan veya telefon hattı üzerinden başka bir cüzdanla elektronik bağlantı sağlayan küçük bir cep hesap makinesine benzer. Taklit edilemeyen bir miktar para depolar ve diğer cüzdanlarla taklit edilemez işlemler sağlar. Sadece parolayı bilen yasal sahip onu çalıştırabildiğinden nakit paradan daha güvenlidir (Even ve diğerleri,1984: 383-386). Elektronik ödeme sistemi, etkileşimde bulunan üç taraf arasında bir dizi protokolden oluşur: bir banka, bir müşteri (ödeyen) ve bir dükkan (alacaklı). Müşteri ve dükkanın her ikisinin de bankada bir hesabı bulunur. Sistemin amacı, müşterinin hesabından mağazanın hesabına güvenli bir şekilde para aktarmaktır (Camenisch ve diğerleri, 1996: 88-94). NFC (ing. Near Field Communication) de elektronik cüzdanlarda çalışabilen akıllı bir teknolojidir. NFC, RFID (ing. Radio Frequency Identification) etiketleri gibi kısa menzilli bir kablosuz teknolojidir. Telefonların içindeki etiketlerde kişisel bilgiler saklanabilir ve bunlar araba anahtarları, para, bilet ve seyahat kartları gibi davranabilirler (Eze ve diğerleri, 2008: 224-231). Ödeme sistemleri, yetkilendirme sürecinin bir parçası olarak ödeme cihazları için fiziksel yakınlığa dayalı ödemeler kullanır. Bu durumda, ödemenin alıcısına gereken yakınlık güvenliği artırır ve mobil ağın maliyet yaratma işlevini doğrudan kullanma ile karşılaştırıldığında bu sistem, saldırıları olası hale getirir (Becher ve diğerleri, 2011: 96-111).

1.1.6. Mobil Biletleme

Bilet, kimlik doğrulama sunucusu ile son sunucu arasında biletin verildiği kişinin kimliğini güvenli bir şekilde doğrulamak için kullanılır. Bilet sunucunun adını, istemcinin adını, istemcinin internet adresini, bir zaman damgasını, bir yaşam süresini ve bir rastgele oturum anahtarını içerir. (Steiner ve diğerleri, 1988: 191-202). Li vve diğerleri (2009) ulaşım sektörü için bilet tiplerini kategorilere ayırdılar. Sektörün yaygın kullanılan biletlerinin kağıt biletler, manyetik kartlar, akıllı kartlar ve cep telefonlar olduğunu tespit ettiler (Li ve diğerleri, 2009: 38-51). Bu çalışmada ulaşım sektöründe mobil ödeme kullanılarak yapılan biletleme sistemi ele alınmıştır.

Mobil biletleme Li ve diğerleri (2009) tarafından yapılan tanıma göre, müşterilerin cihazlar veya akıllı kartlar gibi temassız teknolojiler ya da cep telefonunu kullanarak bilet sipariş etme, ödeme, satın alma ve doğrulama hizmetlerine olanak sağlayan sistemdir. Sistemin avantajı, belli bir dereceye kadar bankalar ve kredi kartı şirketlerini dışarıda tutarak mobil faturalandırma ile süreci kolaylaştırmasıdır. Bu da

demek oluyor ki, mobil biletleme maliyet azaltmak için dikkate değer bir potansiyeldir (Zheng ve Chen, 2003: 24-27).

Patel ve Crowcroft (1997) yayınladıkları bir makalede biletin özelliklerinden bahsetmişlerdir. Onlara göre bilet; az yer kaplayan, taşınabilir, farklı kullanıcı tiplerini, cihazları ve servisleri destekleyebilen esneklikte, güvenli ve basit olmalıdır. Bu özellikler bilet fiziksel ya da mobil uyumlu olsa da geçerlidir (Patel ve Crowcroft, 1997: 223-233). Dijital biletin özellikleri aşağıdaki şekilde açıklanmıştır (Fujimura, Nakajima, 1998: 177-186).

- Güvenilir,
- Bazıları anonim, bazıları değil,
- Taşınabilir,
- Bazıları transfer edilebilir, bazıları edilemez,
- Offline çalışabilir,
- Bölünebilir,
- Kalıcı,
- Kabul edilebilir,
- Kullanıcı dostu,
- Makine tarafından anlaşılabilir,
- Geçiş statüleri yönetilebilir,
- Birleştirilebilir.

Dijital bilet dijital ödeme sistemleri ile entegre çalışır. Biletlerin dijital ortamda satın alınması için mobil ödeme hizmetleri kullanılır. Mobil ödeme hizmeti, kullanıcıya sunulan tüm teknolojilerin yanı sıra ödeme hizmeti sağlayıcıları tarafından ödeme işlemlerini gerçekleştirmek için yürütülen tüm görevleri içerir (Kim ve diğerleri, 2010: 310-322).

Mobil cihazların toplu taşımada sağladıkları faydalar düşünüldüğünde, kullanıcıların elektronik biletleri tercih etmesi şaşırtıcı bir sonuç olmayacaktır. Örneğin, internet erişimi olan herhangi bir kullanıcı, herhangi bir yerde elektronik biletini satın alabilir. Fiziksel bilet taşıma, koruma, saklama zahmetine girmeden istediği yerde mobil cihazı aracılığı ile biletini aktifleştirerek kullanabilir. Fakat bilet satın alma, doğrulama, kullanma gibi işlemlerin gerçekleştirildiği bir cihazın, aynı zamanda kullanıcıların günlük yaşantısından bilgiler barındırıyor olduğu düşünüldüğünde çeşitli kaygılar da beraberinde gelmektedir. Bu tip sistemlerin çalıştığı uygulamalar, biletleme dışında konum tabanlı seyahat planlayıcı ve etkinlik habercisi gibi hizmetleri de sunmaktadır. Bu durum kullanıcılara ait önemli verilerin

ifşa olma kaygısını güçlendirmekte ve önlemler alınmasını zorunlu hale getirmektedir. Bu durum sebebiyle bir takım güvenlik ve gizlilik gereksinimleri ortaya çıkmıştır.

1.2. GÜVENLİK VE GİZLİLİK KAVRAMLARI

Gizlilik, etkili güvenlik önlemleri uygulamak ve uygulamaları saldırılara karşı güvenli hale getirmek için büyük bir itici güçtür. Güvenlik, gizlilikle aynı değildir. Ancak etkili güvenlik, çalışanlar ve müşteriler hakkındaki verilerin gizliliğini korumak için bir ön koşuldur. Bazı durumlarda güvenlik ve mahremiyetin birbirine taban tabana zıt olabileceğini unutmamak da önemlidir. Örneğin; iyi kimlik doğrulama, saygıdeğer ve etkili bir güvenlik savunmasıdır. Ancak gizlilik sorununu da gündeme getirebilir. Birçok kişi gizlilik ve güvenliği aynı sorunun farklı boyutları olarak görür. Bununla birlikte gizlilik, politikayı uygulamanın bir yolu ve güvenlik, politikayı dayatmanın bir yolu olarak görülebilir. Tuvaletler bu konseptin iyi bir benzetimidir. Tuvalet kapısının üzerindeki işaret, tuvalete kimlerin girmesi gerektiğine dair politikayı belirtir. Fakat hiçbir güvenlik, girmek isteyebilecek kişileri engellemez. Kapıya bir kilit eklemek, gizlilik politikasının uygulanmasına yardımcı olacak güvenliği sağlar (Howard ve Lipner, 2006).

Güvenlik kavramı mobil/web sistemlerde siber güvenlik olarak bilinmektedir. Siber güvenlik bilgisayarları, ağları, programları ve verileri; saldırı, yetkisiz erişim, değişiklik veya imhadan korumak için tasarlanmış bir dizi teknoloji ve süreçtir. Siber güvenlik sistemleri, ağ güvenlik sistemleri ve ana bilgisayar güvenlik sistemlerinden oluşur. Bunların her birinin en azından bir güvenlik duvarı, virüsten koruma yazılımı ve saldırı tespit sistemi vardır (Buczak ve diğerleri, 2015: 1153-1176). Bu sistemler siber ortamların güvenlik politikalarını uygulamaya yardımcı olmaktadır.

Özetle güvenlik ve gizlilik kavramları birbirleri ile ilişkili olsalar da birbirlerinden farklılardır. Güvenlik kişinin ya da unsurun korunması ve gizlilik kişinin ya da unsurun mahremiyeti ile ilişkilendirilebilir. Bir uygulamada güvenliğin sağlanması uygulamaya olan saldırıların önlenmesine ya da azaltılmasına yardımcı olur. Dolayısıyla güvenlik, sistemin açıklarının kapatılmasına ya da minimize edilmesine ve güvenilir müşteriler tarafından kullanılabilirliğin artırılmasına yarar. Gizliliğin sağlanması ise uygulama kullanıcılarının kişisel, gizli verilerinin korunmasına, sistem bilgilerinin zararlı kişi ve unsurlardan saklanmasına yardımcı olur. Dolayısıyla gizlilik müşterinin uygulamaya güvenini ve memnuniyetini arttırır. Bu sebeple sistem sahiplerinin uygulamalarında hem güvenliği hem de gizliliği hedeflemeleri yerinde olacaktır.

Android cihazlardan gizli bilgileri çalan veya premium numaraları kullanarak kısa mesaj (SMS) metinleri gönderen kötü amaçlı yazılımlar da dahil olmak üzere mobil ortamlarda çok sayıda güvenlik ve gizlilik riski vardır. Kurumsal uygulamalara, sistemlere ve verilere erişebilen mobil cihazlar ve gizli bilgileri açığa çıkarabilecek mobil uygulamalar buna örnek olabilir. Veri güvenliği ve gizlilik işletmeler ve mobil kullanıcılar için ciddi endişeler doğurmaktadır (Jain ve Shanbhag, 2012: 28-33).

1.2.1. Mobil Biletleme Uygulamalarının Korunması

Elektronik bankacılık, finansal hizmetlerin internet üzerinden yürütülmesini, perakende bankaların işlerini önemli ölçüde değiştiren, aynı zamanda maliyetleri düşüren ve müşteri için rahatlığı artıran bir kolaylıktır. PDA 'ların yaygınlaşması ile elektronik bankacılık kavramına yeni bir "mobil bankacılık" kavramı getirilmiştir (Pousttchi ve Schurig, 2004: 10).

Artan teknoloji ile birlikte yaygınlaşan mobil uygulamaların korunması, karşılanması gereken mutlak bir gereksinim haline gelmiştir. Özellikle mobil ödeme, kullanıldığı sistemlerin uygulamadaki potansiyel ve muhtemel risklerin farkına varılarak aksiyon alınmasını gerektirmektedir. Aksi takdirde uygulamanın kullanım amacı suistimal edilebilir, uygulama güvenilir kullanıcılara erişemeyebilir, asıl kullanıcı sayısı öngörülemeyebilir, uygulamanın ve şirketin itibarı zedelenebilir, dahası büyük maddi kayıplarla karşılaşılabilir. Verimli elektronik ödeme sistemleri, elektronik ticaret için önemli bir ön koşuldur. Bu tür ödeme sistemlerinin tasarımı güvenlikle ilgili birçok sorun ortaya çıkarmaktadır. Sahtekarlıkların önlenmesi gibi ortak güvenlik gerekliliklerinin yanı sıra, katılımcıların mahremiyetinin korunması da önemli bir konudur (Camenisch ve diğerleri, 1996: 88-94).

Mobil ödeme gerektiren uygulamalarda müşterilerin en büyük beklentisi gizliliktir. Hiçbir müşteri ödeme yöntemini, kart bilgilerini ya da hassas alış verişler için sepet içeriğinin öğrenilmesini istemez. İdeal bir ödeme sisteminde, satıcılar müşterilerinin kimliklerini öğrenmemeli ve bankalar müşterilerinin satın aldığı ürünler hakkında fiyatı dışında herhangi bir bilgi almamalıdır. Sipariş gizliliği ve ödeme detayları gizli izlemeye karşı korunmalıdır (Hassinen ve diğerleri, 2006: 86-100).

Qin ve arkadaşları (2017) mobil cüzdan üzerine bir güvenlik ve gizlilik analizi çalışması gerçekleştirmişlerdir. Bu çalışmada da bahsedildiği gibi; her şeyden önce, müşteri ile satıcı arasında gidip gelen ödeme bilgilerinin herhangi bir saldırgan tarafından aşılamadığından veya değiştirilemediğinden emin olunmalıdır. Aksi takdirde, sahte ödeme bilgileri mobil cüzdanın itibarı için büyük zarar verebilir. Diğer

taraftan, kötü niyetli müşterinin gerçek kimliği sistem yöneticisi tarafından açıklanmalıdır. Ancak en dürüst, güvenilir müşterilerin mahremiyeti mümkün olduğunca korunmalıdır (Qin ve diğerleri, 2017: 55-60).

Dijital ödemenin kullanımı kolay olduğu kadar korunması zordur. Uygulamanın temel unsuru olan mobil biletleme, dijital ödeme gerektiren bir bilet türüdür. Özellikle ulaşım sektöründe biletlerin taşınamaması, kopyalanamaması, transfer edilememesi, son kullanma tarihinden sonra kullanılamaması, kullanım sayısının aşılamaması, anonim kalması, farklı kullanıcı özelliklerinden yararlanılamaması gibi gereksinimler vardır. İlgili gereksinimlerin başka bir kullanıcı ya da sistem tarafından ihlal edilmemesi için dijital ortamdaki biletlerin korunması gerekmektedir. Bazı çalışmalar bir mobil ödeme uygulamasının etkin bir şekilde yaşayabilmesi için kullanıcıların gizliliklerinin ve sistemin güvenliğinin mutlaka korunması gerektiğini savunurlar (Karnouskos ve diğerleri, 2004; Linck ve diğerleri, 2006; Dewan ve Chen, 2005: 4-28; Hassinen ve diğerleri, 2006: 86-100). Bu amaçla yayınlanan çalışmalardan biri Pirker ve Slamanig tarafından yapılmıştır. İlgili çalışmada hem mikro hem makro mobil ödeme sistemlerine sahip, çevrimiçi çalışan uygulamalarda, hem gizliliği hem güvenilirliği başarabilmek için ön ödemeli bir strateji önerilmiştir (Pirker ve Slamanig, 2012: 1155-1160).

Mobil biletleme sistemlerinin bir takım gizlilik gereksinimleri bulunmaktadır. Yapılan ya da yapılacak sahtekarlıkların uygulamanın gereksinimlerden türetilebileceği için sistemin gizlilik modülleri ilgili gereksinimleri karşılayabiliyor olmalıdır. Gizlilik gereksinimleri aşağıdaki gibi listelenebilir (Dave, 2010);

- Mahremiyet: Kullanıcılara ve kişisel verilere yetkisiz erişim olmamalıdır.
- Anonimlik: Yetkisiz NFC Etiketi erişim tanımlama olmamalıdır.
- Konum Gizliliği: Kullanıcı konumunun yetkisiz takibine izin verilmemelidir.
- İzlenebilirlik: NFC etiketinin mevcut durumu izlenememelidir.
- Kimlik Doğrulama: Hiçbir yetkisiz kullanıcının sistemi kullanmasına veya sisteme erişmesine izin verilmemelidir.
- Taklit edilemezlik: Geçerli erişim etiketlerinin veya aygıtların taklit edilmesine, kopyalayıp çoğaltılmasına izin verilmemelidir.
- Yükümlülük: Kullanıcıların anlaşmaların sürdürülebildiği hesap ortamı sağlanmalıdır.
- Esneklik ve Açıklık: Kullanıcılara kişisel bilgilerini yönetmek için kontrol esnekliği sağlanmalıdır.

- Kullanım Kolaylığı: Kullanıcılara kişisel verileri kolayca yönetmek için kullanıcı ara yüzü ve kolay işlevler sağlanmalıdır.
- Kullanıcı Algısı ve Denetimi: Kullanıcıların özel verilerini kontrol edebilecekleri ve cihazda hangi verilerin kullanılabileceğini yönettikleri ortam sağlanmalıdır.

Elektronik ödeme sistemlerinin somut güvenlik gereksinimleri, hem özelliklerine hem de faaliyetlerine ilişkin güven varsayımlarına bağlı olarak değişir. Bununla birlikte, genel olarak, elektronik ödeme sistemleri bütünlük, yetkilendirme, gizlilik, kullanılabilirlik ve güvenilirlik sergilemelidir (Asokan ve diğerleri, 1997: 28-35). Bu da demek oluyor ki, mobil biletleme ile çalışan sistemlerde gizlilik gereksinimleri kadar güvenlik gereksinimleri de önem taşımaktadır. Sistemin güvenliğini sağlamak amacıyla korunması gereken gereksinimler aşağıdaki gibi ifade edilebilir (Pousttchi ve Schurig, 2004: 10):

- Şifreleme: Verilerin aktarımı şifrelenmelidir. Bunun nedeni, mobil bankacılık uygulamasının hassas verileri iletmesidir. Bu verileri korumak için bağlantının şifrelenmesi gerekir.
- Yekilendirme: Kullanmadan önce verilere erişim yetkisi verilmelidir. Bir kullanıcının verilerine erişebilmesi için, bunu yapmaya yetkili olduğunu kanıtlaması gerekir. Bütünlüğü olan bir ödeme sistemi, o kullanıcı tarafından açıkça yetkilendirilmeden bir kullanıcıdan para alınmasına izin vermez. İstenmeyen rüşvet gibi şeylerin ortaya çıkmasını önlemek için açık rıza olmadan ödeme alınmasına da izin vermeyebilir. Yetkilendirme, bir ödeme sistemindeki en önemli ilişkiyi oluşturur. Ödeme üç şekilde yetkilendirilebilir: bant dışı yetkilendirme, şifreler ve imza ile yetkilendirme (Asokan ve diğerleri, 1997: 28-35).
- Basitlik: Yetkilendirme basit ve anlaşılır olmalıdır.

Söz konusu gereksinimler korunamadığında sistem büyük zararlar ile karşı karşıya kalır. Karşılaşılacak olası risklerden bazıları şu şekildedir:

- Casus yazılım saldırıları,
- Ağ sızdırma saldırıları,
- Kişisel verilerin ifşa edilmesi,
- Mobil uygulama kodlarının ifşa edilmesi,
- Kimlik hırsızlığı,
- Satın alınan biletlerin izinsiz kopyalanması, dağıtılması
- Biletlerin kullanım hakkından fazla kullanılması,
- Bilet içeriğinin değiştirilmesi,
- Ücret ödemeden bilet hak sahibi olunması,

- Kötü niyetli kullanıcıların şirket itibarını zedelemesi.

Sistem güvenliğinin korunması için bazı çalışmalarda kriptolojik işlemlerle random üretilen karmaşık anahtarlar ile mobil ödemeye ilişkin gizlilik mekanizmaları sunulmuştur (Hwang ve diğerleri, 2006: 101-114; Hashemi ve Soroush, 2006: 294-297). Kullanıcının korunması içinse, genel olarak sahtekarlık tespiti ve önlenmesinin gerekliliği savunulmuştur (Wang ve diğerleri, 2016: 1-5; Van Vlasselaer ve diğerleri, 2015: 38-48; Chan ve diğerleri, 1999: 67-74). Polla ve diğerlerinin (2012) önerisine göre sahtekarlık tespiti ve önlenmesi için uygulama içindeki aktörlerin rolleri incelenmeli, her bir aktörün tehdit mekanizması belirlenmelidir. Yaptıkları çalışmada tehditleri minimize edebilmek için dikkate alınması gereken aktörler şu şekilde belirlenmiştir (La Polla ve diğerleri, 2012: 446-471):

- Cihazı güvenli bir şekilde kullanmak için eğitilmesi gereken kullanıcılar,
- Akıllı telefonu hedef alan güvenlik koruması geliştirebilen yazılım geliştiricisi,
- Saldırıları önleme mekanizmalarıyla ağ altyapısını geliştirebilen ağ operatörü,
- Saldırganlar için güvenlik açıklarından yararlanmak daha zor olacak şekilde aygıtları otomatik olarak güncellemesi gereken telefon üreticileri,
- Önceden tespit edilmiş bir virüsün salgın başlatabileceğini tahmin etmek için yeni epidemiyolojik modeller.

Bu çalışmada kullanıcının gizliliğinin ve güvenliğinin sağlanması için sahtekarlık tespit etme ve önleme yöntemlerine odaklanılmıştır.

1.3. MOBİL SİSTEMLERDE SAHTEKARLIK

Sistemde meydana gelen hareketlerde önceden belirlenen bir eşik değerini aşan sapmalar saldırı olarak nitelendirilir (Zhang ve Zulkernine, 2006: 2388-2393). Sistemin işleyişini bozan, sisteme dışarıdan müdahale getiren, sistem açıklarından faydalanan ve sistemde yeni açıklar yaratan ya da bu eylemleri hedef alan her tür hareket saldırıyı ifade eder.

Sahtekarlık ise, bir tarafın diğerine göre haksız avantaj elde ettiği eylemleri tanımlamak için kullanılan genel terimdir (Behdad ve diğerleri, 2012: 1273-1290). Sahtekarlık, kendini masum göstererek bir kullanıcının diğeri üzerine avantaj elde ettiği tüm yanıltıcı yolları kapsar, her zaman öz güven ve kandırmaca içerir. Sahtekarlık gücün kullanıldığı hırsızlıktan farklıdır (Albrecht ve diğerleri, 2011). Burada söz konusu sistem açıklarının kullanıldığı siber hırsızlıktır. Dolayısı ile siber sahtekarlıkta güç, zorlama, kaba kuvvet kullanılmaz, hileli hareketler ve türlü kandırmacalar ön plandadır.

Vlasselaer ve arkadaşları (2017) çalışmalarında sahtekarlık özelliklerini aşağıdaki gibi derlemişlerdir (Van Vlasselaer ve diğerleri, 2017: 3090-3110):

- Yaygın değildir, tüm veri setinde sahtekarlık içeren veriler normal verilere göre ciddi derecede azdır.
- Dikkatlice planlanmış, iyi düşünülmüştür.
- Zamanla gelişir, sahteciler yakalanmamak için hatalarından ders alır ve zamanla hareketlerini değiştirirler.
- Dikkatle organize olunur, sahteciler birbirleriyle irtibatta olur ve yakalanmadan suç işleme hakkında bilgi paylaşımı yaparlar.
- Belli belirsiz, gizlenmiş masum gözüken hareketler sergilerler.

Mobil telefon güvenliğinin, onu geleneksel bilgisayar güvenliğinden ayıran beş temel yönü vardır (Mulliner, 2006):

- Hareketlilik: Güvenli olabilecek tek bir yerde tutulmazlar ve bu nedenle çalınabilir ve fiziksel olarak kurcalanabilirler.
- Güçlü Kişiselleştirme: Mobil aygıtlar normalde birden çok kullanıcı arasında paylaşılmazken, bilgisayarlar sık sık paylaşılır. Cihazlar sahiplerine yakın tutulur.
- Güçlü Bağlantı: Birçok aygıt, bir ağa veya internete bağlanmak için birçok yolu destekler.
- Teknolojik Yakınsama: Mevcut mobil cihazlar PDA, cep telefonu, müzik çalar ve dijital kamera gibi birçok farklı teknolojiyi tek bir cihazda birleştirir.
- Azalan Yetenekler: Mobil aygıt bir nevi bilgisayardır ancak masaüstü bilgisayarların sahip olduğu birçok özelliğe sahip değildir. Örneğin, bir mobil aygıt tam donanılmlı fiziksel bir klavyeye sahip değildir bu yüzden sınırlı işlem yeteneklerine sahiptir.

Mobil uygulamalar ile gelen taşıma kolaylığı, hemen her yerden çevrimiçi dünyaya erişim, gelişen ödeme sistemleri gibi yeni teknolojiler sebebiyle mobil sistemlerde yeni tehditlerin ortaya çıkması kaçınılmaz bir durumdur. Gelişen mobil sistemlere yapılan tehditler şu şekilde özetlenebilir (Jain ve Shanbhag, 2012: 28-33; Ghosh ve Swaminatha, 2001: 51-57; Sadeghi ve diğerleri, 2008: 102-121):

- Cihazın kaybolması/çalınması,
- Veri engelleme ve yetkisiz güncelleme,
- Kötü amaçlı yazılım,
- Savunmasız uygulamalar,
- Tehlikeli cihazlar,
- Web tarayıcısı sömürme ve işletim sistemi hassasiyeti,

- Sosyal mühendislik,
- Kablosuz ağlardaki hassasiyet,
- Kişisel bilgilerin çalışanlar tarafından görüntülenmesi,
- Coğrafi konum bilgilerinin rahatlıkla alınması,
- Kimliğin gizlenmesi,
- Yetkisiz giriş yapılmaya çalışılması,
- Kriptolojinin kırılması,
- Yetkisiz oturumlar,
- Ağ içindeki kullanıcı hareketlerinin izlenmesinin önlenmesi,
- Dürüst kullanıcıların geçiş ağına erişiminin engellenmeye çalışılması.

Bu tip yapılan saldırıların en çok karşılaşıldığı sistem alanları ise şu şekilde sıralanabilir (Wang ve diğerleri, 2015: 1-6):

- Sistem mimarisi, güvenlik duvarları, yazılım yamaları
- Kötü amaçlı yazılımlar, güvenlik politikaları ve insan faktörleri
- Üçüncü parti zincirleri ve iç tehditler
- Veritabanı şemaları ve şifreleme teknolojileri

Genellikle sahteci, yanlış veya yanıltıcı temsil kullanarak, hileli davranışlarının etkilerini en üst düzeye çıkarma çabası ile mümkün olduğunca uzun süre tespit edilmesini önlemek için aktivitelerini gizleme eğilimindedir (Behdad ve diğerleri, 2012: 1273-1290). Bu sebeple sisteme saldıran kişilerin göz ile tespit edilmesi ve önlenmesi zorlaşır, bir takım önleme ve tespit mekanizmalarına ihtiyaç duyulur. Bugünün ağ tabanlı ekonomik kaynakları ile başarılı bir saldırı, tüketicinin güvenini negatif etkiler ve tüketicilerin elektronik alışveriş yapma isteğini azaltır (Erbacher ve diğerleri, 2002: 38-47). Bu sebeple mobil biletleme sistemlerinde saldırıların önceden önlenmesi, önlenemeyenlerin ise daha sonradan analiz edilerek gerekli tedbirlerin alınmasıyla sisteme müdahale edilmesi gerekmektedir.

1.3.1. Sahtekarlık Önleme

Birçok güvenlik mekanizması, mobil ödeme güvenliğinin sağlanması için benimsenmiştir. Ancak, mobil ödeme aynı zamanda kötü amaçlı yazılım tespiti, çok faktörlü kimlik doğrulama, veri ihlali önleme, sahtekarlık tespiti ve önlenmesi gibi güvenlik sorunlarıyla karşı karşıyadır (Wang ve diğerleri, 2016: 1-5). Bu güvenlik sorunlarını aşmanın ilk adımı sisteme yapılacak atakların önlenmesi, saldırgan kullanıcının sisteme giriş yapmasının engellenmesi ya da saldırılı hareketlerinin kısıtlanmasının sağlanmasıdır. Sahtekarlık önleme Bolton ve diğerleri (2002)

tarafından sahtekarlığın meydana geldiği ilk yerde durdurulması olarak tanımlanmıştır (Bolton ve diğerleri, 2002: 235-249).

Abdallah ve diğerlerinin de (2016) ifade ettiği gibi sahtekarlık önleme aşamasında; bilgisayar sistemlerinde (donanım ve yazılım sistemleri), ağlarda ve verilerde meydana gelen siber saldırıların kısıtlanması, bastırılması, kontrol edilmesi ve meydana gelmesinin önlenmesi gerekmektedir (Abdallah ve diğerleri, 2016: 90-113). Güvenlik duvarlarının güçlendirilmesi, kriptolojik anahtarlar kullanılması ilgili durdurma faaliyetlerini yürütebilmek için önemlidir. Bu önlemler sistemdeki temel atakların birçoğunu ve muhtemelen en tehlikelilerini önler. Bu tip sistemsel önlemlerin ardından tek başına önemsiz gözüken ancak biriktikçe sisteme ciddi zararlar verebilen sahtekarlıkların tespit edilmesi ve önlenmesi gerekmektedir.

Sahtekarlık önleme işlemleri, büyük miktarda tasarruf yapılmasına imkan vermektedir. Sahtekarlık önlendiğinde, işletmede kötü örnekler, sahteci kullanıcılar olmayacağından, herhangi bir tespit ya da araştırma maliyetine gerek kalmaz. İşletme ağır fesih ve ceza kararları vermek zorunda kalmaz. Böylece işletme verimsiz faaliyetlerle ve krizlerle başa çıkmak için çalışma süresi kaybetmez (Albrecht ve diğerleri, 2011). Dolayısıyla sahtekarlık yaşam döngüsü başında sahtekarlık önlendiğinde, işletmede ağır tespit maliyetlerine girilmez. Sahtekarlık önceden önlendiği için sahtekarlık maliyetleri de azalacağından, döngüde güvenlik açısından maliyet minimizasyonu sağlanmış olur. Ancak sahtecilerin davranışlarını güncelleme eğilimi, tespit mekanızmasının her daim güçlü tutulması gerekliliğine sebep olur. Sahtekarlık önleme mekanizmaları başarılı olsa da sahtecilerin sistemde boşluk bulmaları, yeni bir sahtekarlık tipi geliştirmeleri her zaman muhtemeldir. Bu sebeple hem önleme hem tespit mekanizmaları önem kazanmaktadır.

1.3.2. Sahtekarlık Tespiti

Sahtekarlık tespiti, sahtekarlığın işlenmesinin ardından mümkün olduğunca çabuk şekilde sahtekarlığın belirlenmesi işlemidir. Sahtekarlık tespiti, sahtekarlık önleme işlemi başarısız olduktan sonra devreye girer (Bolton ve diğerleri, 2002: 235-249). Ancak çoğu zaman sahtekarlık önleme sistemlerinin başarısız olduğu anlaşılmaz. Bu sebeple tespit sistemleri her ihtimale karşın her zaman güçlü tutulmalıdır. Sistemin güvenlik açığını minimum seviyeye indirmek, karlılığı arttırmak, müşteri memnuniyetini sağlayarak güven kazanmak ve piyasada daha uzun süre, daha etkin kalabilmek için sahtekarlık tespit etme mekanizmaları kullanılmalıdır. Çünkü bu aşama sisteme sızan sahtecileri ayıklamanın tek yoludur.

Rekabetçi bir ortamda sahtekarlık; çok yaygınsa, önleme prosedürleri güvenli değilse ve önlemler işe yaramıyorsa kritik bir sorun haline gelebilir(Phua ve diğerleri, 2010). Bu sebeple sahtekarlığın önlenemediği noktada sistemde oluşacak sızıntıların ve saldırıların köklü bir tespitinin yapılması gerekmektedir.

İyi bir şekilde çalışan sahtekarlık tespit sisteminin özellikleri şunlardır (Zareapoor ve diğerleri, 2012: 52):

- Sahtekarlıkları doğru bir şekilde tanımlamalıdır.
- Sahtekarlıkları hızla tespit etmelidir.
- Gerçek bir işlemi sahtekarlık olarak sınıflandırmamalıdır.

Sahtecilik tespit sistemi sürekli olarak gelişmelidir. Sahteciler hileli davranışlarının tespit edilebilir olduğunu fark ederlerse, stratejilerini geliştirecek ve başka yollar deneyeceklerdir. Tabii ki, yeni sahteciler de sisteme saldırmaya çalışabilir ve bunların çoğu geçmişte başarılı olmuş sahtekarlık tespit yöntemleri tarafından fark edilmeyebilir. Bu yüzden önceki tespit araçlarına en son gelişmelerin uyarlanarak eklenmesi gerekmektedir (Bolton ve diğerleri, 2001: 235-255).

Şekil 2: Sahtekarlık Yaşam Döngüsü

Şekil 2 'de görüldüğü gibi sahtekarlık yaşam döngüsü, sahtekarlık önleme sistemleri ile başlar. Sahtekarlığın önlenemediği noktada tespitin başlanabilmesi için kullanıcı profilinden kullanıcı verileri alınarak kullanıcı hareketleri değerlendirilir. Değerlendirme aşamasında sahtecilik yapan kullanıcılar ve masum kullanıcılar etiketlenerek sahteci olarak belirlenenler uygulamaya göre otomatik olarak belli periyodlarla tespit aşamasından geçirilebilir. Uygulamadaki sahtekarlıklar tespit edildikten sonra risklerin ortadan kaldırılması için yeniden sahtekarlık önleme aşamasına geçilir. Sistemdeki sahtekarlıkların bitmemesi ihtimaline karşın tüm aşamalar tekrarlanır. Bu işlemler sebebiyle, sahtekarlık tespiti asla tamamlanmaz. Döngü, sistem yaşamı boyunca devam eder.

1.3.2.1. Sahtekarlık Tespit Tipleri

Suistimallik tespiti ve aykırı gözlem tespiti olmak üzere iki çeşit sahtekarlık tespitinden bahsetmek mümkündür. Kullanıcı profilleri incelendiğinde kullanıcılardan çıkarılan kalıplar, bilinen saldırılarla uyum gösteriyor ve bu davranış sürdürülüyorsa, bu tip ataklar kötüye kullanım olarak da bilinen suistimallik sahtekarlık tespiti ile ortaya çıkarılır. Ancak kullanıcı hareketlerinde değişiklik gözleniyorsa yeni ataklar aykırılık tespiti ile belirlenir. Sistem saldırıları nasıl tespit edilecekse edilsin, ilk temel adım kullanıcı profillerinin belirlenmesidir.

1.3.2.1.1. Suistimallik Tespiti

Suistimallik tespiti, bilinen saldırılardan çıkarılan kalıplara dayalı saldırılardır (Hall ve diğerleri, 2005: 17-24). Sistemin kötüye kullanılması üzerine zaten tespit edilmiş saldırıların tekrarlandığında belirlenebilmesine dayanır. Suistimallik tespiti kullanıcıların aktivitelerinin, sisteme sızma girişiminde bulunan sahtecilerin bilinen davranışları ile karşılaştırılmasını içerir (Kumar ve diğerleri,1994).

Suistimallik tespiti teknikleri genellikle kural tabanlı bir yaklaşım kullanır. Yanlış kullanım tespit edildiğinde kurallar ağ saldırıları için senaryolara dönüşür. Bu mekanizma, kullanıcının etkinliklerinin belirlenen kurallarla tutarlı olduğunu bulursa, potansiyel bir atak belirlemiş olur (Cannady, 1998: 443-456).

Kötüye kullanım olarak da bilinen suistimallik tespiti düşük yanlış pozitif orana sahiptir, ancak yeni tip saldırıları belirleyemez (Zhang ve Zulkernine, 2006: 2388-2393). Bilinen saldırıları tespit etmede düşük hata oranına sahip olup, bu anlamda etkililerdir. Bununla birlikte, bilinen saldırılara benzer özelliklere sahip olmayan yeni oluşturulan saldırıları tespit edemezler (Kim ve diğerleri, 2014: 1690-1700). Kötüye kullanım tespit sistemleri, tespit maliyet açısından etkindir ve sistemde belirli bir düzeyde güvenlik sağlarlar (Chung ve diğerleri, 1999: 159-178).

1.3.2.1.2. Aykırılık Tespiti

Anormal tabanlı aykırılık saldırı tespiti gözleme ve elektronik profillerde yakalanan ve geliştirilen normal davranıştan sapmaları tanımaya dayanır (Hall ve diğerleri, 2005: 17-24). Aykırılık tespiti temel bir varsayım altında normal davranıştan sapan bilinmeyen saldırıları tespit edebilir (Zhang ve Zulkernine, 2006: 2388-2393). Bu tespit, davranışı beklendik ya da normal olmayan bir veri kümesindeki desenleri bulma işlemidir. Bu beklenmedik davranışlar anormaller veya aykırı değerler olarak da adlandırılır. Anormaller her zaman bir saldırı olarak sınıflandırılamaz, ancak daha

önce bilinmeyen zararlı olan ya da olmayan şaşırtıcı bir davranış olabilir (Agrawal ve Agrawal, 2015: 708-713).

Aykırılık tespiti önemlidir, çünkü önemli ancak nadir olayları belirlerler ve uygulamanın büyük bir kısmında alınan kritik bir hareketi hızla faaliyete geçirebilir. (Ahmed ve diğerleri, 2016: 278-288).

Aykırılık tespitinin en büyük zorluğu kural kümesinin tanımlanmasıdır. Sistemin verimliliği, bu kümenin ne kadar iyi uygulandığına ve tüm protokoller üzerinde ne kadar iyi test edildiğine bağlıdır. Ancak kurallar tanımlandıktan ve protokoller kurulduktan sonra aykırılık algılama sistemleri iyi çalışabilir (Jyothsna ve diğerleri, 2011: 26-35).

Takip eden alt bölümlerde detayları verildiği üzere, aykırılık tespiti için gözetimli (denetimli), gözetimsiz (denetimsiz) ve yarı gözetimli olmak üzere üç temel teknik mevcuttur.

1.3.2.1.2.1. Gözetimli Öğrenme

Denetimli aykırılık tespit teknikleri, "normal" ve "anormal" olarak etiketlenen ve sınıflandırıcının eğitildiği bir veri seti gerektirir (Akhilomen, 2013: 218-228). Gözetimli öğrenme teknikleri normal ve hileli davranışlar arasındaki farkların kalıplarını belirlemek amacıyla tarihsel bilgilerden veya gözlemlerden elde edilen bilgilerden öğrenmeyi amaçlamaktadır. Bu teknikler tam olarak sessiz alarmları, sahtecilerin gizleyemediği yolları bulmayı hedefler (Baesens ve diğerleri, 2015). Denetimli öğrenme, farklı sahtekarlık türlerini veya şüpheli davranışları tespit edebilen sınıflayıcıları hızla geliştirmek için bir yöntem sağlama yeteneğine sahiptir (Tsang ve diğerleri, 2014: 3027-3040). Denetimli bir algoritmanın amacı etiketlenmemiş yeni örnekleri doğru sınıflandırmak için bir sınıflandırıcı, model veya hipotez oluşturmaktır (Godoy ve Amandi, 2005: 329).

Gözetimli öğrenme birçok algoritma ile beraber çalışabilir. K-en yakın komşu, lojistik regresyon ve naif-bayes gibi sınıflandırma algoritmaları, doğrusal regresyon, doğrusal olmyan regresyon gibi regresyon algoritmaları gözetimli öğrenme tekniği ile çalışan algoritmalara örnek olarak verilebilir (Abdallah ve diğerleri, 2016: 90-113).

Bir makine öğrenmesi bakış açısından, sahtekarlık tespit sorunu yeni bir işlemin yasal veya hileli bir işlem olup olmadığına karar vermeye yarayan denetimli bir sınıflandırma görevine karşılık gelir. Denetimli makine öğrenmesi algoritmalarının eğitim aşaması, sınıf dağılımını genelleştiren düzeltilmiş model için tasarlanmıştır (Melo ve diğerleri, 2017: 1-6).

1.3.2.1.2.2. Gözetimsiz Öğrenme

Gözetimsiz aykırılık tespit teknikleri, veri setindeki örneklerin büyük çoğunluğunun normal veya olağan olduğu varsayımı altında veri setinin kalanına uyuyormuş gibi gözüken örnekleri arayarak etiketlenmemiş bir veri setinde anormallikleri saptar (Akhilomen, 2013: 218-228). Gözetimsiz öğrenmenin etiketli veriye ihtiyacı yoktur. Bu sebeple veri setinde etiketleme bilinmediğinde rahatlıkla kullanılabilir. Özellik aramaya rehberlik edecek sınıf etiketlerine sahip denetimli öğrenmenin aksine, gözetimsiz öğrenme "sıradışı" ve "olağan" terimlerinin uygulama içinde ne anlama geldiğini tanımlamasına gereksinim duyar (Dy ve Brodley, 2004: 845-889). Denetimsiz öğrenme yöntemleri eğitim örneklerinin önceden sınıflandırılmasını gerektirmez; bu algoritmalar ortak özellikleri paylaşan örnek kümeleri oluşturur (Godoy ve Amandi, 2005: 329).

Veri madenciliğinde kullanılan en popüler denetimsiz yöntem kümelemedir. Bu teknik verilerdeki doğal gözlem gruplarını bulmak için kullanılır ve özellikle pazar bölümlemesinde yararlıdır. Verilerde bulunan yerel aykırı değerlerden yerel modeller oluşturmamıza yardımcı olmak için kümeleme gibi gözetimsiz yöntemler kullanabiliriz. Sahtekarlık tespiti bağlamında global bir aykırı değer, tüm veri kümesine anormal olan bir gözlemdir. Yerel aykırı değerler, verilerin alt gruplarına kıyasla anormal olan gözlemleri açıklar. Lokal aykırı değer tespiti popülasyonun heterojen olduğu durumlarda etkilidir (Bolton ve diğerleri, 2001: 235-255).

Gözetimsiz öğrenmede iki basit klasik algoritma çalıştırılır; K-ortalamalar gibi kümeleme algoritmaları ve temel bileşen analizi gibi boyut azaltma algoritmaları (Abdallah ve diğerleri, 2016: 90-113). Gözetimsiz öğrenmede yaygın uygulamalar şu şekilde listelenebilir (Zhu ve Goldberg, 2009: 1-130):

- n örneği gruplara ayırmayı amaçlayan kümeleme,
- Çoğunluktan çok farklı olan birkaç örneği tanımlayan yenilik algılama,
- Eğitim örneğinin temel özelliklerini korurken her bir örneği daha düşük bir özellik vektörü boyutuyla temsil etmeyi amaçlayan boyut indirgeme.

1.3.2.1.2.3. Yarı Gözetimli Öğrenme

Yarı denetimli öğrenme, hem etiketli hem etiketlenmemiş verilerden bağımsız değişkenler ile ilgili bir karar kuralını öğrenmeyi ifade eder (Grandvalet ve Bengio, 2005: 529-536). Bu yöntem, gözetimli ve gözetimsiz öğrenmenin arasındadır, dolayısıyla veri kümesini etiketli ve etiketsiz kümelere böler (Chapelle ve diğerleri,

2009: 542-542). Yani veri setinde hem etiketli hem de etiketsiz veriler bulunmaktadır. Bu durumda hem sınıflandırma hem de kümeleme yöntemleriyle birlikte çalışabilir.

Yarı denetimli öğrenmenin çok geniş bir pratik değeri vardır. Birçok uygulamada, etiketli veri yetersizliği mevcuttur. İnsan tarafından yorumlama, özel cihazlar veya pahalı ve yavaş deneyler gerektirdiği için bağımlı değişken etiketlerini elde etmek zor olabilir. Bu tip uygulamalar için yarı gözetimli öğrenme kullanılabilir (Zhu ve Goldberg, 2009: 1-130). Yarı denetimli öğrenmeden özellikle görüntü işleme, bilgi erişimi ve biyoinformatik gibi etiketlenmemiş verilerin bol miktarda işlendiği uygulama alanlarında faydalanılmaktadır.

1.3.2.1.3. Aykırı Tipte Sahtekarlık Tespit Yöntemleri

Sahtekarlık problemine çözüm arayan bir araştırmacı için birçok yöntem bulunmaktadır. Yapay zeka, dağıtılmış ve paralel bilgi işleme, uzman sistemler, bulanık mantık, genetik algoritmalar, makine öğrenimi, sinir ağları, örüntü tanıma, istatistik, görselleştirme, operasyonel veya eşik metrik modeli, markov süreci veya işaretleyici modeli, zaman serisi modeli, sezgi tabanlı, sonlu durum makine modeli, açıklama senaryo modeli, bayesyen model, bilgisayar immünolojisi tabanlı, kullanıcı niyeti tabanlı gibi sayısız sahtekarlık belirleme çözümleri bulunmaktadır (Phua ve diğerleri, 2010; Jyothsna ve diğerleri, 2011: 26-35).

Öngörülü modellemede, tarihsel veriler, hileli davranış profilleri oluşturmak için kullanılır. Amaç mevcut profille benzerliğine dayanan aynı davranışların gelecekteki mevcudiyetini tespit etmektir. Ancak, kural tabanlı sistemler ve öngörülü modelleme sadece bilinen sahtekarlık türlerini belirleyebilir. Diğer taraftan veri madenciliği sistemleri, bilinmeyen şüpheli davranışın örüntüsünü keşfetmek için, büyük veri kümelerini kullanır (Alexopoulos ve diğerleri, 2007: 269-276). Takip eden alt bölümlerde aykırı tipte sahtekarlık tespit adımları verilmiştir.

1.3.2.1.3.1. Kullanıcı Profili Oluşturulması

Sahtekarlık tespiti için kullanıcı profili oluşturulması, kullanıcının sahtekar olup olmadığını anlayabilmeyi kolaylaştıracak, önemli bir yoldur. Bu yolla kullanıcının hareketleri derinlemesine analiz edilir. Kullanıcının her hareketinin sahtekarlık sonucuna etkisi olacağından hareketleri oluşturan girdiler dikkatle belirlenmelidir. Çünkü profilde yer alan hareketler, sahtekarlık tespit sisteminin kurallarını oluşturmaktadır.

Grosser ve diğerleri (2005) diferansiyel analize dayalı bir sahtekarlık tespit sistemi kurmak için iki probleme odaklanmak gerektiğini belirtmektedirler. Bunlar, kullanıcıların profillerini oluşturma ve sürdürme sorunu ve davranışlarda meydana gelen değişiklikleri tespit etme sorunudur (Grosser ve diğerleri, 2005: 613-615). Yani hazırlanan sahtekarlık tespit sistemi ne kadar iyi olursa olsun, iyi bir kullanıcı profili oluşturulamadıysa, oluşturulan profil iyi bir şekilde analiz edilemediyse ve değişen sahteci hareketleri profil oluşturmak üzere belirlenemediyse, tespit sistemi etkin bir şekilde çalışmaz.

Girdi verileri genellikle kullanıcının belirli bir alanda ilgilendiklerine veya tercihlerine ait örnekler biçimindedir. Profil ise bu gibi kullanılabilecek verilerin genelleştirilmesidir (Krulwich, 1997: 37). Diğer bir deyişle profil, kullanıcıyı tanımlayan bir bilgi topluluğudur (Adomavicius ve Tuzhilin, 1999: 377-381).

Uygulamanın kullanıcısını tanımak, davranışlarını analiz etmek, uygulamaya olan bakışını anlayabilmek için ilgili kullanıcının profilinin belirlenmesi gerekmektedir. Kullanıcı profili; cinsiyet, mail adresi, telefon numarası gibi kişinin temel bilgilerinin yanı sıra uygulama içi hareketlerini de içermelidir. Uygulama içi hareketlerin analizi kullanıcının uygulamayı kullanım amacının ve niyetinin ortaya çıkarılmasını sağlar. Örneğin uygulamayı yoğun olarak kullanma saatleri, uygulama içinde en çok kullandığı modülün belirlenmesi, satın alma alışkanlıkları kullanıcı profilini belirleyen temel verilerden sayılabilir.

Kullanıcının ilgi alanlarını temsil etmenin tipik bir yolu, alakalı anahtar kelimelerin bir listesini oluşturmaktır. Bununla birlikte, böyle bir profil kullanıcıların davranışlarını modellemek ve anlamak için yetersizdir. Yüksek kaliteli, internet tabanlı hizmet sağlamak için eksiksiz bir kullanıcı profili oldukça önemlidir (Tang ve diğerleri, 2010: 1-44).

Kişiselleştirilmiş uygulamalarının geliştirilmesindeki temel teknik sorunlardan biri, müşterilerin kim olduklarını ve nasıl davrandıklarını anlatan bu bakımdan en önemli bilgileri sağlayan bireysel müşterilerin doğru ve kapsamlı profillerinin nasıl oluşturulacağıdır. Profil oluşturma sorunu ilk olarak cep telefonu endüstrisinde sahtekarlık tespiti kapsamında incelenmiştir (Fawcett ve Provost, 1996: 8-13). Bu, kural öğrenme sistemini kullanarak cep telefonu kullanım verilerinden bireysel müşterilere ait öğrenme kuralları geliştirilmesiyle ve daha sonra çeşitli müşteri gruplarında genel sahtekarlık koşullarını öğrenmek adına farklı müşteri segmentleri için genelleştirilmiş profiller üretilmesiyle yapılmıştır (Adomavicius ve Tuzhilin, 1999: 377-381).

Schiaffino ve Amandi (2009) çalışmalarında kullanıcı profili içeriğinde, kullanıcının ilgi alanları, bilgi düzeyi, yetenekleri, amacı, davranışları, etkileşim tercihleri, bireysel karakteristiği, bağlamsal bilgileri, grup profilleri olması gerektiğini belirtmişlerdir. Aynı çalışmada bu bilgilerin kullanıcının kendi verdiği bilgilerden, davranışlarının gözlemlenmesinden, geri bildirimlerinden, hiyerarşik olarak kullanım kalıplarının belirlenmesinden alınabileceğinden de söz edilmiştir (Schiaffino ve Amandi, 2009: 193-216).

Sahtekarlık analizi yapan bir diğer çalışmada ise, kullanıcı profili oluşturma sorununa yanıt olarak aşağıdaki adımlar izlenmiştir (Moreau ve diğerleri,1997: 1065-1070):

- Her bir kullanıcı için ilgili parametrelerin bir setinin depolandığı büyük bir veri tabanı kurulur.
- Uygulamadan kullanıcı hareketleri ayıklanır.
- Kullanıcı hareketleri veri tabanına yazılır.
- Veri tabanından alınan bilgilerden bir alarm verilip verilmemesi gerekliliği kontrol edilir.
- Veriler sahtekarlık tespit motoruna aktarılır.
- Analiz için kullanıcı profili kısa vadeli ve uzun vadeli olmak üzere ikiye bölünür.
- Bölünen kısımlar analiz edilerek karşılaştırılması sağlanır.

Profili oluşturan kişi kural koşullarına göre başlatılan şablonların listesine sahiptir. Profil oluşturucuya bir dizi kural ve şablonlar kümesi verilir ve bu kişi verilen kural şablon çiftinin her birinden bir profil oluşturur. Her profil oluşturucunun tipik (sahtekarlık içermeyen) hesap hareketlerinde eğitildiği bir eğitim adımı ve mevcut hesabın tipik davranışlarından ne kadar uzak olduğunu gösteren bir kullanım adımı vardır (Fawcett ve Provost, 1996: 8-13). Eğitim adımı kullanıcının uygulama geçmişindeki hareketlerinden yola çıkılarak, kullanıcı hakkındaki genel bilgilerin çıkarılması ve kullanıcı çizgilerinin belirlenmesi için kullanılır. Kullanım adımı ise kullanıcının mevcut davranışları göz önünde bulundurularak şuan uygulamayı nasıl kullandığı, geçmiş kullanımı ile kıyaslandığında eğitimde belirlenen çizgisinden ne kadar farklılaştığının anlaşılması içindir. Kullanıcı geçmişinden yararlanarak toplanan eğitim verileri aslında uzun vadeli uygulama kullanımını, mevcut durumu belirten kullanım verileri ise kısa vadeli hareketleri içerir.

Uygulamadaki bağlamların toplanması, kullanıcı profili çıkartılması için etkin bir yoldur. Burada söz konusu bağlamlar, sıra dışı hareketlerin belirlenmesi için getirilmiş kuralladır. Bu sebeple uygulamaya özgü bağlamlar kullanılarak

sahtekarlığın mutlak göstergelerindense, sahtekarlık göstergesi olan davranış değişikliklerine karşılık gelen göstergeleri keşfetmek ve müşterilerin normal davranışlarını karakterize etmek için profil oluşturmak gerekir (Fawcett ve Provost, 1997: 291-316).

1.3.2.1.3.2. Risk Analizi

Oxford sözlük tarafından risk; "gelecekte bir zamanda kötü bir şey olma olasılığı; tehlikeli olabilecek veya kötü sonuç alabilecek bir durum" olarak tanımlanır (Oxford University Press, https://www.oxfordlearnersdictionaries.com/, (09.01.2021)). Yani sistemde tehdit oluşturabilecek açıkların her biri risk olarak adlandırılır. Risk ortaya çıktığında işletmenin vizyonunu ve misyonunu olumsuz olarak etkiler, ilgili projenin teslim tarihini yavaşlatır, atakların artmasına yol açar. Bu sebeple riskli durumların yönetilmesi gerekmektedir.

TDK risk yönetimini; "kurum veya işletmelerin çalışmalarını gerçekleştirirken oluşabilecek risklerin önceden dikkatli ve ayrıntılı bir biçimde tanımlanıp değerlendirilmesi, riskleri ortadan kaldıracak veya en aza indirecek önlemlerin alınması süreci" olarak tanımlar (TDK, https://sozluk.gov.tr/, (09.01.2021)). Özellikle sahtekarlık tespiti için risk yönetimi büyük önem taşır. Sistemde mevcut ya da yaşanması muhtemel risklerin belirlenerek risk yöneticileri tarafından özenle analizinin yapılması ve uygun aksiyon planının oluşturularak risklerin en az hasarla ortadan kaldırılması gerekmektedir. Böylece sisteme yapılacak atakların önlenmesi ya da yapılmış atakların belirlenmesi konusunda daha hızlı ve daha güvenilir bir eylem hazırlanmış olacaktır. Risk azaltıcı eylemlerin aynı zamanda sistemdeki sahtekarlık olasılığının azalmasına da fayda sağladığı bilinen bir gerçektir.

Risk yönetimi, dolandırıcılık tespiti ve izinsiz giriş tespiti risklerini tahmin etmek, planlamak, önlemek veya tespit etmek için kullanıcıların davranışlarının (veya hesaplarının) izlenmesini içerir (Fawcett ve diğerleri, 1998: 107). Risk, en azından sistemi etkileyebilecek ve sistemin yönetme yeteneği bulunan olaylar hakkında bilgi eksikliği olmayan bazı bileşenler içerir. Riski anlamak için ilk olarak riskleri tanımlamak ve ayrıştırmak gereklidir (Lockamy ve McCormack, 2010: 593-611). Riskleri anlamak ve sınıflandırmak daha hızlı aksiyon planı almaya yarar. Aksiyon planı ne kadar hızlı oluşturulursa tehdit modeli de o kadar hızlı kurulur. Dolayısı ile risk analizi sahtekarlık tespitinin temel yapı taşını oluşturur.

Risk analizi için gerekli adımlar şu şekilde sıralandırılabilir;

- 1.Adım: Risklerin belirlenmesi.

- 2.Adım: Risklerin sınıflandırılması.
- 3.Adım: Meydana gelme olasılıklarının derecelendirilmesi.
- 4.Adım: Meydana geldiklerinde işletmede yaratacak etki düzeylerinin belirlenmesi.
- 5.Adım: Risk azaltma aksiyonlarının belirlenmesi.
- 6.Adım: Risk planının oluşturulması.

1.3.2.1.3.2.1. Risklerin Belirlenmesi

Risklerin belirlenmesi için mevcut kaynaklar, uygulamanın çalışma şekli ve tüm sistem haritası gözden geçirilir. Sistem sahteci gözünden incelenerek muhtemel atakların yapılabileceği giriş çıkışlar incelenir. Potansiyel sistem açıkları ve meydana gelmesi muhtemel riskler listelenir.

1.3.2.1.3.2.2. Risklerin Kategorize Edilmesi

Uygulamaya göre risk sınıfları önceden belirlenir. Risk sınıflandırması, sistemde hangi türde riskler olduğunu ifade eder. Risk türleri aşağıda verildiği gibi farklı çalışmalarda farklı şekillerde sınıflara ayrılmıştır.

- Kredi veya ödeme gücü riski, likidite riski, piyasa riski, zaman boşluğu riski, dolandırıcılık riski, yönetimsel risk, yasal risk ve sistemsel risk (McAndrews ve James, 1999: 348-357).
- Parasal risk, proje riski, işlevsellik riski, örgütsel risk, rekabetçi risk, çevresel risk, sistemsel risk, teknolojik risk (Benaroch, 2002: 43-84).
- Müşteri ile ilişkili riskler, iletişim ile ilişkili riskler, piyasa riskleri, kaynak riskleri (kaynağa bağımlılık, platforma bağımlılık, hane halkı yatırımları), finansal riskler, teknik riskler (algoritma, platform, grafiksel kullanıcı arayüzü, test stratjileri), yönetimsel riskler, performans riskleri, bakım riskleri, dışsal riskler (Kakkar ve diğerleri, 2013).

1.3.2.1.3.2.3. Meydana Gelme Olasılıklarının Derecelendirilmesi

Risklerin gerçekleşme olasılığının derecelendirilmesi uygulamaya göre değişiklik gösterilebilir. Örnek olarak risklerin meydana gelme olasılıkları aşağıdaki gibi kategorilere ayrılabilir (Güven ve Hartoka, 2019: 1-6).

- Düşük: % 30 ‘dan düşük olma olasılığı.
- Orta: % 30 ile % 70 arasında olma olasılığı.
- Yüksek: % 70 ‘ten büyük olma olasılığı.

Şekil 3: Risk Oluşumundaki Olasılık Sınıfları

Uygulama içinde risklerin meydana gelme olasılıkları şu şekilde yorumlanabilir; Eğer riskin sistemde karşılaşılma olasılığı %70 'den büyük ise yüksek ihtimalle uygulamada görülecektir. Eğer riski sistemde karşılaşılma olasılığı %30 ile %70 arasında bulunursa, bu riskin karşılaşılma olasılığı orta seviyededir denilebilir. Ancak riskin uygulamada görülme olasılığı %30 'dan az ise muhtemelen sistemde bu risk ile karşılaşılmaz ya da karşılaşılma olasılığı düşüktür.

1.3.2.1.3.2.4. Meydana Geldiklerinde İşletmede Yaratacak Etki Düzeylerinin Belirlenmesi

Risklerin etki düzeyleri gerek meydana gelen risklerin etkilerinden gerekse proje yürütücülerinin tecrübelerinden ve mantıksal tahminlerden yola çıkarak uzmanlarca karar verilmelidir. Belirlenen risklerin etki dereceleri aşağıdaki gibi sınıflandırılabilir (Güven ve Hartoka, 2019: 1-6):

- Aşırı: Projede yıkıcı başarısızlığa neden olabilecek risk.
- Yüksek: Proje maliyetini, proje zamanlamasını veya performansını büyük ölçüde etkileme potansiyeli olan risk.
- Orta: Proje maliyetini, proje zamanlamasını veya performansını hafifçe etkileme potansiyeli olan risk.
- Düşük: Maliyet, program veya performans üzerinde nispeten az etkisi olan risk.

1.3.2.1.3.2.5. Risk Azaltma Aksiyonlarının Belirlenmesi

Risklerin etkilerini ortadan kaldırmak ya da en azından minimize etmek için risk azaltıcı faaliyetlere ihtiyaç duyulur. Risk azaltma aksiyonlarının belirlenebilmesi içinde ilk önce risk seviyeleri belirlenmelidir. Risk seviyeleri riskin işletme içindeki genel durumunu ifade eder. Risk seviyesi tablosuna bakıldığında riskin önem seviyesi anlaşılmalıdır. Dolayısı ile risk seviyesi tablosu risklerin olasılıkları ile etkilerinin çarpımından oluşur. Çarpım sonucu elde edilecek değerler aşağıdaki tabloda mevcuttur. Bu tablo risk seviyelerini yani risklerin olasılık ve etkilerinin birleştirilmiş halini göstermektedir (Güven ve Hartoka, 2019: 1-6).

Tablo 1: Seviye: Olasılık ve Etkinin Birleştirilmesi

	Etki				
Olasılık		Düşük	Orta	Yüksek	Aşırı
	Düşük	D	D	C	A
	Orta	D	C	B	A
	Yüksek	C	B	A	A

Mevcut risklerin minimize edilmesi için alınacak aksiyonlar aşağıdaki tabloda sunulmuştur.

Tablo 2: Farklı Seviye Riskler İçin Tavsiye Edilen Aksiyonlar

Seviye	Risk Azaltma Aksiyonları
A	Olasılık ve etkiyi azaltmak için, proje başlar başlamaz öncelikli olarak belirlenecek ve uygulanacak etki azaltma eylemleri
B	Olasılık ve etkiyi azaltmak için, proje yürütülmesi sırasında tanımlanan ve uygulanan eylemler
C	Olasılık ve etkiyi azaltmak için, proje yürütülmesi sırasında tanımlanan ve kaynaklar elverdiği ölçüde uygulanan eylemler
D	Not edilmelidir - derecelendirme zamanla artmadıkça bir eyleme gerek yoktur.

1.3.2.1.3.2.6. Risk Planının Oluşturulması

Risk planı, risk analizinin son durumunu gösterir. Bu tabloda işletme için belirlenen riskler, risklerin karşılaşılma olasılığı, karşılaşıldığında işletmeye olan etkileri ve aksiyon alma durumu bulunur. Yani bu adım önceki 5 adımın özeti niteliğindedir.

Risk analizi tamamlandığında işletmenin saldırı altında karşılaşacağı tablo hazırlanmış olur. Saldırıları en az hasarla atlatabilmek için risk planında bulunan her bir riske karşılık, kullanıcılardan veriler toplanarak kullanıcı profili oluşturulmalı ve riskli kullanıcılar belirlenerek ilgili kullanıcılara şirket politikası ile belirlenen bir yaptırım uygulanmalıdır.

1.3.2.1.3.3. Bağlam Duyarlı Yaklaşım

Abowd ve diğerleri (1999) tarafından yapılan tanıma göre bağlam, bir varlığın durumunu karakterize etmek için kullanılabilecek herhangi bir bilgidir. Bu varlık, kendisi de dahil olmak üzere bir kullanıcı ile bir uygulama arasındaki etkileşim ile ilgili olduğu kabul edilen bir kişi, bir yer veya bir nesnedir (Abowd ve diğerleri, 1999: 304-307).

Bağlam duyarlılığı ise esneklik ve özerklik gereksinimleri karşılanmış özgün bir tasarım yaklaşımı olarak sunulmuştur. Kullanıcı girdilerinde güveni arttırmak, bilgi işleme çevresinde ve kullanıcı gereksinimlerinde dinamik faktörlere adaptasyonu desteklemek amaçlarıyla uygulamalar tarafından bağlam bilgilerinin kullanılmasını içerir (Henricksen, 2003). Kullanıcıya kendi görevlerine bağlı olarak ilgili bilgileri ve/veya servisleri sağlayan bağlamların kullanıldığı sistemler, bağlam duyarlıdır (Abowd ve diğerleri, 1999: 304-307).

Bağlam duyarlı hesaplama, mobil kullanıcıların konumlandıkları çevrede yaşanan değişiklikleri keşfetmeleri ve tepki vermeleri için uygulamaların yeteneğini ifade eder (Schilit ve Theimer, 1994: 22-32). Bağlama duyarlı sistemler, kullanılan konuma, yakınlardaki kişilerin, sunucuların ve erişilebilir cihazların tümüne ve bunların zaman içindeki değişikliklerine adapte olur. Bu özelliklere sahip bir sistem, bilgi işlem ortamını inceleyebilir ve ortamdaki değişikliklere tepki verebilir (Schilit ve diğerleri, 1994: 85-90). Bağlam duyarlılık adaptasyonunu etkinleştirmek için bağlam bilgileri toplanmalı ve sonunda adaptasyonu gerçekleştiren uygulamaya sunulmalıdır. Bu nedenle, bağlam bilgileri için ortak bir temsil biçimi gereklidir (Held ve diğerleri, 2002: 167-180).

Bazı çalışmalarda bağlam duyarlı profilin temsili için gerekli nitelikler şu şekilde sıralanmıştır:

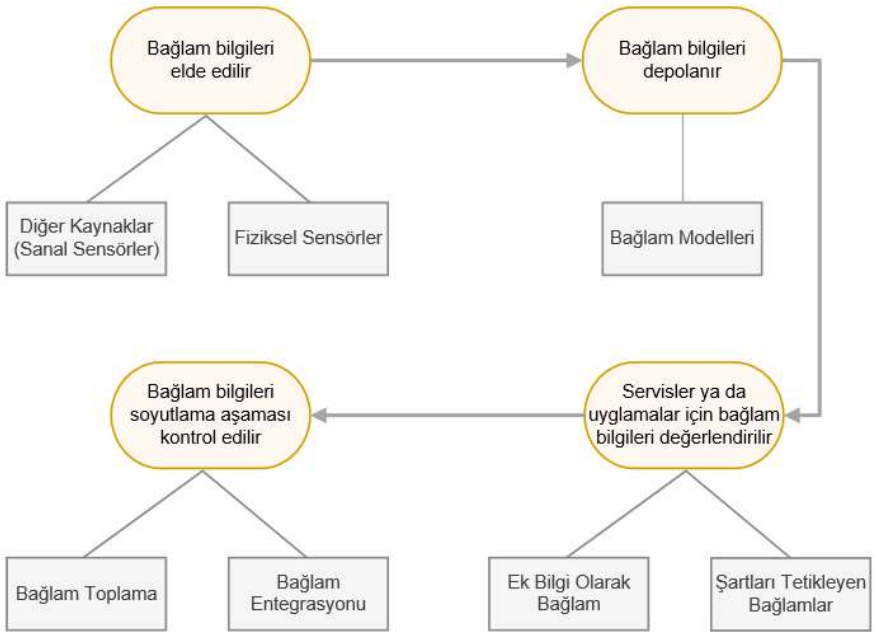
- Yapısal olma, değiştirilebilir, birleştirilebilir/ ayrıştırılabilir, genişletilebilir, standardize edilebilir olma, tek tipte olma (Held ve diğerleri, 2002: 167-180).
- İşlemci gücü, genişletilebilirlik, sistem dayanıklılığı, meta bilgi içerme, bağlam paylaşımı (Hofer ve diğerleri, 2003: 10).

Bağlam duyarlı yaklaşım ile ilgili bir diğer konu da uygulamada bağlamların bulunması ve uygulamanın bağlama duyarlılığının belirlenmesi konusudur. Aşağıdaki soruların yanıtları sistem içerisinde aranarak, uygulamanın bağlamları ve bağlama duyarlılığı belirlenebilir (Schmidt, 2000: 191-199):

- Uygulama kullanımda iken uygulamada neler oluyor? Herhangi bir değişiklik var mı?
- Şartlar (davranış, çevre, koşullar) uygulama için herhangi değerli bir bilgi taşıyor mu? Uygulama için önemli mi?
- Uygulama veya cihaz için kabul edilebilir bir bilgiyi (işlem maliyeti, sensör maliyeti, ağırlığı vb.) yakalamak ve çıkarmak için herhangi bir yol var mı?
- Bilgi nasıl anlaşılır? Yorumlama ve akıl yürütme mümkün ve işe yarar mıdır? Uygulamanın tepki vermesi için uygun yol nedir?

Bağlam duyarlı yaklaşım sürecinin adımları Şekil 4 'de gösterilmiştir. İlk adım, sensörlerden bağlam bilgilerini elde etmektir. Sensörler gerçek dünya bağlam bilgilerini hesaplanabilir bağlam verisine dönüştürür. Fiziksel ve sanal sensörler kullanarak sistem, içeriğe duyarlı çeşitli türde bilgileri yakalayabilir. Sistem daha sonra verileri deposuna saklar. Bağlam verilerini saklarken, bağlam bilgisini temsil etmek için kullanılan model çok önemlidir. Bağlam modelleri çeşitlidir ve her birinin kendine özgü özellikleri vardır. Depolanan içeriği kolayca kullanmak için, sistem veriyi yorumlayarak veya toplayarak verinin soyutlama seviyesini kontrol eder. Son olarak sistem bağlama duyarlı uygulamalar için soyutlanmış bağlam verilerini kullanır (Galar, 2014).

Şekil 4: Bağlam Duyarlı Sistemde Genel Süreç

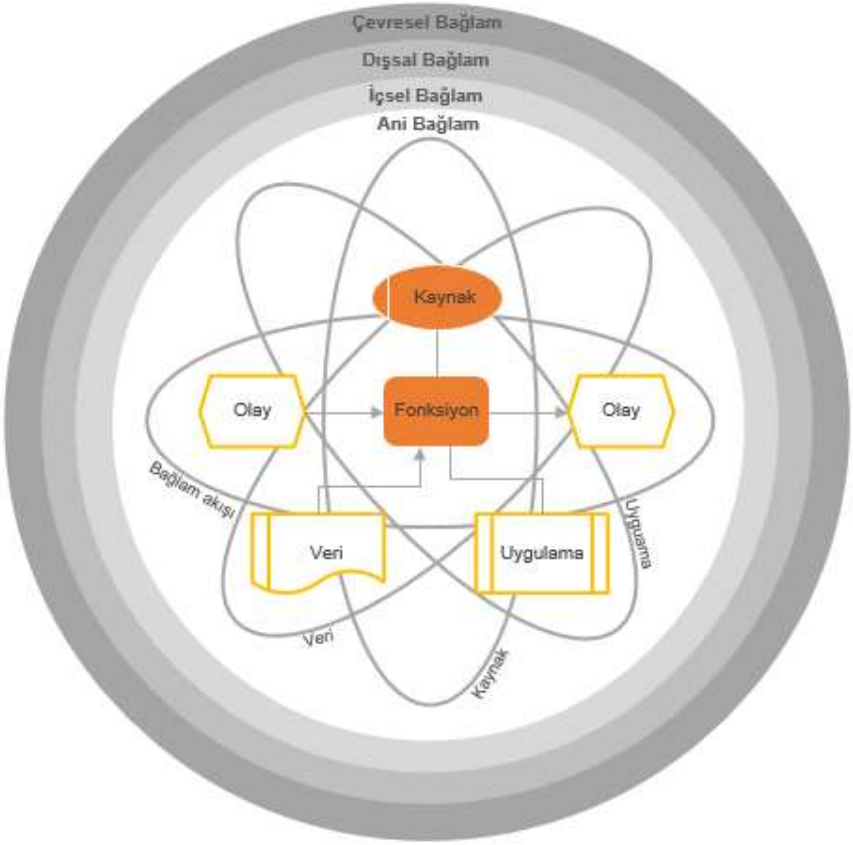


Kaynak: Galar, 2014

Bir uygulamada bağlamlar insan faktörleri ve çevresel faktörlerden yola çıkılarak belirlenebilir. Kullanıcı hakkında bilgi (alışkanlıklar bilgisi, duygusal durum, biyofizyolojik koşullar vb.), kullanıcının sosyal çevresi (başkalarının ortak yerleşimi, sosyal etkileşim, grup dinamiği vb.) ve kullanıcının görevleri (anlık aktiviteler, ilgili görevler, genel hedefler vb.) ile ilgili bağlamlar insan faktörleri altında sınıflandırılmaktadır. Konum (mutlak konum, göreceli konum, ortak konum, vb.), altyapı (hesaplama için çevre kaynakları, iletişim, görev performansı vb.) ve fiziksel koşullar (gürültü, ışık, basınç vb.) ise fiziksel çevre ile ilişkilendirilebilir (Schmidt ve diğerleri, 1999: 893-901).

Bağlam duyarlı yaklaşım iş süreçlerinde sıklıkla kullanılmaktadır. İş süreçlerinde bağlam duyarlı yaklaşım için Rosemann ve diğerleri (2006) tarafından yapılan çalışmada süreç modeli için ani bağlamların, içsel bağlamların, dışsal bağlamların, çevresel bağlamların belirlenerek sırasıyla içten dışa oluşturulan katmanlarla iş sürecinin modellenişini savunulmuştur. Sundukları iş akış modeli ile bağlamların belirlenmesi, dokümante edilmesi, anlaşılması ve ilişkili olanların birleştirilmesi konularında referans sağlamışlardır. Söz konusu model Şekil 5 'de verilmiştir.

Şekil 5: Bağlama Duyarlı Çerçeve



Kaynak: Rosemann ve diğerleri, 2006: 1-10

Bağlam bilgileri çok çeşitli kaynaklardan toplanabilir. Bazı bilgiler mutlaka kullanıcılar tarafından sağlanmalıdır. Bazı bilgilerse donanım ya da yazılımdan elde edilebilir. Bu kaynakların yanı sıra bazı bağlam verileri kullanıcının hareketlerinden, konum bilgilerinden alınabilir (Henricksen ve diğerleri, 2002: 167-180). Bağlam bilgileri; konum, kullanıcı aktiviteleri, zaman, mümkün hesaplamsal kaynaklar seti gibi fiziksel ve bilgi işlem çevresine ilişkin yönleri içerir (Henricksen, 2003). Bağlam duyarlı yaklaşım herhangi bir nesnenin karakteristiğini çıkartmak için kullanılır. Gerek

kullanım amacı gerekse veri toplama kaynakları sebebiyle bağlam duyarlı yaklaşımın, bir sistemin sahtekarlık analizinde kullanılması mümkündür.

1.3.2.1.3.3.1. Bağlam Duyarlı Yaklaşım İle Risk Yönetimi

Risk yönetiminin temel hedefi, risklerin belirlenerek ortadan kaldırılmasıdır. Bu sebeple yapılması gereken, hangi kullanıcının risk taşıdığını belirlemektir. Bunun için bağlam duyarlı yaklaşımdan yararlanılabilir. Her bir riske karşılık gelen en az bir bağlam belirlenerek ilgili bağlam verileri toplanır. Böylece hangi kullanıcının risk taşıyıp taşımadığı anlaşılır. Bağlam duyarlı yaklaşım ile riskleri yönetmek, uygulamaya hem zaman, hem kesinlik kazandırır.

Bağlam duyarlı risk yönetimi modeli, bir değişiklik olduğunda kendini çok çabu şekilde güncelleyebilir. Böylece değişen riskli durumlar ile başa çıkabilir. Bu risk senaryoları, hesaplamalar ve diğer ilgili veriler devamlı olarak bir depoya eklenebilir. Mevcut karar vermenin ve risk yönetiminin yanı sıra, bu depodan elde edilen sonuçlar savunmasız alanları tespit etmek kadar modelin sonraki güncellemeleri için ve yeni uygulamaların planlanmasında kullanılabilir. Riske daha yatkın olan bağlam değişiklikleri, bunların etkileri veya en yüksek risk oranları ve olasılıkları oluşturan bağlam değişiklikleri tespit edilebilecek savunmasız alanlara örnek olarak gösterilebilir (Samad ve diğerleri, 2013: 1378-1385).

1.3.2.1.3.4. Tehdit Puanlaması

Risk puanlaması, bireysel bir müşterinin davranış riskini değerlendirir. Analizden sonra, her işleme bir puan atanır ve daha yüksek bir puan, yüksek sahtekarlık olasılığını gösterir (Wei ve diğerleri, 2013: 449-475). Tehdit puanı, kullanıcı profillerindeki hareketler için makine öğrenmesi algoritmalarından alınan sınıflandırma olasılıklarını gösterir. Diğer bir deyişle kullanıcının yüzde kaç ihtimalle sahtekar kategorisinde bulunma durumunu belirtir. Yani tehdit puanı uygulamanın nihai sonucu niteliğindedir. Tehdit puanları, doğru tahminlerin, gözden kaçırılanların, yanlış alarmların ve doğru tahmin edilmeyen durumların olasılık tablosundan üretilir. Olasılık tablosunu oluşturan durumlar aşağıdaki gibi açıklanabilir (Hamill, 1999: 155-167):

- Doğru tahminler, hem tahminleme sonucunda hem de gerçek durumda belirlenen bir eşik değerinden daha büyük veya eşit olan değerleri gösterir.
- Kaçırılanlar, gerçekte belirlenen eşik değerinin üstündeyken, eşik değerinin altında tahminlenen değerleri gösterir.

- Yanlış alarmlar; gerçekte belirlenen eşik değerinin altındayken, eşik değerinin üzerinde tahminlenen değerleri gösterir.
- Yanlış tahminler; hem tahminleme sonucunda hem de gerçekte belirlenen bir eşik değerinin altında kalan değerleri gösterir.

Tehdit skorunun gösterilmesi, sistemdeki sahteci kullanıcıların yakından takip edilmesi ve kullanıcı hareketlerine ait sonuçların matematiksel olarak izlenebilmesi ihtiyacından doğmuştur. Her bir kullanıcıdan alınan bu skorlar, bir nevi kullanıcıların güvenilirliklerini temsil etmek üzere kullanılır. Tehdit puanından alınan sonuca göre uygulamada belirlenen politika yönetim kararı uygulanır. Bu karara göre kullanıcı sistemden tamamen atılabilir ya da bazı davranışları belirli ölçülerde kısıtlanabilir. Söz konusu karar işletmenin politikasına bağlıdır. Riskli davranışları kısıtlamak amacıyla literatürde genel olarak kara liste yönetimi önerilmiştir.

1.3.2.1.3.5. Kara Liste Yönetimi

Kara liste, sistem kullanıcılarının güvenilir olup olmadığını belirleyen bir duvardır. Duvarın bir kısmında sahtekar olmadığı bilinen, güvenilir kullanıcılar mevcutken diğer kısmında tehdit skorları yüksek hileli hareketler sergileyen kullanıcılar bulunmaktadır. Politika kararına göre duvar kara liste, beyaz liste olarak ikiye ayrılabileceği gibi güvenilir düzeyi aşan ancak risk seviyesinde kalan, tehdit skoru kara liste eşik değerine yakın kullanıcılar için de gri liste belirlenerek kullanıcı grubunu üçe ayırabilir.

Duvar sahibi, kullanıcılardan hangisinin duvarı aşabileceğine ve burada ne kadar süre kalacağına karar vermelidir. Yani duvar sahibi kara liste kurallarını belirlemelidir. Belirlenen kurallarla kara liste yönetimi, kullanıcı profillerini ve çevrimiçi sosyal ağlardaki ilişkiyi temel alarak kullanıcıları engelleyebilecektir. Böylece duvar sahibi kullanıcıları tanımlamak için bir anahtara sahip olacaktır (Ezhilvani ve diğerleri, 2014).

İKİNCİ BÖLÜM

MAKİNE ÖĞRENMESİ YÖNTEMLERİ

2.1. MAKİNE ÖĞRENMESİ

Makine öğrenmesi, bilgisayarların otomatik veri odaklı modeller kurmasını ve mevcut veride istatistiksel olarak anlamlı kalıpların sistematik bir şekilde keşfedilmesiyle programlamayı sağlayan bir yöntemdir (Chowdhury ve diğerleri, 2017). Modelin parametrelerini optimize ederek verilerdeki örüntüleri ve verinin yapısını otomatik olarak bulmak için öğrenme yoluna gidilir (Poggio ve Smale, 2003: 537-544). Buradaki asıl vurgu otomatik kelimesidir. Yani, makine öğrenmesinde genel amaç, anlamlı bir şey üretirken birçok veri kümesine uygulanabilen yöntem arayışı ile ilgilidir. Öğrenme yapılmasındaki amaç, yeni durumlara sürekli olarak uyum sağlayabilmek, problemin değişen kurallarına hızla adapte olabilmektir. Bir makine, tahminler dikkate alındığında rasyonel kararlar ve gelecek veriler hakkında tahminlerde bulunmak için uygun modelleri kullanabilir. Burada belirsizlik temel bir rol oynamaktadır. Gözlemlenen veriler birçok modelle tutarlı olabilir ve bu nedenle verilerde hangi modelin uygun olduğu belirsizdir. Benzer şekilde, gelecekteki veriler ve eylemlerin gelecekteki sonuçları ile ilgili tahminler de belirsizdir. Olasılık teorisini kullanan öğrenmeye dayalı zeki bir sistemde bu belirsizlik durumunun üstesinden gelinebilir (Ghahramani, 2015: 452-459).

Öğrenme algoritması ne olursa olsun, temel bilimsel ve pratik amaç, belirli öğrenme algoritmalarının yeteneklerini ve verilen herhangi bir öğrenme sorununun doğal zorluklarını teorik olarak tanımlamaktır. Bu aşamada genel olarak şu sorulara yanıt aranır (Jordan ve diğerleri, 2015: 255-260):

- Algoritma belirli bir eğitim türü ve hacminden ne kadar doğru bir şekilde öğrenebilir?
- Algoritma, modelleme varsayımlarındaki hatalara veya eğitim verilerindeki hatalara karşı ne kadar sağlamdır?
- Verilen eğitim seti doğrultusunda başarılı bir algoritma tasarlamak mümkün müdür?

Makine öğrenmesi yaklaşımı genellikle iki aşamadan oluşur; eğitim ve test. Eğitim seti makinenin geçmiş ve mevcut verilerden öğrendiği kümedir. Test seti ise makinenin öğrendiklerini uygulayarak öğrenme performansının test edildiği kümedir.

Bu kümeler üzerinde genellikle aşağıdaki adımlar gerçekleştirilir (Buczak ve diğerleri, 2015: 1153-1176):

- Eğitim verilerinden sınıf özelliklerini ve sınıfları tanımlama.
- Sınıflandırma için gerekli niteliklerin bir alt kümesini tanımlama (yani boyutsallığın azaltılması).
- Eğitim verilerini kullanarak modeli öğrenme.
- Bilinmeyen verileri sınıflandırmak için eğitilmiş modeli kullanma.

Makine öğrenmesinin temel amacı, eğitim setindeki örneklerin ötesinde genelleme yapmaktır. Çünkü ne kadar veriye sahip olursak olalım, test zamanı boyunca belirli örnekleri tekrar görmemiz pek olası değildir (Domingos, 2012: 78-87). Bu durumda makinenin genelleme yapabilmesi için yeterli sayıda ve gerçeği iyi bir şekilde yansıtan eğitim örneklerine ihtiyacı vardır. Tüm veri setini makineye iletmektense az ve yeterli veri ile makinenin tüm veri setini genellemesi beklenmelidir.

Bir makine öğrenmesi yöntemi uygulanırken, veri örnekleri uygulamanın temel bileşenlerini oluşturur. Her örnek çeşitli özelliklerle tanımlanır ve her özellik farklı değer türlerinden oluşur. Ayrıca, kullanılan özelliklerin veri türünü önceden bilmek, makine öğrenmesi analizleri için kullanılabilecek araç ve tekniklerin doğru seçilmesine izin verir. Verilerle ilgili bazı sorunlar verilerin kalitesini ve bunları makine öğrenmesi için daha uygun hale getirmek amacıyla ön işleme adımlarını ifade eder. Veri kalitesi sorunları arasında gürültü, aykırı değerler, eksik veya yinelenen veriler ve temsil edilmeyen sapmalı veriler bulunur (Kourou ve diğerleri, 2015: 8-17). Makine öğrenmesi ön veri işleme adımında öncelikli olarak bu sorunlara odaklanılır ardından öğrenme aşaması uygulanır.

Makine öğrenmesi ile ele alınabilen bazı yöntemler aşağıdaki gibi sıralanabilir:

- Planlama
- Kümeleme
- Sınıflandırma
- Regresyon
- Değişken ve boyut indirgeme
- Doğal dil işleme
- Veri madenciliği
- Sezgisel yöntemler
- Ses tanıma
- Örüntü tanıma
- Robotik

- Optimizasyon

2.1.1. Kural Tabanlı Sistemler ile Öğrenme Tabanlı Sistemlerin Karşılaştırılması

Birçok yapay zeka sistemi geliştiricisi, çeşitli uygulamalar için, istenen girdi-çıktı davranışı örneklerini göstererek bir sistemi eğitmenin, tüm olası girdiler için istenen yanıtı tahmin ederek manuel olarak programlamaktan daha kolay olabileceğini kabul etmektedir. Makine öğrenmesinin etkisi, bilgisayar bilimlerinde ve yığın veri bulunan konularla ilgili bir dizi endüstride de yaygın olarak hissedilmiştir (Jordan ve diğerleri, 2015: 255-260). Makine öğrenmesi, sistem geliştiricilerini büyük veri yoğunluğu içerisinde girdi tahmini yapma ve kural tanımlama zorluklarından kurtarabilmektedir.

İnsanlar genellikle analizler sırasında veya muhtemelen, çeşitli özellikler arasında ilişkiler kurmaya çalışırken hata yapma eğilimindedirler. Bu durum, insanların belirli sorunlara çözüm bulmalarını zorlaştırır. Kural tabanlı sistemlerde kurallar insanlar tarafından belirlenebildiği için benzer zorluklar yaşanmaktadır. Makine öğrenimi genellikle sistemlerin verimliliğini artırırken ve makine tasarımlarını geliştirirken başarılı bir şekilde bu problemlere uygulanabilir (Kotsiantis ve diğerleri, 2007, 3-24).

Kural tabanlı yerine öğrenme tabanlı bir sistem tanımlanmasının temel avantajı, çok değişkenli bir problemde tüm kurallarının tanımlanma zorluğunun aşılmasıdır. Kural tabanlı sistemde tekil kurallar kolayca tanımlansa da binlerce kombinasyondan oluşabilecek kuralları tanımlamak, hem zaman hem kaynak açısından oldukça zorlayıcıdır. Üstelik çoğu durumda, problemde insan sezgisi ile belirlenemeyen kurallar mevcuttur. Ancak bu tip kuralların makine tarafından öğrenilmesi çok daha kolaydır. Bilgisayarın insan zekası için zorlayıcı olabilecek problemleri çözebilmesi sebebiyle makine öğrenmesi insan zekasından esinlenen bir yapay zekadır. Buna karşın, insanlar tarafından belirlenen kurallara göre kodlama yapılarak belirlenen kural tabanlı sistemler ise ancak ve ancak sahte zeka olabilir.

Kural tabanlı sistemlerin aksine öğrenme tabanlı sistemler gerçek zamanlı karar verme yapısına sahiptir. Karar tabanlı yöntemlerde tüm kurallar bir şekilde tanımlanabilse bile problemin etkin çözümlenebilmesi için sürekli değişen kurallardan haberdar olup yeni durumlara karşın kuralların güncellenebilmesi gerekir. Ancak gerçek zamanlı karar verme, kural tabanlı sistemlerde pek mümkün değildir. Çünkü bu sistemlerde karar verici insandır ve bu sebeple anlık değişikliklerden kaynaklanan

kurallar belli sayıda tekrar edildikten sonra görülebilir. Büyük veride öğrenme yapabilen makineler ise bu değişiklikleri daha hızlı yakalayabildiği için eş zamanlı kararlar sunabilir.

Sahtekarlık tespiti temel alındığında, kural tabanlı bir sahtekarlık algılama sisteminde, sahtekarlık kalıpları kural olarak tanımlanır. Kurallar bir veya daha fazla koşuldan oluşabilir. Tüm koşullar karşılandığında sistem alarm verir (Rosset ve diğerleri, 1999: 409-413). Makine öğrenmesinde ise kurallar tanımlanmaya gerek olmadan makine tarafından öğrenilir. Kurala bağlı kalınmaksızın sahtekarlık algılanılırsa sistemde yine alarm oluşturulabilir ya da doğrudan belirlenen bir politika uygulanabilir.

2.1.2. Boyutsallık Problemi

Makine öğrenmesi probleminin etkin ve verimli bir çözüme ulaştırılabilmesi için boyutsallık önemli bir etmendir. Makineni öğrenmesi ile çalışan algoritmaların mevcut durumu öğrenebilmesi için geçmiş profili yansıtan yeterli miktarda veri bulunması gerekmektedir. Özellikle veri seti, etiketli verilerden oluşuyorsa eğitim setinde tüm etiket sınıflarının dengeli miktarda verisi bulunmalıdır. Dengesiz bir veri setinde algoritmalar gerektiği gibi problemi öğrenemez, yanlış sonuçlara, yanıltıcı doğruluk oranlarına yol açar. Böyle bir problemde performans ölçüm metriklerine güvenilemez. Boyutsallık sorunu aşağıdaki sebeplerle ortaya çıkabilir:

- Problemin farklı etiketleri için veri miktarının birbirinden farklı olması.
- Veri setinde eksik gözlemler bulunması.
- Bazı değişkenler için veri setinde tek tip veri olması.

Dengesiz veri setlerine ait çözümler veri seviyesinde ve algoritmik seviyede geliştirilmiş olanlar şeklinde kategorilere ayrılabilir. Veri seviyesindeki yöntemler dengesiz veri setlerinin dağılımını değiştirir ve daha sonra azınlık sınıfının tespit oranını iyileştirmek için makine öğrenmesi tekniğine dengeli veri setleri sağlanır. Algoritma düzeyindeki yöntemler mevcut veri madenciliği algoritmalarını değiştirir veya dengesizlik problemini çözmek için yeni algoritmalar ortaya koyar (Han ve diğerleri, 2005: 878-887).

Boyutsallık problemini önleyerek veri setinde dengeli veri yaratabilmek için uygulanabilecek birkaç yöntem mevcuttur. Yukarı örnekleme (ing. over sampling) ya da aşağı örnekleme (ing. under sampling) bu yöntemlerin en sık kullanılanlarıdır.

Yukarı örnekleme basitçe veri çoğaltma işlemi, aşağı örnekleme ise veri azaltma işlemi olarak tanımlanabilir. Veri çoğaltma amacıyla sıklıkla SMOTE (Sentetik

Azınlık Aşırı Örnekleme Teknikleri) yöntemi kullanılır. Bu yöntem azınlık sınıfının, yenisi ile değiştirme ile fazla örnekleme yapmak yerine sentetik örnekler yaratarak fazla örnekleme yapılması, verilerin çoğaltılması amacıyla önerilmiştir. Sentetik örnekler aşağıdaki şekilde üretilir (Chawla ve diğerleri, 2002: 321-357):

- İncelenen özellik vektörü ile en yakın komşusu arasındaki fark ele alınır.
- Bu fark 0 ile 1 arasında rastgele bir sayıyla çarpılır.
- Elde edilen çarpım dikkate alınan özellik vektörüne eklenir. Bu, iki belirli özellik arasında çizgi segmenti boyunca rastgele bir nokta seçimine neden olur. Bu yaklaşım, azınlık sınıfının karar bölgesini daha genel olmaya zorlar.

Sentetik veri yaratma yöntemi gerçek verilerden yola çıkarak yapay verilerin oluşturulması ile gerçekleştirilir. Yapay veriler gerçeklerinden simüle edilmeli, gerçekçi olmalıdır. Yani, gerçek verilerin seçilmiş özelliklerini yansıtmalıdırlar. Bu nedenle, ilk adım olarak, orijinal veriler toplanmalı ve sentetik örneklerde korunması gereken anahtar parametreleri tanımlanmalıdır. Sentetik veriler, gerçek verilerde görünmeyen belirli temel özellikleri göstermek üzere tasarlanabilir. Bu metodolojinin ana avantajı, kontrollü bir ortamda manipülasyon derecesini belirleyebileceğimiz için, hassasiyeti ve belirliliği doğrulamaya olanak sağlamasıdır (Cantú ve Saiegh, 2011: 409-433).

Veri boyutunun dengesizliği ile başa çıkmak için kullanılan bir diğer yöntem ise eksik verileri çoğaltmak ya da fazla verileri silmektir. Dengesiz verileri analiz eden araştırmacının karşılaştığı metodolojik zorluklar nedeniyle, çabalar geleneksel olarak dengeyi dayatmaya odaklanmıştır. Fazla miktarda verisi olan sınıflardan rastgele seçilen gözlemleri silmek ve daha sonra verilerin dengeli bir alt kümesini analiz etmek mümkündür. İstatistiksel olarak geçerli olmasına rağmen, bu yaklaşım mevcut tüm verileri kullanmadığından istenmeyen bir durumdur. Bu nedenle tahminlerin kesinliğini ve hipotez testlerinin gücünü azaltması muhtemeldir (Shaw ve diğerleri, 1993: 1638-1645). Etiketli veriler üzerinde veri silme işlemi yapılacaksa bu işlem azınlık sınıf üzerinden değil çoğunluk sınıfı üzerinden yapılmalıdır. Azınlık sınıfında sahte işlem olarak etiketlenebilecek işlemleri tespit etmek için bir dizi analiz yapılması gerekliliğinden ötürü veriler arasındaki ilişkiler makine öğrenmesi açısından daha kıymetlidir. Ancak bu gibi bir örnekte masum veriler muhtemelen daha fazla olacağından, tekrarlı gözüken verileri silmek test gücünü azaltmadan veri boyutunu dengeler. Verilerin doğası veri azaltmaya izin vermediği durumlarda eksik verileri arttırma işlemi de tercih edilebilir. Bunun için daha önce bahsedilen sentetik veri oluşturma yoluna gidilebilir. Yapay veri üretilmek istenmediğinde ise etiketli sınıfta veri

sayısı az olan sınıf diğer sınıfın veri sayısına ulaşıncaya kadar kopyalanıp tekrar veri setine eklenerek gerçek verilerin çoğaltılması ile yeni dengeli veri seti üretilmesi de mümkündür.

2.1.3. Boyut İndirgeme

Makine öğrenmesinin başarısını etkileyen birçok faktör vardır. Örnek verilerin temsil gücü ve kalitesi her şeyden önce gelir. Uygulamada alakasız ve gereksiz bilgiler çoksa, gürültülü ve güvenilir olmayan veriler ile eğitim aşamasında bilgi keşfi daha zordur. Özellik seçimi, alakasız ve gereksiz olanı tespit etme ve mümkün olduğunca çoğunu kaldırma işlemidir. Bu, verilerin boyutsallığını azaltır ve öğrenme algoritmalarının daha hızlı ve daha etkin çalışmasını sağlar (Hall ve diğerleri, 1998).

Boyut indirgeme algoritmalarının amacı; istatistiksel olarak gereksiz bileşenleri azaltan veya ortadan kaldıran verilerin sıkıştırılmış ve doğru bir gösterimini elde etmektir. Boyut küçültme, bir dizi veri işleme hedefinin merkezinde yer alır. Sınıflandırma ve regresyon problemleri için girdi seçimi, boyut küçültme biçiminde uygulamaya özgüdür (Kambhatla ve Leen, 1997: 1493-1516). Boyut küçültme, ilgili yapıyı (alaka düzeyi genellikle uygulamaya bağlıdır) koruyan yüksek boyutsal verilerin düşük boyutlu bir temsilini üretmeyi hedefler. Boyut küçültme veri biliminde hem görselleştirme hem de makine öğreniminde potansiyel bir ön işleme adımı olarak önemli bir sorundur (McInnes ve diğerleri, 2018: 1802).

Boyut indirgeme; değişken çıkarımı ve değişken seçimi olmak üzere iki şekilde yapılır. Bir sonraki alt bölümde bunlarla ilgili bilgi verilmiştir.

2.1.3.1. Değişken Çıkarımı

Özelliklerin doğasında var olan sınıf hakkındaki bilgi büyük önem taşır. Teorik anlamda, daha fazla özelliğe sahip olmak bize sadece daha fazla seçim gücü verir ancak gerçek dünya bize daha fazla özelliğe sahip olmanın gerekli olmadığını gösteren birçok neden sunar. Birçok çıkarım yöntemi boyutsallık probleminden müzdariptir. Özellikle, bir çıkarım çalışmasındaki değişkenlerin sayısı arttığı için, algoritmanın hesaplamak için ihtiyaç duyduğu zaman da dramatik olarak bazen katlanarak artar. Bu nedenle, verilerdeki özellikler kümesi yeterince büyük olduğunda, birçok algoritmada hesaplama zorlayıcı hale gelir. Bu sorun, bir öğrenme görevindeki birçok özelliğin, bir örneğin sınıfını tahmin etme açısından diğer özelliklerle alakasız veya gereksiz olabileceği gerçeğiyle daha da kötüleşir. Bu bağlamda, bu tür özellikler, hesaplama süresini artırmak dışında hiçbir amaca hizmet etmez. (Koller ve Sahami,

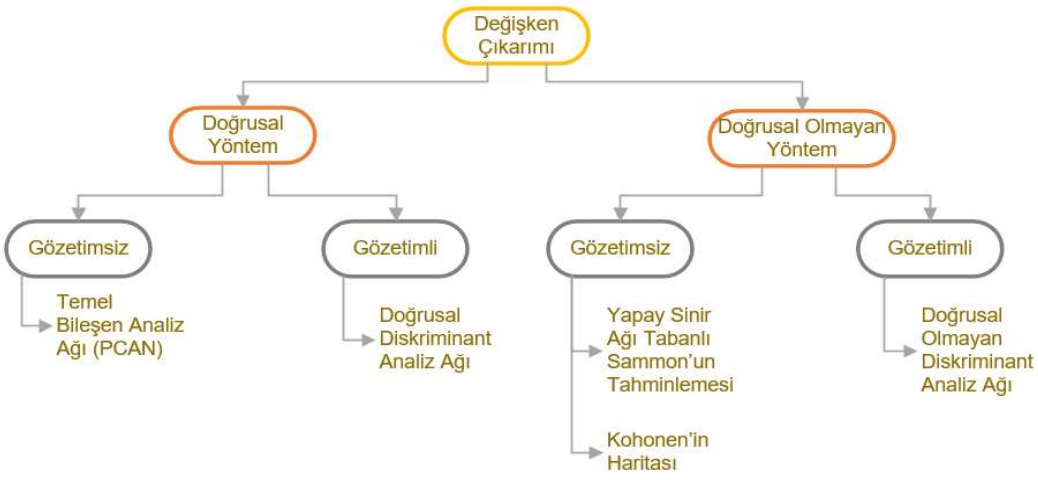
1996). Bu sebeple uygulamada toplanan değişkenler arasında yüksek korelasyonlu değişkenlerin, gereksiz değişkenlerin ve bağımlı değişkeni benzer açıklama gücüne sahip değişkenlerin göz ardı edilmesi gerekmektedir. Böylece daha az ancak daha verimli değişkenlerle hem daha güvenilir sonuçlar elde edilir hem de algoritmanın çalışma zamanından tasarruf edilir.

Boyutsallık probleminden kaçınabilmek amacıyla yapılan özellik çıkarma işlemi ile sınıflandırıcıların genelleme yeteneği geliştirilir ve örüntü sınıflandırmanın hesaplama gereksinimleri azalır. Benzer şekilde özellik çıkarma işlemi; veri tahmini, altta yatan yapıyı daha iyi anlamak, içsel boyutsallığı keşfetmek ve çok değişkenli verilerin kümelenme eğilimini analiz etmek için yüksek boyutlu veriyi görselleştirmemize olanak sağlar (Mao ve Jain, 1995: 296-317).

Özellik çıkarma yaklaşımları, özellikleri daha düşük boyutlara sahip yeni bir özellik alanına yerleştirir ve yeni oluşturulan özellikler genellikle orijinal özelliklerin birleşimidir. Özellik çıkarma, k-ortalama kümeleme, hiyerarşik kümeleme ve temel bileşen analizi gibi çeşitli teknikleri içerir. (Sahu ve diğerleri, 2018: 11).

Şekil 6 'da da görüldüğü gibi değişken çıkarımı, doğrusal ve doğrusal olmayan şeklinde hesaplanabilir. Her iki yöntemin de hem etiketli hem etiketsiz verilerle çalışabilen algoritmaları bulunmaktadır. Sıklıkla kullanılan uygulamalarından biri, veri etiketine gerek duymayan gözetimsiz temel bileşen analizidir.

Şekil 6: Değişken Çıkarımı Algoritmaları



Kaynak: Mao ve Jain, 1995: 296-317

2.1.3.1.1. Temel Bileşen Analizi

Temel bileşen analizi (PCA), korelasyon matrisinin temel bileşenlerini veya özvektörler açısından önemli özelliklerini çıkarır. Bu vektörler gözlemlenen verilerde varyansın ortagonalliğinin ve bağımsızlığının nedenini açıklayan doğrusal kombinasyonlardır. Gözlenen varyansın çoğunluğunu açıklamak için genellikle sadece birkaç ana bileşen gereklidir. İşlevsel bağlantı açısından, temel bir bileşen yüksek korelasyonların olduğu sistemin dağıtılmış bir beynini temsil eder (Friston ve diğerleri, 1993: 5-14). PCA minimum çaba ile karmaşık bir veri setinin, altında yatan basitleştirilmiş yapıyı ortaya çıkartmak için daha düşük bir boyuta nasıl düşürüleceğini gösteren bir yol haritası sunar (Shlens, 2014).

Temel bileşenlerde gelişen anlayış, hem temel bileşen tekniklerinin rutin veri analizine dahil edilmesini mümkün kılmış hem de PCA uygulamalarını yaygın hale getirmiştir. Bu gelişmenin sebebi regresyon analizi ve çok değişkenli kalite kontrol gibi uygulamalarda PCA 'nın oynadığı rol ve yüksek hızlı bilgisayarlar ile uygun yazılım paketlerinin elverişliliğine bağlanabilir (Jackson ve diğerleri, 1979: 341-349).

Bir veri matrisinde PCA birçok hedefe hizmet edebilir (Wold ve diğerleri, 1987: 37-52):

- Sadeleştirme,
- Veri indirgeme,
- Modelleme,
- Sapan gözlem tespiti,
- Değişken çıkarımı,
- Sınıflandırma,
- Tahminleme,
- Ayrıştırma.

PCA 'nın başarısı aşağıdaki iki önemli optimal özellikten kaynaklanmaktadır (Zou ve diğerleri, 2006: 265-286):

- Temel bileşenler sırayla bağımsız değişkenlerin sütunları arasındaki maksimum değişkenliği yakalar, böylece minimum bilgi kaybını garanti eder.
- Temel bileşenler birbiriyle ilişkisizdir, bu nedenle diğer bileşenlere başvurmadan bir temel bileşen hakkında konuşabiliriz.

Temel bileşen analizinin nasıl uygulanacağı Shlens tarafından aşağıdaki şekilde özetlenmiştir (Shlens, 2014):

- Veriler m × n matrisi olarak düzenlenir, burada m ölçüm tipi sayısıdır ve n örnek sayısıdır.

- Her ölçüm türü için ortalama çıkartılır.
- Tekil değer ayrışımı veya kovaryansın özvektörleri hesaplanır.

PCA, kovaryans matrisinin S özdeğer ayrışması veya X, S = XTX / (m - 1) veri matrisinin tekil değer ayrışması ile hesaplanabilir (Ku ve diğerleri, 1995: 179-196).

$X' = [X_1, X_2, \dots, X_p]$ random vektörü için doğrusal kombinasyon aşağıdaki gibidir (Johnson ve diğerleri, 2002).

$$Y_p = a'_p X = a'_{p_1} X_1 + a'_{p_2} X_2 + \cdots + a'_{p_p} X_p$$

Herhangi bir bağımlı değişkenin varyansı ve başka bir bağımlı değişkenle kovaryansı aşağıdaki gibidir.

$$Var(Y_i) = a'_i \sum a_i \quad i = 1,2, \dots, p$$

$$Cov(Y_i, Y_k) = a'_i \sum a_k \quad i, k = 1,2, \dots, p$$

i.temel bileşen = Var ($a'_i X$) ve Cov($a'_i X, a'_k X$) 'ı maksimize eden $a'_i X$ doğrusal kombinasyonudur.

Var (a'_iX) , $a'_i a'_i$ = 1,

Cov(a'_iX, a'_kX) = 0, k <1 için.

Random vektörü X 'in kovasyans matrisi Σ olmak üzere, kovaryans matrisinin öz değer-öz vektör çiftleri şu şekildedir:

$$(\lambda_1, e_1), (\lambda_2, e_2), \dots, (\lambda_p, e_p) \quad , \quad \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p$$

Bu durumda i.temel bileşen denklemi aşağıdaki gibi ifade edilebilir.

$$\mathrm{Y}_i = e'_i \mathrm{X} = e'_{i_1} \mathrm{X}_1 + e'_{i_2} \mathrm{X}_2 + \cdots + e'_{i_p} \mathrm{X}_p \ , \qquad i = 1,2, \dots, p$$

$$Var(Y_i) = e'_i \sum e_i = \lambda_i \quad i = 1,2, \dots, p$$

$$Cov(Y_i, Y_k) = e'_i \sum e_k = 0 \quad i \neq k.$$

2.1.3.2. Değişken Seçimi

Değişken seçiminin temel amacı, çalışan algoritmaların performansını önemli ölçüde düşürmeden, (d<D olmak üzere) verilen D adet ölçüm kümesinden d adet değişken alt kümesi seçmektir. (Pudil ve diğerleri, 1994:1119-1125).

Değişken seçiminin bilinen yöntemleri, bazı indeksler kullanarak ve değişkenler arasından en iyilerini seçerek farklı özellik alt kümelerinin değerlendirilmesini içerir. İndeks, genellikle seçim sürecinin gözetimli ya da gözetimsiz olmasına bağlı olarak sınıflandırma ya da kümelemede ilgili seçimin yeteneğini ölçer. Bu yöntemin problemi, büyük veri setine uygulandığında aramada

büyük bir hesaplama karmaşası oluşmasıdır. Söz konusu karmaşa, kapsamlı bir arama için veri boyutu açısından üstel olarak artar (Mitra ve diğerleri, 2002: 301-312).

Aralarında yüksek korelasyon bulunan bağımlı değişkenler, sınıflar hakkında hiç bir ekstra bilgi sağlamaz ve böylece tahminci için gürültü görevi görür. Bu, toplam bilgi içeriğinin, sınıflar hakkında maksimum ayrım bilgisi içeren daha az benzersiz özellikten elde edilebileceği anlamına gelir. Dolayısıyla bağımlı değişkenleri ortadan kaldırarak, veri miktarı azaltılabilir. Böylece sınıflandırma performansında iyileşme sağlanabilir. Bazı uygulamalarda, saf gürültü olarak işlev gören, sınıflarla hiçbir ilişkisi olmayan değişkenler, tahmincide yanlılık yaratabilir ve sınıflandırma performansını düşürebilir. Bu problemler, incelenen süreç hakkında bilgi eksikliği olduğunda meydana gelir. Değişken seçim teknikleri uygulayarak süreç hakkında fikir edinilerek, tahmin doğruluğunun geliştirilmesi mümkündür (Chandrashekar ve Sahin 2014: 16-28).

Değişken seçiminin birçok potansiyel faydası vardır. Uygulamaya önemli katkıları aşağıdaki gibi sayılabilir:

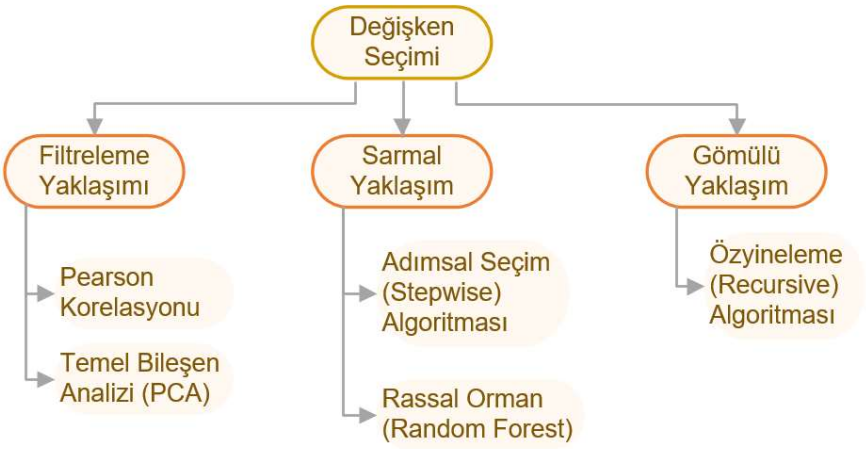
- Veri görselleştirmeyi kolaylaştırmak.
- Veriyi anlamak, veri analizini kolaylaştırmak.
- Ölçüm ve depolama gereksinimlerini azaltmak.
- Eğitim ve kullanım sürelerini azaltmak.
- Tahmin performansını iyileştirmek, tahmin yanlılığını azaltmak.
- Boyutsallık problemini azaltmak.

Özellik seçimi, aşağıdaki maddelere göre minimum boyutta özellik alt kümesini seçmeye çalışır. Özellik seçim kriterleri şunlar olabilir (Dash ve Liu, 1997:131-156):

- Sınıflandırma doğruluğu önemli ölçüde azalmamalıdır.
- Sadece seçilen özellikler için değerler verildiğinde, sonuçta ortaya çıkan sınıf dağılımı verilen tüm özellikler göz önüne alındığında, orijinal sınıf dağılımına mümkün olduğunca yakın olmalıdır.

Değişken seçimi için kullanılan yöntemler filtreleme, sarmal ve gömülü olmak üzere üç farklı yaklaşım ile sınıflandırılabilir. Şekil 7 ‘de değişken seçimi için sınıflandırılan yaklaşımlar ve bu yaklaşımların sıklıkla kullanılan yöntemleri verilmiştir.

Şekil 7: Değişken Seçimi Algoritmaları



2.1.3.2.1. Filtreleme Yaklaşımı

Filtreleme yöntemleri, indirgeme meydana gelmeden önce ilgisiz özellikleri çıkarabilecek bir ön işleme adımı olarak tanımlanır (Weston ve diğerleri, 2001: 668-674). Değişken seçimine yönelik filtre yaklaşımında, değişken altkümesi ön işleme adımı olarak seçilir. Burada değişken seçimi, verinin özelliklerine göre ve çıkarım algoritmasından bağımsız olarak yapılır (Kohavi ve Sommerfield, 1995: 192-197).

Değişken seçimi için filtre yaklaşımı ile çıkarım algoritmasını göz ardı eden verilerden özelliklerin değerleri değerlendirilmeye çalışılır. Yaklaşımın temel dezavantajı, seçilen özellik alt kümesinin çıkarım algoritması performansı üzerindeki etkilerini tamamen yok saymasıdır (Kohavi ve John, 1997: 273-324).

Özelliklerin bağımsız olarak filtrelendiği modelde, tüm değişken girdilerinden oluşan kümeler belirlenir. En iyi özellik kümesi seçilir ve seçilen girdi kümesi çıkarım algoritmasına aktarılır. Filtreleme modelindeki adımlar temel olarak Şekil 8 'de özetlenmiştir.

Şekil 8: Özellik Filtreleme Modeli

Kaynak: John ve diğerleri,1994: 121-129

Algoritma tabanlı filtre modelinde, öğrenme aşamasından önce özellikleri değerlendirmek için verilerin gerçek özellikleri incelenir. Filtre tabanlı yaklaşımlar neredeyse her zaman sınıf etiketlerine dayanır, yani denetimli öğrenme tabanlıdır. Filtreleme yaklaşımının yöntemleri genel olarak özellikler ve sınıf etiketi arasındaki korelasyonları değerlendirir (He ve diğerleri, 2006: 507-514). Filtreleme çok kullanışlı

bir değişken seçim mekanizmasıdır. Her bir değişken ve hedef değişken arasındaki karşılıklı ilişkileri ölçerek çalışır. Bu nedenle, daha fazla analiz için hangi değişkenlerin tutulması gerektiğinin hızlı bir şekilde taranmasına izin verirler (Baesens ve diğerleri, 2015).

Filtreleme yaklaşımında çeşitli özellik sıralaması ve özellik seçme teknikleri önerilmiştir. Önerilen bazı teknikler şu şekilde sıralanabilir; Korelasyon tabanlı özellik seçimi, temel bileşen analizi, kazanç oranı özellik değerlendirme, ki-kare özellik değerlendirme, hızlı korelasyon tabanlı özellik seçimi, bilgi kazancı, öklid mesafesi, markov battaniye filtresi (Karegowda ve diğerleri, 2010: 13-17).

2.1.3.2.1.1. Pearson Korelasyonu

Literatürde önerilen yöntemleren biri de pearson korelasyon katsayısıdır. Karl Pearson korelasyonu, iki değişken arasındaki doğrusal ilişkinin kuvvetini ve yönünü ölçer. Bir değişkenin doğrusal olarak başka bir değişkenle ilişkili olduğu yönü ve derecesini tanımlar (Bolboaca ve Jäntschi, 2006: 179-200).

Pearson korelasyonu ρ_P aşağıdaki gibi hesaplanır.

$$\rho_P = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}}$$

Korelasyon çıktısı −1 ile +1 arasında değişir. Bir filtre olarak uygulamak için, pearson korelasyonunun 0 ‘dan (p değerine göre) önemli ölçüde farklı olduğu tüm değişkenler seçilir veya örnek olarak şu şekilde seçim yapılabilir: | ρ_P| > 0,50.

2.1.3.2.2. Sarmal Yaklaşım

Sarmal model, çıkarım, indirgeme algoritmasını kullanır (Kohavi ve John, 1997: 273-324). Sarmal modeli teknikleri, özellikleri, öğrenme algoritmasını kullanarak değerlendirir. Özellik seçim yöntemlerinin çoğu sarıcı yöntemlerdir.

Filtreleme yaklaşımında özellik seçim yöntemleri öğrenme algoritmalarından bağımsızdır, oysa sarmal yaklaşım, özellik alt kümesinin değerlendirmesinin bir parçası olarak öğrenme algoritmasını kullanır (Godoy ve Amandi, 2005: 329).

Bir sarmal yöntemde, sınıflayıcı, bir özellik sıralaması veren bir kara kutu olarak kullanılır. Bu nedenle özelliklerin sıralanmasını sağlayabilen herhangi bir sınıflayıcı kullanılabilir. Pratik nedenlerden dolayı, bu problemde kullanılan bir sınıflandırma hem hesaplama açısından verimli hem de muhtemelen kullanıcı tanımlı

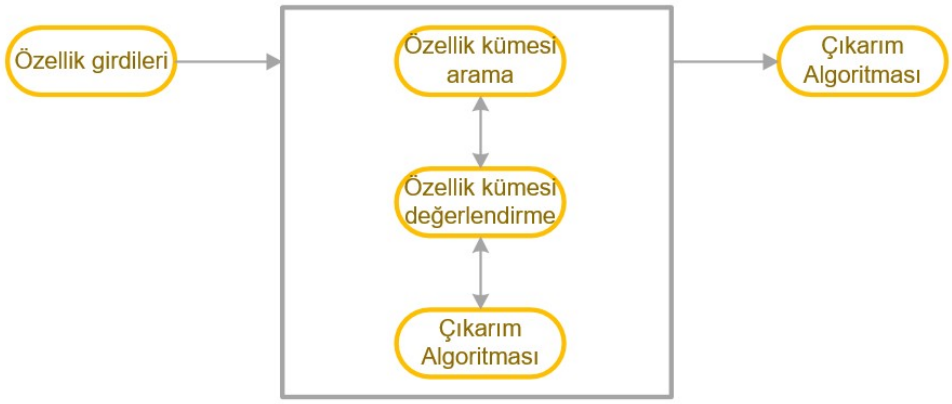
parametreler olmadan yararlanılacak şekilde basit olmalıdır (Kursa ve Rudnick, 2010: 1-13).

Sarmal yöntem, belirli bir özellik alt kümesinin iyilik ölçüsü olarak bir indirgeme algoritmasından tahmin edilen doğruluğu kullanır. Elde edilen doğruluk ile değişken alt kümelerinin uzayında arama yapar (Weston ve diğerleri, 2001: 668-674).

Önceden tanımlanmış bir sınıflandırıcı göz önüne alındığında, tipik bir sarmal modeli aşağıdaki adımları gerçekleştirir (Tang ve diğerleri, 2014: 37). Bu adımlar Şekil 9 ile özetlenmiştir.

- Adım 1: Bir özellik alt kümesi aranır ve seçilir.
- Adım 2: Seçilen özellik alt kümesi, sınıflandırıcının performansına göre değerlendirilir.
- Adım 3: İstenilen kaliteye ulaşılana kadar Adım 1 ve Adım 2 tekrarlanır.

Şekil 9: Özellik Seçiminde Sarmal Model



Kaynak: John ve diğerleri,1994: 121-129

Seçilen arama uzayının her bir aşaması bir değişken alt kümesini temsil eder. N sayıda değişken için, her bir aşamada n adet ikili değer vardır ve her ikili değer bir özelliğin mevcut olup (1) olmadığını (0) belirtir. Operatörler, aşamalar arasındaki bağlantıyı belirler ve arama uzayına karşılık gelen bir aşamadan bir özellik ekleme veya silme işlemi yapar. Bu yaklaşım istatistikte adımsal (ing. stepwise) yöntemlerde yaygın olarak kullanılır (Kohavi ve John, 1998: 33-50).

Adımsal seçim ve rassal orman sarmal yaklaşıma göre değişken seçen önemli algoritmalar arasındadır. Bir sonraki alt bölümde bunlara yer verilmiştir.

2.1.3.2.2.1. Adımsal Seçim Algoritması

Adımsal seçim veya aşamalı eleme olarak bilinen yaklaşım, arama yolunu açıkça takip etmeden önceki bir kararı geri çekmenize izin veren her bir karar noktasındaki özelliklerin hem eklenmesi hem de kaldırılması ile ilgilenir (Blum ve Langley, 1997: 245-271). Adımsal regresyon, önce regresyon uygulayan, ardından sırayla değişkenleri ekleyip çıkartarak tüm farklı kümeler arasında etkin bir arama yapmak için regresyon bilgilerini kullanan bir açgözlü (ing. greedy) algoritmadır (Becker, 2015). Bu değişken seçim yöntemi, değişkenler analize eklendikçe veya analizden düştükçe ortalama değişimlerini dikkate alarak değişkenlerin doğrudan verimlilik üzerindeki etkisini ölçer (Wagner ve Shimshak, 2007: 57-67). Adımsal seçim, regresyon modelindeki diğer değişkenler için eşzamanlı olarak ayarlanan, istatistiksel olarak önemli bir etkiye sahip değişkenleri belirler (Steyerberg ve diğerleri, 1999: 935-942).

Adımsal seçimin çalışma şekli şu şekilde açıklanabilir; her bir bağımsız değişken gruba girdikten sonra bir adım eklenir. Ardından, grubun her birinin içindeki değişkenler açısından uygulamaya dahil etmeye değer olup olmadığını görmek için yeniden araştırılır. Diğer bir deyişle, ileride herhangi bir adımda bağımsız değişkenin bir alt kümesi tahmin değerlerinin çoğunu içeriyorsa, ilgili bağımsız değişken uygulamadan çıkartılabilir (Pope ve Webster, 1972: 327-340).

Adımsal seçim yöntemi temelde F-testine dayanarak hangi değişkenin en yüksek F puanına sahip olduğuna karar verir ve daha sonra bu değişkeni modele ekler veya F puanı düşük olan tahminciyi modelden siler. Tahminci değerlendirmesi modele eklenecek ya da modelden çıkartılacak hiçbir değişken kalmayıncaya dek devam eder. Böylece değişken seçimi gerçekleştirir. Adım adım ileri seçim ve geri eleme süreçlerinin bir kombinasyonu olarak oluşur.

Adımsal değişken seçimi yöntemi için kullanılan formülasyon aşağıda verilmiştir (Becker, 2015):

$$F = \frac{(RSS_0 - RSS_1)/(p_1 - p_0)}{RSS_1/(P - p_1 - 1)} \quad , \qquad F(p_1 - p_0\,, N - p_1 - 1)$$

RSS_0: ilk model için hata kareler toplamı.

RSS_1: yeni model için hata kareler toplamı.

p_0: ilk modeldeki parametre sayısı,

p_1: yeni modeldeki parametre sayısı.

P: toplam değişken sayısı.

F (p_1 - p_0, N - p_1 - 1) dağılımın serbestlik derecesini gösterir.

2.1.3.2.2.2. Değişken Seçiminde Rassal Orman

Rassal ormanlar (RF), güvenilir bir gösterge olduğu gösterilen permütasyon önemi ölçüsüne dayanarak önemli değişkenlerin seçimi için ölçümler sunmaktadır (Chehata ve diğerleri, 2009). RF yönteminde mevcut olan en gelişmiş değişken önem ölçüsü "permütasyon doğruluk önemi" ölçüsüdür. Bunun mantığı, bir tahminci değişkenin değerlerine rastgele izin vererek, bağımsız değişkenin bağımlı değişkenle olan orijinal ilişkisinin bozulmasıdır (Strobl ve diğerleri, 2009: 323).

RF yöntemine göre ormandaki her bir ağaç için, örnek dışı gözlemler için bir yanlış sınıflandırma oranı bulunur. Belirli bir tahminci değişkenin önemini değerlendirmek için değişkenin değerleri örnek dışı gözlemler için rassal olarak değiştirilir. Ardından değiştirilmiş örnek dışı veriler yeni tahminler almak için ağaca aktarılır (Cutler ve diğerleri, 2007: 2783-2792).

RF ile değişken seçimi, budanmamış ağaçlar arasındaki korelasyonu azaltır ve yanlılığı düşük tutar. Kesilmemiş ağaçlardan oluşan bir topluluk alındığında varyans da azalır. RF yönteminin bir başka avantajı, öngörülen çıktının sadece bir kullanıcı tarafından seçilen parametreye, her bir düğümde rastgele seçilecek tahmincilerin sayısına bağlı olmasıdır (Prasad ve diğerleri, 2006: 181-199).

Değişken seçimi için rassal orman uygulama adımları aşağıdaki gibi önerilmiştir (Genuer ve diğerleri, 2010: 2225-2236);

- Adım 1. Ön eleme ve sıralama:
  - RF önem puanları hesaplanır, küçük öneme sahip değişkenler çıkartılır.
  - Kalan m değişkenlerin önem sırası azalan sırada sıralanır.
- Adım 2. Değişken seçimi:
  - Yorumlama için: k=1 'den m 'e kadar ilk k değişkeni içeren RF modelin yuvalı koleksiyonu oluşturulur ve en küçük hataya sahip değişkenler seçilir.
  - Tahmin için: Yorumlama için alınan sıralı değişkenlerden başlayarak, değişkenler aşamalı olarak test edilerek artan sırayla RF modeli oluşturulur. Son modelin değişkenleri seçilir.

2.1.3.2.3. Gömülü Yaklaşım

Filtre modelleri, sınıflandırıcıdan bağımsız olan özellikleri seçer. Tipik bir sarmal model ise çapraz doğrulama adımından kaçınır, bu nedenle hesaplama açısından etkilidirler. Ancak sınıflayıcının sapmasını hesaba katmazlar. Gömülü model ise filtreleme ve sarmal model arasındaki boşluğu dolduran bir yöntem olarak önerilmiştir. Gömülü model genellikle hem sarmal modele göre karşılaştırılabilir

doğruluk hem de filtre modeline göre karşılaştırılabilir verimlilik sağlar. Gömülü model öğrenme zamanında özellik seçimi yapar (Tang ve diğerleri, 2010: 1-44).

Bu yöntemlerin en popüler örnekleri, aşırı uyumu azaltmak için dahili cezalandırma işlevlerine sahip En Küçük Mutlak Küçültme ve Seçim Operatörü (LASSO) ve Ridge regresyonudur (Saurav, 2016).

- LASSO regresyonu, katsayıların büyüklüğünün mutlak değerine eşdeğer ceza ekleyen L_1 düzenlemesi yapar.
- Ridge regresyonu, katsayıların büyüklüğünün karesine eşdeğer ceza ekleyen L_2 düzenlemesi yapar.

Bir diğer çalışmaya göre ise bu yönetimin popüler bir örneği, destek vektör makinası tabanlı tekrarlamalı algoritmadır (SVM-RFE) (Sahu ve diğerleri, 2018: 11).

2.1.3.2.3.1. Özyineleme Algoritması

Özyinelemeli özellik seçimi (ing. recursive feature elimination) (RFE), özellik setini sıralar ve sınıflandırmaya en az katkıda bulunan sıralamanın altındaki özellikleri çıkartır (Zeng ve diğerleri, 2009: 1205-1208). RFE, başlangıçta bir modelin tüm değişkenlerle oluşturulduğu bir tekniktir. Daha sonra algoritma, yeterli sayıda özelliğe ulaşana kadar en zayıf özellikleri tek tek kaldırır. Bu şekilde değişken seçimi tamamlanmış olur.

RFE döngüsü şu şekilde sıralanabilir: Mevcut veri kümesi için önceden tanımlanmış özellik ölçüm kuralları, tüm özelliklerin sıralama puanlarını hesaplamak için kullanılır ve sıralama puanı bir özelliğin sınıflandırma yeteneğini temsil eder. En düşük puana sahip özellik kaldırılır. Kalan özellikler için sıralama puanı yeniden hesaplanır ve geçerli en düşük puanı alan özellik kaldırılır. Bu işlem, özellik kümesinde yalnızca bir özellik kalana kadar tekrarlanır (You ve diğerleri, 2014: 1463-1475).

SVM-RFE, her bir özelliğin ağırlığını mevcut öğrenme modelinin destek vektörlerine göre hesaplayan bir yöntemdir (Lin ve diğerleri, 2012: 149-155).

SVM-RFE özellikleri değerlendirmek için Destek Vektör Makinesi (SVM) modellerindeki katsayılardan türetilen ölçütleri kullanır ve küçük ölçütlere sahip özellikleri yinelemeli olarak çıkartır. SVM-RFE, değişkenler arasındaki bağımlılıkları modelleyebilir. Sarmal yönteme kıyasla SVM-RFE, seçim kriteri olarak eğitim verilerindeki çapraz geçerlilik doğruluğunu kullanmaz. Böylece, aşırı uyuma daha az eğilimlidir ve eğitim verilerinin tamamını kullanabilir. Özellikle çok fazla aday değişken olduğunda çok daha hızlıdır (Yan ve Zhang, 2015: 353-363).

Değişken seçiminde tekrarlama gereklidir. Çünkü her bir özelliğin göreceli önemi, aşamalı eleme işlemi sırasında (özellikle yüksek derecede ilişkili değişkenler için) farklı bir değişken alt kümesi üzerinde değerlendirildiğinde önemli ölçüde değişebilir. Özelliklerin elimine edildiği sıra, son bir sıralama oluşturmak için kullanılır. Özellik seçme süreci, sadece bu sıralamadan ilk n özelliği almaktan ibarettir (Granitto ve diğerleri, 2006: 83-90).

RFE sırasıyla aşağıdaki iteratif prosedürleri içerir (Guyon ve diğerleri, 2002: 389-422):

- Sınıflandırıcı eğitilir.
- Tüm özellikler için sıralama ölçütü hesaplanır.
- En küçük sıralama kriterine sahip özellik kaldırılır.

2.1.3.2.3.2. En Az Mutlak Küçülme Ve Seçme Operatörü

Düzenleme terimi tarafından cezalandırılan deneysel bir hatanın en aza indirilmesiyle seyrek yani olası tüm seçenekler arasında sıfır olmayan tahminlerin bulunduğu doğrusal modellerin tahmin edilmesi, istatistik ve makine öğreniminde çok popüler ve başarılı bir yaklaşımdır. Veri ayarlama ve düzenleme arasındaki dengeyi kontrol ederek, çok büyük boyutta bile iyi istatistiksel özelliklere sahip tahmin ediciler elde edilebilir. Ceza terimi ile bunu başaran örneklerden biri en az mutlak küçülme ve seçme (LASSO) yöntemidir (Jacob ve diğerleri, 2009: 433-440). Tibshirani tarafından önerilen LASSO, bazı katsayıları küçültür ve diğerlerini 0 'a atar. Böylece hem alt küme seçiminin hem de RIDGE regresyonunun iyi özelliklerini korur (Tibshirani, 1996: 267-288). Bu yöntem önemli bağımsız değişkenleri etkin bir şekilde seçebilir ve regresyon parametrelerini aynı anda tahmin edebilir. Değişken sayısı, veri sayısından fazla olsa bile değişken seçiminde başarı sağlayabilir. LASSO, katsayı değerlerini cezalandırmak yerine mutlak değerleri cezalandırır.

LASSO, regresyon katsayılarına L_1 ceza uygulayan bir cezalandırılmış en küçük kareler yöntemidir. L_1 cezasının niteliği nedeniyle, LASSO hem sürekli küçülme hem de otomatik değişken seçimini aynı anda yapar (Zou ve Hastie, 2005: 301-320). L_1 iyi bir ceza fonksiyonu olma özelliklerini sağlamaktadır. Fan ve Li 'ye göre iyi bir ceza fonksiyonu aşağıdaki özellikleri içerir (Fan ve Li, 2001: 1348-1360).

- Sapmasızlık: Gerçek bilinmeyen parametre büyük olduğunda, gereksiz modelleme yanlılığından kaçınmak için ortaya çıkan tahminci neredeyse sapmasızdır.
- Seyreklik: Ortaya çıkan tahmin edici, küçük tahminlenen katsayıları model karmaşıklığını azaltmak için otomatik olarak 0 'a ayarlayan bir eşik kuralıdır.

- Süreklilik: Sonuç tahmincisi, model tahmininde kararsızlığı önlemek için veri süreklidir.

Model seçimi amacıyla LASSO kullanmak için, LASSO tarafından verilen seyrek modelin gerçek modelle ne kadar iyi ilişkili olduğunu değerlendirmek gerekir. Bu değerlendirmeyi Zhao ve Yu doğrusal modellerde LASSO 'nun model seçim tutarlılığını araştırarak yapmıştır. İlgili çalışmada büyük miktarda veri verildiğinde LASSO 'nun hangi koşullar altında gerçek modeli seçip seçemeyeceği araştırılmıştır (Zhao ve Yu, 2006: 2541-2563).

Aşağıda LASSO modelinin genel formülasyonu verilmiştir:

$y = (y_1,...,y_n)^T$: Bağımlı değişken,

$x_j = (x_{1j},...,x_{nj})^T$: Bağımsız değişken,

λ : negatif olmayan LASSO düzenleme parametresi,

β : parametre vektörü,

$$\hat{\beta} = \underset{\beta}{arg\min} \left\| y - \sum_{j=1}^{p} x_j \beta_j \right\|^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

Burada LASSO, λ arttıkça katsayıları sürekli olarak 0 'a çeker ve λ ceza parametresinin yeterince büyük değerleri için, $\hat{\beta}$ 'in bazı bileşenlerini tam olarak 0 'a ayarlar. Ayrıca, sürekli küçülme sapma-varyans dengesi açısından genellikle tahmin doğruluğunu arttırır (Zou, 2006: 1418-1429).

Hedef değişken ikili olduğunda klasik LASSO uygulamaktansa lojistik regresyona cezalandırma terimi ekleyerek lojistik LASSO kullanmak, uygulamaya avantaj sağlayabilir. Cezalandırılmış regresyon, yüksek düzeyde ilişkili verilerde ve örneklem büyüklüğünden çok daha fazla sayıda öngörücüye sahip verilerde daha kararlı sonuçlar üretme avantajına sahiptir (Shofiyah ve Sofro, 2018). Lojistik regresyon için LASSO ile cezalandırılmış yöntem, negatif log-olabilirlik fonksiyonuna ceza terimi eklenerek elde edilir (Algamal ve Lee, 2015: 9326-9332).

2.2. MAKİNE ÖĞRENMESİNDE SONUÇLARIN DEĞERLENDİRİLMESİ

2.2.1. Karmaşıklık Matrisi

Hata matrisi olarak da bilinen karmaşıklık matrisi sınıflandırma probleminde sınıflayıcıların doğruluğunu tespit etmek için kullanılır. Karmaşıklık matrisi (n x n) boyutlu, satırları referans verilen sınıfları ve sütunları tahmin edilen sınıfları temsil eden bir kare matristir. Köşegen elemanlar doğru sınıflandırılmış örnek sayısını temsil

eder. Köşegen olmayan elemanlar ise sınıflandırma hatalarının sayısını temsil eder (Ballabio ve diğerleri, 2018: 33-44).

Bir sınıflandırmanın doğruluğu, doğru tanınan sınıf örneklerinin (gerçek pozitifler), sınıfa ait olmayan doğru tanınan örneklerin (gerçek negatifler) ve sınıfa yanlış atanan örneklerin (yanlış pozitifler) veya sınıf örneği olarak tanınmayan örneklerin (yanlış negatifler) hesaplanmasıyla değerlendirilebilir. Bu dört nicelik, ikili sınıflandırma durumunda aşağıdaki tabloda gösterilen bir karmaşıklık matrisini oluşturur (Sokolova ve Lapalme, 2009: 427-437).

y_0 normal işlemler kümesi, y_1 hileli işlemler kümesi, $\widehat{y_0}$ normal işlemler tahmini, $\widehat{y_1}$ hileli işlemler tahmini olmak üzere, ikili sınıflandırma problemi için modellenebilecek geleneksel karmaşıklık matrisi aşağıdaki tabloda sunulmuştur (Dal Pozzolo ve diğerleri: 4915-4928).

Tablo 3: Karmaşıklık Matrisi

	Normal-Tahmini $\widehat{y_0}$	**Hileli-Tahmini** $\widehat{y_1}$
Normal-Gerçek y_0	Doğru Negatif (TN)	Yanlış Pozitif (FP)
Hileli-Gerçek y_1	Yanlış Negatif (FN)	Doğru Pozitif (TP)

- FN: y_1 sınıfına ait olmayanlar olarak yanlış sınıflandırılmış, y_1sınıfının üyelerinin yüzdesini ifade eder. Yani gerçekte hileli kullanıcı olmasına rağmen normal, sahtekar olmayan sınıfında tahminlenmiş kullanıcıları gösterir.
- FP: y_1 sınıfına ait olanlar olarak yanlış sınıflandırılmış diğer sınıfların üyelerinin yüzdesini ifade eder. Yani gerçekte normal bir kullanıcı olmasına rağmen hileli olarak tahminlenmiş kullanıcıları gösterir.
- TP: y_1 sınıfına ait olanlar olarak doğru bir şekilde sınıflandırılmış y_1 sınıfı üyelerin yüzdesini ifade eder (%100 - FN). Yani gerçekte hileli bir kullanıcı iken sınıflandırma sonucu yine hileli olarak tahmin edilen kullanıcıları gösterir.
- TN: y_1 sınıfına ait olmayanlar olarak doğru sınıflandırılan diğer sınıfların üyelerinin yüzdesini ifade eder (%100 - FP). Yani gerçekte normal bir kullanıcı iken sınıflandırma sonucu yine normal olarak tahmin edilen kullanıcıları gösterir.

2.2.2. Performans Ölçüm Teknikleri

Sınıflandırma sonucunun kalitesini ölçmek amacıyla bir takım teknikler kullanılır. Bazı çalışmalar yalnızca doğruluk (ing. accuracy) ölçütünü bir değerlendirme yöntemi olarak kullanır. Makine öğrenmesi literatürü genellikle geri çağırma (ing. recall), kesinlik (ing. precision), F-skoru (ya da F1 skoru), ROC eğrisi

gibi ek metrikler kullanır. Bu metrikler sonraki alt bölümlerde ifade edilmiştir (Sokolova ve Lapalme, 2009: 427-437; Ballabio ve diğerleri, 2018: 33-44; Zhu ve diğerleri, 2017; Ikonomakis ve diğerleri, 2005: 966-974; Gu ve diğerleri,2009: 461-471; Nguyen ve Armitage, 2008: 56-76; Wang ve Yao, 2011: 206-219).

2.2.2.1. Doğruluk

Doğru olarak sınıflandırılan sahtekarlık sayısı ile toplam örnek sayısı arasındaki orandır. Bu metrik genel olarak sınıflandırıcının etkinliğini verir. Bu ölçüt genellikle, toplam örnek sayısı arasında doğru sınıflandırılmış örneklerin yüzdesi olarak tanımlanır. Özellikle veri kümesinin dengesiz olduğu yan etiket sınıflarının dengeli sayıda örneklere sahip olmadığı durumlarda doğruluk ölçütü tek başına yetersizdir. Bu durum bu ölçütün yanlış pozitif, yanlış negatif durumlarını ayırmaksızın genel doğruluk oranı vermesinden kaynaklanır. Bu sebeple bu tip örneklerde diğer ölçütlerden de yararlanmak mantıklıdır. Doğruluk skoru aşağıdaki gibi hesaplanır.

$$Doğruluk = A = \frac{TP + TN}{TP + TN + FP + FN}$$

2.2.2.2. Geri Çağırma

Doğru sınıflandırılmış pozitif örnek sayısının verilerdeki pozitif örnek sayısına bölünmesiyle elde edilen değerdir. Yani gerçekte pozitif olarak sınıflandırılan verilerin ne kadarının doğru sınıflandırıldığını gösteren oranı verir. Pozitif etiketleri tanımlamak için sınıflandırıcının etkinliğini gösterir. Bu ölçüt pozitif sınıf için hesaplanırsa ölçümün hassasiyetini (ing. sensitivity), negatif sınıf için hesaplanırsa ölçümün belirliliğini (ing. specificity) verir. Bu durumda, belirlilik de bir sınıflandırıcının negatif etiketleri ne kadar etkili bir şekilde tanımladığını gösterir. Her iki geri çağırma ölçütü için hesaplama formülü aşağıda verilmiştir.

$$Geri\ Çağırma\ (Hassasiyet) = \rho^{+} = \frac{TP}{TP + FN}$$

$$Geri\ Çağırma\ (Belirlilik) = \ \rho^{-} = \frac{TN}{TN + FP}$$

2.2.2.3. Kesinlik

Doğru sınıflandırılmış pozitif örnek sayısının verilerdeki pozitif örnek sayısına bölünmesiyle elde edilen değerdir. Yani pozitif olarak sınıflandırılan verilerin gerçekte ne kadarının pozitif olduğunu gösterir. Sınıflandırıcının yanlış sınıflandırmadan kaçınma yeteneği olarak da bilinir. Bu sebeple pozitif tahmin değeri olarak da adlandırılır. Kesinlik formülü aşağıda sunulmuştur.

$$Kesinlik = \pi = \frac{TP}{TP + FP}$$

2.2.2.4. F-skoru

F1 olarak da bilinen F skoru, geri çağırma ölçütü içindeki hassasiyet ve kesinlik ölçütlerinin bir birleşimidir. Söz konusu iki ölçüt birbirleri ile zıt ilişkililerdir. F skoru bu iki ölçüt arasında bir denge bulmayı sağlar. Dolayısıyla bu ölçüt hem yanlış pozitifleri hem de yanlış negatifleri hesaba katar. Hassasiyet ve kesinlik sınıflarının harmonik ortalaması alınarak aşağıdaki gibi hesaplanır.

$$F = (1 + p^2)\frac{\pi_i {\rho_i}^+}{(p^2\pi_i) + {\rho_i}^+}$$

Burada p hassasiyet ve kesinlik ölçütlerinin göreli önemini gösterir. Ancak çoğu zaman bu değer 1 'e eşittir. Bu sebeple yeni formül aşağıdaki gibi olur:

$$F = \frac{2\pi_i {\rho_i}^+}{\pi_i + {\rho_i}^+}$$

$$F = \frac{2TP}{2TP + FP + FN}$$

2.2.2.5. ROC Eğrisi

Gerçek pozitif oranının dikey eksen üzerine, yanlış pozitif oranınsa yatay eksen üzerine çizildiği iki boyutlu bir grafiktir. Grafik üzerine tüm sınıflandırma kararlarının eşik değerleri çizilir. ROC eğrisi, faydalar (TP oranı) ve maliyetler (FP oranı) arasındaki göreceli dengeyi gösterir. ROC eğrileri, TP ve FP 'nin göreceli maliyetleri için en iyi karar sınırları ailesini temsil eder. ROC eğrisindeki ideal nokta tüm pozitif örneklerin doğru olarak sınıflandırıldığı, hiçbir negatif örneğin pozitif olarak yanlış sınıflandırılmadığı (0,100) noktasıdır (Chawla ve diğerleri, 2002: 321-357).
X ekseni için FP oranı aşağıdaki gibidir.

$$\%FP = \frac{FP}{TN + FP}$$

Y ekseni için TP oranı aşağıdaki gibidir.

$$\%TP = \frac{TP}{TP + FN}$$

2.2.2.6. AUC Değeri

ROC eğrisinin altında kalan alandır. Yani, AUC tüm ROC eğrisinin altındaki iki boyutlu alanı ölçmek için kullanılır. Sınıflandırıcının yanlış sınıflandırmayı önleme yeteneğini gösterir. Tahminlerin ne kadar iyi sıralandığını ölçer. ROC eğrisinin şekli ve eğrinin altındaki alan (AUC), bir testin ayırt edici gücünün ne kadar yüksek olduğunu tahmin etmemize yardımcı olur. Eğri sol üst köşeye ne kadar yakın

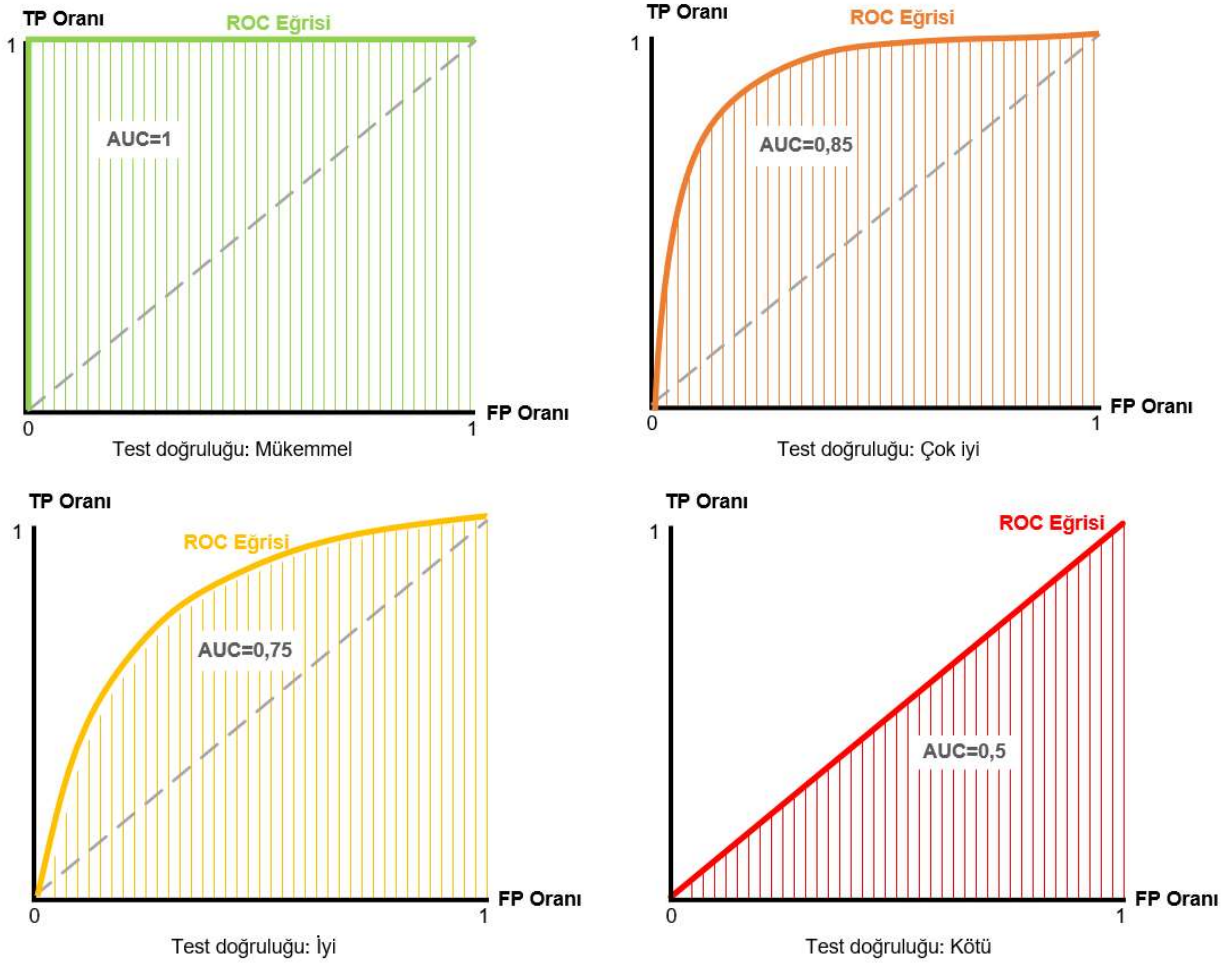
yerleştirilirse ve eğrinin altındaki alan ne kadar büyük olursa, testin etiketler arasında ayrım yapması o kadar iyidir. Yani AUC değeri büyüdükçe testin güvenilirliği artar.

AUC değeri ve testin doğruluğu aşağıdaki gibi sınıflandırılabilir (Šimundić, 2009: 203):

Tablo 4: AUC Ölçeği Sınıflandırılması

AUC Değer Sınıfları	Test Doğruluğu
0,9 – 1,0	Mükemmel
0,8 – 0,9	Çok iyi
0,7 – 0,8	İyi
0,6 – 0,7	Fena Değil
0,5 – 0,6	Kötü
< 0,5	Test işe yaramaz

Şekil 10: Test Doğruluklarına göre ROC Eğrisi



2.3. MAKİNE ÖĞRENMESİNDE SINIFLANDIRMA

Makine öğrenmesi, sahtekarlığın nasıl gerçekleştirilebileceğini geniş yelpazede hesaba katmayı mümkün kılar. Bir makine öğrenme bakış açısıyla, sahtekarlık tespit sorunu, yeni bir işlemin yasal veya hileli olup olmadığını belirlemeyi amaçlayan bir sınıflandırma görevine karşılık gelir. Hileli durumların tanımlanması, tipik bir sınıflandırma sorunu olarak görülmektedir (Melo-Acosta ve diğerleri, 2017: 1-6).

Veri madenciliğinin sınıflandırma yaklaşımları, veri kümelerinin eğitimi ve test edilmesi için sınıflandırma modellerinin geliştirilmesinde kullanılan sistematik bir yaklaşımdır. Veri kümelerinin tahmin modellemesi ve sınıflandırması için karar ağacı sınıflandırıcısı, kural tabanlı sınıflandırıcı, sinir ağı sınıflandırıcısı, naif bayes sınıflandırıcısı, nöro bulanık sınıflandırıcı, destek vektör makinesi, regresyon vb. çeşitli sınıflandırma teknikleri kullanılmaktadır (Monika ve Kaur, 2018: 19-23). Bu tür sınıflandırma görevleri kredi kartı, sağlık ve otomobil sigortalarının tespitinde, kurumsal dolandırıcılık ve diğer sahtekarlık türlerinde kullanılır. Ayrıca sınıflandırma, sahtekarlık tespitinde veri madenciliği uygulamasının en yaygın kullanıldığı alanlardan biridir (Ngai ve diğerleri, 2011: 559-569).

Sınıflandırma iki adımlı bir prosedürdür. İlk adımda, bir eğitim örneği kullanılarak bir model eğitilir. İkinci adımda ise model, eğitim örneğine ait olmayan nesneleri onaylamaya ve doğrulama örneğini oluşturmaya çalışır (Kirkos ve diğerleri, 2007: 995-1003).

Sınıflandırmada hedef değişken kategoriktir. Bu da yalnızca sınırlı sayıda önceden tanımlanmış değer alabileceği anlamına gelir. İkili sınıflandırmada, sadece iki sınıf dikkate alınır (örneğin, sahteciye karşı sahteci olmayan). Ancak çok sınıflı sınıflandırmada hedef ikiden fazla sınıfa ait olabilir (örneğin, şiddetli sahtekarlık, orta düzey sahtekarlık, sahtekarlık yok) (Baesens ve diğerleri, 2015). Takip eden alt bölümlerde sınıflandırma amacıyla kullanılan makine öğrenmesi algoritmaları ele alınmıştır.

2.3.1. Destek Vektör Makineleri

Destek vektör makineleri karmaşık veri setleri ile başa çıkabilen bir gözetimli öğrenme algoritmasıdır (Vapnik, 1998: 1-740). Bu yöntemde veriler daha büyük boyutlu bir girdi alanı üzerine konumlandırılır ve bu alan üzerinde optimal bir ayırıcı hiper düzlem oluşturur. Her bir hiper düzlem farklı bir sınıflandırma sistemi oluşturur. Oluşabilecek bütün sınıflandırma sistemleri, veri kümesini doğrusal olarak ayırabilir.

Ancak içlerinden yalnızca biri tüm sınıfların birbirine en yakın noktalarını maksimum ayırma gücüne sahiptir. Sınıfları birbirinden en uzak ve en eşit şekilde ayırması sebebiyle bu doğrusal sınıflandırıcı sisteme optimal ayırıcı hiper düzlem adı verilir. Böylece sınıflar arasındaki ayrım keskinleşir, sapmalar minimize edilir. Sezgisel olarak, bu sınırın diğer mümkün sınırlara göre daha iyi genelleme yeteneği olması beklenir (Gunn, 1998: 5-16). Maksimum marj hiper düzlemi, sınıflar arasında en büyük ayrımı veren düzlemdir. Maksimum marj hiper düzlemine en yakın örneklere destek vektörleri denir. Her sınıf için her zaman en az bir destek vektörü bulunur, çoğu zaman daha fazlası vardır (Zareapoor ve diğerleri, 2012: 52).

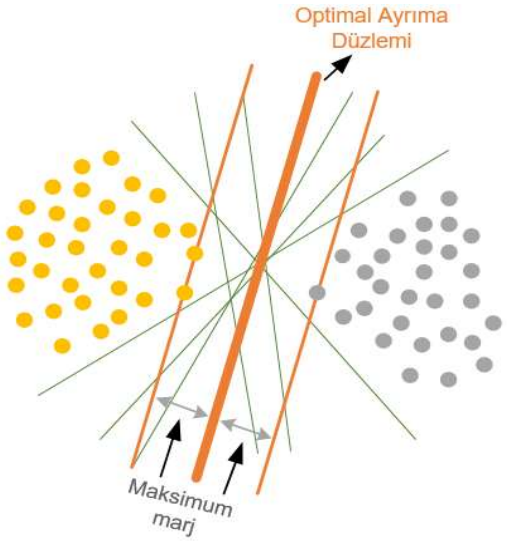
Düzlem ve hiper düzlem kavramları teorik olarak aşağıdaki gibi tanımlanabilir (Deisenroth ve diğerleri, 2020):

- Düzlem: R^n (n boyutlu uzay) 'in iki boyutlu afin alt uzaylarına düzlem denir. Parametrik düzlemler denklemi $y = x_0 + \lambda_1 x_1 + \lambda_2 x_2$ şeklinde ifade edilebilir. Burada $\lambda_1, \lambda_2 \in R$ ve $U = [x_1; x_2] \subseteq R^n$. Bu durumda, düzlem, bir destek noktası x_0 ve yön uzayını kapsayan doğrusal olarak bağımsız iki vektör x_1 ve x_2 ile tanımlanır.
- Hiper düzlem: R^n 'de, $(n-1)$ boyutlu afin alt uzaylara hiper düzlemler denir. Hiper düzlemi ifade eden parametrik denklem $y = x_0 + \sum_{i=1}^{n-1} \lambda_i x_i$ şeklinde ifade edilir. Burada $x_1, \dots, x_{n-1}$, R^n 'in $(n-1)$ boyutlu U alt uzayının bir temelini oluşturur. Bu durumda, hiper düzlem, bir destek noktası x_0 ve yön uzayını kapsayan $(n-1)$ adet doğrusal olarak bağımsız vektörler $x_1, \dots, x_{n-1}$ ile tanımlanır. R^2 'de bir çizgi aynı zamanda bir hiper düzlemdir. R^3 'de bir düzlem de aynı zamanda bir hiper düzlemdir.

Doğrusal olarak ayrılabilir $\{(x_1; y_1), \dots, (x_n; y_n)\}$ bir veri kümesi için, sonsuz sayıda aday hiper düzlemi bulunmaktadır. Bu nedenle sınıflandırıcılar sınıflandırma problemini herhangi bir eğitim hatası olmadan çözer. Benzersiz bir çözüm bulmak için, pozitif ve negatif örnekler arasındaki marjı maksimize eden ayırma hiper düzlemini seçilmelidir. Diğer bir deyişle, pozitif ve negatif örneklerin büyük bir farkla ayrılması istenmektedir.

Şekil 11 'de de görüldüğü gibi, sonsuz sayıda ayırma düzlemi arasında maksimum marj genişliğini veren düzlem optimal ayırma düzlemidir. Çünkü marj genişliğinin maksimize edilmesi sınıflar arasındaki ayrımı keskinleştirir, sapmaları minimize eder, sınıflandırmanın genelleme yeteneğini arttırır. Bu durumda, eğitim verilerinde herhangi bir aykırı değer bulunmadığı ve bilinmeyen test verilerinin eğitim verileriyle aynı dağılıma uyduğu varsayımı altında, optimal hiper düzlem ayırma hiper düzlemi olarak seçilirse genelleme yeteneği en üst düzeye çıkarılır (Abe, 2005: 44).

Şekil 11: İki Boyutlu Uzayda Optimal Ayırma Düzlemi



İki sınıflı durumda, eğitim prosedürünün arkasındaki temel fikir bir hiper düzlem bulmaktır. $\vec{w}$ ile temsil edilen hiper düzlemin görevi, sadece sınıfları birbirinden ayırmak değil ayrılan marjın mümkün olduğunca büyük olmasını sağlamaktır.

$$\vec{w} = \sum_j a_j c_j \vec{d}_j \ , \qquad a_j \geq 0$$

Burada cj ∈ {1, −1} olmak üzere, $\vec{d}_j$ referans vektörünün doğru sınıfını temsil eder. αj 'ler, çift optimizasyon problemi çözülerek elde edilir. $\vec{d}_j$ sıfırdan büyük olan a_j 'ler için destek vektörleridir. Çünkü bunlar $\vec{w}$ 'ye katkıda bulunan tek referans vektörleridir. Test örneklerinin sınıflandırılması, sadece hiper düzlemin hangi tarafına düştüğünü belirlemekten ibarettir (Pang ve diğerleri, 2002).

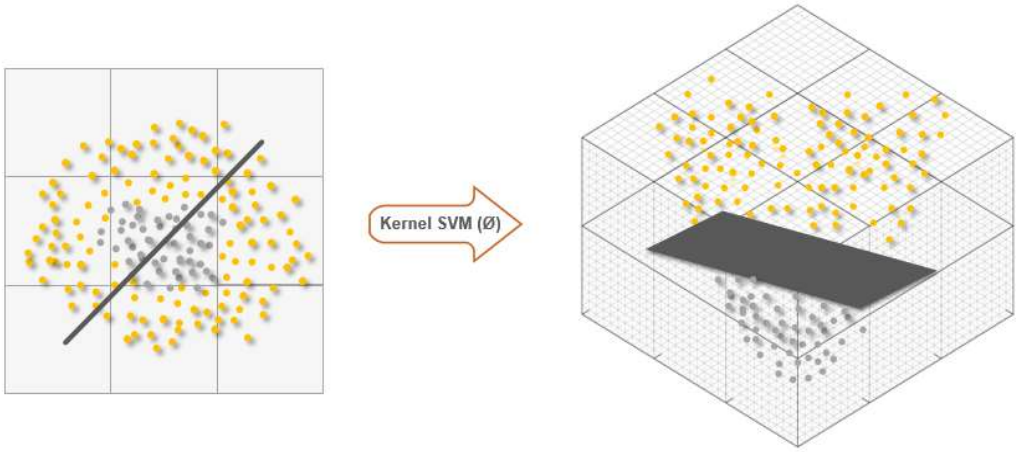
Sınıflandırma yeteneği göz önünde bulundurularak SVM 'in avantajları şu şekilde sıralanabilir(Abe, 2005: 44):

- Genelleştirme yeteneğinin maksimize edilebilmesi
- Yerel minimum noktası bulunmaması
- Geniş uygulama alanı bulunması
- Sapan gözlemlerin ortadan kaldırılabilmesi

SVM doğrusal problemlere başarı ile uygulanabilmesinin yanı sıra doğrusal olarak birbirinden ayrılamayan sınıflar için de uygulanabilmektedir. Doğrusal olmayan problemler için Kernel SVM teknikleri kullanılır. Doğrusal ayırmanın mümkün olmadığı durumlarda, kernel teknikleri özellik uzayına otomatik olarak doğrusal olmayan bir haritalandırma getirir. Ardından özellik uzayındaki SVM tarafından, girdi uzayında doğrusal olmayan karar sınırına karşılık gelen hiper düzlem bulunur (Furey ve diğerleri, 2000: 906-914). Kernel SVM tekniği ∅ ile ifade edilen formüldeki dönüşümü

kullanarak örnek olarak üç boyutlu düzlem üzerinde doğrusal olmayan problemlerde Şekil 12 ‘de gösterildiği gibi sınıflandırma yapar.

Şekil 12: Doğrusal Olmayan Örnek Uzayı için Optimal Ayırma Düzlemi



$$\emptyset(x_1, x_2) = (z_1, z_2, z_3) = (x_1, x_2, {x_1}^2 + {x_2}^2)$$

2.3.2. Rassal Orman

Rassal Orman (RF) birçok karar ağacından ve her bir karar ağacı çıktılarının birleştirilmesiyle elde edilen sınıfların çıktılarından oluşan topluluk öğrenmeye ait bir sınıflandırıcıdır (Rajora ve diğerleri, 2018: 1958-1963). Topluluk yöntemlerinin ardındaki temel ilke, bir grup “zayıf öğrenicinin” bir “güçlü öğrenici” oluşturmak için bir araya gelebilmesidir. Rassal ormanlar birçok karar ağacı yetiştirir. Burada her bireysel karar ağacı “zayıf öğrenicidir”, birlikte ele alınan tüm karar ağaçları ise “güçlü öğrenicidir” (Seeja ve Zareapoor, 2014).

Rassal bir orman çok yollu sınıflandırıcı, her bir ağaç bir çeşit randomizasyon kullanılarak yetiştirilen birkaç ağaçtan oluşur (Bosch ve diğerleri, 2007: 1-8). RF, Breiman tarafından önerilmiş olup sınıflanırma ve regresyon ağacı (CART) methodolojisini kullanan karar ağacı tabanlı bir öğrenme algoritmasıdır (Breiman ve diğerleri, 1984). Ancak RF ağaçları, iki aşamalı bir rassallaştırma prosedürü kullanılarak belirsiz olarak yetiştirildiklerinden CART ‘tan farklıdır. Orijinal verinin bir önyükleme (ing. bootstrap) örneği kullanılarak ağacın büyütülmesiyle ortaya çıkan rassallaştırmanın yanı sıra, ağacı büyütürken düğüm seviyesinde ikinci bir rassallaştırma katmanı eklenir. Bir ağaç düğümünü tüm değişkenleri kullanarak bölmek yerine, RF her ağacın her bir düğümünde, değişkenlerin rastgele bir alt kümesini seçer ve düğüm için en iyi ayrımı bulmak amacıyla aday olarak yalnızca bu değişkenleri kullanır. Bu iki aşamalı rassallaştırmanın amacı, ağaçların

korelasyonunu ortadan kaldırmak için orman topluluğunun düşük bir varyanslı torbalama (ing. bagging) olayına sahip olmasıdır (Chen ve Ishwaran, 2012: 323-329).

Rastgele ormanların prensibi, öğrenme örneğinden gelen ve her bir düğümde açıklayıcı değişkenlerinin bir alt kümesini rastgele seçen bir takım önyükleme örneği kullanarak oluşturulan birçok ikili karar ağacını birleştirmektir (Genuer ve diğerleri, 2010: 2225-2236). Standart ağaçlarda, her bir düğüm tüm değişkenler arasındaki en iyi ayrım kullanılarak bölünür. Rastgele bir ormanda, her düğüm bu düğümde rastgele seçilen tahmincilerin alt kümeleri arasında en iyisi kullanılarak bölünür (Liaw ve Wiener, 2002: 18-22). Bir sonraki ayırıcı arayışındaki tüm uygun öngörücüleri dikkate almak yerine RF, rastgele seçilen değişken alt kümeleri arasında bölünmek için en iyi değişkeni arar. Sonuç olarak, tek bir RF ağacında, herhangi iki farklı düğümün tamamen farklı değişken grupları olarak değerlendirilebilir. Yani RF ağaçlarının oldukça fazla sayıda, farklı ayırıcı içermesi muhtemeldir. Sonuç, rastgele ormandaki ağaçların her birinin, genellikle etkili performans gösteren dinamik olarak yapılandırılmış en yakın komşu sınıflandırıcı formu olmasıdır (Whiting ve diğerleri, 2012: 505-527).

RF algoritması süreci aşağıdaki gibidir (Liu ve diğerleri, 2015: 7):

- Adım 1: Örnek “m” (seçilen örnek sayısı) (m<M, M ana kütle sayısı) bir bootstrap yöntemi ile tüm örneklerden rastgele olarak seçilir.
- Adım 2: Çıkarılan örnekler ile budama olmadan bir karar ağacı oluşturulur.
- Adım 3: Önceki adımlar tekrarlanır, çok sayıda karar ağacı oluşturulur ve karar ağacı sınıflandırmaları $\{h_1(x), h_2(x), \dots, (h_{ntree}(x)\}$ art arda geliştirilir.
- Adım 4: Nihai sınıflandırma, karar ağacı sınıflandırmasının sonuçlarından elde edilen her bir kaydın oyu ile belirlenir.

Formülasyon aşağıdaki gibi ifade edilebilir. h_i tek bir karar ağacı modelini, Y çıktı değişkenini (veya hedef değişkeni) ve $I(\,)$ gösterge fonksiyonunu temsil eder.

$$H(x) = arg\max_{Y} \sum_{i=1}^{n} I(h_i(x) = Y)$$

RF algoritması aşağıda ifade edilen bir takım karakteristik avantajlara sahiptir (Breiman, 2001: 5-32), (Pang ve diğerleri, 2006: 2028-2036), (Gislason ve diğerleri, 2006: 294-300);

- Yüksek doğruluk oranına sahiptir.
- Aykırı değerlere ve gürültüye karşı nispeten dayanıklıdır.
- Torbalama veya önyükleme yöntemlerinden daha hızlıdır.
- Hata, kuvvet, korelasyon ve değişken önemi açısından yararlı iç tahminler verir.

- Basit ve kolayca paralelleştirilebilir.
- Orman yapıldıkça sınıflandırmanın sapmasız bir tahminini sağlar.
- Karşılaştırma gruplarında dengesiz örnek sayısı gözlenirse hata oranını ayarlayabilir.
- Sunduğu çıktı diyagramları değişkenler ile sınıflandırma veya regresyon arasında ilişki önerebilir.
- Yüksek boyutlu veriler ile başa çıkabilir.
- Topluluktaki yüksek sayıdaki ağacı kullanabilir.

2.3.3. Lojistik Regresyon

Lojistik regresyon (LR), iki koşul veya sınıf arasındaki girdi özelliklerinin doğrusal bir fonksiyonu olan ayırıcı bir hiper düzleme uyar. Verilen bir eğitim verisi setinde amaç (Ryali ve diğerleri, 2010: 752-764):

- Yeni bir örneğin sınıf etiketini doğru bir şekilde öngören hiper düzlemi tahmin etmek,
- Sınıf ayrımı hakkında en bilgilendirici özelliklerin alt kümesini belirlemektir.

Lojistik regresyon modelindeki çıkarımlar şu şekilde sıralanabilir (Zhu ve Hastie, 2004: 427-443);

- Sınıflar genellikle doğrusal olarak ayrılabilir.
- Logaritmik olabilirliği en fazla 0 olur.
- Katsayı parametreleri benzersiz bir şekilde tanımlanmamıştır ve aslında birçoğu sonsuz olacaktır.

İkili değişken birbirine zıt iki farklı değer ile sınırlandırılır. Evet/hayır, yaşam/ölüm, siyah/beyaz şeklinde belirlenebileceği gibi 1 ve 0 şeklinde de ikili değişken oluşturulabilir. Söz konusu değişkenler bir olayın meydana gelmesi ve gelmemesi, özelliğin mevcudiyeti ya da yokluğunu gösterir. Lojistik regresyon bu ikili değişkene sahip bir model üzerine kolaylıkla kurulur. Sahip olunan sınıflardan birinin başarıyı (p), diğerinin de başarısızlığı $(1-p)$ temsil ettiği düşünülecek olursa p, bağımlı değişkenin 1 değerini alma olasılığını, $1-p$ bağımlı değişkeninin 0 değerini alma olasılığını gösterir. p 'nin $1-p$ 'ye oranı da bahis oranı (ing. odds ratio) olarak adlandırılır. Bahis oranı aşağıdaki şekilde hesaplanır;

$$\left(\frac{p}{1-p}\right) = \frac{1+e^{z}}{1+e^{-z}}$$

Değişkenin uygulamaya anlamlı bir katkısı olup olmadığının anlaşılabilmesi için bahis oranından yararlanılır. Bahis oranı 1 'e yakın çıkan değişkenler bağımlı değişkenin açıklanmasında çok önem arz etmeyen değişkenlerdir. Bahis oranı değeri

1 ‘den büyük olan ve katsayısı anlamlı olan değişkenlerin bağımlı değişkeni açıklamada önemli bir etkisi olduğunu gösterir. Katsayının anlamlı olan ve bahis oranı değerleri 0 ‘a yakın olan değişkenlerin ise, bağımlı değişkeni açıklamada önemli bir etmen olduğunu fakat bağımlı değişkenin düşük değerler almasına neden olduğu için negatif bir etki sağladığı söylenebilir.

Sınıfların bahis oranlarının doğal logaritması lojistik dönüşümü verir. Başarıyı temsil eden p için logit olarak da temsil edilen lojistik dönüşüm aşağıdaki şekilde gösterilir. Burada p değeri 0 ‘a yaklaştığında $logit(p)$ -∞ ‘a, p değeri 1 ‘e yaklaştığında ise $logit(p)$ +∞ ‘a yakınsamaktadır.

$$logit(p) = log\ log\ \left(\frac{p}{1-p}\right)$$

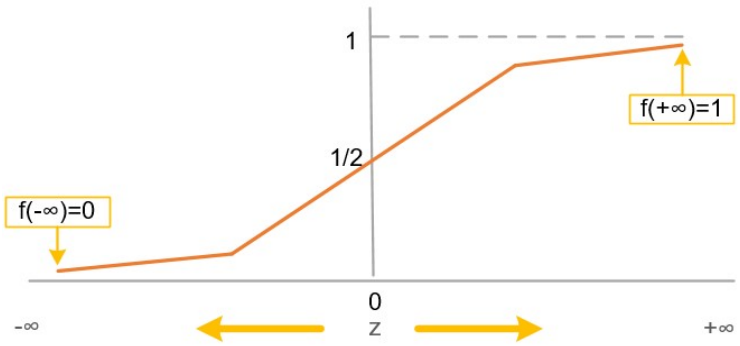
LR, bağımlı değişkenin kendisini hesaplamaktansa, bağımlı değişkenin olasılığının logaritmasındaki değişiklikleri hesaplar. Çünkü olasılığın logaritması açıklayıcı değişkenlerle doğrusal ilişkilidir, açıklanan ve açıklayıcı değişkenler arasındaki regresif ilişki doğrusal değildir (Subasi ve Ercelebi, 2005: 87-99).

LR, bazı bağımsız değişkenlerin ikili bağımlı değişken üzerindeki ilişkisini tanımlamak için kullanılabilen matematiksel bir modelleme yaklaşımıdır. Lojistik modele dayalı matematiksel yapıyı tanımlayan lojistik fonksiyon $f(z)$ olarak ifade edilir. $f(z)$ fonksiyonu her zaman 0 ile 1 arasındadır. Bu nedenle lojistik model için, asla 1 ‘in üzerinde veya 0 ‘ın altında bir risk tahmini alınamaz. (David ve Mitchel, 1994). $f(z)$ fonksiyonu aşağıdaki şekilde hesaplanmaktadır.

$$f(z) = \frac{e^{-z}}{(1+e^{-z})^2}$$

Şekilde de görüldüğü gibi z değeri -∞ iken $f(z)$ 0 ‘a eşittir. z değeri +∞ iken $f(z)$ 1 ‘e eşittir.

Şekil 13: Lojistik Fonksiyon Gösterimi



$$f(-\infty) = \frac{1}{1+e^{-(-\infty)}} = \frac{1}{1+e^{\infty}} = 0 \qquad f(+\infty) = \frac{1}{1+e^{-(+\infty)}} = \frac{1}{1+e^{-\infty}} = 1$$

Genel olarak, lojistik regresyon kategorik bir sonuç değişkeni ile bir veya daha fazla kategorik veya sürekli tahminci değişken arasındaki ilişkiler hakkındaki hipotezlerin tanımlanması ve test edilmesi için çok uygundur. Doğrusal regresyonun en basit senaryosunda, bir sürekli bağımsız değişken tahmincisi ve bir ikili bağımlı değişken için bu verilerin grafiği her biri ikili çıktının değerine karşılık gelen iki paralel çizgi ile sonuçlanır. İki paralel çizginin, sonuçların ikiliği nedeniyle bir en küçük kareler regresyon denklemi ile tanımlanması zor olduğu için, bunun yerine tahminci için kategoriler oluşturulabilir ve ilgili kategoriler için sonuç değişkeninin ortalaması hesaplanabilir. Yani LR, bağımlı değişkene logit dönüşümünü uygulayarak bu problemleri çözer (Peng ve diğerleri, 2002: 3-14).

Lojistik formülasyon sonuçları sadece açıklayıcı değişkenlerin eşit kovaryans matrisli normal dağılıma uyduğu varsayımından gelmez, aynı zamanda açıklayıcı değişkenlerin bağımsız ve ikili (0 ya da 1) değişken olduğu ya da bazılarının çok değişkenli normal ve bazılarının da ikili oldukları varsayılır. Dolayısıyla, lojistik modeli diskriminant analizi için (doğrusal bir diskriminant fonksiyonu yerine) kullanmanın bir avantajı, nispeten dayanıklı olmasıdır. Yani, altta yatan birçok varsayım aynı lojistik formülasyona yol açar (Press ve Wilson, 1978: 699-705). Bu sebeple LR analizinde En Küçük Kareler yönteminde olduğu gibi normallik, doğrusallık gibi şartlar yoktur. Temel bir varsayımın olmayışı esnekliği, yöntemin sıklıkla kullanılabilmesini sağlar.

2.3.4. Naif Bayes

Sınıflandırma için yaygın olarak kullanılan Naif Bayes (NB) sınıflandırıcısı, basit bir olasılık teoremi olan Bayes teoremi ile sağlanır. Diskriminant fonksiyonları seti kullanılarak elde edilen sınıflandırıcıya ve eğitim setinden ilgili olasılıkların hesaplanmasına Naif Bayes sınıflandırıcısı denir. Çünkü sınıf göz önüne alındığında niteliklerin bağımsız olduğu “saf” varsayımı yapılırsa, bu sınıflandırıcı yanlış sınıflandırma oranı veya sıfır-bir kaybı en aza indirgeme açısından kolaylıkla optimal olarak gösterilebilir (Domingos ve Pazzani, 1997: 103-130).

Bayesyen öğrenmenin temel varsayımları aşağıdaki gibidir (Duda ve diğerleri, 1973: 731-739).

- Sınıf sayısı bilinmektedir.
- Her sınıf için öncül olasılıklar bilinmektedir.
- Sınıf şartlı olasılık yoğunlukları için formlar bilinir, ancak tam parametre vektörü bilinmemektedir.

- Parametre vektörü hakkındaki bilgilerin bir kısmı bilinen bir öncül olasılık yoğunluk fonksiyonunda bulunmaktadır.
- Parametre vektörü hakkındaki bilgilerin geri kalanı, bilinen karışım yoğunluğundan bağımsız olarak çekilen bir dizi örneğin setinde bulunmaktadır.

Bayes sınıflandırıcı $h^*(x)$ (aynı zamanda Bayes-optimal sınıflandırıcı olarak da adlandırdığımız ve $BO(x)$ olarak da gösterebildiğimiz), $f_i{}^*(x)$ gibi bir özellik vektörü verildiğinde diskriminant fonksiyonu olarak sınıf sonlu olasılıklarını (class posterior probabilities) kullanır. Bu yapıya Bayes kuralı uygulandığında aşağıdaki teorem elde edilir (Rish, 2001: 41-46).

$$P(C = c_i \,|X = x) = \frac{P(C = c_i)P(X = x \mid C = c_i)}{P(X = x)}$$

Burada X, $x = (x_1,\ x_2, \dots, x_n)$ şeklinde x özellik değerlerinden oluşan rassal bir değişken vektörüdür. C ise, $c = (c_1,\ c_2, \dots, c_i)$ sınıflarından oluşan rassal bir değişkendir. $P(X = x \mid C = c_i)$, olabilirliği maksimize eden C sınıfını temsil eder. Bayes teoremini veren $P(C = c_i \,|X = x)$, burada x özellik vektörü seçildiğinde c_i sınıflandırmanın gerçekleşme olasılığını verir. Diğer bir deyişle bu teorem, x vektörü özelliğine sahip olduğunu bildiğimiz durumda, referans olarak verilen c_i sınıfına ait koşullu olasılık değeridir. Bayes kuralı, bu koşullu olasılığın, her sınıfın referansı için özellik değerlerinin belirli vektörlerini gören koşullu olasılıklarından ve her sınıfın bir referansını gören koşulsuz olasılığından nasıl hesaplanabileceğini belirler (Lewis, 1998: 4-15).

$P(X = x)$ tüm sınıflar için özdeştir ve bu yüzden göz ardı edilebilir. $P(X = x \mid C = i)$ sınıf şartlı olasılık dağılımı olmak üzere bayes sınıflandırıcı aşağıdaki şekilde elde edilmiş olur (Rish, 2001: 41-46).

$$h^*(x) = \underset{i}{\operatorname{argmax}}\, P(C = c_i)P(X = x \mid C = c_i)$$

Yukarıdaki Bayes sınıflandırıcıdan yola çıkarak NB sınıflandırıcı aşağıdaki gibi formüle edilebilir.

$$h^*{}_{NB}(x) = \underset{i}{\operatorname{argmax}}\, P(C = c_i) \prod_i P(X = x \mid C = c_i)$$

Bayes sınıflandırıcılarının temeline dayanarak Naif Bayes sınıflandırıcıları üretilir. Bu sınıflandırıcı tüm kombinasyonların olasılık değerini hesaplayarak en iyi (en yüksek) olasılık değerine sahip kombinasyona göre sınıflandırma yapar. Basitçe tüm şartlı olasılıklar çarpıldığında NB sınıflandırıcısı elde edilmiş olur.

2.3.5. K-En Yakın Komşu

Bir veri kaydının sınıflandırılması için en yakın komşuları yeniden düzenlenir. Komşuluktaki veri kayıtları arasında çoğunluk oylaması genellikle mesafeye dayalı ağırlıklandırmayı dikkate alarak veya bu ağırlıklandırmayı hesaba katmaksızın sınıflandırmaya karar vermek için kullanılır (Guo ve diğerleri, 2003: 986-996). K-en yakın komşu (KNN) kuralı, etiketlenmemiş her örneği, eğitim setindeki en yakın k komşuları arasındaki çoğunluk etiketi ile sınıflandırır. Bu nedenle performansı büyük ölçüde en yakın komşuları tanımlamak için kullanılan uzaklık metriğine bağlıdır. Ön bilginin yokluğunda, çoğu KNN sınıflandırıcısı, vektör girdileri olarak temsil edilen örnekler arasındaki farklılıkları ölçmek için basit öklid mesafelerini kullanır. Bununla birlikte, öklid uzaklık ölçütleri, verilerdeki büyük bir etiketli örnek eğitim setinden tahmin edilebilecek herhangi bir istatistiki doğruluktan faydalanmamaktadır (Weinberger ve Saul, 2009). p adet özelliğe sahip bağımsız değişken vektörü $x_i = (x_{i1}, x_{i2}, \dots, x_{ip})$ ve $x_j\ j = (1,2, \dots, n)$ arasındaki öklid uzaklığı aşağıdaki şekilde formüle edilebilir.

$$d(x_i, x_j) = \sqrt{(x_{i1}, x_{j1})^2 + (x_{i2}, x_{j2})^2 + \cdots + (x_{ip}, x_{jp})^2}$$

Mutlak uzaklık ise aşağıdaki şekilde ifade edilir.

$$d|x_i, x_j| = \sum_{n=1}^{p} |x_{is}, x_{js}|$$

KNN 'yi uygulamak amacıyla k için uygun bir değer seçilmesi gerekir. Sınıflandırmanın başarısı seçilen bu k değerine bağlıdır. Bir anlamda, KNN yöntemi k açısından yanlıdır. k değerini seçmenin birçok yolu vardır, ancak algoritmayı farklı k değerleri ile birçok kez çalıştırmak ve en iyi performansa sahip olanı seçmek basit bir seçim yöntemidir (Guo ve diğerleri, 2003: 986-996). Özetle k parametresi kullanıcı tarafından seçilmelidir. Daha sonra karar, komşuların çoğunun ait olduğu sınıf etiketi lehine olur. k_r, r sınıfına ait en yakın komşuların grubundan gözlem sayısını göstersin;

$$\sum_{r=1}^{c} k_r = k$$

Daha sonra k_l ile l sınıfına yeni bir gözlem tahmin edilir;

$$k_l = \max_r(k_r)$$

Bu, öngörülen sınıf hakkında karar veren öğrenme setinden tekil bir gözlem yapılmasını önler. Tekniğin yerellik derecesi k parametresi ile belirlenir. k=1 için, en

yakın yerel teknik olarak basitçe en yakın komşu yöntemi, k→n_L k → nL için ise tüm öğrenme seti sonuçlarının global çoğunluk oyu alınır. Bu, sınıflandırılması gereken tüm yeni gözlemler için sürekli bir öngörü anlamına gelir: Her zaman öğrenme setindeki frekansı en sık sınıf öngörülür (Hechenbichler ve Schliep, 2004). Uygulama adımları temel olarak aşağıdaki gibi sıralanabilir;

- k parametresi belirlenir.
- Eğitim setindeki tüm noktalar ile test noktasının öklid uzaklığı hesaplanır.
- En yakın komşuların sınıf üyeliği baz alınarak veri setindeki tüm diğer üyelikler öklid uzaklığına göre sınıflandırılır.
- Uzaklıklar artan sırada sıralanır.
- Her üyelik sınıfı için olasılıklar hesaplanır.
- En yüksek olasılığa sahip sınıf tahmin edilir.

Yeni bir işlemi normal veya sahteci sınıfa göre sınıflandırmak için KNN sınıflandırıcısı, yeni işlem ile her bir eğitim süreci örneği arasındaki benzerliği hesaplar ve yeni işlemin sınıfını tahmin etmek için en çok benzer komşuların sınıf etiketlerini kullanır. Buradaki temel varsayım, aynı sınıfa ait işlemlerin vektör uzayında bir araya geleceğidir. Yani KNN sınıflandırıcısı, bir örneğin sınıflandırmasının, vektör uzayında bulunan diğer örneklerin sınıflandırmasına en çok benzediği varsayımına dayanır. (Liao ve Vemuri, 2002: 439-448).

2.4. TOPLULUK ÖĞRENME

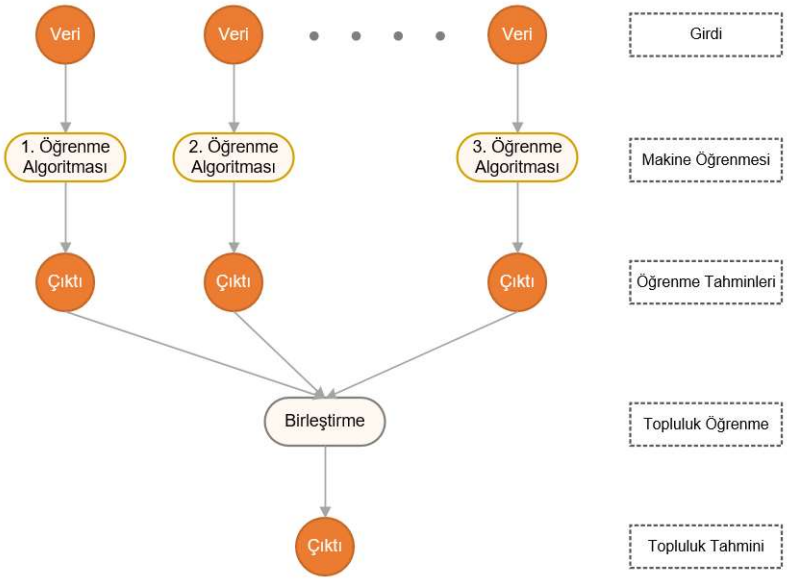
Her yöntemin güçlü yönlerini ve sınırlarını daha iyi anladıktan sonra, bir sorunu çözmek için iki veya daha fazla algoritmanın birleştirilmesi olasılığının araştırılması gerekir. Amaç, bir yöntemin güçlü yönlerini, bir diğerinin zayıf yönlerini tamamlamak için kullanmaktır. Yalnızca mümkün olan en iyi sınıflandırma doğruluğu ile ilgileniyorsak, iyi bir sınıflandırıcı topluluğu kadar iyi performans gösteren tek bir sınıflandırıcı bulmak zor veya imkansız olabilir (Kotsiantis ve diğerleri, 2006: 159-190).

Sınıflandırıcılar topluluğu (ing. ensemble), yeni örnekleri sınıflandırmak için bireysel kararları bir şekilde (tipik olarak ağırlıklı veya ağırlıksız oylama ile) birleştirilen bir grup sınıflandırıcıdır (Thomas, 1997: 97-136). Topluluk yöntemleri sınıflandırmanın bir kümesini oluşturan ve yeni veri noktalarını sınıflandıran öğrenme algoritmalarıdır. Bu model, tahminleme performansını arttırmak için kullanılır.

Topluluk öğrenmesi öğrenme algoritmasından aldığı sonuçları birleştiren bir yöntemdir. Dolayısı ile veriler her bir makine öğrenmesi algoritmasında ayrı ayrı

çalışır, makine eğitim veri seti ile eğitilir ve test veri setinde doğrulama yapar. Ardından makine öğrenmesinden elde edilen tüm çıktılar belirlenen topluluk öğrenme yönteminde birleştirilir. Birleştirilmenin ardından nihai çıktı elde edilir. Topluluk öğrenmenin genel akışı Şekil 14 'de verilmiştir.

Şekil 14: Topluluk Öğrenme Algoritması Akışı



Literatürde birleştirme olarak geçen yöntem, mümkün olan her bir veri seti için sınıflandırıcılar topluluğunu eğitmek amacıyla kullanılabilir ve uygun bir birleştirme kuralı kullanarak eğitim çıktılarını birleştirebilir. Topluluk öğrenme sistemlerinde amaç, sınıflayıcıların farklı bir kümesini oluşturmak ve böylece nispeten farklı bir karar sınırı yaratmaktır. Bu şekilde ilgili sınıflayıcıların stratejik olarak birleştirilmesi ile toplam hatanın azaltılması sağlanır (Muhlbaier ve diğerleri, 2008: 152-168).

Genel olarak, topluluk öğrenme yöntemleri kara kutu tipi sınıflandırıcılar olarak değerlendirilirler. Topluluk sınıflandırıcıların doğası, yan ürün verilerinin akıllıca kullanılmasıyla olabildiğince çok bilgi toplanabilmesidir (Gislason ve diğerleri, 2006: 294-300). Topluluk yöntemleri yalnızca bir model kullanmak yerine birden çok analitik modeli tahmin etmeyi amaçlamaktadır. Buradaki fikir, birden fazla modelin veri girdi alanının farklı bölümlerini kapsayabilmesi ve böylece birbirlerinin eksikliklerini tamamlayabilmesidir. Bunu başarılı bir şekilde gerçekleştirmek için, analitik tekniğin, temel verilerdeki değişikliklere duyarlı olması gerekir (Baesens ve diğerleri, 2015).

Bir öğrenme algoritmasında istatistiksel problemlerde yüksek varyans sorunu büyük bir problem teşkil eder. Yine öğrenme algoritmasına temsili problemlerde yaşanan ciddi sorunlardan biri de yüksek sapma durumudur. Topluluk öğrenme

yöntemleri öğrenme algoritmalarında hem varyans hem de sapmayı azaltır. Bu yüzden sıklıkla tercih edilir (Dietterich, 2002: 110-125).

Sınıflandırıcılar topluluğu oluşturmak için kullanılan mekanizmalar şunları içerir (Ikonomakis ve diğerleri, 2005: 966-974);

- Tek bir öğrenme yöntemiyle farklı eğitim verileri alt kümesini kullanma.
- Tek bir eğitim yöntemiyle farklı eğitim parametrelerini kullanma (örneğin, bir toplulukta her sinir ağı için farklı başlangıç ağırlıkları kullanma).
- Farklı öğrenme yöntemlerini kullanma.

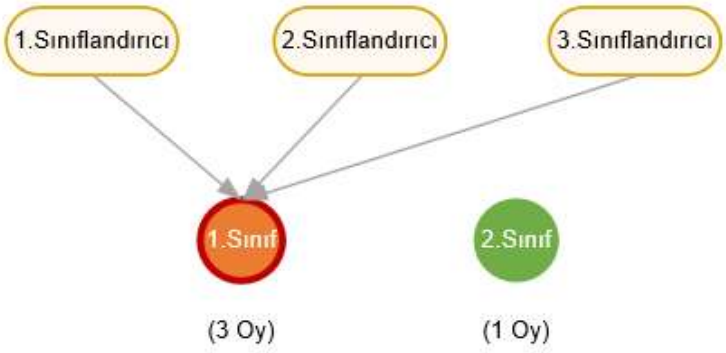
Topluluk öğrenme için oylama, istifleme ve ön yükleme gibi farklı yöntemler mevcuttur. Oylama, topluluk öğrenme teknikleri içinden sıklıkla kullanılan bir yöntemdir. Stratejik oylamaya dayalı topluluk birleşimi kuralı sadece performansta önemli bir gelişme sağlamakla kalmaz, aynı zamanda önemli ölçüde daha az sınıflandırıcı kullanılmasını sağlar. Oylama prosedürü, bireysel sınıflandırıcıları akıllı uzmanlar olarak ele alır. Bu prosedür, belirli bir örneği tanımlama konusunda daha fazla güvenilir olan sınıflandırıcıları kollektif olarak belirlemek için sınıflandırıcıların birbirleriyle iletişim halinde olmasını sağlar ve oylama ağırlıklarını bu örneğe göre dinamik olarak ayarlar (Muhlbaier ve diğerleri, 2008: 152-168).

Çoğunluk oylamasının, topluluk kararına bağlı olarak oy birliği ile oylama, basit çoğunluk, çoğul oylama ve yumuşak oylama gibi çeşitleri bulunur (Zhang ve Ma, 2012), (Kima ve diğerleri, 2011: 437-449).

2.4.1. Oy Birliğiyle Karar

Tüm sınıflandırıcıların üzerinde anlaştığı sınıf oylamasıdır. Şekil 15 'de verilen örnekte olduğu gibi tüm sınıflandırıcıların oyladığı sınıf kazanan sınıftır, sapma beklenmez. Bir sınıfın seçilebilmesi için tüm sınıflandırıcıların aynı sınıfı oylaması gerekir.

Şekil 15: Oy Birliğiyle Oylama



2.4.2. Basit Çoğunluk

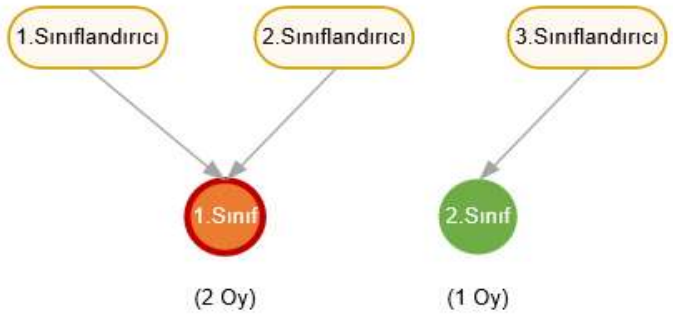
Öngörülen sınıflandırıcı sayısının yarıdan en az bir fazlası kadar oy toplamaya dayalı oylamadır. Örnek olarak üç sınıflandırıcı ve iki sınıf varsa, sınıflandırıcılardan ikisi 1.sınıfı seçiyorsa oylama basitçe çoğunluk olan sınıftan yana olur ve 1.sınıf seçilir. Bu oylama, ağırlıklı bir çoğunluk oylaması örneğidir ve her bir sınıflandırıcıya eşit ağırlık 1/j atar; burada j, bir topluluktaki sınıflandırıcı sayısıdır.

Bu çalışmada topluluk öğrenme yöntemi olarak basit çoğunluk ele alınmıştır. J 'inci sınıfa ait x_n girdisi için t 'inci sınıflandırıcının çıktısı ${f^{(j)}}_t(x_n)$ olarak temsil edilmek üzere, çalışmada kullanılan basit oylamaya dayalı topluluk öğrenme formülasyonu aşağıda sunulmuştur (Cao ve diğerleri, 2015: 275-284).

$$ {f^{(j)}}_{com}(x) = \sum_{t=1}^{T} {f^{(j)}}_t(x_n), \qquad {f^{(j)}}_t(x_n) \in \{0,1\} $$

Burada sınıflandırıcı j 'yi sınıf etiketi olarak tahminlerse ${f^{(j)}}_t(x_n) = 1$ olur. Aksi takdirde bu değer 0 'dır. Basit çoğunluk oylamasının çalışma şekli Şekil 16 'da verilen örnekte özetlenmiştir. 1.ve 2.sınıflandırıcılar 1.sınıfı seçtikleri için ilk sınıf basit çoğunluğa göre oylamada galip gelir.

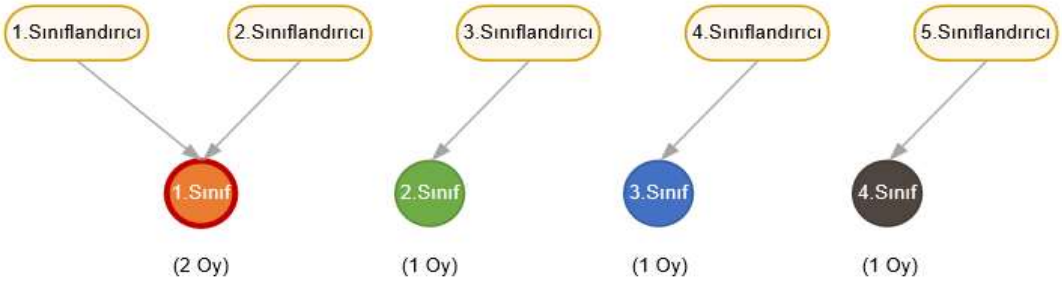
Şekil 16: Basit Çoğunluk Oylaması



2.4.3. Çoğul Oylama

Oyların toplamı % 50 'yi aşsın ya da aşmasın en fazla oy alan sınıfın kazandığı oylamadır. Şekil 17 'de bulunan örnekte olduğu gibi çoğunluk oylanan sınıfın sınıflandırıcı sayısı, toplam sınıflandırıcı sayısının yarısından az olmasına rağmen yine de bu sınıf kazanan sınıftır.

Şekil 17: Çoğul Oylama



Aksi belirtilmedikçe oy çokluğu genellikle matematiksel olarak aşağıdaki gibi tanımlanabilen çoğul oylamayı önerir;

$$\sum_{t=1}^{T} d_{t,c^*} = max_c \sum_{t=1}^{T} d_{t,c}$$

Çoğul oylamada T sınıflandırıcı ve C sınıf sayısı olmak üzere, t 'inci sınıflandırıcının kararı $d_{t,c^*} \in \{0,1\}$, t = 1, … , T ve c = 1, … , C şeklinde ifade edilir.

2.4.4. Yumuşak Oylama

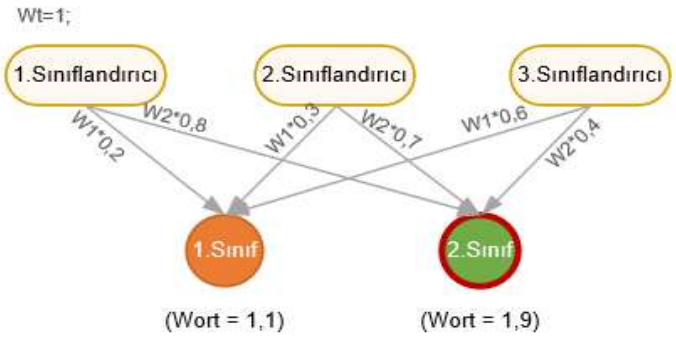
Sınıf olasılıklarının ortalaması %50 'den fazla olan alan sınıfın kazandığı oylamadır. Yaklaşım, her sınıflandırıcının her bir sınıfının ağırlığını dikkate alır. Bu sebeple daha hassas sonuçların elde edilmesi muhtemeldir.

Bu çalışmada topluluk öğrenme yöntemi olarak ele alınan bir diğer yöntem yumuşak oylama yöntemidir. J 'inci sınıfa ait x_n girdisi için t 'inci sınıflandırıcının çıktısı ${f^{(j)}}_t(x_n)$ olarak temsil edilmek üzere, çalışmada kullanılan yumuşak oylama topluluk öğrenme formülasyonu aşağıda sunulmuştur (Cao ve diğerleri, 2015: 275-284).

$${f^{(j)}}_{com}(x) = \sum_{t=1}^{T} w_t {f^{(j)}}_t(x_n), \qquad {f^{(j)}}_t(x_n) \in [0,1]$$

w_t, sınıflandırıcılar için her bir sınıfın ağırlığını ifade eder. Sınıflandırıcıların ağırlıkları sabit ve eşit olduğu varsayımında, yumuşak oylama yöntemi Şekil 18 'de verilen bir örnekle açıklanmıştır. Şekil 18 'de olduğu gibi her bir sınıflandırıcının tüm sınıflar için toplam olasılık fonksiyonu 1 'e eşit olacak şekilde dağıtılır. Ardından her bir sınıfa gelen olasılık değerlerinin ağırlıklı ortalaması alındığında en yüksek ortalamaya sahip olan 2. sınıf yumuşak oylamada galip gelir.

Şekil 18: Yumuşak Oylama


Wt=1;
1.Sınıflandırıcı
2.Sınıflandırıcı
3.Sınıflandırıcı
W1*0,2
W2*0,8
W1*0,3
W2*0,7
W1*0,6
W2*0,4
1.Sınıf
2.Sınıf
(Wort = 1,1)
(Wort = 1,9)

ÜÇÜNCÜ BÖLÜM

UYGULAMA

3.1. UYGULAMA KAPSAMI

Uygulamada, çalışmanın temel konusunu oluşturan mobil uygulamada sahtekarlık tespiti üzerine yapılandırılmış ve bu amaçla bir gerçek hayat problemi ele alınmıştır. Hem yurtiçi hem yurtdışı müşteri ağına sahip, akıllı ulaşım sistemlerinde öncü bir firma olan Kentkart firmasının mobil biletleme kullanan ABD'deki Kansas müşterisine ait veriler kullanılarak uygulama gerçekleştirilmiştir. Yapılan çalışma aynı zamanda TUBITAK tarafından desteklenen bir TEYDEB-1511 kodlu ve 1180046 numaralı projede yürütülmüştür.

Uygulamada Kentkart-Kansas mobil kullanıcılarının mobil uygulamayı kullanım, bilet satın alma ve bilet kullanım gibi temel verileri baz alınarak bu kullanıcıların sahtekar olup olmama durumları araştırılmıştır. Toplu taşımada Kentkart-Kansas iş birliği 2017 yılında başlamıştır. Mobil uygulamanın sağladığı yeteneklerden yararlanabilmek için uygulamaya kayıt olmak gerekmektedir. Mobil uygulamanın toplamda 108652 kişi kayıtlı kullanıcısı bulunmaktadır. İlk kayıt işleminde kullanıcılar beyaz listede olarak kaydedilmektedir. Geliştirilen akıllı sistem öncesinde kara liste yönetimi gözle kontrollerle sağlanmakta ve sahteci hareketler gözle kontrolle yakalanabilirse, kullanıcılar kara listeye eklenmekteydi. Gözle kontrolün zorluğu ve gözden kaçırılan sahteci kullanıcıların yarattığı maliyet, mobil uygulamada akıllı bir tespit sistemi geliştirilmesi gerekliliğini ortaya çıkarmıştır.

Akıllı tespit sistemi geliştirilmesi için, ana kütleden rassal örnek olarak alınan kullanıcılar yapılan kontrollerle kara listede/ kara listede değil şeklinde etiketlenmiş, sistem bu kullanıcılar ile eğitilerek sistemin diğer kullanıcılarının sınıflandırma yöntemleri ile otomatik olarak etiketlenmesi yapılmıştır. Faydalanılan sınıflandırma yöntemlerinin ardından topluluk öğrenme yöntemleri çalıştırılarak, sınıflandırma yöntemleri sonuçları birleştirilmiş ve bu sayede sınıflandırma performans ölçüm değerlerinde daha da iyileşme sağlanmıştır.

Çalışmanın ilk aşamasında risk kategorileri ve sistemde mevcut olan, gelecekte mevcut olabilecek piyasa riskleri belirlenmiştir. Ardından uygulamanın risk haritasını gösteren bir risk planı oluşturulmuş ve bu doğrultuda sahtekarlık analizinde kullanılmak üzere bağlamlar belirlenerek riskler ile eşleştirilmiştir. Çalışmanın ikinci

aşaması ise makine öğrenmesi modelleri, değişken seçimi, boyut indirgeme teknikleri ve topluluk öğrenme yöntemlerinden oluşmaktadır.

3.2. SİSTEM RİSKLERİNİ ANLAMA VE ÖNLEME

Sistemde mevcut sahtekarlıkların tespit edilebilmesi için ilk adım olarak sistemdeki riskler kategorilere ayrılmıştır. Bu kapsamda aşağıdaki risk kategorileri belirlenmiştir.

- Ücretlendirme-biletlendirme riskleri
- Kullanıcı riskleri
- Uygulama riskleri
- İşletim sistemi riskleri

Riskler kategorilere ayrıldıktan sonra mobil uygulamaların elektronik ödeme sistemlerinde genel olarak rastlanan piyasa riskleri, risk kategorileri ile eşleştirilmiştir. Şirketin gizlilik politikası gereği hangi risklerin sistemde mevcut olduğu, hangi riskler ile karşılaşılmasının muhtemel olduğu, hangilerinin piyasada olup sistemde gözlenmediği gibi bilgiler gizli tutulmuştur. Belirlenen genel piyasa riskleri Tablo 5 'deki gibidir.

Tablo 5: Ödeme Sistemlerinde Karşılaşılabilen Genel Piyasa Riskleri

Risk Sınıfları	Riskler
Ücretlendirme - Biletlendirme Riskleri	Bilet kopyalama
	Bilet üretme
	Gerekenden fazla bilet kullanma
	Hileli satın alım
	Gerekenden uzun süre kullanma
	Aynı bileti tekrar kullanma
	Sahte sunucudan bilet satma
	Devretme/devralma
Kullanıcı Riskleri	Mahremiyet İhlali
	Farklı kullanıcı imtiyazlarından yararlanma
	Kullanıcı Yaratma / Yok Etme / Kopyalama
Uygulama Riskleri	Verilere / uygulama koduna erişim / denetim / kopyalama / değişiklik vb. güvenlik zafiyeti
	İşlev görememe
	Sahte yazılım kullanımı
	Kullanıcı erişim/denetim hakları zafiyeti
İşletim Sistemi Riskleri	İşletim sistemine müdahale

Riskler ve kategoriler eşleştirildikten sonra uygulamada kullanılmak üzere risklere kodlar atanmıştır. Ancak yine şirketin gizlilik politikası gereği kodlar ile risklerin sıraları karıştırılarak verilmiş, hangi risk kodunun hangi riske karşılık geldiği karartılmıştır. Tablo 5 'deki riskler ile Tablo 6 'daki risk kodları sıralı olarak verilmemiştir, aralarında birebir eşleşme olmayabilir.

Tablo 6: Ödeme Sistemlerinde Karşılaşılabilen Genel Piyasa Risklerinin Kodlandırılması

Risk Sınıfları	Risk Kodları
Ücretlendirme - Biletlendirme Riskleri (RUB)	RÜB1
	RÜB2
	RÜB3
	RÜB4
	RÜB5
	RÜB6
	RÜB7
	RÜB8
	RÜB9
	RÜB10
	RÜB12
Kullanıcı Riskleri (RK)	RK1
	RK2
	RK3
	RK4
	RK5
Uygulama Riskleri (RU)	RU1
	RU2
	RU3
	RU4
İşletim Sistemi Riskleri (RIS)	RİS1

3.2.1. Risk Planı

Sistemdeki saldırıları önleyebilmek ve önlenemeyen saldırılara yanıt verebilmek için saldırı anında yaşanacak durumu gösteren bir risk planı hazırlanmıştır. Risk planı her bir riskin işletmede yaşanma olasılığını, yaşandığında karşılaşılacak etkileri ve buna takiben alınması gereken aksiyon planını gösterir. Aksiyonlar daha önce de belirtildiği gibi A 'dan D 'ye azalan şiddettedir. İşletmede belirlenen risk planı Tablo 7 'de sunulmuştur.

Tablo 7: Uygulama Sisteminin Risk Planı

Risk Kodu	Olasılık	Etki	Aksiyon
RÜB1	Orta	Yüksek	B
RÜB2	Orta	Yüksek	B
RÜB3	Orta	Yüksek	B
RÜB4	Düşük	Yüksek	C
RÜB5	Düşük	Yüksek	C
RÜB6	Düşük	Yüksek	C
RÜB7	Düşük	Orta	D
RÜB8	Yüksek	Orta	B
RÜB9	Düşük	Yüksek	C
RÜB10	Düşük	Aşırı	A
RÜB12	Düşük	Yüksek	C
RK1	Orta	Orta	C
RK2	Düşük	Düşük	D
RK3	Yüksek	Düşük	C
RK4	Yüksek	Orta	B
RK5	Yüksek	Düşük	C
RU1	Orta	Yüksek	B
RU2	Düşük	Yüksek	C
RU3	Düşük	Yüksek	C
RU4	Orta	Yüksek	B
RİS1	Yüksek	Düşük	C

3.2.2. Bağlam Duyarlı Yaklaşımın Uygulamaya Dahil Edilmesi

Risk planının nihai sonucu olan aksiyon planından yola çıkılarak belirlenen şiddetleri karşılayacak şekilde, işletme uzmanları tarafından bağlamlar belirlenmiştir. Bağlamlar kullanıcıların riskli durumları sergileyip sergilemediğini gösteren sorgulara karşılık gelen verilerdir. Yani her bir bağlam bir riski işaret eder. Her bir risk için aksiyon şiddetine göre en az bir bağlam belirlenmiştir. Bağlamlar kullanıcılardan toplanmaya başladığında hangi kullanıcının belirlenen riskleri taşıdığı anlaşılmaktadır. Bağlam duyarlı yaklaşım ile her bir risk sistem içinde karakterize edilerek kullanıcılarla eşleştirilmiştir. Bu sebeple, bağlamlar kullanıcının sistemdeki profillerini oluşturmak için kullanılmıştır.

Şirketin gizlilik politikasına doğrultusunda sistemde belirlenen bağlamlar ve uygulamada kullanılmak üzere belirlenen kodlar karışık sıra ile verilmiştir. Sistemin bağlam listesi Tablo 8 ‘de verilmiştir.

Tablo 8: Bağlam (Değişken) Listesi

Bağlamlar
Farklı ürünlerin aynı anda satın alınması
Çok sayıda ürünün aynı anda alınması (Farklı ürün ya da aynı ürün)
Sık aralıklarla satın alma işlemi yapılması
Kullanılmamış ürün adedi
Kullanıcıların biletleri uzun süre kullanmaması
Başarılı bir satın alma işlemi öncesinde çok sayıda başarısız işlem olması
Başarısız işlemlerin farklı kredi kartlarıyla yapılması
Başarılı işlemin hemen ardından farklı bir alım daha denenmesi
Kullanıcıların yeni olmaları (eski kullanıcılar sistemde daha güvenilir oluyor)
Kullanıcıların sistemde hiç hareketi olmaması
Kullanıcıların sistemde hiç satın alımı olmaması
Müşteri adı soyadının tek harf olması
Müşteride çok fazla sayıda farklı kredi kartı tanımlı olması (aynı anda veya faklı zamanlarda)
Müşterinin çok fazla sayıda silinmiş kredi kartı olması
Müşterinin tanımlı ödeme yönteminin olmaması
Aynı şifrenin farklı müşteriler tarafından kullanılması
Geçersiz e-posta adresi verilmesi
Kullanıcıdan banka aracılığı ile iade talebi gelmesi (Kullanıcı satın alımın kendisine ait olmadığını ifade eder)
Aynı kredi kartını kullanan farklı kullanıcıların olması
Hesabın başka bir telefona transfer edilme sıklığı
Yanlış şifre denemesi
Yanlış e-mail denemesi
Hesap transferde IMEI zincirinin kırılmış olması
Servise çıkma sayısının maksimum sayıyı aşılması
Adı ve soyadı aynı olan kullanıcılar olması
E-mail adresi belirli anahtar kelimeler içeren kullanıcılar
Kara listeye alınmış kullanıcının sisteme tekrar giriş yapması
IMEI numarası sıfır olan, boş olan, tek haneli olan kayıtların olması
Telefon numarası boş olan, geçersiz karakterli olan kayıtların olması
Kullanıcı bilet alma alışkanlığının değişmesi
Sistemde aktiviteye ara veren kullanıcılar olması
Aynı IMEI numarasını kullanan kullanıcılar olması
Uygulamanın mümkün olmayan bir konumda çalıştırılması
Bileti satın alma ve aktifleştirme arasında geçen konum zaman ilişkisinin uyumsuz olması
Yalnızca ilk hesap açılışında verilen ücretsiz biletin kullanılması, başka bilet alınmaması
E-mail doğrulamasından geçemeyen kullanıcılar ile aynı ad-soyada sahip kişilerin olması
E-mail doğrulamasından geçemeyen kullanıcılar ile aynı IMEI numarasına sahip kişiler olması
E-mail doğrulamasından geçemeyen kullanıcılar ile aynı telefon numarasına sahip kişiler olması
Yanlış kredi kartı bilgisi girişi yapılması

Maksimum numara transfer sayısının aşılması
Müşteri ad ve soyadının anlamsız harf dizilimlerinden oluşması
Müşterinin daha önce tanımlanmış ödeme yöntemi olmaması
Belirli miktarlarda işlemler yapılması
Sunucuda aynı numaralı ve farklı aşamalarda bir bilet olması
Cihaz veritabanında aynı numaralı birden fazla bilet olması
Veritabanı kaydında silme işlemi olması
Cihazdaki bilet sağlamasının olması gerekenden farklı olması
Daha önce aktivasyonu yapılmış bilet numaraları bulunması
Cihazda yapılan tarih, saat değişiklik sıklığı
Uygulama yükleme sayısı
Uygulamaların yüklenmesi aralarında geçen zaman
Uygulamaya giriş çıkış sayısı
Çıkış yapmadan giriş yapma sayısı
Kullanıcı hesabı oluşturulan konum bilgisi
Kullanıcının uygulama kullanımı esnasında telefon kullanım hareketleri
Kullanıcının uygulama kullanımı esnasında uygulama içindeki hareketleri
Yanlış e-mail denemesi
Emülatör modunda çalışması
Kara listenin yetkisiz değiştirilmesi, silinmesi
Fiyat kombinasyonlarına uymayan ödenen bilet ücretleri
Sadece güncelleme olduktan sonraki 2 gün içinde uygulamayı kullanan/yükleyen kullanıcı bilgisi

Bağlamlara rassal olarak kodlar atanmış ve işaret ettiği riskler eşleştirilmiştir. Bağlam ve risk eşleştirmeleri Tablo 9 ‘daki gibidir.

Tablo 9: Risk-Bağlam Eşleştirilmesi

Risk Sınıfları	**Karşılık Gelen Bağlamlar**
RÜB1	C22
RÜB2	C22
RÜB3	C24
RÜB4	C23
RÜB5	C23
RÜB6	C25, C27, C28, C29, C30
RÜB7	C57
RÜB8	C1, C2, C3, C4, C5, C6, C7, C9, C10, C11, C12, C13, C14, C15, C16, C17, C19, C26, C49, C58, C61
RÜB9	C8
RÜB10	C23
RÜB12	C20, C21
RK1	C20, C21, C22, C31, C63

RK2	C31, C33, C34, C35, C36
RK3	C27, C28, C29, C30, C31, C40, C41
RK4	C41, C,42, C43
RK5	C32, C40, C41, C50, C51, C59, C18, C60
RU1	C39
RU2	C22, C37, C44, C45, C48
RU3	C18, C35, C36, C38, C62
RU4	C37, C55, C56
RİS1	C25, C55, C56

Belirlenen risklere karşılık gelen bağlamlar kullanıcı ve ödeme bilgilerinin bulunduğu veri tabanlarından çeşitli SQL sorguları yazılarak toplanmıştır. Sistemde olmadığı bilinen riskler, bağlamlar sayesinde kesinleştirilmiştir. Bunun yanı sıra uygulamada, gerekli olup da içinde bilgi tutulmamış bağlamlar elimine edilmiştir. Böylece hiçbir değer almayan bağlamlar sistemden çıkarılmıştır. Son durumda uygulamada kullanılan bağlam kodları şu şekilde sıralandırılabilir: C1, C2, C3, C4, C5, C6, C7, C8,C9, C10, C11, C12, C13, C14, C16, C18, C19, C26, C31, C35, C38, C39, C40, C41, C42, C44, C45, C49, C50, C51, C55, C57, C58, C59, C60, C61, C62, C63.

3.3. UYGULAMANIN YAPISI

Çalışma kapsamında, Kentkart şirketinin ABD Kansas müşterisinden alınan veriler kullanılmıştır. ABD Kentkart mobil uygulamasının ulaşımda kullanılmak üzere çeşitli biletlerin satışı, biletlerin aktifleştirilerek toplu taşıma içerisinde kullanılması, yolcu bilgilendirme ve seyahat planlayıcısı gibi yetenekleri mevcuttur. Bu özelliklerden yararlanabilmek için mobil uygulamada hesap oluşturarak giriş yapmış olmak gerekmektedir. Hesap oluşturmak için de ad, soyad, geçerli bir e-posta adresi, 6 haneli şifre verilmelidir. Her IMEI ‘de tek bir hesap çalışır. Telefon değiştirmek için hesap transferi yapılabilir. Her hesap açıp yeni üye olan kullanıcıya ise tek binişlik ücretsiz bir bilet promosyon olarak sunulmaktadır. Zamanla hediye bilet avantajı ile kullanıcıların sahte hesap açma eğilimleri artış göstermiştir. Ayrıca sistemde çalıntı kredi kartı, izinsiz kredi kartı ödemesi dolayısı ile bilet iadeleri de gözlenmiştir. Bu sebeplerle, uygulamada kullanıcının hareketlerini kısıtlayan kara liste yapısı kullanılmıştır. Akıllı algoritma tasarlanması öncesinde, sahtekarlık yapan kullanıcılar için bilinen kurallar tanımlanmış ve kural kontrolünün gözle yapılmasıyla kara listeye ekleme işlemleri gerçekleştirilmiştir. Kara liste kuralı olarak, listeye alınan kullanıcılar

hesaplarına erişemez ve uygulamanın ayrıcalıklarından yararlanamazlar. Ancak kullanıcı operatöre erişerek sahteci olmadığını ispat ederse, kara listeden çıkarılabilir.

Zamanla, kullanıcı ve kullanım sayısı arttıkça kara listenin gözle kontrolünün uygulamada yeterli olmadığı, sahtecilerin sahtekarlık yöntemlerini sürekli değiştirdiği ve geliştirdiği anlaşılmış olup, tespit ve engelleme amacı ile akıllı ve otomatik bir ile kara liste yönetimi kullanılması gerekliliği ortaya çıkmıştır. Bu sebeple, makine öğrenmesi içerisindeki sınıflandırma algoritmalarından faydalanılarak sistemin mevcut verilerden öğrenmesi gerçekleştirilmiş ve kullanıcıların kara listede olup olmadığının otomatik olarak belirlenmesi amaçlanmıştır. Çalışmada verilere ait sınıf etiketleri halihazırda mevcut olduğundan gözetimli (denetimli) öğrenme gerçekleştiren sınıflandırma algoritmaları tercih edilmiştir.

Uygulama kapsamında 5 senaryo hazırlanarak test edilmiştir. Senaryoların her birinde sınıflandırma ve topluluk öğrenme yöntemleri kullanılmıştır. Bazı senaryolar içinse özellik seçimi ve özellik çıkarımı da uygulamaya dahil edilmiştir. Sınıflandırma yapmak amacıyla; rassal orman, destek vektör makinesi, naif bayes, k-en yakın komşu ve lojistik regresyon algoritmaları kullanılmıştır. Bu sınıflandırma teknikleri çalıştırılmış ve performans ölçümleri doğrultusunda, tamamının nihai sınıflandırma kararı için kullanılmasına karar verilmiştir. Nihai karar içinse topluluk öğrenme yöntemlerinden oylama teknikleri tercih edilerek, basit çoğunluk ve yumuşak oylama yöntemlerinden faydalanılmıştır. En iyi sonuç veren 5 senaryodan biri gerçek hayatta kullanılmak üzere uygulamada kullanılmıştır. Kurgulanan senaryolar Tablo 10 ‘daki gibidir.

Tablo 10: Uygulama Senaryoları

Senaryolar	**Açıklama**
Senaryo 1	Tüm Değişkenler ile Sınıflandırma & Topluluk Öğrenme
Senaryo 2	Değişken Çıkarımı & Sınıflandırma & Topluluk Öğrenme
Senaryo 3	Değişken Seçimi & Sınıflandırma & Topluluk Öğrenme
Senaryo 4	Değişken Seçimi & Değişken Çıkarımı & Sınıflandırma & Topluluk Öğrenme
Senaryo 5	Değişken İndirgeme & Değişken seçimi & Sınıflandırma & Topluluk Öğrenme

Uygulamada makine öğrenmesi teknikleri Şekil 19 ‘da verilen akışa göre çalışmaktadır. Sırası ile tüm veriler veri ön işleme aşamasına alınır. Burası değişkenlerin değerlendirildiği adımdır. Burada senaryo tercihlerine göre özellik çıkarımı ve özellik seçimi yapılır. Ardından sınıflandırma algoritmaları çalıştırılır.

Algoritmadan alınan sonuçlar üzerinde birleştirme yapılabilmesi için topluluk öğrenme algoritmaları çalıştırılır. Çıktı olarak her bir kullanıcıya kara liste/beyaz liste etiketleri verilir. Böylece makine öğrenmenin ilk döngüsü tamamlanır. Bu akış belli periyodlarla tekrar tekrar çalışmak üzere hazırlanmıştır. Makine öğrenmesindeki temel akış Şekil 19 ile sunulmuştur.

Şekil 19: Makine Öğrenmesi İş Akışı

Otomatik karar veren algoritmanın çalışma süresi de göz önünde bulundurularak her günün sonunda çalışması planlanmış ve böylece sistemin kullanıcının yeni hareketlerine adapte olması sağlanmıştır. Makine öğrenmesinden elde edilen son kararın ardından yanlış pozitif (FP) oranını minimize etmek için yani masum olarak işaretlenen ancak algoritma sonucu sahteci olarak belirlenen müşterileri kaybetmemek amacıyla son bir kontrol katmanı oluşturulmuştur. Bu kontrol için her bir kullanıcıdan algoritma sonucu tehdit skoru alınmıştır. Tehdit skoru bir kullanıcının "sahteci" olmasına ilişkin bir olasılık değeridir. Bu katman sınıflandırma yöntemlerinin çıktılarını etkilemez, sistem kullanıcısının kara liste otomasyonunu yönetme imkanı sağlar. Başka bir deyişle, geliştirilen sistem kara listede ve kara listede değil şeklinde kendi sınıf etiketlerini tahminler, sistem yöneticisi isterse tehdit skorları doğrultusunda yöntemin çıktı kararlarından tatmin etmeyen kısmını diğer

sınıfa öteleyebilir. Sistem yöneticisi kararı ile hazırlanan tehdit skoru karar mekanizması Şekil 20 ’de verilmiştir.

Şekil 20: Tehdit Skorları Sınıflandırması

- Güvenilir bölge: [0-50) puan arasında skor alan kullanıcılar bu bölgededir. Bu kullanıcılar kara listede değilse herhangi bir işlem yapılmaz, kara listedeyse kara listeden çıkartılır.
- Riskli bölge: [50-85) puan arasında skor alan kullanıcılar bu bölgededir. Bu kullanıcılar için otomatik işlem yapılmaz. Sistem yetkilisi tarafından kontrol sağlanır. Gerekli olanlar kara listeye alınır.
- Tehdit Bölgesi: [85-100] puan arasında skor alan kullanıcılar bu bölgededir. Bu kullanıcılar için otomatik kara listeye alma mekanızması çalışır.

Tehdit bölgeleri işletmede algoritmanın güven aralıklarını belirlemek için kullanılmıştır. Bu sayede, güvenilir, riskli ve tehdit bölgelerinin tolerans limitleri güncellenebilen, güven aralıkları yeniden yapılandırılabilen bir kontrol mekanizması oluşturulmuştur. Bu tehdit skorlarına göre faydalanılan yöntemin kararları ile işletmenin kararları birleştirilerek ikinci bir karar katmanı oluşturulması, gözle kontrolü minimize ederek otomasyon sağlanması amaçlanmıştır.

3.3.1. Veri Seti Yapısı

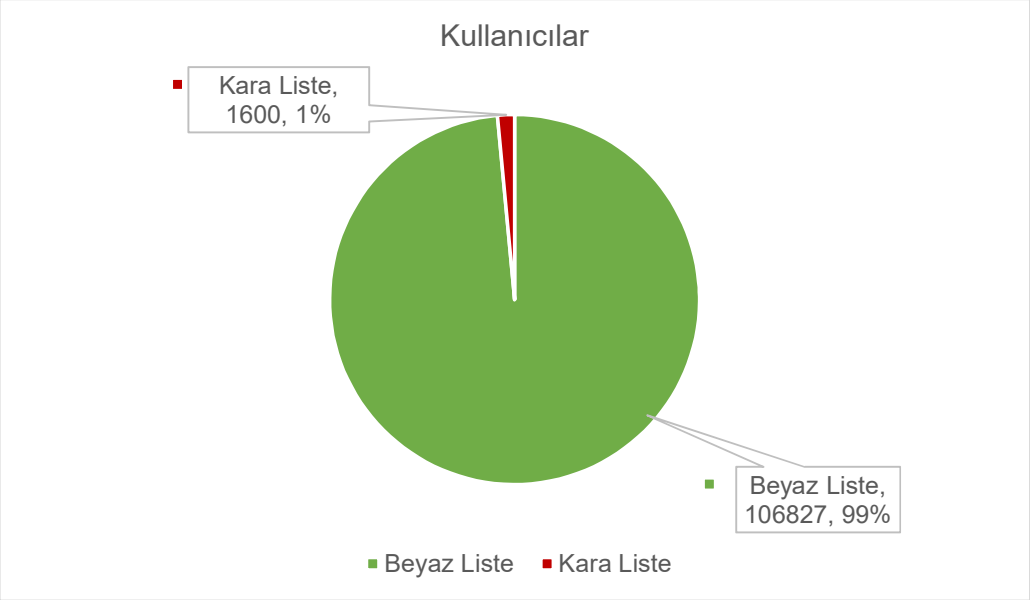
Otomatik kara liste yönetiminde kullanılmak üzere mobil uygulamada hesap oluşturan tüm kullanıcıların profilleri ele alınmıştır. Önceden belirlenmiş olan bağlamlar uygulamanın bağımsız değişkenlerini oluşturmaktadır. Toplam 38 bağlam bağımsız değişken olarak nitelendirilmiştir. Bu değişkenler için kategorik, ikili ve nicel değerler mevcuttur. Bağımlı değişken ise 0-1 ‘lerden oluşan kullanıcının kara listede olup olmadığını gösteren ikili değişkendir. Tanımlamak gerekirse:

- 0: kullanıcının kara listede olmama (masum olma) durumunu

- 1: kullanıcının kara listede olma (sahteci olma) durumunu ifade eder.

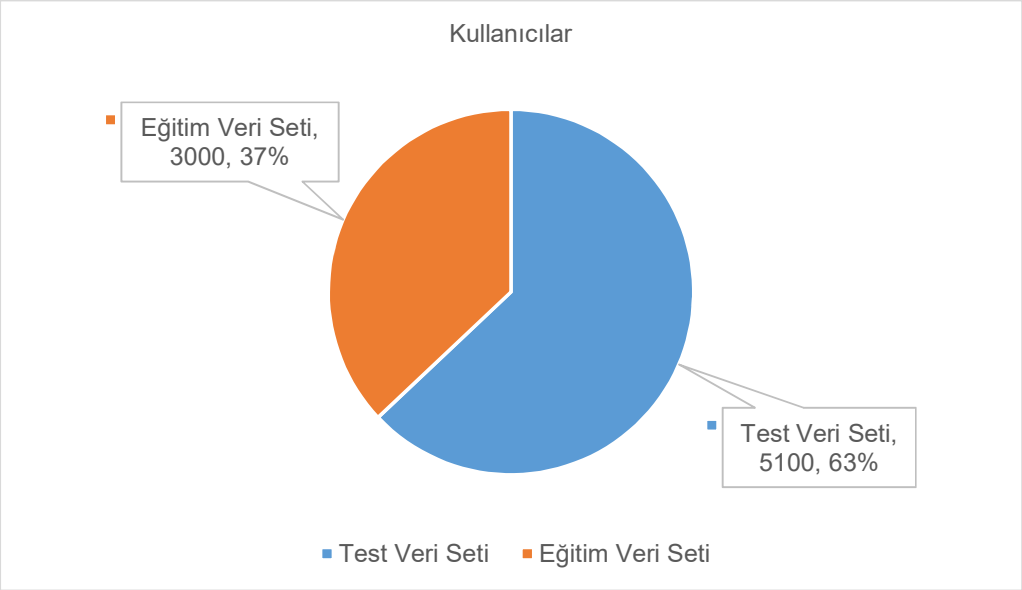
Sistemde kayıtlı toplam 108427 kullanıcı mevcuttur. Bu kullanıcıların 1600 tanesi kara listeyi oluşturmaktadır. Kalan 106827 kullanıcı masum kullanıcılar kategorisindedir. Şekil 21, bu durumu görselleştirmiştir.

Şekil 21: Ana kütle Kullanıcı Dağılımı



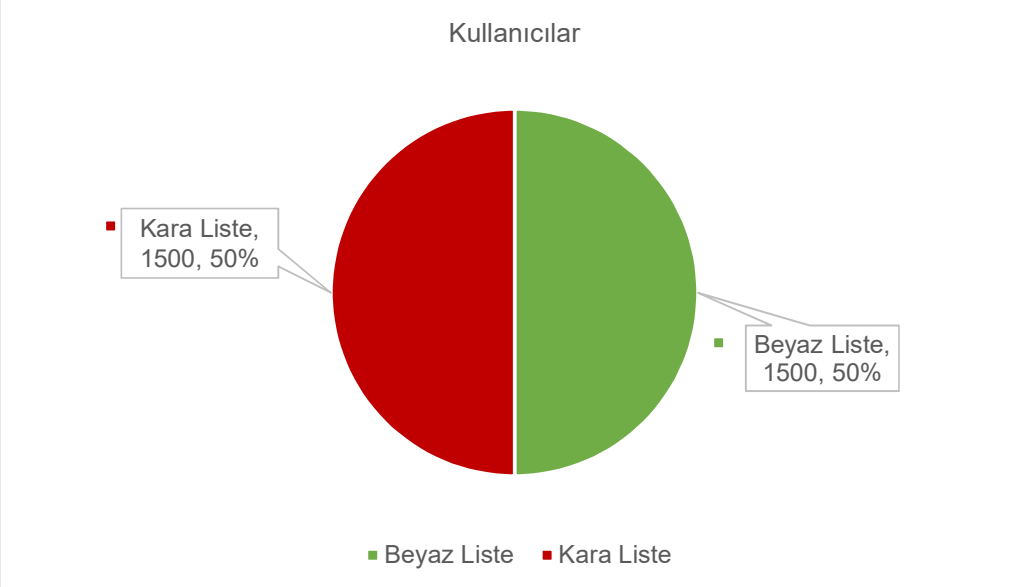
Kullanıcılar mobil uygulama üzerinde hesap oluşturduklarında uygulamada hayatlarına beyaz liste kullanıcıları olarak başlamaktadırlar. Ardından sergiledikleri riskli hareketler kural tabanlı kontrollerle yakalanabilirlerse kara listeye alınırlar. Kara liste kontrolü başlangıçta otomatik olmayan kurallara göre kontrol edildiği için masum kullanıcılar kategorisi aslında güvenilir olmayan, gözden kaçırılan sahteci kullanıcıları da içerebilmektedir. Aynı zamanda veri setinde kara liste/beyaz liste denge uyumsuzluğu da mevcuttur. Kara listedeki kullanıcılar tüm kullanıcıların %1,475 'ini oluşturmaktadır. Hem veri setinin tamamına güvenilemediğinden hem de veri seti içerisindeki sınıf dengesizliği problemi sebebiyle ana kütle üzerinden örneklem kümesi alınması yoluna başvurulmuştur. Bu amaçla mevcut veri setinden veriler ayıklanarak küçük bir örnek uzayı oluşturulmuştur. Yeni durumda, kara listede olduğu kesin olarak bilinen 1600 kişi ve kara listede olmadığı bilinen 6500 kişi oluşturulmuştur. Hazırlanan örnek uzayı makine öğrenmesinde kullanılmak üzere, eğitim ve test veri seti olarak ikiye ayrılmıştır. Tüm veri seti içinden eğitim ve test veri seti ayrımı Şekil 22 'de gösterilmiştir.

Şekil 22: Örnek Uzayı Test-Eğitim Dağılımı



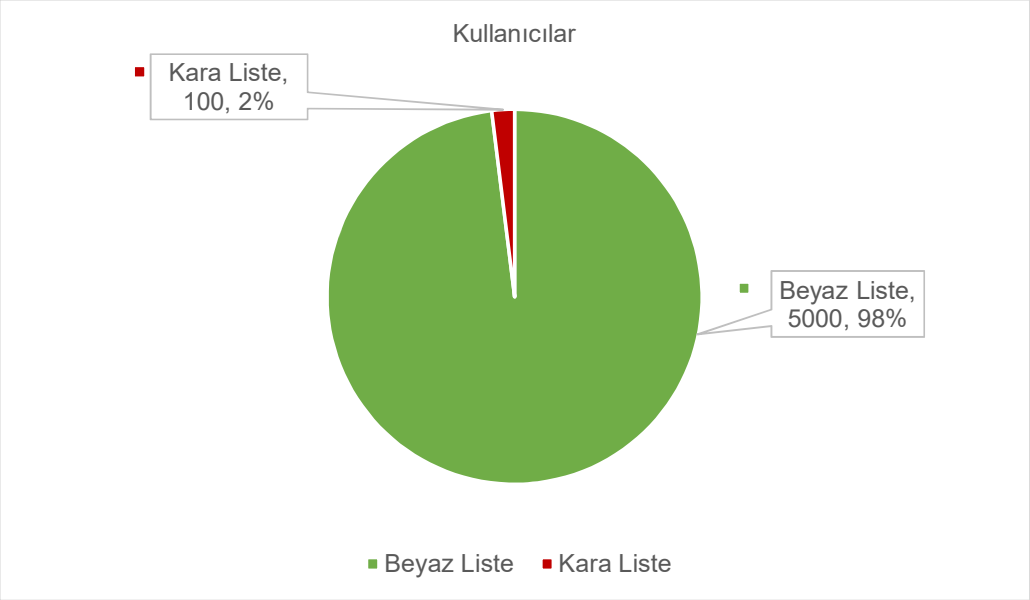
Eğitim ve test veri seti ayrımı yaparken dengeli bir eğitim seti oluşturmak için, kara liste kullanıcılarının büyük bir bölümü eğitim veri setinde kullanılacaktır. Örneklem içindeki test veri setinin ana kütle içerisindeki kara liste/beyaz liste dağılımına uymasına özen gösterilecek şekilde ayrılmasına dikkat edilmiştir. Eğitim veri setinin dengeli olması ve çalışmanın kara liste kullanıcılarından maksimum faydayı sağlaması amacıyla, eğitim setinde 1500 kullanıcı kara listeden 1500 kullanıcı da masum kullanıcı olarak beyaz listeden alınarak oluşturulmuştur. Bu durum Şekil 22'de verilmiştir. Böylece 5100 hacimli kalan kullanıcılar ise test veri seti içerisine alınmıştır. Örneklem üzerindeki veri seti dağılımı Şekil 23 'deki gibi özetlenebilir.

Şekil 23: Örnek Uzayı Eğitim Veri Setinde Kullanıcı Dağılımları



Ana kütlede, kara listedeki kullanıcılar tüm kullanıcıların %1,475 'ini oluşturmaktaydı. Örneklem üzerinde test veri setinin ana kütleyi en iyi şekilde yansıtabilmesi için, test veri seti içindeki kullanıcıların benzer şekilde %1,475 'inin kara listede olması gerekmektedir. Bu bilgi göz önünde bulundurularak örneklem için, test veri setinin yaklaşık %2 'si kara listede olacak şekilde belirlenmiştir. Bu durum Şekil 24 ile temsil edilmiştir.

Şekil 24: Örnek Uzayı Test Veri Setinde Kullanıcı Dağılımı

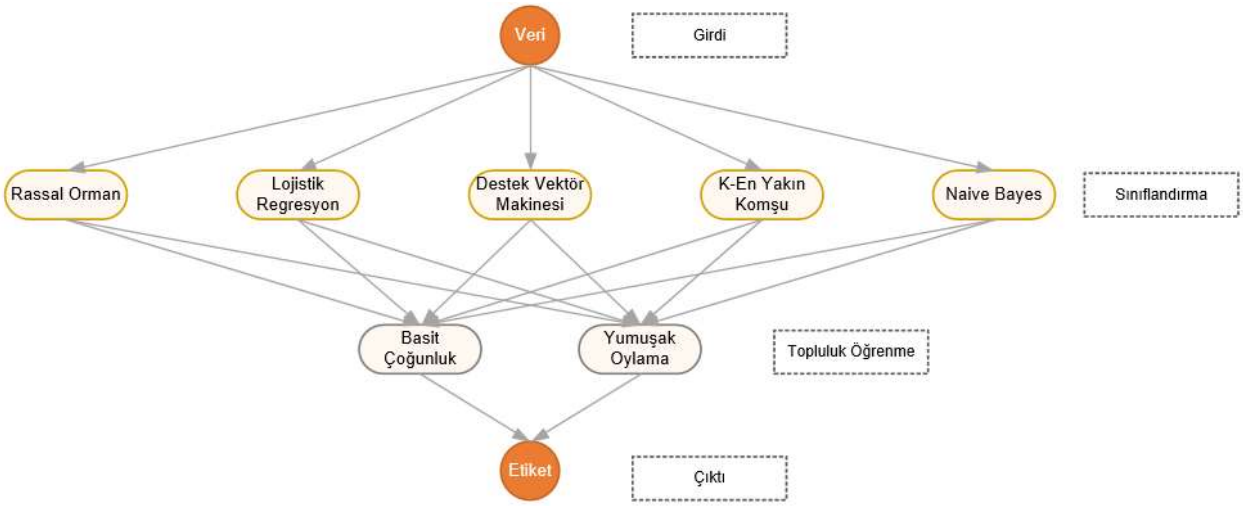


3.3.2. Senaryolar

3.3.2.1. Senaryo 1: Tüm Değişkenler İle Öğrenme

Çalışma kapsamındaki ilk senaryoda, herhangi bir değişken seçimi ya da değişken çıkarımı yöntemi kullanılmadan tüm değişkenlerle sınıflandırma yapılmış ve topluluk öğrenme yöntemi gerçekleştirilmiştir. Bu senaryonun amacı, boyut indirgeme yöntemlerinin elde edilen sonuçlar üzerindeki muhtemel olumlu ve olumsuz etkilerini tespit etmektir. Şekil 25 'de izlenen yöntem özetlenmiştir.

Şekil 25: Senaryo 1: Yöntem Akışı



İlk senaryo gereği eğitim veri seti üzerinde tüm değişkenlerle çalışılmıştır. Ardından test veri seti üzerinden sonuçlar alınmıştır. Deneyin karmaşıklık matrisi sonuçları Tablo 11-17 ile sunulmuştur.

Tablo 11: Senaryo 1: Rassal Orman Karmaşıklık Matrisi

Rassal Orman		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4953	42
	Sahteci (1)	7	98

Tablo 12: Senaryo 1: Lojistik Regresyon Karmaşıklık Matrisi

Lojistik Regresyon		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4894	101
	Sahteci (1)	11	94

Tablo 13: Senaryo 1: Destek Vektör Makinesi Karmaşıklık Matrisi

Destek Vektör Makinesi		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4892	103
	Sahteci (1)	8	97

Tablo 14: Senaryo 1: K-En Yakın Komşu Karmaşıklık Matrisi

K-En Yakın Komşu	Tahmin	
	Masum (0)	Sahteci (1)

Gerçek	Masum (0)	4922	73
	Sahteci (1)	11	94

Tablo 15: Senaryo 1: Naif Bayes Karmaşıklık Matrisi

Naif Bayes		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4947	48
	Sahteci (1)	16	89

Tablo 16: Senaryo 1: Basit Çoğunluk Topluluk Öğrenme Karmaşıklık Matrisi

Basit Çoğunluk		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4974	21
	Sahteci (1)	7	98

Tablo 17: Senaryo 1: Yumuşak Oylama Topluluk Öğrenme Karmaşıklık Matrisi

Yumuşak Oylama		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4973	22
	Sahteci (1)	5	100

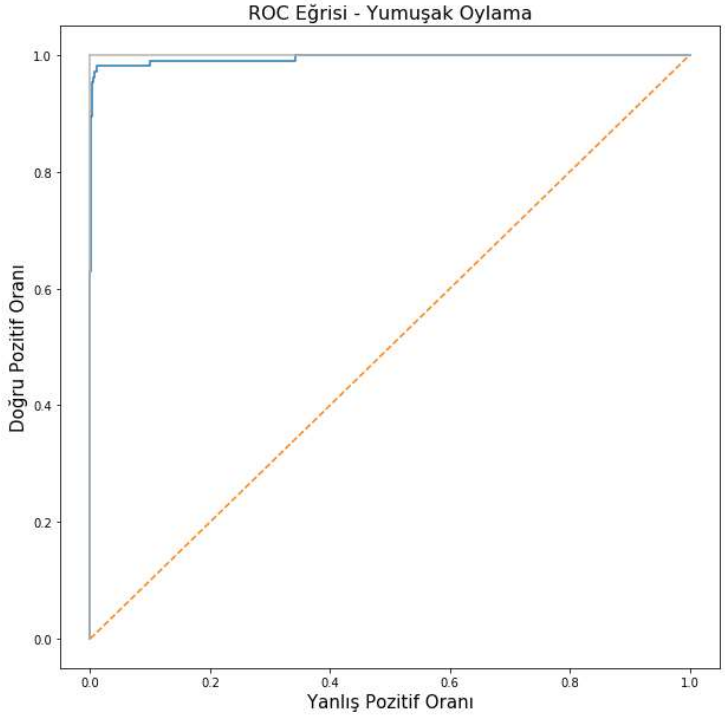
Kullanılan sınıflandırma tekniklerinin ve topluluk öğrenme yöntemlerinin yararlanılan performans ölçümlerine göre elde edilen sonuçları Tablo 18 aracılığıyla sunulmuştur. Bu tabloda AUC, doğruluk, hassasiyet, belirlilik, kesinlik ve F1 hesaplamaları yer almaktadır. Çalışmadaki sınıflandırma modelleri ve yöntemleri arasından söz konusu sınıflandırma performans ölçülerine göre en iyi performansa sahip olanların tablo içerisindeki ilgili hücreleri gölgelendirilmiştir. Tablo 18 'de tüm performans ölçümleri içinden en yüksek AUC, doğruluk, hassasiyet ve F1 değerleri yumuşak oylama yönteminde belirlenmiştir. Kesinlik değeri basit çoğunluk oylama yönteminde nispeten daha yüksek olmakla birlikte, tablo içindeki en yüksek değer olan belirlilik ölçüm değeri her iki oylama yönteminde de eşit şekilde 0,996 olarak ölçülmüştür. Tablonun genel sonuçları göz önünde bulundurulduğunda performans ölçüm sonuçları en yüksek olan yöntemin yumuşak oylama yöntemi olduğu söylenebilir.

Tablo 18: Senaryo 1: Performans Ölçüm Sonuçları

Yöntem	AUC	Doğruluk	Hassasiyet	Belirlilik	Kesinlik	F1
RF	0,994	0,990	0,933	0,992	0,700	0,800
LR	0,961	0,978	0,895	0,980	0,482	0,627
SVM	0,978	0,978	0,924	0,979	0,485	0,636
KNN	0,977	0,984	0,895	0,985	0,563	0,691
NB	0,982	0,987	0,848	0,990	0,650	0,736
Basit Çoğunluk	-	0,994	0,933	0,996	0,824	0,875
Yumuşak Oylama	0,995	0,995	0,952	0,996	0,820	0,881

İlk senaryonun yumuşak oylama yöntemiyle elde edilen AUC değerine ilişkin ROC eğrisi Şekil 26 'da gösterilmiştir. ROC eğrisi yanlış pozitif oranına karşı doğru pozitif oranını farklı eşik değerleri için değerlendirerek çizen bir olasılık eğrisidir. Basit çoğunluk oylamasında olasılık değerleri değil, çoğunluğu oluşturan oy sayıları temel alındığı için bu yöntemde ROC eğrisi çizilemez. Dolayısı ile ROC eğrisinin altında kalan alandan hesaplanan AUC değeri de hesaplanamaz. Şekil 26 'da mavi ile gösterilen ROC eğrisi grafiğin sol üst köşesine, 0,95 üzerinde seyrettiği için ilk senaryoda test edilen yumuşak oylama modelinin sınıflandırmada başarılı olduğuna işaret etmektedir.

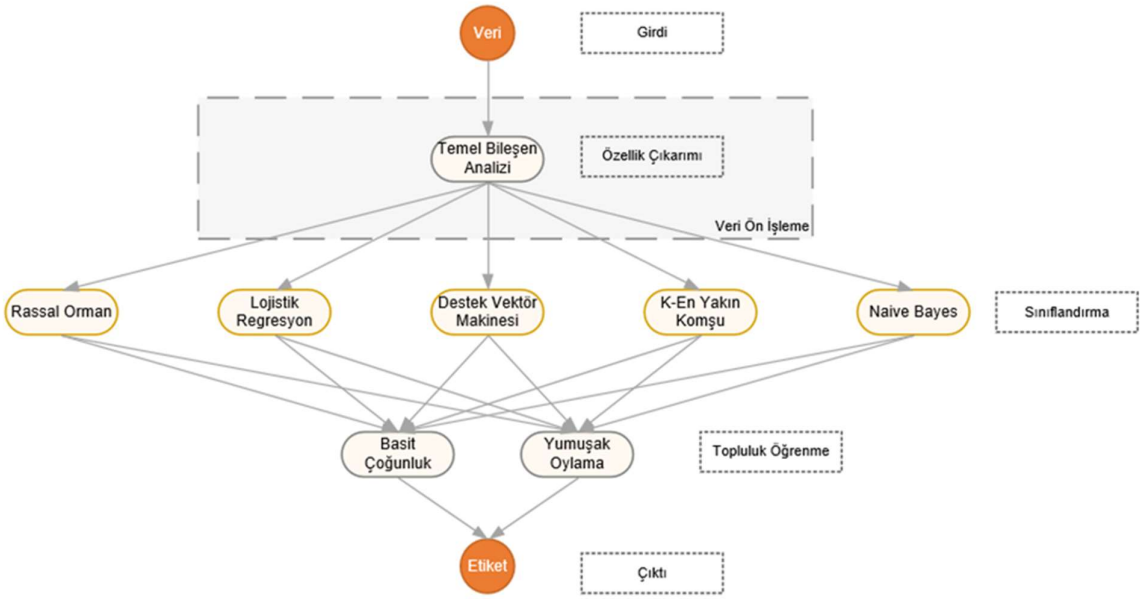
Şekil 26: Senaryo 1: Yumuşak Oylamaya Ait ROC Eğrisi



3.3.2.2. Senaryo 2: Değişken Çıkarımı İle Öğrenme

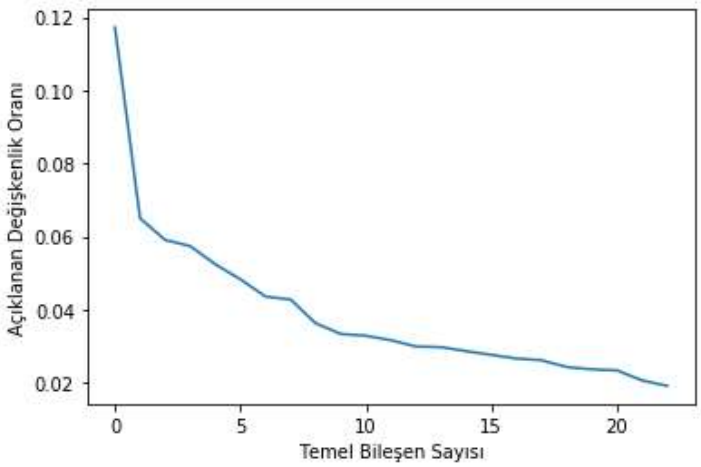
İkinci senaryoda sırasıyla özellik çıkarımı, sınıflandırma teknikleri ve topluluk öğrenme yöntemleri kullanılmıştır. Bu senaryo ile tüm değişkenleri kullanmak yerine, özellik çıkarımında faydalanılan ana bileşenler PCA tekniği ile yararlanılacak değişken sayısını azaltarak elde edilecek sonuçların tahmin doğruluğunun arttırılması amaçlanmıştır. Böylece bu senaryo ile özellik çıkarımının sonuçlar üzerindeki etkisi gözlenmiştir. PCA aracılığıyla mevcut değişkenliğin %90 'ını açıklayacak şekilde baz vektör uzayında dönüşüm yapılmıştır. Şekil 27, senaryo 2 'ye yönelik akış diyagramını göstermektedir.

Şekil 27: Senaryo 2: Yöntem Akışı



Şekil 28 'deki grafik, temel bileşen sayısı arttıkça açıklanan değişkenliğin yüzdesel olarak değişimini göstermektedir. PCA sonucu toplamda 38 değişken içinden boyut 23 değişkene indirgenmiştir. Bu 23 değişkenle varyansın %90 'ı açıklanabilmiştir.

Şekil 28: Senaryo 2: Temel Bileşen Analizi



Boyut indirgeme yapıldıktan sonra makine öğrenmesi yöntemleri eğitim veri seti üzerinde çalıştırılmıştır. Ardından test veri seti ile yapılan testin karmaşıklık matrisi sonuçları aşağıdaki Tablo 19-25 ile sunulmuştur.

Tablo 19: Senaryo 2: Rassal Orman Karmaşıklık Matrisi

Rassal Orman		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4922	73
	Sahteci (1)	5	100

Tablo 20: Senaryo 2: Lojistik Regresyon Karmaşıklık Matrisi

Lojistik Regresyon		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4756	239
	Sahteci (1)	11	94

Tablo 21: Senaryo 2: Destek Vektör Makinesi Karmaşıklık Matrisi

Destek Vektör Makinesi		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4874	121
	Sahteci (1)	8	97

Tablo 22: Senaryo 2: K-En Yakın Komşu Karmaşıklık Matrisi

K-En Yakın Komşu	Tahmin	
	Masum (0)	Sahteci (1)

Gerçek	Masum (0)	4934	61
	Sahteci (1)	12	93

Tablo 23: Senaryo 2: Naif Bayes Karmaşıklık Matrisi

Naif Bayes		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4624	371
	Sahteci (1)	32	73

Tablo 24: Senaryo 2: Basit Çoğunluk Topluluk Öğrenme Karmaşıklık Matrisi

Basit Çoğunluk		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4941	54
	Sahteci (1)	7	98

Tablo 25: Senaryo 2: Yumuşak Oylama Topluluk Öğrenme Karmaşıklık Matrisi

Yumuşak Oylama		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4942	53
	Sahteci (1)	11	94

Kullanılan sınıflandırma tekniklerinin ve topluluk öğrenme yöntemlerinin yararlanılan performans ölçümlerine göre elde edilen sonuçları Tablo 26 aracılığıyla sunulmuştur. Tablo 26 'da performans ölçümleri en yüksek olan sonuçlar renklendirilmiştir. Belirlilik değeri hem basit çoğunluk hem de yumuşak oylamada eşit hesaplanmıştır. Diğer ölçümlere bakıldığında en yüksek AUC değeri hem rassal ormanda hem yumuşak oylamada eşit olarak hesaplanmıştır. En yüksek hassasiyet değeri rassal ormanda; en yüksek doğruluk, kesinlik ve F1 değerleri ise basit çoğunluk oylamada ölçülmüştür. Tüm performans ölçümleri arasından en fazla sayıda yüksek değerin elde edildiği basit çoğunluk oylaması, ikinci senaryo için genel olarak en iyi performans gösteren yöntem olarak belirlenmiştir.

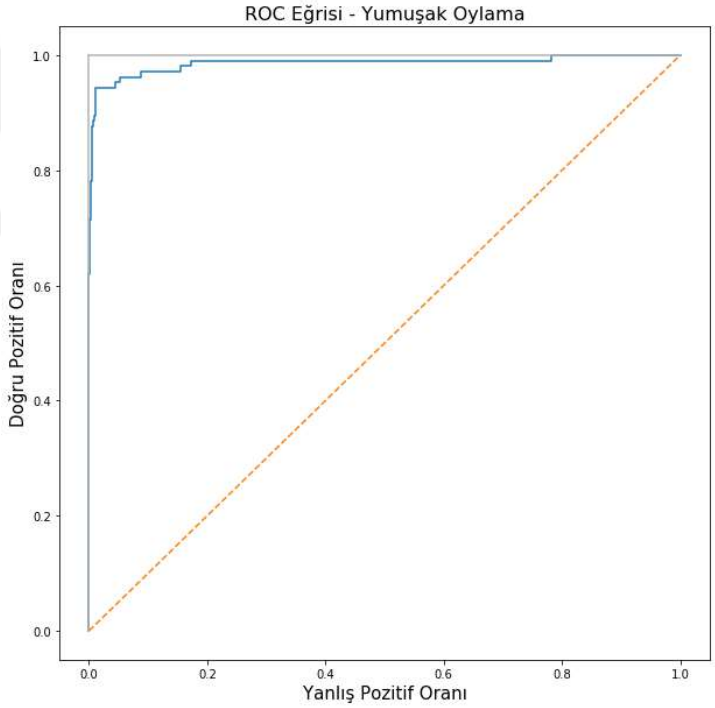
Tablo 26: Senaryo 2: Performans Ölçüm Sonuçları

Yöntem	**AUC**	**Doğruluk**	**Hassasiyet**	**Belirlilik**	**Kesinlik**	**F1**
RF	0,986	0,985	0,952	0,985	0,578	0,719
LR	0,970	0,951	0,895	0,952	0,282	0,429
SVM	0,965	0,975	0,924	0,976	0,445	0,601

KNN	0,973	0,986	0,886	0,988	0,604	0,718
NB	0,900	0,921	0,695	0,926	0,164	0,266
Basit Çoğunluk	-	0,988	0,933	0,989	0,645	0,763
Yumuşak Oylama	0,986	0,987	0,895	0,989	0,639	0,746

İkinci senaryoda basit çoğunluk seçilmesine rağmen basit çoğunluğa göre ROC eğrisi çizilemediği için, AUC değeri en yüksek olanlardan yumuşak oylamada ROC eğrisi çizilmiştir. Senaryonun yumuşak oylama çıktısı AUC değerine ilişkin ROC eğrisi Şekil 29 'da verilmiştir. ROC eğrisi, yanlış pozitif oranı ekseninden ne kadar uzaklaşırsa başarı seviyesi o kadar artar. Şekil 29 'da ROC eğrisinin, yanlış pozitif oranı ekseni ile uzaklığına bakarak yumuşak oylamanın başarı seviyesinin oldukça yüksek olduğu söylenebilir.

Şekil 29: Senaryo 2: Yumuşak Oylamaya Ait ROC Eğrisi

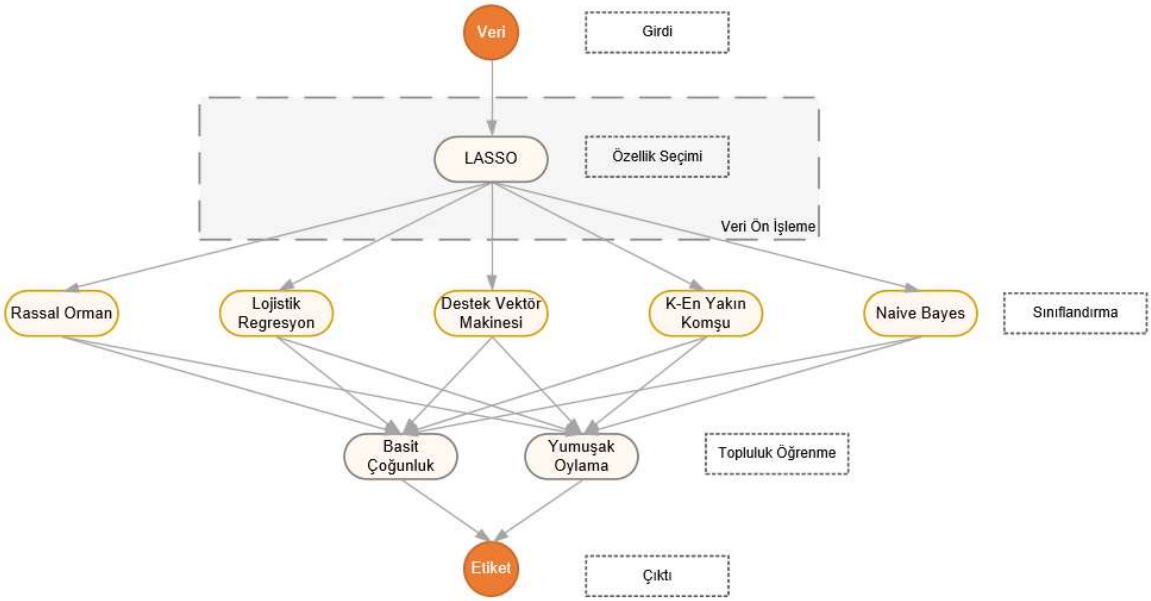


3.3.2.3. Senaryo 3: Değişken Seçimi İle Öğrenme

Senaryo 3 için sırasıyla özellik (değişken) seçimi, sınıflandırma teknikleri ve topluluk öğrenme yöntemlerinden faydalanılmıştır. Bu senaryo ile orijinal değişkenler üzerinden birbirinden farklı bilgi taşıyan önemli değişkenleri bir dönüşüm yapmadan seçerek, bu daraltılmış değişken havuzu üzerine kurulan sınıflandırma tekniklerinin elde edilen sonuçlara etkisinin incelenmesi amaçlanmıştır. Bu aşamada değişken

seçimi için sıklıkla kullanılan LASSO modeli tercih edilmiştir. Şekil 30, senaryo 3 için oluşturulan yöntem akışını göstermektedir.

Şekil 30: Senaryo 3: Yöntem Akışı



Üçüncü senaryo akışı gereği değişkenleri seçme yoluna gidilmiştir. Bu amaçla LASSO yöntemi kullanılarak 38 değişken arasından 29 değişken seçilmiştir. LASSO modeli içerisinde bulunan α parametresi 10 katlı çapraz doğrulama yapılarak 0,00256 olarak belirlenmiştir. Seçilen değişkenler ile söz konusu modeller kurulmuştur. Analiz sonucunda elde edilen karmaşıklık matrisi değerleri Tablo 27-33 'de verilmiştir.

Tablo 27: Senaryo 3: Rassal Orman Karmaşıklık Matrisi

Rassal Orman		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4954	41
	Sahteci (1)	7	98

Tablo 28: Senaryo 3: Lojistik Regresyon Karmaşıklık Matrisi

Lojistik Regresyon		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4901	94
	Sahteci (1)	11	94

Tablo 29: Senaryo 3: Destek Vektör Makinesi Karmaşıklık Matrisi

Destek Vektör Makinesi	Tahmin

		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4878	117
	Sahteci (1)	10	95

Tablo 30: Senaryo 3: K-En Yakın Komşu Karmaşıklık Matrisi

K-En Yakın Komşu		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4908	87
	Sahteci (1)	11	94

Tablo 31: Senaryo 3: Naif Bayes Karmaşıklık Matrisi

Naif Bayes		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4956	39
	Sahteci (1)	17	88

Tablo 32: Senaryo 3: Basit Çoğunluk Topluluk Öğrenme Karmaşıklık Matrisi

Basit Çoğunluk		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4970	25
	Sahteci (1)	6	99

Tablo 33: Senaryo 3: Yumuşak Oylama Topluluk Öğrenme Karmaşıklık Matrisi

Yumuşak Oylama		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4970	25
	Sahteci (1)	7	98

Kullanılan sınıflandırma tekniklerinin ve topluluk öğrenme yöntemlerinin yararlanılan performans ölçülerine göre elde edilen sonuçları Tablo 34 aracılığıyla sunulmuştur. Performans ölçüm sonuçlarının en yüksek değerleri topluluk öğrenme yöntemleri tarafından elde edilmiştir. Her iki topluluk öğrenme yönteminde de neredeyse eşit sonuçlar alınmış, ufak sapmalar gözlemlenmiştir. Tüm performans değerleri karşılaştırıldığında en çok sayıda ve en yüksek performans ölçüm sonuçları nispeten basit çoğunluk oylama yönteminde hesaplanmıştır. Bu sebeple senaryo 3

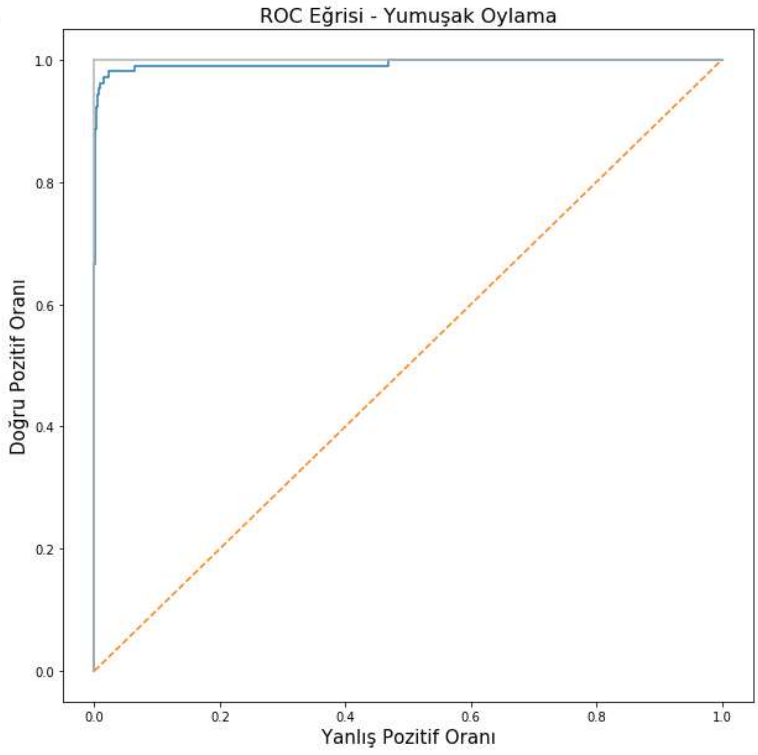
için en iyi sınıflandırma performansına ulaşan yöntemin basit çoğunluk olduğu belirlenmiştir.

Tablo 34: Senaryo 3: Performans Ölçüm Sonuçları

Yöntem	AUC	Doğruluk	Hassasiyet	Belirlilik	Kesinlik	F1
RF	0,993	0,991	0,933	0,992	0,705	0,803
LR	0,969	0,979	0,895	0,981	0,500	0,642
SVM	0,975	0,975	0,905	0,977	0,448	0,599
KNN	0,975	0,981	0,895	0,983	0,519	0,657
NB	0,984	0,989	0,838	0,992	0,693	0,759
Basit Çoğunluk	-	0,994	0,943	0,995	0,798	0,865
Yumuşak Oylama	0,994	0,994	0,933	0,995	0,797	0,860

Üçüncü senaryonun yumuşak oylama çıktısıyla en yüksek olarak elde edilen AUC değerine ilişkin ROC eğrisi Şekil 31'de verilmiştir. Şekildeki ROC eğrisi, doğru pozitif oranının yer aldığı eksene oldukça yakın bir konumda seyretmekte ve böylece eğri altında %100 'e yakın bir alan oluşturmaktadır. Bu da modelin tahmin performansının oldukça başarılı olduğunu göstermektedir.

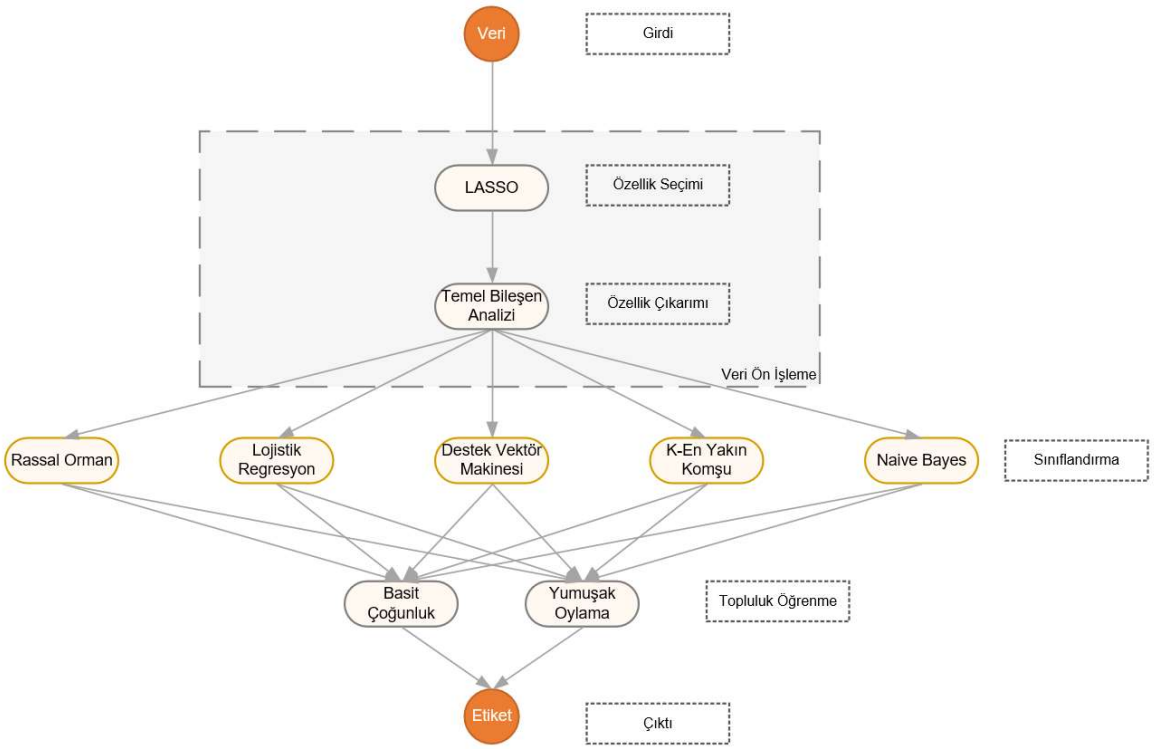
Şekil 31: Senaryo 3: Yumuşak Oylamaya Ait ROC Eğrisi



3.3.2.4. Senaryo 4: Değişken Seçimi ve Çıkarımı İle Öğrenme

Çalışma kapsamında oluşturulan senaryo 4 'de sırasıyla özellik seçimi, özellik çıkarımı, sınıflandırma teknikleri ve topluluk öğrenme yöntemlerinden faydalanılmıştır. Özellik seçimi için LASSO modelinden, özellik çıkarımı için PCA yönteminden yararlanılmıştır. Bu senaryo aracılığıyla, öncelikle birbirinden farklı bilgi taşıyan önemli değişkenlerin seçilerek değişken havuzunun küçültülmesi ve bu küçültülen değişken havuzunda değişkenliği en fazla açıklayacak şekilde dönüşüm yapılarak, elde edilecek yeni ve daha küçük boyutlu değişken kümesinin sınıflandırma performansı üzerindeki etkisi belirlenmeye çalışılmıştır. Şekil 32, senaryo 4 için oluşturulan yöntem akışını göstermektedir.

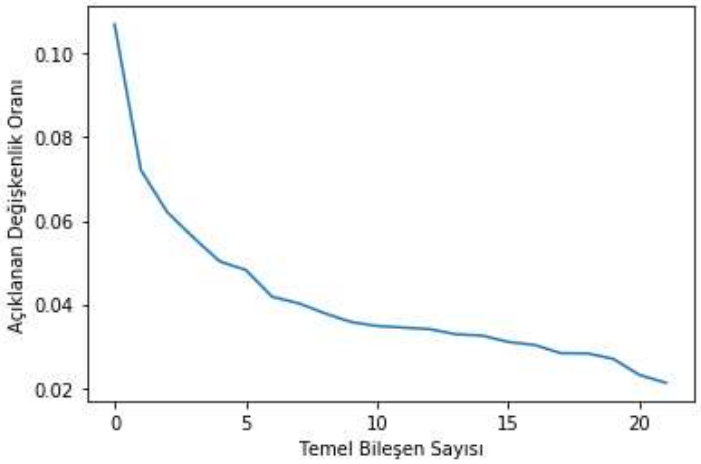
Şekil 32: Senaryo 4: Yöntem Akışı



Değişken seçimi amacıyla LASSO yöntemi kullanılarak 38 değişken arasından 29 değişken seçilmiştir. LASSO modeline ait α parametresi 10 katlı çapraz doğrulama yapılarak 0,00256 olarak belirlenmiştir.

LASSO ile özellik seçimi yapıldıktan sonra PCA ile özellik çıkarımı yapılarak boyut indirgenmiştir. Şekil 33 'deki grafik, temel bileşen sayısı arttıkça açıklanan değişkenliğin yüzdesel olarak değişimini göstermektedir. LASSO yönteminde 29 değişken seçilmesinin ardından PCA kullanılarak bu 29 değişkene ait toplam varyansın %90,9 'u açıklanacak şekilde 22 değişkene boyut indirgenmiştir.

Şekil 33: Senaryo 4: Temel Bileşen Analizi



Bu prosedür aracılığıyla kurulan sınıflandırma tekniklerine ait karmaşıklık matrisi değerleri Tablo 35-41 'de verilmiştir.

Tablo 35: Senaryo 4: Rassal Orman Karmaşıklık Matrisi

Rassal Orman		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4931	64
	Sahteci (1)	8	97

Tablo 36: Senaryo 4: Lojistik Regresyon Karmaşıklık Matrisi

Lojistik Regresyon		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4781	214
	Sahteci (1)	9	96

Tablo 37: Senaryo 4: Destek Vektör Makinesi Karmaşıklık Matrisi

Destek Vektör Makinesi		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4876	119
	Sahteci (1)	7	98

Tablo 38: Senaryo 4: K-En Yakın Komşu Karmaşıklık Matrisi

K-En Yakın Komşu	Tahmin	
	Masum (0)	Sahteci (1)

Gerçek	Masum (0)	4895	100
	Sahteci (1)	17	88

Tablo 39: Senaryo 4: Naif Bayes Karmaşıklık Matrisi

Naif Bayes		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4452	543
	Sahteci (1)	25	80

Tablo 40: Senaryo 4: Basit Çoğunluk Topluluk Öğrenme Karmaşıklık Matrisi

Basit Çoğunluk		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4922	73
	Sahteci (1)	8	97

Tablo 41: Senaryo 4: Yumuşak Oylama Topluluk Öğrenme Karmaşıklık Matrisi

Yumuşak Oylama		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4927	68
	Sahteci (1)	10	95

Kullanılan sınıflandırma tekniklerinin ve topluluk öğrenme yöntemlerinin yararlanılan performans ölçülerine göre elde edilen sonuçları Tablo 42 aracılığıyla sunulmuştur. Tablo 42 'de hassasiyet ölçümüne göre en yüksek değer, destek vektör makinesi olarak bulunmuştur. Geri kalan tüm ölçüler için rassal orman sınıflandırma yöntemi, en yüksek performansa sahiptir. Tablonun en yüksek ölçümü, yine rassal orman sınıflandırma değerinde hesaplanan %98,8 'lik AUC değeridir. Senaryo 4 kapsamında topluluk öğrenme yöntemlerinin, sınıflandırma gücünü arttırmada başarılı olamadığı söylenebilir. En yüksek performansa sahip yöntem olan rassal orman, senaryo 4 için nihai sınıflandırma kararını verecek yöntem olarak seçilmiştir.

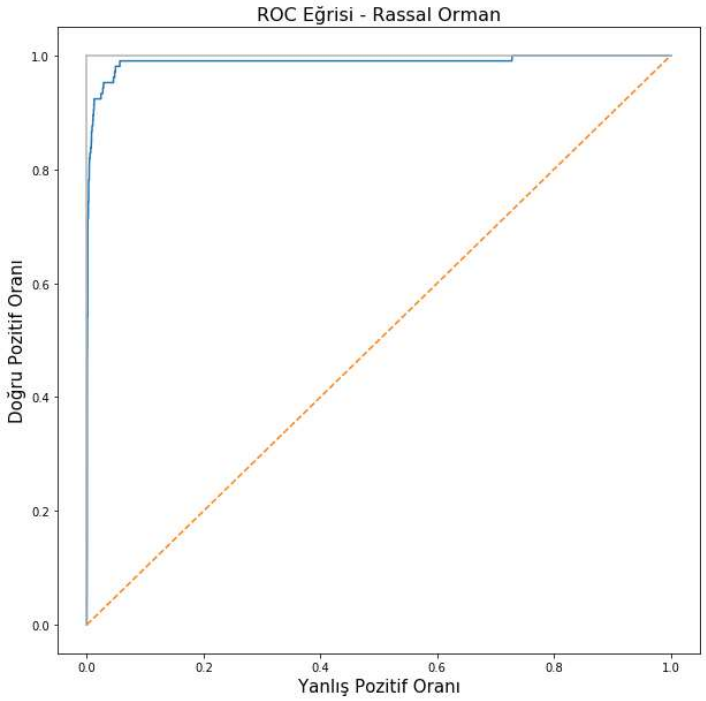
Tablo 42: Senaryo 4: Performans Ölçüm Sonuçları

Yöntem	**AUC**	**Doğruluk**	**Hassasiyet**	**Belirlilik**	**Kesinlik**	**F1**
RF	0,988	0,986	0,924	0,987	0,602	0,729
LR	0,972	0,956	0,914	0,957	0,310	0,463
SVM	0,979	0,975	0,933	0,976	0,452	0,609
KNN	0,978	0,977	0,838	0,980	0,468	0,601
NB	0,905	0,889	0,762	0,891	0,128	0,220

Basit Çoğunluk	-	0,984	0,924	0,985	0,571	0,705
Yumuşak Oylama	0,985	0,985	0,905	0,986	0,583	0,709

Dördüncü senaryo sonucunda en yüksek performans ölçüm değerlerine sahip olan rassal orman çıktısıyla elde edilen AUC değerine ilişkin ROC eğrisi Şekil 34 'de verilmiştir. ROC eğrisinin yatay eksene olan uzaklığı sebebiyle eğri altında kalan bölgenin değeri 1 'e yakın olduğundan, dördüncü senaryoda rassal orman yönteminin sınıflandırma başarısı oldukça yüksektir.

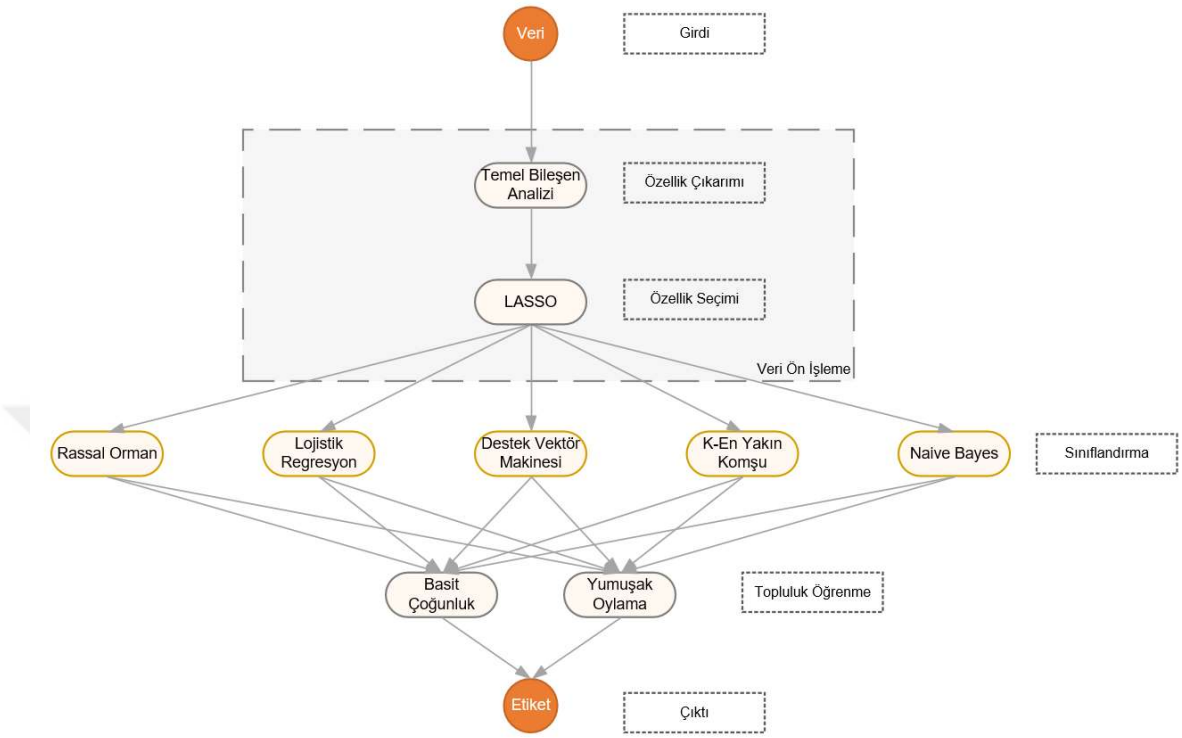
Şekil 34: Senaryo 4: Rassal Ormana Ait ROC Eğrisi



3.3.2.5. Senaryo 5: Değişken Çıkarımı ve Seçimi İle Öğrenme

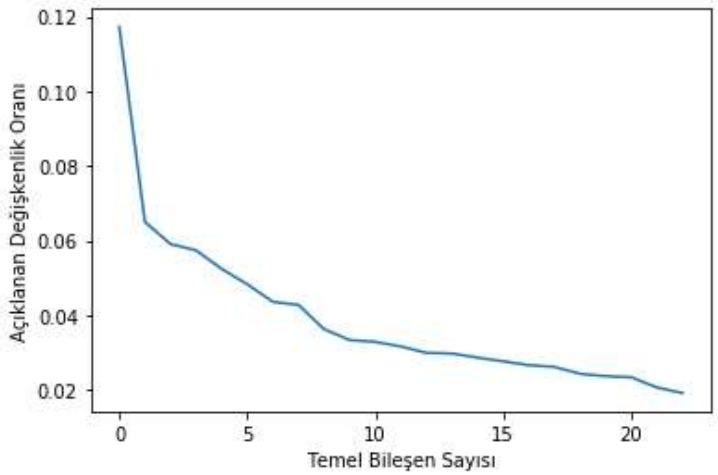
Bu senaryoda sırasıyla özellik çıkarımı, özellik seçimi, sınıflandırma teknikleri ve topluluk öğrenme yöntemleri uygulanmıştır. Amaç, öncelikle değişken dönüşümü sayesinde daha az değişkenle toplam değişkenliği açıklamak ve ardından bu yeni değişken uzayında önemli değişkenleri seçerek değişken havuzunu bir kez daha daraltmaktır. Böylece nihai olarak elde edilen dönüştürülmüş ve indirgenmiş değişkenlerle yapılan sınıflandırmanın performans istatistikleri üzerinden etkisi tespit edilmiştir. Şekil 35, senaryo 5 için oluşturulan yöntem akışını göstermektedir. Özellik çıkarımı için PCA yönteminden, özellik seçimi için LASSO modelinden yararlanılmıştır.

Şekil 35: Senaryo 5: Yöntem Akışı



Beşinci senaryo gereği ilk olarak değişken çıkarımı yöntemlerinden PCA 'ya başvurulmuştur. Şekil 36 'daki grafik, temel bileşen sayısı arttıkça açıklanan değişkenliğin yüzdesel olarak değişimini göstermektedir. PCA sonucunda varyansın %90 'ı açıklanacak şekilde 38 değişken içinden 23 değişkene boyut indirgenmiştir.

Şekil 36: Senaryo 5: Temel Bileşen Analizi



Bu prosedür ile oluşturulan sınıflandırma tekniklerine ait karmaşıklık matrisi değerleri Tablo 43-49 arasında verilmiştir.

Tablo 43: Senaryo 5: Rassal Orman Karmaşıklık Matrisi

Rassal Orman		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4912	83
	Sahteci (1)	8	97

Tablo 44: Senaryo 5: Lojistik Regresyon Karmaşıklık Matrisi

Lojistik Regresyon		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4720	275
	Sahteci (1)	20	85

Tablo 45: Senaryo 5: Destek Vektör Makinesi Karmaşıklık Matrisi

Destek Vektör Makinesi		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4859	136
	Sahteci (1)	8	97

Tablo 46: Senaryo 5: K-En Yakın Komşu Karmaşıklık Matrisi

K-En Yakın Komşu		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4914	81
	Sahteci (1)	14	91

Tablo 47: Senaryo 5: Naif Bayes Karmaşıklık Matrisi

Naif Bayes		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4681	314
	Sahteci (1)	33	72

Tablo 48: Senaryo 5: Basit Çoğunluk Topluluk Öğrenme Karmaşıklık Matrisi

Basit Çoğunluk		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4929	66
	Sahteci (1)	14	91

Tablo 49: Senaryo 5: Yumuşak Oylama Topluluk Öğrenme Karmaşıklık Matrisi

Yumuşak Oylama		Tahmin	
		Masum (0)	Sahteci (1)
Gerçek	Masum (0)	4929	66
	Sahteci (1)	12	93

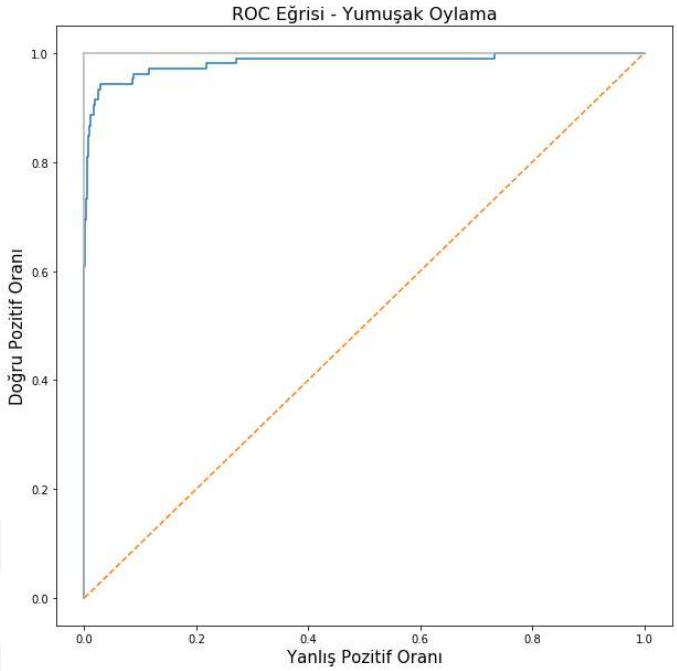
Kullanılan sınıflandırma tekniklerinin ve topluluk öğrenme yöntemlerinin yararlanılan performans ölçülerine göre elde edilen sonuçları Tablo 50 'de gösterilmiştir. Tablodaki tüm değerler karşılaştırıldığında, her iki topluluk öğrenme yöntemi de belirlilik performans istatistiğine göre %98,7 olarak ölçülen en yüksek değere sahiptir. Tablo 50 'ye göre tüm performans ölçüm teknikleri içinden hassasiyet dışındakilerde en yüksek sonuçlar yumuşak oylama yönteminde elde edilmiştir. Bu sebeple beşinci ve son senaryo için sınıf etiketlerinin belirlenmesinde nihai karar olarak yumuşak oylama topluluk öğrenme yönteminin kullanılması gerektiğine karar verilmiştir.

Tablo 50: Senaryo 5: Performans Ölçüm Sonuçları

Yöntem	**AUC**	**Doğruluk**	**Hassasiyet**	**Belirlilik**	**Kesinlik**	**F1**
RF	0,983	0,982	0,924	0,983	0,539	0,681
LR	0,965	0,942	0,810	0,945	0,236	0,366
SVM	0,971	0,972	0,924	0,973	0,416	0,574
KNN	0,980	0,981	0,867	0,984	0,529	0,657
NB	0,906	0,932	0,686	0,937	0,187	0,293
Basit Çoğunluk	-	0,984	0,867	0,987	0,580	0,695
Yumuşak Oylama	0,983	0,985	0,886	0,987	0,585	0,705

Beşinci senaryonun yumuşak oylama çıktısıyla elde edilen AUC değerine ilişkin ROC eğrisi şekil 37 'de verilmiştir. Son senaryoda ROC eğrisi (0,1) koordinat değerlerine, diğer senaryoların ROC eğrilerine göre daha uzakta bulunduğu için yumuşak oylamanın sınıflandırma başarısı önceki senaryolara nispeten daha düşük elde edilmiştir. Yine de grafikte yanlış pozitif oranına karşılık oldukça yüksek bir seviyede doğru pozitif oranı gözlendiği için yumuşak oylama yönteminin genel başarı düzeyinin oldukça yüksek olduğu söylenebilir. Bu durumda sahteci-masum kullanıcı sınıflarının gerçek durumları ile tahmin sonuçları yüksek derecede uyumludur.

Şekil 37: Senaryo 5: Yumuşak Oylamaya Ait ROC Eğrisi



3.3.3. Senaryolar Arasından Seçim

Belirlenen yöntemlerin doğru ve etkin bir şekilde çalışabilmesi için performans ölçüm değerlerinin karşılaştırılarak performansı güçlü bir senaryonun ve bu senaryonun en etkili çalışan yönteminin seçilmesi gerekmektedir. Bu amaçla performans ölçütleri içinden çalışmanın beklenen çıktısına uygun şekilde bir ölçüt seçilerek, bu ölçütün çıktıları değerlendirilmelidir. Performans değerlendirme ölçütlerinin her biri, modeli farklı bir yönden ölçer. Bu sebeple doğru ölçüte güvenmek için model iyi bir şekilde tanınmalıdır. Örneğin uygulamadaki etiketli veri seti dengeli ise doğruluk ölçütüne güvenilebilir. Ancak bir sahtekarlık analizinde sahteci veri etiketleri, masum veri etiketlerine göre oldukça düşük sayıda olacağından bu tip uygulamalarda doğruluk ölçütüne tek başına güvenilmemelidir. Eğer sahtekarlık analizinde sahteci olarak tahmin edilenlerin kaçının gerçekte sahteci olduğu önemli ise hassasiyet, masum olarak tahmin edilen kullanıcıların kaçının gerçekte masum olduğu önemli ise belirlilik ölçütü kullanılabilir. Ancak sistemde sahtekar kullanıcıların yarattığı maliyet daha büyükse ve kesin olarak gerçekte sahteci olanların kaçının doğru tahmin edildiği önemliyse kesinlik ölçütüne güvenilebilir.

Bu çalışmada hem yanlış negatifler hem de yanlış pozitifler eşit derecede önemlidir. Çünkü çalışmanın yürütüldüğü firma için, masum olduğu halde sahteci olarak işaretlenen kullanıcıların kaybedilme maliyeti ile sahteci olduğu halde masum olarak işaretlenen kullanıcıların yarattığı sahtecilik maliyeti eşit önem arz etmektedir.

Bu sebeple yanlış negatiflerle hesaplanan hassasiyet ve yanlış pozitiflerle hesaplanan kesinlik ölçütü arasında harmonik ortalama alarak bir denge kuran F skoru, çalışmanın performansı için seçilen performans istatistiği olmuştur. Tablo 51 'de her bir senaryoda tüm yöntemlere göre elde edilmiş F1 değerleri bir araya getirilmiştir. Bu değerler karşılaştırıldığında en yüksek F1 değerinin %88,1 olarak ölçüldüğü ve bu değerin senaryo 1 için yumuşak oylama yönteminde hesaplandığı görülmektedir.

Tablo 51: F1 Skoruna Göre Senaryolar

F1 Skoru	1.Senaryo	2.Senaryo	3.Senaryo	4.Senaryo	5.Senaryo
RF	0,800	0,719	0,803	0,729	0,681
LR	0,627	0,429	0,642	0,463	0,366
SVM	0,636	0,601	0,599	0,609	0,574
KNN	0,691	0,718	0,657	0,601	0,657
NB	0,736	0,266	0,759	0,220	0,293
Basit Çoğunluk	0,875	0,763	0,865	0,705	0,695
Yumuşak Oylama	**0,881**	0,746	0,860	0,709	0,705

Diğer performans ölçütleri ile F1 arasında bir çatışma varsa çalışmanın tahmin gücünün etkilenip etkilenmeyeceğini görmek için önceki bölümde sunulmuş olan her bir senaryoda seçilen yöntemin performans ölçütleri kullanılarak Tablo 52 hazırlanmıştır. Tablodaki diğer skor değerleri incelendiğinde ise, tüm performans ölçüm değerlerinin en yüksek olduğu durumun yine senaryo 1 olduğu görülmektedir. Bu durumda yalnızca F1 skoru için değil tüm performans ölçüm değerlerinin uzlaşma sağladığı senaryo 1 'deki yumuşak oylama yöntemi, tüm çalışmanın en iyi sınıflandırma performansını vermektedir. Dolayısıyla çalışmada herhangi bir değişken indirgeme ve değişken seçimi olmaksızın tüm değişkenler ile sınıflandırma yapılarak sonuçların yumuşak oylama ile birleştirilmesi, çalışma için en iyi performansı vermektedir.

Tablo 52: Tüm Senaryoların En İyi Yöntemlerine Göre Performans Ölçümleri

Senaryo	Yöntem	AUC	Doğruluk	Hassasiyet	Belirlilik	Kesinlik	F1
1	Yumuşak Oylama	0,995	0,995	0,952	0,996	0,820	0,881
2	Basit Çoğunluk	-	0,988	0,933	0,989	0,645	0,763
3	Basit Çoğunluk	-	0,994	0,943	0,995	0,798	0,865
4	RF	0,988	0,986	0,924	0,987	0,602	0,729
5	Yumuşak Oylama	0,983	0,985	0,886	0,987	0,585	0,705

3.3.4. Tehdit Skoru

Senaryolardan elde edilen sonuca göre performans istatistiği en iyi olan Senaryo 1 'de kullanılan yumuşak oylama yöntemi temel alınarak kullanıcıların tehdit skorları ve kara listede olma durumları ele alınmıştır. Yumuşak oylama tarafından yanlış sınıfta tahmin edilen 27 kişinin tehdit skorları incelenmiştir. Tehdit skorları, kullanıcıların sahtekar olma ihtimallerini gösteren olasılık değerlerinin 100 ile çarpılması ile oluşturulmuştur. Tehdit skorları, önceden belirlenen eşik değerlerine göre (0-50: Güvenilir, 50-85: Riskli, 85-100: Tehdit) bölgelerine ayrılmıştır. Riskli bölgedeki kullanıcıların hangi sınıfa dahil olması gerektiğine sistem sorumlusu tarafından karar verilmek istendiğinden, bu kullanıcılar karmaşıklık matrisindeki yanlış etiketlenmiş bölüme dahil edilmemelidir. Kullanıcıları kara liste sınıflandırmasına göre sınıfladıktan sonra tehdit skorlarına göre bölgelere ayırmak, yanlış tahmin edilen kullanıcı sayısının azalmasında etkili olmuştur. Bu durumda tüm değişkenlerin kullanıldığı yumuşak oylama yöntemi tarafından yanlış sınıflandırılan 27 kişiden 3 'ünün riskli bölgede olduğu, başka bir deyişle bu 3 kişinin sistem sorumlusu tarafından kontrol edilerek sınıflandırılacağı anlaşılmıştır. Hem otomatik, hem riskli bölgelerin gözle kontrolü sonrası, test veri setindeki 5100 kullanıcının 24 'ünün yanlış tahmin edildiği söylenebilir. Tablo 53 'de kullanıcıların kara listede olma olasılıkları ile hesaplanan tehdit skorları ve tehdit skorlarının derecelerini temsil eden tehdit bölgeleri verilmiştir.

Tablo 53: Tehdit Skorlarına Göre Tehdit Bölgelerinin Belirlenmesi

Kullanıcılar	Tehdit Skoru (p*100)	Tehdit Bölgesi
1	0,013	Güvenilir Bölge
2	4,911	Güvenilir Bölge
3	5,984	Güvenilir Bölge
4	9,465	Güvenilir Bölge
5	20,552	Güvenilir Bölge
6	53,898	Riskli Bölge
7	57,058	Riskli Bölge
8	57,573	Riskli Bölge
9	90,000	Tehdit Bölgesi
10	92,656	Tehdit Bölgesi
11	93,606	Tehdit Bölgesi
12	93,964	Tehdit Bölgesi
13	94,365	Tehdit Bölgesi
14	95,765	Tehdit Bölgesi
15	95,883	Tehdit Bölgesi

16	95,986	Tehdit Bölgesi
17	96,637	Tehdit Bölgesi
18	96,641	Tehdit Bölgesi
19	97,295	Tehdit Bölgesi
20	97,614	Tehdit Bölgesi
21	97,899	Tehdit Bölgesi
22	98,346	Tehdit Bölgesi
23	98,639	Tehdit Bölgesi
24	98,839	Tehdit Bölgesi
25	98,966	Tehdit Bölgesi
26	99,069	Tehdit Bölgesi
27	99,306	Tehdit Bölgesi

SONUÇ

Bu çalışmada, bir akıllı ulaştırma sistemleri işletmesi olan Kentkart ‘ın ABD ‘nin Kansas eyaletinde toplu taşıma mobil uygulamasına kayıtlı kullanıcılarının hareketleri göz önünde bulundurularak, kullanıcıların kara listede olup olmama durumlarının makine öğrenmesi yöntemleri ile otomatik olarak sınıflandırılması üzerine çalışılmıştır. Bu sayede gerçek bir toplu ulaşım sistemlerinde kullanılan bir mobil bilet uygulamasının güvenliğinin arttırılması hedeflenmiştir.

Biletler farklı teknolojiler kullanılarak yolcuların toplu taşıma araçları ile seyahat etmelerini sağlayan; kişisel bilgiler, ödeme yöntemleri, bilet tipleri ve kullanım hakları, bilet bedelleri gibi hassas ve korunması gereken bilgiler içeren uygulamalardır. Mobil bilet uygulaması kısa sürede kolay erişilebilirlik, kolay taşıma ve kullanma gibi avantajları sayesinde fiziki biletlere göre daha çok tercih edilmektedir. Gerek müşteri memnuniyeti ve sürdürülebilirliği gerekse şirket itibarı ve karlılığı sebepleriyle elektronik bilet gereksinimlerinin karşılanması işletmede hayati önem taşır. Biletin sunucudan güvenli bir şekilde mobil uygulamalara indirilmesi, toplu taşıma içinde doğrulanması ve onaylanması, mobil cihaz içerisinde bilet verisinin bütünlüğünün ve özgünlüğünün sağlanması, kopyalanamaması, devredilememesi, çalınamaması gibi ihtiyaçlar mobil uygulamanın güvenliği konusu içerisinde yer almaktadır. Bu gereksinimleri karşılamak için mobil işletim sistemlerinin sağladığı özellikler ve sunucu teknolojileri değerlendirilerek, çeşitli teknikler geliştirilmektedir. Güvenlik ihtiyaçların yanı sıra uygulamanın ve kullanıcıların gizliliğinin korunması gerekliliği de son derece önemlidir.

Her uygulamada olabileceği gibi mobil biletleme uygulamasında da sahtekarlık içeren davranışlar, ihlalci kullanıcılar bulunur. Uygulamaların bu kullanıcıların yol açabileceği olası zararlardan korunması gerekmektedir. Aksi takdirde, mobil uygulama; kullanım amacının suistimali, doğru kullanıcılara erişilememesi, satış ve kullanım tahminlemesinde hatalar, itibar, güven ve müşteri kaybı gibi zararlara uğrar. Sahtekarlıklardan korunmayan bir mobil uygulamada bu zararlara gerekçe oluşturan aşağıdaki durumların gözlenmesi muhtemeldir:

- Kullanıcının kişisel verilerinin, üçüncü şahısların eline geçmesi.
- Satın alınan biletin kopyalanarak aynı kişi tarafından tekrarlı kullanılması.
- Satın alınan biletin kopyalanarak farklı kişilere dağıtılması veya yasal olmayan yöntemler ile satılması.
- Bilet verisinin değiştirilmesi.

- Mobil uygulamanın kodlarına erişim sağlanarak, aynı görüntüye sahip farklı bir mobil uygulamanın devreye alınması, bu uygulama ile ücret ödemeden sahte bilet üretilmesi.
- Kötü niyetli kişiler tarafından uygulamanın güvensiz olarak çalıştığının gösterilmesiyle firma itibarının zedelenmesi.
- Sistemde güvenilir olmayan, risk taşıyan ve sistemi tehdit eden kullanıcıların gözle tespit edilme zorluğu, bu yolla tüm riskli kullanıcıların belirlenememesi.

Toplu taşıma uygulamasının sahtekarlıklardan, ihlalci kullanıcılardan korunabilmesi amacıyla bir yöntem geliştirilmesi ihtiyacını karşılamak için bu çalışma gerçekleştirilmiştir. Bu çalışmada toplu taşıma uygulamasındaki açıkları kötüye kullanan, çalıntı kredi kartı ile işlem yapan, kendine ait olmayan hesapları ele geçiren, şüpheli ve riskli hareketler sergileyen kullanıcıları tespit etmek hedeflenmiştir. Bu doğrultuda otomatik kara liste kararının verildiği makine öğrenme tekniklerine dayalı bir yöntem geliştirilmiştir. Bu sayede yüksek sayıda kullanıcısı bulunan gerçek bir sistemin ihlalci kullanıcılardan korunma ihtiyacı karşılanmış, sistemde mevcut sahtekarlıkların tespit edilerek bu kullanıcıların otomatik olarak kara listeye alınması sağlanmıştır.

Çalışma kapsamında ilk olarak mobil uygulama ve elektronik ödeme piyasasında ve aynı zamanda uygulamanın yapıldığı işletmede var olan riskler ve olası riskli durumlar tespit edilmiştir. Ücretlendirme/biletlendirme riskleri, kullanıcı riskleri, uygulama riskleri ve işletim sistemi riskleri olmak üzere dört temel risk kategorisi belirlenmiş ve her bir kategori altında riskli durumlar listelenmiştir. Risk analizi yapılarak işletmenin risk planı hazırlanmıştır. Risk planının sonucunda aksiyon derecesi en yüksek olması sebebiyle, belirlenen riskler içinden öncelikli olarak aksiyon alınması gereken risk kodunun RUB10 olduğu belirlenmiştir. Aksiyon derecesi en düşük olan risk kodu ise RUB7 olarak bulunmuştur. Alınması gereken aksiyon öncelikleri ile paralel olarak her bir riskle ilişkilendirilecek bağlamlar türetilmiştir. Bağlamlar riskleri belirlemeye yönelik, kullanıcıların risk profillerini oluşturan sorgular olarak alınmıştır. Her bir riske karşılık en az bir bağlam mevcutken, bir bağlam birden fazla riske işaret edebilmektedir. Bağlam sorguları özellikle önem dereceleri yüksek olan riskleri temsil edecek şekilde hazırlanmıştır. Böylece mobil uygulamadan ve ödeme sistemlerinden alınan veriler ile sistem risklerini karşılayan, bağlam duyarlı bir veri toplama modeli oluşturulmuştur. Model gereği kullanıcıların tüm bağlam verileri toplanarak bir veri havuzuna aktarılmıştır. Tüm değişkenlerin toplandığı bağlam havuzunun son halinde: kullanıcı numaraları, kullanıcının kara

listede olup olmadığını gösteren kara liste durumları ve bağlam değişken değerlerini gösteren (C1, C2, … ,C63) kolonlar mevcuttur. İlgili bağlam değişkenlerinin tamamı sayısal değerlerden oluşmaktadır. Oluşturulan sistemin canlı olarak çalışabilmesi ve değişen kullanıcı hareketlerine adapte olabilmesi için bu veriler belirlenen bir zaman parametresinde (gün sonu olarak belirlenen) yeniden toplanmakta ve böylece kullanıcıların sistem içindeki davranış geçmişine göre bir profili oluşturulmaktadır.

Sahtekarlık tespit mekanizması kullanıcı profilinin değerlendirilmesi aşamasında devreye girer. Kullanıcı profillerindeki sapmalar değerlendirilerek, eşik değerini aşan kullanıcılar için alarm üretilmesi sağlanmıştır. Sapma değerlendirmesi ise makine öğrenmesi algoritmalarından alınan sonuçlar ile gerçekleştirilmiştir. Veri seti içerisindeki sınıf dengesizliğini gidermek ve doğru şekilde etiketlenmiş veriler üzerine model kurmak için toplanan tüm verilerin yalnızca gözle doğrulanabilmiş kısmı örnek olarak alınmıştır. Sahtekarlık tespit sistemi için tüm kullanıcılar içerisinden örnek bir veri seti oluşturulmuş ve örnek veri setinin yaklaşık %37 'si eğitim, %63 'ü test veri seti olarak ayrılmıştır. Eğitim veri setinin yarısı kara listede bulunan kullanıcılar olarak alınmışken, ana kütleye uyumlu olarak test veri setinin %1,96 'sı kara listede bulunan kullanıcılar ile oluşturulmuştur.

Sahtekarlık tespit mekanizmasının üç aşaması bulunmaktadır. Bu aşamaların sıralaması belirlenen senaryolarda değişiklik göstermektedir. İlk aşama, uygulamadaki toplanan değişkenler arasındaki ilişkiler göz önünde bulundurularak varsa gereksiz değişkenlerin ya da bağımlı değişkeni benzer açıklama gücüne sahip değişkenlerin göz ardı edilmesidir. Bunun için bu çalışmada veri ön işleme adımında boyut indirgeme problemine yönelik olarak özellik seçimi ve çıkarımı tekniklerine başvurulmuştur. Değişken çıkarımı için temel bileşen analizi, değişken seçimi için LASSO modeli kullanılmıştır. Bu tekniklerle elde edilen sonuçlara dayanarak eldeki değişken havuzu üzerinde daraltma yapılmıştır. Bu aşamanın ardından ikinci aşamaya geçilmiştir.

İkinci aşama çalışmanın odak noktası olan kara liste kararının verileceği yerdir. Bağlam havuzundan alınan son değişkenler bu aşamada belirlenen makine öğrenmesi algoritmaları tarafından kullanılmıştır. Uygulamada sistem sorumlusu tarafından belirlenmiş kara liste ve beyaz liste etiketleri mevcut olduğu için sınıf etiketlerinden yararlanan gözetimli öğrenmeden faydalanılmıştır. Bu doğrultuda RF, LR, SVM, KNN ve NB olmak üzere 5 farklı sınıflandırma yöntemine başvurulmuştur.

Sınıflandırma algoritmaları arasından faydalanılan performans ölçülerine dayanarak bir seçim yapmaktansa, sahtekarlık tespit mekanizmasının son aşaması

için topluluk öğrenme yöntemi uygulanmıştır. Böylece algoritmaların güçlü yönleri birleştirilerek tüm algoritmalardan daha iyi tahmin performansı veren bir öğrenme mekanizması kurulmuş, sınıflandırmanın güvenilirliği arttırılmış ve toplam tahmin hatası düşürülmüştür. Topluluk öğrenmesi için oylama tekniği kullanılarak sınıflandırıcıların tahminleri üzerinden oylama yapılmıştır. En uygun oylama yönteminin belirlenmesi içinse basit çoğunluk oylaması ve yumuşak oylama olmak üzere iki farklı oylama yöntemine başvurulmuştur.

Öğrenmenin üç aşaması tamamlandıktan sonra, tespit mekanizmasından kullanıcının kara listede olma durumları ile birlikte kara listede olma olasılıkları da alınmıştır. Bu olasılık değeri uygulama içinde tehdit skorunu belirlemek için kullanılmıştır. Ardından, tehdit skoru sistem yöneticisi tarafından belirlenen güven aralıkları ile oluşturulan bölgelere ayrılmıştır. Eğer bu skor bir kullanıcı için belirlenen eşik değerinin üzerindeyse kullanıcı tehdit bölgesinde yer alır, bu durumda kullanıcı otomatik olarak kara listeye alınır. Skor, belirlenen eşik değerinin altındaysa kullanıcı güvenilir bölgededir, bu durumda kullanıcı otomatik olarak beyaz listeye alınır. Eğer sonuç kara liste eşik değerinin altında ama riskli bir bölgedeyse sistem yalnızca alarm verir, sistem sorumlusunun durumu inceleyerek kara listeye alma kararını gerçekleştirmesi beklenir.

Uygulamaya ve verilere uygun tespit mekanizmasının belirlenebilmesi için çalışma kapsamında beş farklı senaryo hazırlanmıştır. Tüm senaryoların karmaşıklık matrisleri karşılaştırıldığında hemen hemen her seferinde en az hatanın topluluk öğrenme yöntemlerinde olduğu gözlenmiştir. Bu sebeple tüm senaryolar için nihai sonucun topluluk öğrenme yöntemleri çıktılarından alınmasına karar verilmiştir. Uygulamaya topluluk öğrenme yöntemlerini dahil etmenin çalışma üzerinde anlamlı bir katkı sağladığı gözlenmiştir. Aşağıda her bir senaryoya göre elde edilen sonuçlar verilmiştir.

İlk senaryoda herhangi bir değişken seçimi ya da değişken çıkarımı kullanmadan tüm değişkenlerle sınıflandırma ve ardından topluluk öğrenme yöntemi gerçekleştirilmiştir. İlk senaryo çıktılarına göre sınıflandırma algoritmaları arasında tüm performans ölçüleri karşılaştırıldığında, topluluk öğrenme yöntemleri arasında kayda değer bir fark olmadığı gözlemlenmiştir. Ancak karmaşıklık matrisindeki yanlış tahminler arasındaki fark 1 kullanıcı olarak tespit edilmiş ve performans ölçülerinin nispeten daha iyi sonuçlar üretmesi sebebiyle, bu senaryodaki en iyi sonuç yumuşak oylama olarak belirlenmiştir. Bu senaryoda, yumuşak oylama yönteminde toplam 27 kişi yanlış sınıflandırılmıştır. Bunlardan 22 ‘si yanlış pozitif yani kara listede olmadığı

halde kara listede olarak tahminlenen kullanıcılardır. Kalan 5 ‘i ise yanlış negatifleri yani kara listede olduğu halde kara listede değil şeklinde tahminlenen kullanıcıları göstermektedir.

İkinci senaryoda PCA ile gerçekleştirilen özellik çıkarımı sonucunda, dönüştürülmüş değişkenler üzerinden 15 değişkenlik boyut uygulamadan çıkarılmıştır. 23 değişkenlik boyut ile sınıflandırma algoritmaları ve ardından topluluk öğrenme oylamaları yapılmıştır. Oylama sonucunda en az yanlışın ve en iyi performans çıktılarının basit çoğunluk yönteminde olduğu görülmüştür. Bu senaryoda, yanlış tahminlenen toplam 61 kişiden 54 ‘ü yanlış pozitif, 7 ‘si ise yanlış negatif değerlerine atanmıştır.

Üçüncü senaryoda değişken seçimi yapan LASSO modeli aracılığı ile 38 değişkenin 29 ‘u seçilerek, bu değişkenlerle sınıflandırma teknikleri ve topluluk öğrenmesi yöntemleri oluşturulmuştur. Topluluk öğrenmesi yöntemlerinin her ikisinde de 25 kişilik yanlış pozitif tahminlemesi yapıldığı görülmüştür. Yanlış negatif olarak tahminlenen değerlere bakıldığında ise basit çoğunluk yönteminde 6 kişi, yumuşak oylamada ise 7 kişi olarak tahminlendiği görülmüştür. 1 kişilik fark performans ölçülerinden yalnızca hassasiyet ölçütünde %1 ‘lik farka yol açmıştır. Böylece basit çoğunluk öğrenmesi üçüncü senaryonun en iyi sonucu olarak seçilmiştir.

Dördüncü senaryoda önce LASSO modeli ile 29 değişken seçilmiş, ardından temel bileşen analizi ile 29 değişkenden oluşan boyut 22 değişkene indirgenmiştir. Kalan değişkenlerle belirlenen algoritmalar çalıştırıldığında ise performans ölçümlerinin en yüksek olduğu ve en az yanlış sınıflandırmanın olduğu yöntem, 64 yanlış negatif, 8 yanlış pozitif sınıflandırılması ile RF yöntemi olarak belirlenmiştir. Böylece ilk kez dördüncü senaryoda topluluk öğrenme yöntemleri dışında bir yöntem senaryonun nihai kararı olarak belirlenmiştir.

Uygulamanın beşinci ve son senaryosu için önce PCA ile uygulama boyutu 15 birim kadar küçültülerek 23 değişkene indirgenmiş böylece 23 değişken kullanılmıştır. Ardından 23 dönüştürülmüş değişken arasından LASSO modeliyle 14 değişken seçilerek, belirlenen makine öğrenmesi algoritmaları uygulanmıştır. Topluluk öğrenme yöntemleri kıyaslandığında yanlış pozitifler her iki oylama yönteminde de aynı tahminlenirken, yanlış negatifler yumuşak oylama yönteminde 2 kişi daha az olarak tahmin edilmiştir. Bu sebeple performans ölçümleri de daha yüksek olan yumuşak oylama tercih edilmiştir.

Tüm senaryolarda belirlenen en iyi kararların performans ölçüleri ve karmaşıklık matrisleri karşılaştırılması ardından en iyi sonuca 1. senaryo ile ulaşıldığı

görülmüştür. Dolayısı ile sahtekarlık tespit mekanizması için tüm değişkenler kullanılarak yapılan ve topluluk öğrenme yöntemi olarak yumuşak oylama kullanılan senaryonun uygulanmasına karar verilmiştir.

Verilen senaryo ve yöntem kararının ardından elde edilen tahmini oluşturan sahtekar olma olasılıkları incelenmiştir. Olasılıklar uygulamada tehdit skorunun elde edilmesinde değerlendirilmiş ve tehdit skorları işletme açısından üç güvenilirlik bölgesine ayrılmıştır. Böylece uygulamada, makine öğrenmesi çıktılarının yanında bir de tehdit skorları baz alınarak belirlenen tehdit bölgeleri ile ikinci güvenlik katmanı geliştirilmiştir. Test veri seti içinde yanlış etiketli olarak tahminlenen 27 kullanıcının tehdit skorları değerlendirildiğinde; bu kullanıcıların 5 'i güvenilir bölgede, 19 'u tehdit bölgesinde, 3 'ü de iki bölge arasında kalan şüpheli olarak bakılan riskli bölgede olarak belirlenmiştir. Bu durumda tehdit bölgesinde ve güvenilir bölgede olarak yer alan toplam 24 kişi otomatik olarak sınıflandırılmış, 3 kişi ise son kez sistem sorumlusu tarafından kontrol edilerek yanlış sınıflandırmadan kurtarılmıştır. Böylece test veri setindeki yanlış sınıflandırma oranı %0,53 'den %0,47 'ye düşürülmüştür.

Çalışma kapsamında mobil uygulamadaki riskli davranışlar sergileyen ihlalci kullanıcıları tespit eden algoritma geliştirilmiş olup bu kullanıcılar otomatik olarak kara listeye alınmıştır. Geliştirilen öğrenme tabanlı sistem, yeni riskleri ve saldırıları gerçekleştiren kişileri kısa zamanda, etkin şekilde ve minimum sapma ile belirlemesi yönü ile öne çıkmaktadır. Kullanılan yöntem sahtekarlık tespitindeki binlerce farklı olası risk kuralları kombinasyonları tanımlama yükünden kurtardığı gibi, verilerin kullanıldığı işletmeye hem zaman hem de maliyet optimizasyonu sağlamıştır. Söz konusu algoritma ile mevcutta kara listede olan kullanıcıların atak seviyeleri ve sisteme genel davranış biçimleri otomatik olarak öğrenilerek bu çıkarım doğrultusunda diğer kullanıcıların hareketleri sistem tarafından incelenmiş ve kullanıcıların riskli olup olmadığına karar verilmiştir. Belli bir tehdit skoru seviyesini aşan kullanıcılar otomatik olarak kara listeye alınmış, böylece gözle kontrol sayısı önemli ölçüde azaltılmıştır. Gelecek çalışma kapsamında, bir mobil ya da web uygulamasının tüm kullanıcılarını makine öğrenme teknikleri ile otomatik olarak sınayarak kullanıcılarının risk skorlarını gösteren ve böylece güvenilir olan ve olmayanların belirlenebildiği bir yapı oluşturulabilir.

KAYNAKÇA

Abdallah, A., Maarof, M. A., & Zainal, A. (2016). Fraud detection system: A survey. Journal of Network ve Computer Applications, 68, 90-113.

Abe, S. (2005). Support vector machines for pattern classification, London: Springer. (Vol. 2, p. 44).

Abowd, G. D., Dey, A. K., Brown, P. J., Davies, N., Smith, M., & Steggles, P. (1999, September). Towards a better understanding of context ve context-awareness. In International symposium on handheld ve ubiquitous computing, Springer, Berlin, Heidelberg, (pp. 304-307).

Adomavicius, G., & Tuzhilin, A. (1999, August). User profiling in personalization applications through rule discovery ve validation. In Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery ve data mining (pp. 377-381).

Agrawal, S., & Agrawal, J. (2015). Survey on anomaly detection using data mining techniques. Procedia Computer Science, 60, 708-713.

Ahmed, M., Mahmood, A. N., & Islam, M. R. (2016). A survey of anomaly detection techniques in financial domain. Future Generation Computer Systems, 55, 278-288.

Akhilomen, J. (2013, July). Data mining application for cyber credit-card fraud detection system. In Industrial Conference on Data Mining, Springer, Berlin, Heidelberg, (pp. 218-228).

Albrecht, W. S., Albrecht, C. O., Albrecht, C. C., & Zimbelman, M. F. (2011). Fraud examination. Cengage Learning.

Alexopoulos, P., Kafentzis, K., Benetou, X., Tagaris, T., & Georgolios, P. (2007, July). Towards a Generic Fraud Ontology in e-Government. In ICE-B (pp. 269-276).

Algamal, Z. Y., & Lee, M. H. (2015). Penalized logistic regression with the adaptive LASSO for gene selection in high-dimensional cancer classification. Expert Systems with Applications, 42(23), 9326-9332.

Amoako, K. B. (2011). The Effect Of Delayed Payment On Cash Flow Forecasting Of Ghanaian Road Contractors (Doctoral dissertation).

Asokan, N., Janson, P. A., Steiner, M., & Waidner, M. (1997). The state of the art in electronic payment systems. Computer, 30(9), 28-35.

Baesens, B., Van Vlasselaer, V., & Verbeke, W. (2015). Fraud analytics using descriptive, predictive, ve social network techniques: a guide to data science for fraud detection. John Wiley & Sons.

Ballabio, D., Grisoni, F., & Todeschini, R. (2018). Multivariate comparison of classification performance measures. Chemometrics ve Intelligent Laboratory Systems, 174, 33-44.

Becher, M., Freiling, F. C., Hoffmann, J., Holz, T., Uellenbeck, S., & Wolf, C. (2011, May). Mobile Security Catching Up? Revealing The Nuts Ve Bolts Of The Security Of Mobile Devices. In 2011 IEEE Symposium on Security ve Privacy, IEEE, pp. 96-111.

Becker, T. (2015). Variable Selection, BSc report Applied Mathematics.

Behdad, M., Barone, L., Bennamoun, M., & French, T. (2012). Nature-inspired techniques in the context of fraud detection. IEEE Transactions on Systems, Man, ve

Benaroch, M. (2002). Managing information technology investment risk: A real options perspective. Journal of management information systems, 19(2), 43-84.

Blum, A. L., & Langley, P. (1997). Selection of relevant features ve examples in machine learning. Artificial intelligence, 97(1-2), 245-271.

Bolboaca, S. D., & Jäntschi, L. (2006). Pearson versus Spearman, Kendall's tau correlation analysis on structure-activity relationships of biologic active compounds. Leonardo Journal of Sciences, 5(9), 179-200.

Bolton, R. J., & Hand, D. J. (2001). Unsupervised profiling methods for fraud detection. Credit scoring ve credit control VII, 235-255.

Bolton, R. J., & Hand, D. J. (2002). Statistical fraud detection: A review. Statistical science, 235-249.

Bosch, A., Zisserman, A., & Munoz, X. (2007, October). Image classification using random forests ve ferns. In 2007 IEEE 11th international conference on computer vision, IEEE (pp. 1-8).

Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. A. (1984). Classification ve regression trees. CRC press.

Buczak, A. L., & Guven, E. (2015). A survey of data mining ve machine learning methods for cyber security intrusion detection. IEEE Communications surveys & tutorials, 18(2), 1153-1176.

Camenisch, J., Piveteau, J. M., & Stadler, M. (1996, January). An efficient fair payment system. In Proceedings of the 3rd ACM Conference on Computer ve Communications Security, pp. 88-94.

Cannady, J. (1998, October). Artificial neural networks for misuse detection. In National information systems security conference (Vol. 26, pp. 443-456).

Cantú, F., & Saiegh, S. M. (2011). Fraudulent democracy? An analysis of Argentina's infamous decade using supervised machine learning. Political Analysis, 19(4), 409-433.

Cao, J., Kwong, S., Wang, R., Li, X., Li, K., & Kong, X. (2015). Class-specific soft voting based multiple extreme learning machines ensemble. Neurocomputing, 149, 275-284.

Chan, P. K., Fan, W., Prodromidis, A. L., & Stolfo, S. J. (1999). Distributed data mining in credit card fraud detection. IEEE Intelligent Systems ve Their Applications, 14(6), 67-74.

Chandrashekar, G., & Sahin, F. (2014). A survey on feature selection methods. Computers & Electrical Engineering, 40(1), 16-28.

Chapelle, O., Scholkopf, B., & Zien, A. (2009). Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews], IEEE Transactions on Neural Networks, 20(3), 542-542.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research, 16, 321-357.

Chehata, N., Guo, L., & Mallet, C. (2009, September). Airborne lidar feature selection for urban classification using random forests.

Chen, X., & Ishwaran, H. (2012). Random forests for genomic data analysis. Genomics, 99(6), 323-329.

Chowdhury, M., Apon, A., & Dey, K. (Eds.). (2017). Data analytics for intelligent transportation systems, Elsevier.

Chung, C. Y., Gertz, M., & Levitt, K. (1999, November). Demids: A misuse detection system for database systems. In Working Conference on Integrity ve Internal Control in Information Systems, Springer, Boston, MA. (pp. 159-178).

Cutler, D. R., Edwards Jr, T. C., Beard, K. H., Cutler, A., Hess, K. T., Gibson, J., & Lawler, J. J. (2007). Random forests for classification in ecology. Ecology, 88(11), 2783-2792.

Dal Pozzolo, A., Caelen, O., Le Borgne, Y. A., Waterschoot, S., & Bontempi, G. (2014). Learned lessons in credit card fraud detection from a practitioner perspective. Expert systems with applications, 41(10), 4915-4928.

Dash, M., & Liu, H. (1997). Feature selection for classification. Intelligent data analysis, 1(3), 131-156.

Dave, R. (2010). Privacy in Mobile-ticketing. In Aalto University School of Science ve Technology Seminar on Network Security T-110.5290.

David, K., & Mitchel, K. (1994). Logistic regression: A self learning text.

Deisenroth, M. P., Faisal, A. A., & Ong, C. S. (2020). Mathematics for machine learning. Cambridge University Press.

Dewan, S. G., & Chen, L. D. (2005). Mobile payment adoption in the US: A cross-industry, crossplatform solution. Journal of Information Privacy ve Security, 1(2), 4-28.

Dietterich, T. G. (2002). Ensemble learning. The handbook of brain theory ve neural networks, 2, 110-125.

Domingos, P. (2012). A few useful things to know about machine learning. Communications of the ACM, 55(10), 78-87.

Domingos, P., & Pazzani, M. (1997). On The Optimality of The Simple Bayesian Classifier Under Zero-One Loss. Machine learning, 29(2-3), 103-130.

Duda, R. O., Hart, P. E., & Stork, D. G. (1973). Pattern classification ve scene analysis, New York: Wiley (Vol. 3, pp. 731-739).

Dy, J. G., & Brodley, C. E. (2004). Feature selection for unsupervised learning. Journal of machine learning research, 5(Aug), 845-889.

Erbacher, R. F., Walker, K. L., & Frincke, D. A. (2002). Intrusion ve misuse detection in large-scale systems. IEEE computer graphics ve applications, 22(1), 38-47.

Even, S., Goldreich, O., & Yacobi, Y. (1984). Electronic wallet. In Advances in Cryptology, Springer, Boston, MA, pp. 383-386.

Eze, U. C., Gan, G. G. G., Ademu, J., & Tella, S. A. (2008). Modelling User Trust Ve Mobile Payment Adoption: A Conceptual Framework. Communications of the IBIMA, 3(29), 224-231.

Ezhilvani, V., Malathi, K., & Nedunchelian, R. (2014). Rule Based Message Filtering Ve Blacklist Management For Online Social Network. IJRET, 3(06).

Fan, J., & Li, R. (2001). Variable selection via nonconcave penalized likelihood ve its oracle properties. Journal of the American statistical Association, 96(456), 1348-1360.

Fawcett, T., & Provost, F. (1997). Adaptive fraud detection. Data mining ve knowledge discovery, 1(3), 291-316.

Fawcett, T., & Provost, F. J. (1996, August). Combining Data Mining ve Machine Learning for Effective User Profiling. In KDD (pp. 8-13).

Fawcett, T., Haimowitz, I., Provost, F., & Stolfo, S. (1998). Ai approaches to fraud detection ve risk management. AI Magazine, 19(2), 107.

Friston, K. J., Frith, C. D., Liddle, P. F., & Frackowiak, R. S. J. (1993). Functional connectivity: the principal-component analysis of large (PET) data sets. Journal of Cerebral Blood Flow & Metabolism, 13(1), 5-14.

Fujimura, K., & Nakajima, Y. (1998, August). General-purpose Digital Ticket Framework. In USENIX Workshop on Electronic Commerce, pp. 177-186.

Furey, T. S., Cristianini, N., Duffy, N., Bednarski, D. W., Schummer, M., & Haussler, D. (2000). Support vector machine classification ve validation of cancer tissue samples using microarray expression data. Bioinformatics, 16(10), 906-914.

Galar, D. (2014). Context-driven Maintenance: an eMaintenance approach. Management Systems in Production Engineering.

Gao, J., Kulkarni, V., Ranavat, H., Chang, L., & Mei, H. (2009, June). A 2D barcode-based mobile payment system. In 2009 Third International Conference on Multimedia ve Ubiquitous Engineering, IEEE, 320-329.

Genuer, R., Poggi, J. M., & Tuleau-Malot, C. (2010). Variable selection using random forests. Pattern recognition letters, 31(14), 2225-2236.

Ghahramani, Z. (2015). Probabilistic machine learning ve artificial intelligence. Nature, 521(7553), 452-459.

Ghosh, A. K., & Swaminatha, T. M. (2001). Software security ve privacy risks in mobile e-commerce. Communications of the ACM, 44(2), 51-57.

Gislason, P. O., Benediktsson, J. A., & Sveinsson, J. R. (2006). Random forests for lve cover classification. Pattern Recognition Letters, 27(4), 294-300.

Godoy, D., & Amandi, A. (2005). User profiling in personal information agents: a survey. The Knowledge Engineering Review, 20(4), 329.

Grandvalet, Y., & Bengio, Y. (2004). Semi-supervised learning by entropy minimization. Advances in neural information processing systems (pp. 529-536).

Granitto, P. M., Furlanello, C., Biasioli, F., & Gasperi, F. (2006). Recursive feature elimination with random forest for PTR-MS analysis of agroindustrial products. Chemometrics ve Intelligent Laboratory Systems, 83(2), 83-90.

Grosser, H., Britos, P., & García-Martínez, R. (2005, June). Detecting fraud in mobile telephony using neural networks. In International Conference on Industrial, Engineering ve Other Applications of Applied Intelligent Systems, Springer, Berlin, Heidelberg, (pp. 613-615).

Gu, Q., Zhu, L., & Cai, Z. (2009, October). Evaluation measures of the classification performance of imbalanced data sets. In International symposium on intelligence computation ve applications, Springer, Berlin, Heidelberg. (pp. 461-471).

Gunn, S. R. (1998). Support vector machines for classification ve regression. ISIS technical report, 14(1), 5-16.

Guo, G., Wang, H., Bell, D., Bi, Y., & Greer, K. (2003, November). KNN model-based approach in classification. In OTM Confederated International Conferences" On the Move to Meaningful Internet Systems", Springer, Berlin, Heidelberg. (pp. 986-996).

Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. Machine learning, 46(1-3), 389-422.

Güven, Ö., & Hartoka, A. (2019), Automatization in Blacklist Management. In 2019 Innovations in Intelligent Systems ve Applications Conference (ASYU), IEEE, (pp. 1-6).

Hall, J., Barbeau, M., & Kranakis, E. (2005, August). Anomaly-based intrusion detection using mobility profiles of public transportation users. In WiMob'2005), IEEE International Conference on Wireless Ve Mobile Computing, Networking Ve Communications, 2005, IEEE, (Vol. 2, pp. 17-24).

Hall, M. A., & Smith, L. A. (1998). Practical feature subset selection for machine learning.

Hamill, T. M. (1999). Hypothesis tests for evaluating numerical precipitation forecasts. Weather ve Forecasting, 14(2), 155-167.

Han, H., Wang, W. Y., & Mao, B. H. (2005, August). Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning. In International conference on intelligent computing, Springer, Berlin, Heidelberg. (pp. 878-887).

Hashemi, M. R., & Soroush, E. (2006, May). A secure m-payment protocol for mobile devices. In 2006 Canadian Conference on Electrical ve Computer Engineering, IEEE, pp. 294-297.

Hassinen, M., Hyppönen, K., & Haataja, K. (2006, June). An open, PKI-based mobile payment system. In International Conference on Emerging Trends in Information ve Communication Security, Springer, Berlin, Heidelberg, pp. 86-100.

He, X., Cai, D., & Niyogi, P. (2006). Laplacian score for feature selection. In Advances in neural information processing systems (pp. 507-514).

Hechenbichler, K., & Schliep, K. (2004). Weighted k-nearest-neighbor techniques ve ordinal classification.

Held, A., Buchholz, S., & Schill, A. (2002). Modeling of context information for pervasive computing applications. Proceedings of SCI, 167-180.

Henricksen, K. (2003). A framework for context-aware pervasive computing applications.

Henricksen, K., Indulska, J., & Rakotonirainy, A. (2002, August). Modeling context information in pervasive computing systems. In International Conference on Pervasive Computing, Springer, Berlin, Heidelberg (pp. 167-180).

Hirschman, E. C. (1979). Differences in Consumer Purchase Behavior By Credit Card Payment System. Journal of Consumer Research, 6(1), 58-66.

Hofer, T., Schwinger, W., Pichler, M., Leonhartsberger, G., Altmann, J., & Retschitzegger, W. (2003, January). Context-awareness on mobile devices-the hydrogen approach. In 36th annual Hawaii international conference on system sciences, 2003, IEEE, (pp. 10).

Howard, M., & Lipner, S. (2006). The security development lifecycle (Vol. 8). Redmond: Microsoft Press.

Hwang, Y. S., Han, S. W., & Nam, T. Y. (2006, June). Secure rejoining scheme for dynamic sensor networks. In International Conference on Emerging Trends in Information ve Communication Security (pp. 101-114). Springer, Berlin, Heidelberg.

Ikonomakis, M., Kotsiantis, S., & Tampakas, V. (2005). Text classification using machine learning techniques. WSEAS transactions on computers, 4(8), 966-974.

Jackson, J. E., & Mudholkar, G. S. (1979). Control procedures for residuals associated with principal component analysis. Technometrics, 21(3), 341-349.

Jacob, L., Obozinski, G., & Vert, J. P. (2009, June). Group lasso with overlap ve graph lasso. In Proceedings of the 26th annual international conference on machine learning (pp. 433-440).

Jain, A. K., & Shanbhag, D. (2012). Addressing security ve privacy risks in mobile applications. IT Professional, 14(5), 28-33.

John, G. H., Kohavi, R., & Pfleger, K. (1994). Irrelevant Features ve The Subset Selection Problem. In Machine Learning Proceedings 1994 (pp. 121-129). Morgan Kaufmann.

Johnson, R. A., & Wichern, D. W. (2002). Applied multivariate statistical analysis (Vol. 5, No. 8). Upper Saddle River, NJ: Prentice hall.

Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, ve prospects. Science, 349(6245), 255-260.

Jyothsna, V. V. R. P. V., Prasad, V. R., & Prasad, K. M. (2011). A review of anomaly based intrusion detection systems. International Journal of Computer Applications, 28(7), 26-35.

Kakkar, K., Shah, R., & Kakkar, M. (2013). Risk analysis in mobile application development.

Kambhatla, N., & Leen, T. K. (1997). Dimension reduction by local principal component analysis. Neural computation, 9(7), 1493-1516.

Karegowda, A. G., Jayaram, M. A., & Manjunath, A. S. (2010). Feature Subset Selection Problem Using Wrapper Approach in Supervised Learning. International journal of Computer applications, 1(7), 13-17.

Karnouskos, S. (2004). Mobile payment: A Journey Through Existing Procedures Ve Standardization Initiatives. IEEE Communications Surveys & Tutorials, 6(4), 44-66.

Karnouskos, S., Hondroudaki, A., Vilmos, A., & Csik, B. (2004, July). Security, trust ve privacy in the secure mobile payment service. In 3rd International Conference on Mobile Business (Vol. 35).

Kim, C., Mirusmonov, M., & Lee, I. (2010). An empirical examination of factors influencing the intention to use mobile payment. Computers in Human Behavior, 26(3), 310-322.

Kim, G., Lee, S., & Kim, S. (2014). A novel hybrid intrusion detection method integrating anomaly detection with misuse detection. Expert Systems with Applications, 41(4), 1690-1700.

Kima, H., Kimb, H., Moonc, H., & Ahnb, H. (2011). A weight-Adjusted voting algorithm for ensemble of classifiers. Journal of the Korean Statistical Society, 40, 437-449.

Kirkos, E., Spathis, C., & Manolopoulos, Y. (2007). Data mining techniques for the detection of fraudulent financial statements. Expert systems with applications, 32(4), 995-1003.

Kohavi, R., & John, G. H. (1997). Wrappers for feature subset selection. Artificial intelligence, 97(1-2), 273-324.

Kohavi, R., & John, G. H. (1998). The wrapper approach. In Feature extraction, construction ve selection, Springer, Boston, MA (pp. 33-50).

Kohavi, R., & Sommerfield, D. (1995, August). Feature Subset Selection Using the Wrapper Method: Overfitting ve Dynamic Search Space Topology. In KDD (pp. 192-197).

Koller, D., & Sahami, M. (1996). Toward Optimal Feature Selection. Stanford InfoLab.

Kotsiantis, S. B., Zaharakis, I. D., & Pintelas, P. E. (2006). Machine learning: a review of classification ve combining techniques. Artificial Intelligence Review, 26(3), 159-190.

Kourou, K., Exarchos, T. P., Exarchos, K. P., Karamouzis, M. V., & Fotiadis, D. I. (2015). Machine learning applications in cancer prognosis ve prediction. Computational ve structural biotechnology journal, 13, 8-17.

Krulwich, B. (1997). Lifestyle finder: Intelligent user profiling using large-scale demographic data. AI magazine, 18(2), 37.

Ku, W., Storer, R. H., & Georgakis, C. (1995). Disturbance detection ve isolation by dynamic principal component analysis. Chemometrics ve intelligent laboratory systems, 30(1), 179-196.

Kumar, S., & Spafford, E. H. (1994). A pattern matching model for misuse intrusion detection.

Kursa, M. B., & Rudnicki, W. R. (2010). Feature selection with the Boruta package. J Stat Softw, 36(11), 1-13.

La Polla, M., Martinelli, F., & Sgandurra, D. (2012). A survey on security for mobile devices. IEEE communications surveys & tutorials, 15(1), 446-471.

Lewis, D. D. (1998, April). Naive (Bayes) at forty: The independence assumption in information retrieval. In European conference on machine learning, Springer, Berlin, Heidelberg. (pp. 4-15).

Li, T., Van Heck, E., & Vervest, P. (2009). Information capability ve value creation strategy: advancing revenue management through mobile ticketing technologies. European Journal of Information Systems, 18(1), 38-51.

Liao, Y., & Vemuri, V. R. (2002). Use of k-nearest neighbor classifier for intrusion detection. Computers & security, 21(5), 439-448.

Liaw, A., & Wiener, M. (2002). Classification ve regression by randomForest. R news, 2(3), 18-22.

Liébana-Cabanillas, F., Sánchez-Fernández, J., & Muñoz-Leiva, F. (2014). Antecedents of the adoption of the new mobile payment systems: The moderating effect of age. Computers in Human Behavior, 35, 464-478.

Lin, X., Yang, F., Zhou, L., Yin, P., Kong, H., Xing, W., ... & Xu, G. (2012). A support vector machine-recursive feature elimination feature selection method based on artificial contrast variables ve mutual information. Journal of chromatography B, 910, 149-155.

Linck, K., Pousttchi, K., & Wiedemann, D. G. (2006). Security issues in mobile payment from the customer viewpoint.

Listfield, R., & Montes-Negret, F. (1994). Modernizing Payment Systems in Emerging Economies (Vol. 1336). World Bank Publications.

Liu, C., Chan, Y., Alam Kazmi, S. H., & Fu, H. (2015). Financial fraud detection model: Based on random forest. International journal of economics ve finance, 7(7).

Lockamy III, A., & McCormack, K. (2010). Analysing risks in supply networks to facilitate outsourcing decisions. International Journal of Production Research, 48(2), 593-611.

Mao, J., & Jain, A. K. (1995). Artificial neural networks for feature extraction ve multivariate data projection. IEEE transactions on neural networks, 6(2), 296-317. Data. The Open Bioinformatics Journal, 11(1).

McAndrews, J. J. (1999). E□Money ve Payment System Risks. Contemporary Economic Policy, 17(3), 348-357.

McInnes, L., Healy, J., & Melville, J. (2018). Umap: Uniform manifold approximation ve projection for dimension reduction. arXiv preprint arXiv:1802.03426.

Melo-Acosta, G. E., Duitama-Muñoz, F., & Arias-Londoño, J. D. (2017, August). Fraud detection in big data using supervised ve semi-supervised learning techniques. In 2017 IEEE Colombian Conference on Communications ve Computing (COLCOM), IEEE, (pp. 1-6).

Mitra, P., Murthy, C. A., & Pal, S. K. (2002). Unsupervised feature selection using feature similarity. IEEE transactions on pattern analysis ve machine intelligence, 24(3), 301-312.

Monika, E., & Kaur, A. (2018). A comparative study of classification techniques for fraud detection. Int J Future Revolution Comput Sci Commun Eng, 4(5), 19-23.

Moreau, Y., Verrelst, H., & Vandewalle, J. (1997, October). Detection of mobile phone fraud using supervised neural networks: A first prototype. In International Conference on Artificial Neural Networks, Springer, Berlin, Heidelberg, (pp. 1065-1070).

Muhlbaier, M. D., Topalis, A., & Polikar, R. (2008). Learn++.NC: Combining Ensemble of Classifiers With Dynamically Weighted Consult-and-Vote for Efficient Incremental Learning of New Classes. IEEE transactions on neural networks, 20(1), 152-168.

Mulliner, C. R. (2006). Security of smart phones (Doctoral dissertation, University of California, Santa Barbara).

Ngai, E. W., Hu, Y., Wong, Y. H., Chen, Y., & Sun, X. (2011). The application of data mining techniques in financial fraud detection: A classification framework ve an academic review of literature. Decision support systems, 50(3), 559-569.

Nguyen, T. T., & Armitage, G. (2008). A survey of techniques for internet traffic classification using machine learning. IEEE communications surveys & tutorials, 10(4), 56-76.

Oxford University Press (2021), Oxford Learner's Dictionary, Erişim adresi: https://www.oxfordlearnersdictionaries.com/, Erişim tarihi: 09.01.2021.

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up? Sentiment classification using machine learning techniques. arXiv preprint cs/0205070.

Pang, H., Lin, A., Holford, M., Enerson, B. E., Lu, B., Lawton, M. P., ... & Zhao, H. (2006). Pathway analysis using random forests classification ve regression. Bioinformatics, 22(16), 2028-2036.

Patel, B., & Crowcroft, J. (1997, September). Ticket based service access for the mobile user. In Proceedings of the 3rd annual ACM/IEEE international conference on Mobile computing ve networking, pp. 223-233

Peng, C. Y. J., Lee, K. L., & Ingersoll, G. M. (2002). An introduction to logistic regression analysis ve reporting. The journal of educational research, 96(1), 3-14.

Phua, C., Lee, V., Smith, K., & Gayler, R. (2010). A comprehensive survey of data mining-based fraud detection research. arXiv preprint arXiv:1009.6119.

Pirker, M., & Slamanig, D. (2012, June). A framework for privacy-preserving mobile payment on security enhanced arm trustzone platforms. In 2012 IEEE 11th International Conference on Trust, Security ve Privacy in Computing ve Communications, IEEE, pp. 1155-1160.

Poggio, T., & Smale, S. (2003). The mathematics of learning: Dealing with data. Notices of the AMS, 50(5), 537-544.

Pope, P. T., & Webster, J. T. (1972). The use of an F-statistic in stepwise regression procedures. Technometrics, 14(2), 327-340.

Pousttchi, K., & Schurig, M. (2004, January). Assessment of today's mobile banking applications from the view of customer requirements. In 37th Annual Hawaii International Conference on System Sciences, 2004. Proceedings of the, IEEE, pp. 10-pp.

Prasad, A. M., Iverson, L. R., & Liaw, A. (2006). Newer classification ve regression tree techniques: bagging ve random forests for ecological prediction. Ecosystems, 9(2), 181-199.

Press, S. J., & Wilson, S. (1978). Choosing between logistic regression ve discriminant analysis. Journal of the American Statistical Association, 73(364), 699-705.

Pudil, P., Novovičová, J., & Kittler, J. (1994). Floating search methods in feature selection. Pattern recognition letters, 15(11), 1119-1125.

Qin, Z., Sun, J., Wahaballa, A., Zheng, W., Xiong, H., & Qin, Z. (2017). A secure ve privacy-preserving mobile wallet with outsourced verification in cloud computing. Computer Standards & Interfaces, 54, 55-60.

Rajora, S., Li, D. L., Jha, C., Bharill, N., Patel, O. P., Joshi, S., ... & Prasad, M. (2018, November). A comparative study of machine learning techniques for credit card fraud detection based on time variance. In 2018 IEEE Symposium Series on Computational Intelligence (SSCI), IEEE (pp. 1958-1963).

Rish, I. (2001, August). An empirical study of the Naive Bayes classifier. In IJCAI 2001 workshop on empirical methods in artificial intelligence (Vol. 3, No. 22, pp. 41-46).

Rosemann, M., Recker, J., Flender, C., & Ansell, P. D. (2006). Understanding context-awareness in business process design. In Proceedings of the 17th Australasian Conference on Information Systems, Australasian Association for Information Systems (pp. 1-10).

Rosset, S., Murad, U., Neumann, E., Idan, Y., & Pinkas, G. (1999, August). Discovery of fraud rules for telecommunications—challenges ve solutions. In Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery ve data mining (pp. 409-413).

Ryali, S., Supekar, K., Abrams, D. A., & Menon, V. (2010). Sparse logistic regression for whole-brain classification of fMRI data. NeuroImage, 51(2), 752-764.

Sadeghi, A. R., Visconti, I., & Wachsmann, C. (2008). User privacy in transport systems based on RFID e-tickets. PiLBA'08 Privacy in Location-Based Applications, 102-121.

Sahu, B., Dehuri, S., & Jagadev, A. (2018). A Study on the Relevance of Feature Selection Methods in Microarray

Samad, J., Loke, S. W., & Reed, K. (2013, July). Quantitative risk analysis for mobile cloud computing: a preliminary approach ve a health application case study. In 2013 12th IEEE International Conference on Trust, Security ve Privacy in Computing ve Communications, IEEE (pp. 1378-1385).

Saurav, K. (2016). Introduction to Feature Selection methods with an example (or how to select the right variables?). Analytics Vidhya.

Schiaffino, S., & Amandi, A. (2009). Intelligent user profiling. In Artificial Intelligence An International Perspective, Springer, Berlin, Heidelberg, (pp. 193-216).

Schilit, B. N., & Theimer, M. M. (1994). Disseminating active map information to mobile hosts. IEEE network, 8(5), 22-32.

Schilit, B., Adams, N., & Want, R. (1994, December). Context-aware computing applications. In 1994 First Workshop on Mobile Computing Systems ve Applications, IEEE, (pp. 85-90).

Schmidt, A. (2000). Implicit human computer interaction through context. Personal technologies, 4(2), 191-199.

Schmidt, A., Beigl, M., & Gellersen, H. W. (1999). There is more to context than location. Computers & Graphics, 23(6), 893-901.

Seeja, K. R., & Zareapoor, M. (2014). FraudMiner: A novel credit card fraud detection model based on frequent itemset mining. The Scientific World Journal.

Shaw, R. G., & Mitchell-Olds, T. (1993). ANOVA for unbalanced data: an overview. Ecology, 74(6), 1638-1645.

Shlens, J. (2014). A tutorial on principal component analysis. arXiv preprint arXiv:1404, 1100.

Shofiyah, F., & Sofro, A. (2018, November). Split and Conquer Method in Penalized Logistic Regression with Lasso (Application on Credit Scoring Data). In Journal of Physics: Conference Series (Vol. 1108, No. 1, p. 012107), IOP Publishing.

Šimundić, A. M. (2009). Measures of diagnostic accuracy: basic definitions. Ejifcc, 19(4), 203.

Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. Information processing & management, 45(4), 427-437.

Steiner, J. G., Neuman, B. C., & Schiller, J. I. (1988, February). Kerberos: An Authentication Service for Open Network Systems. In Usenix Winter, pp. 191-202.

Steyerberg, E. W., Eijkemans, M. J., & Habbema, J. D. F. (1999). Stepwise selection in small data sets: a simulation study of bias in logistic regression analysis. Journal of clinical epidemiology, 52(10), 935-942.

Strobl, C., Malley, J., & Tutz, G. (2009). An introduction to recursive partitioning: rationale, application, ve characteristics of classification ve regression trees, bagging, ve random forests. Psychological methods, 14(4), 323.

Subasi, A., & Ercelebi, E. (2005). Classification of EEG signals using neural network ve logistic regression. Computer methods ve programs in biomedicine, 78(2), 87-99.

Tanaka, Y., & Mori, Y. (1997). Principal component analysis based on a subset of variables: variable selection ve sensitivity analysis. American Journal of Mathematical ve Management Sciences, 17(1-2), 61-89.

Tang, J., Alelyani, S., & Liu, H. (2014). Feature selection for classification: A review. Data classification: Algorithms ve applications, 37.

Tang, J., Yao, L., Zhang, D., & Zhang, J. (2010). A combination approach to web user profiling. ACM Transactions on Knowledge Discovery from Data (TKDD), 5(1), 1-44.

TDK (2019), Erişim adresi: https://sozluk.gov.tr/ , Erişim tarihi: 09.01.2021

Thomas, G. D. (1997). Machine learning research: Four current directions. Artificial Intelligence, Magazine, 18(4), 97-136.

Tibshirani, R. (1996). Regression shrinkage ve selection via the lasso. Journal of the Royal Statistical Society: Series B (Methodological), 58(1), 267-288.

Tsang, S., Koh, Y. S., Dobbie, G., & Alam, S. (2014). Detecting online auction shilling frauds using supervised learning. Expert systems with applications, 41(6), 3027-3040.

Upadhayaya, A. (2012). Electronic Commerce ve E-wallet. International Journal of Recent Research ve Review, 1(1), 37-41.

Van Vlasselaer, V., Bravo, C., Caelen, O., Eliassi-Rad, T., Akoglu, L., Snoeck, M., & Baesens, B. (2015). APATE: A novel approach for automated credit card transaction fraud detection using network-based extensions. Decision Support Systems, 75, 38-48.

Van Vlasselaer, V., Eliassi-Rad, T., Akoglu, L., Snoeck, M., & Baesens, B. (2017). Gotcha! Network-based fraud detection for social security fraud. Management Science, 63(9), 3090-3110.

Vapnik, V. N. (1998). Statistical learning theory. Adaptive ve learning systems for signal processing. Communications ve Control, 2, 1-740.

Wagner, J. M., & Shimshak, D. G. (2007). Stepwise selection of variables in data envelopment analysis: Procedures ve managerial perspectives. European journal of operational research, 180(1), 57-67.

Wang, P., Ali, A., & Kelly, W. (2015, August). Data security ve threat modeling for smart city infrastructure. In 2015 International Conference on Cyber Security of Smart Cities, Industrial Control System ve Communications (SSIC), IEEE (pp. 1-6)

Wang, S., & Yao, X. (2011). Relationships between diversity of classification ensembles ve single-class performance measures. IEEE Transactions on Knowledge ve Data Engineering, 25(1), 206-219.

Wang, Y., Hahn, C., & Sutrave, K. (2016, February). Mobile payment security, threats, ve challenges. In 2016 second international conference on mobile ve secure services (MobiSecServ) IEEE, (pp. 1-5)

Wang, Y., Hahn, C., & Sutrave, K. (2016, February). Mobile payment security, threats, ve challenges. In 2016 second international conference on mobile ve secure services (MobiSecServ), IEEE, (pp. 1-5).

Wei, W., Li, J., Cao, L., Ou, Y., & Chen, J. (2013). Effective detection of sophisticated online banking fraud on extremely imbalanced data. World Wide Web, 16(4), 449-475.

Weinberger, K. Q., & Saul, L. K. (2009). Distance metric learning for large margin nearest neighbor classification. Journal of Machine Learning Research, 10(2).

Weston, J., Mukherjee, S., Chapelle, O., Pontil, M., Poggio, T., & Vapnik, V. (2001). Feature selection for SVMs. In Advances in neural information processing systems (pp. 668-674).

Whiting, D. G., Hansen, J. V., McDonald, J. B., Albrecht, C., & Albrecht, W. S. (2012). Machine learning methods for detecting patterns of management fraud. Computational Intelligence, 28(4), 505-527.

Wold, S., Esbensen, K., & Geladi, P. (1987). Principal component analysis. Chemometrics ve intelligent laboratory systems, 2(1-3), 37-52.

Yan, K., & Zhang, D. (2015). Feature selection ve analysis on correlated gas sensor data with recursive feature elimination. Sensors ve Actuators B: Chemical, 212, 353-363.

You, W., Yang, Z., & Ji, G. (2014). Feature selection for high-dimensional multi-category data using PLS-based local recursive feature elimination. Expert Systems with Applications, 41(4), 1463-1475.

Zareapoor, M., Seeja, K. R., & Alam, M. A. (2012). Analysis on credit card fraud detection techniques: based on certain design criteria. International journal of computer applications, 52(3).

Zeng, X., Chen, Y. W., Tao, C., & van Alphen, D. (2009, September). Feature selection using recursive feature elimination for handwritten digit recognition. In 2009 Fifth International Conference on Intelligent Information Hiding ve Multimedia Signal Processing, IEEE (pp. 1205-1208).

Zhang, C., & Ma, Y. (Eds.). (2012). Ensemble machine learning: methods ve applications. Springer Science & Business Media.

Zhang, J., & Zulkernine, M. (2006, June). Anomaly based network intrusion detection with unsupervised outlier detection. In 2006 IEEE International Conference on Communications, IEEE (Vol. 5, pp. 2388-2393).

Zhao, P., & Yu, B. (2006). On model selection consistency of Lasso. Journal of Machine learning research, 7(Nov), 2541-2563.

Zheng, X., & Chen, D. (2003, June). Study of mobile payments system. In EEE International Conference on E-Commerce, 2003. CEC 2003, IEEE, pp. 24-27.

Zhu, J., & Hastie, T. (2004). Classification of gene microarrays by penalized logistic regression. Biostatistics, 5(3), 427-443.

Zhu, X., & Goldberg, A. B. (2009). Introduction to semi-supervised learning. Synthesis lectures on artificial intelligence ve machine learning, 3(1), 1-130.

Zhu, X., Tao, H., Wu, Z., Cao, J., Kalish, K., & Kayne, J. (2017). Fraud prevention in online digital advertising. Springer International Publishing.

Zou, H. (2006). The adaptive lasso ve its oracle properties. Journal of the American statistical association, 101(476), 1418-1429.

Zou, H., & Hastie, T. (2005). Regularization ve variable selection via the elastic net. Journal of the royal statistical society: series B (statistical methodology), 67(2), 301-320.

Zou, H., Hastie, T., & Tibshirani, R. (2006). Sparse principal component analysis. Journal of computational ve graphical statistics, 15(2), 265-286.