T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

AĞ TRAFİĞİNDE ETKİLİ OLAN ÖZNİTELİKLERİN TESPİTİ VE YAPAY SİNİR AĞLARI İLE TRAFİKLERİN TAHMİNİ

YÜKSEK LİSANS TEZİ

Muhammed ÖZDEMİR

Bilgisayar Mühendisliği Anabilim Dalı

ŞUBAT 2024

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

AĞ TRAFİĞİNDE ETKİLİ OLAN ÖZNİTELİKLERİN TESPİTİ VE YAPAY SİNİR AĞLARI İLE TRAFİKLERİN TAHMİNİ

YÜKSEK LİSANS

Muhammed ÖZDEMİR

Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı: Dr.Öğr.Üyesi. Hüseyin ESKİ

ŞUBAT 2024

Muhammed ÖZDEMİR tarafından hazırlanan "Ağ Trafiğinde Etkili Olan Özniteliklerin Tespiti ve Yapay Sinir Ağları ile Trafiklerin İzin Tahmini" adlı tez çalışması 20.02.2024 tarihinde aşağıdaki jüri tarafından oy birliği/oy çokluğu ile Sakarya Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı'nda Bilgisayar Mühendisliği Yüksek Lisans tezi olarak kabul edilmiştir.

Tez Jürisi

Jüri Başkanı :	**Dr.Öğr.Üyesi Hüseyin ESKİ** (Danışman) Sakarya Üniversitesi	.....
Jüri Üyesi :	**Dr.Öğr.Üyesi. Muhammed KOTAN** Sakarya Üniversitesi	.....
Jüri Üyesi :	**Doç. Dr. Halit ÖZTEKİN** Sakarya Uygulamalı Bilimler Üniversitesi	.....

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Sakarya Üniversitesi Fen Bilimleri Enstitüsü Lisansüstü Eğitim-Öğretim Yönetmeliğine ve Yükseköğretim Kurumları Bilimsel Araştırma ve Yayın Etiği Yönergesine uygun olarak hazırlamış olduğum "Ağ Trafiğinde Etkili Olan Özniteliklerin Tespiti ve Yapay Sinir Ağları ile Trafiklerin İzin Tahmini" başlıklı tezin bana ait, özgün bir çalışma olduğunu; çalışmamın tüm aşamalarında yukarıda belirtilen yönetmelik ve yönergeye uygun davrandığımı, tezin içerdiği yenilik ve sonuçları başka bir yerden almadığımı, tezde kullandığım eserleri usulüne göre kaynak olarak gösterdiğimi, bu tezi başka bir bilim kuruluna akademik amaç ve unvan almak amacıyla vermediğimi ve 20.04.2016 tarihli Resmi Gazete'de yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince Sakarya Üniversitesi'nin abonesi olduğu intihal yazılım programı kullanılarak Enstitü tarafından belirlenmiş ölçütlere uygun rapor alındığını, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun ortaya çıkması halinde doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi beyan ederim.

(27/02/2024).

(imza)

Muhammed ÖZDEMİR

Aileme ve Bugünlerime Katkısı Olan Herkese,

TEŞEKKÜR

Yüksek lisans eğitimim boyunca değerli bilgi ve deneyimlerinden yararlandığım, her konuda bilgi ve desteğini almaktan çekinmediğim, araştırmanın planlanmasından yazılmasına kadar tüm aşamalarında yardımlarını esirgemeyen, teşvik eden, aynı titizlikte beni yönlendiren değerli Dr.Öğr.Üyesi. Hüseyin ESKİ ve Dr.Öğr.Üyesi. Mustafa AKPINAR hocalarıma,

Göstermiş olduğu anlayış ve desteklerinden dolayı eşim Şeyma Nur ÖZDEMİR ve oğlum Ömer ÖZDEMİR'e,

Tezin konusunu oluşturan veri seti loglarını aldığım ve çalıştığım kurum olan Türkiye Denizcilik İşletmeleri A.Ş. personeline en içten teşekkürlerimi sunarım.

Muhammed ÖZDEMİR

İÇİNDEKİLER

Sayfa

KISALTMALAR

AIC : Akaike Bilgi Kriteri
BGYS : Bilgi Güvenli Yönetim Sistemi
DHCP : Dinamik Bilgisayar Yapılandır Protokolü
DNS : Alan Adı Sistemi
DVM : Destek Vektör Makinesi
IDS : Saldırı Tespit Sistemi
IPS : Saldırı Önleme Sistemi
MSSQL : Microsoft SQL Server Veritabanı
PCA : Principal Componenet Analyzes
SAS : İstatistiksel Analiz Sistemi
SBC : Schwarz Baiyes Kriteri
SIEM : Güvenlik Bilgi ve Olay Yönetimi
SOME : Siber Olaylara Müdahale Ekibi
TDİ : Türkiye Denizcilik İşletmeleri A.Ş.
UTM : Birleşik Tehdit Yönetimi
WAF : Web Uygulama Güvenlik Duvarları
YSA : Yapay Sinir Ağı

TABLO LİSTESİ

Sayfa

ŞEKİL LİSTESİ

Sayfa

AĞ TRAFİĞİNDE ETKİLİ OLAN ÖZNİTELİKLERİN TESPİTİ VE YAPAY SİNİR AĞLARI İLE TRAFİKLERİN İZİN TAHMİNİ

ÖZET

Gelişen teknoloji ile internet üzerinde her geçen gün artarak devam eden bir veri trafiği oluşmaktadır. Artan bu veri trafiğinin kontrol edilebilmesi için bazı güvenlik sistemlerine ihtiyaç duyulmaktadır. Gerek donanım ve gerekse yazılım olarak birçok güvenlik sistemi bulunmaktadır. Bu güvenlik sistemlerinden biri ise güvenlik duvarıdır. Güvenlik Duvarı sistemleri, bütün internet trafiğini üzerinden geçiren ve ihtiyaçlara göre var olan kurallar çerçevesinde trafiğe bir sonuç veren bir sistemdir. Tez çalışmasında veri setlerinin elde edildiği Kuruluş olan TDİ'de var olan sistemde oluşan bütün internet trafiği güvenlik duvarı üzerinden yapılmaktadır. Yapılan internet trafiğine güvenlik duvarı tarafından bir sonuç verilmektedir. Her güvenlik duvarının varsayılan kuralları bulunmakla beraber sistem yöneticileri bunları ihtiyaçlara göre düzenleyebilmektedir. Güvenlik duvarları, internet trafiklerinin birçoğunu belli düzen içerisinde kategorize etmektedir. Bu durumda yerel ağdan internete çıkarken bu kategoride bulunan internet sayfası veya uygulamaların engellenmesi beklenmektedir. İnternet üzerinden yerel ağa erişim sağlamak isteyen ve zararlı web sitelerinin veya uygulamalarının erişim sağlaması istenmediğinden dolayı bunlar engellenmektedir.

Bu çalışmada, güvenlik duvarı üzerinden 5 farklı gün ve saatlerde elde edilen log kayıtları ile 5 farklı veri seti oluşturulmuştur. Elde edilen veri setleri, veri önişleme aşamasından geçtikten sonra 26 parametre ile bir sonuç üreten veri setleri haline dönüştürülmüştür. Nihai olarak elde edilen veri setlerinde var olan 26 değişkenli, hangilerinin sonuç üzerinde daha fazla etki etki ettiğinin tespiti yapıldı. Hangi log alanının sonuç üzerinde etkisini daha fazla olduğunun tespit edilmesi için Çoklu Doğrusal Regresyon ve Pricipal Componenet Analyzes (PCA) kullanılarak 6 değişkenli yeni veri setleri oluşturulmuştur. Yapay Sinir Ağları algoritması ile güvenlik duvarından elde edilen 26 değişkenli veri setleri ile Çoklu Doğrusal Regresyon ve PCA kullanılarak elde edilen yeni 6 değişkenli veri setlerinin oluşturulmasında elenen değişkenlerin ne derece doğru değişkenler olduklarının tespiti, yapılan sınıflandırmalar ile hem 26 hem de 6 değişkenli veri setlerinin accuracy, precision, recall ve f1-score, specificity değerlerine bakılarak sonuçların kıyaslanması ile yapılmıştır. Çoklu Doğrusal Regresyon ve PCA kullanılarak veri setinde çıkarılan değişkenlerin doğru değişkenler olduklarının tespiti yapılmıştır. Accuracy, f1-score, recall, precision ve specificity değerleri incelendiğinde, Yapay Sinir Ağları algoritması ile yüksek oranda, güvenlik duvarı trafiğinin sonuç tahminin yapılabileceği görülmüştür.

DETECTION OF FEATURES THAT ARE EFFECTIVE IN NETWORK TRAFFIC AND PERMISSION ESTIMATION OF TRAFFIC WITH ARTIFICIAL NEURAL NETWORKS

SUMMARY

With the evolving technology, there has been a continuous surge in data traffic over the internet. Machine learning and big data are utilizing this data for their studies among various other fields. The substantial increase in data volume not only augments the research conducted on data but also elevates the outcomes of these studies. While these aspects are burgeoning, there is a significant rise in cyber threats, jeopardizing data privacy and integrity. Hence, the necessity to safeguard data has become imperative. In the present day, cybersecurity is not merely an option but a necessity for businesses. Security firewalls, solutions, and cyber security practices stand as fundamental components of information systems. These not only provide effective protection against evolving threats but are also vital in preserving data privacy and integrity.

The escalating data traffic necessitates certain security systems for control. Employing these security systems has become unavoidable. These security solutions effectively secure systems and networks across small, medium, and large-scale operations, providing protection against various system and network attacks.

These security solutions are designed across different layers and methods to counter cyber threats. Security systems comprise numerous components, both hardware and software. Moreover, modern security systems employ technologies like artificial intelligence and behavioral analysis to detect and prevent cyber attacks. Security systems such as firewalls, antivirus programs, antispam filters, mail gateways, among others, protect against malicious software, generating alerts. Among these security systems, the firewall stands out as one of the most critical.

Firewalls primarily aim to monitor network traffic and prevent or isolate unwanted and harmful content. It acts as a barrier created to protect against unauthorized access and malicious software. Unlike security systems solely preventing malware, firewalls aim not to detect a specific malicious software but rather to protect systems by blocking suspicious traffic seen within the existing internet traffic. Security firewalls analyze internet traffic, determine permitted and blocked traffic based on defined rules, thus safeguarding systems. Firewall systems encompass security systems such as IPS (Intrusion Prevention System) and IDS (Intrusion Detection System). In older systems, IPS and IDS were separate, while modern firewall systems incorporate IPS and IDS solutions within, termed Unified Threat Management.

Firewalls operate based on specific rules. They can regulate rules for web, application, DNS, VPN, antivirus, among others, governing traffic based on these rules. Some traffic is blocked, while others are permitted. Traffic monitoring and blocking take place within the firewall. As data traffic grows and threats become more complex, the probability of traditional firewalls becoming inadequate increases.

Machine learning intervenes at this point, classifying and analyzing data, detecting abnormal activities, and identifying new threats through learning. Machine learning establishes a critical relationship between security firewalls, security solutions, and cybersecurity. Data analysis through algorithms enhances the effectiveness of these systems, enabling robust protection against increasingly sophisticated threats.

Within the organization TDİ, where the datasets are obtained, all internet traffic occurs through the security firewall. As a Public Institution, TDİ differs from standard companies in that it only experiences specific traffic occurring within public institutions. For instance, there is a dedicated internet circuit exclusive to public institutions called KamuNet, which is not accessible from the internet. Services are availed through this KamuNet circuit, such as electronic signature verification and e-government services. Traffic observed in the firewall includes unique data compared to other companies. Additionally, IPSec traffic between different locations and the center exists within log records, presenting IPSec traffic among the logs. The datasets used in the thesis comprise logs obtained from the security firewall traffic.

All internet traffic passing through the security firewall receives a result from the firewall. While every firewall has default rules, system administrators can modify them as needed. Firewalls categorize the majority of internet traffic within a certain structure. Thus, when exiting from the local network to the internet, the expectation is to block internet pages or applications within this category. Access is not desired from malicious websites or applications accessed from the internet to the local network, hence these are blocked. The blockages are not limited to web and application only; DNS, URL, and other prohibitions are also possible.

In this study, logs obtained from the security firewall on five different days and times were used to create five different datasets. The reason for obtaining datasets on different days was to encompass different users and accessed addresses. Different times were chosen to capture peak internet traffic hours, resulting in increased data traffic. The obtained datasets underwent data preprocessing, involving cleaning, organizing, and preparing the datasets for machine learning models. In data preprocessing, certain log entries were removed from the datasets to obtain their raw form. Log removals include system logs and log entries with some empty variables. Since machine learning models typically work with numerical data, textual expressions need to be converted into numerical formats. Programming processes were applied to represent textual expressions with numerical values. Data preprocessing and digitization significantly impact the performance of machine learning models, allowing for more accurate predictions. Thus, the datasets were entirely transformed into numerical form. After passing through data preprocessing and digitization steps, the datasets were transformed into datasets producing a result based on 26 variables. Examining the variables obtained from the security firewall logs, a total of 219 variables are present for each traffic. Many of these variables contain empty values. Upon scrutinizing the acquired variables, some were eliminated from the traffic, reducing them to 26 variables. Ultimately, it was determined which variables from the acquired dataset had a greater influence on the result.

To determine which variables among the obtained dataset had a greater impact on the outcome, Multiple Linear Regression and Principal Component Analysis (PCA) were utilized. Multiple Linear Regression identified the variables that had the most impact on the result. PCA was employed to analyze relationships between the 26 variables, reducing their dimensions. Variables with the most significant impact were also determined using PCA. Among the variables with high impact in both algorithms, six common variables were identified. Subsequently, new datasets with six variables were created. To ascertain the effectiveness of the selected variables in the created new datasets, Artificial Neural Networks algorithm was utilized.

The Artificial Neural Networks classification algorithm analyzed datasets obtained from the security firewall with 26 variables, as well as the new datasets with six variables derived from Multiple Linear Regression and PCA. Through this analysis, it was determined whether datasets with six variables obtained through Multiple Linear Regression and PCA were accurate variables. Comparisons were made on the results based on metrics such as accuracy, precision, recall, f1-score, and specificity for both 26 and six-variable datasets using the Artificial Neural Networks algorithm. The comparison showed that the results obtained from the six-variable datasets were mostly superior to or closely matched the outcomes of the 26-parameter datasets. Hence, it was concluded that the variables extracted from the dataset using Multiple Linear Regression and PCA were indeed accurate variables. Examination of accuracy, f1-score, recall, precision, and specificity values revealed that the Artificial Neural Networks algorithm could predict the outcome of security firewall traffic to a high extent. Consequently, it was observed that in the absence of an existing security firewall system, the system's continuity could be maintained through Machine Learning algorithms.

1. GİRİŞ

Günümüzde Bilişim Teknolojilerinin gelişmesiyle internet trafiğinde önemli bir oranda veri artışı olmuştur. İnternet kullanımı genellikle kurumsal ihtiyaçların karşılanmasını sağlamak amacıyla kullanılmaktadır. Fakat kötü niyetli kişilerce bu durum ihtiyaçların karşılanması dışında, var olan bir sitemin veya son kullanıcı tarafından oluşabilecek bir zafiyetin sömürülerek sistemin ele geçirilmesi veya şifrelenmesi gibi durumları da ortaya çıkarmaktadır. Bu tarz durumlar da siber tehditleri ortaya çıkarmıştır.

Ağ sistemleri, e-posta gönderip alma, rezervasyon yapma, haber okuma, kişisel belgeleri saklama veya paylaşma, alışveriş, eğitim gibi ağ işlemleri için kullanılan yazılım ve donanım öğeleridir [1]. Ağ trafiği üzerinde yaşanan artışlar, siber saldırı ve tehditlerin her geçen gün artarak devam etmesine neden olmaktadır. Sürekli artmakta olan siber saldırı ve tehditler, sistemlerin bir güvenlik mekanizmasıyla kontrol edilmesi ihtiyacını ortaya çıkarmıştır. Bu kontrol mekanizmaları Saldırı Tespit Sistemi (IDS) ve Saldırı Önleme Sistemlerini (IPS) kullanılması gerektirmektedir. Bu sistemler siber saldırıları ve oluşabilecek zafiyetleri engellemek için kullanılmaktadır. Güvenlik önlemleri alınırken sadece IDS, IPS gibi sistemler yeterli olmamaktadır. Bu durumda son kullanıcı için anti-virüs kullanılması veya gelen-giden e-postaların güvenliğinin sağlanması ile içerisinde zararlı bağlantı veya çalıştırılabilir programların bulunabileceği e-postalarının engellenebilmesi için anti-spam veya mail-gateway gibi güvenlik ürünleri de kullanılabilmektedir. Yine sistem de oluşabilecek anomalilerin kolayca tespit edilmesi ve alarm üretilerek sistem yöneticileri veya son kullanıcılara bilgilendirme geçebilecek Security Information and Event Management (SIEM) ürünleride kullanılabilmektedir.

TDİ'de kullanılmakta olan güvenlik duvarı üzerinden hem son kullanıcı hem sistemin bütün internet trafiğin yapılmaktadır. TDİ merkeze bağlı 2 farklı lokasyonda daha bulunmaktadır. Bu lokasyonlar ile merkez arasında güvenlik bağlantı türlerinde IPSec bağlantısı sağlanmış olup merkez ile bağlantı sağlayan lokasyonların internete çıkışı, merkezde bulunan güvenlik duvarı üzerinden

sağlanmaktadır. Kuruluş aynı zamanda Kamu Kurumlarına ait olan ve internete kapalı bir sanal ağ olan KamuNet ağına dahildir. Dışarıdan alınan servis vb. hizmetler eğer Kamu Kurumlarınca KamuNet ağı üzerinden verilmekte ise bu hizmetler KamuNet ağı üzerinde alınmaktadır. TDİ tarafından da KamuNet ağı üzerinde verilen hizmetler bulunmakla beraber eğer internet üzerinden alınan hizmetler KamuNet ağı üzerinden verilmekteyse KamuNet ağı üzerinden bu hizmet alınmaktadır.

Kurumda kullanılmakta olan güvenlik duvarı üzerinden, Kurumun bütün internet trafiği loglanmaktadır. Bu loglama işlemleri, 23 Mayıs 2007 tarih ve 26530 sayılı Resmi Gazete'de 5651 sayılı "İnternet Ortamın Yapılan Yayınların Düzenlenmesi ve Bu Yayınlar Yoluyla İşlenen Suçlarla Mücadele Edilmesi Hakkında Kanun" gereği yapılmaktadır [2]. Bu logların, 5651 sayılı Kanun gereği zaman damgasıyla imzalanarak değiştirilmesi ve inkar edilemezliği sağlanmaktadır. Bu loglar içerisinde ip adresi, mac adresi gibi zorunlu alanlar olmalıdır.

Güvenlik duvarı üzerinde Kurum tarafından belirlenen kurallar çerçevesinde internet trafiği yapılmaktadır. Bu kurallar ise varsayılan ayarlar olmasıyla beraber Kurumun ihtiyaçları doğrultusunda da oluşturulmuştur. Bunlar arasında Web Filter, Application Filter, DNS Filter gibi güvenlik profilleri oluşturulur ve kurallar arasına bunlardan da etkilensin seçenekleri seçilerek trafiğin bu güvenlik profillerinden de etkilenerek yapılması sağlanır. Örnek vermek gerekirse Kurum güvenlik kurallarında yerel ağdan internete çıkarken "Sosyal Medya", "Hacking", "İllegal", "Proxy", "Terörizm", "Pornografi" içeriği olan trafiklerin hepsi engellenir diğerlerine ise izin verilir. İnternet trafiğinin yanında Merkez ile bağlı lokasyonlar arasında yapılan IPSec trafiği, SSL VPN trafiği ve KamuNet trafiği de yine güvenlik duvarı tarafından loglanmaktadır. Bahsedilen trafiklerin hepsi güvenlik duvarı tarafından belli alanlara bölümlenerek loglanmaktadır. Herhangi bir logun yaptığı trafiğin güvenlik duvarı tarafında toplamda 219 alan şeklinde ayrıştırılmaktadır. Bu alanların birçoğu boş gelmektedir. Dolu gelen kısımlarda yapılan trafiğin türüne (web,application,vpn vb.) göre bazı alanlar dolu bazı alanlar boş gelmektedir. Logun türüne göre boş veya dolu alanlar ise sürekli değişkenlik göstermektedir. Veri setleri oluşturulurken bütün log alanları incelenmiş olup log alanlarının boş gelen kısımlarının büyük bir çoğunluğu elenmiştir. Kalan alanları veri önişleme aşamasında geçirilerek log alanlarından toplamda sonuç hariç 26 log alanı seçilmiştir. Seçilen alanlara karar verilirken,

güvenlik duvarı tarafından yapılan trafiğe kurallar çerçevesinde onay veya ret vermesi kararı üzerinde hangi alanların etkili olup olmadığına bakılarak verilmiştir.

1.1. Tezin Amacı

Bu çalışmanın amacı, Güvenlik duvarının devre dışı kaldığı bir durumda yerine geçecek alternatif bir sistemin oluşturulması için gerekli değişkenlerin belirlenmesi ve yapay sinir ağları temelli bir IPS / IDS sisteminin ağın trafiğinin ne ölçüde başarılı bir biçimde kontrol edebileceğini tespit etmektir. Bu amaca ulaşmak için, güvenlik duvarı üzerinden elde edilen loglardan oluşturulan veri setleri kullanılarak Çoklu Doğrusal Regresyon, Temel Bileşem Analizi (PCA) ile veri setlerinde bulunan log alanlarından hangilerinin sonuç üzerine daha fazla etki ettiğini tespit edilmiş, yapay sinir ağları (YSA) ile elde edilen yeni veri setleri ile eski veri setleri eğitilerek test edilmiş, sonuç üzerinde etkisi belirlenmiştir.

1.2. Tezin Kapsamı

Tez çalışmasında Bölüm 2'de Ağ güvenliğinden ve bunları oluşturan güvenlik duvarı, log kanunu gibi kavramlardan bahsedilmiştir. Bölüm 3'de ise tez çalışmasında kullanılan istatistiksel yöntemler ve kullanılan Makine Öğrenimi algoritmalarında açıklanmıştır. Bölüm 4'te verilerin nasıl hazırlandığı ve oluşturulduğu ile ilgili bilgilerden bahsedildikten sonra Bölüm 5'te kullanılan yöntemler ve algoritmalar ile elde edilen sonuçlardan bahsedilmiştir. Bölüm 6'da ise elde edilen sonuçların değerlendirmesi yapılarak sonraki çalışmalar için öneriler getirilmiştir.

1.3. Literatür Araştırması

Gelen ve giden internet trafiği, önceden belirlenmiş bir dizi kuralla otomatik bir internet güvenlik sistemi aracılığıyla kontrol edilmektedir. Makine öğrenimi algoritmaları, güvenlik duvarı üzerindeki bulunan trafiğin incelemesi için kullanılmakta ve sonuçlara dayanarak internet trafiklerini kontrol etmektedir. Al-Behadili tarafından yapılan çalışmada, güvenlik duvarı trafiği, Karar Ağacı sınıflandırma algoritması uygulanarak kontrol edilmiştir [1]. Bu çalışmada, güvenlik duvarı günlük erişimlerini sınıflandırmak için detaylı bir analiz yapılmıştır. Elde edilen sonuçlar, Karar Ağacı algoritmasının performansını kontrol etmek için

karşılaştırılmıştır. Karar Ağacı algoritmasının verimliliği, DVM, YSA, Çoklu sınıflandırıcı, makine öğrenimi algoritmalarıyla karşılaştırıldığında en başarılı sınıflandırma algoritması olduğu görülmüştür. Performans değerlendirmesi ayrıca sınıflandırma hatası, Kappa İstatistiği, F-ölçüsü ve ortalama mutlak hata dahil olmak üzere başka ortak metriklerle yapılmıştır. Sonuçlar, Karar Ağacı algoritmasının farklı türdeki güvenlik duvarı etkinlikleri için tüm performans ölçütlerinde üstün olduğunu göstermektedir.

Firewall üzerinden oluşturulmuş kuralların ne derece doğru hazırlandığı, bu kurallar arasında önemini yitiren bir kural olup olmadığı veya kurallar arasında güvenlik zafiyeti oluşturacak kuralların varlığının gözlemlenmesi bir sistem yönetici için zor olmaktadır. Makine Öğrenmesi yöntemleri kullanılarak firewall loglarının sınıflandırılmasıyla yapılacak bir analiz çalışmasının ne derece etkili olup olmayacağı [3] nolu referansta tespit edilmeye çalışılmıştır. Bu çalışmada 5.000.000 satır log, Firewall log alanlarınıda 17'si alınmış ve 6 farklı algoritma kullanılmıştır. Bu çalışmada kullanılan algoritmalar HyperPipes, Decission Table, ZeroR, Navie Bayes, IBk(-kNN) ve OneR algoritmalarıdır. Nihai olarak Correctly Classified Instances, Kappa, Root Mean Squared Error değerlerine bakıldığında birinci olarak IBk ikinci olarak ise Decission Table algoritmasının en yüksek skorda başarılı sınıflandırma yapıldığı görülmüştür. Bu durumda IBk algoritmasını verdiği F skor sonucuna göre güvenlik duvarı kurallarının Policy ID'sinin yaptığı sınıflandırmada çıkan sonuçlara göre Policy ID'lerin tekrar gözden geçirilmesi gerektiği görülmüştür. Sonuçlara göre makine öğrenmesi yöntemleri ile güvenlik duvarı üzerindeki kurallardan kaynaklanacak zafiyetlerin, sınıflandırma algoritmaları ile analiz edilebileceği görülmüştür.

Sistem güvenliği açısından kullanılması gereken yazılım veya donanımlar ile bilgi güvenliğinin nasıl sağlanabileceğinin bilinmesi ile log kayıtlarının tutulması büyük önem arz etmektedir. Log kayıtları bir sistemin en önemli parçalarından biridir. Network cihazlarının birçoğundan elde edilebilen Log kayıtlarının ise bugünün teknolojisinde birçok alanda kullanarak analizler yapılabileceği ve bu analizler sonucunda ise güvenlik önlemleri alınabileceği Akbaş tarafından bahsedilmiştir [4]. Sistemlerden toplanabilecek loglar; işletim sistemleri, DHCP gibi uygulamalar, güvenlik duvarı ağ cihazları, e-posta, veritabanları gibi birçok sistemden log kayıtları tutulabilmektedir. Logların toplanmasında zaman bilgisi çok önem arz etmekte olup

bu nedenle NTP servislerini çalışır durumda olması gerekir. Log Yönetimini yanında bir de Security Information Event Management (SIEM) yazılımlarımda önemli olduğu görülmüştür. Bunlar birbirlerini tamamlamaktadır. Bir bilgi sisteminde Log kayıtların toplanmasından bunları analizi ve alarm üretilmesine kadar olan süreçlerin nasıl olması gerektiği ve bilgi güvenliği süreçleri hakkında bilgiler verilmiştir.

Bir sistemde bulunan güvenlik duvarları kurallarının daha verimli bir şekilde oluşturulmasını amaçlanmıştır. Mohamed tarafından yapılan çalışmada, güvenlik açıklıklarının ya da bir sistemin hatalı çalışmasının nedenlerinden biri olarak güvenlik duvarını göstermektedir [5]. Güvenlik duvarın kurallarının, hatalı bir şekilde yapılandırılması, benzer kuralların kendisini tekrar etmesi, eksik kurallarını bulunması vb. gibi yapılandırmadan kaynaklı sorunlara çözüm önerisi sunulmuştur.

Güvenlik duvarında üzerinden veri seti için loglar alınmıştır. Bu loglar 1.048.576 satırdır. Loglar, Yapay Zeka ve Derin Öğrenme yöntemleri ile sınıflandırılarak analiz edilmiştir. Aljabri ve ark. tarafından yapılan çalışmada 2 farklı deney yapılmış bu deneylerin ilkinde genel olarak kullanıla güvenlik duvarı log alanlarından toplamda 11 özellik kullanılmış olup ikinci deneyde ise bunlara uygulama ve kategori alanları da eklenerek 13 özellik ile deneyler yapılmıştır [6]. Bu deneylerde logların verdiği "action" sonuçlarından "deny","allow","drop" ve "Reset-Both" kullanılmıştır. Burada Allow sonuçlu log 925.151, deny sonuçlu log sayısı 28.133, drop sonuçlu log sayısı 42.018, Rest-Both sonuçlu log sayısı ise 53.118 dir. Burada en az deny sonucu veren 28.133 log sayısı olduğu için diğerler sonuçlarda 28.133 sayısı kadar alınmıştır. Yapılan deneylerde ise KKN, NB,RF J48, ANN algoritmaları kullanılmış olup bunlar arasında en iyi sonucu ilk deneyde %99,11, ikinci deneyde ise %99,64 doğruluk oranı ile Random Forest algoritması vermiştir.

Güvenlik duvarı belirlenmiş kurallar çerçevesinde gelen ve giden ağ trafiklere "İzin ver", "Reddet" veya "Bırak/Sıfırla" gibi sonuçlar vermektedir. Al Haija ve ark., güvenlik duvarı tarafından verilen kararları makine öğrenmesi yöntemleri ile sınıflandırarak doğru sonuç üreten akıllı bir sınıflandırma modeli ortaya koymak amaçlanmıştır [7]. Shallow Neural Network (SNN) ve Decision Tree (ODT) algoritmaları kullanıldı. Çalışmada kullanılan veri seti Fırat Üniversitesinin Firewall cihazından elde edilen loglardan elde edilen verilerdir. Güvenlik duvarı tarafından sonuç üreten log alanları toplam 11 özellik bulunmaktadır. Veri seti 3 parçaya bölünmüştür. Bunlar %15 doğrulama, %15 test, %70 ise eğitim veri setidir. Reset-

both çok az sayıda olduğu için drop ile birleştirilmiş ve tek sınıfı düşürülmüştür. Sınıflandırma sonuçlarına göre Shallow Neural Network (SNN) %98,50 ve Decision Tree (ODT) ise %99,8 sonucu vermektedir.

İlhan tarafından yapılan çalışmada, web trafiği üzerinden var olan anomali hareketleri, Yapay Bağışıklık Sistemi (YBS)'ne ait Negatif Seçim algoritması kullanılarak tespit edilmesi amaçlanmıştır [8]. Veri seti olarak ise Yahoo Webscope S5 veri seti kullanılmıştır. Bu veri seti içerisinde anormal ve normal sonuç üreten veri trafiği bulunmaktadır.. Bu veri seti toplamda 4 farklı sınıf ile 367 adet zaman serisi sinyal örüntüsünden oluşmaktadır. Sinyal örüntüler 1500 veri bölgesi içermektedir. 4 sınıfta toplamda 5050000 veri noktası bulunaktadır. 7 farklı deney sonucu yapılmış olup bu deneyler de belli sinyal noktalarından için bulundukları sinyal örüntüdeki bölümler için test ve eğitim verileri kullanılarak deneyler yapılmış olup bu deneyler sonucunda; Deney 1 için %93.18, Deney 2 için %95.34, Deney 3 için %93.67, Deney 4 için %93.65, Deney 5 için %98.33, Deney 6 için %93.42, Deney 7 için %98.33 doğru sınıflandırma bulunmuştur. Ortalama bütün deneylerde bulunan doğru sınıflandırma oranı ise %94.30'dur. Yapılan deneyler sonucunda Web trafiği üzerinde anomali tespitinde YBS'nin Negatif Seçim Algoritması kullanılabileceği görülmüştür.

Şahin tarafında yapılan çalışma ile uygulama katmanı için bir güvenlik duvarı geliştirilmesi ve sınıflandırma algoritmaları ile sistemin ne kadar doğru tahmin üretebileceği görülmek istenmiştir [9]. Veri seti olarak açık kaynaklı CSIC 2010 http veri seti kullanılmıştır. Veri setinde kullanılan trafikte Post, Get ve Put istekleri bulunmaktadır. Bunları doğruluk değerleri hesaplanmış ve öznitelik tablosu ortaya çıkarılmış ve ardından sınıflandırma işlemleri yapılmıştır. Veri setinin %70 eğitim %30'u ise test seti veri seti olarak kullanılmıştır. Yapılan sınıflandırma işlemlerinin ardından %96,26 ile C4.5 karar ağacı algoritmasının en iyi sonucu verdiği görülmüştür. Diğer algoritmaların ise yakın sonuç verdiği görülmüş olup bunun nedeni olarak ise doğru seçim yapılan özniteliklerin başarısını göstermektedir. Web uygulamalarında saldırı tespitinden benzer yöntemlerin kullanılabileceği önerilmiştir [9].

Fırat Üniversitesinin Firewall cihazından alınan logların sınıflandırılması yapılmıştır. Support Vector Machine (SVM) kullanılarak sınıflandırma yapılmıştır. Ertam ve Kaya bu çalışmada, sınıflandırma çekirdek fonksiyonlar olarak liner, sigmoid, radial

basis function (RBF) ve polinom fonksiyonları kullanılmıştır [10]. Burada hangi fonksiyonun sınıflandırma üzerinde daha etkili sonuç verdiği gözlemlenmiştir. Bu sonuca varırken F1, Precision ve Recall değer sonuçlarına bakılmıştır. Toplamda 65532 log üzerinden 11 özellik seçilerek inceleme yapılmış, bu özelliklerden biri ise action'dır. Bu özellik ""deny,"drop","allow","reset-both" sonuçlarını içerir. Yapılan çalışma neticesinde en iyi recall değerinin %98,5 Sigmoid fonksiyonun kullanıldığın sınıflandırmada verdiği görülmüştür. En yüksek precesion değerini ise %67,75 ile lineer fonksiyonun kullanıldığı sınıflandırmada verdiği görülmüştür. F1 skorunun ise en iyi sonucunu %76,4 ile RBF'nin kullanıldığı sınıflandırmada verdiği görülmüştür. Polinom fonksiyonlarının kullanıldığı sınıflandırmada precesion ile recall değerleri çok düşük seviyede olduğu görülmüştür. Precesion, recall ve F1 değerlerine bakıldığında ortalama değerler dikkate alındığında RBF ile yapılan sınıflandırmaların en iyisi olduğu görülmüştür [10].

Güvenlik duvarı Loglarının analizi, ağ trafiğini izlerken dikkate alınan en önemli uygulamalardan biridir. Batool ve ark. yapılan çalışmada, Fırat Üniversitesi güvenlik duvarı cihazının log kayıtları K-En Yakın Komşu (KNN), Rastgele Orman (RF) ve Derin Sinir Ağı (DNN) sınıflandırıcıları kullanılarak analiz edilmiştir [11]. Sınıflandırıcının performansını accuary, recall, precision ve F1–Score değerleri ölçülerek bir karşılaştırma yapılmıştır. 65532 kayıt 12 öznitelik kullanılarak incelendi; burada sonuç, bu özniteliklerin bir etiketi olarak tanımlandı çünkü paketleri özelliklerine göre izin verme, engelleme, etkinliklerini engelleme ya da isteğin kendisini engellemesi olarak ele alınmıştır. Bu analizin sonucunda, uygun eyleme göre en iyi özellikleri seçen en iyi algoritmanın Rastgele Orman olduğunu göstermiştir.

Derin Sinir Ağı (DNN) kullanarak bu makine öğrenimi modeli eliştirilmiştir. UCI Makine Öğrenimi Merkezinden alınan internet güvenlik duvarından alınan veriler ile oluşturulan veri seti üzerinde 11 farklı özellik incelenmiştir. Veri setinin dengesizliği nedeniyle, örnekleme yöntemi ile genelleştirilmiş bir model oluşturulmuştur. Illmond ve Suddul tarafından yapılan çalışmada, aşırı uyumlamayı önlemek için k-katlamalı çapraz doğrulama tekniği ve düzenlileştirme gibi teknikler kullanılmıştır [12]. Test verileriyle %94,49 ve eğitim verileriyle %95,81 doğruluk elde edilen model, diğer benzer çalışmalara kıyasla daha iyi bir performans sergilemiştir. Çalışma

geliştirilerek, eğitilen modelin canlı bir sisteme entegre edilip, performansının daha detaylı değerlendirilmesi amaçlanmıştır.

Web Uygulama Güvenlik Duvarları (WAF), web uygulamalarını saldırılardan korumak için kritik bir rol oynamaktadır. İmza tabanlı yaklaşımlar, uygulama özel kuralları ile kötü amaçlı trafiği engelleyerek tehditlere karşı bir savunma mekanizması sağlamaktadır [13]. Bu yaklaşımlar, sıfırıncı gün saldırıları gibi beklenmeyen tehditlere karşı savunmasız hale gelebilmektedir. Makine öğrenimi tabanlı WAF'lar, geleneksel yöntemlere göre daha esnek ve güncel kalabilirken, aynı zamanda daha az yanlış pozitif ve negatif sonuçlar üretebilirler. Literatürde yapılan incelemeler, bu yeni yaklaşımın özellikle sıfırıncı gün saldırılarına karşı savunmada daha etkili olduğunu ortaya koymuştur. Bu bağlamda, WAF teknolojilerinde makine öğrenimi tabanlı yaklaşımların kullanılması, hem mevcut hem de beklenmeyen saldırılara karşı daha etkili bir savunma mekanizması sunabilir. Ancak, mevcut araştırmaların, web uygulamalarına yönelik saldırı tiplerine karşı bu yeni yaklaşımın ne kadar etkili olduğunu tam olarak belirlemede sınırlı olduğu da göz önünde bulundurulmalıdır.

Kurumların güvenlik politikaları, güvenlik duvarı kuralları olarak uygulanmaktadır. Bu kurallardaki herhangi bir anormallikler güvenlik açıklarına yol açabilir. Ağ büyük ve kuralların karmaşık olduğu durumlarda, manuel kontrol yetersiz kalabilir ve anormallikleri tespit etmekte zorlanılabilir. Uçar ve Özhan yaptıkları çalışmada, güvenlik duvarı kurallarındaki anormallikleri tespit etmek için makine öğrenimi ve yüksek performanslı hesaplama yöntemlerine dayalı otomatik bir model önerilmektedir [14]. Bu amaçla, güvenlik duvarı kayıtları analiz edilir ve çıkarılan özellikler Naive Bayes, kNN, Karar Tablosu ve HyperPipes gibi makine öğrenimi sınıflandırma algoritması ile analiz edilir. Performans değerlendirmesi için F-ölçütü kullanışmıştır. Deneylerde, kNN'nin en iyi performansı gösterdiği görülmüştür. Ardından, F-ölçütü dağılımına dayalı bir model öngörülmüştür. Bu model üzerinden 93 güvenlik duvarı kuralı analiz edilmiştir. Model, 6 güvenlik duvarı kuralının anormallik oluşturduğunu öngörmüştür. Makine öğrenimi yöntemleriyle büyük ölçekli log dosyalarının otomatik olarak analiz edilerek güvenlik duvarı kurallarındaki anormalliklerin tespit edilebileceğini görülmektedir.

Zaman serisi verileri üzerinde anormallikleri tespit etmek için Generative Adversarial Networks (GAN) modeli sunulmuştur. Kulyadı ve ark. tarafından

sunulan model, firewall tarafından kaydedilen bu mesajlar arasındaki zaman içindeki ilişkileri ve karmaşık yapıları öğrenerek normal davranışları anlamlandırmayı amaçlamaktadır [15]. Elde edilen bu normal davranış modeli üzerine anormallik tespiti yapabilmek için ise farklı anomali tespit teknikleri uygulanmıştır. Kosinüs benzerliği kullanılarak yeniden oluşturma hataları hesaplanmış ve bu verileri beslemek için iki farklı kodlama tekniği sunulmuştur. Yapılan deneysel sonuçlar, her iki kodlama tekniğini kullanan GAN modelinin, ARIMA ve LSTAM modellerine göre daha iyi performans gösterdiğini ortaya koymuştur [15].

Web uygulama güvenlik duvarları için web saldırılarını tespit etmek üzere Shaheed ve Kurdy tarafından makine öğrenimi teknikleri kullanılan bir model önerilmiştir [16]. Önceki araştırmalardaki eksiklikleri göz önünde bulundurarak, HTTP isteklerinden çıkarılan genel ve kapsamlı özellikler üzerinde odaklanılmıştı. Bu özellikler, dört farklı veri seti üzerinde hesaplanmış olup bunlar: CSIC 2010, HTTPParams 2015, Hibrid veri seti (CSIC 2010 ve HTTPParams), ve tehlikeye giren web sunucusunun loglarıdır. Temel ve çıkarılan özelliklerin birleşimiyle, normal ve anormal istekleri sınıflandırmak için Logistic Regression, Decision Tree ve Naive Bayes gibi algoritmaları kullanılmıştır. Önerilen model, standart veri setleriyle %99,6, gerçek web sunucusu veri setleriyle ise %98,8 sınıflandırma doğruluğu elde etmiştir. Web uygulama güvenlik duvarlarında, makine öğrenimi tabanlı yöntemlerin etkinliğini gösterilmiştir [16].

2. AĞ GÜVENLİĞİ

Bu bölümde Ağ güvenliğinde kullanılan ve çalışmanın konusu olan güvenlik duvarı ve ondan alınan log kayıtlarının, Kamu Sanal Ağı'nın ne olduğu ve bunlarla ilgili bilgilerden bahsedilmiştir.

2.1. Güvenli Duvarı

Konumlandırıldığı yerin internet trafiğini (gelen – giden) üzerinden geçiren ve varsayılan veya özelleştirilmiş kurallar çerçevesinde filtreleme yapan ve bunları inceleyen donanım veya yazılım programlarıdır [17]. Güvenlik duvarı, bir sistemin en önemli parçalarından biridir. Zararlı yazılımların yol açtığı ve sisteme birçok izinsiz veya yetkisiz erişim taleplerini, önceden belirlenen kurallar ile beraber engellemektedir.

Güvenlik duvarları bir antivirüs yazılım ile aynı şeyi ifade etmemektedir. Güvenlik duvarları bir paketin içeriğinde zararlı bir yazılım olup olmadığı kontrol etmez, trafiğin şüpheli görüldüğü durumda engellemeler yapar [18].

Modern güvenlik duvarları IPS ve IDS güvenlik çözümlerini de bütünleşik olarak içermektedir. Bunlara da Birleşik Tehdit Yönetimi (UTM) denmektedir. Bu güvenlik duvarları, içerisinde web filter, application control, vpn, ips, ids, içerik filtreleme gibi birçok özelliği bütünleşik olarak barındırmaktadırlar. Tez çalışmasında güvenlik duvarı üzerinden birçok kategoride (web, application, vpn, system) elde edilen loglar kullanılmıştır.

2.2. Log Kaydı

Bilgisayar sistemleri içerisinde kullanılan web yazılımları, ağ cihazları gibi sistem elemanlarında yaşanan her türlü aktiviteler, olay kayıtları olarak adlandırılmaktadır. Bir sisteme, yetkisiz erişim olup olmadığı veya başka anomali bir durumunun tespiti için log kayıtlarının tutulması gerekmektedir. Bu log kayıtları:

- Firewall

- Switch

- DNS

- DHCP

- Uygulama Yazılımları,

- E-Posta Sistemleri

- Veritabanı Sistemleri

gibi protokol ve uygulama bazlı sistemlerden alınmaktadır.

Bu log kayıtlarının toplanması ile elimizde önemli bir veri oluşmaktadır. Bu verilerin anlamlandırılması ile log analizleri yapılabilmekte ve bu sayede uyarı sistemleri de geliştirilebilmektedir. Log analizi ile beraber sisteme erişim sağlayan bir hareket oluşturan kullanıcıların bilgilerine ulaşılmaktadır. Kullanıcıların yapmış oldukları hareketlerin kayıtları (internet gezintisi, kullanıcı oluşturma, dosya kaydetme/silme yazıcı üzerinden tarama/çıktı yapılması) kayıt altına alınarak belirli analizler sayesinde (SIEM yazılımlar), sistemlerin kontrol edilebilmesi sağlanmaktadır [18].

2.3. 5651 Sayılı Log Kanunu

23.05.2007 tarih ve 26530 sayılı Resmi Gazete'de 5651 no.lu "İnternet Ortamında Yapılan Yayınların Düzenlenmesi Ve Bu Yayınlar Yoluyla İşlenen Suçlarla Mücadele Edilmesi Hakkında Kanun" yayımlanmıştır [2]. Kanun ile beraber log yönetim ile ilgili birçok düzenlemeler yapılmıştır. Bu düzenlemelere göre sistem loglarından oluşan dosya zaman damgasıyla damgalanarak arşivlenmesi gerekmektedir. Bu sayede verilerin bütünlüğü ve inkar edilemezliği sağlanmış olacaktır. Log arşiv dosyaları 6 aydan az 2 yıldan fazla olmayacak şekilde saklanmalıdır. Saklanacak log dosyaların içerisinde kaynak ve hedef adresler, zaman, bağlantı türü, aktarılan veri miktarı gibi alanların olması gerekmektedir [2].

2.4. Kamu Sanal Ağı (KamuNet)

03.12.2016 tarih 2016/28 sayılı "Kamu, Kurum ve Kuruluşlarının KamuNet'e Dahil Edilmesi" konulu Başbakanlık Genelgesi yayımlanmıştır. Bu genelge e-devlet uygulamalarını ve Kamu, Kurum ve Kuruluşlarının bulut üzerinden verdikleri hizmetlerde internetten bağımsız daha güvenli bir yol izlenmesi amacıyla Kamu Sanal Ağı (KamuNet) kurulmasına karar verilmiştir [19].

21.06.2017 tarih ve 30103 sayılı Resmi Gazetede ki “KamuNet Ağına Bağlanma ve KamuNet Ağının Denetimine İlişkin Usul ve Esaslar Hakkında Tebliğ“ gereğince Kamu, Kurum ve Kuruluşlarının ISO 27001 Bilgi Güvenliği Yönetim Sistemi (BGYS), SOME kurulması, düzenli zafiyet taramalarının yapılması, sunucu sistemleri için gerekli güvenlik önlemlerinin alınması vb. asgari gereksinimleri yerine getirerek KamuNet hattına dahil olmaları istenmektedir [20]. Tez çalışmasında kullanılan veri setindeki log kayıtlarında KamuNet trafikleri de bulunmaktadır. Bu KamuNet trafikleri bazı e-devlet erişimlerini, e-devlete verilen hizmetleri, elektronik imza uygulamasını kullanılmasını ifade etmektedir.

3. GÜVENLİK DUVARI ANALİZİNDE KULLANILAN YÖNTEMLER

Tez çalışmasının bu bölümünde güvenlik duvarı analizi ve tahmininde kullanılan yöntemler ve açıklanacaktır. Burada Çoklu Doğrusal Regresyon, Temel Bileşen Analizi ve Yapay Sinir Ağları teknikleri açıklanacak olup, bulgular bölümünde sonuçları gösterilecektir.

3.1. Makine Öğrenmesi

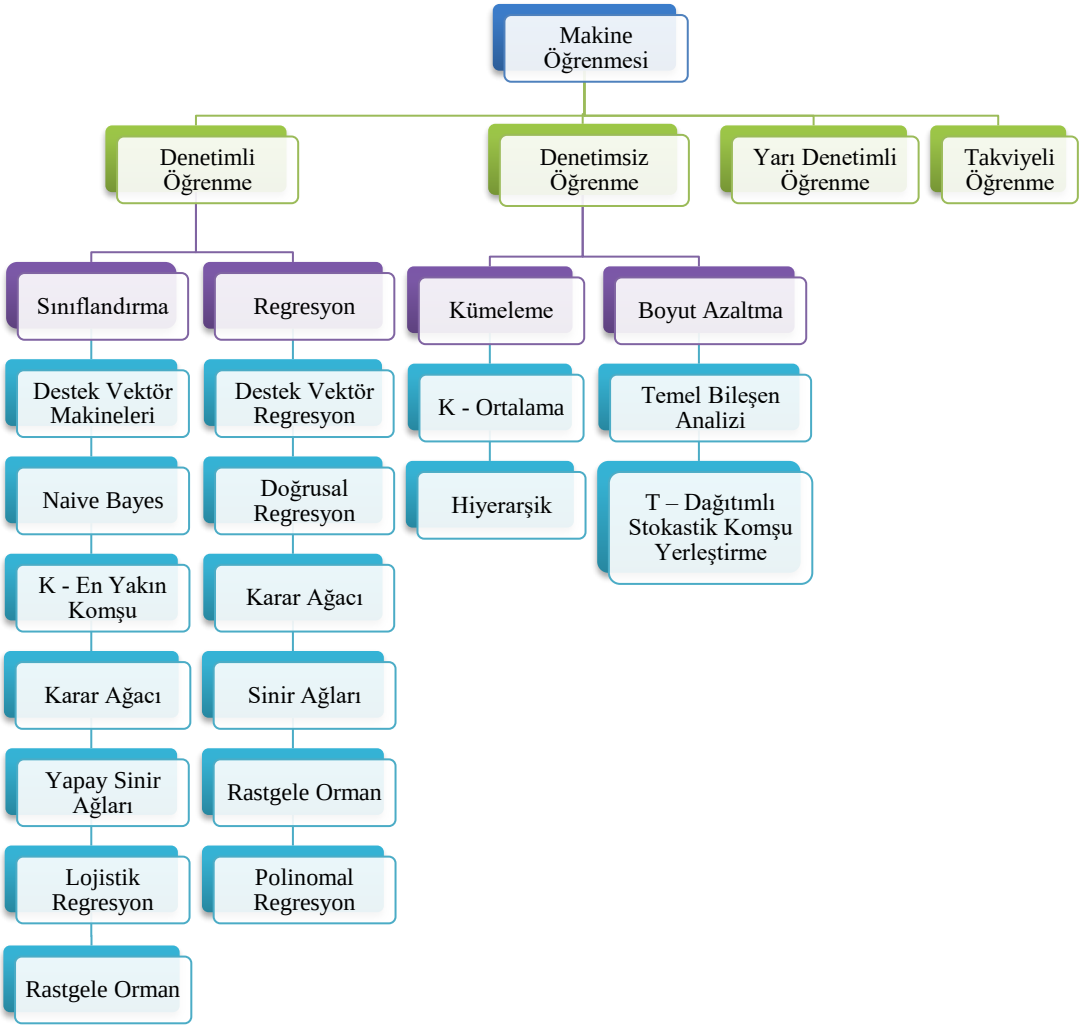
Bilgi sistemlerinin deneyimler aracılığıyla öğrenme işlevini gerçekleştirmesi ve bu öğrenmeleri kullanarak talep edilen görevi yerine getirmesi veya bir problemin daha iyi çözümlenebilmesini sağlayan bir yapay zeka dalıdır.

Dijitalleşen dünyada siber güvenlik, nesnelerin interneti (IoT), büyük veri (Big Data), bulut bilişim, yapay zeka, makine öğrenmesi kavramlarının kullanımı artış göstermeye başlamıştır [21]. Dijital dünyanın gelişimi ile beraber ortaya çıkan verilerde de önemli oranda artış olmuştur. İnsanlarda hayatta elde ettiği tecrübeler sonucunda ve almış olduğu eğitimler ile ileriye dönük bazı tahminlerde bulunabilmektedir. İnsanların tahmin de bulunduğu durumlarda genel olarak duygularıyla karar verdiği ve gerçek anlamda karar vermekte zorlanmakta, bununla beraber veri yoğunluğu olduğu durumlarda insanların kaçırdığı bazı ayrıntılar alınan kararların yanlış olmasına sebebiyet vermektedir [21]. Elde edilen veriler ile anlamlı sonuçlar elde etmek için makine öğrenmesi yöntemleri kullanılır.

Makine öğrenmesinin mucidi Arture Samuel'e göre makine öğrenmesi, bilgisayarlara açıkça programlanma yapmadan onlara öğrenme yeteneği kazandıran bir çalışma alanı olarak tanımlanmaktadır [22].

Makine öğrenimi, sınıflandırma, kümeleme, tahmin, regresyon gibi farklı uygulama alanlarında kullanılabilir (Şekil 3.1). Veri bilimi, yapay zeka istatistiksel bilimler gibi disiplinleri birleştirerek sistemlerde veya verilerde bulunan karmaşıklığı öğrenmelerini ve anlamalarını da sağlamaktadır.

Makine öğrenmesi yöntemleri 4'e ayrılmaktadır. Aşağıda Şekil 3.1'de detayları gösterilen yöntemler; denetimli öğrenme, denetimsiz öğrenme, yarı denetimli öğrenme, takviyeli öğrenme yöntemleridir.



Şekil 3.1. Makine öğrenmesi alt dalları ve yöntemleri.

Denetimli öğrenme, bir algoritmanın var olan verilerden elde edilen bilgileri etiketli veya hedefli verilerle genelleştirme yeteneğini yansıtmaktadır. Bu durumda algoritma yeni yani etiketlenmemiş verileri tahmin etmek için kullanılmaktadır [23]. Denetimli öğrenmenin temel amacı, herhangi bir modelin öncelikle verilerinin analizini yaparak, daha sonra diğer veriler için eşleşen en iyi çıkışların üretilebilmesinin sağlamaktır. Bu yaklaşım, regresyon ve sınıflandırma yöntemlerini kullanarak çıkarımlarda bulunmaya çalışmaktadır [24]. Tez çalışmasında kullanılan Çoklu Doğrusal Regresyon ve Yapay Sinir Ağları denetimli öğrenme yöntemlerinde bulunan algoritmalardır. Burada öncelikle algoritmanın eğitilebileceği ve

etiketlenmiş bir veri seti olması gerekmektedir. Sonrasında ise bir model oluşturulur ve model eğitilir, ardından ayrı bir veri seti ile doğrulanması yapılır ve elde edilen sonuçlar ile performans ölçümü yapılır.

Denetimsiz makine öğrenimi, verilerin içindeki kalıpları ve aralarındaki ilişkileri belirlemek için kullanılan bir makine öğrenimi yöntemidir. Denetimsiz makine öğrenimi algoritmaları, verilerin etiketleri ve sonuç değişkenleri olmadan, veri setlerinin yapılarını anlama ve bunları keşfetmek amaçlanır. Yaygın olarak kümeleme algoritmaları olarak bilinen denetimsiz makine öğrenmesi algoritmaları, herhangi bir verinin sınıf etiketinin bilinmesini gerektirmez [25]. Çünkü veri seti elemanlarının benzerlikleri hesaplanır ve daha sonra veri seti bu benzerliklere göre küme adı verilen çeşitli parçalara ayrılır. Denetimsiz öğrenmede yöntemleri verinin keşfini kendisi yapacağından dolayı güvenirlilik veri kalitesi ile sınırlıdır [26].

Tez çalışmasında Dogrusal Regresyon, Yapay Sinir Ağları denetimli makine öğrenimi algoritmaları ve Temel Bileşen Analizi denetimsiz makine öğrenimi algoritmaları kullanılmıştır. Bu algoritmalar ile ilgili bilgiler aşağıda verilmiştir.

Tez çalışmasında veri setlerinin, Bölüm 4'te nasıl yapıldığı detaylı bir şekilde anlatıldığı veri ön işleme aşamasında geçirilirken Çoklu Doğrusal Regresyon ve PCA algoritması da kullanılmış ve veri setlerinin alanlarından, hangi alanın daha fazla etkili olduğu tespit edilmiştir. Dogrusal Regresyon, Yapay Sinir Ağları, PCA algoritması ile ilgili bilgi aşağıda verilmiştir.

3.1.1. Çoklu doğrusal regresyon

Çoklu Doğrusal Regresyon, bir bağımlu değişken ile bir veya daha fazla bağımsız değişken arasındaki ilişkiyi modellemek için kullanılan bir istatistiksel bir yöntemdir. Çoklu Doğrusal Regresyon'un amacı analiz edilecek bağımsız değişkenler x ve bağımlı değişken y arasındaki doğrusal ilişkiyi modellemektir [27].

Çoklu Doğrusal Regresyon, her bir bağımsız değişkenin bağımlı değişkenler üzerindeki etkisini nicel olarak ölçer. Her bir değişkenin katsayısı, ilgili özelliğin bağımlı değişken üzerindeki etkisini göstermektedir. Modeller ise katsayılar ve bağımsız değişkenler arasındaki ilişkileri ifade eden denklemler kullanarak oluşturulur.

Bu tez çalışmasında Çoklu Doğrusal Regresyon için modeller oluştururken bazı metotlar ve kriterler kullanıldı. Kullanılan metotlar; İleriye Doğru Eleme (Forward), Geri Doğru Eleme (Backward), Adımsal Seçim (Stepwise) metotlarıdır.

3.1.1.1. İleriye doğru eleme

İleriye doğru seçim yönteminde değişken seçim yapılırken model üzerinde sadece sabit terimin bulunduğu denklem ile işleme başlanır ve sistemdeki değişkenler modele teker teker eklenir. Modele eklenmesi düşünülen ilk bağımsız değişken, veri seti üzerinde bulunan bağımlı değişken ile en yüksek ilişkiye sahip olan değişkendir. Modele eklenen bu bağımsız değişken, bağımlı değişkeni ile en yüksek F istatistiğine sahip değişkendir [28].

Bu model hiçbir değişken olmadan başlar, daha sonra modele dahil edilmeyen değişkenler, modelin sonucuna önemli bir katkı sağlayacak hale gelene kadar değişkenleri tek tek modele ekler. Her adımda, modelin dışında bırakılan değişkenin modele dahil edilmesi ile yeniden test edilir. En anlamlı değişken modele ilk önce eklenir. Yine kalan değişkenler arasından anlamlılık değerlerine göre modele seçilerek eklenir [29]. Bu yöntemde ilk olarak en iyi 2 bağımsız değişken seçimi yapılır ve ardından en iyi 3 bağımsız değişken seçimi yapılarak bu şekilde ilerleme yapılmaktadır.

3.1.1.2. Geriye doğru eleme

Geriye Doğru Eleme seçim metodu, İleriye Doğu Seçim metodunun tam tersidir. Bu yöntem tüm değişken seçim yöntemlerinin en basitidir. Bu yöntem, modele dahil edilecek tüm değişkenleri dikkate alan tam bir modelle başlar [30]. İlk an itibariyle modele bütün değişkenler eklenir. Sonraki aşamalarda ise her işlem sonrasında en düşük F değerine sahip olan bağımsız değişken modelden çıkarılır ve ardından işlem yapılır. Modelden çıkarılan değişkenin katkısı her seferinde tekardan test edilir. Modelden çıkarılan değişkenin sisteme katkısı istatistiki olarak önemli olarak tespit edilirse modelden çıkarılma işlemi gerçekleştirilmez ve işlem orada sonlandırılır [28].

3.1.1.3. Adımsal seçim yöntemi

Bu yöntem, her iki yönde hareket etmeye, değişkenleri farklı adımlarda eklemeye ve çıkarmaya izin veren ileri ve geri seçim yöntemlerinin bir kombinasyonudur [29]. Her adımda, bir değişken eklendikten sonra prosedür, modelde anlamlı olmayan

herhangi bir değişkeni silmek için modele önceden eklenmiş olan tüm değişkenleri kontrol eder. Süreç hem geriye doğru eleme hem de ileri seçim yaklaşımıyla başlayabilir. İleriye doğru seçim ile başlıyorsa, modele önceden eklenmiş olan bağımsız değişkenler F istatistikleriyle yeniden değerlendirmeye alınır. Süreç, modeldeki her değişken anlamlı, hariç tutulan her değişken ise anlamsız oluncaya kadar devam eder. Ancak modele girilen değişkenlerin mutlaka modelde kalmaması nedeniyle ileri seçimden farklıdır. Ancak, adım adım seçim geriye doğru elemeyle başlarsa, değişkenler istatistiksel anlamlılığa göre modelden silinir ve daha sonra anlamlı görünmeleri durumunda tekrardan eklenir [29].

Tez çalışmasında kullanılan 3 farklı metot için bir anlamlılık düzeyi belirlenmiştir. Bu anlamlılık düzeyi $p < 0.05$ olarak belirlenmiştir.

Çoklu Doğrusal Regresyon ile işlemler yapılırken yukarıda bahsedilmiş olan Modeller seçilerek kullanılmıştır. Bu modellerin her birinin kullanımında ise 3 farklı bilgi kriteri kullanılmıştır. Bunlar;

- Akaike Bilgi Kriteri (AIC)
- Schwarz Baiyes Kriteri (SBC)
- R^2 Kriteri

3.1.1.4. Akaike bilgi kriteri

AIC, istatistiksel modelin uygunluğunu değerlendirmek amacıyla kullanılan bir bilgi kriteridir. Model seçimlerinde yaygın olarak kullanılan bir kriter ölçüsüdür. AIC, hem modelin uygunluğunu hem de karmaşıklığını göz önüne alarak model seçimlerine yardımcı olur. 1973 yılında Hirotugu Akaike tarafından maksimum olabilirlik ilkesinin bir uzantısı olarak tanıtılmıştır [31].

AIC, model seçimlerinde ve karşılaştırılmasın sırasında oldukça faydalıdır. AIC, modelin karmaşıklığını ve uyumu dengelerken verilerin arasındaki uyumuda değerlendirmektedir. Daha düşük AIC değerleri, modelin uyumlu olduğunu ve karmaşıklığının daha az olduğunu göstermektedir.

3.1.1.5. Schwarz baiyes kriteri

SBC, AIC'ye benzer bir şekilde, yine model seçimlerinde kullanılan bir kriterdir. SBC'de AIC gibi modellerin karmaşıklığını dikaket almaktadır. Bununla beraber veri uyumunu da göz önünde bulundurmaktadır. SBC, çok düzeyli modelleme

çalışmalarında model seçimi için yararlı olmaktadır [32]. Dolayısıyla büyük veri setlerinde AIC'den daha etkili olmaktadır. SBC, daha düşük değerlere sahip modellerin diğerlerine göre daha iyi olduğunu göstermektedir.

3.1.1.6. R^2 kriteri

Doğrusal regresyon modellerinin uyumunu ölçen bir istatistiksel kriterdir. Bağımsız değişkenlerin bağımlı değişken üzerindeki varyansın ne kadarını açıkladığını göstermektedir [33]. R^2 modelin veriye uygunluğunu ölçmek için kullanılmaktadır.

3.1.2. Yapay sinir ağları

İnsan beyninin ortaya çıkardığı matematiksel bir modeldir [34]. YSA, ağırlıklarını ayarlayarak ve deneyimlerden öğrenerek değişen durumlara uyum sağlama yeteneğine güvenir. Yapay nöronlar, katmanlar halinde düzenlenmiştir. Bu katmanlar benzer işlevleri yerine getiren bir dizi nörondan oluşur. Giriş, çıkış ve gizli katmanlar, bu katmanların türleridir [35].

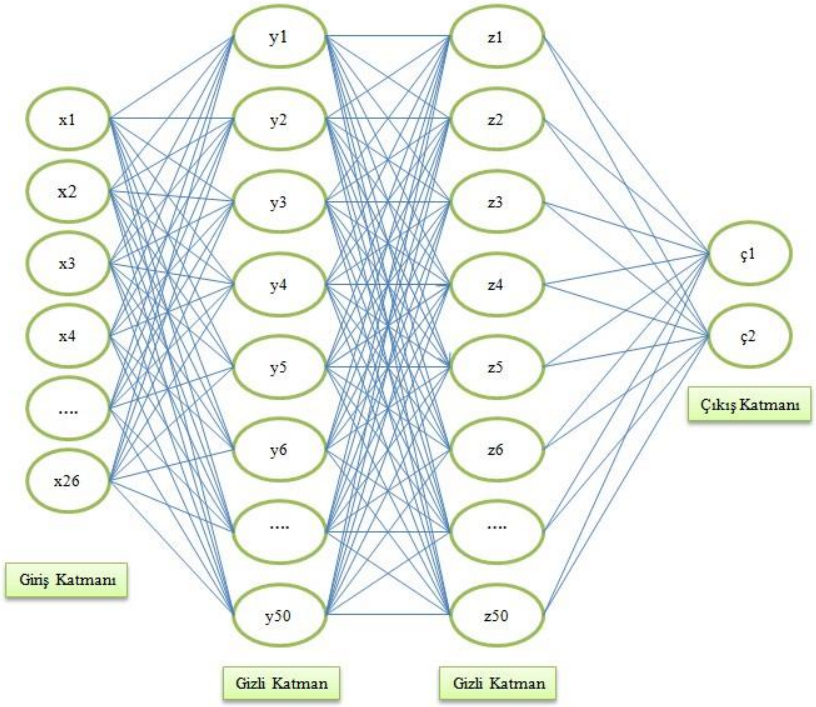
YSA ağırlıkları öğrenmek ve veri üzerinde bir genelleme yapabilecek durumda olması için öncelikle eğitilmesi gerekmektedir. Geri yayılım diye adlandırılan bu yöntem ile çıkışta bulunan hataları geriye doğru yayarak ağırlıkları günceller. Bu esnek yapı YSA'nın karmaşık sayılabilecek problemleri çözme ve öğrenme yeteneğini artırmaktadır.

Giriş Katmanı: Bağımsız değişkenleri ifade eden nöronlardan oluşmaktadır [36]. Dışarıdan alınan bilgiler, YSA'ya giriş katmanı ile aktarılır. Bu katmanda veriler işlenir, analiz edilir ve sonraki katmana aktarılır.

Gizli Katman: Bilgileri, önceki gizli katmanda ya da giriş katmanında almaktadır. Değişkenlerin doğrusallığını inceler ve doğrusal olmayanların ilişkilerin yakalanmasını sağlar [37]. YSA'da birçok gizli katman bulunabilir. Bunlar veri setinin özelliklerine göre değişim gösterebilmektedir. Bu tez çalışmasında 2 tane gizli katman bulunmaktadır. Gizli katmanlarda önceki katmandan gelen bilgileri analiz ederek sonraki katmana aktarmaktadır.

Çıkış Katman: YSA ile analiz edilen verilerin hepsinin sonucunu vermektedir. Bu katmanda bir veya daha fazla düğüme sahip olabilmektedir. Sınıflandırma problemine göre yine değişim göstermektedir. Tez çalışmasında tek çıkış katmanı

bulunmakla beraber bu katmanlar, veri setlerinde bulunan sonuç değişkeninde olduğu gibi 2 sonuç göstermektedir.



Şekil 3.2. Yapay Sinir Ağları.

3.1.3. Temel bileşenler analizi

Temel Bileşen Analizi (PCA), 1901 yılında Karl Pearson tarafından tanıtılan ve etkili bir istatistiksel yöntemdir [16]. Boyutu büyük veri setlerine sahip olmak, yüksek hesaplama maliyetlerine yol açabileceğinden dolayı çoğu zaman sorun olmaktadır. Verilerin, sistemin performansını artırmak ve orijinal verilerden mümkün olduğunca fazla bilgi tutmak amacıyla veri seti boyutunun azaltılması istenmektedir [38].

PCA, temel olarak veri setlerinin boyutunu azaltmak ve veri setlerinde bulunan değişkenlerin bir birleri arasındaki ilişkileri anlamak için yaygın olarak kullanılmakta olan denetimsiz makine öğrenme tekniğidir. PCA en basit ve en çok kullanılan boyut azaltma yöntemlerinde biridir [37].

PCA boyut azaltma, yüksek boyutlu veri setlerindeki değişken sayısını azaltmayı ve bu değişkenler arasındaki karmaşıklığı anlamayı amaçlar. Tez çalışmasında kullanılan 26 parametreli 5 farklı veri setinin, üzerinde bulunan verilerin bir birleri arasındaki ilişkileri incelenerek en etkili verilerin tespit edilmesi amacıyla PCA kullanılmıştır.

3.1.4. Kullanılan performans değerlendirme metrikleri

Makine Öğrenmesi algoritmalarının verimliliğinin ölçümleri yapılırken bazı metrikler kullanılmaktadır [36]. Tez çalışmasında, Accuracy, Precision, Recall ve F-Score, Specificity metrikleri kullanılmıştır. Kullanılan metrikler ile ilgili bilgiler aşağıda verilmiştir.

Metrikle formülize edilirken bazı parametreler kullanılmaktadır. Bunlar TP, TN, FP, FN'dir.

- TP: Gerçek pozitiflerden, pozitif olarak tahmin edilenlerin sayısını göstermektedir.
- TN: Gerçek negatiflerden, negatif olarak tahmin edilenlerin sayısını göstermektedir.
- FP: Gerçek negatiflerden, yanlış pozitif olarak tahmin edilenlerin sayısını göstermektedir.
- FN: Gerçek pozitiflerden, yanlış negatif olarak tahmin edilenlerini sayısını göstermektedir [39].

Accuracy: Bu değer doğru tahmin edilen verilerin sayısının, tüm veri setinin sayısına oranını göstermektedir [39].

$$\text{Accuary} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{3.1}$$

Precision: Gerçek pozitif olarak tahmin edilenlerin, pozitif olarak tahmin edilenlere oranını göstermektedir [39].

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{3.2}$$

Recall: Pozitif olarak tahmin edilmesi gerekenlerin, gerçekte ne kadarının pozitif olarak tahmin edildiğini göstermektedir [39].

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3.3}$$

F1-Score: Precision ve recall değerlerinin harmonik ortalamasını verir [39].

$$\mathrm{F1-Score} = 2 * \frac{\mathrm{precision} * \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}} \tag{3.4}$$

Specificity: Negatif olarak tahmin edilmesi gerekenlerin, gerçekte ne kadarının negatif olanların tahmin edildiğini göstermektedir [40].

$$\mathrm{Specificity} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} \tag{3.5}$$

3.1.5. Kullanılan yazılım ve programlama dilleri

Bu kısımda, bu tez çalışmasında kullanılan yazılımlar, programlar ve programlama dillerinde bahsedilmiştir.

5 farklı veri setinin, değişkenlerinden hangilerinin sonuç üzerinde daha fazla etkisinin olduğunu tespit edebilmek için öncelikle alınan log dosyaları csv formatıyla kaydedilmiştir. Elde edilen verilerinin sayısallaştırılması ve bazı değişkenler üzerinde işlemler yapılabilmesi için veritabanı programlama kullanılmıştır. Bu veritabanı ise Microsoft SQL Server veritabanıdır.

Veritabanında yapılan programlamanın ardından 26 değişkenli ilk veri setleri son haline getirilmiştir. Çoklu Doğrusal Regresyon ve PCA yöntemleriyle hangi değişkenin sonuç üzerinde etkisinin daha yüksek olduğu tespit edilmiştir. Bu tespitte İstatistik Analiz Sistemi (SAS) yazılımının web platformu kullanılmıştır.

Veri setlerinin analizi ile sonuca en fazla etki eden parametreler tespit edildikten sonra, YSA algoritması ile ortaya çıkan 6 değişkenli yeni veri setleri ve 26 değişkenli eski veri setlerinin sınıflandırması yapılarak ne kadar doğru oranda tahminde bulunduğu tespit edilmiştir. YSA kullanıldığı analiz, PyCharm platformunda, python programlama dili kullanılarak yazılmıştır.

4. VERİLERIN HAZIRLANMASI

4.1. Verilerin Toplanması

Bu çalışmada kullanılan veri seti bir Kamu Kuruluşu olan TDİ'den alınmıştır. TDİ Genel Müdürlüğünde Fortinet markasına ait olan Fortigate 101F güvenlik duvarı bulunmaktadır. Bu güvenlik duvarında alınan loglar syslog yönlendirmesi ile Fortilogger yazılımına aktarılmakta ve burada parse edilerek okunabilir bir hale getirilmektedir. Bu yazılım aynı zamanda yukarıda da bahsedildiği üzere 5651 sayılı Kanun'un gerektirdiği zorunluluğu sağlamakta ve bu logları KamuSM üzerinden zaman damgası ile damgalayarak bütünlüğünü ve inkar edilemezliğini sağlamaktadır [2].

Veri setinin oluşturulması aşamasında 5 farklı Log dosyası oluşturulmuştur. Bu log dosyalarının tarihi ve saat aralığı aşağıdaki Tablo 4.1'de gösterilmektedir. Bu log kayıtlarının seçiminde haftanın 5 farklı gün ve hafta kullanılmıştır. Ayrıca log kayıtlarının alındığı günlerin mesai günleri olmasına da dikkat edilmiştir. Üç farklı log kaydı öğleden önce, 2 farklı log kaydı öğleden sonra alınmıştır. Buradaki loglar birer saatlik kayıtlardır. Farklı tarih ve saatlerdeki logların seçimindeki temel sebep farklı trafiklerinde tespit edilmesidir.

Tablo 4.1. Farklı günlerde ve saat aralıklarında alınan 5 farklı log bilgileri.

Logun Adı	Tarih	Saat
Logs1	15-08-2022	11:00 – 12:00
Logs2	24-08-2022	15:00 – 16:00
Logs3	08-09-2022	10:00 – 11:00
Logs4	20-09-2022	14:00 – 15:00
Logs5	09-11-2022	11:00 – 12:00

Güvenlik duvarı üzerinden yapılan trafiğe ait loglar parse edildiğinde bir trafiğe ait 219 adet bilgi elde edilebilmektedir. Bu bilgiler trafiğin türüne göre boş ya da dolu olarak gelebilmektedir. Örneğin; "Application" trafiği ise bununla ilgi bir sonuç

aksiyonu alınmışsa burada web trafiğinde gelmesi gereken alanlar boş gelmekte ya da tam tersi bir durum olabilmektedir. Log alanların dolu ya da boş gelme durumu trafiğe göre tamamen bir değişkenlik gösterebilmektedir. Tez çalışmamızda veri seti oluşturulurken bütün log trafiklerinde boş gelen alanlar, veri setinde çıkarılmıştır ve dolu alanlar ile veri setleri oluşturulmuştur. Bu durumda veri seti ile ilk etapta, aşağıda Tablo 4.2'de açıklamaları yapılan 29 tane alan bilgisi ile oluşturulmuştur [41].

Tablo 4.2. Güvenlik duvarının loglarının alanları ve açıklamaları.

Log Alanları	Log Açılımı	Log Açıklaması
SrcIP	Source IP	Kaynağın ip adresini belirtir.
SrcPort	Source Port	Kaynağın portunu belirtir.
DstIP	Destination IP	Hedefin ip adresini belirtir.
DstPort	Destination Port	Hedefin portunu belirtir.
Appcat	Application Category	Kullanılan uygulamanın kategorisini belirtir.
SrcIntf	Source Interface	Güvenlik duvarındaki kaynağın interface'ni ifade eder.
DstIntf	Destination Interface	Güvenlik duvarındaki hedefin interface'ni ifade eder.
SubType	Sub-Type	Trafiğin alt kategorisini ifade eder.
Level	Level	Güvenlik derecesini ifade eder.
PolicyId	Policy-ID	Güvenlik duvarı kurallarının ifade eden Id değeridir.
Proto	Protocol	Kullanılan protokolü ifade eder.
AppList	Application List	Uygulama kontrol tarafından oluşturulan Güvenlik kurallarının hangisinden geçtiğini gösteren alan.
Profile	Profile	Web tabanlı olan (WebFilter,Dns,System) trafikler için güvenlik kurallarının hangisinden geçtiğini gösteren alan.

Tablo 4.2. (Devamı) Güvenlik duvarının loglarının alanları ve açıklamaları.

Log Alanları	Log Açılımı	Log Açıklaması
PolicyName	Policy Name	Yapılan trafiğin adını gösteren alan.
App	Application	Log trafiğinin hangi application ile yapıldığını ifade eder.
TrandIsp	Nat Translation	Hangi tür natlama yapıldığını gösterir.
SentByte	Sent Byte	Firewall un gönderdiği Byte miktarı
TransIP	Translation IP	Nat'lanan trafiğin Hangi IP ye natlandığını gösterir.
RcvdByte	Recieved Byte	Firewall'un aldığı Byte miktarı
AppRisk	Application Risk	Uygulama kontrolünde olan risk seviyesi
TransPort	Translation Port	Natlama işleminin yapıldığı port.
Action	Action	Trafik sonucunda güvenlik duvarının verdiği sonuç
SrcIntfRole	Source Interface Roles	Kaynak Interface'in hangi role ait olduğunu gösterir.
User	User	FSSO Agent ile bulunan ve trafiği oluşturan kullanıcının adı.
DstIntfRole	Destination Interface Roles	Hedef Interface'in hangi role ait olduğunu gösterir.
SrcCountry	Source Country	Kaynağın bulunduğu lokasyon.
PolicyType	Policy Type	Herhangi bir policy uygulanıp uygulanmadığını gösterir.
Cat	Category ID	Web trafiklerinde kategorinin id'sini belirtir.
CatDesc	Category Description	Web trafiklerinde kategorinin açıklamasını belirtir.

Güvenlik duvarından elde edilen çıktılar csv formatında her biri ayrı kaydedilmiştir. Bu log dosyalarının sayısallaştırılması ve bazı programlama işlemleri Microsoft SQL Server (MSSQL) veritabanı kullanılmıştır. Elde edilen log dosyaları MSSQL'de Logs1, Logs2, Logs3, Logs4, Logs5 adında oluşturulan tablolara kaydedilmiştir.

Veri Önişleme; Veri setleri üzerinden bulunan verilerin düzenlenmesi, değiştirilmesi, eksik veriyi tamamlama vb. işlemlerin yapıldığı aşamadır. Tatmin edici doğruluğu elde etmek için veri setlerinin ön işlemede işlenmesi gerekir [42]. Log alanları üzerinde elde edilen ham veriler de veri önişleme aşamasından geçirilerek analiz için uygun bir hale getirilmiştir.

Log dosyalarında elde edilen veriler farklı tipte olmaktadır. Bunlar hem sayısal hem de metin olabilmektedir. Örneğin "srcport" ve "dstport" sayısal iken "srcip", "dstip", "apprisk" gibi log alanları ise metinsel bir ifadedir. Sonraki aşamalarda yapılacak olan verilerin sonuç üzerindeki anlamlılık değerlerini bulmak ve daha sonra tahmin için kullanılacak olan algoritmaların daha doğru sonuç vermesi için veri setinde bulunan bütün veriler sayısal hale getirilmiştir.

Verilerin sayısallaştırılması için MSSQL veritabanı kullanıldı. Veritabanı üzerinde 5 farklı log kaydı için 5 farklı tablo oluşturulmuştur. Yine veritabanında, log alanlarının metinsel ifadelerini sayısal bir değer ile ifade edebilmek amacıyla öncelikle bazı log alanları için tablolar oluşturuldu. Bu veritabanı tabloları Actions, IPList, Levels, Appcat, AppList, AppRisk, CatDesc, Direction, InterFaceRoles, InterFace, PolicyName, PolicyType, Profiles, SrcCountry, Subtype, Trandisp, User'dır.

Veritabanına üzerine, verilerin sayısallaştırılmasını sağlamak amacıyla bir store procedure yazıldı. Yazılan store procedure'nin amacı, 5 farklı log tablosunda bulunan alanların farklı değerlerini yukarıda da belirtildiği gibi her alan için ayrı oluşturulan veritabanı tablolarına eklemektir. Veritabanı üzerinde yazılan store procedure ile bir dinamik yapı oluşturuldu. Bu çalışma için sadece 5 farklı log tablosu bulunmakta ve bu log tablosunun adı bir parametre olarak store procedure'ye gönderilmektedir. Store procedure, kendisine parametre olarak adı gönderilen log tablosunun içinde bulunan bütün log alanlarını kontrol eder ve log alanlarında bulunan farklı verileri ilgili tabloya kaydetmektedir. Aşağıda şekilde, yazılan store procedure'nin küçük bir örneği bulunmaktadır.

```
create procedure sp_Insert_Column

@TableName nvarchar(100)
as
begin
DECLARE @Sql NVARCHAR(MAX)
        --Bir tablodan Actions tablosuna farklı verileri ekleme.
        SET @Sql = 'insert into Actions(_action) (select distinct action from '+
        @TableName+' where action is not null and action not in (select _action from Actions))'
        EXECUTE sp_executesql @Sql

        --Bir tablodan IpList tablosuna farklı srcip alanlarındaki verileri ekleme.
        SET @Sql = 'insert into IpList(_ip) (select distinct srcip from '+
        @TableName+' where srcip is not null and srcip not in (select _ip from IpList))'
        EXECUTE sp_executesql @Sql

        --Bir tablodan IpList tablosuna farklı dstip alanlarındaki verileri ekleme.
        SET @Sql = 'insert into IpList(_ip) (select distinct dstip from '+
        @TableName+' where dstip is not null and dstip not in (select _ip from IpList))'
        EXECUTE sp_executesql @Sql

end
```

Şekil 4.1. Store procedure örneği.

Yukarıda Şekilde 4.1'de görüldüğü gibi tablo adı store procedure'ye parametre olarak gönderilmektedir. Öncelikle tabloda bulunan srcip, dstip, action alanları için IpList ve Actions tabloları kontrol edilir ve burada farklı olan ip adresleri ve action alanları ilgili tabloya eklenir ve eklenen her bir alan adı için bir "Id" verilmektedir. Bu işlemler bütün alanlar için yapılmıştır. Store procedure ile yapılan işlemlerin ardından sayısal olarak ifade edilemeyen her bir veri için, önceki Id değeri bir artırılarak yeni Id değeri verilmektedir. Tablolardan birini örnek vermek gerekirse log dosyalarında bulunan "level" alanlarındaki bütün farklı değerleri "Levels" tablosunda bulunmaktadır. "Levels" tablosunda bulunan değerler de Tablo 4.3.'de gösterilmektedir. Bu değerler, "Levels" tablosunda bulunan 5 farklı veri setinden elde edilen level alanlarını göstermektedir.

Tablo 4.3. Veri setlerinde bulunan levels değişkenin içerikleri.

ID	1	2	3	4	5
Level	Warning	Notice	Alert	Error	Information

4.2. Verilerin Düzenlenmesi

Güvenlik duvarından alınan loglardan elde edilen her veri seti için toplamda 29 alan bulunmaktadır. Log alanları incelendiğinde Cat ve CatDesc diye ifade edilen alanların aslında aynı şeyi ifade ettikleri görülmektedir. Eğer güvenlik duvarı üzerinden yapılan trafik bir web trafiği ise bu durumda log üzerinde cat ve catdesc alanları bir birlerini ifade edecek şekilde dolu gelmektedir. Cat, trafiğin kategorisini sayısal olarak ifade ederken catdesc ise metin olarak ifade etmektedir. Örnek vermek gerekirse bir web trafiğinin log kaydında cat alanı 49 olduğunda, catdesc "Business" içeriğini göstermektedir. Log kaydında cat alanının 49 olduğu bütün trafiklerde catdesc alanının "Business" olduğu görülmektedir. Veri setini sayısallaştırılmak istenmesi, cat ile catdesc alanlarının aynı şeyleri ifade etmesi ve bütün log trafiklerinde zaten cat alanının olması nedeniyle bütün veri setlerinden catdesc alanı çıkarılmıştır. Veri setlerinde log değişkenlerinin sayısı bir azaltılarak 28 olmuştur.

Bir diğer veri düzenleme işlemi, veri setinde bulunan "rcvdbyte" ve "sentbyte" alanlarında yapıldı. Bu log alanları da trafiğin dışarıdan içeri mi, içeriden dışarı mı olduğunu gösteren bir trafiktir. Log kayıtların bulunan "direction" alanında da "incoming", "outgoing", "inbound" ve "outbound" içerikleri bulunmaktadır. Eğer trafikte "rcvdbyte" ve "sentbyte" alanlarının değerleri dolu geliyorsa "direction" alanı boş gelmektedir. Eğer "rcvdbyte" ve "sentbyte" alanları boş geliyorsa "direction" alanlarında yukarıda belirtilen sonuçlar gelmektedir.

Log kaydının trafiğinde boş gelen "direction" alanlarını doldurmak için MSSQL veritabanı üzerinden bir fonksiyon yazılmıştır. Bu fonksiyon toplamda 3 parametre almaktadır. Bu parametreler "rcvdbyte", "sentbyte", "direction" alanlarıdır. Öncelikle bu parametrelerden "direction" kontrol edilir eğer bu parametre boş değilse "direction" ile gelen metinsel değer "incoming" ise 1, "outgoing" ise 2, "inbound" ise 3, outbound ise 4 sayısal değerine dönüştürülmektedir. Parametre değerlerinden "direction" boş ise bu durumda eğer "rcvdbyte" değeri "sentbyte" değerinden büyük ise bu trafik bir "incoming" trafiktir ve sayısal olarak 1 değerini ifade etmektedir. "sentbyte" değeri "rcvdbyte" değerinden büyükse bu trafik bir "outgoing" trafiktir ve sayısal olarak 2 değerini ifade etmektedir.

Veritabanı üzerinde yazılan fonksiyonla veri setine elde edilen bu sonuçlara göre "fn_direction" adıyla yeni bir alan eklenmiştir. Eklenen yeni alanın elde edilmesinde

etki gösteren “rcvdbyte” ve “sentbyte” alanları da veri setinden çıkartılmıştır ve bu durumda bütün veri setlerinde bulunan log alanlarının sayısı 27 olarak belirlenmiştir.

Veri setinde bulunan “action” alanı güvenlik duvarının kurallar çerçevesinde yapılan trafiğe verdiği sonucu göstermektedir. Yukarıda Şekilde 4.1’de görüntü olan store procedure örneğinde de gösterildiği gibi “action” alanları için bir “Actions” tablosu oluşturulmuş ve farklı olan bütün sonuçlar bu tabloya eklenmektedir. Veri setlerinde toplamda 34 farklı “action” bulunmaktadır. Tez çalışmasında da ise sonucu ifade eden “action” değeri 2 farklı sonucu indirgenmiştir. Güvenlik duvarını belirli trafiklere göre sonuçlar verebilmektedir.

Güvenlik duvarı tarafında trafiğe izin verildiği durumda bu trafiğin sonucu “accept”, “pass”, “passthrough”, “update”,” ip-conn”, “perf-stats”, “login”, “logout”, “update”, “dns”, “ip-conn”, “roll-log”, “load” olur ve bu durumların hepsinde veri setinde “action” sayısal olarak 1’i olarak ifade edilmektedir. Trafiğin engellendiği durumlarda ise trafiğin sonucu “block”, “deny”, “blocked”, “close”, “timeout”, “client-rst”, “server-rst” olur ve bu sonuçların hepsinde veri setinde “action” sayısal olarak 0 değerini ifade etmektedir.

5 farklı veri setinde de boş veya “NULL” gelen bütün log alanları da 0 ile doldurulmuştur.

4.2.1. Verilerin temizlenmesi

Güvenlik duvarından elde edilen veri setlerinde bazı log trafikleri veri setinde çıkarılmıştır. Nihai olarak elde edilen veri seti yine MSSQL üzerinden yapılan bir SQL sorgusu ile elde edilmiştir. Veri setinden aşağıda şartları sağlayan trafikler çıkartılmıştır. Bunlar;

- Veri setinde “action” alanının boş veya null geldiği durumlar
- Veri setinde hem “srcintf” ve hemde “dstint” içeriğinde “root” ve “unknown-0” olduğu durumlar
- Veri setinde “action” alanının “FSSO-polling-logon”, “tunnel-stats”, “FSSO-polling-logoff”, “ssl-new-con”, “ssl-exit”, “error”, “ssl-login-fail”, “install_sa”, “auth-logon”, “FSSO-logon”, “auth-logout”, “FSSO-logoff”, “ssl-alert” gibi sonuçlar içermediği durumlardır.

Tablo 4.4. Veri setleri loglarının ham ve işlenmiş hallerinin veri sayısı.

Veri Seti	Ham Halini Veri Sayısı	İşlenmiş Halini Veri Sayısı
Logs1	140.900	130.617
Logs2	169.639	161.969
Logs3	223.644	213.158
Logs4	187.413	178.223
Logs5	233.614	223.282

Elde edilen 5 farklı veri setinin ham halinin veri sayısı ve işlenmiş halinin veri sayısı yukarıdaki Tablo 4.4.'de gösterilmiştir. Burada veri önişlemden geçirildikten sonra bazı log trafikleri veri setinden çıkarılmıştır. Bunlar sistem logları gibi, ya da boş gelmesi istenmeyen bazı değişkenlerin değerlerinin boş gelmesi gibi durumlarda bu log trafiğinin veri setinden çıkarılması ile veri setlerinin boyutu küçülmüştür.

5. MODELLER VE BULGULAR

Bu bolümde ilk olarak tezin temel hedefi olan anlamlı değişkenlerin belirlenmesi gösterilecektir. İkinci aşamada ise belirlenen anlamlı değişkenler ile ağdaki normal trafiğin tahmini gerçekleştirilmiştir.

5.1. Anlamlı ve Etkin Değişkenlerin Tespiti

Güvenlik duvarında ilk aşamada elde edilen log alanlarını sayısı 29 iken yapılan veri ön işleme ile log alanlarını sayısı 27'e düşürülmüştür. Bu log alanlarının hangisinin sonuç üzerinde daha fazla etkisinin olduğunu tespit edilmesi için Bölüm 3'te anlatılan Çoklu Doğrusal Regresyon ve PCA algoritmaları ile veri setinde bulunan log alanlarından hangisinin sonuç üzerine daha fazla etki ettiği tespit edilmiştir. Burada iki algoritmanın temel farkı, trafiğin sonuç bilgisinin olup olmamasıdır. Çoklu Doğrusal Regresyon denkleminde trafiğin sonuç bilgisi verilirken, PCA'da bu bilgi verilmemektedir. Böylece iki farklı yaklaşım ile tüm veri setinin trafik izni ile ilişkisi de belirlenmiş olacaktır.

Veri setinde bulunan alanların etkilerini tespiti için SAS yazılım paketi kullanılmıştır. Hazırlanmış olan 5 farklı veri seti üzerindeki gerçekleştirilen işlemler sonucunda anlamlı ve etkin değişkenler belirlenmiştir.

Veri setleri ile analiz işlemlerinde öncelikle Çoklu Doğrusal Regresyon kullanılmıştır ve güven aralığı %95 olarak seçilmiştir. Çoklu Doğrusal Regresyon modellerinde değişkenlerin seçiminde üç farklı metot bulunmaktadır. Bunlar; İleri yönlü seçim (Forward selection), geriye doğru eleme (Backward Elimination) ve adım adım seçme (Stepwise selection) olmaktadır. Tez çalışmasında her bir veri seti için log alanlarının etki değerleri hesaplanırken bütün metotlar kullanılmıştır. Yukarıda bahsedilen değişken seçim metotlarında model AIC, SBC, R^2 istatistiksel ölçütler kullanılarak sınanmıştır. Her üç istatistik ölçütü, 5 farklı veri seti için 3 değişken seçim metodu kullanılarak sonuçlar elde edilmiştir.

Tablo 5.1. Model oluşturulurken seçilen metotlar ve kriterler.

Model	Forward	Backward	Stepwise
Kriter	SBC	AIC	R2

Yukarı Tablo 5.1'de olduğu gibi SAS ile veri setlerinin log alanlarının etki sıralamasının tespiti için seçilen her metot için, bütün AIC, SBC, R^2 ile işlemler yapılmış ve toplamda 9 farklı sonuç ortaya çıkmıştır. Yapılan işlemlerin ardından ortaya çıkan sonuçların bazılarında seçilen model ve kritere göre bazı öznitelikler (değişkenler) sonuçlardan çıkarılmıştır. Ortaya çıkan sonuçlara göre bazı değişken alanları, model üzerinde etkisinin az olması sebebiyle veri setinden çıkarılmıştır.

Veri setinde bulunan log alanlarını model üzerindeki etkisinin tespiti için sonuçlar üzerinde, t-value parametresi referans alınarak işlem yapılmıştır. Etki sıralaması yapılırken t-value değerinin mutlak değeri alınmıştır ve ortaya çıkan değere göre büyükten küçüğe doğru bir sıralama yapılmıştır. Yapılan sıralama sonucuna göre, en büyük t-value değerinin etkisinin en yüksek olduğu tespit edilmiştir. Örnek olarak Logs1 veri setine ait "Forward" model ve "R^2" kriter seçimi ile ortaya çıkan etki sıralaması sonucu aşağıdaki Tablo 5.2'de gösterilmiştir.

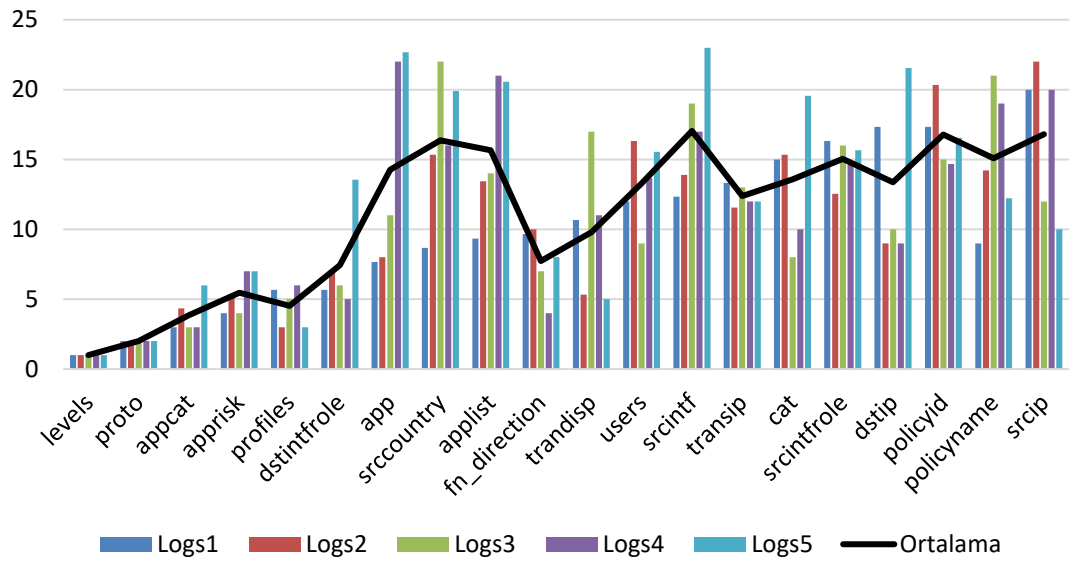
Tablo 5.2. Forward metodu ve R^2 bilgi kriterine göre etki sıraması.

Parametre	T-Value	Sıralama
Levels	177.51	1
Proto	101.19	2
Appcat	-76.7	3
Apprisk	64.58	4
dstintfrole	42.8	5
Profiles	29.14	6
App	26.71	7
srccountry	-19.91	8
fn_direction	-19.51	9

Tablo 5.2. (Devamı) Forward metodu ve R^2 bilgi kriterine göre etki sıraması

Parametre	T-Value	Sıralama
Applist	-18.57	10
Users	18.41	11
Srcintf	-17.89	12
trandisp	16.42	13
Transip	-15.47	14
cat	15.26	15
srcintfrole	-13.11	16
dstip	12.24	17
policyid	12.15	18
policyname	-8.87	19
srcip	4.89	20
srcport	3.41	21
subtype	-2.9	22
dstport	-1.68	23

Her bir veri seti için model ve kriterlere göre sıralamalar yapılmıştır ve ortaya çıkan sıralamaların 9 farklı model için ortalama değerleri alınmıştır. Yeni belirlenen ortalama değerlere göre yeni bir sıralama yapılmış bu sıralamada rank değeri en küçük olan parametre, en fazla etki gösteren değeri ifade etmektedir. Dolayısıyla parametrenin nihai olarak yani ortalama değerinin küçük olması, etkisinin daha yüksek olduğu anlamına gelmektedir. Ortaya çıkan sonuçlar aşağıdaki Şekil 5.1'de ki grafikte görülmektedir.



Şekil 5.1. Değişkenlerin veri setindeki etkisi ve ortalama değerleri.

Bütün veri setlerindeki değişkenlerin etki sıralaması Şekil 5.1'de görülmektedir. Aynı grafikte siyah çizgi bu değişkenleri ortalama rankını göstermektedir. Değişkenlerin veri setleri üzerindeki etki sıralaması, ortalama değerin altında olduğu ve sıralamada düşük seviyede olmaları, etkilerinin daha yüksek olduğunu ve tahminde kullanılabileceğini göstermektedir. Bu durumda grafikte de görüleceği üzere sıralamalı olarak "levels", "proto", "appcat", "apprisk", "profiles", "dstinnfrole", "fn_direction", "trandisp" değişkenleri daha yüksek etkiye sahiptir. Bu şekilde Çoklu Doğrusal Regresyon yöntemiyle hangi değişkenlerin sonuç üzerinde daha fazla etkisi olduğu tespit edilmiştir.

PCA değişkenlerin fazla sayıda olduğu ve indirgeme yapılmak istenen durumlarda kullanılan diğer bir yaklaşımdır. Burada bileşen adı verilen denklemler, modeldeki özniteliklerin belirlenen katsayılar ile çarpılarak elde edilir. Böylece her bileşenin değişkenleri temsil gücü farklı olmaktadır. İlk birkaç bileşenin temsil gücü genellikle değişkenlerin %75'nin oluşturmaktadır. Böylece değişkenler bileşenler yardımıyla ayrıştırılmaktadır. PCA yöntemiyle de veri setindeki değişkenlerin sonuç üzerinde etkisinin tespiti yapılmıştır.

PCA'da hedef değeri modelde yer almamaktadır. Bu durumda sonuç parametresi hariç diğer parametrelerin hepsi seçilerek analiz yapılmıştır. PCA analizi öncesinde tüm verilerin katsayılarının etkisinin eşit biçimde tespiti için veriler [0,1] aralığında

normalize edilmiştir. Toplamda 26 farklı bileşenle işlem yapılmıştır. İlk 5 bileşen için ortaya çıkan sonuçlar aşağıda Tablo 5.3'de gösterilmiştir.

Tablo 5.3. PCA analizinde parametreleri ilk 5 bileşendeki değerleri.

	Bileşen 1	Bileşen 2	Bileşen 3	Bileşen 4	Bileşen 5
Srcip	0.013137	-0.10555	-0.00474	-0.21464	0.391516
Srcport	-0.00596	0.031809	0.045008	0.103276	0.26088
Dstip	0.000151	-0.03505	0.241091	0.022773	0.161659
Dstport	0.041506	-0.01788	-0.14743	0.202663	0.308791
Srcintf	0.000535	-0.06307	0.177739	-0.25058	0.131343
Srcintfrole	0.001584	-0.08541	0.476522	-0.06939	-0.06084
Dstintf	-0.00487	-0.03088	0.278789	-0.06826	-0.06698
Dstintfrole	0.003235	0.109204	-0.40813	0.114097	-0.09677
Appcat	0.27566	0.197122	0.003896	-0.17847	-0.08691
Subtype	-0.37778	0.064988	0.069135	-0.04235	-0.0386
Levels	-0.11948	0.431319	0.052395	-0.26177	0.087164
Policyid	0.078101	-0.21846	-0.29393	-0.41983	0.04458
Applist	0.200214	0.413336	0.060717	0.142505	0.081792
Profiles	-0.30308	-0.26835	0.034772	0.177778	-0.09597
Policyname	0.337449	-0.13176	0.058768	0.278533	0.050395
fn_direction	-0.23772	0.000505	0.012809	0.261683	0.301522
App	0.07063	0.311337	0.065008	0.019105	0.22231
Apprisk	0.179627	0.301041	0.121751	0.021301	-0.24489
Trandisp	0.35697	-0.19757	-0.01764	0.115717	0.012415
Transip	0.312958	-0.20289	0.071022	0.090828	0.010833
Transport	0.342579	-0.19614	0.079114	0.113404	0.032728
Users	-0.00813	0.052364	0.033914	0.121389	-0.60511

Tablo 5.3. (Devamı) PCA analizinde parametreleri ilk 5 bileşendeki değerleri.

	Bileşen 1	Bileşen 2	Bileşen 3	Bileşen 4	Bileşen 5
Srccountry	-0.00092	0.10772	-0.4948	0.189885	-0.00285
proto	0.090683	-0.20434	-0.18698	-0.47058	-0.1003
policytype	0	0	0	0	0
cat	-0.26143	-0.25053	0.032979	0.180102	-0.08617

Tablo 5.3'de görülen ve değişkenlerin bu bileşenlerde görülen değerleri, diğer 26 bileşen için de bulunmaktadır. Etki sıralamaları her bir bileşen için ayrı yapılmıştır. Etki sıralaması hesaplanırken Tablo 5.3'de gösterilen değişkenlerin bileşen değerlerinin mutlak değeri alınır ve ardından büyükten küçüğe doğru bir sıralama yapılır ve bu bir etki sıralamasını ifade etmektedir. Bu işlem her bileşen için yapılır. Nihai sıralama bulunurken, sadece bileşen değerlerinin sıralaması ile işlem yapılmamaktadır. Her bileşenin bir ağırlığı (oranı) bulunmaktadır. Bu ağırlık aslında bileşenin etki gücüdür. Bunun örneği Tablo 5.4'de gösterilmiştir.

Tablo 5.4. Her bileşen için oran değeri.

	Özvektör	Fark	Oran	Toplam
1	6.45506007	3.16001875	0.2582	0.2582
2	3.29504132	0.27975222	0.1318	0.3900
3	3.01528911	1.29725483	0.1206	0.5106
4	1.71803427	0.46217403	0.0687	0.5793
5	1.25586024	0.09892084	0.0502	0.6296
6	1.15693940	0.13874058	0.0463	0.6758
7	1.01819882	0.04423571	0.0407	0.7166
8	0.97396311	0.04004806	0.0390	0.7555
9	0.93391506	0.03783302	0.0374	0.7929
10	0.89608204	0.10771277	0.0358	0.8287

Tablo 5.4. (Devamı) Her bileşen için oran değeri.

	Özvektör	Fark	Oran	Toplam
11	0.78836927	0.02750346	0.0315	0.8603
12	0.76086581	0.12133686	0.0304	0.8907
13	0.63952895	0.11149514	0.0256	0.9163
14	0.52803381	0.10994205	0.0211	0.9374
15	0.41809177	0.09366952	0.0167	0.9541
16	0.32442224	0.07939530	0.0130	0.9671
17	0.24502694	0.04875323	0.0098	0.9769
18	0.19627371	0.04938312	0.0079	0.9848
19	0.14689059	0.05927993	0.0059	0.9906
20	0.08761067	0.02696886	0.0035	0.9941
21	0.06064181	0.02116832	0.0024	0.9966
22	0.03947349	0.01388615	0.0016	0.9981
23	0.02558734	0.00641369	0.0010	0.9992
24	0.01917364	0.01754714	0.0008	0.9999
25	0.00162650	0.00162650	0.0001	1.0000
26	0.00000000		0.0000	1.0000

Etki sıralaması yapılırken, Denklem 5.1'de görüldüğü gibi ilgili bileşendeki her değişkenin etki sırasıyla oran değeri çarpılır ve yeni bir değer hesaplanır.

$$Etki\ Sırası\ Sonuç = Parametrenin\ Etki\ Sırası\ \times Oran \tag{5.1}$$

Her bileşende ortaya çıkan bileşen değeri ile yine oran değeri Denklem 5.2.'de görüldüğü gibi çarpılır ve yeni bir değer daha hesaplanır.

$$Bileşen\ Sonuç = Bileşen\ Değeri\ \times Oran \tag{5.2}$$

Bu matematiksel işlemler 26 bileşen içinde yapılır ve her bileşen için ortaya sonuçlar çıkartılır. Sonuçlar da 3 farklı şekilde oluşur. Her veri seti için; bütün değişkenlerin

bileşen değerine göre yapılan sıralamanın ortalaması, değişkenlerin etki sırasının oran değeri ile çarpılması sonucu ortaya çıkan "Etki Sırası Sonuç" ortalaması ve bileşen değerinin oran değeri ile çarpılması sonucunda ortaya çıkan "Bileşen Sonuç" değerlerinin ortalaması hesaplanır. Bu işlemler 5 farklı veri seti içinde yapılmıştır ve onlarında ortalaması alınarak ortaya nihai sonuç çıkarılmıştır. Ortaya çıkan sonuçlara göre "Etki Sıralaması" ve "Etki Sırası Sonuç" ortalama sonuçlarında düşük değer olan değişkenlerin etki derecesi daha yüksektir. "Bileşen Sonuç" ortalamasında ise değeri yüksek olan değişkenlerin etki derecesi daha yüksektir.

PCA ile yapılan özniteliklerin etki derecesinin tespiti için Tablo 5.5'de ortaya çıkan sonuçlar gösterilmiştir. 5 farklı veri setinde yapılan işlemlerin ardında nihai olarak ortaya 3 farklı sonuç çıkarılmıştır. Sonuç değerlerine göre koyu olarak gösterilen değişkenlerin sonuç üzerinde daha fazla etki ettiği tespit edilmiştir. Bu durumda Çoklu Doğrusal Regresyon da olduğu gibi PCA ile de belirli değişkenlerin etkisinin yüksek olduğu görülmektedir. Bunlar aşağıda Tablo 5.5'te de görüldü gibi "policyid", "apprisk", "proto", "applist","appcat", "levels", "cat", "transip", "profiles", "fn_direction", "policyname" olmak üzere toplamda 11 adet değişkenin sonuç üzerine etkisi daha fazladır.

Tablo 5.5. PCA ile sonuç üzerinde etkisi yüksek olan değişkenler.

Değişkenler	Etki Sırası Sonuç	Etkisi Sırası	Bileşen Sonuç	Seçilen Alanlar
policyid	0.4311	9.6904606	0.0067794	X
apprisk	0.4590504	10.37323	0.0067672	X
cat	0.4610958	13.569538	0.0067732	X
proto	0.435712	10.136308	0.0066034	X
levels	0.4524264	11.440616	0.0062797	X
dstport	0.5546936	13.343078	0.0053248	
applist	0.4879608	10.906154	0.00646	X
app	0.5083904	13.004308	0.0058155	
Transip	0.4927648	12.841232	0.0064391	X

Tablo 5.5. (Devamı) PCA ile sonuç üzerinde etkisi yüksek olan değişkenler.

Değişkenler	Etki Sırası Sonuç	Etkisi Sırası	Bileşen Sonuç	Seçilen Alanlar
profiles	0.4859192	12.283078	0.0067213	X
transport	0.519372	13.484308	0.006189	
appcat	0.4547896	11.238154	0.0068714	X
fn_direction	0.511256	13.632616	0.0061622	X
policyname	0.4941368	13.226462	0.0062521	X
dstintfrole	0.5426352	11.757538	0.0055161	
subtype	0.5680048	14.592616	0.0054029	
srcport	0.5793088	13.987384	0.0049525	
srcintfrole	0.5592808	13.067078	0.0052021	
srcintf	0.5772432	13.103076	0.0054246	
trandisp	0.5813	15.468616	0.005768	
dstip	0.5299512	14.380922	0.0054656	
srcip	0.606132	15.168308	0.0049115	
Dstintf	0.5682144	12.404	0.0052339	
Srccountry	0.5733848	12.263692	0.0052386	
Users	0.5457688	15.085232	0.005213	
Policytype	0.8238272	25.511384	0	

Tabloda 5.5'te de görüldüğü gibi koyu olarak belirtilen değişkenlerin etkisi daha yüksek olmaktadır. Çoklu Doğrusal Regresyon ile yapılan analizde etki eden değişkenler "levels", "proto", "appcat", "apprisk", "profiles", "dstinnfrole", "fn_direction", "trandisp" olarak belirlendiği yukarıda gösterilmiştir. PCA ile yapılan analizlerde de etki eden değişkenler de "levels", "proto", "appcat", "apprisk", "profiles", "fn_direction", "cat", "transip", "applist", "policyid", "policyname" olarak belirlenmiştir. Bu durumda iki farklı analizde ortaya çıkan değişkenlerde ortak değişkenler alınarak sonuç kısmında kullanılan veri setleri oluşturulmuştur.

Tablo 5.6. Çoklu Doğrusal Regresyon ve PCA için etki eden ortak değişkenler.

Çoklu Doğrusal Regresyon		PCA
levels	⟷	levels
proto	⟷	proto
appcat	⟷	appcat
apprisk	⟷	apprisk
profiles	⟷	profiles
fn_direction	⟷	fn_direction
dstinnfrole		cat
trandisp		transip
		applist
		policyid
		policyname

Tablo 5.6'da ok işaretleri ile gösterilen log alanları hem Çoklu Doğrusal Regresyon'da hem de PCA'da sonuç üzerinde etkisi fazla olan değişkenlerdir. Bunlar; "levels", "proto", "appcat", "apprisk", "profiles", "fn_direction" değişkenleridir.

Verilerin düzenlemesi bölümünde güvenlik duvarında oluşan trafiklerden elde edilen log kayıtlarının incelenmesi ve düzenlenmesinin akabinde sonuç ile beraber toplamda 27 log alanı ile 5 veri seti düzenlenmiştir. Bu log alanlarından hangisinin sonuç üzerinde etkisinin daha fazla olduğunu tespit edebilmek için Çoklu Doğrusal Regresyon ve PCA algoritmaları kullanılmıştır. Kullanılan algoritmalar ile sonuç üzerine en fazla etki eden 6 farklı log alanı tespit edilmiştir. Her iki algoritmada da ortak olan değişkenlerin olması, bu değişkenlerin etkisinin güçlü olduğunu göstermektedir. Bu log alanları ile de yeni veri setleri oluşturulmuştur.

5.2. Güvenlik Duvarı Trafiğinin Tahmini

Bu bölümde yukarıda belirtildiği gibi hazırlanan 5 farklı veri seti bulunmaktadır. Her bir veri seti 26 parametre ve 6 parametre olmak üzere 2 farklı veri seti olarak hazırlanmıştır. Hazırlanan veri setlerinde %70 oranında eğitim verisi, %30 oranında

ise test veri bulunmaktadır. Bölüm 3'de açıklanan Yapay Sinir Ağları kullanılarak güvenlik duvarı trafiğinin tahmini de yapılmıştır. Yapılan işlemlerde öncelikle sonuç dahil 27 değişkenin olduğu veri setleri kullanılmıştır. Daha sonra ise sonuç dahil 7 log alanının olduğu veri setleri kullanılmıştır.

5 farklı veri setinin Yapay Sinir Ağları algoritması ile yapılmış analizi ve bu analiz ile elde edilen sonuç değerleri Tablo 5.7'de görülmektedir. Yapay Sinir Ağları algoritması ile analizler yapılırken Bölüm 3'te anlatılan bazı katmanlar kullanılmıştır. Bu katmanlar kullanırken, YSA'nın programlanması yapılırken bazı parametreler kullanılmıştır.

Giriş Katman: YSA'da giriş katmanı 5 veri setinde 6, yine aynı verisetinde ise 26 olarak kullanılmıştır.

Gizli Katman: YSA'da 2 tane gizli katman kullanılmıştır. Bu gizli katmanların sayısı belirlenirken yapılan testler sonucunda en iyi sonucu, 2 gizli katman ve bu gizli katmanın değerinin 50 olduğu durumda verdiği görülmüştür. Bu sebeple tez çalışmasında kullanılan YSA algoritmasında 2 tane gizli katman kullanılmış bu gizli katmanların değeri ise 50 kullanılmıştır.

Çıkış Katman: Bu katmanda ise toplam eski ve yeni olan 10 farklı veri setinin de sonuç olarak "0" ve "1" olmak üzere 2 farklı değer üretmektedir.

YSA ile sınıflandırması yapılan veri setlerinin, "İzin Verilen" yani "1" ve "Engellenen" yani "0" sonuçlarının doğru veya yanlış tahmin sonuçlarının özet tablosunu ifade eden Karmaşıklık Matrisi aşağıda Tablo 5.7'de gösterilmiştir.

Tablo 5.7. Karmaşıklık Matrisi.

		Gerçek	
		İzin Verilen Trafik	Engellenen Trafik
Tahmin	İzin Verilen Trafik	TP	FP
	Engellenen Trafik	FN	TN

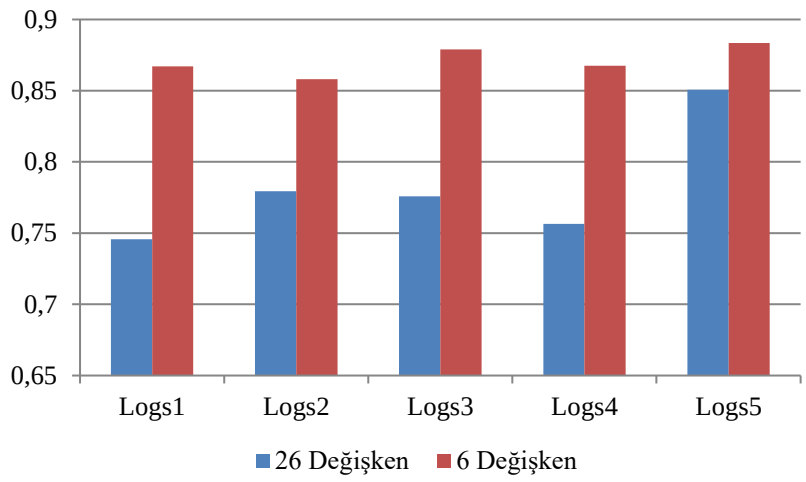
YSA eğitim sonrasında test veri setindeki sonuçlar Tablo 5.8'de verilmiştir. Logs1, Logs2, Logs3, Logs4, Logs5 veri setleri için YSA ile yapılmış olan analizde 26 değişkenli sonuçların ardından 6 değişkenli veri setlerinin eğitim sonrası test veri seti

sonuçları da Tablo 5.8'de sunulmuştur. Burada bütün veri seri üzerinde doğru tahmin oranını ifade eden doğruluk (accuracy) değeri incelendiğinde bütün veri setlerinin 6 değişkenli hallerinin 26 değişkenli durumlarına göre yüksek olduğu görülmüştür. Bu aslında veri setindeki degişkenlerin modele negatif etki ettiğini ve modelin esnekliğini azalttığını göstermektedir. Daha az sayıda değişken kullanımı ile tahmin gücünün artması, ağı etkilediği belirlenen değişkenlerin gerçekten de önemli olduğunu göstermektedir. Ayrıca sadece doğruluk değil, recall, precision, specificity ve F1-Score değerlerinde de kayda değer yükselişler görülmektedir. Tüm veri setlerinin ortalama sonuçları elde edildiğinde 26 değişkenli modellerin F1-Score, Recall, Precision, Specificity ve Accuracy değerleri sırasıyla 0.83268, 0.77534, 0.93217, 0.7954, 0.7816 olurken, 6 değiskenli modellerin değerleri sırasıyla %7.42, %9.45, %1.6, %7.79, %8.94 artarak 0.90694, 0.8699, 0.94822, 0.8733 ve 0.87106 olmuştur.

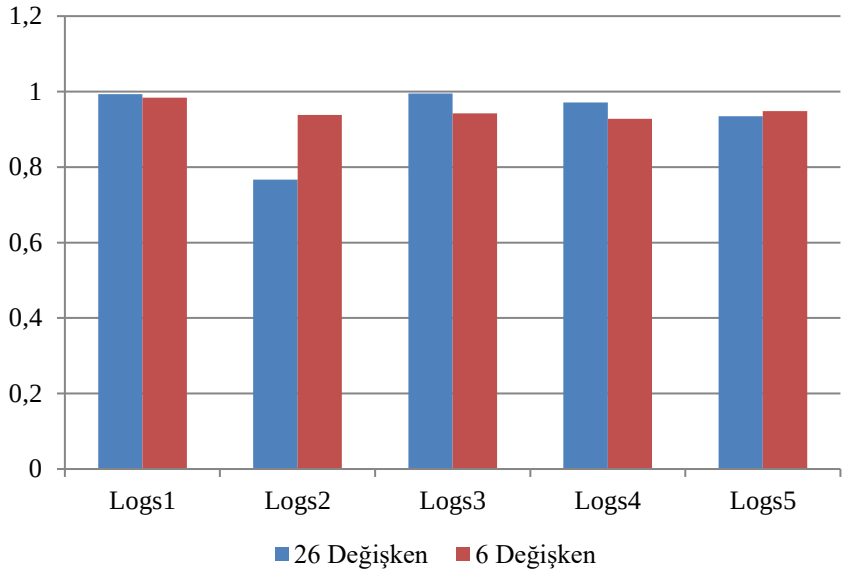
Tablo 5.8. Veri setlerinin 26 ve 6 değişkenli Yapay Sinir Ağları algoritmasına göre sonuçları.

Veri Seti	F1-Score	Recall	Precision	Specificity	Accuracy
Logs1 (26)	0.78732	0.65218	0.99311	0.98825	0.74559
Logs1 (6)	0.90020	0.82975	0.98374	0.96438	0.86717
Logs2 (26)	0.86600	0.99434	0.76701	0.23552	0.77944
Logs2 (6)	0.89652	0.85819	0.93843	0.85750	0.85799
Logs3 (26)	0.81727	0.69327	0.99528	0.99141	0.77572
Logs3 (6)	0.91383	0.88690	0.94243	0.85830	0.87899
Logs4 (26)	0.79498	0.67304	0.97088	0.95258	0.75650
Logs4 (6)	0.90303	0.87921	0.92819	0.84019	0.86756
Logs5 (26)	0.89782	0.86386	0.93456	0.80924	0.85071
Logs5 (6)	0.92112	0.89543	0.94833	0.84614	0.88357

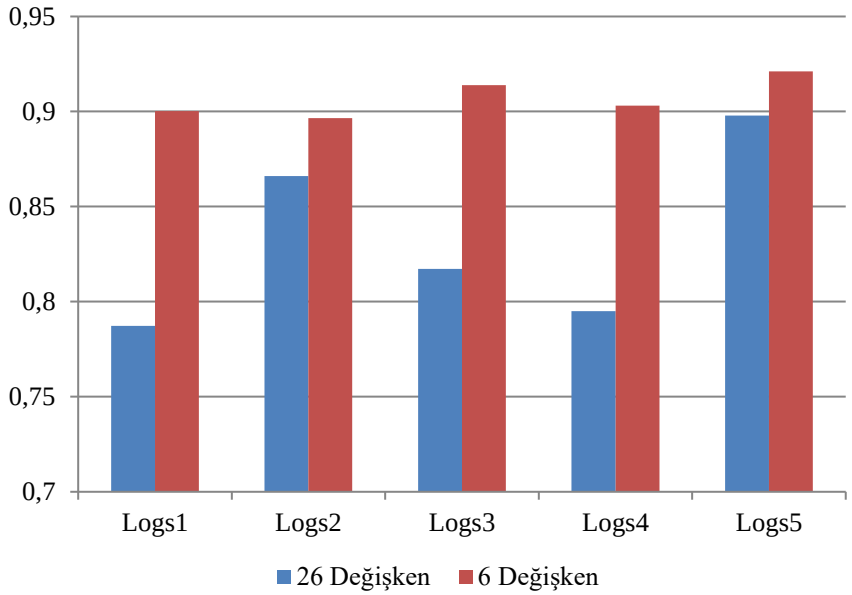
Accuracy, Precision, F1 – Score, Recall ve Specificity metriklerinin 6 değişkenli ve 26 değişkenli sonuçlarını karşılaştıran grafikler sırasıyla aşağıda Şekil 5.2, Şekil 5.3, Şekil 5.4, Şekil 5.5, Şekil 5.6'da gösterilmiştir.



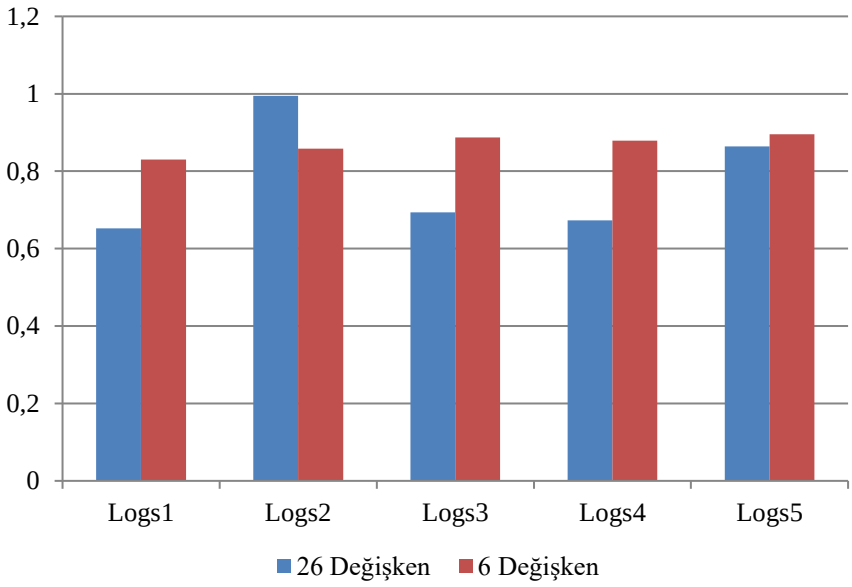
Şekil 5.2. Veri setlerinin 6 ve 26 değişkenli durumlarının accuracy sonuçları.



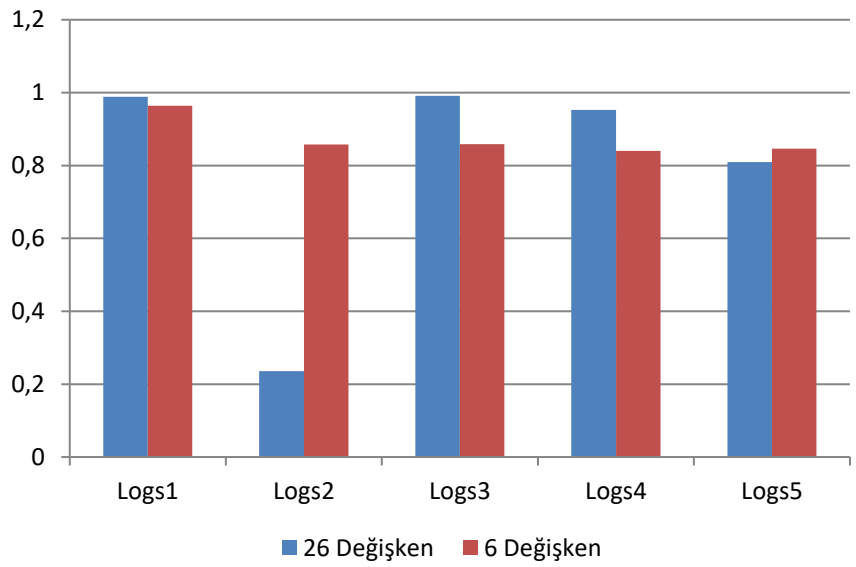
Şekil 5.3. Veri setlerinin 6 ve 26 değişkenli durumlarının precision sonuçları.



Şekil 5.4. Veri setlerinin 6 ve 26 değişkenli durumlarının F1-Score sonuçları.



Şekil 5.5. Veri setlerinin 6 ve 26 değişkenli durumlarının recall sonuçları.



Şekil 5.6. Veri setlerinin 6 ve 26 değişkenli durumlarının specificity sonuçları.

Ayrıca her veri seti ve model için TP, TN, FP, FN değerlerleri de Tablo 5.9'da verilmiştir. Bütün veri setlerinde, trafiğe izin verilen log sayısı, trafiği reddedilen log sayısından fazladır. Bu durumda Tablo 5.9'da da görüldüğü üzere TP trafiğe izin verilen log sayısını göstermektedir. Bu durumda recall değeri, trafiğine izin verilenlerin ne kadar doğru oranda tahmin edildiğini gösteren bir değerdir. Tablo 5.8'de recall değerinin Logs1, Logs3, Logs4, Logs5 veri setlerinde, 6 değişkenli veri setlerinde 26 değişkenli veri setlerine göre daha yüksek olduğu görülmüştür. Logs2, veri setinde ise 6 değişkenli veri seti 26 değişkenli veri setine göre %13.615 kadar daha az başarı göstermiştir.

Tablo 5.9. 26 ve 6 değişkenli veri setlerinin TP, TN, FP, FN sayıları.

Veri Seti	TP	TN	FP	FN
Logs1 (26)	18453	10764	128	9841
Logs1 (6)	23477	10504	388	4817
Logs2 (26)	34633	3241	10520	197
Logs2 (6)	29891	11800	1961	4939
Logs3 (26)	32073	17533	152	14190
Logs3 (6)	41031	15179	2506	5232
Logs4 (26)	25242	15206	757	12262

Tablo 5.9. (Devamı) 26 ve 6 değişkenli veri setlerinin TP, TN, FP, FN sayıları.

Veri Seti	TP	TN	FP	FN
Logs4 (6)	32974	13412	2551	4530
Logs5 (26)	43936	13049	3076	6924
Logs5 (6)	45542	13644	2481	5318

Tablo 5.9'da TN olarak gösterilen değer ise veri setlerinde engellenen trafiklerinin ne kadarının doğru tahmin edildiğini göstermektedir. Bu durumda Tablo 5.8'de görülen specificity değeri ise analizde engellenen trafiklerin ne kadarının başarılı bir şekilde tahmin edildiğin göstermektedir. Tablo 5.8 incelediğinde specificity değeri de Logs1, Logs3 ve Logs4 veri setlerinde, 6 değişkenli veri setleri 26 değişkenli veri setlerine göre daha az başarılı sonuç verirken Logs2 ve Logs5 veri setlerinde ise daha başarılı sonuç vermiştir.

Tablo 5.8'de belirtilmiş sonuçlara göre F1 – Score değerleri de 6 değişkenleri veri setlerinde 26 değişkenli verisetlerine göre daha iyi sonuçlar vermiştir. Recall değerleri incelendiğinde Logs2 veri seti dışında 6 değişkenli veri setleri 26 değişkenli veri setlerinden daha iyi sonuç vermektedir.

Bu durumda Çoklu Doğrusal Regresyon ve PCA ile yapılan etkili verilerin tespitinde, YSA ile yapılan tahmin sonuçları değerlendirildiğinde doğru değişkenlerin veri setinde elde edildiği görülmüştür.

6. TARTIŞMA VE SONUÇ

Bu tez çalışmasında TDİ'de kullanılan Fortigate marka güvenlik duvarından 5 farklı gün ve saatlerde, mesai saait içerisinde olunmasına dikkat edilerek loglar kullanılmıştır. Alınan bu loglar bazı yazılım ve programlar aracılığıyla gerekli bazı işlemlerden geçirilerek kullanılabilir hale getirilmiştir. Toplamda 26 değişken ile sonuç üreten 5 farklı veri seti elde edilmiştir.

5 farklı veri setinde bulunan 26 farklı değişkenin Çoklu Doğrusal Regresyon ve PCA yöntemlerini kullanarak, bu yöntemleri içinde de yukarıda Bölüm 3'te bahsedilmiş olan bazı metotlar ve kriterler kullanarak, sonuç üzerinde en fazla etkisi olan değişkenlerin tespit edilerek, daha az sayıda değişken ile sonuçların tahmin edilip edilemeyeceği tespit edilmek istenmiştir. Çoklu Doğrusal Regresyon ve PCA ile yapılan işlemlerin ardından Bölüm 5'te de nasıl yapıldığı anlatılmış olan yöntemler ile her iki algoritmada da tespit edilen toplamda 6 değişkenin sonuç üzerinde etkisinin yüksek olduğu tespit edilmiştir. Bu değişkenler; "levels", "proto", "appcat", "apprisk", "profiles", "fn_direction" değişkenleridir.

Çoklu Doğrusal Regresyon ve PCA ile sonuç üzerinde etkisi fazla olan değişkenler YSA Makine Öğrenmesi algoritması ile analiz edildiğinde Accuary ve F1 – Score metriklerine göre 6 parametreli veri setleri, 26 parametreli veri setlerine göre daha başarılı sonuç vermiştir. Recall, Specificity ve Precision metrikleri ise bazı veri setlerinde 6 değişkenli veri setleri bazı veri setlerinde ise 26 değişkenli veri setleri daha iyi sonuç vermektedir.

Etkin parametrelerin tespit edilmesinde kullanılan yöntemler ile başarılı bir şekilde veri setlerinin azaltıldığı ortaya çıkan sonuçlar ile görülmektedir. Elde edilen yeni veri setleri ile YSA algoritması, veri setleri ile tahmin edildiğinde accuary değeri %85-88 arasında değişim göstermekte ve precision değeri ise %92-98 arasında değişim göstermektedir. Yine F1 – Score incelendiğinde %89-92 arasında değişim gösteren sonuçlar içermektedir.

Sonuç olarak Çoklu Doğrusal Regresyon ve PCA yöntemiyle etkin verilerin başarılı bir şekilde tespit edilebileceği görülmüştür. Yine veri setlerine göre YSA algoritması

ile güvenlik duvarında oluşacak bir trafiğin, engellenen bir trafik mi yoksa izin verilecek olan bir trafik mi olduğunun kararını yüksek bir oranda doğru bir şekilde verebileceği görülmüştür.

KAYNAKLAR

[1] Al-Behadili, Khraibet, H. N. (2021). Decision tree for multiclass classification of firewall access. *Internal Journal Inteligent Engineering & Systems,* 14(3), 294-302. https://dx.doi.org/10.22266/ijies2021.0630.25

[2] T.C. Cumhurbaşkanlığı (2007, 23 Mayıs). Resmi Gazete, https://www.resmigazete.gov.tr/eskiler/2007/05/20070523-1.htm adresinden 15 Eylül 2023 tarihinde alınmıştır.

[3] Özhan, E. (2013). *Güvenlik duvarı günlüklerinin makine öğrenmesi yöntemleri ile analizi ve bir model çıkartılması* [Doktora Tezi]. Trakya Üniversitesi.

[4] Akbaş, E. (2012). Bilgi Güvenliği ve Log Yönetimi Sistemlerinin Analizi (1). https://dx.doi.org/10.13140/RG.2.2.24222.87360

[5] Mohammed, A. E. (2018). *Güvenlik duvarı kurallarına ait anomalilerin tespiti ve optimizasyonu* [Yüksek Lisans Tezi]. Sakarya Üniversitesi.

[6] Aljabri, M., Alahmadi A. A., Mohammed, R. M. A., Aboulnout, M., Alomari, D. M., Almotiri, S. H. (2022). Classification of Firewall Log Data Using Multiclass Machine Learning Models. *Electronics*, 11(12), 1851. https://doi.org/10.3390/electronics11121851

[7] Al-Haija, Q. A., Ishtaiwi, A. (2021), Machine Learning Based Model to Identify Firewall Decisions to Improve Cyber-Defense. *International Journal on Advanced Science, Engineering and Information Technology*. 11(4), 1688-1695. http://dx.doi.org/10.18517/ijaseit.11.4.14608

[8] İlhan, K. (2019). *Web trafik verilerinde yapay bağışıklık algoritmaları ile anomali tespiti* [Yüksek Lisans Tezi]. Bilecik Şeyh Edebali Üniversitesi.

[9] Şahin, M. (2017). *Uygulama katmanı için güvenlik duvarı geliştirilmesi* [Yüksek Lisans Tezi]. Gebze Teknik Üniversitesi.

[10] Ertam, F., Kaya, M. (2018). Classification of firewall log files with multiclass support vector machine. *International Symposium on Digital Forensic and Security (ISDFS)*, Antalya, Türkiye, 1-4. https://doi.org/10.1109/ISDFS.2018.8355382

[11] Al-Tarawneh, B. A., Bani-Salameh, H. (2023), Classification of firewall logs actions using machine learning techniques and deep neural network. *AIP Conference Proceedings,* Amman, Jordan, 2979(1). https://doi.org/10.1063/5.0174750

[12] Lillmond, C., Suddul, G. (2021). A deep neural network approach for analysis of firewall log data. *International Conference On Advances In Technology and Computing (ICAT-2021)*, Sri Lanka, 42-46, https://dx.doi.org/10.13140/RG.2.2.27458.04808

[13] Applebaum, S., Gaber, T., Ahmed, A. (2021). Signature-based and machine-learning-based web application firewalls: a short survey. *Procedia Computer Science*, 189(2021), 359-367. https://doi.org/10.1016/j.procs.2021.05.105

[14] Ucar, E., Ozhan, E. (2017). The analysis of firewall policy through machine learning and data mining. *Wireless Personal Communications*, 96, 2891-2909. https://doi.org/10.1007/s11277-017-4330-0

[15] Kulyadi, P. K., Mohandas, P., Kumar, S. K. S., Raman, M. J. S., Vasan, V. S. (2021). Anomaly detection using generative adversarial netwoks on firewall log message data. *2021 13th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, Pitesti, Romanya, 1-6. https://doi.org/10.1109/ECAI52376.2021.9515086

[16] Shaheed, A., Kurdy, M. H. D. B. (2021). Web application firewall using machine learning and features engineering. *Security and Communication Networs*, 2022, 1-14. https://doi.org/10.1155/2022/5280158

[17] McAfee, (2023, 15 Kasım). Güvenlik duvarı nedir. https://www.mcafee.com/tr-tr/antivirus/firewall.html

[18] Turhost, (2023, 15 Kasım). Firewall nedir. https://www.turhost.com/blog/donanimsal-firewall-nedir

[19] T.C. Cumhurbaşkanlığı (2016, 03 Aralık). Resmi Gazete, https://www.resmigazete.gov.tr/eskiler/2016/12/20161203.htm adresinden 20 Aralık 2023 tarihinde alınmıştır.

[20] T.C. Cumhurbaşkanlığı (2017, 06 Haziran). Resmi Gazete, https://www.resmigazete.gov.tr/eskiler/2017/06/20170621.htm adresinden 20 Aralık 2023 tarihinde alınmıştır.

[21] Tosunoğlu, E., Yılmaz, R., Özeren, E., Sağlam, Z. (2021). Eğitimde makine öğrenmesi: araştırmalardaki güncel eğilimler üzerine inceleme. *Ahmet Keleşoğlu Eğitim Fakültesi Dergisi (AKEF)*, 3(2), 178-199.

[22] Mahesh, B. (2019). Machine learning algorithms – a review. *International Journal of Science and Research (IJSR)*, 9 (1), 381-386. https://doi.org/10.21275/ART20203995

[23] Berry, M. W., Mohameed, A., Yap, B. E. (Eds). (2020). S*upervised and unsupervised learning for data science* (1), Springer Cham.

[24] Emre, İ. E., Taş, C., Erol, Ç. (2021). Psikiyatride makine öğrenmesi yöntemlerinin kullanımı. *Psikiyatride Güncel Yaklaşımlar*, 13(2), 332-353. https://doi.org/10.18863/pgy.779987

[25] Şenol, A., Canbay, Y., Kaya, M. (2021). Makine öğrenmesi yaklaşımlarını kullanarak salgınları erken evrede tespit etme alanındaki eğilimler. *Bilişim Teknolojiler Dergisi*, 14(4), 355-366. https://doi.org/10.17671/gazibtd.878089

[26] Özkan, M., Kayhan, Cenk. (2021). Astronomi alanında yaygın kullanılan makine öğrenmesi ve uygulamalarına örnekler. *Turkish Journal of Astronomomy and Astrophysics*, 2(1), 13-20.

[27] Maulud, D. H., Abdulazeez, A. M. (2020). A review on linear regression comprehensive in machine learning. *Journal of Applied Science and Technology Trends*, 1(4), 140-147. https://doi.org/10.38094/jastt1457

[28] Kayaalp, G. T., Güney, M. Ç., Cebeci, Z. (2015). Çoklu doğrusal modelinde değişken seçiminin zootekniye uygulanışı. *Çukurova Üniversitesi Ziraat Fakültesi Dergisi*, 30(1), 1-8.

[29] Chowdhury, M. Z. I., Truin, T. C. (2020). Veriable selection strategies and its importance in clinical prediction modelling. *Family medicine and community health*, 8(1). https://doi.org/10.1136/fmch-2019-000262

[30] Putri, D. H., Aprilla, R., Lubis, R. S. (2020). Analysis of factors affecting production rice in langkat regency with methods backward in multiple linear regression year 2018. *Journal of Mathematics and Scientific Computing With Applications*, 1(1), 23-30.

[31] Cavanaugh, J. E., Neath, A. A. (2019). The akaike information criterion: background, derivation, properties, application, interpretation, and refinements. *Wires Computational Statistics*, 11(3). https://doi.org/10.1002/wics.1460

[32] Lorah, J., Womack, A. (2019). Value of sample size for computation of the bayesian information criterion (BIC) in multilevel modeling. *Behavior Research Methods*, 51, 440-450. https://doi.org/10.3758/s13428-018-1188-3

[33] Asil, M., Alptekin, G. I. (2022). Pırlanta fiyat tahmini için regresyon modellerinin karşılaştırmalı analizi. *Niğde Ömer Halisdemir Üniversitesi Mühendislik Bilimleri Dergisi*, 11(4), 838-845. https://doi.org/10.28948/ngumuh.1088916

[34] Nasser, I. M., Al-Shawwa, M. O., Abu-Nase, S. S. (2019). A-proposed artifical neural network for predicting movies rates category. *International Journal of Academic Engineering Rersearch (IJAER)*, 3(2), 21-25.

[35] Özdemir, Ş. N., Yıldız, K. (2023). Detection of autistic spectrum disorder using artificial neural network. *Afyon Kocatepe Üniversitesi Fen ve Mühendislik Bilimleri Dergisi*, 23(4), 955-961. https://doi.org/10.35414/akufemubid.1239360

[36] Kayakuş, M., Terzioğlu, M. (2021). Yapay sinir ağları ve çoklu doğrusal regresyon kullanarak emeklilik fonu net varlık değerlerinin tahmin edilmesi. *Bilişim Teknolojileri Dergisi*, 14(1), 95-103. https://doi.org/10.17671/gazibtd.742995

[37] Ejaz, M. S., Islam, M. R., Sifatullah, M., Sarker, A. (2019). Implementation of principal component analysis on masked and non-masked face recognition. *International Conference on Advances In Science, Engineering and Robotics Technology (CASERT)*, Dhaka, Bangladesh, 1-5. https://doi.org/10.1109/ICASERT.2019.8934543

[38] Saranya, T., Sridevi, S., Deisy, C., Chung, T. C., Khan, M. K. A. A. (2020). Performance analysis of machine learning algorithms in intrusion detection system: a review. *Procedia Computer Science*, 171, 1251-1260. https://doi.org/10.1016/j.procs.2020.04.133

[39] Akhtar, A., Akhtar, S., Bakhtawar, B., Kashif, A., Aziz, N., Javeid, M. S. (2021). Covid-19 detection from CBC using machine learning techniques. *Technology, Innovation and Management (IJTIM)*, 1(2), 65-78. https://doi.org/10.54489/ijtim.v1i2.22

[40] Banaei, N., Moshfegh, J., Mohseni-Kabir, A., Houghton, J. M., Sun, Y., Kim, B. (2019). Machine learning algorithms enhance the specificity of cancer biomarker detection using SERS-based immunoassays in microfluidic chips. *RSC Advances*, 9, 1859-1868. https://doi.org/10.1039/C8RA08930B

[41] Fortinet, (2023, 16 Aralık). Log Message Fields. https://docs.fortinet.com/document/fortigate/7.2.0/fortios-log-message-reference/357866

[42] Vargas, V. W., Aranda, J. A. S., Costa, R. D. S., Pereira, P. R. S., Barbosa, J. L. V. (2023). Imbalanced data preprocessing techniques for machine learning: a systematic mapping study. *Knowledge and Information Systems*, 65(1), 31-57.

ÖZGEÇMİŞ

Ad-Soyad : Muhammed ÖZDEMİR

ÖĞRENİM DURUMU:

- **Lisans :** 2015 , Fırat Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği

MESLEKİ DENEYİM:

- 2015-2016 yılları arasında Bilim Sanayi ve Teknoloji Bakanlığı tarafından desteklenen Teknogirişim Ar-Ge projesinde çalıştı.
- 2017 yılında Türkiye Denizcilik İşletmeleri A.Ş. Genel Müdürlüğünde Programcı olarak işe başlamış olup 2022 yılına kadar bu görevi sürdürmüştür. 2022 yılından bu yana Türkiye Denizcilik İşletmeleri A.Ş. Genel Müdürlüğünde Müdür pozisyonunda çalışmaya devam etmektedir.