

**T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**E-TİCARET PLATFORMLARINA AİT OPERASYON
MERKEZİNDE MAKİNE ÖĞRENMESİ İLE BİRLİKTELİK
ANALİZİ VE ÜRÜN LOKASYONLARININ BELİRLENMESİ**

YÜKSEK LİSANS TEZİ

Melih Yüce KILIÇARSLAN

Endüstri Mühendisliği Anabilim Dalı

Mühendislik Yönetimi Bilim Dalı

AĞUSTOS 2023

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

E-TİCARET PLATFORMLARINA AİT OPERASYON
MERKEZİNDE MAKİNE ÖĞRENMESİ İLE BİRLİKTELİK
ANALİZİ VE ÜRÜN LOKASYONLARININ BELİRLENMESİ

YÜKSEK LİSANS TEZİ

Melih Yüce KILIÇARSLAN

Endüstri Mühendisliği Anabilim Dalı

Mühendislik Yönetimi Bilim Dalı

Tez Danışmanı: Prof.Dr. İbrahim ÇİL

AĞUSTOS 2023

Melih Yüce KILIÇARSLAN tarafından hazırlanan “E-TİCARET PLATFORMLARINA AİT OPERASYON MERKEZİNDE MAKİNE ÖĞRENMESİ İLE BİRLİKTELİK ANALİZİ VE ÜRÜN LOKASYONLARININ BELİRLENMESİ” adlı tez çalışması 18.08.2023 tarihinde aşağıdaki jüri tarafından oy birliği ile Sakarya Üniversitesi Fen Bilimleri Enstitüsü Endüstri Mühendisliği Anabilim Dalı Mühendislik Yönetimi Bilim Dalı’nda Yüksek Lisans tezi olarak kabul edilmiştir.

Tez Jürisi

Jüri Başkanı :

Jüri Üyesi :

Jüri Üyesi :



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 “E-TİCARET PLATFORMLARINA AİT OPERASYON MERKEZİNDE MAKİNE ÖĞRENMESİ İLE BİRLİKTELİK ANALİZİ VE ÜRÜN LOKASYONLARININ BELİRLENMESİ” 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.

(...../...../20.....).

(imza)

Melih Yüce KILIÇARSLAN



TEŞEKKÜR

Bu tez çalışmasına katkılarından ötürü Trendyol Grubu'na teşekkür ederim. Ayrıca Trendyol Ar-Ge Merkezi çalışanı Selim Hanedar'a da teşekkür ederiz. Yüksek Lisans hayatım boyunca yakın ilgisini hiç eksik etmeyen, çalışma azmimi perçinleyen danışman hocam Prof.Dr. İbrahim ÇİL'e, proje çalışmamda yardım ve katkılarıyla benden desteğini esirgemeyen aileme ve arkadaşlarıma sonsuz teşekkürlerimi bir borç bilirim.

Melih Yüce KILIÇARSLAN



İÇİNDEKİLER

Sayfa

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ	v
TEŞEKKÜR	vii
İÇİNDEKİLER	ix
KISALTMALAR	xi
SİMGELER	xiii
TABLO LİSTESİ	xv
ŞEKİL LİSTESİ	xvii
ÖZET	xix
SUMMARY	xxi
1. GİRİŞ	1
2. MAKİNE ÖĞRENİMİ, DEPOLAMA VE LİTARATÜR ARAŞTIRMASI	3
2.1. Makine Öğrenimi Tarihcesi ve Kullanım Alanları	3
2.2. Depolama	4
2.2.1. Depolama faaliyetleri	5
2.2.1.1. Mal kabul	6
2.2.1.2. Adresleme	6
2.2.1.3. Depolama	6
2.2.1.4. Sipariş toplama	7
2.2.1.5. Sevkiyat	7
2.2.2. Depolama yerleşim planı	8
2.3. Literatür Araştırması	9
2.4. E Ticaret Sektöründe Sipariş Toplama	14
3. MAKİNE ÖĞRENİMİ ALGORİTMALARI VE YAZILIMLARI	17
3.1. ML Algoritmaları	17
3.1.1. Yapay sinir ağları	17
3.1.2. Karar ağaçları	17
3.1.3. Destek vektör makinaları	18
3.1.4. Naive bayes algoritmaları	18
3.1.5. K –means kümeleme	18
3.1.6. Birliktelik kuralları ve apriori algoritması	19
3.2. ML Yazılımları	21
4. PROBLEM TANIMI	25
4.1. Problemin Tanımı	25
4.2. Verinin Tanıtılması	26
4.3. Mevcut Durum Analizi	28
5. UYGULAMA	33
5.1. Algoritmanın uygulanması	34
5.2. Algoritmanın Sonuçlarının Değerlendirilmesi	37
5.3. Önerilen Yöntem ve Uygulanması	39

6. SONUÇLAR VE ÖNERİLER.....	45
KAYNAKÇA	49
ÖZGEÇMİŞ	55



KISALTMALAR

AS/RS	: Automated Storage and Retrieval Systems
CART	: Sınıflandıma ve Regresyon Ağaçları
CRISP-DM	: Cross-industry Standard Process for Data Mining
min_sup	: Minimum Destek
MHS	: Malzeme Taşıma Sistemi
ML	: Makine Öğrenimi
PbL	: Işıklı Seçim
PbV	: Sese Göre Seçim
PP	: Picking Policy
LSCM	: Logistics and Supply Chain Management
SAS	: Depolama Tahsis Stratejisi
SPSS	: Statistical Package for the Social Sciences
ST	: Depolama Teknolojisi
SMV	: Destek Vektör Makinaları
TSP	: Travelling Salesman Problem
VM	: Veri Madenciliği
VLM	: Dikey Kaldırma Modülü
WMS	: Depo Yönetim Sistemi
YPS	: Yapay Sinir Ağları



SİMGELER

K	: Lift Değeri
m	: Metre
m²	: Metrekare
N	: Görölme Sayısı
n	: Gözlem Sayısı
P	: Destek Değeri
sn	: Saniye





TABLO LİSTESİ

Sayfa

Tablo 3.1. En Popüler ML Yazılım Araçları	22
Tablo 4.1. Müşteri Sipariş Verisi.	26
Tablo 4.2. Verinin Tanımlanması.....	27
Tablo 4.3. Sipariş Verisi Özet Tablo.	27
Tablo 4.4. Sipariş Grubu Yürüme Mesafesi.....	30
Tablo 5.1. Destek Değerleri.....	37
Tablo 5.2 Güven Değerleri.	38
Tablo 5.3. Koordinat Düzlemindeki Ortalama Konumlar	41
Tablo 5.4. Lokasyon Atamasının Gerçekleştirilmesi	43
Tablo 6.1. Önerilen Duruma Göre Sonuçlar.	46



ŞEKİL LİSTESİ

Sayfa

Şekil 2.1. Tesis Yerleşim Gösterimi.	9
Şekil 3.1. Apriori Algoritması Akış Diyagramı.	20
Şekil 4.1. En Çok Tercih Edilen Kategoriler.	27
Şekil 4.2. Depo Yerleşimi.	29
Şekil 4.3. Sipariş Grubu Toplama Rotası.	30
Şekil 5.1. Uygulama Genel Süreç Akışı.	35
Şekil 5.2. Veri Setinin Uygulamaya Yüklenmesi.	35
Şekil 5.3. Veri Setinde Grup No Hazırlaması.	35
Şekil 5.4. Veri Setinin Hazırlanması.	36
Şekil 5.5. Algoritmanın Programda Uygulanması.	36
Şekil 5.6. Ürün Kategorileri Arasındaki İlişki Grafiği.	39
Şekil 5.7. Depo Yerleşimi Konum Matrisi.	40
Şekil 5.8. Lokasyonların Konumlandırılması.	40
Şekil 5.9. Python'da Koordinat Ortalamalarının Uygulanması.	42
Şekil 5.10. Python'da Lokasyon Ataması Yapılması.	42
Şekil 6.1. Önerilen Yerleşim Düzeni.	46



E-TİCARET PLATFORMLARINA AİT OPERASYON MERKEZİNDE MAKİNE ÖĞRENMESİ İLE BİRLİKTELİK ANALİZİ VE ÜRÜN LOKASYONLARININ BELİRLENMESİ

ÖZET

Hızlı bir şekilde büyümekte olan e-ticaret sektörüne olan ilgi her geçen gün artmaktadır. Pandeminin hayatımıza girmesi e-ticaret sektörüne olan ilgiyi artırmış ve sektörün gelişimine büyük katkı sağlamıştır. Bu dönemde e ticaret sektörü küresel anlamda büyüme göstermiştir.

Pandemi, sektörün gelişimine sağladığı katkının yanında bazı zorlukları da yanında getirmiştir. Herkesin evlere kapandığı dönem ve sonrasında olmak üzere Pandemi, pek çok insanın alışveriş alışkanlığının da değişmesine yol açmıştır. Bu değişiklik sektörde faaliyet gösteren e ticaret platformların depolama operasyonlarında çeşitli zorluklar yaşamasına sebep olmuştur. Bu yüzden de sektörde faaliyet gösteren firmalar müşteri ihtiyaçlarına karşılık verebilmek için depolama faaliyetleri verdikleri önemi artırmaları gerekmektedir.

Depo içerisinde farklı süreçler bulunmaktadır. Bu süreçlerde sağlanacak verimlilik artışları depo operasyonlarının performansı üzerinde etkili olmaktadır. Bu süreçlerden en önemlisi sipariş toplama sürecidir. Sipariş toplama süreci gerek müşteri ihtiyaçlarının karşılanmasında gerekse sektörel gelişimden kaynaklı etkilendiğinden ötürü farklılık gösterebilmektedir.

Günümüz teknolojisi hızlı bir şekilde gelişirken firmalar operasyonlarında bu teknolojilerde maksimum oranda faydalanmak istmektedirler. Gelişen teknoloji ve sektörel hacmin büyümesi ile birlikte firmalarda büyük verileri üretilmeye ve depolanmaya başlanmıştır. Depolanan veriler, firmaların sahip oldukları makinelerin öğretilmesi ile birlikte daha anlamlı sonuçlar üretme ihtiyacını oluşturmuştur. Bu veriler ile makine öğrenimi algoritmaları ile birlikte çalışmalar yapılarak bu ihtiyaçlar giderilmeye başlanmıştır.

Bu çalışma ise müşteri alışkanlıklarının makine öğrenimi algoritmalarından Apriori algoritması kullanılarak müşteri siparişlerinin analiz edilmesi ve elde edilen sonuçlarla depo raf yerleşiminin yeniden planlanmasını kapsamaktadır.

Bu çalışmada, Türkiye’de faaliyet gösteren e ticaret platformu Trendyol’un satış verileri ele alınmıştır. Veri içerisinde müşteri sipariş numarası, müşteri sipariş grup numarası, ürün kategorisi, lokasyon ve sipariş edilen miktar yer almaktadır. Ele alınan veriler incelendiğinde, toplam 100.004 adet ürün satışı ve 100.000 adet satırdan oluşmaktadır. Toplamda 12 adet ürün kategorisinden en çok tercih edilen ürünler TekstilC, TekstilA ve TekstilE olmuştur.

Bu alıřmada, ele alınan verilere ilk olarak Python programlama dilinde Apriori algoritması uygulanmıřtır. Bu algoritmada, gruplanan müşteri sipariřlerinin ürün kategorileri arasındaki birliktelik iliřkisi analiz edilmiřtir. Daha sonra analiz sonucu birbiri arasındaki iliřki yoğunluęu fazla ıkan ürün kategorilerinin, depo raf yerleřiminde daha yakın lokasyonlarda adreslenebilmesi iin algoritma tasarlanmıř ve Python programlama dilinde uygulanmıřtır. Bu algoritma ile ürün kategorisinin adreslenmesi iin lokasyon öneri sistemi geliřtirilmiřtir.

Bu alıřmada ürün kategorileri iin önerilen lokasyon ile müşteri sipariřlerinin daha az mesafe yürütölerek toplanması ve böylelikle firmanın daha verimli depo operasyonuna sahip olması hedeflenmiřtir.



ASSOCIATION ANALYSIS AND DETERMINING PRODUCT LOCATIONS WITH MACHINE LEARNING IN OPERATION CENTER OF E- COMMERCE PLATFORMS

SUMMARY

The epidemic disease caused by the coronavirus, which first appeared in Wuhan, China in December 2019 and was later declared a pandemic, affected the whole world in a short time in health, economy, business life, social life, education and many other areas.

The coronavirus has also accelerated the transformation of the trade that is taking place, in other words, the process of shifting from the physical environment to the virtual environment. E-commerce is one of the sectors that will be most affected in this process.

The entry of the pandemic into our lives has increased the interest in the e-commerce sector and contributed greatly to the development of the sector. In this period, the e-commerce sector has grown globally. The interest in the rapidly growing e-commerce sector is increasing day by day.

The pandemic has brought some difficulties along with its contribution to the development of the sector. The Pandemic, in the period when everyone was closed to their homes and afterwards, has also led to a change in the shopping habits of many people. This change has caused e-commerce platforms operating in the sector to experience various difficulties in storage operations. Therefore, companies operating in the sector need to increase the importance they attach to storage activities in order to respond to customer needs.

There are different processes in the warehouse. The productivity gains to be achieved in these processes have an impact on the performance of warehouse operations. The most important of these processes is the order picking process. The order picking process may differ due to both meeting customer needs and being affected by sectoral development.

Order picking is the process of selecting and picking products from a warehouse to fulfill customer orders. Order picking is a very important process in the e-commerce industry as it directly affects customer satisfaction and delivery time. In most companies it is still predominantly done manually by human operators, and this is costly due to significant manual labor.

Efficient order picking is critical to ensuring e-commerce success. Customers expect timely and accurate deliveries, and any delay or error in the order picking process can lead to customer dissatisfaction and lost revenue. Additionally, inefficient order

picking can lead to higher operational costs, reducing the profitability of the e-commerce business.

On the other hand, efficient order picking can lead to increased customer satisfaction and repeat business. It can also increase operational efficiency, reduce costs and increase profitability.

While today's technology is developing rapidly, companies want to make maximum use of these technologies in their operations. With the developing technology and the growth of the sectoral volume, big data has started to be produced and stored in companies. The stored data has created the need for companies to produce more meaningful results with the teaching of their machines. These needs have begun to be met by working with these data together with machine learning algorithms.

Machine learning is a branch of computer science that allows computers to learn directly without being programmed. Algorithms have been increasingly used to analyze these collected datasets and create decision-making systems to come up with an algorithmic solution to complex problems.

This study covers the analysis of customer orders using the Apriori algorithm, one of the machine learning algorithms of customer habits, and the replanning of the warehouse shelf layout with the results obtained.

In this study, the sales data of Trendyol, an e-commerce platform operating in Turkey, are discussed. The data includes customer order number, customer order group number, product category, location and quantity ordered. When the data discussed are examined, it consists of a total of 100,004 product sales and 100,000 lines. Among the 12 product categories in total, the most preferred products were TekstilC, TekstilA and TekstilE.

The problem discussed in the study and the model applied for its solution are very important in terms of warehouse operations. However, it will be least affected by the changing sector and customer habits and will be able to adapt quickly. In addition, the shortening of customer order preparation times will contribute to increasing customer satisfaction and increasing sales volume.

In this study, the data and the problem are introduced. Afterwards, Apriori algorithm was applied to these data in Python programming language.

Python is a versatile program used by data analysts and other professionals to perform complex statistical calculations, create data visualizations, create machine learning algorithms, manipulate and analyze data, and complete other data-related tasks. The Apriori algorithm finds the relationship between objects.

It refers to the algorithm used to calculate the association rules. It reveals how two or more objects are related to each other. In other words, the apriori algorithm is an association rule that analyzes whether people who buy product A also buy product B.

In this algorithm, the association relationship between the product categories of grouped customer orders is analyzed. Then, the algorithm was designed and implemented in Python programming language in order to address the product categories with higher correlation density as a result of the analysis, in closer locations in the warehouse shelf layout. With this algorithm, a location suggestion system has been developed to address the product category.

With the proposed method in this study, it was also seen that the warehouse personnel had a positive effect on the hourly product picking productivity. It has been observed that an efficiency of 31.8% was achieved in the hourly product collection performance of the warehouse personnel.

Thus, with the location suggested for the product categories in this study, it is aimed to collect customer orders by walking less distance and to have a more efficient warehouse operation for the company. Finally, suggestions are shared in order to take this study further.





1. GİRİŞ

2019 yılında pandeminin hayatımıza girmesi e-ticaret sektörüne olan ilgiyi artırmış ve sektörün gelişimine büyük katkı sağlamıştır. Bununla birlikte pandemi, pek çok insanın alışveriş alışkanlığının da değişmesine yol açmıştır. Sektör hacminde büyük bir artmış olması ve müşteri alışkanlıkların değişmesi, sektörde faaliyet gösteren firmalara zorluklar da yanında getirmiştir. Bu zorlukların başında ise depo operasyonlarının daha verimli bir şekilde yönetilme ihtiyacıdır. Bu yüzden müşteri ihtiyaçlarının en hızlı şekilde sağlanması için daha verimli operasyon yürütmek e-ticaret sektöründe faaliyet gösteren firmalar için kaçınılmaz oldu.

Depo operasyonları e - ticaret sektöründe büyük öneme sahiptir ve bu önem özellikle pandemi sürecinden sonra gittikçe artmıştır. Depo içerisinde mal kabul, adresleme, depolama, sipariş toplama ve sevkiyat gibi farklı süreçler bulunmaktadır. Bu süreçlerde sağlanacak verimlilik artışları depo operasyonlarının performansı üzerinde etkili olmaktadır. Bu süreçlerden en önemlisi ise kuşkusuz sipariş toplama sürecidir. Sipariş toplama süreci gerek müşteri ihtiyaçlarının karşılanmasında gerekse sektörel gelişimden kaynaklı etkilendiğinden ötürü farklılık gösterebilmektedir.

Bununla birlikte günümüz teknolojisi çok hızlı bir şekilde gelişmekte olup firmalar operasyonlarında bu teknolojilerde maksimum oranda faydalanmak istemektedirler. Gelişen teknoloji ve sektörel hacmin büyümesi ile birlikte firmalarda büyük verileri üretilmeye ve depolanmaya başlanmıştır. Depolanan veriler, firmaların sahip oldukları makinelerin öğretilmesi ile birlikte daha anlamlı sonuçlar üretme ihtiyacını oluşturmuştur. Bu veriler ile makine öğrenimi algoritmaları ile birlikte çalışmalar yapılarak bu ihtiyaçlar giderilmeye başlanmıştır.

Bu çalışma kapsamında e - ticaret sektöründe hizmet veren türkiyenin önde gelen firması Trendyol'un depo operasyonu ve yerleşim düzeni incelenmiştir. Depo operasyonu ve yerleşim düzeninin üretilen veriler kullanılarak daha verimli olabilmesi için müşteri siparişleri incelemeye alınmıştır. Çalışmada ilk olarak firmanın sahip olduğu depo yerleşim düzeni ve üretilen veri hakkında bilgi verilmiştir. Firmanın sipariş toplama sürecini almış olduğu müşteri siparişlerinin depo yazılım sistemi

tarafından gruplanması ile oluşan müşteri grup numaraları analiz edilmeye karar verilmiştir.

Bu çalışmada müşteri grup numaralarının analizinde birliktelik kuralı uygulanarak makine öğrenimi algoritmalarından bu alanda en yaygın olanu apriori algoritması kullanılmıştır. Birliktelik kuralı ile elde edilen ilişkiler kullanılarak müşteri sipariş gruplarının toplanması için gerekli mesafe ve sipariş toplama süresinden kazanç sağlanabilmektedir. Apriori algoritmasının uygulanması sonucu elde edilen ilişkiler incelenerek ürün kategori ileri için lokasyon öneri sistemi geliştirilmiştir. Geliştirilen lokasyon öneri sistemi modelinde ilk birliktelik analizinde yer alan kurallar içerisindeki ürün kategorilerin kartezyen konum düzlemindeki ortalama konum değerleri hesaplanmaktadır. Daha sonra birliktelik ilişkisine göre belirlenen kuralda yer alan ürün kategorisinin bunlara en yakın lokasyonda yer alan konuma göre öneri sistemi geliştirilerek, birbirleri ile ilişkisi fazla olan ürün gruplarının daha yakın lokasyonlarda adreslenmesi sağlanmıştır.

Çalışmanın son bölümünde ise geliştirilen lokasyon öneri sistemi sonucunda belirlenen sipariş gruplarının toplama mesafeleri karşılaştırılarak elde edilen kazanç ortaya konulmuştur. İlave olarak da modelin daha verimli hale getirilmesi için gelecek çalışmalarda yapılacak öneriler paylaşılmıştır.

2. MAKİNE ÖĞRENİMİ, DEPOLAMA VE LİTARATÜR ARAŞTIRMASI

2.1. Makine Öğrenimi Tarihcesi ve Kullanım Alanları

Makine öğrenimi (ML), bilgisayarların doğrudan programlanmadan "öğrenmesini" sağlayan bir bilgisayar bilimi dalıdır (Verbraeken ve ark, 2020). Yapay zekâ hareketinin kökleri makine öğrenimine dayanır ve özellikle tahmin ve optimizasyon gibi hedef ve uygulamalara yönelik çalışmaları kapsar. Günümüzde makine öğrenimi, çok boyutlu veri kümeleri ile birlikte ele alınmaktadır. Yeni teknolojilerin akıl almaz gelişimi, veri toplamayı hızlandırdığı gibi, daha fazla veriye ulaşmayı da kolaylaştırmıştır. ML algoritmaları, toplanan bu veri kümelerini analiz etmek ve karmaşık sorunlara algoritmik bir çözüm getirmede karar verme sistemlerini oluşturmak için giderek daha fazla kullanılır olmuştur (Bi ve ark, 2019). ML geçmişten günümüze önemli derecede yol katetmesi, neredeyse girmedığı alan bırakmaması ve kompleks yapılara sahip olması, bu kavramın tanımını net olarak yapmayı oldukça zorlaştırmaktadır. Makine Öğreniminin kavramsal tanımından ziyade tarihsel süreç içerisinde ne tür amaçlar doğrultusunda kullanıldığının ve geldiği noktanın ortaya konulması, bu sistemin doğru anlaşılmasına katkı sağlayabilir.

En genel ve yüzeysel tanımıyla Makine Öğrenimi, toplanan verileri ayrıştıran, bu verilerden öğrenen ve daha sonra bilinçli kararlar vermek için öğrendiklerini uygulayan algoritmalar içeren bir yapay zekâ uygulamasıdır (Mohri, ver ark, 2018). Bu uygulama, sayısız sektöre yayılım göstermekte ve her türlü görevi yerine getirmeye odaklanmaktadır. Geri plandaki algoritmalar, sanal bir kişisel asistan gibi çalışmaktadır ve bu sistemler, sürekli olarak öğrenmeye programlanmıştır.

1950'ler, makine öğreniminin basit algoritmalar kullanılarak yürütüldüğü dönemdir. Ancak 1943 yılında Walter Pitts ve sinirbilimci Warren McCulloch tarafından yayımlanan bir makalede, beyindeki her nöronun basit bir dijital işlemci, beyin de bütün bir bilgi işlem makinesi olduğu iddia edilmiştir (Gopani, 2022). 1950'ye gelindiğinde ise matematik ve bilgisayar bilimcisi olan Alan Turing, makinenin düşünebildiği ve bir insanı kandırabilecek kadar mantıklı olabildiğini ispatlayan Turing testini geliştirmiştir (Ciardo ve ark, 2022). 1952'de IBM'den Arthur Samuel,

dama oynayan ilk bilgisayar programını yazmıştır. Bu algoritma iki oyunculu oyunlar için halen daha kullanılmaktadır. Modern anlamda ise makine öğreniminin tarihinin, Cornell Üniversitesi'nden psikolog Frank Rosenblatt'ın, insan sinir sistemini temele alarak harfleri tanımak için geliştirdiği bir uygulamaya dayandığı söylenebilir (Fradkov, 2020). Rosenblatt, geliştirdiği bu sistemde elde ettiği matematiksel verileri, bizzat kendisi analiz etmiştir. Bush ve Mosteller'a (1951) göre bu makinenin öğrenme modeli, psikolojide geliştirilen hayvan ve insan öğrenme modellerine yakındır. Buna benzer bir diğer yazılım 1967'de, rastgele bir şehirde başlayan ve komşu şehirleri tekrar tekrar ziyaret eden bir satıcının, 'gezgin satıcı problemini' çözen algoritmasıdır (Gubin ve ark, 1967). 1981'de Dejong, algoritma tabanlı, “Açıklamaya Dayalı Öğrenmeyi” keşfederken, 1985’de Sejnowski tarafından metin ve ses eşleştirmesi yapan NetTalk sinir ağı yazıldı. 1989'da Watkins, pratik öğrenme uygulamalarına dayalı Q-öğrenme algoritmasını geliştirmiştir. 1990'larda ise ileri istatistiksel yöntemler popüler hale gelmiştir. 1997 yılı ise Makine Öğrenmesi için dönüm noktasıdır denilebilir. IBM tarafından geliştirilen Deep Blue, dünya satranç şampiyonunu mağlup etmeyi başarmıştır (Gopani, 2022). 2000’li yıllarda Makine Öğrenmesi tüm dünyaya yayılırken, 2009 yılında ilk defa dünya çapında büyük veri toplanmıştır. Günümüzde ise özellikle mobile reklamlar aracılığıyla veri toplanması akıl almaz boyutlara ulaşmıştır (Gopani, 2022).

İlk ortaya çıktığı günden bu yana profesyonel alanların çoğuna entegre olabilen Makine Öğrenmesine, günlük hayatta da pek çok yerde rastlanılabilir. Sosyal medya, haritalar, tercüme programları, hastalıkların teşhisi, yüz tanıma sistemleri, sanal asistanlar, trafik uyarıları, dolandırıcılık tespiti, bankacılık sistemleri, sürücüsüz (otonom) araçlar, ürün önerileri (kişiselleştirilmiş reklamlar) gibi uygulamalar, tüm insanların faydasına sunulan Makine Öğrenmesi sistemlerinden oluşmaktadır (Alenezi ve Faisal, 2020; Al Garadi ve ark, 2019; He ve ark, 2014; Kulkarni ve ark, 2017; Saranya ve ark, 2020).

2.2. Depolama

Tedarik zincirinde ve lojistik süreçlerde şirket stratejisini plânlarken, “depolama” fonksiyonu ve bu fonksiyonun işleyişindeki süreçte şirketin benimseyeceği “depo yönetimi” anlayışı önem arz etmektedir. Bu bölümde depolama kavramına ilişkin başlıklar açıklanacaktır.

Depo operasyonlarının doğru işlevi, kullanılan teknolojinin türüne ve bunların kullanımına bağlıdır. Üreticilerden satıcılara ve müşterilere her gün binlerce ton mal taşınmaktadır. Ürün bir satıcıya teslim edildikten sonra, belirli bir süre depolanmak üzere saklanılabilir (Kocifaj, 2013). Depo, bir şirketin lojistik sistemi veya tedarik sistemi içindeki ambar işlevine ve konumuna bağlı olarak birden fazla faaliyetin yürütüldüğü bir yerdir (Marasová ve Šaderová, 2019).

Depolar, dağıtım ve üretim olmak üzere iki türden oluşurlar. Üretim tesisinde hammadde, yarı mamul ve mamul ürünlerin depolanması için üretim depoları kullanılırken, dağıtım deposu, farklı tedarikçilerden gelen ürünlerin müşterilere teslim edilmek üzere toplandığı bir depodur (Saderová, 2017). Depolar, tedarik zincirlerinin düzenleyici unsurlarıdır, malzeme akışlarını yönlendirir ve dönüştürür. Bunun dışında depolar, ürünlere değer katan ve bulunabilirliğini artıran görevleri de yerine getirmektedir (Jacyna ve ark, 2015).

Bunların dışında depoların işlevsel ve teknik yönü de vardır. İşlevsel yön, malların alınması, kontrol, depolama, sipariş toplama, biriktirme, tasnif etme ve nakliye gibi faaliyetleri kapsarken, teknik yön, malzeme ve bilgi akışını sağlayan araçlara (raflar, forkliftler, donanım ve yazılımlar vb.) vurgu yapar (Alias ve ark, 2017).

Depo tasarımı; depo güvenliği, işçi sağlığı ve güvenliği, iş kazaları ile ürün güvenliği konularında en önemli etkenlerden biridir (Çavuşlar, 2020). Bu yüzden depo tasarımı yapılırken özenli ve titiz bir çalışma sergilenmelidir.

2.2.1. Depolama faaliyetleri

Tipik depo faaliyetleri, malzemelerin kabul ve kontrol prosedürüne karşılık gelen teslim alma süreci ile başlar. Depolama süreci, stok tutma birimi numaralarının ürünlere atanmasını, stoklama alanında öğelerin sınıflandırılmasını ve tutulmasını içerir. Sipariş toplama sürecinde, sipariş edilen ürünler depolarından alınır. Önceden belirlenmiş lokasyonlara toplama politikasına göre toplanır. Paketleme süreci, siparişlerin paketlenmesi ve nakliye için hazırlanmasından oluşur. Malzemelerin depo faaliyetleri içerisindeki son işlemi olan sevkiyat sürecinde ise malzemelerin sevkiyat sürecidir (Cergiboza, 2019).

Depolama faaliyetlerinin en önemli adımları aşağıda sıralanmıştır:

2.2.1.1. Mal kabul

Mal kabul; malzemenin fiziki olarak teslim alınmasını, satın almaya uygunluğunun incelenmesini, tanımlanmasını, varış noktasına teslim edilmesini ve teslim alma raporlarının hazırlanmasını kapsayan ilk aşamadır. Mal kabul aynı zamanda, tüm malzemelerin muhasebeleştirilmesini ve zamanında uygun varış noktalarına teslim edilmesini sağlamak için şirket içindeki diğer departmanlarla koordinasyonu içerir. Verimli bir alım departmanı, bir şirket içinde sorunsuz operasyonları sürdürmek için büyük bir öneme sahiptir. Bu aşama, üretim veya diğer amaçlar için ihtiyaç duyulan tüm malzemelerin muhasebeleştirilmesini, zamanında ve verimli bir şekilde uygun varış noktalarına teslim edilmesini sağladığından, lojistik sürecinin kritik bir parçasıdır (Rimienė ve Grundey, 2007). Nakliye araçlarından boşaltılan malzemeler depo personelleri tarafından gerekli görüldüğü takdirde kalite ve miktar açısından kontrol edilmektedir.

2.2.1.2. Adresleme

Malın kabul edilmesinden sonraki aşama “Adresleme” aşamasıdır. Lojistik tesis içi operasyon sırasında ürünlerin veya taşıma birimlerinin (sepet, koli, palet vb.) fiziki olarak tesis içindeki bir adrese yerleştirilmesi, mala verilen kimlik numarası ile malın yerleştirileceği alanın kimlik numarasının eşleştirilerek sisteme kaydedilmesi sürecidir (Logistics Terms Glossary – Lojistik Terimler Sözlüğü, 2022). Ürünlerin raflara ve stoklanacağı alana hangi mantık içerisinde adresleneceği de sipariş toplama süreci açısından büyük önem arz etmektedir.

2.2.1.3. Depolama

Lojistik faaliyetinin en önemli adımlarından biri; bitmiş ürünlerin teslim edilene kadar düzgün bir şekilde depolanmasıdır. Doğru depolama, bir şirketin pazarlamasının etkinliğinde belirleyici bir role sahiptir. Teknolojideki son gelişmelerle birlikte, depolamanın önemli ölçüde geliştiğinden söz edilebilir. Tek katlı otomatikleştirilmiş depolar, sınırlı sayıda çalışanla, eski çok katlı depoların yerini almıştır. Ayrıca doğru ve modern depolama ile işçilik maliyetleri azaltabilir ve daha fazla stok kontrolüne sahip olunabilir (Saderova ve ark, 2021).

Hem kısa hem de uzun vadeli depolama, lojistik planlamanın parçalarıdır. Depo yönetim sistemleri lojistik planlamayı da mümkün kılar. Örneğin, lojistik

planlamacıları, depo alanı mevcudiyetini, sıcaklığı, tesisleri, demiryolu hattı veya tersanelere yakınlık gibi özel gereksinimleri dikkate almalıdır. Ayrıca, depolardaki organizasyon, lojistik planlamanın bir parçasıdır. Sık sık hareket eden veya yakında taşınması planlanan mallar deponun önüne yerleştirilir. Daha az talep gören ürünler arkaya doğru saklanır. Çabuk bozulan ürünler ise genellikle rotasyona tabi tutulur, bu nedenle önce en eski ürünler gönderilir (Jenkins, 2022).

2.2.1.4. Sipariş toplama

Depolarda sipariş toplama için birçok farklı teknik mevcuttur. Geleneksel yöntemde işçiler, siparişleri kâğıt listeler tutarak toplarlar. Ancak modern sistemler, kâğıdı neredeyse hiç kullanmadan bu süreci yönetir. Bu sistemler, verileri işleyen depo yönetim sistemine (WMS) çevrimiçi olarak bağlanan mobil veri giriş cihazlarıyla çalışır. Modern depolarda; Sese Göre Seçim (PbV) veya Işıklı Seçim (PbL) sistemleri bulunmaktadır (Reif ve Günthner, 2009). Tüm bu teknolojilerin belirli avantajları olduğu gibi dezavantajları da bulunmaktadır. PbV, çalışana, bilgisayarın konuşma çıktısı aracılığıyla tüm talimatları vererek onu destekler. Ne yazık ki, bu sistemleri gürültülü endüstriyel ortamlarda kullanmak zorlayıcıdır. Sesli destek sistemleriyle karşılaştırıldığında PbL, her saklama bölmesine küçük lambalar takarak işçiye görsel yardım sunar. PbL sistemleri, raf konstrüksiyonuna entegre edilmesi gerektiği, çok pahalı oluşu ve yeniden inşa edilmeye karşı esnek olmaması nedeniyle problemli olarak görülür. Günümüzde artırılmış gerçeklik uygulamalarıyla sipariş toplama daha esnek ve kolay bir şekilde gerçekleştirilebilmektedir (Reif ve Günthner, 2009).

2.2.1.5. Sevkiyat

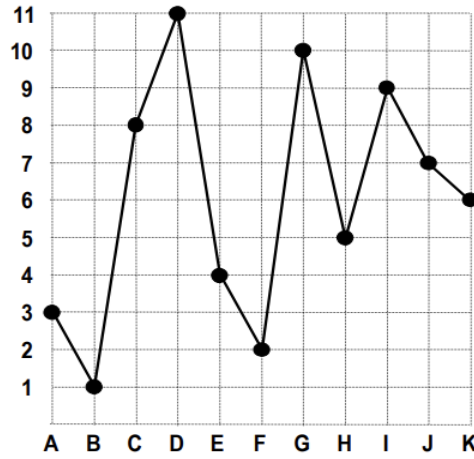
Siparişlerin teslim tarihi, yeri, yükleme aracı, güzergâhı, miktarı vb. belirli kriterlere göre tasnif edilerek resmi belgelerle birlikte gerekli yerlere teslim edilmek üzere depodan alınmasıdır (Lojistik Terimler Sözlüğü, 2022). Bu aşama, nakliye olarak da bilinse de nakliye, aslında sevkiyat adımı içerisinde yer alır. Sevkiyat özünde, malların, bir lojistik merkezinden, müşterilere ulaştırılmak üzere fiziksel olarak taşınmasını ifade eder. Sevkiyat aşaması da kapsamlı bir süreçtir. Bu sürece, dokümantasyondan taşıyıcılarla çalışmaya, izleme ve işlemeye, teslimat sürelerine ve nakliyeye kadar, tedarik zinciri boyunca, mallar hareket ederken gerçekleşen tüm işlemler dâhildir (Lopienski, 2021).

2.2.2. Depolama yerleşim planı

Depo sistemi, depo alt sisteminin teknik tarafını oluşturur. Depo sistemi, depolama alanı için bir bütünü oluşturan iki veya daha fazla cihazı ve aracı (depolama ve hizmet ekipmanı) bir arada tutar. Bu sistem ayrıca, deponun yönetimini de içerir. Depo sisteminin bileşimi, seçilen depo teknolojisine bağlıdır ve raf montajı ile servis ekipmanı olmak üzere iki temel parçadan oluşur (Baker ve Canessa, 2009). Depo sisteminin en önemli parçası ise raf yerleşim planıdır. Yerleşim planı, ürünün muhafazasında önemli bir konuma sahip olduğundan, müşterilerin memnuniyet, sadakat ve güvenini sağlamada önemli bir konuma sahiptir. Bununla birlikte doğru bir şekilde tasarlanan raf yerleşim planı, ürünlerin toplanmasında depo çalışanlarına kolaylık sağlamaktadır.

Depolama yönetimi, üretim verimliliğini artırmada çok önemli bir faktördür ve WMS'in önemli bir parçasıdır (Masae ve ark, 2020). Depolama sürecinde yaygın olarak kullanılan iki toplama yöntemi vardır. Bunlardan birincisi siparişe dayalı toplamadır ve bu yöntem, çeşitliliği az ancak büyük miktarda olan malların depolanması için uygundur. Bir diğeri ise toplama verimliliğini artırmayı amaçlayan grup bazlı toplamadır. Bu yöntemde, birden fazla sipariş tek bir grupta birleştirilir. Ancak bu yöntem, siparişlerin gecikmesine neden olabilir. Siparişlerin daha hızlı bir şekilde tüketiciye ulaştırılması; depo düzeni, depolama yeri, toplama yolu ve siparişlerin toplu işlenmesi gibi faktörlerden etkilenir (Masae ve ark, 2020). Dolayısıyla raf yerleşim planı, depolamada temele alınabilecek bir argümandır ve “sevk edilebilirlik” ile “ulaşılabilirlik” özellikleri adına işlevlere sahiptir.

Bir yerleşimin temsili gösterimi; permütasyon, ikili diziler, veya graf şeklinde yapılabilir. Permütasyon matrisi, satır ve sütununda tesis ve yerden oluşan $n \times n$ boyutlu matristen oluşmaktadır. Aynı zamanda bu çalışmada kullanılmış olan bir diğer gösterim şekli olan graf gösterimi, çizgi ve noktalardan oluşmaktadır. Noktanın bulunduğu kısımda x eksenini yerleri, y eksenini tesisleri ifade etmektedir. Şekil 2.1’de graf gösterime ait bir örnek sunulmaktadır (Çalış, 2009).



Şekil 2.1.Tesis Yerleşim Gösterimi

Depo yerleşim planı, günümüzde makine öğrenmesi temelinde, çeşitli algoritmalar kullanılarak etkili ve pratik bir şekilde yapılabilmektedir. Bu planın yapılması;

Küçük yerleşim alanlarını verimli kullanmak,

Depo operasyonlarını kolaylaştırmak,

Maliyetleri (malzeme hasarı, yerleşim alanı kullanımı, işçilik vb) azaltmak,

Elleçleme ve depolama esnekliği oluşturmak,

Düzenli bir şekilde depo faaliyetlerini yürütmek,

başta olmak üzere, çeşitli amaçlara yönelik gerçekleştirilir (Evinsel, 2010).

2.3. Literatür Araştırması

Depo operasyonlarını optimize etmeye yönelik bilimsel araştırmalar uzun süredir devam eden bir geleneğe sahiptir ve yıllar boyunca literatür birikmiştir. Özellikle pandemi sonrası önemi gittikçe artan tedarik zinciri yönetimi gerek bilimsel alanda gerekse uygulamada firmaların bu alandaki araştırmalarını artırmasına yol açmıştır.

Sipariş toplama, müşteri siparişleri bazında malların depo yerlerinden alınmasını ifade eder ve bir depodaki en önemli faaliyetlerden biridir. Bu süreç aynı zamanda tüm depolama süreçlerinin en zahmetli ve zaman alıcı sürecidir. Bu nedenle, bir depoyu sipariş toplama konumları ile modellemek ve tüm sipariş toplayıcıları aynı modelde düşünmek, iyi sipariş toplayıcı rotalarını belirlemek ve aynı anda seyahat sürelerini azaltmak için çok faydalıdır.

Gerçekleştirilen kapsamlı literatür taraması neticesinde, makine öğrenimi ve depo içi yerleşim planı üzerine çok sayıda çalışma yapıldığı görülmüştür. Aşağıda bunlardan bazıları özetlenmiştir:

Calzavara ve ark. (2017) sipariş toplama için gereken süreyi azaltmaya yardımcı olan planlama modelleri geliştirerek sipariş toplama süreçlerinin operasyonel verimliliğini artırmayı amaçladıkları çalışmalarında, siparişlerin geleneksel paletler, yarım paletler ve çekme sistemiyle donatılmış yarım paletlerden toplandığı durumlar için ekonomik ve ergonomik performans ölçümleri geliştirmişlerdir. Farklı raf konfigürasyonlarını kapsamlı bir şekilde değerlendirmek için çok senaryolu bir analizde matematiksel modeller geliştirilmiş ve uygulanmıştır. Sonuç olarak hem maliyet hem de enerji tasarrufu açısından, ürünlerin geleneksel paletler üzerinde depolanmasının yalnızca büyük ürün hacmi söz konusu olduğunda uygun olduğu, ürün hacmi küçükse veya çekme listesi başına çekilen ürün sayısı yüksekse, önerilen raf konfigürasyonlarının daha iyi performans gösterdiği anlaşılmıştır.

Demircan (2018), örnek olay yöntemi kullanarak gerçekleştirdiği araştırmasında, taşıma ve enerji sarfiyatının minimize edilmesi amacıyla, yeni bir depo sistemi önermiştir. Bu noktada sistemin odağı, doğru lokasyon belirlemek, benzer ilişki ağı içerisinde yer alan müşterileri kümelemek ve onları bir arada tutmaktır. Araştırmanın bulguları için halihazırda kullanılan sistem sonuçları ile önerilen modülün sonuçları kıyaslanmıştır. Sonuç olarak, önerilen modülün enerji sarfı ve erişilebilirliği dikkate alındığında, müşteri alış hacminin %9 ila %56 arasında seyrettiği, yine ürünlerin sevkinde indirilen paletlerin %12 ila %56 arasında bir iyileşme gösterdiği görülmüştür. Bu iyileşme, Anova sonuçlarına göre anlamlı düzeydedir. Buna ek olarak müşteriler tarafından sipariş edilen ürünlerin aynı koridorda bulundurulma yüzdesi %44 civarında olmuştur. Sonuç olarak, önerilen modülle daha erişilebilir bir depo tasarımının yapıldığı kanıtlanmıştır.

Olçay (2018) depo malzemelerinin taşınması, depolama yöntemleri ve bilgi teknolojileri yeteneklerini kategorize etme ve sınıflandırma amacıyla yaptığı çalışmada, İzmir ilinden seçilen şirketlerin depolarından örnek olay uygulaması yapmak için veriler toplamıştır. Yazar, Warehouse 3.0 ve Warehouse 4.0 sistemlerinin teknik kullanımı sonucu daha iyi performans gösteren, otomasyon ve akıllı teknoloji kullanımı ve bunların teknolojiyle etkileşimini ortaya koyan sonuçlar elde etmiştir.

Ayrıca bu çalışma, alanda yapılabilecek diğer çalışmalar için için bir yol haritası olarak kullanılabilir.

Wu ve Chiu (2018), sosyal sermaye olarak adlandırılan LSCM (Logistics and Supply Chain Management- Lojistik ve Tedarik Zinciri Yönetimi) içindeki insan ilişkilerinin, örgütsel aktörler arasındaki sosyal bağların, paylaşılan kimlik duygusunun, LSCM'nin çatışma olasılığını azaltmada giderek daha fazla önemli hale gelen rolünün, paylaşılan vizyon ve güven duygusu ile ilgili işbirlikçi davranışı savunma becerisini ortaya koymak amacıyla bir nitel çalışma yürütmüşlerdir. Elde edilen bulgular yapay zekânın depo tasarımına yaptığı katkı bağlamında önemlidir. Nitekim çalışmanın sonuçlarından birinde; yapay zekâ deneyimine sahip depo yöneticilerine göre, depo içi tasarımda yapay zekânın benimsenmesi, yalnızca çıktıyı ve üretkenliği iyileştirmekle kalmayıp aynı zamanda farklı vardiyalarda süreçleri ve operasyonları standartlaştırmakta ve böylece her iki vardiyada da kurumsal aktörlere yeni fırsatlar sunmaktadır.

Zunic ve ark. (2018), bir depoda optimum ürün yerleştirme ile ilgili sorunları, gerçek dünyadan bir vaka çalışması sunarak ayrıntılı olarak analiz etmişlerdir. Çalışmada, Bosna Hersek'te orta ila büyük ölçekli bir lojistik şirketi deposuna dahil edilen bir optimizasyon modülü geliştirilmiş, elde edilen iyileştirmelerin analizi yapılmış ve sonuçlar, boş bir depo yaklaşımıyla kıyaslanmıştır. Sonuç olarak, geliştirilen modül sayesinde önceki iş akışının iyileştirilmesi noktasında, çok fazla alan kaldığı görülmüştür. Ürün boyutları ve ağırlığı net olarak sunulduğunda, ortaya çıkan sorunun karmaşıklığı ortaya konulduğunda ve şirket veritabanlarında bu tür bilgiler bulundurulduğunda, geliştirilen modülün depo raf tasarımı için oldukça faydalı olacağı ortaya çıkmıştır.

Mahroof'un (2019) çalışmasının odak noktası, yapay zekâ depolama teknolojisine hazır olup olunmadığını, depo operasyon personeli ve yöneticilerin bakış açıları aracılığıyla keşfetmektir. Bu nedenle araştırmada, insan davranışını ve davranış değişikliklerini incelemek için yarı yapılandırılmış görüşme sorularından yararlanılmıştır. Toplanan verilerden elde edilen bulgulara göre yapay zekânın benimsenmesine ilişkin çeşitli fırsatlar ve potansiyel engeller söz konusudur. Yapay zekâ deneyimi çok az olan veya hiç olmayan yönetim, bu teknolojiye önemli faydalar algılamada başarısızdır. Bu durum, operasyonel yöneticilerin olumsuz eğilimlerine katkıda bulunmaktadır. Ancak uygulama ve teknoloji ekipleri, şirketin teknolojik

yönüne ilişkin daha fazla iç görü sahibidir. Bulgular aynı zamanda, operasyonel yönetimin, dijital becerilerden yoksun olduğunu göstermektedir.

Öz (2019), savunma sanayinde faaliyet gösteren bir atölye imalat firmasının işlem kayıtlarını veri olarak kullanmış ve bu tür ortamlarda bir deponun yanıt verebilirliğini önemli kılan faktörleri sıralamıştır. Önerilen modül parçaları tepsilere, tepsileri dikey depolama üniteleri ve raflara atayan tamsayı bir doğrusal programlama modelidir. Önceki sistem ile önerilen modülün çıktıları kıyaslandığında, zaman açısından %10 daha fazla verim alındığı ve sipariş toplamada, tepsilerin çağrılmasının %65 civarında azaldığı görülmüştür. Elde edilen bulgular, deponun yanıt verebilirliğinin önemini gösterse de analiz edilen şirketin deposu, teslim tarihlerinden önce, çekme siparişlerinin yüksek bir yüzdesini yerine getirememektedir. Bu sorunun altında yatan neden bu tür ortamların ihtiyaçlarına yönelik otomatik depolama ve geri alma sistemi (AS/RS) kurulumunun, dikey kaldırma modülü (VLM) sistemi için uygun olmamasıdır.

Wei, Feng ve Liu (2020) araştırmalarında, depo verimliliğini artırmak amacıyla, Yinelemeli Yerel Arama algoritmasına ve seçici yönlendirme optimizasyon stratejisine dayalı bir tasarım geliştirmişlerdir. Çalışmada yinelemeli süreçteki yüksek kaliteli çözümleri ortaya koymak için bir havuz stratejisi belirlenmiş ve yinelemeli evrimdeki baskın olmayan çözümler saklanarak güncellenmiştir. Bununla birlikte geniş arama alanını sınırlandırmak ve başka bir nesnel yöndeki önemli bozulmayı sınırlamak için epsilon kısıtlaması benimsenmiştir. Sonuç olarak çalışmada önerilen algoritmanın, baskın olmayan çözüm sayısı, hiper hacim ve boşluk metriği açısından karşılaştırma algoritmalarından daha üstün ve algoritmanın kararlılığının da yine iyi seviyelerde olduğunu görülmüştür.

Subrahmanyam ver ark. (2021) ambar tasnifindeki sorunları gidermek amacıyla, ürünleri kolay ve verimli bir şekilde sıralamak için hem gezgin robotları hem de rafları yeniden tasarlayan bir modül geliştirmişlerdir. Ayrıca piyasada var olan robotların maliyetinin düşürülmesine ve düşük ölçekli endüstriler için daha uygun hale getirilmesine odaklanılmıştır. Tamamen otomatik olan gezici robotun işlem yapması için, kullanıcının ürünün ilk konumunu ve son hedefini vermesi yeterlidir. Bunun için de gezici robotun güncellemelerini, kullanıcının cep telefonuna almak için bir GSM modülü de geliştirilmiştir. Rover, rafın gerekli konumuna ulaşmak için programlanmıştır. Bunun yanı sıra depo rafları, nesnenin boyutlarına ve amacına bağlı

olarak tasarlanır. Rafin gezici üzerine gerekli bir nesneyi yerleřtirmesi iin bir mekanizma geliřtirilmiřtir. Bu sayede iřlemler daha da basitleřtirilmiřtir ve operatrn tm rafta rn aramasına gerek yoktur.

Quan, Ge ve Zhang (2022), depolar iin sipariř toplama problemini zmek iin, grafik teorisi bilgi modellemesini kullanarak TSP (Travelling Salesman Problem) kavramını ortaya koymuřlardır. Lineer programlamada tipik bir tamsayı programlama problemi Matlab'da zme ulařtırılmıř, veri iřleme iin ise Python programından faydalanılmıřtır. Sonu olarak geliřtirilen model, hayatın birok ynne uygulanabilir. Klasik uygulama alanları esas olarak; depo toplama, spermarket alıřveriři ve tur rotası dzenleme gibi uygulamalardır.

Tufano, Accorsi ve Manzini (2022), depo tasarımını desteklemek amacıyla; depolama teknolojisi (ST), malzeme tařıma sistemi (MHS), depolama tahsis stratejisi (SAS) ve deponun toplama politikasını (PP) tahmin edebilen sınıflandırıcılar geliřtirmiřlerdir. rnek olay yntemiyle gerekleřtirilen alıřmada, kıyaslama metrikleri ile ğrenme tablosu oluřturulmuř, depo ynetim sistemindeki verilerin mevcudiyetinin, tahminlerin doğruluğunu nasıl etkilediğiy araştırılmıřtır. Yaklařımı doğrulamak iin; otomotiv, imalat, yiyecek-iecek, kozmetik ve yayıncılık gibi farklı endstriyel sektrlere ve dağıtım merkezleri ile nc taraf lojistik sağlayıcılara ait 16 farklı gerek řirketten veri toplanmıřtır. Ampirik testler, makine ğrenimi modellerinin SKU'ları uygun bir ST, MHS, SAS ve PP'ye atayarak stratejik kararların sonucunu doğru bir řekilde tahmin ettiğiy gstermektedir. Bu bulgu, 3PL sağlayıcılarının esnekliğiy ve organizasyonu iyileřtirmeye yardımcı olduğuy kanıtlamaktadır.

Zhuang ve ark. (2022) sipariř iřleme sırasında raf depolama ve raflara robot atama problemini zme ulařtırma amacıyla, devrik ufuk erevesi (rolling horizon framework) ve benzetilmiř sertleřtirme (simulated annealing) yntemlerine dayanarak, her ařamada zel bir eksenel 3-indeks atama probleminin zmn ieren matematiksel bir ayrıřtırma yaklařımı geliřtirmiřlerdir. nerilen epizodik raf alma ve yeniden konumlandırma yaklařımının performansı, byk bir e-ticaret řirketi tarafından sağlanan gerek dnya veri kmesinin dağılımına dayalı olarak oluřturulan girdi verileriyle hem kk hem de byk lekli problem rnekleri zerinde test edilmiřtir. Yaklařımın, kk lekli durumları birkaç saniye iinde sorunu zdğiy ve optimum zme yakın bir dzeltme sresi ile bir zm sağladığı

görülürken, büyük ölçekli durumlarda ise pratikte sıklıkla kullanılan poliçelerden daha iyi performans gösterdiği anlaşılmıştır.

Petersen ve Gerald (2004) , sipariş toplayıcının seyahati üzerindeki toplama, depolama ve yönlendirme kararlarının etkilerini analiz etmiş ve sınıf tabanlı veya hacim tabanlı depolama politikasının kullanılmasının, toplu değerlendirmeye neredeyse aynı düzeyde tasarruf sağladığı sonucuna varmıştır.

Chabot ve ark. (2018), depo yeniden atama problemi için yeni bir formülasyon önerdi. Önerdikleri yöntem ise grafik tabanlı yerleşim algoritması ile ürün toplayıcının hareket mesafesini en aza düşürerek depolardaki iş gücünü azalttıklarını savunmuşlardır. Uygulamış oldukları modelin aynı zamanda iş gücünü 3 kata kadar azalttığını belirtmişlerdir.

ABC depolama, ambarlarda depolama yeri ataması için en popüler sınıf tabanlı politikadır. Bir depolama alanını üç bölgeye ayırır ve en çok talep edilen ürünleri en uygun bölgeye atar. (Silva ve ark ,2022) yılında yapmış oldukları çalışma ile bu üç bölge boyutlarını tahmin etmek için regresyon modelleme kullanarak problemin çözümü için yeni bir metodoloji önermektedir. Önerilen metodoloji, modelleri eğitmek için yeterli veri mevcut olduğunda farklı depolama sistemleri ve sorunları için kolayca uyarlanabilir. Ortaya çıkan doğrusal fonksiyonlar ve karar ağaçları kullanıma sunulur ve uygulayıcılar tarafından kendi depoları için bölge büyüklüklerini belirlemek için kullanılabilir.

2.4. E Ticaret Sektöründe Sipariş Toplama

Pandemi ile birlikte tüm dünyada e ticaret sektörüne olan ilgi artmış ve sektör hacminde yüksek miktarda büyüme gerçekleşmiştir. E ticaret işletmelerinin depo operasyonlarındaki en önemli süreçlerden birisi de sipariş toplama sürecidir.

Sipariş toplama, müşteri siparişlerini yerine getirmek için bir depodan ürün seçme ve toplama işlemidir. Sipariş toplama, müşteri memnuniyetini ve teslimat süresini doğrudan etkilediği için e-ticaret sektöründe çok önemli bir süreçtir. Çoğu şirkette hala ağırlıklı olarak insan operatörler tarafından manuel olarak yapılmaktadır ve bu, önemli ölçüde el emeği nedeniyle maliyetlidir.

E-ticarete sipariş toplamanın zorlukları da vardır. En yaygın zorluklardan biri, günlük olarak karşılanması gereken yüksek hacimli siparişlerdir. E-ticaretin yükselişiyle

birlikte müşterilerin hızlı ve doğru teslimat beklentileri artmış, e-ticaret işletmeleri üzerinde siparişleri hızlı bir şekilde yerine getirme baskısı oluşmuştur.

Verimli sipariş toplama, e-ticaret başarısını sağlamada kritik öneme sahiptir. Müşteriler zamanında ve doğru teslimatlar bekler ve sipariş toplama sürecindeki herhangi bir gecikme veya hata, müşteri memnuniyetsizliğine ve gelir kaybına yol açabilir. Ek olarak, verimsiz sipariş toplama, daha yüksek operasyonel maliyetlere yol açarak e-ticaret işinin karlılığını azaltabilir.

Öte yandan, verimli sipariş toplama, müşteri memnuniyetinin artmasına ve işin tekrarlanmasına yol açabilir. Ayrıca operasyonel verimliliği artırabilir, maliyetleri azaltabilir ve karlılığı artırabilir.

Bu nedenle akademisyenler, e-ticaret depolarında sipariş toplama verimliliği ile ilgili konuları analiz etmeye ilgi gösterdi.

E-ticaret depolarının operasyonel verimliliğini artırmak için, her bir stok tutma biriminin birden fazla lokasyonda saklanabileceği veya bir lokasyonun birden fazla stok tutma ünitesi içerebileceği anlamına gelen çok lokasyonlu depolama sistemleri geliştirilmiş ve uygulamaya alınmıştır. Siparişler toplu olarak alındığında, aynı stok tutma birimini tutan depolama konumlarından toplama yerinin nasıl seçileceği, toplayıcıların toplama görevlerini tamamlamak için ne kadar uzağa gitmeleri gerektiğini açıkça etkiler. Bununla birlikte, farklı depolama sistemleri perspektifinden toplu sipariş toplamanın nasıl optimize edileceğini sistematik olarak inceleyen çok az çalışma vardır. Peng ve ark. (2020) hazırlamış oldukları makalede üç tipik depolama sistemi için toplu sipariş toplama optimizasyon problemlerini formüle etmiş ve bunların üstesinden gelmek için konum aralığı mesafe algoritması, konum seçimi algoritması, yönlendirme algoritması ve sipariş gruplama algoritmasını içeren algoritma paketini geliştirmiştir.

Abelfotoh ve ark. (2019) yapmış oldukları çalışmada küme sipariş toplama yöntemini benimseyen bir el ile toplayıcıdan parça sistemi için bir vaka çalışmasında lojistik firması işbirliği yaparak sipariş toplama problemini çözmeye amaçlamışlardır. Küçük ölçekli örnekler için sipariş toplama problemini çözmek için matematiksel bir model geliştirilmiştir. Daha büyük örnekler için, firma tarafından önerilen bir sipariş toplama heuristiğinin basitleştirilmiş bir versiyonu farklı boyutlardaki problemler için

kullanılmış ve test edilmiştir. Amaç, tüm toplayıcıların toplam katettiği mesafeyi, S-şekil rota stratejisine dayalı olarak hesaplanarak minimize etmektir.

Günümüzde dağıtım merkezlerinde (DC) ve depolarda birçok destek sistemi kullanılmaktadır. WMS olmadan yüksek etkinlikte ürün stoklamak ve sipariş almak imkansızdır. Toplama işleminin etkinliği aynı zamanda depo büyüklüğüne de bağlıdır. Yöneticiler, gerçek zamanlı sonuçların anlaşılması kolay bir düzenleme bekler. Augustyn ve Aurelijia (2019) yapmış oldukları çalışmada, WMS'den elde edilen veriler, toplama sürecinin verimliliğini artırmak için ürünler ve depo yerleşimi hakkındaki istatistiklerle birleştirilmiştir. Verilerin bu şekilde revize edilmesi, depoya doğru yerleştirilmeyen ürünlerin fark edilmesini sağlar. Bu nedenle, deponun etkinliğini artırmanın ilk olası yolu, ürün atama problemini çözmektir. Bu durum ise verileri kullanarak ve basit bir analiz yaparak sipariş toplama sürecinin verimliliğini artırmanın mümkün olduğunu göstermektedir. Çalışmada, WMS'den sıra verileri, depo düzeni ve topolojisi kullanılmaktadır. Veri temizliği, ön işleme ve her konumun koordinatlarının ölçümü yapıldıktan sonra, sipariş toplama simülasyonu yapılabilir. Sonuçlar, toplama süresinin %17.4 - % 25.6 azaltılabileceğini göstermektedir.

Teknolojinin gelişmesi ile birlikte, robotların ve insanların e-ticaret depolama sistemlerindeki koordinasyonu kritik önem taşımaktadır. Son zamanlarda birçok e-ticaret şirketi, siparişlerinin yerine getirilmesine destek olmak için depolarında mobil robotlar kullanmaya başlamışlardır. Suryakant ve ark. (2023) yapmış oldukları çalışmada, parçadan toplayıcıya temelli bir depolama sistemde robot-insan koordinasyonu planlama sorununu araştırmak için kuyruk teorisi tabanlı bir analitik model sunmaktadır. Çalışmada, sipariş toplama sürecinde insan gücü problemini ele almak için kuyruk teorisini kullanılmıştır. Amaç, algılanan iş yükünün, parçadan toplayıcıya sipariş toplama sistemini benimseyen bir e-ticaret deposunun performansı üzerindeki etkisini analiz etmektir. Bu çalışma, bir sipariş toplama sisteminin planlama kararlarında insan faktörlerini dahil etmenin, depo yöneticilerine sistemdeki robotların sayısını dikkatli bir şekilde belirlemelerine yardımcı olabileceğini önermektedir.

3. MAKİNE ÖĞRENİMİ ALGORİTMALARI VE YAZILIMLARI

3.1. ML Algoritmaları

Bu başlık altında, yapay sinir ağları, karar ağaçları, destek vektör makinaları, Naive Bayes, k - means kümeleme ve çalışmada ele alınan apriori algoritması tanıtılmıştır.

3.1.1. Yapay sinir ağları

Yapay sinir ağları (YSA), biyolojik sinir ağlarındaki nöronların sinyal davranışlarından esinlenilerek oluşturulmuştur (Olden ve Jackson, 2002). Karmaşık sinyal yollarıyla birbirine bağlanan bir nöron popülasyonundan oluşan YSA'lar, sonucu tahmin etmek için bir grup ölçülebilir ortak değişken arasındaki karmaşık etkileşimleri analiz etmeye yarar. YSA'lar "aksonlar" ile birbirine bağlanan "nöron" katmanlarına sahiptir. Bu katmanlar 1) bir girdi katmanı, 2) bir veya daha fazla orta "gizli" katman ve 3) bir çıktı katmanı olarak gruplandırılır. Girdi ve çıktı katmanlarındaki nöronlar sırasıyla bağımsız ve bağımlı değişkenlere karşılık gelir. Bitişik katmanlardaki nöronlar, bir nöronun girdilerinin ağırlıklı toplamını bir çıktıya dönüştüren aktivasyon fonksiyonları aracılığıyla birbirleriyle iletişim kurar. Aktivasyon fonksiyonunun türüne bağlı olarak, çıktı, ikili (ağırlıklı toplam belirli bir eşiği aştığında "1" ve aksi takdirde "0") veya sürekli olabilir. Bir nöronun girdilerinin ağırlıklı toplamı, doğrusal veya lojistik regresyondaki katsayılarla biraz benzer (Olden ve Jackson, 2002).

3.1.2. Karar ağaçları

Karar ağaçları (sınıflandırma ve regresyon ağaçları - CART), bir sonucu tahmin etmek için sürekli veya kategorik girdi değişkenlerine dayalı bir dizi karar kuralı oluşturur (Duda ve ark, 2012). Sınıflandırma ağaçları kategorik sonuçları, regresyon ağaçları ise sürekli sonuçları tahmin eder. CART analizi, herhangi bir karar ağacı öğrenme yöntemi için bir şemsiye terim olarak popüler hale gelmiştir (Duda ve ark, 2012). Bununla birlikte, farklı karar ağacı algoritmaları da geliştirilmiş olmasına rağmen, CART en yaygın olan uygulama algoritmasıdır.

3.1.3. Destek vektör makinaları

Destek vektör makinaları (SVM'ler), sınıflandırma ve regresyon problemleri için kullanılan bir dizi denetimli öğrenme yöntemidir (Noble, 2006). SVM'ler, farklı sınıfların gözlemlerini en iyi şekilde ayıran hiper düzlem adı verilen optimal bir sınır oluşturur. 1 boyutta bu sınır bir nokta; 2 boyutta bir çizgi; 3 boyutta ise bir uzaydır. Bununla birlikte, çoğu gözlemin, bir hiper düzlemle ayrılmadan önce dönüştürülmesi gerekir. SVM'ler, verilere "çekirdek işlevi" adı verilen bir veri dönüşümü uygulayarak bu sorunu çözer (Rasmussen ve Williams, 2006). Çekirdek işlevleri, verileri girdi değişkenlerinin ayrılabilir olduğu daha yüksek boyutlu bir alana yansıtır. Popüler çekirdek işlevleri arasında polinom çekirdeği, gauss çekirdeği ve sigmoid çekirdeği bulunur. Çekirdek fonksiyon dönüşümünün ardından, en iyi hiperdüzlem, belirli bir yanlış sınıflandırma düzeyini tolere ederken, farklı sınıflar arasındaki ayrımı maksimize eder. SVM'ler geleneksel olarak ikili sınıflandırma için kullanılır, ancak çok sınıflı sınıflandırma için çoklu ikili karşılaştırma uygulanabilir (Noble, 2006). Sürekli sonuçları tahmin etmek için kullanılabilen ve destek vektörü regresyonu olarak adlandırılan SVM tekniklerinin uzantıları da geliştirilmiştir (Smola ve Schölkopf, 2004).

3.1.4. Naive bayes algoritmaları

Naive Bayes algoritması, bağımsızlık varsayımı yapan Bayes teoremine dayanan basit bir olasılıksal sınıflandırma algoritmasıdır (Webb ve ark, 2010). Naive Bayes, her bir sınıfla ilişkili olasılığı, bir dizi ortak değişkene, yani önceki olasılık ve olabilirlik fonksiyonunun ürününe koşullu olarak hesaplar ve daha sonra en yüksek olasılığa sahip sınıfı "doğru" sınıf olarak seçer (Webb ve ark, 2010). Önceki olasılık, tipik olarak, çalışmanın kendisine veya diğer yayınlanmış literatüre dayalı olarak kişinin sonuç hakkındaki yorumunu yansıtır. Naive Bayes'in regresyon için uyarlamaları mevcut olsa da algoritma, yaygın olarak sınıflandırma için kullanılır.

3.1.5. K –means kümeleme

K -means kümeleme, en basit denetimsiz öğrenme algoritmalarından biridir (Jain, 2010). Kümeleme algoritmaları, veri dağılımının altında yatan yapıyı kullanır ve benzer özelliklere sahip verileri gruplamak için kurallar tanımlar (Jain ve ark, 1999). Bu işlem, veri kümesi hakkında herhangi bir ön bilgi olmaksızın, belirli bir veri kümesinin, kümeleme kriterlerine göre bölümlenmesiyle sonuçlanır. İdeal bir kümeleme senaryosunda, her bir küme, diğer kümelerdeki örneklerden oldukça farklı

olan benzer veri örneklerinden oluşur. Böyle bir farklılık ölçüsü, algoritmanın temel verilerine ve amacına dayanır. Kümeleme, birçok veriye dayalı uygulamanın merkezinde yer alır ve makine öğreniminde ilginç ve önemli bir görev olarak kabul edilir. Ayrıca istatistik, örüntü tanıma, hesaplamalı geometri, biyoinformatik, optimizasyon, görüntü işleme ve çeşitli diğer alanlarda da kullanılmaktadır (Ahmed ve ark, 2017).

3.1.6. Birliktelik kuralları ve apriori algoritması

Birliktelik kuralları, veri madenciliği alanında araştırma ve çalışma yapılmış bir konudur.

Kuralları oluşturabilmek için kullanıcı tarafından belirlenmiş minimum destek ve minimum güven değerlerinden yaygın birlikteliklerin belirlenmesi amaçlanmıştır. Birliktelik kuralı analizinde en çok kullanılmakta olan bazı ifadeleri şu şekilde sıralayabiliriz; destek, güven ve lift değeri.

Destek (support) değeri: Bir değer tüm gözlemler arasındaki görülme sıklığı olarak tanımlanır. X için destek değeri denklem aşağıdaki gibidir.

$$P(X) = N(X) / n$$

Burada n gözlem sayısını, N(X) ise X'in görülme sayısını ifade eder.

Güven (confidence) değeri: X değerini seçenlerin Y seçeneğini de seçme olasılığıdır ve aşağıdaki gibi gösterilmektedir.

$$P(Y/X) = N(X,Y) / N(X) \text{ veya } P(Y/X) = P(X,Y) / P(X)$$

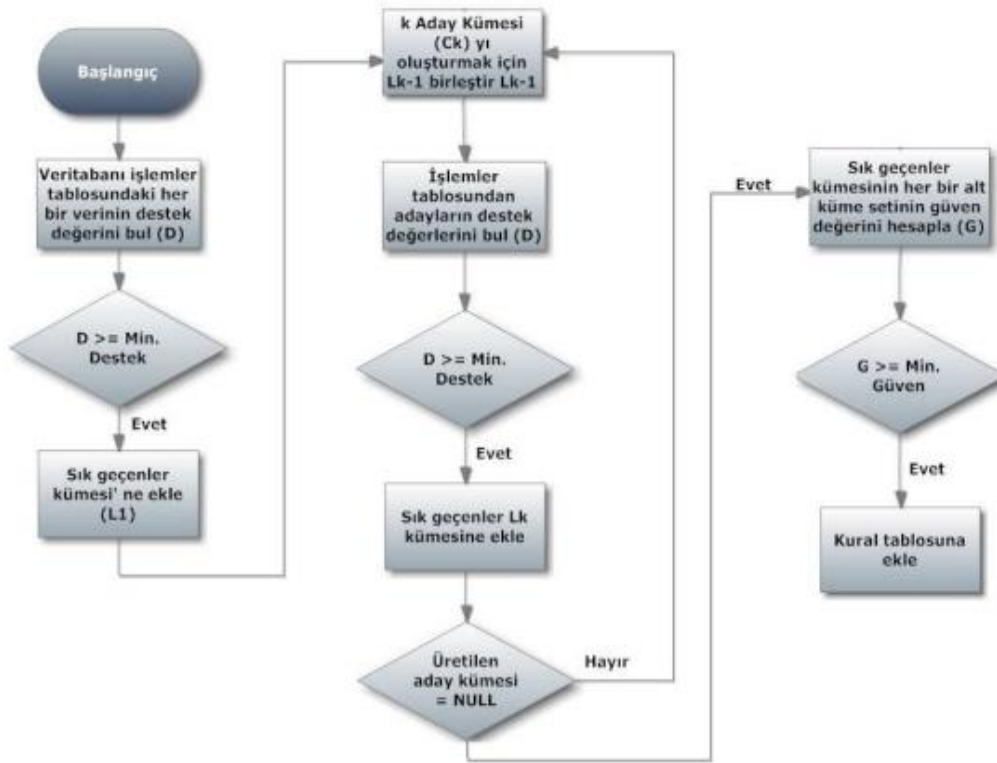
İlgi (lift) değeri: X ve Y'nin istatistiksel olarak bağımsız olması durumunda ne kadar birlikte geçtiklerini bulmayı sağlayan değerdir. Aşağıdaki gibi gösterilmektedir.

$$K(X) = P(X/Y) / P(X)$$

Birliktelik kurallarının oluşturulmasında yararlanılan ve yaygın olarak kullanılan algoritma ise apriori algoritmasıdır.

Apriori algoritması, nesneler arasındaki ilişkilendirme kurallarını hesaplamak için kullanılan algoritmayı ifade eder. İki veya daha fazla nesnenin birbiriyle nasıl ilişkili olduğunu ortaya koyar (Santoso, 2021). Diğer bir deyişle apriori algoritması, A

ürünü satın alan kişilerin B ürünü de satın aldıklarını analiz eden bir birliktelik kuralıdır. Bu algoritmanın birincil amacı, farklı nesneler arasında ilişkilendirme kuralı oluşturmaktır. İlişkilendirme kuralı, iki veya daha fazla nesnenin birbiriyle nasıl ilişkili olduğunu açıklar (Santoso, 2021). Apriori algoritması aynı zamanda sık örüntü madenciliği olarak da adlandırılır. Genel olarak, Apriori algoritması, çok sayıda işlemten oluşan bir veri tabanı üzerinde çalıştırılır. Apriori algoritması akış diyagramı aşağıdaki gibidir (Şekil 3.1.):



Şekil 3.1. Apriori Algoritması Akış Diyagramı.

Bu algoritmanın temel mantığı, eğer k-öge kümesi, minimum destek ölçütünü sağlıyorsa, k-öge kümesinin alt kümelerinin tamamı da minimum destek ölçütünü sağlaması üzerinedir (Gülce, 2010). Apriori algoritmasının adımları şu şekildedir (Softwaretestinghelp, 2023a):

1) Algoritmanın ilk yinelemesinde, her bir öge 1 öge kümesi adayı olarak alınır. Algoritma, her ögenin oluşumlarını sayar.

2) Minimum destek (min_sup) oluşumunu sağlayan öge kümeleri belirlenir. Yalnızca min_sup'a eşit veya daha fazla sayılan adaylar, bir sonraki yinelemeye alınır ve diğerleri budanır.

3) min_sup ile 2 öğeli sık kullanılan öğeler keşfedilir. Bunun için birleştirme adımında, ikili öge kümesi oluşturulur.

4) İkili öge kümesi adayları, min-sup eşik değeri kullanılarak budanır.

5) Bir sonraki yineleme, birleştirme ve budama adımını kullanarak 3 öge kümesi oluşturacaktır. Bu yineleme, 3 öğeli kümelerin alt kümelerinin, yani her grubun 2 öğeli alt kümesinin min_sup'a düştüğü antimonotone özelliğini izleyecektir. Tüm 2 öğeli küme alt kümeleri sıkça, üst küme sık olacaktır, aksi takdirde budanır.

6) 3 öğeli kümeyi kendisiyle birleştirerek 4 öğeli küme oluşur ve alt kümesi min_sup kriterlerini karşılamıyorsa budama gerçekleşir. Algoritma, en sık kullanılan öge kümesine ulaşıldığında durdurulur.

3.2. ML Yazılımları

ML çözümlerini tasarlamak için kullanılabilecek çok sayıda popüler yazılım mevcuttur. Bu yazılımların temel özellikleri; örüntü tanıma, sınıflandırma ve regresyon, analitik tahmin, boyut azaltma işlevi, vektör destek makinaları, Apache Spark MLlib gibi popüler makine öğrenimi kitaplıklarıyla bağlantı, önemli programlama dilleriyle çalışmak (Scala, Java, C++ , R, Python, vb.), uçtan uca açık kaynaklı makine öğrenimi şeklinde sıralanabilir (Boog, 2022).

Aşağıdaki tabloda en popüler ML yazılım araçlarının, platform, dil ve algoritma özellikleri yer almaktadır (Softwaretestinghelp, 2023b):

Tablo 3.1. En Popüler ML Yazılım Araçları

	Platform	Yazıldığı Diller	Algoritma ve Özellikler
Scikit Learn	Linux, Mac OS, Windows	Python, Cython, C, C++	Sınıflandırma Regresyon Kümeleme Ön işleme Model seçimi Boyutsal küçülme Otograd Modülü Optim Modülü nn Modülü
TensorFlow	Windows	CUDA	Veri akışı programlaması için bir kitaplık sağlar.
Weka	Linux, Mac OS,	Java	Veri Hazırlama Sınıflandırma Regresyon Kümeleme Görselleştirme Birliktelik kuralları madenciliği Büyük veri hacmi ile çalışabilir.
KNIME	Windows	Java	Eklentiler aracılığıyla metin madenciliği ve görüntü madenciliğini destekler.

Tablo 3.1. (Devamı) En Popüler ML Yazılım Araçları

Colab	Linux, Mac OS,		PyTorch, Keras, TensorFlow ve OpenCV kitaplıklarını destekler.
			Ön işlemciler
			Regresyon
Apache Mahout	Windows	Java, Scala	Kümeleme
			Recomenderler
			Dağıtılmış Doğrusal Cebir.
			Sınıflandırma
			Regresyon
			Dağıtım
Accors.Net	Linux, Mac OS,	C#	Kümeleme
			Hipotez Testleri &
			Çekirdek Yöntemleri
			Görüntü, Ses ve Sinyal & Görüş
			Regresyon
			Sınıflandırma
			Kümeleme
Shogun	Windows	C++	Vektör destek makineleri
			Boyutsal küçülme
			Çevrimiçi öğrenme vb.
Keras.io	Cloud Service	Python	Sinir ağları için API
			Veri yükleme ve Dönüşüm
Rapid Miner	Cloud Service	Java	Veri ön işleme ve görselleştirme.

Tablo 3.1’den de anlaşılaçağı üzere Python dili, ML yazılımları arasında oldukça sık kullanılmaktadır. Bu sebeple bu çalışmada da kullanılan dil Python’dır.

Pythhon ilk olarak 1991 yılında Hollandalı yazılımcı Guido van Rossum tarafından piyasaya sürülen Python’ın, 20’den fazla alt sürümü olan iki ana sürümü vardır. Genellikle web sitesi ve yazılım oluşturmak, görevleri otomatikleştirmek ve veri analizi yapmak için kullanılan bir bilgisayar programlama dilidir (Kelly ve Kelly, 2016). Python genel amaçlı bir dil olduğundan, farklı programlar oluşturmak için kullanılabilir. Bu dil, herhangi bir özel sorun için uzmanlaşmamıştır. Bu çok yönlü özelliğı sayesinde Python, günümüzün en çok kullanılan programlama dillerinden biri haline gelmiştir. Endüstri analisti firması RedMonk tarafından yürütölen bir anketin sonuçlarına göre, 2021’de geliştiriciler arasında en çok kullanılan ve tavsiye edilen ikinci programlama dili Python’dır (RedMonk, 2021). Ayrıca, Stackoverflow tarafından yapılan anketin sonuçları (Stackoverflow, 2021), Python’ın, “en çok öğrenilmesi gerekenler” listesinde birinci sırada olduğu görölmektedir.

Python, veri analistlerinin ve diğör profesyonellerin karmaşık istatistiksel hesaplamalar yapmak, veri görselleştirmeleri oluşturmak, makine öğrenimi algoritmaları oluşturmak, verileri manipöle ve analiz etmek ile verilerle ilgili diğör görevleri tamamlamak için kullanılan çok yönlü bir programdır. Ayrıca, çizgi, çubuk ve pasta grafikler, histogramlar ve üç boyutlu grafikler gibi çok farklı veri görselleştirmeleri oluşturulmasına da izin vermektedir (Khisanova, 2022). Python ayrıca, kodlayıcıların, veri analizi ve makine öğrenimi için daha hızlı ve verimli programlar yazmasını sağlayan TensorFlow ve Keras gibi kitaplıklara da sahiptir. Yazılım geliştirmede Python, derleme kontrolü, hata izleme ve test etme gibi görevlerde kullanılır. Python ile yazılım geliştiriciler, yeni ürünler veya özellikler için testleri otomatikleştirebilir (Khisanova, 2022).

Pyton; Windows, Mac, Linux, Raspberry Pi ve daha pek çok platformlarla uyumludur. Geliştiricilerin diğör programlama dillerinden daha az satır kullanmasına izin veren, İngilizce ile karşılaştırılabilir basit bir sözdizimi kullanır. Kodun hemen yürütölmesine izin veren bir tercüman sistemi üzerinde çalışır, hızlı takip edilen prototipleme özelliğine sahiptir. Buna ek olarak prosedürel, nesne yönelimli veya işlevsel bir şekilde ele alınabilir.

4. PROBLEM TANIMI

Bu bölümde ilk olarak problemin tanımına, problemin çözümü için ele alınan verinin tanıtılmasına ve depo operasyonunun mevcut durum analizi hakkında detaylı bilgi verilecektir.

4.1. Problemin Tanımı

Pandemi salgını, dünya genelinde pek çok sektörde büyük değişikliklere neden oldu. Özellikle perakende sektöründe, müşteri alışkanlıklarında büyük bir değişim yaşandı. İnsanlar, mağazalara gitmek yerine çevrimiçi alışveriş yapmayı tercih ettiler. Bu durum, perakende şirketleri için depo operasyonlarında yeni sorunlar ortaya çıkardı.

Depo operasyonlarında, müşteri taleplerine göre ürünleri toplama işlemi oldukça önemlidir. Ancak, pandemi sebebiyle müşteri alışkanlıklarındaki değişim, bu süreci zorlaştırdı. Online alışverişteki artış, müşteri siparişlerindeki yoğunluğu da beraberinde getirdi. Depolarda, ürünlerin doğru ve zamanında toplanması daha zor hale geldi. Bu durum, depo operasyonlarında toplama problemini ortaya çıkardı.

Bu problemin çözümü için, e ticaret şirketleri yeni yöntemler arayışına girdi. Teknolojik yenilikler, otomatik depo sistemleri ve verimli yönetim yaklaşımları, bu süreci kolaylaştırmak için kullanılan çözümler arasındadır.

Bu problemin çözümü için çalışmada, Türkiye merkezli bir e-ticaret platformu olan Trendyolu'un satış verileri kullanılmıştır.

Trendyol, kullanıcılara moda, giyim, ayakkabı, elektronik, ev eşyaları, kozmetik, spor ürünleri ve daha birçok kategoride geniş bir ürün yelpazesi sunar. Firmanın mobil uygulaması da mevcuttur ve kullanıcılar akıllı telefonlarından kolayca alışveriş yapabilirler.

Trendyol, hem kendi stoklarından satış yapar hem de farklı satıcılar ve markalar aracılığıyla ürünler sunar. Bu nedenle, çeşitli markaların ürünlerini Trendyol üzerinden bulabilirsiniz.

4.2. Verinin Tanıtılması

Veri, E Ticaret sektöründe faaliyet gösteren Trendyol'un 2023 yılına ait satış verilerinden oluşmaktadır. Ele alınan veriler incelendiğinde, toplam 100.004 adet ürün satışı ve 100.000 adet satırdan oluşmaktadır. Toplamda 12 adet ürün kategorisinden en çok tercih edilen ürünler TekstilA, TekstilC ve TekstilE olmuştur. Sipariş verisine ait örnek 20 satır Tablo 4.1'de gösterilmiştir. Satış verileri Tablo 4.2'de yer aldığı haliyle oluşturulmuştur. (Tablo 4.1.), (Tablo 4.2.).

Ayrıca çalışmada elde edilecek verilerin anlamlı sonuçlar üretebilmesi için veri setinde elleçleme çalışması yapılmıştır. Bu kapsamda farklı siparişler tarafından gruplanmış olan IS_EMRI1 satırı eşit oranda olacak şekilde toplamda 480 grup numarasına göre birleştirilmiştir.

Tablo 4.1. Müşteri Sipariş Verisi.

IS_EMRI 2	IS_EMRI 1	GRUP_ NR1	URUN_ KATEGORISI	LOKASYON	MIKTAR
GrupNo0	IS5518946	2565366204	TekstilA	N012	1
GrupNo0	IS5522902	2565379406	TekstilC	N002	1
GrupNo0	IS5517555	2565403525	TekstilA	K03	1
GrupNo0	IS5517291	2565445792	TekstilE	K12	1
GrupNo0	IS5523811	2565446651	TekstilA	M01	1
GrupNo0	IS5518935	2565470433	TekstilE	K04	1
GrupNo0	IS5519213	2565522376	TekstilD	K10	1
GrupNo0	IS5518423	2565528575	TekstilB	N021	1
GrupNo0	IS5518430	2565567161	TekstilD	K07	1
GrupNo0	IS5518459	2565635029	TekstilA	K04	1
GrupNo0	IS5518421	2565676087	TekstilE	K12	1
GrupNo0	IS5518465	2565704644	TekstilD	N011	1
GrupNo0	IS5518628	2565719867	TekstilA	N001	1
GrupNo0	IS5519281	2565740825	TekstilA	K03	1
GrupNo0	IS5518393	2565791718	TekstilD	N012	1
GrupNo0	IS5518600	2565835749	TekstilA	M03	1
GrupNo0	IS5518580	2565844861	TekstilC	K10	1
GrupNo0	IS5518476	2565846100	Tekstil	K02	1
GrupNo0	IS5518406	2565859778	TekstilC	K08	1
GrupNo0	IS5518949	2565880555	TekstilA	M05	1
...	...	...	...	...	...

Çalışmada ele alınan veride, 32.513 adet müşteri siparişi baz alınmıştır. Sipariş verileri toplamda 12 adet ürün kategorisinden oluşmaktadır. Bir sipariş tekil bir üründen oluşuyorsa tekli sipariş olarak adlandırılmaktadır. Eğer bir sipariş birden fazla üründen oluşuyorsa çoklu sipariş olarak adlandırılmaktadır.

Tablo 4.2. Verinin Tanımlanması.

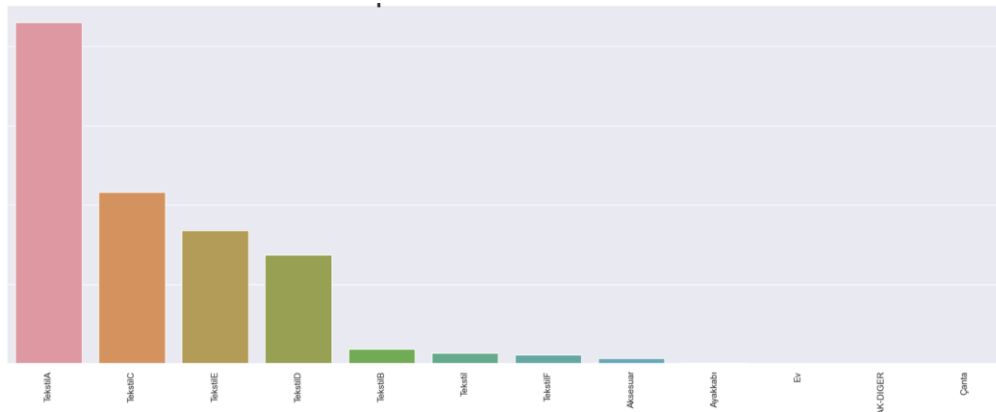
Açıklama	Miktar
MIKTAR	Siparişte satılan ürün miktarı
LOKASYON	Siparişte satılan ürünün depo lokasyonundaki yeri
URUN_KATEGORISI	Siparişte satılan ürünün kategorisi
GRUP_NR1	Sipariş numarası
IS_EMRI 1	Farklı siparişlerin gruplandığı grup numarası
IS_EMRI 2	Gruplanan siparişlerin tekrar gruplandığı grup numarası

Aşağıdaki tabloda toplam sipariş sayısı, toplam ürün sayısı, tekli sipariş sayısı, tekli siparişlerde en çok satın alınan ürün kategorisi, çoklu sipariş sayısı, çoklu sipariş başı ortalama ürün sayısı ve genel sipariş başı ortalama ürün sayısı bilgilerine yer verilmiştir (Tablo 4.3.)

Tablo 4.3. Sipariş Verisi Özet Tablo.

Açıklama	Miktar
Toplam Sipariş Sayısı	32.513 Sipariş
Toplam Ürün Sayısı	100.004 Adet
Tekli Sipariş Sayısı	32.510 Sipariş
Tekli Siparişlerde En Çok Satılan Ürün Kategorisi	TekstilA
Çoklu Sipariş Sayısı	3 Sipariş
Çoklu Sipariş Başı Ortalama Ürün Sayısı	2.33 Adet
Genel Sipariş Başı Ortalama Ürün Sayısı	1,00 Adet

Aşağıdaki şekilde ise en çok tercih edilen 10 ürün kategorisine yer verilmiştir. 42.986 adet ile TekstilA birinci Sırada yer alırken TekstilC 21.626 adet ile ikinci Sıradadır. Üçüncü sırada ise 16.768 adet ile TekstilE yer almaktadır. (Şekil 4.1)

**Şekil 4.1.** En Çok Tercih Edilen Kategoriler.

4.3. Mevcut Durum Analizi

Firmanın depo operasyonu geleneksel yöntemler yürütülmektedir. Firmanın depo operasyonu, gelen siparişleri karşılamak ve müşterilere hızlı bir şekilde teslim etmek için tasarlanmış bir sistemdir. Firma online siparişler için yürüttüğü depo operasyonunun aşamaları 8 aşamadan oluşmaktadır;

Ürün alımı: E-ticaret depo operasyonları, ürün alımıyla başlar. Ürünler tedarikçilerden alınır ve depoya getirilir.

Stok yönetimi: Ürünler depoya geldikten sonra, stok yönetimi yapılır. Bu aşamada, ürünlerin sayısı, tarihleri, fiyatları vb. kaydedilir.

Depolama: Stok yönetimi yapıldıktan sonra, ürünler depoya yerleştirilir. Depolama işlemi, ürünlerin kolaylıkla bulunabilmesi ve müşterilere hızlı bir şekilde gönderilebilmesi için önemlidir.

Sipariş toplama: E-ticaret depo operasyonlarında, müşterilerin verdiği siparişlerin toplanması aşaması oldukça önemlidir. Bu aşamada, depodaki ürünler müşteri siparişine göre toplanır.

Ürün Dağıtım: Müşteri siparişleri, sipariş verildiği tarih ve lokasyonları baz alınır gruplanmaktadır. Depo personelleri gruplanan siparişleri toplamaktadır. Bu sebeple bu siparişlerin müşteri bazında dağıtılması gerekmektedir.

Paketleme: Sipariş toplama işlemi tamamlandıktan sonra, ürünler müşterilere gönderilmek üzere paketlenir. Paketleme işlemi, ürünlerin güvenli bir şekilde taşınmasını sağlar.

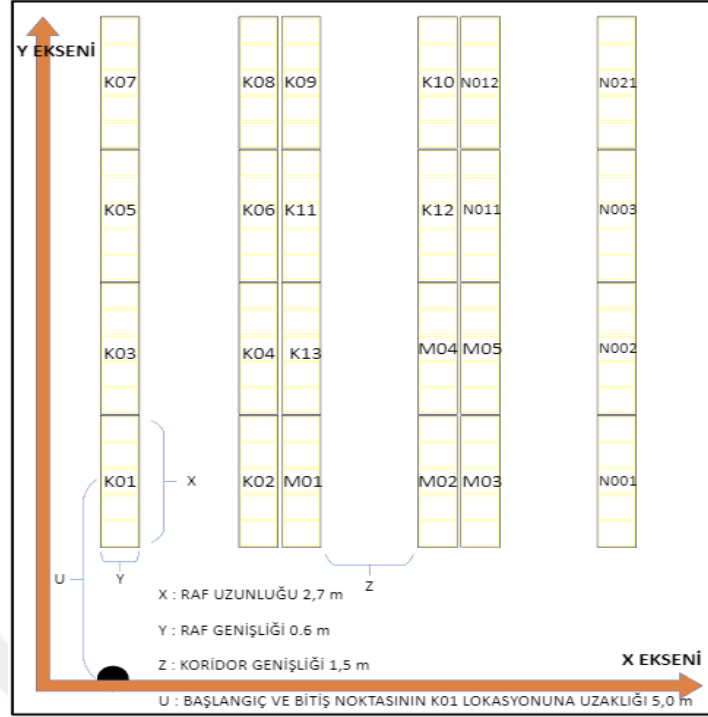
Kargoya verme: Ürünler, paketlenip hazırlandıktan sonra kargo şirketine teslim edilir.

Kargo şirketi, müşterilere ürünlerin teslimatını gerçekleştirir.

İade yönetimi: E-ticaret depo operasyonlarında iade yönetimi de önemli bir aşamadır.

Müşterilerin iade talepleri depoda işlenir ve ürünler geri alınır.

Firmanın bu operasyonu gerçekleştirmek için sahip olduğu depo yerleşimi Şekil 4.2’de olduğu gibi varsayılmıştır. (Şekil 4.2)

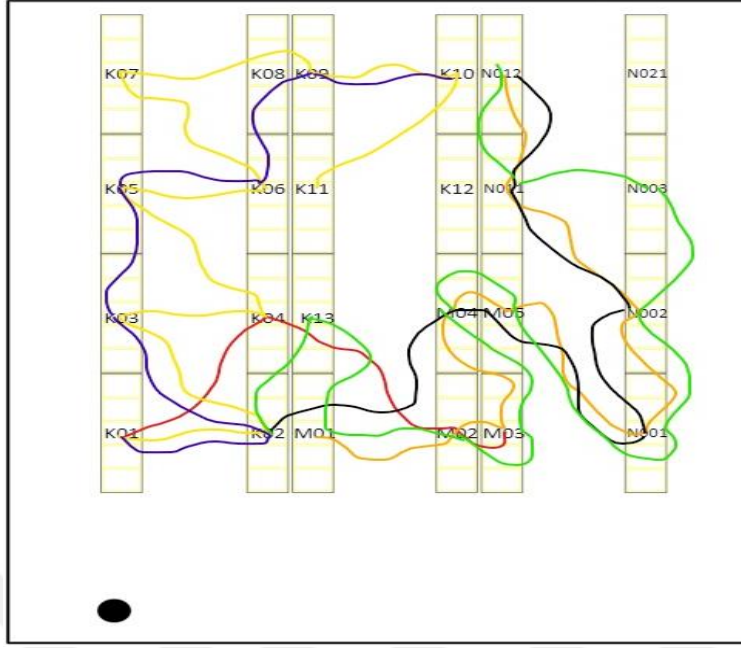


Şekil 4.2. Depo Yerleşimi.

Varsayılan depo yerleşimi 3 koridor, 4 sıra ve yaklaşık 130 m²'den oluşmaktadır. Toplamda 24 lokasyondan oluşan yerleşimde market tipi raf sistemi kullanılmaktadır. Market tipi raf sisteminde raf uzunluğu 2,7 m olarak tasarlanmış ve görselde X harfi ile ifade edilmiştir. Bu rafların genişliği 0,6 m'dir. İki raf arası mesafe koridor geniş olarak ifade edilmektedir ve bu mesafe 1,5 m olarak tasarlanmıştır. Siparişlerin toplanması için depo personelinin başlangıç noktası olarak belirlenen noktanın K01 lokasyonuna olan uzaklığı ise U harfi ile ifade edilip 5,0 m'dir.

Çalışmada mevcut durum ortaya çıkarılırken ilk olarak firmanın mevcut siparişleri toplaması için gerekli eforun belirlenmesi gerekmektedir. Depo yazılım sisteminde müşteri siparişleri, sipariş edilen ürünün tarih zaman ve saat kısıtlarına göre gruplanmaktadır. Depo personelleri siparişleri gruplanan numaralar üzerinden toplanmaktadır.

Şekil 4.3 farklı müşteri siparişlerden gruplanmış ürünlerin toplama rotası görsel olarak gösterilmektedir. Çalışmada ele alınan veride yer alan GrupNo127 nolu grup kırmızı renkle, GrupNo207 nolu grup sarı renkle, GrupNo27 nolu grup mavi renkle, GrupNo389 nolu grup turuncu renkle, GrupNo463 nolu grup siyah renkle ve GrupNo257 nolu grup yeşil renkle ifade edilmiştir (Şekil 4.3).



Şekil 4.3. Sipariş Grubu Toplama Rotası.

Tablo 4.3’de ise Şekil 4.4’de gösterilen sipariş gruplarında yer alan ürün kategorilerinin toplanması için raf yerleşiminde mevcut durumda yürütülmesi gereken mesafelere yer verilmiştir. Grup numaraları için yürütülen mesafenin hesaplanması Şekil 4.2’de belirtilen ölçülere göre hesaplanmıştır. (Şekil 4.2)

Tablo 4.4. Sipariş Grubu Yürüme Mesafesi

Grup No	Yürüme Mesafesi
GrupNo127	19,3 m
GrupNo207	27,9 m
GrupNo27	21,9 m
GrupNo389	29,6 m
GrupNo257	24,1 m
GrupNo463	34,5 m

TekstilA, TekstilC ve TekstilE’nin toplam ürün kategorindeki oranı %25 iken toplam sipariş edilen ürün sayıları arasında %80’dir.

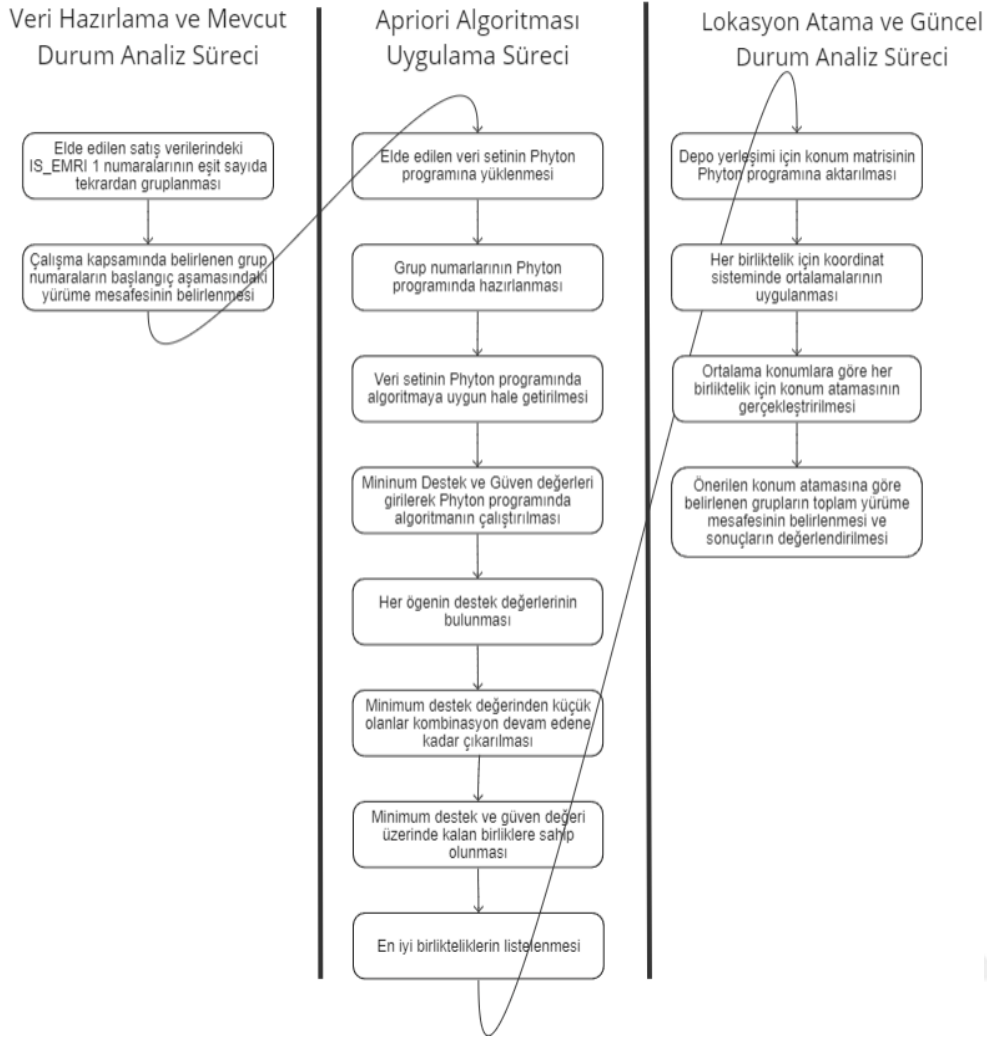
Bu kategoriler 200 adet üzeri olan en çok tercih edildiđi ürün grupları oluđu için çalışmada öncelik verilmiştir.

Firmanın depo yapısını incelediğimizde düzenli bir yapı görülsede tedarikçilerden satışa sunulacak ürünler geldikçe ürünlerin hangi raf lokasyonuna adresleneceđi öneri sistemi yer almadığından bir ürün çeşidinin birden fazla lokasyonda olduđu karmaşık bir yapıya sahip olmuştur. Yeni ve doğru bir depo yerleşimi gerektiđi aşikârdır. Bu çalışmada depo raflarında yer alan ürünlerin doğru raf lokasyonlarında konumlandırılması yapılırken işletmenin amacı birliktelik analizine göre oluşturmaktır. Böylelikle işgücünü azaltarak firmanın daha az maliyete ulaşması hedeflenmektedir. İlave olarak Müşteri ihtiyacına daha hızlı bir karşılık verilmesiyle müşteri memnuniyeti artırılması, müşteri memnuniyetinin artması sonucu satış potansiyelin artması, sürekli deđişebilecek müşteri alışkanlıklarına cevap verebilecek esnekliğe sahip olunması hedeflenmektedir.



5. UYGULAMA

Bu bölümde çalışmaya konu olan problemin uygulanması ve bulguların ortaya çıkarılması yer almaktadır. Uygulamada, firmanın deposunda stoklanan ürünler için yeni bir raf yerleşim çalışması amaçlanmaktadır. Mevcut durum bölümünde gruplanan siparişlerin toplama yolunu incelediğimizde karmaşıklıklar olduğunu gözlemlemiştik. Mevcut durumda depolanacak ürünlerin raflara yerleşimi bir mantığa göre değil gelişigüzel ve boş lokasyona bağlı olarak oluşturulduğu görülmüştür. Bu çalışmada, ele alınan verilere ilk olarak Python programlama dilinde Apriori algoritması uygulanmıştır. Bu algorithmada, gruplanan müşteri siparişlerinin ürün kategorileri arasındaki birliktelik ilişkisi analiz edilmiştir. Daha sonra analiz sonucu birbiri arasındaki ilişki yoğunluğu fazla çıkan ürün kategorilerinin, depo raf yerleşiminde daha yakın lokasyonlarda adreslenebilmesi için algoritma tasarlanmış ve Python programlama dilinde uygulanmıştır. Bu algoritma ile ürün kategorisinin adreslenmesi için lokasyon öneri sistemi geliştirilmiştir. Bu çalışmada ürün kategorileri için önerilen lokasyon ile müşteri siparişlerinin daha az mesafe yürütülerek toplanması ve böylelikle firmanın daha verimli depo operasyonuna sahip olması hedeflenmiştir. Ayrıca uygulamaya ait genel süreç akışı da Şekil 5.1 'de belirtilmiştir.



Şekil 5.1. Uygulama Genel Süreç Akışı

5.1. Algoritmanın Uygulanması

Uygulamada, depo içerisinde birliktelik kurallarına ulaşarak birlikte daha fazla satılan ürünlerin tespit edilmesi amaçlanmaktadır. Birlikte sipariş edilme olasılığı yüksek olan ürünler arasındaki ilişkiler yüksek olacağından depo içerisine ürünler yerleştirilirken bu ilişkilerden faydalanılmıştır. Çalışmaya konu olan problemin çözümüne apriori algoritması yöntemiyle çözüm aranmıştır. Uygulamada yer alan bilgiler firmanın e-ticaret satış verilerinden oluşmaktadır.

Apriori algoritmasının uygulanmasında Phyton programından faydalanılmıştır. Aşağıda algoritmanın bu programda uygulama aşamalarına yer verilmiştir;

```
# Loading row data
row = pd.read_excel('row.xlsx')
row.head()
```

	IS_EMRI 2	IS_EMRI 1	GRUP_NRI	Barkod	URUN_KATEGORISI	LOKASYON	MIKTAR
0	GrupNo0	IS5518946	2565366204	8683351135595	TekstilA	N012	1
1	GrupNo0	IS5522902	2565379406	8683048168332	TekstilC	N002	1
2	GrupNo0	IS5517555	2565403525	8682164928905	TekstilA	K03	1
3	GrupNo0	IS5517291	2565445792	8683047301839	TekstilE	K12	1
4	GrupNo0	IS5523811	2565446651	8683048192009	TekstilA	M01	1

Şekil 5.2. Veri Setinin Uygulamaya Yüklenmesi.

Şekil 5.2’de uygulamada kullanılan veri setinin program tarafından okutulması sağlanmıştır. Row isimli dosyada yer alan veri seti okutulmuş ve ilk dört satırı örnek olarak gösterilmiştir.

```
# Filling N/A with 0
base_data.fillna(0,inplace=True)
base_data.info()

# One Hot Encoding to have products on columns
one_hot_encoded_data = pd.get_dummies(base_data, columns = [''])
one_hot_encoded_data.head()

# We need to group rows by their 'GRUP_NRI' value
sum_data = one_hot_encoded_data.groupby(['group']).sum()
sum_data.head()
```

	_0	_Aksesuar	_Ayakkabı	_Ev	_OYUNCAK-DIGER	_Tekstil	_TekstilA	_TekstilB	_TekstilC	_TekstilD	_TekstilE	_TekstilF	_Çanta
group													
GrupNo0	0	1	0	0	0	1	63	4	37	29	29	3	0
GrupNo1	0	1	0	0	0	6	92	7	43	39	39	3	0
GrupNo10	0	4	0	0	0	2	86	3	46	30	38	0	0
GrupNo100	0	0	0	0	0	3	71	9	40	25	38	2	0
GrupNo101	0	1	0	0	0	2	86	0	38	34	42	1	0

Şekil 5.3. Veri Setinde Grup No Hazırlaması.

Apriori algoritmasında genel olarak birliktelik ilişkisi incelenirken müşteri sipariş sepetleri analiz edilmektedir. Ancak bu çalışmada bunun aksine WMS siparişlerinin birleştirildiği grup numalarına, müşteri siparişleri olarak davranılmıştır. Bunun nedeni ise depo personellerinin sipariş toplama sürecinde depo yerleşiminde ürünleri toplarken yürüdüğü mesafe grup numaralarında yer alan ürün kategorilerine göre şekillenmesidir. Bu yüzden algoritma programda uygulanması için veri setinin Şekil 5.3’de olduğu gibi grup numarası bazında birleştirilerek bu gruplarda yer alan ürün kategorileri özet tablo haline getirilmiştir.

```
# In apriori algorithm, we need the information that a product is included or not included, that means True or False
for column_name in sum_data.columns:
    sum_data[column_name] = sum_data[column_name].astype('bool')
sum_data.head()
```

	_0	_Aksesuar	_Ayakkabi	_Ev	_OYUNCAK-DIGER	_Tekstil	_TekstilA	_TekstilB	_TekstilC	_TekstilD	_TekstilE	_TekstilF	_Çanta
group													
GrupNo0	False	True	False	False	False	True	True	True	True	True	True	True	False
GrupNo1	False	True	False	False	False	True	True	True	True	True	True	True	False
GrupNo10	False	True	False	False	False	True	True	True	True	True	True	False	False
GrupNo100	False	False	False	False	False	True	True	True	True	True	True	True	False
GrupNo101	False	True	False	False	False	True	True	False	True	True	True	True	False

Şekil 5.4. Veri Setinin Hazırlanması.

Apriori algoritmasında bir sipariş sepetinde yer alan ürünün miktarından çok sepette yer alan ürünlerin kategorisi önemlidir. Bu yüzden algoritma analiz edilirken uygulanan veri setinde ürün o siparişte yer alıyorsa “True” veya “1” , yer almıyorsa “False” veya “0” olarak tanımlanmaktadır. Şekil 5.4’de çalışmada kullanılan veri seti, programda bu kurala uygun hale getirilmiştir. (Şekil 5.4).

Apriori algoritmasının tanımlandığı bölümde Güven, Destek ölçütlerin bu algoritmada birliktelik ilişkisi oluşturulurken kullanıldığı belirtilmiştir. Destek değeri bir ürünün toplam satış içerisindeki oranı ifade ederken, bu oranın en yüksek olduğu ürün kategorisi toplam satışlar içerisindeki satış miktarının en çok olduğunu ifade etmektedir. Güven değeri beraber satış yapılan ürün kategorilerinin birlikte satın alınma olasılığını ifade eder. Bu oranın en yüksek değeri bulunduğu ilişkiler ise tüm satışlar arasında birlikte en çok satın alınan ürünler olduğunu ifade etmektedir. Bu çalışmada algoritmaya uygun olarak min. destek ve güven değerleri belirlenmiş ve uygulamaya alınmıştır. Algoritma destek değeri min 0,1 güven değeri ise min 0,4 alınarak Python programında Şekil 5.5.’de görüldüğü üzere uygulanmıştır. Apriori algoritmasının veri setindeki sonuçları bir sonraki bölümde aktarılacaktır. (Şekil 5.5.)

```
## Let's go through APRIORI
frequent_itemsets = apriori(sum_data, min_support=0.1, use_colnames=True)
frequent_itemsets['length'] = frequent_itemsets['itemsets'].apply(lambda x: len(x))
frequent_itemsets

# now making the rules from frequent itemset generated above
rules = association_rules(frequent_itemsets, metric = "lift", min_threshold = 1)
rules.sort_values('confidence', ascending = False, inplace = True)
rules

particular_rules = rules[(rules['confidence'] > 0.4) &
                        (frequent_itemsets['support'] >= 0.1) ]
```

Şekil 5.5. Algoritmanın Programda Uygulanması.

5.2. Algoritmanın Sonuçlarının Değerlendirilmesi

Phyton programı ile apriori algoritması uygulanmış olup, bu bölümde sonuçların değerlendirilmesi gerçekleştirilecektir. Uygulamada ilk olarak güven ve destek değerleri kullanarak veri setinde yer alan ürün kategorileri arasında ilişki ortaya konulmuştur.

Tablo 5.1’de Şekil 8.’da yer alan kodun çıktısını görebiliriz. Tablo 5.1’de destek değeri en büyük olan 20 satıra yer verilmiştir. Tabloyu incelediğimizde TekstilA, TekstilC, TekstilD ve TekstilE kategorilerinin tüm siparişler arasında en çok tercih edilen ürünler olduğunu göstermektedir. Bu kategoriler ürün setinde tekil olarak yer almaktadır. Aynı ürünler birlikte satın alınma olasılığında aynı şekilde en yüksek olan ürün kategorileri olduğu Tablo 5.1’de görülmektedir. (Tablo 5.1).

Tablo 5.1. Destek Değerleri.

No	Support	Itemsets	Lenght
27	1	frozenset({'_TekstilC', '_TekstilA'})	2
28	1	frozenset({'_TekstilD', '_TekstilA'})	2
29	1	frozenset({'_TekstilE', '_TekstilA'})	2
35	1	frozenset({'_TekstilD', '_TekstilC'})	2
36	1	frozenset({'_TekstilE', '_TekstilC'})	2
38	1	frozenset({'_TekstilE', '_TekstilD'})	2
87	1	frozenset({'_TekstilD', '_TekstilC', '_TekstilA'})	3
88	1	frozenset({'_TekstilE', '_TekstilC', '_TekstilA'})	3
90	1	frozenset({'_TekstilE', '_TekstilD', '_TekstilA'})	3
99	1	frozenset({'_TekstilE', '_TekstilD', '_TekstilC'})	3
168	1	frozenset({'_TekstilE', '_TekstilD', '_TekstilC', '_TekstilA'})	4
26	0.975051975	frozenset({'_TekstilB', '_TekstilA'})	2
31	0.975051975	frozenset({'_TekstilB', '_TekstilC'})	2
32	0.975051975	frozenset({'_TekstilB', '_TekstilD'})	2
33	0.975051975	frozenset({'_TekstilB', '_TekstilE'})	2
83	0.975051975	frozenset({'_TekstilB', '_TekstilC', '_TekstilA'})	3
84	0.975051975	frozenset({'_TekstilB', '_TekstilD', '_TekstilA'})	3
85	0.975051975	frozenset({'_TekstilB', '_TekstilE', '_TekstilA'})	3
93	0.975051975	frozenset({'_TekstilD', '_TekstilB', '_TekstilC'})	3
94	0.975051975	frozenset({'_TekstilE', '_TekstilB', '_TekstilC'})	3
96	0.975051975	frozenset({'_TekstilE', '_TekstilB', '_TekstilD'})	3
...	...	...	...

Tablo 5.1’de çalışmada uygulanan birliktelik analizinde en yüksek destek değerini 100 % oranına sahiptir. Tablo 5.2’de ise çalışmada uygulana algortmada en yüksek 20 güven değerlerine yer verilmiştir. 100% oranı ile en yüksek güven değerine sahip

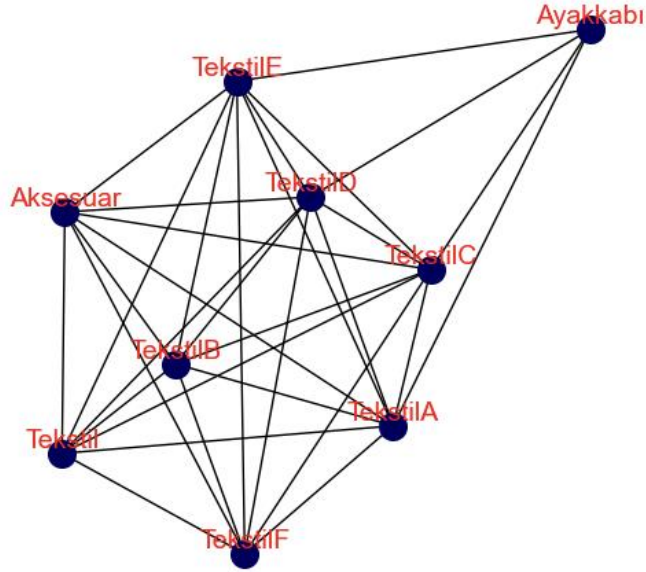
birliktelikler TekstilA, TekstilB, TekstilC TekstilD ve TekstilE ürünleri arasındaki ilişkilerdir. (Tablo 5.1), (Tablo 5.2).

Tablo 5.2 Güven Değerleri.

No	Antecedents	Consequents	Support	Confidence
46	frozenset({'_TekstilD'})	frozenset({'_TekstilC'})	1.000	1.000
47	frozenset({'_TekstilC'})	frozenset({'_TekstilD'})	1.000	1.000
48	frozenset({'_TekstilE'})	frozenset({'_TekstilC'})	1.000	1.000
49	frozenset({'_TekstilC'})	frozenset({'_TekstilE'})	1.000	1.000
52	frozenset({'_TekstilE'})	frozenset({'_TekstilD'})	1.000	1.000
37	frozenset({'_TekstilA'})	frozenset({'_TekstilE'})	1.000	1.000
36	frozenset({'_TekstilE'})	frozenset({'_TekstilA'})	1.000	1.000
35	frozenset({'_TekstilA'})	frozenset({'_TekstilD'})	1.000	1.000
34	frozenset({'_TekstilD'})	frozenset({'_TekstilA'})	1.000	1.000
53	frozenset({'_TekstilD'})	frozenset({'_TekstilE'})	1.000	1.000
33	frozenset({'_TekstilA'})	frozenset({'_TekstilC'})	1.000	1.000
32	frozenset({'_TekstilC'})	frozenset({'_TekstilA'})	1.000	1.000
270	frozenset({'_TekstilB', '_TekstilC'})	frozenset({'_TekstilA'})	0.975	1.000
44	frozenset({'_TekstilB'})	frozenset({'_TekstilE'})	0.975	1.000
42	frozenset({'_TekstilB'})	frozenset({'_TekstilD'})	0.975	1.000
40	frozenset({'_TekstilB'})	frozenset({'_TekstilC'})	0.975	1.000
30	frozenset({'_TekstilB'})	frozenset({'_TekstilA'})	0.975	1.000
45	frozenset({'_TekstilE'})	frozenset({'_TekstilB'})	0.975	0.975
43	frozenset({'_TekstilD'})	frozenset({'_TekstilB'})	0.975	0.975
41	frozenset({'_TekstilC'})	frozenset({'_TekstilB'})	0.975	0.975
31	frozenset({'_TekstilA'})	frozenset({'_TekstilB'})	0.975	0.975
...	...	...	...	...

Şekil 5.6'da yer alan grafikte ise algoritmanın uygulama sonucu örümcek ağı grafiğinde gösterilmektedir. Burada ürün kategorileri aralarındaki ilişki birlikte bulundukları kural sayısı baz alınarak uzaklık veya yakınlığa göre ifade edilmektedir. Belirtilen çizgi ürünlerin birlikte ilişkisi yer aldığını göstermektedir. En çok bir arada

alınan ürün kategorileri birbirine olan mesafe en yakın mesafede olanların olduğu görülmektedir. (Şekil 5.6).



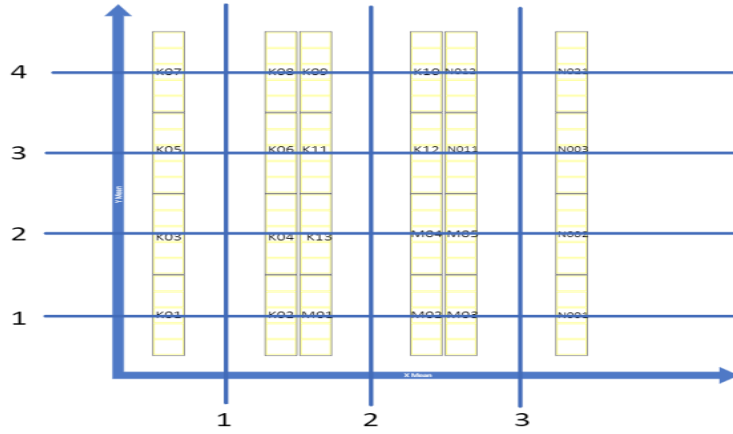
Şekil 5.6. Ürün Kategorileri Arasındaki İlişki Grafiği.

5.3. Önerilen Yöntem ve Uygulanması

Bir önceki bölümde firmanın veri setinde yer alan siparişlerin apriori algoritması ile birliktelik ilişkisi analiz edilmiş olup, hangi ürün kategorilerinin birlikte en çok tercih edilen ikililer olduğu ortaya konulmuştur. Bu çalışmada elde edilen sonuçlar ile ürünlerin depo yerleşiminde daha doğru lokasyonlara adreslenmesi amaçlanmaktadır.

Apriori Algoritması uygulanması sonrası birbiri ile olan ilişkisi yüksek için ürün kategorileri için lokasyon öneri sistemi geliştirilmiştir. Lokasyon öneri sistemi geliştirilirken apriori algoritması sonuçlarında güven değeri yüksek olan ürün kategorileri baz alınmıştır. Lokasyon arası uzaklıklar öneri sistemine dahil edilirken 4.3. bölümde belirtilen depo yerleşimindeki uzaklıklar varsayılmıştır. Önerilen sistem python programında uygulama alınmıştır.

Önerilen yöntemin uygulanmasında ilk olarak depo yerleşiminde ilk olarak lokasyonların kartezyen sisteminde konumlandırılması sağlanmıştır. Şekil 5.7’de görüldüğü üzere depoda bulunan her bir lokasyon x ve y eksenlerinde karşılık gelen bir konuma sahiptir. Örneğin K01 ve K02 lokasyonlarının konumu [1 , 1] olarak söylenilebilir.(Şekil 5.7)



Şekil 5.7. Depo Yerleşimi Konum Matrisi.

Şekil 5.7’de belirlenen lokasyonların kartez sistemindeki konumları python programında tanıtilarak uygulamaya başlanmıştır. Buna göre Şekil 5.8’de bazı lokasyonlardan örnek gösterilmiştir. Örneğin M04 [2 , 2] konumunda olduğunu görebiliriz. (Şekil 5.7), (Şekil 5.8).

```
### Locating products on a cartesian system
# First we add x and y columns to hold their location
row_data['x_axis'] = 1
row_data['y_axis'] = 1
# second, we should change their x and y by their 'Lokasyon' data
# 1,1
row_data.loc[row_data.LOKASYON == 'K01', ['x_axis', 'y_axis']] = 1, 1
row_data.loc[row_data.LOKASYON == 'K02', ['x_axis', 'y_axis']] = 1, 1
# 1,2
row_data.loc[row_data.LOKASYON == 'K03', ['x_axis', 'y_axis']] = 1, 2
row_data.loc[row_data.LOKASYON == 'K04', ['x_axis', 'y_axis']] = 1, 2
# 1,3
row_data.loc[row_data.LOKASYON == 'K05', ['x_axis', 'y_axis']] = 1, 3
row_data.loc[row_data.LOKASYON == 'K06', ['x_axis', 'y_axis']] = 1, 3
# 1,4
row_data.loc[row_data.LOKASYON == 'K07', ['x_axis', 'y_axis']] = 1, 4
row_data.loc[row_data.LOKASYON == 'K08', ['x_axis', 'y_axis']] = 1, 4
# 2,1
row_data.loc[row_data.LOKASYON == 'M01', ['x_axis', 'y_axis']] = 2, 1
row_data.loc[row_data.LOKASYON == 'M02', ['x_axis', 'y_axis']] = 2, 1
# 2,2
row_data.loc[row_data.LOKASYON == 'K13', ['x_axis', 'y_axis']] = 2, 2
row_data.loc[row_data.LOKASYON == 'M04', ['x_axis', 'y_axis']] = 2, 2
# 2,3
row_data.loc[row_data.LOKASYON == 'K11', ['x_axis', 'y_axis']] = 2, 3
row_data.loc[row_data.LOKASYON == 'K12', ['x_axis', 'y_axis']] = 2, 3
# 2,4
row_data.loc[row_data.LOKASYON == 'K09', ['x_axis', 'y_axis']] = 2, 4
row_data.loc[row_data.LOKASYON == 'K10', ['x_axis', 'y_axis']] = 2, 4
# 3,1
row_data.loc[row_data.LOKASYON == 'M03', ['x_axis', 'y_axis']] = 3, 1
row_data.loc[row_data.LOKASYON == 'N001', ['x_axis', 'y_axis']] = 3, 1
# 3,2
row_data.loc[row_data.LOKASYON == 'M05', ['x_axis', 'y_axis']] = 3, 2
row_data.loc[row_data.LOKASYON == 'N002', ['x_axis', 'y_axis']] = 3, 2
# 3,3
row_data.loc[row_data.LOKASYON == 'N011', ['x_axis', 'y_axis']] = 3, 3
row_data.loc[row_data.LOKASYON == 'N003', ['x_axis', 'y_axis']] = 3, 3
# 3,4
row_data.loc[row_data.LOKASYON == 'N012', ['x_axis', 'y_axis']] = 3, 4
row_data.loc[row_data.LOKASYON == 'N021', ['x_axis', 'y_axis']] = 3, 4
row_data.tail()
```

	IS_EMRI 2	IS_EMRI 1	GRUP_NRI	Barkod	URUN_KATEGORISI	LOKASYON	MIKTAR	x_axis	y_axis
99995	GrupNo99	IS5525294	2571761962	8683048219874	TekstilA	K06	1	1	3
99996	GrupNo99	IS5525203	2571813721	8683043176073	Tekstil	K05	1	1	3
99997	GrupNo99	IS5525445	2571829912	8682165375524	TekstilA	K02	1	1	1
99998	GrupNo99	IS5525884	2571899123	270858032485	TekstilE	K07	1	1	4
99999	GrupNo99	IS5526603	2572303370	8683045806473	TekstilA	K01	1	1	1

Şekil 5.8. Lokasyonların Konumlandırılması.

Daha sonra algoritma sonucu elde edilen Tablo 5.2.'daki her bir satır için ürün kategorilerinin koordinat düzlemindeki konumları hesaplanarak ortalama bir x ve y değeri hesaplanır. Örneğin, Tablo 5.3'da yer alan 47 nolu satırda TekstilC ve TekstilD alan müşterilerin siparişlerinin bulunduğu grupta bu üç kategorinin koordinat sistemindeki konumlarının x ve y düzlemlerindeki ortalama değerleri [3 , 2] olarak belirtilmiştir. Şekil 5.9'da ise bu hesaplamaların Python programında uygulamasına yer verilmiştir. (Tablo 5.2), (Tablo 5.3), (Şekil 5.9)

Tablo 5.3. Koordinat Düzlemindeki Ortalama Konumlar

No	Antecedents	Consequents	Support	Confidence	X	Y
47	frozenset({'_TekstilC'})	frozenset({'_TekstilD'})	1.000	1.000	3	2
48	frozenset({'_TekstilE'})	frozenset({'_TekstilC'})	1.000	1.000	1	3
49	frozenset({'_TekstilC'})	frozenset({'_TekstilE'})	1.000	1.000	3	2
52	frozenset({'_TekstilE'})	frozenset({'_TekstilD'})	1.000	1.000	1	4
37	frozenset({'_TekstilA'})	frozenset({'_TekstilE'})	1.000	1.000	1	1
36	frozenset({'_TekstilE'})	frozenset({'_TekstilA'})	1.000	1.000	1	3
35	frozenset({'_TekstilA'})	frozenset({'_TekstilD'})	1.000	1.000	1	4
34	frozenset({'_TekstilD'})	frozenset({'_TekstilA'})	1.000	1.000	3	4
53	frozenset({'_TekstilD'})	frozenset({'_TekstilE'})	1.000	1.000	3	3
33	frozenset({'_TekstilA'})	frozenset({'_TekstilC'})	1.000	1.000	3	2
32	frozenset({'_TekstilC'})	frozenset({'_TekstilA'})	1.000	1.000	3	4
40	frozenset({'_TekstilB'})	frozenset({'_TekstilC'})	0.975	1.000	3	4
30	frozenset({'_TekstilB'})	frozenset({'_TekstilA'})	0.975	1.000	3	4
31	frozenset({'_TekstilA'})	frozenset({'_TekstilB'})	0.975	0.975	3	4
256	frozenset({'_Tekstil'})	frozenset({'_TekstilD', '_TekstilE'})	0.933	1.000	1	4
203	frozenset({'_Tekstil'})	frozenset({'_TekstilE', '_TekstilA'})	0.933	1.000	3	4
200	frozenset({'_TekstilE', '_Tekstil'})	frozenset({'_TekstilA'})	0.933	1.000	3	1
197	frozenset({'_Tekstil'})	frozenset({'_TekstilD', '_TekstilA'})	0.933	1.000	3	4
24	frozenset({'_Tekstil'})	frozenset({'_TekstilD'})	0.933	1.000	3	1
26	frozenset({'_Tekstil'})	frozenset({'_TekstilE'})	0.933	1.000	3	4
22	frozenset({'_Tekstil'})	frozenset({'_TekstilC'})	0.933	1.000	1	1
...	...	...	...	...	...	...

```

x_mean_list = []
y_mean_list = []
recommend_list = []
for row in particular_rules.iterrows():
    # antecedents print(row[1][0]) -- list(row[1][0])
    # consequents print(row[1][1]) -- list(row[1][1])
    x_locs = 0
    y_locs = 0
    for item_str in list(row[1][0]):
        # item string print(item_str[1:])
        item_string = item_str[1:]
        item_x = row_data.loc[row_data['URUN_KATEGORISI'] == item_string, 'x_axis'].iloc[0]
        x_locs = x_locs + item_x
        item_y = row_data.loc[row_data['URUN_KATEGORISI'] == item_string, 'y_axis'].iloc[0]
        y_locs = y_locs + item_y
    x_mean = x_locs/len(list(row[1][0]))
    y_mean = y_locs/len(list(row[1][0]))
    x_mean_list.append(x_mean)
    y_mean_list.append(y_mean)
    recommend_label = recommend_function(x_mean,y_mean)
    recommend_list.append(recommend_label)

particular_rules['x_mean'] = x_mean_list
particular_rules['y_mean'] = y_mean_list
particular_rules['recommend'] = recommend_list

```

Şekil 5.9. Pyhton'da Koordinat Ortalamalarının Uygulanması.

Lokasyon öneri sisteminin çalışması için depo yerleşiminde yer alan her bir lokasyona karşılık gelen kartezyen sisteminde koordinat ataması yapılması gerekmektedir. Şekil 5.10'da her bir koordinat için depo yerleşiminde yer alan lokasyonun atamasının Phyton programında yapıldığını görebiliriz. Son aşamada ise apriori algoritmasının analizi sonucu ortaya çıkan birlikteliklerin Tablo 5.3'de hesaplanan ortalama konum koordinatına göre lokasyon atamalarının gerçekleştirilmiştir. (Tablo 5.3), (Şekil 5.10).

```

def recommend_function(x_mean,y_mean):
    x_r = round(x_mean) #x_rounded
    y_r = round(y_mean) #y_rounded
    recommend_label = ''
    # 1,1
    if x_r == 1 and y_r == 1:
        recommend_label = 'K01'
        #recommend_label = 'K02'
    # 1,2
    elif x_r == 1 and y_r == 2:
        recommend_label = 'K03'
        #recommend_label = 'K04'
    # 1,3
    elif x_r == 1 and y_r == 3:
        recommend_label = 'K05'
        #recommend_label = 'K06'
    # 1,4
    elif x_r == 1 and y_r == 4:
        recommend_label = 'K07'
        #recommend_label = 'K08'
    # 2,1
    elif x_r == 2 and y_r == 1:
        recommend_label = 'M01'
        #recommend_label = 'M02'
    # 2,2
    elif x_r == 2 and y_r == 2:
        recommend_label = 'K13'
        #recommend_label = 'M04'
    # 2,3
    elif x_r == 2 and y_r == 3:
        recommend_label = 'K11'
        #recommend_label = 'K12'
    # 2,4
    elif x_r == 2 and y_r == 4:
        recommend_label = 'K09'
        #recommend_label = 'K10'
    # 3,1
    elif x_r == 3 and y_r == 1:
        recommend_label = 'M03'
        #recommend_label = 'M001'
    # 3,2
    elif x_r == 3 and y_r == 2:
        recommend_label = 'M05'
        #recommend_label = 'M002'
    # 3,3
    elif x_r == 3 and y_r == 3:
        recommend_label = 'N011'
        #recommend_label = 'M003'
    # 3,4
    elif x_r == 3 and y_r == 4:
        recommend_label = 'N012'
        #recommend_label = 'M002'
    else:
        recommend_label = 'K13'
    return recommend_label

```

Şekil 5.10. Pyhton'da Lokasyon Ataması Yapılması.

Tablo 5.4. Lokasyon Atamasının Gerçekleştirilmesi

No	Antecedents	Consequents	Support	Confidence	X	Y	Recommend
47	frozenset({'_TekstilC'})	frozenset({'_TekstilD'})	1.000	1.000	3	2	M05
48	frozenset({'_TekstilE'})	frozenset({'_TekstilC'})	1.000	1.000	1	3	K06
49	frozenset({'_TekstilC'})	frozenset({'_TekstilE'})	1.000	1.000	3	2	M05
52	frozenset({'_TekstilE'})	frozenset({'_TekstilD'})	1.000	1.000	1	4	K08
37	frozenset({'_TekstilA'})	frozenset({'_TekstilE'})	1.000	1.000	1	1	K02
36	frozenset({'_TekstilE'})	frozenset({'_TekstilA'})	1.000	1.000	1	3	K05
35	frozenset({'_TekstilA'})	frozenset({'_TekstilD'})	1.000	1.000	1	4	K07
34	frozenset({'_TekstilD'})	frozenset({'_TekstilA'})	1.000	1.000	3	4	N012
53	frozenset({'_TekstilD'})	frozenset({'_TekstilE'})	1.000	1.000	3	3	N011
33	frozenset({'_TekstilA'})	frozenset({'_TekstilC'})	1.000	1.000	3	2	M05
32	frozenset({'_TekstilC'})	frozenset({'_TekstilA'})	1.000	1.000	3	4	N012
40	frozenset({'_TekstilB'})	frozenset({'_TekstilC'})	0.975	1.000	3	4	N012
30	frozenset({'_TekstilB'})	frozenset({'_TekstilA'})	0.975	1.000	3	4	N012
31	frozenset({'_TekstilA'})	frozenset({'_TekstilB'})	0.975	0.975	3	4	N012
256	frozenset({'_Tekstil'})	frozenset({'_TekstilD', '_TekstilE'})	0.933	1.000	1	4	K08
203	frozenset({'_Tekstil'})	frozenset({'_TekstilE', '_TekstilA'})	0.933	1.000	3	4	N012
200	frozenset({'_TekstilE', '_Tekstil'})	frozenset({'_TekstilA'})	0.933	1.000	3	1	M03
197	frozenset({'_Tekstil'})	frozenset({'_TekstilD', '_TekstilA'})	0.933	1.000	3	4	N012
24	frozenset({'_Tekstil'})	frozenset({'_TekstilD'})	0.933	1.000	3	1	M03
26	frozenset({'_Tekstil'})	frozenset({'_TekstilE'})	0.933	1.000	3	4	N012
22	frozenset({'_Tekstil'})	frozenset({'_TekstilC'})	0.933	1.000	1	1	K01
...	...	...	...	...	...	...	...

Bu çalışmanın uygulamasında firmanın e ticaret satış verileri ele alınmış olup ilk olarak apriori algoritması ile müşteri siparişlerinden oluşan grup numaralarındaki ürün kategorilerin birliktelik ilişkisi analiz edilmiştir. Daha sonra elde edilen bu analizler yorumlanmış ve depodaki ürünlerin daha verimli bir lokasyonlara adreslenebilmesi için lokasyon öneri sistemi geliştirilmiştir. Şekil 5.9’da belirlenen lokasyon atamalarına göre ürün setinde yer alan birlikteliklerin lokasyon önerileri güven değeri en yüksek olan 20 satır için Tablo 5.4.’de belirtilmiştir. Buna göre 47 nolu birliktelikte yer alan TekstilC ve TekstilD ürün kategorilerinin birlikteliğine göre önerilen lokasyon M05 lokasyonu olmuştur.

Önerilen yöntemde aynı zamanda bir ürün kategorisi için farklı lokasyon önerilerinde gerçekleştirildiğini görebilmekteyiz. Birliktelik ilişkisi sonuçlarına baktığımızda, birden fazla ürün kategorisinin farklı ürünleler ile ilişkisi olduğu görülmektedir. Böylelikle ilişkili ürünler depo yerleşiminde daha yakın lokasyonda bulunması sağlanmaktadır. Bu sebeple farklı lokasyonlarda aynı ürün kategorisinden ürünlerin yer alması siparişlerin toplanma süreci açısından fayda sağlamaktadır. Diğer birlikteliklere ait önerilen lokasyonlar Tablo 5.4’de yer verilmiştir.

Çalışma sonuçlarının ürünlerin bulunduğu lokasyon çeşitliği daha fazla olan sipariş gruplarında anlamlı sonuçlar ürettiği görülmüştür. Bu nedenle mevcut durum analizinde belirtilen ve sonuç bölümünde ortaya konular veriler veri setinde yer alan ve lokasyon çeşitliliği en fazla olan ürün grupları arasından seçilmiştir.

Geliştirilen bu yöntemde ürünlerin depo yerleşimde daha verimli adreslenmesi amaçlanmış olup, mevcut durumla karşılaştırılması ve elde edilen avantajlar bir sonraki bölümde değerlendirilmiştir. (Tablo 5.4), (Şekil 5.9).



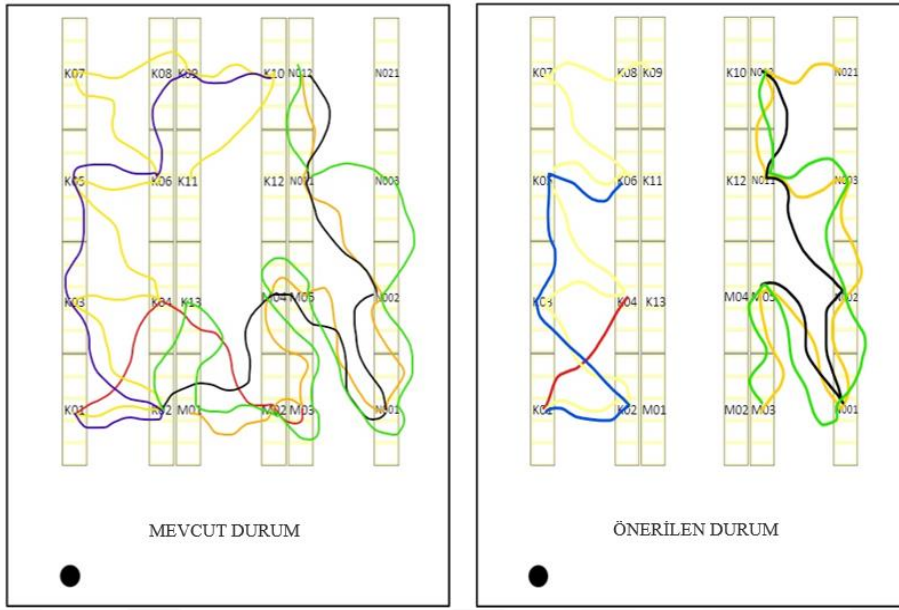
6. SONUÇLAR VE ÖNERİLER

Gelişmiş teknoloji ve pandemi etkisi ile birlikte e-ticaret sektörü hızla büyümüş, firmalar rekabete ayak uydurabilmek için depo operasyonlarını geliştirmeye odaklanmıştır. Bu çalışmada öncelikle makine öğrenimi tanımları, kullanılan alanlar, depolama tanımı ve faaliyetleri, makine öğrenimi algoritmaları ve yazılımları hakkında bilgi verilmiştir. Sonrasında çalışmaya konu olan firmanın mevcutta depo yerleşimi ve veri seti hakkında bilgi verilerek mevcut durum analizi ortaya konulmuştur.

Mevcut durum analizinde firmanın ürünlerinin depoda plansız ve düzensiz bir yerleşiminin olduğu görülmüştür. Depo operasyonunda müşteri siparişlerinin daha yüksek üretkenlikle toplanabilmesi ve değişen müşteri ihtiyaçlarına hızlı cevap verebilmek adına ürünler için yeni ve esnek yapıya sahip depo yerleşim planı oluşturulmasına karar verilmiştir.

Verilen bu karar doğrultusunda çalışmanın uygulama bölümünde müşteri siparişleri arasındaki birliktelik ilişkisini analiz etmek için makine öğrenimi algoritmalarından apriori algoritmasından faydalanılmıştır. Bu alanda geniş bir kullanıma sahip olan apriori algoritması aracılığıyla gruplanan müşteri siparişleri arasındaki ilişkilerin analizi yapılmış ve birliktelik kuralları ortaya konulmuştur. Ortaya çıkan kurallar neticesinde depo yerleşim planında hangi ürünlerin birbirlerine daha yakın lokasyonlarda adreslenmesini sağlayacak bir yöntem geliştirilmiş olup güven değeri 0,4 üzerinde olan birliktelikler için lokasyon önerileri paylaşılmıştır.

Elde edilen analiz sonuçları değerlendirilerek önerilen yerleşim düzenine Şekil 6.1’de yer verilmiştir. Şekil 6.1.’de mevcut durum ile farkları da görülmektedir.(Şekil6.1)



Şekil 6.1. Önerilen Yerleşim Düzeni

Bu önerilen örnek düzene göre müşteri siparişleri sipariş toplama mesafeleri hesaplanmış Tablo 6.1’de kazanılan yürüme mesafeleri paylaşılmıştır. Kırmızı renkle gösterilen GrupNo127 nolu grupta 11,2 m, sarı renkle gösterilen GrupNo207 nolu grupta 4,6 m, mavi renkle gösterilen GrupNo27 nolu grupta 8,1 m, turuncu renkle gösterilen GrupNo389 nolu grupta 12,1 m, siyah renkle gösterilen GrupNo257 nolu grupta 5,8 m ve yeşil renkle gösterilen GrupNo463 nolu grupta 14,5 m olmak üzere toplamda bu 6 grup numarası için 56,3 m yürütülen mesafeden kazanç sağlanmıştır. (Tablo 6.1)

Tablo 6.1. Önerilen Duruma Göre Sonuçlar.

Grup No	Mevcut Durum Yürüme Mesafesi	Önerilen Durum Yürüme Mesafesi	Önerilen Durum Kazanç Mesafesi	Önerilen Durum Kazanç Süresi
GrupNo127	19,3 m	8,09 m	11,2 m	37 sn
GrupNo207	27,9 m	23,3 m	4,6 m	15 sn
GrupNo27	21,9 m	13,8 m	8,1 m	27 sn
GrupNo389	29,6 m	17,5 m	12,1 m	40 sn
GrupNo257	24,1 m	18,3 m	5,8 m	19 sn
GrupNo463	34,5 m	20,0 m	14,5 m	48 sn

Bu çalışmada kazanç sağlanan mesafenin depo personelinin saatlik ürün toplama üretkenliğine pozitif anlamda etkisi olduğu da görülmüştür. Depo personelinin ürün toplarken saatte yürüdüğü mesafe 0,3 m varsayımına göre grup no bazında elde edilen kazanç süreleri yine Tablo 6.1’de belirtilmiştir. Bu sonuçlar, sipariş toplama üretkenliğine ortalama 31,28 sn kazanç sağlanmıştır. Buna göre de elde edilen bu kazanç ile depo personelinin saatlik ürün toplama performansında %31,8’lik verimlilik sağlandığı görülmüştür.

Çalışmada ele alınan problem ve çözümü için uygulanan model depo operasyonları açısından oldukça önemlidir. Bu çalışma sayesinde firma depo operasyonu ürünlerin daha doğru lokasyonlarda adreslenmesi ile birlikte daha hızlı ürün toplama performansına sahip olabilecektir. Ayrıca ürünlerin depo lokasyonlarına adreslenmesi personel inisiyatifinden makine öğrenimine aktararak sürecin daha doğru yönetilmesi sağlanmıştır. Bununla birlikte değişen sektör ve müşteri alışkanlıklarından en az oranda etkilenmiş olup hızlı bir şekilde adapte olabilecektir. Bununla birlikte müşteri sipariş hazırlama sürelerinin kısılması müşteri memnuniyetinin artmasına ve satış hacminin çoğalmasına katkıda bulunacaktır.

Son olarak bu çalışmayı daha ileriye götürmek amacıyla, kullanılan veri setlerinin analiz edilen zaman periyodunun optimum aralığa göre belirlenmesi, apriori algoritmasının kümeleme algoritmaları ile birlikte değerlendirilmesi ve önerilen depo lokasyon sistemi atama problemlerindeki matematiksel model ve algoritmalar kullanarak lokasyon atama sistemi geliştirilmesi gibi konular ele alınarak araştırma önerileri değerlendirilebilir.



KAYNAKÇA

- Abelfotoh, A., Singh, M. & Süer, A. G. (2019). Order Batching Optimization for Warehouses with Cluster-Picking. 25th International Conference on Production Research Manufacturing Innovation: Cyber Physical Manufacturing, Chicago, USA, August 9- 14, ss. 1464 –1473.
- Ahmed, M., Choudhury, N., & Uddin, S. (2017). Anomaly detection on big data in financial markets. İçinde Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, Sydney, Australia, 31 Temmuz – 3 Ağustos, ss. 998–1001.
- Alenezi, H. S., & Faisal, M. H. (2020). Utilizing crowdsourcing and machine learning in education: Literature review. *Education and Information Technologies*, 25(4), 2971-2986.
- Al-Garadi, M. A., Hussain, M. R., Khan, N., Murtaza, G., Nweke, H. F., Ali, I., ... & Gani, A. (2019). Predicting cyberbullying on social media in the big data era using machine learning algorithms: review of literature and open challenges. *IEEE Access*, 7, 70701-70718.
- Alias, C., Salewski, U., Ortiz Ruiz, V. E., Alarcón Olalla, F. E., Neirão Reymão, J. D. E., & Noche, B. (2017, June). Adapting warehouse management systems to the requirements of the evolving era of industry 4.0. *International Manufacturing Science and Engineering Conference* (Sayı: 50749, ss. V003T04A051). American Society of Mechanical Engineers.
- Andrii, G. (2015). System approach for logistics distribution network's organization and planning. *International Journal of Automation, Control and Intelligent Systems*, 1(2), 27-33.
- Augustyn, L., & Aurelija B. (2020). Improve the orders picking in ecommerce by using wms data and bigdata analysis. *FME Transactions*, 49, 233-243.
- Baker, P., & Canessa, M. (2009). Warehouse design: A structured approach. *European journal of operational research*, 193(2), 425-436.
- Bi, Q., Goodman, K. E., Kaminsky, J. vd. (2019). What is machine learning? A primer for the epidemiologist. *American Journal of Epidemiology*, 188(12), 2222–2239.
- Boog, J. (2022). 10 Best Machine Learning Software [2023 List]. <https://theqalead.com/tools/best-machine-learning-software/>
- Bush, R. R., & Mosteller, F. (1951). A mathematical model for simple learning. *Psychological Review*, 58(5), 313–323.
- Calzavara, M., Glock, C. H., Grosse, E. H., vd. (2017). Analysis of economic and ergonomic performance measures of different rack layouts in an order picking warehouse. *Computers & Industrial Engineering*, 111, 527-536.
- Ciardo, F., De Tommaso, D., & Wykowska, A. (2022). Human-like behavioral variability blurs the distinction between a human and a machine in a nonverbal Turing test. *Science Robotics*, 7(68).

- Çalış, G. (2009). *Şantiye Yerleşim problemi Çözümünde Karınca koloni Optimizasyonunun kullanımı* [Yayımlanmamış Doktora Tezi]. Ege Üniversitesi.
- Çavuşlar, M. (2020). *Mühendis bakış açısıyla depo tasarımı. Mühendis ve makine. TMMOB.*
- Demircan, Ö. (2018). *Depo Raf Yerleşim Problemi Ve Kümeleme Temelli Bir Yaklaşım* [Yayımlanmamış Yüksek Lisans Tezi]. Erciyes Üniversitesi, Kayseri.
- Duda, R. O., Hart, P. E., & Stork, D. G. (2012). *Pattern Classification*. (2th ed). Hoboken, NJ: John Wiley & Sons, Inc.
- Evinsel, C. (2010). *Depo Tasarımı ve Yerleşimi* [Yayımlanmamış Yüksek Lisans Tezi]. İstanbul Üniversitesi.
- Fradkov, A. L. (2020). Early History of Machine Learning. *Preprints of the 21st IFAC World Congress (Virtual)* Berlin, Germany, July 12-17, 1411-1416.
- Gopani, A. (2022). The history of machine learning algorithms. <https://analyticsindiamag.com/the-history-of-machine-learning-algorithms/> adresinden erişilmiştir. the machine is also a human being.
- Gubin L.G., Polyak B.T., Raik E.V. (1967). The method of projections for finding the common point of convex sets. *USSR Computational Mathematics and Mathematical Physics*, 7(6), 1–24.
- Gülce, A. C. (2010). *Veri Madenciliğinde Apriori Algoritması ve Apriori Algoritmasının Farklı Veri Kümelerinde Uygulanması* [Yayımlanmamış Yüksek Lisans Tezi]. Trakya Üniversitesi, Edirne.
- Gümüş, Y. (2009). Lojistik faaliyetlerin rekabet stratejileri ve işletme kârı ile olan ilişkisi. *Muhasebe ve Finansman Dergisi*, (41), 97-114.
- He, X., Pan, J., Jin, O., Xu, T., Liu, B., Xu, T., ... & Candela, J. Q. (2014, August). Practical lessons from predicting clicks on ads at facebook. *Proceedings of the eighth international workshop on data mining for online advertising* (ss. 1-9).
- Jacyna, M., Lewczuk, K. & Kłodawski, M. (2015). Technical and organizational conditions of designing warehouses with different functional structures. *J. Kones*, 22, 49–58.
- Jain, A. K. (2010). Data clustering: 50 years beyond K-means. *Pattern Recognit Lett.*, 31(8), 651–666.
- Jain, A.K., Murty, M.N., & Flynn, P.J. (1999). Data clustering: A review. *ACM Comput. Surv.*, 31, 264–323.
- Jenkins, A. (2022). Logistics for Business Defined: Importance Role & Benefits. <https://www.netsuite.com/portal/resource/articles/erp/logistics.shtml>
- Kelly, S., & Kelly, S. (2016). What Is Python?. *Python, PyGame and Raspberry Pi Game Development*. Springer.
- Khisanova, V. (2022). *Python libraries and their usage in visualizing meteorological data* [Bachelor's Thesis]. Haaga-Helia University of Applied Sciences.
- Kocifaj, M. (2013). Modelling of infrastructure for warehouse simulation. *J. Inf. Control Manag. Syst.*, 11, 15–22.

- Kulkarni, C. S., Bhavsar, A. U., Pingale, S. R., & Kumbhar, S. S. (2017). BANK CHAT BOT—an intelligent assistant system using NLP and machine learning. *International Research Journal of Engineering and Technology*, 4(5), 2374-2377.
- Logistics Terms Glossary [Lojistik Terimler Sözlüğü]. (2022). *Addressing*. <https://www.ekol.com/en/logistics-terms-glossary/>
- Lopienski, K. (2021). Shipping Logistics: Definition, Best Practices, & Guide. <https://www.shipbob.com/blog/shipping-logistics/#:~:text=What%20does%20shipping%20logistics%20mean,drayage%20to%20end%20customer%20shipments>
- Mahroof, K. (2019). A human-centric perspective exploring the readiness towards smart warehousing: The case of a large retail distribution warehouse. *International Journal of Information Management*, 45, 176-190.
- Marasová, D., & Šaderová (2019). J. Possibilities to increase the warehouse capacity: Case study. *Proceedings of the 8th Carpathian Logistics Congress on Logistics, Distribution, Transport and Management* (CLC 2018), Prag, Çek Cumhuriyeti, 3–5 Aralık 2018; Tanger: Ostrava, Çek Cumhuriyeti 2019; 633–638.
- Mohri, M., Rostamizadeh, A., & Talwalkar, A. (2018). *Foundations of machine learning*. MIT Press.
- Noble, W. S. (2006). What is a support vector machine? *Nat Biotechnol*, 24(12), 1565–1567.
- Olçay, D. (2018). *Development of Warehousing Systems: A Roadmap to Warehouse 4.0*. [Yayımlanmamış Yüksek Lisans Tezi]. Yaşar Üniversitesi, İzmir.
- Olden, J. D., & Jackson, D. A. (2002). Illuminating the “black box”: A randomization approach for understanding variable contributions in artificial neural networks. *Ecol Modell*, 154(1-2), 135–150.
- Öz, B. (2019). *Order Picking Oriented Storage Assignment Problem in a Vertical Lift Module System* [Thesis M.S.]. Middle East Technical University, Ankara.
- Pen Y., Zhijie Z., Huijie G. (2020). Order batch picking optimization under different storage scenarios for e-commerce warehouses. *Transportation Research Part E*, 136,
- Petersen, C., Aase G., (2004). A comparison of picking, storage, and routing policies in manual order picking, *Int. J. Production Economics* 92(2004) 11–19
- RedMonk. (2021). The RedMonk Programming Language Rankings: June 2021. <https://redmonk.com/sograpy/2021/08/05/language-rankings-6-21/>
- Reif, R., & Günthner, W. A. (2009). Pick-by-vision: augmented reality supported order picking. *Vis Comput*, 25, 461–467.
- Rimienė, K., & Grundey, D. (2007). Logistics centre concept through evolution and definition. *Engineering economics*, 54(4).
- Saderová, J. (2017). *Methodology for the Design of the Warehouse System* [Habilitation Thesis]. Technical University of Košice, 156.

- Saderova, J., Rosova, A., Sofranko, M. & Kacmary, P. (2021). Example of Warehouse System Design Based on the Principle of Logistics. *Sustainability*, 13, 4492.
- Santoso, M. H. (2021). Application of Association Rule Method Using Apriori Algorithm to Find Sales Patterns Case Study of Indomaret Tanjung Anom. *Brilliance: Research of Artificial Intelligence*, 1(2), 54-66.
- Saranya, T., Sridevi, S., Deisy, C., Chung, T. D., & Khan, M. A. (2020). Performance analysis of machine learning algorithms in intrusion detection system: a review. *Procedia Computer Science*, 171, 1251-1260.
- Silva A., Roodbergen J. K., Coelho L. C., Darvish M. (2017). Estimating optimal ABC zone sizes in manual warehouses. 10.13140/RG.2.2.15357.28645.
- Smola, A. J., & Schölkopf, B. (2004). A tutorial on support vector regression. *Stat Comput.*, 14(3), 199–222.
- Softwaretestinghelp. (2023a). Apriori Algorithm In Data Mining: Implementation With Examples. <https://www.softwaretestinghelp.com/apriori-algorithm/>
- Softwaretestinghelp. (2023b). 11 Most Popular Machine Learning Software Tools in 2023. <https://www.softwaretestinghelp.com/machine-learning-tools/>
- Subrahmanyam, R.V. , Kumar, M. K., Nair, S. vd. (2021). Design and development of automatic warehouse sorting Rover. *Materials Today: Proceedings*, 46, 4497-4503.
- Stackoverflow. (2021). Developer Survey - Programming, scripting, and markup languages. <https://insights.stackoverflow.com/survey/2021#most-loved-dreaded-and-wanted-language-want>
- Thomas Chabot, Leandro C. Coelho, Jacques Renaud (2018). Thomas Chabot Leandro C. Coelho Jacques Renaud *The International Journal of Advanced Manufacturing Technology*, 119, 2369–2392.
- Tufano, A., Accorsi, R., & Manzini, R. (2022). A machine learning approach for predictive warehouse design. *Interuniversity Research Centre on Enterprise Networks*,
- Verbraeken, J., Wolting, M., Katzy, J. vd. (2020). A survey on distributed machine learning. *ACM Computing Surveys*, 53(2), 1-33.
- Wan, Q., Suhui, G., Shulian., Z. (2022). An optimization algorithm for TSP-like problem in warehouse picking. *International Conference on Computer Science and Communication Technology*, China, 125060K (28 December 2022), <https://doi.org/10.1117/12.2661781>
- Webb, G. I., Keogh, E., & Miikkulainen, R. (2010). Naïve Bayes. *Encyclopedia of machine learning*, 15, 713-714
- Wei, L., Feng, B., & Liu, Q. (2020). Picker Routing Optimization of Storage Stacker Based on Improved Multi-Objective Iterative Local Search Algorithm. *Journal of Industrial and Management Optimization*, 1-18.
- Wu, I., & Chiu, M. (2018). Examining supply chain collaboration with determinants and performance impact: Social capital, justice, and technology use perspectives. *International Journal of Information Management*, 39, 5–19.

- Zhuang, Y., Zhou, Y., Hassini, E. vd. (2022). Rack retrieval and repositioning optimization problem in robotic mobile fulfillment systems. *Transportation Research Part E*, 167, 1-28.
- Zunic, E., Hasic, H., Hodzic, K., Delalic, S., & Besirevic, A. (2018, Mayıs). Predictive analysis based approach for optimal warehouse product positioning. *41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)* (ss. 0950-0954). IEEE.





ÖZGEÇMİŞ

Ad-Soyad : Melih Yüce KILIÇARSLAN

ÖĞRENİM DURUMU:

Lisans : 2013 - 2017, Sakarya Üniversitesi, Mühendislik Fakültesi, Endüstri Mühendisliği

Yükseklisans : 2019 - , Sakarya Üniversitesi, Endüstri Mühendisliği, Mühendislik Yönetimi

MESLEKİ DENEYİM VE ÖDÜLLER:

2019-2022 yılları arasında Ekol Lojistikte Çözüm Tasarım uzmanı olarak çalıştı.

2022 yılından beri Ford Otosanda Hammadde Satın Alma Mühendisi olarak çalışmaya devam etmektedir.

TEZDEN TÜRETİLEN ESERLER:

Kılıçarslan M. Y., Çil İ., (2023, 25-26, Şubat). Apriori Algoritması Kullanılarak Depo Raf Yerleşiminin Yeniden Planlanması: Bir E-Ticaret Firmasında Uygulaması. 5. *Uluslararası Mühendislik Bilimleri Ve Multidisipliner Yaklaşımlar Kongresi*, İstanbul, Turkey.

DİĞER ESERLER: