# DEPOLARDA SİPARİŞ TOPLAMA İŞLEMLERİNE YÖNELİK AKILLI LOKASYONLAMA UYGULAMASI

Ceyhun KARATAŞ
211402104

**YÜKSEK LİSANS TEZİ**
Bilgisayar Mühendisliği Anabilim Dalı
Bilgisayar Mühendisliği Tezli Yüksek Lisans Programı
Danışman: Dr. Öğr. Üyesi Önder ŞAHİNASLAN

İstanbul
T.C. Maltepe Üniversitesi
Lisansüstü Eğitim Enstitüsü
Ekim, 2022

# DEPOLARDA SİPARİŞ TOPLAMA İŞLEMLERİNE YÖNELİK AKILLI LOKASYONLAMA UYGULAMASI

Ceyhun KARATAŞ
211402104
Orcid: 0000-0001-5560-2867

**YÜKSEK LİSANS TEZİ**
Bilgisayar Mühendisliği Anabilim Dalı
Bilgisayar Mühendisliği Tezli Yüksek Lisans Programı
Danışman: Dr. Öğr. Üyesi Önder ŞAHİNASLAN

İstanbul
T.C. Maltepe Üniversitesi
Lisansüstü Eğitim Enstitüsü
Ekim, 2022



# JÜRİ VE ENSTİTÜ ONAYI

Bu belge, Yükseköğretim Kurulu tarafından 19.01.2021 tarihli "*Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge*" ile bildirilen 6689 Sayılı Kişisel Verilerin Korunması Kanunu kapsamında gizlenmiştir.

# ETİK İLKE VE KURALLARA UYUM BEYANI

Bu belge, Yükseköğretim Kurulu tarafından 19.01.2021 tarihli "*Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge*" ile bildirilen 6689 Sayılı Kişisel Verilerin Korunması Kanunu kapsamında gizlenmiştir.

# TEŞEKKÜR

Bu çalışmada maddi ve manevi desteğiyle yanımda olan eşim **Gökçe Karataş'a**, moral desteği sağlayan kızım **Duru Karataş'a**, çalışma esnasında sürekli fikir ve öneriyle destek veren değerli çalışma arkadaşım **Muaz Gültekin'e**, tez çalışması esnasında birlikte çalıştığım okul arkadaşım **Mustafa Günerkan'a**, yapılan çalışmalar esnasında işyerindeki imkanları kullanmamda yardımcı olan çalışma arkadaşım **Serkan Dinga'ya**, yine Barsan Global Lojistik firmasının imkanlarını bu çalışmamda kullanmama izin veren Barsan Global Lojistik kurucusu **Kamil Barlın'a**, değerli görüşlerinden faydalandığım **Dr. Ender Şahinaslan'a** ile deneyim ve tecrübesiyle her zaman yanımda olan akademik danışmanım **Dr. Öğr. Üyesi Önder Şahinaslan'a** teşekkürlerimi sunarım.

Ceyhun KARATAŞ

Ekim, 2022

# ÖZ

## DEPOLARDA SİPARİŞ TOPLAMA İŞLEMLERİNE YÖNELİK AKILLI LOKASYONLAMA UYGULAMASI

Ceyhun Karataş
Yüksek Lisans Tezi
Bilgisayar Mühendisliği Anabilim Dalı
Bilgisayar Mühendisliği Yüksek Lisans Programı
Danışman: Dr. Öğr. Üyesi Önder Şahinaslan
Maltepe Üniversitesi Lisansüstü Eğitim Enstitüsü, 2022

Depolama, lojistik faaliyetlerinin temel faktörlerinden biridir. Lojistik depolarda ise çok çeşitli operasyonlar yapılabilmektedir. Bu operasyonlardan en kritiklerinden biri de ürün toplama operasyonudur. Yapılan araştırmalara göre, depolardaki maliyetlerin %50'den fazlasını ürün toplama süreçleri oluşturmaktadır. Bu nedenden dolayı, sipariş toplama süreçlerindeki süre ve maliyetleri en aza indirgemek ve bu şekilde depoların etkin ve verimli kullanımını sağlamak, sektörde faaliyet gösterenler için önemlidir. Doğru sipariş toplamanın gerçekleştirilmesi, doğru adreslemenin yapılmasıyla mümkün olabilir. Bu adreslemenin, optimum şekilde yapılmaması neticesinde; depolarda sipariş toplama süreçleri uzamakta, gereğinden fazla atıl alan oluşmakta ve maliyetler artmaktadır. Bu bağlamda; geliştireceğimiz yeni bir algoritma yaklaşımla, karar vericiler doğru adresleme gerçekleştirebilecektir. Bu sayede depolarda hacimsel kullanım verimliliği artacak ve ürün dolaşım süreleri azalacaktır. Sonucunda, ciddi ekonomik kayıplar önlenecektir. Projenin gerçekleşmesi durumunda hem hizmet sağlayıcıların hem de hizmet alıcıların depolama süreçlerinden beklentileri de iyileştirilmiş olacaktır.

**Anahtar Sözcükler:** Teknoloji ve yenilik, veri yönetimi, lojistik depolama algoritmaları, sipariş toplama algoritmaları, ürün yerleştirme, depolarda raflama, lokasyonlama

# ABSTRACT

## SMART LOCATION APPLICATION FOR ORDER COLLECTION IN WAREHOUSES

Ceyhun Karataş
Master Thesis
Department of Computer Engineering
Computer Engineering Programme
Thesis Advisor: Asst. Prof. Önder Şahinaslan
Maltepe University Graduate School, 2022

Storage is one of the main factors of logistics activities. A wide variety of operations can be performed in logistics warehouses. One of the most critical of these operations is the product collection operation. According to the researches, more than 50% of the costs in warehouses are composed of product collection processes. For this reason, it is important for those operating in the sector to minimize the time and costs in order picking processes and to ensure the effective and efficient use of warehouses in this way. Correct order picking can be achieved by correct addressing. As a result of this addressing not being done optimally; order picking processes in warehouses are prolonged, more idle space is created than necessary, and costs increase. In this context; With a new algorithm approach that we will develop, decision makers will be able to perform correct addressing. In this way, volumetric utilization efficiency in warehouses will increase and product circulation times will decrease. As a result, serious economic losses will be avoided. In the event that the project is realized, the expectations of both service providers and service buyers from storage processes will also be improved.

**Keywords:** Technology and innovation, data management, logistics storage algorithms, order picking algorithms, product placement, shelving in warehouses, locating

# İÇİNDEKİLER

JÜRİ VE ENSTİTÜ ONAYI ..... i
ETİK İLKE VE KURALLARA UYUM BEYANI ..... ii
TEŞEKKÜR ..... iii
ÖZ ..... iv
ABSTRACT ..... v
İÇİNDEKİLER ..... vi
TABLOLAR LİSTESİ ..... x
ŞEKİLLER LİSTESİ ..... xii
KISALTMALAR ..... xiv
BÖLÜM 1. GİRİŞ ..... 1
Literatür Taraması ..... 2
Problem ..... 4
Amaç ..... 5
Önem ..... 5
Varsayımlar ..... 6
Sınırlıklar ..... 6
Tanımlar ..... 6
Depo ..... 6
Warehouse Management System ..... 7
Raf ..... 7
Lokasyon ..... 7
Forklift ..... 8
Elleçleme ..... 8
Clarke-Wright Algoritması ..... 8
Meta-Sezgisel Algoritmalar ..... 8
ABC Analizi ..... 9
Delphi Tekniği ..... 9
Principal Component Analysis (PCA) ..... 9
Aykırı Değer (Outlier) ..... 10

Çeyrekler Açıklığı Metodu ..... 10
Makine Öğrenmesi ..... 10
BÖLÜM 2. ARAŞTIRMA ve PLANLAMA ..... 11
Depoların Genel Fiziki Yapısı ..... 11
Yerleştirme ve Toplama Döngüsü ..... 13
Depo Operasyonlarının İcra Biçimleri ..... 15
Mal Kabul Operasyonu ..... 16
Mal Kabul Problemi Literatür Taraması ..... 18
Sipariş Toplama Operasyonu ..... 20
Toplama Listelerinin Hazırlanması ..... 22
Sipariş Toplama Operasyonu Üzerinde Toplama Listelerinin Etkisi ..... 23
BÖLÜM 3. YÖNTEM ..... 26
Araştırma Modeli ..... 26
Depo Sayısının Fazlalığı ve Yaygınlığı ..... 27
Depo Sirkülasyon Hızının Yüksekliği ..... 29
Depolama Hizmeti Verilen Müşteri Sayısı ve Sektör Çeşitliliği ..... 29
Evren ve Örneklem ..... 30
Veriler ve Toplanması ..... 33
Verilerin Çözümlenmesi ve Yorumlanması ..... 38
Ham Verinin Çözümlenmesi ..... 38
Uzman Grubunun Oluşturulması ..... 43
Ham Verinin Yorumlanması ..... 44
KutuId ..... 49
StatuID ..... 50
UrunKodu ..... 50
AmbarId ..... 50
Tipi ..... 50
IslemTarihi ..... 50
Marka ..... 50
UrunGrubu ..... 50

ModelKodu ..... 50
TipDeger ..... 51
TakipNotu ..... 51
Lot ..... 51
SonKullanmaTarihi ..... 51
FirmaId ..... 51
StokGirisTarihi ..... 51
KutuFirmaId ..... 51
KutuTipId ..... 51
KutuTipGrup ..... 51
CikisTarihi ..... 51
IrsaliyeTarihi ..... 51
Yorumlanmış Verinin Gruplanması ..... 53
Yorumlanmış Verinin Yeniden Filtrelenmesi ..... 54
Filtrelenmiş Verinin Yeniden Gruplanması ..... 55
BÖLÜM 4. UYGULAMA ..... 58
Anahtar Değerinin Belirlenme Yöntemi ..... 60
Lokasyon Gruplamalarının Belirlenmesi ..... 64
Veri Setinin Son Hali ve Yapısı ..... 67
Bölge Grup Bulma Fonksiyonunun Hazırlanması ..... 69
Fonksiyonun çağrılması için örnek ..... 70
Hazırlanan Fonksiyonun Depo Yönetim Yazılımı İçine Yerleştirilmesi ..... 71
BÖLÜM 5. BULGULAR VE YORUMLAR ..... 74
Bulgular ..... 74
Ham Datanın Yeni Uygulama Aracılığıyla Analizi ..... 74
Yeni Uygulamanın Saha Sonuçlarının Analizi ..... 80
Yeni Uygulamanın Envanter Yönetime Üzerine Etkilerinin Analizi ..... 80
Yeni Uygulamanın Mesafeler Üzerinde Etkilerinin Analizi ..... 83
Yorumlar ..... 85

Makine Öğrenmesi Algoritmaları ..... 85
Naive Bayes Sınıflandırması ..... 86
Karar Ağacı (Decision Tree) Algoritması ..... 86
En Yakın Komşuluk (k Nearest Neighbours) Algoritması ..... 86
Rastgele Orman (Random Forest) Algoritması ..... 86
Ridge Sınıflandırma Algoritması ..... 86
Torbalama (Bagging) Yöntemi ..... 87
Yükseltme (Boosting) Yöntemi ..... 87
Çok Katmanlı Algılayıcılar (Multi-Layer Perceptron) ..... 87
Makine Öğrenmesi Yöntemleri İçin Yeni Bir Veri Seti Hazırlanması ..... 87
Karşılaştırma Uygulamasının Hazırlanması ..... 88
Karşılaştırma Sonuçlarının Analizi ..... 90
ABC Analizi Yöntemi ..... 90
Devir Bazlı Yerleştirme Yöntemi ve Gruba Dayalı Yerleştirme Yöntemi ..... 91
Sınıfa Dayalı Yerleştirme Yöntemi ..... 91
Sipariş Toplama Problemine Yönelik Yapılmış Çalışmalar ..... 92
Literatürdeki Geçmiş Çalışmaların DSTİYAU ile Karşılaştırılması ..... 92
BÖLÜM 6. SONUÇ ..... 94
Özet ..... 94
Yargı ..... 95
Öneriler ..... 95
KAYNAKÇA ..... 97
ÖZGEÇMİŞ ..... 102

# TABLOLAR LİSTESİ

**Tablo 1.** Sipariş toplama sürecinde çeşitli aktivitelerin zaman payı ..... 6
**Tablo 2.** Ürün yerleştirmede kullanılan çeşitli yöntemler ve bunların özellikleri ..... 19
**Tablo 3.** Çalışmada incelenen BGL'nin depo tesisleri ..... 27
**Tablo 4.** BGL'ye ait depolara 2020 yılında aylık giriş sayıları ..... 29
**Tablo 5.** BGL'nin hizmet verdiği müşteriler ve bu müşterilerin sektörleri ..... 30
**Tablo 6.** Willebroek tesislerinin sirkülasyon hızının BGL içindeki oranı ..... 32
**Tablo 7.** Tüm BGL ve Willebroek ölçeğinde 2020 yılı için giriş yapılan kutu sayıları 32
**Tablo 8.** HamVeri tablosu örneği ..... 43
**Tablo 9.** LDS sisteminin çalışmasına özgü alanlara örnek tablo ..... 44
**Tablo 10.** Mükerrer alanlara örnek tablo ..... 45
**Tablo 11.** İçeriği sürekli tekrarlayan veya hep boş olan kayıtlara örnek tablo ..... 46
**Tablo 12.** Depo içinde değiştirilen alanlarla ilgili örnek tablo ..... 47
**Tablo 13.** AileGrupKodu alanının toplama göre doluluk durumu ..... 47
**Tablo 14.** Giriş emri ile ilgili bilgilerin 20 kayıtlık örnek tablosu ..... 48
**Tablo 15.** Kutunun fiziki ölçüleriyle sirkülasyon hızı arasında bağlantı kurulamayan analiz çalışmasından örnek 30 kayıt ..... 49
**Tablo 16.** YorumlanmisVeriSeti tablosu örneği ..... 52
**Tablo 17.** UruneGoreGrupluVeriSeti tablosu örneği ..... 54
**Tablo 18.** GenelGrupluVeriSeti tablosu örneği ..... 56
**Tablo 19.** Anahtar değer hesaplamalarına örnek 20 kayıtlık tablo ..... 62
**Tablo 20.** UygulamaTablosu tablosu örneği ..... 64
**Tablo 21.** UygulamaTablosu tablosunun içinde bulunan alanlar listesi ..... 68
**Tablo 22**. UygulamaTablosu alanlarının veri çeşitliliği ..... 69
**Tablo 23.** Ham veri üzerinde yapılan gün tahminlerinin sapma sayıları ..... 75
**Tablo 24.** İşlenmiş uygulama verisi üzerinde bölge gruplarının anahtar aralıkları ..... 76
**Tablo 25.** Ham veri üzerindeki kayıtlara göre yerleştirilmesi gereken bölgeler ve uygulamanın tahmin ettiği bölgeler ..... 78
**Tablo 26.** Programın bölgelere göre önerdiği kayıt sayıları tablosu ..... 79
**Tablo 27.** Depodaki lokasyonların doluluk bilgileriyle ilgili bazı sayısal veriler ..... 82

**Tablo 28.** Örnek olarak seçilen X kümesinin 1 aylık durumu ..... 83
**Tablo 29.** Örnek olarak seçilen Y kümesinin 1 aylık durumu ..... 84
**Tablo 30.** Makine öğrenmesi yöntemlerinin ulaşılan ortalama başarı oranları..... 90
**Tablo 31.** Literatürdeki bazı yöntemlerin DSTİYALU ile karşılaştırılması ..... 93

# ŞEKİLLER LİSTESİ

**Şekil 1.** Sırt sırta raf sisteminin temel yapısı ..... 11
**Şekil 2.** Sırt sırta raf sistemi örneği ..... 12
**Şekil 3.** Çapraz geçitli sırt sırta raf sisteminin temel yapısı ..... 12
**Şekil 4.** Tompkins ve arkadaşlarına göre tipik depo fonksiyonları ve akışı ..... 13
**Şekil 5.** Frazelle'e göre tipik bir ticari deponun akış şeması ..... 14
**Şekil 6.** Modern bir ticari deponun örnek fiziki yerleşim planı ..... 14
**Şekil 7.** Sipariş toplama sistemlerinin sınıflandırılması ..... 15
**Şekil 8.** Ürün yerleştirme esnasında kat edilen mesafelere bir örnek ..... 17
**Şekil 9.** Sipariş toplama işlemi esnasında kat edilen mesafelere bir örnek ..... 21
**Şekil 10.** Bir depoya gelen müşteri sipariş örnekleri ..... 22
**Şekil 11.** LDS sisteminde hazırlanmış örnek bir toplama listesi ..... 23
**Şekil 12.** Örnek bir toplama senaryosu ..... 24
**Şekil 13**. Barsan Global Lojistik'in depolama tesisleri ..... 27
**Şekil 14.** BGL Willebroek tesisleri ..... 31
**Şekil 15.** LDS programının ana ekranından bir görüntü ..... 33
**Şekil 16.** LDS'nin çalıştığı mobil cihazlar ve LDS'nin el terminali versiyonunun bir görüntüsü ..... 34
**Şekil 17.** Örnek veritabanındaki kutu giriş okutmalarının sayısı ..... 35
**Şekil 18.** Örnek veritabanındaki kutu çıkış okutmalarının sayısı ..... 35
**Şekil 19.** Sirkülasyon hızının mesafe üzerindeki etkisine 1. örnek ..... 36
**Şekil 20.** Sirkülasyon hızının mesafe üzerindeki etkisine 2. Örnek ..... 37
**Şekil 21.** BlokeAciklama alanıyla ilgili veri seti üzerinden çekilen sorgu ve sonucunun ekran görüntüsü ..... 47
**Şekil 22.** Ham veri setinden yorumlanmış veri seti oluşturulurken kullanılan kod ve sonuç ..... 52
**Şekil 23.** UruneGoreGrupluVeriSeti tablosunun oluşturulması ..... 53
**Şekil 24.** Standart dışı kayıtların 183 sıra numaralı örnek üzerinden grafik gösterimi ..... 55
**Şekil 25.** GenelGrupluVeriSeti tablosunun oluşturulma süreci ve sonucu ..... 56
**Şekil 26.** Uygulama akış diyagramı ..... 59

**Şekil 27.** Anahtar değeri oluşturma algoritmasının akış diyagramı ..... 61
**Şekil 28.** Uygulama için anahtar değeri belirlenmiş ve bu değere göre sıralanmış UygulamaTablosu isimli tablonun yaratılması ..... 63
**Şekil 29.** Willebroek tesisleri için oluşturulan A, B, C, D, E bölgeleri ..... 65
**Şekil 30.** LDS programı üzerinde yapılan bölge grup tanımları ..... 66
**Şekil 31.** UygulamaTablosu'na BolgeGrup isimli bir alan eklenerek bölgesel gruplandırma yapılması ..... 67
**Şekil 32.** UrunBolgeselGrupBelirle isimli fonksiyonun kodları ..... 70
**Şekil 33.** Fonksiyonun çalışma zamanı örneği 1 ..... 71
**Şekil 34.** UrunIcinLokasyonIdBul ismiyle hazırlanan Delphi fonksiyonunun kodları .. 72
**Şekil 35.** UrununLokasyonunBolgeselGrubunuKontrolEt ismiyle hazırlanan Delphi fonksiyonunun kodları ..... 73
**Şekil 36.** Ham veri üzerinde günlük analiz için çalıştırılan kod ..... 75
**Şekil 37.** Ham veri üzerindeki tahmin sapmalarının grafiksel gösterimi ..... 76
**Şekil 38.** Anahtar bazında ham veri ile yapılan karşılaştırmalı analizin kodları ..... 77
**Şekil 39.** Ham veri üzerindeki kayıtlara göre yerleştirilmesi gereken bölgeler ve uygulamanın tahmin ettiği bölgeler tablosunun grafik gösterimi ..... 79
**Şekil 40.** Program tarafından önerilen dağılım sayılarının grafik gösterimi ..... 80
**Şekil 41.** 15 Mart 2022'de Willebroek tesisleri lokasyon doluluk şeması ..... 81
**Şekil 42.** 15 Nisan 2022'de Willebroek tesisleri lokasyon doluluk şeması ..... 81
**Şekil 44.** MLVeriSeti isimli tablonun hazırlanması için gerekli SQL kodu ..... 88
**Şekil 45.** ML yöntemleri karşılaştırma programından bir görüntü ..... 88
**Şekil 46.** Testi gerçekleştiren ve sonuçları ekrana yazdıran Python kodu ..... 89
**Şekil 47.** Uygulamanın çıktılarının örnek gösterimi ..... 89

# KISALTMALAR

**BGL** : Barsan Global Lojistik

**DC** : Distribution Center (dağıtım merkezi)

**DSTİYALU** : Depolarda Sipariş Toplama İşlemlerine Yönelik Akıllı Lokasyonlama Uygulaması

**ERP** : Enterprise Resource Planning (kurumsal kaynak planlaması)

**ET** : El Terminali

**IQR** : Inter Quantile Range (çeyrekler açıklığı)

**LDS** : Lojistik Depo Sistemi (yazılım)

**ML** : Machine Learning (makine öğrenmesi)

**MLP** : Multi-Layer Perceptron (çok katmanlı ağlar)

**MS-SQL** : Microsoft SQL Server (bir veritabanı yönetim yazılımı)

**PCA** : Principal Component Analysis (temel bileşen analizi)

**SKU** : Stock Keeping Unit (stok saklama birimi)

**SQL** : Structured Query Language (yapılandırılmış sorgu dili)

**T-SQL** : Transact-Structured Query Language

**WMS** : Warehouse Management System (depo yönetim sistemi)

**YSA** : Yapay Sinir Ağları

# BÖLÜM 1. GİRİŞ

Depolama; bir materyalin, tedarik edilip ulaşımı tamamlandıktan sonra, o ürünün veya hammaddenin üretim veya sipariş aşamasında kullanılmasına dek saklanması ve ihtiyaç duyulduğu anda sevke hazır hale getirilmesidir.

Etkin bir depolama yönetimi; materyalin hızlı bir şekilde muhafaza altına alınması ve talep edildiği anda hızlı bir şekilde tekrar çıkış işleminin gerçekleştirilmesi ile mümkün olur. Bir materyal, depolarda birkaç saat durabileceği gibi yıllarca saklanması da gerekebilir. Yapılan operasyona göre materyallerin çok kısa sürede girip çıktığı alanlara genel olarak dağıtım merkezi (distribution center, DC) ismi verilirken, uzun süreli muhafaza altında tutuldukları alanlara depo (warehouse) ismi verilmektedir.

Günümüzde, depolama faaliyetlerinin tedarik zincirinin önemli bir parçası olduğu genel olarak kabul görmüştür. Depolama esnasında yapılacak başarılı konsolidasyon işlemleri, ürünlerin ulaşım ve üretim maliyetlerinde ölçeklenebilir düşüşler oluşturabilmektedir (Ramaa, Subramanya ve Rangaswamy, 2012).

Edward Frazelle'nin 90'lı yıllarda geliştirdiği RightChain modeline göre; başarılı bir tedarik zinciri yönetimi müşteri hizmetleri, envanter yönetimi, tedarik, nakliye ve depolama süreçlerinin doğru bir şekilde yönetimi ile mümkündür. (Frazelle E. , 2015)

Depolama süreçlerinin etkin yönetiminde bazı temel kriterler mevcuttur. Bunlardan bazıları fiziki ve insani süreçlerdir. Depo yerinin seçimi, depo için ayrılmış alanın boyutları, deponun mülkiyeti, personelin yeterliliği gibi konular ve bu hususlarda yapılacak seçimler bu tür süreçlere örnek olarak verilebilir.

Depoların etkin yönetiminde fiziki ve insani birçok etken bulunduğu gibi; bilgisayarlar tarafından yönetilen süreçler de yer alabilmektedir. Mal kabulü ve sevkiyat süreleri, depolama ekipmanlarının etkin ve verimli kullanımı, çalışan ve ekipman masraflarında tasarruf sağlanması, sayım ve kontrol süreçleri gibi birçok depolama faaliyetinde bilgisayar ve yazılım müdahaleleri söz konusudur.

Günümüz koşullarında depoların yönetim ve planlama süreçlerinin etkin yönetiminde çeşitli bilgisayar programları kullanılmaktadır. Bunlar başka bir programın parçası olabilecek yazılım parçacıkları olabileceği gibi tamamen bağımsız yazılımlar da olabilir. Depo yönetim sistemi (Warehouse Management System, WMS) ismi verilen bu yazılımlar depoların verimliliği için belirleyici roller üstlenmişlerdir.

## Literatür Taraması

Sipariş toplama problemine yönelik çeşitli yaklaşımlar ve çözüm önerileri bu çalışma kapsamında incelenmiş ve analiz edilmiştir.

Caron vd (1998) yaptıkları çalışmada, devir bazlı yerleştirme yaklaşımını temel alan yerleşimin sonuçlarını **S şekilli** ve geri dönüş **sezgisel rota belirleme** algoritmaları ile oluşturulan rotaların mesafelerini karşılaştırmışlardır (Caron ve Marchet, 1998).

De Koster vd (1999) yaptıkları çalışmada, toplayıcıların farklı kapasitelerle çalıştığı varsayımıyla; çeşitli sipariş hacimleri kullanmışlar ve yüksek kapasitelerdeki toplayıcılarda **s şekilli yöntem** ve **çekirdek algoritmalarının** en verimli sonucu verdiğini; düşük kapasiteli toplayıcılarla ise en büyük boşluk yöntemi ve tasarruf algoritmalarının verimli sonuçlara ulaştığını ifade etmişlerdir (De Koster, Van der Poort, ve Wolters, 1999).

Roodbergen vd (2001) yaptıkları çalışmada **rastgele yerleştirme** yönteminin kullanılmış olduğu depolarda, çok çeşitli sipariş büyüklüklerinde **çapraz geçitlerin** sipariş toplama problemi üzerindeki etkilerini incelemişlerdir (Roodbergen ve Koster, 2001).

Petersen vd (2004) çalışmalarında siparişlerin kabul edildikten sonra hazırlanan toplama listelerinin toplama mesafeleri üzerindeki etkilerini değerlendirmişlerdir (Petersen, Aase ve Heiser, 2004).

Chen vd (2005) yapmış oldukları çalışmada büyük veri tabanlı bazı incelemeler yapmış ve yeni bir özel algoritma önermişlerdir. Algoritmalarını başka çekirdek algoritmalar ile mukayese eden Chen vd, kendi algoritmalarının karşılaştırılan çekirdek algoritmalardan daha kısa mesafeler ortaya koyduğunu tespit etmişlerdir. Ancak söz konusu algoritmanın çalıştırılması oldukça uzun sürmektedir (Chen ve Wu, 2005).

Hwang vd (2006) tarafından yapılan çalışmadaysa, depolardaki sipariş toplama problemine iki aşamalı ve yeni bir çözüm önerisi ortaya atmışlardır. Çalışmanın ilk aşamasında belirli bir periyotta talep edilen ürünlerin ve diğer bazı kısıtların değişken olarak alındığı yeni bir matematiksel model geliştirilmiştir. İkinci aşamada ise depo tasarımı ile ilgili değişkenlerin de hesaba katıldığı ek bir çalışma ortaya atılmıştır (Hwang ve Cho, 2006).

Pan vd (2008) benzer çalışmalardan farklı olarak sadece tek toplayıcının olduğu senaryolardaki problemi değil, birden fazla toplayıcıyla çalışan depolardaki sorunları da incelemişlerdir. Araştırmacılar, depo içindeki raflar arasındaki geçişlerde ortaya

çıkabilecek trafik bazlı sorunları da çözmeye yönelik bir çalışma yapmışlardır. Bu trafik sorununu da hesaba katarak birim zamanda en fazla sipariş tamamlama üzerine araştırma yapmışlardır (Pan ve Shih, 2008).

Roodbergen vd (2009) yaptıkları çalışmada sipariş toplama problemiyle ilgili bir literatür araştırması yapmışlardır. Araştırmacılar, parçalardan toplayıcıya yöntemiyle çalışan depolara yönelik yaptıkları bu araştırmada ürün yerleşimi, sipariş toplama, otomatik ürün gezinti sistemleri ve bu sistemlerin gezinme mesafelerinin tahmin edilmesine yönelik çözüm önerilerini incelemişlerdir (Roodbergen ve Vis, 2009).

Henn vd (2012) çalışmalarında tabu arama metodu kullanarak meta sezgisel gruplama yöntemleri önerisinde bulunmuşlardır. Çeşitli büyüklükte sipariş büyüklükleri ile çalışan araştırmacılar, tasarladıkları yöntemi mevcut bazı algoritmalar ile geliştirmiş ve başka yöntemlerle karşılaştırmışlardır. Araştırmacıların ortaya attığı yöntemin karşılaştırılan yöntemlere göre toplam seyahat mesafesini azalttığı ve algoritmanın daha kısa sürede çalıştığı ortaya konmuştur (Henn ve Wäscher, 2012).

Tuna vd (2012) yaptıkları literatür araştırmasında da belirtildiği gibi, mal kabul ve ürün yerleştirme süreçlerine yönelik araştırmalarda ortaya atılan sistemlerde daha çok, otomasyona dayalı sistemler göz önünde bulundurulmuş, genel olarak **parçadan toplayıcıya** usulüyle çalışan depolara dikkat çekilmiştir. **Toplayıcıdan parçaya** yöntemine göre çalışan ve otomasyon sistemlerinin daha az etkin olduğu depolarda ise çoğunlukla **rastgele yerleştirme** metodu tercih edilmiştir (Tuna ve Tunçel, 2012).

Özçakar vd (2012) çalışmalarında toplayıcının seyahat süresini azaltmak amacıyla **gruba dayalı genetik** algoritma modelini açıklamışlar ve bu modeli klasik bir gruplama ve rotalama metodu olan **Clarke-Wright** sezgiseli ile karşılaştırmışlardır. Araştırmacılar önerdikleri algoritmanın, test edilen bütün veri setleri için Clarke-Wright sezgisel algoritmasından daha iyi sonuçlar ortaya koyduğunu belirtmişlerdir (Özçakar, Görener ve Arıkan, 2012).

Şahin vd (2013) çalışmalarında sipariş toplama hazırlıklarına ve toplama aracının rotalandırılması ile ilgili algoritmalarla ilgili araştırmalar yapmışlardır. Araştırmacılar çalışmalarında genetik algoritma tabanlı iki yeni varyasyon önermişlerdir. En yakın komşu ve kazanç isimli iki yeni genetik algoritma tabanlı yeni yöntem öneren araştırmacılar, farklı sipariş sayıları, ağırlık ve yerleşim koşullarında deneysel problemler oluşturmuşlar ve önerdikleri algoritmayı test etmişlerdir (Şahin ve Kulak, 2013).

Baray vd (2014) yaptıkları çalışmada birden çok katlı, z eksenli depolar için **parçacık sürü optimizasyonu** algoritması tabanlı bir tasarım metodolojisi önermişlerdir. Araştırmacılar, söz konusu çalışmalarında parçacıkları, kullandıkları algoritma vasıtasıyla kabul edilebilir çözüm uzayı içinde tutmaya çalışmışlardır (Baray ve Çakmak, 2015).

**Problem**

Dünya genelinde tüketime dayalı ekonomik yapıyla birlikte daha hızlı üretim ihtiyacı ve bu üretimin tedarik zinciri yönetimi ile bütünleşmesi neticesinde depolama faaliyetlerinin önemi daha da artmıştır. Daha etkili bir envanter yönetimi ve stok üzerinde daha yüksek doğrulukta bir kontrol, daha hızlı yerleştirme ve toplama süreleri ve ürün çeşitliliğinin doğru yönetimi gibi ihtiyaçlar ortaya çıkmış bu da depo yönetim süreçlerinin daha dikkatli incelenmesi ve bu konuda çıkacak problemlere yeni yaklaşımlar getirilmesi ihtiyacını doğurmuştur (Gu, Goetschalckx ve McGinnis, 2007).

Modern depoların ana işlevi tedarikçilerden stok birimlerini (stock keeping unit, SKU) almak, bu birimleri depolamak, müşterilerden gelen siparişleri almak ve bu siparişlere göre depodaki birimleri sevkiyat için bir araya getirip müşterilere sevk etmektir (Gu, Goetschalckx ve McGinnis, 2007).

Bu işlevleri gerçekleştirmek için depolarda birçok operasyon gerçekleştirilmektedir. Bu operasyonların belli başlıları; malzeme kabulü, malzemenin depolanması, siparişlerin toplanması, malzemenin sevkiyatı şeklinde dört ana gruba ayrılabilir (Baray ve Çakmak, 2015).

Depo yönetiminde bugüne kadar belirlenmiş ana sorunlar; envanter yönetimiyle ilgili sorunlar ve depo yeri belirlenmesi ile ilgili sorunlardır. Envanterin sağlıklı bir şekilde yönetilmesi maliyetleri ciddi biçimde azaltmakta aynı zamanda sipariş toplama süreçlerindeki verimliliği de arttırmaktadır. Aynı zamanda depolama yerinin atanması politikası sipariş toplama sürelerini azaltmakta, iş hacmini ve verimliliği arttırmaktadır (Van den Berg ve Zijm, 1999).

Depo yönetiminde ortaya çıkabilecek problemleri analiz edebilmek için modern depoların işlevini ve bu kapsamda içerdikleri operasyonları ele almak faydalı olacaktır. Üzerinde çalışılan ortamdaki süreçler, fonksiyonlar ve operasyonları derinlemesine

tanımak, problemin doğru tespit edilmesinde ve etkin çözümünde hayati bir role sahip olmaktadır.

**Amaç**

Bu çalışmanın temel amacı, daha önce de ele alınan bir sorun olan sipariş toplama problemine, gelişen teknolojik araçları da kullanarak, yeni bir yaklaşım getirmektir. Sipariş toplama probleminde geçmişte önerilen çözüm yöntemlerini incelemek, bu yöntemlerin uygulanabilir olduğu ve uygulanamaz olduğu senaryoları gözden geçirmek ve gelişen yeni teknolojileri de kullanarak sipariş toplama sürecini ele almak amacı güdülmüştür. Sipariş toplama sürecinin büyük oranda mal kabulü esnasında yapılan adreslemeye dayandığı varsayılarak, sürecin toplama esnasında veya öncesinde değil, daha ürünler depolara girerken başlatılması gerektiği vurgusu temel alınmıştır.
Bu şekilde problemin yeniden ele alınması, problemin çözümünde bugüne kadar kullanılan algoritmaların incelenmesi, bu bakış açısına uyarlanması amaçlanmıştır. Problemin çeşitli aşamalarında mevcut çalışmaların veya daha önce ortaya atılan algoritmaların yetersiz görülmesi durumunda, yeni bir algoritmanın belirlenmesi, uygulanıp sonuçların değerlendirilmesi, örneklerle desteklenmesi ve algoritmanın detaylarının sonradan kullanılabilir şekilde açıklanması da amaçlar arasındadır.

**Önem**

De Koster vd (2007) sipariş toplama maliyetlerinin, toplam depo süreçlerindeki maliyetlerinin yaklaşık %55'ini oluşturduğunu belirtmişlerdir. Bu operasyon esnasında kazanılacak zaman ve düşürülecek diğer maliyet kalemlerinin toplam depo maliyetlerine etkisinin önemini vurgulamışlardır (De Koster, Le-Duc ve Roodbergen, 2007).
Yine Tompkins vd (2010) depolama alanından, ürünlerin müşterinin taleplerine göre toplanması sürecini, depolama faaliyetlerinin "en önemli aşaması" olarak kabul ediyorlar (Tompkins, White, Bozer ve Tanchoco, 2010).
Bartholdi vd (2014) sipariş toplama sürecinin %55'inin gezintiyle, %15'inin ise arama ile geçirildiği sonucuna ulaşmışlardı (Bartholdi ve Hackman, 2014). Araştırmacıların sipariş toplama zamanının hangi aktivite ile geçirildiğine dair oluşturdukları yüzdelik paylar **Tablo 1**'de gösterilmiştir.

**Tablo 1.** Sipariş toplama sürecinde çeşitli aktivitelerin zaman payı

| Aktivite | Zaman Payı |
|---|---|
| **Gezinti** | 55% |
| **Arama** | 15% |
| **Ayıklama** | 10% |
| **Kâğıt işleri ve diğer aktiviteler** | 20% |

Bu araştırmalar ve ulaşılan rakamlar, sipariş toplama süreçlerindeki optimizasyonların depolama faaliyetlerinde yaratabileceği kazançları ortaya koymaktadır. Bu alanda yapılacak her geliştirme hem lojistik firmalarına hem de tedarik zinciri vasıtasıyla bütün ekonomiye zaman ve maliyet tasarrufu olarak yansıyacaktır.

## Varsayımlar

Problemi ortaya koyarken de bahsedildiği gibi, bu çalışmada ele alınan problemin çok daha genel bir hali incelense de çalışmada kullanılan örnekler çoğunlukla belli bir depo tipinde yapılmıştır. Günümüzde çok kullanılan çapraz geçitli sırt sırta raf sisteminin var olduğu ve toplayıcıdan parçaya sistemine dayalı otomasyonun sürece görece az etkili olduğu depolardaki örnekler ve sonuçlar incelenmiştir.

## Sınırlıklar

Çalışma esnasında kullanılan bütün örnekler ve deneyler sırt sırta dayalı çapraz geçitli raf sistemi kullanan ve konveyör sistemlerinin bulunmadığı depolarla ilgili olduğu için; konveyör sistemlerinin aktif olarak kullanıldığı veya daha çok otomasyona dayalı depolarda istenilen sonuçları vermeyebilir. Burada gerçekleşen senaryolarla ilgili elde yeterince örnek yoktur.

## Tanımlar

### Depo

Ticari değeri olan ürünlerin muhafazası amacıyla yapılmış özel yapılardır (Shiau ve Lee, 2010). Mamul mal üreticileri, ticari işletmeler, ithalat ve ihracat firmaları, toptan satıcılar, nakliye firmaları gibi pek çok sektör tarafından kullanılmakta olan depolar, daha çok şehirlerin endüstriyel bölgelerinde yer almaktadır. Depo ayrıca demiryolu, deniz taşımacılığı ve hava yolu taşımacılığı bünyesinde yükleme ve boşaltma mekanları olarak

da kullanılabilmektedir. İçerisindeki mal sirkülasyonunun sağlanabilmesi amacıyla vinç, forklift gibi araçlar ve buna yardımcı palet adı verilen genellikle ağaç hammaddeli altlıkların kullanımı önemlidir. Çalışmada bahsedilen depo ifadesi, ticari bir ürünü saklamak için özel olarak tasarlanmış, ihtiyaçlara göre her zaman yeni ürünlerin gelip mevcut ürünlerin çıkışının yapıldığı bağımsız yapıları kastetmektedir.
Depolar kendi içinde geleneksel depo (warehouse), dağıtım merkezi (distribution center), toplama merkezi (consolidation center) ve aktarma merkezi (transshipment center) gibi gruplara ayrılabilir (Tanyaş, 2015). Bu çalışmada genel olarak geleneksel depolar incelenecektir.

**Warehouse Management System**

Türkçesi Depo Yönetim Sistemi, kısaltması WMS olan yönetim bilişim sisteminin genel adıdır. WMS bir ambarda gerçekleşebilecek tüm operasyonları etkin ve verimli şekilde yönetmek, operasyonların kâğıtsız, en kısa sürede, en az insan ve makine kaynağı ile hatasız tamamlanmasını sağlamak, karar vermek (decision support system), gerçekleştirilen operasyonları gerçek zamanlı olarak izlemek ve gerçekleştirilmiş işlemlerin bilgilerini raporlar halinde sunmak üzere tasarlanmış bir sistemdir. WMS tek başına bir yazılım olabilmekle beraber bir ERP sisteminin modülü olarak da çalışabilir. Depo süreçleri, şirketlerin diğer süreçleriyle (finans, satış vs.) bağlantılı olduğundan WMS 'de diğer süreçlerde kullanılan otomasyon programlarıyla bütünleşik çalışabilmektedir (Baruffaldi, Accorsi ve Manzini, 2019).

**Raf**

Raf, sergilenen, depolanan veya satışa sunulan eşyaları tutmak için evde, konutta, işletmede, mağazada veya başka yerlerde kullanılan düz yatay bir düzlemdir. Bu çalışmada bahsedilen raf ise depolarda kullanılan; ürün, kutu veya paletlerin yerleştirildiği; metal, ahşap veya plastik gibi malzemelerden yapılan ekipmanlardır.

**Lokasyon**

Genel olarak uzaydaki belirli bir noktayı ifade etmek için kullanılsa da bu çalışmada depodaki belirli bir alanı adreslemek için kullanılmıştır. Depoda bölünebilen en küçük adresi temsil eder. ÖRNEK: A koridorunun 12. Sırasının 2. Katı gibi.

**Forklift**

Ağır yükleri çatalları aracılığıyla kaldırmak ve özellikle bir araca ya da rafa yüklemek için kullanılan bir çeşit iş makinesidir. Forkliftler daha çok paletlerin üzerine yüklü ağırlıkları taşımak, kaldırmak ve istif etmek için kullanılır.

**Elleçleme**

Gelen veya giden ürünün gözetim altında, özelliklerini değiştirmeden veya herhangi bir zarar vermeden yerlerinin değiştirilmesi, farklı bir kutuya ve/veya farklı bir şekilde dizilmesi, etiketlenmesi, tamir edilmesi, havalandırılması veya karıştırılması gibi işlerin geneline verilen isimdir. Daha çok nakliye, gümrük ve kargo işlerinde kullanılan bir terimdir. İngilizce "handling" olarak ifade edilmektedir. St-Vincent vd iddiasına göre elleçleme faaliyetleri depodaki vardiya içinde adam/saat faaliyetlerinin %74'üne kadar bir zaman alabilmektedir (St-Vincent, Denis, Imbeau ve Laberge, 2005).

**Clarke-Wright Algoritması**

Genellikle rotalama problemlerinde kullanılan sezgisel bir yöntem algoritması. 1964 yılında Clarke ve Wright tarafından ortaya atılmıştır. Bir adet depo ve birden fazla dağıtım noktası arasındaki mesafe problemlerini inceleyen bir algoritmadır (Yazgan, Cömert ve Kılıç, 2020).

İki dağıtım noktasının a ve b olduğunu varsayalım. Dağıtım yapan aracın önce a'ya gidip, geri dönüp sonra b'ye gidip tekrar geri dönmesi durumunda ortaya çıkan formül şu olacaktır: $d(1) = c(0a) + c(a0) + c(0b) + c(b0)$

Aracın önce a'ya gidip, oradan b'ye gidip sonra dağıtım noktasına dönmesi durumunda ise şu formül ortaya çıkacaktır: $d(2) = c(0a) + c(ab) + c(b0)$

İki yöntem arasında sağlanan tasarrufun formülü ise şudur: $s(ab) = d(1) - d(2)$

Clarke-Wright sezgisel algoritmasına göre bütün ikili dağıtım noktaları ele alınır, büyükten küçüğe sıralanır ve kapasiteler göz önüne alınarak rotalar oluşturulur.

**Meta-Sezgisel Algoritmalar**

Yunanca "meta" ve "heurestic" kelimelerinden türetilmiş meta-sezgisel terimi, kelime anlamı olarak "sezgiselden daha ileri" demektir. Büyük bölümü, sezgisel algoritmalar

üzerine doğadan esinlenilerek yeni eklemeler yapılarak oluşturulmuştur (Shadkam ve Bijari, 2014). Örneğin kuşların besin arama davranışlarının gözlemlenmesine dayanan "parçacık sürü optimizasyonu" ve ateş böceklerinin aralarındaki etkileşimlerden yola çıkılarak geliştirilen "ateş böceği algoritması" bu tip algoritmalara örnek olarak verilebilir.

**ABC Analizi**

İtalyan ekonomist Vilfredo Frederico Damaso Pareto tarafından geliştirilen Pareto Prensibi'nden yola çıkılarak geliştirilen sınıflandırma tekniği. Etkin stok yönetimi, maliyet düşürülmesi ve üretim planlama gibi alanlarda kullanılan basit ve etkili bir tekniktir. Müşteri talepleri miktar ve karakteristiğine göre A, B ve C tipi olarak büyükten küçüğe üç temel gruba ayrılır. A tipi ürünler adet ve ciro anlamında kapasitenin en büyük bölümünü ama kalem sayısı bazında en küçük bölümünü ifade ederler ve "koşan ürünler" olarak adlandırılırlar. C tipi ürünler ise düşük talep görmesine rağmen kalem sayısı anlamında büyük yer kaplarlar. Bu tür ürünlere ise "emekleyen ürün" ismi verilir. Bu ürünler üzerinde yapılan Pareto Analizi vasıtasıyla stok yönetim politikalarına yön verilebilmektedir (Dursun ve Gürgen, 2020).

**Delphi Tekniği**

Adını Antik Yunan'daki kahinler şehri Delphi'den alan bu yöntem, bir tartışmada konu hakkında uzman kişilerin, anonim olarak fikirlerinin toplanmasına dayanır. Sorun çözümlemede uzmanların fikirlerini almaya dayalı bir metot. Ancak bir konu hakkında bir uzmana danışmak olarak düşünülmemelidir. Öncelikle bir dizi uzmanın bulunması ve uzmanların birbirlerinden haberdar olmaması gibi çeşitli prensipleri vardır. 1950'li yıllarda Olaf Helmer ve Normal Dalkey tarafından askeri konulardaki bazı araştırmalarda kullanılmak amacıyla geliştirilmiştir (Şahin, 2001).

**Principal Component Analysis (PCA)**

Orijinali Principal Component Analysis olan ve Türkçe'ye Temel Bileşenler Analizi olarak çevrilen, analiz tekniğidir. Tanıma, sınıflandırma ve sıkıştırma alanlarında kullanılan bir tekniktir. Temel amacı büyük boyutlu verilerde en büyük varyansları

kullanarak boyut indirgemeyi sağlamaktır. Veriyi bir dizi tekrarlanan işlem sonucunda temel bileşenlerine indirgemek için kullanılır (Kılıç, Özgönenel, Usta ve Thomas, 2009).

### Aykırı Değer (Outlier)

İngilizce karşılığı "**outlier**" olan, istatistik biliminde diğer gözlemlerden önemli ölçüde farklı olan veri noktasıdır. Aykırı değer ölçümden, özel durumlardan veya hatalardan kaynaklanabilir ve veri setinden hariç tutulabilir. Aykırı değerler istatistik hesaplamalarında ciddi sorunlara yer açabileceği için göz ardı edilebilmektedir (Aydin ve Tasabat, 2017).

### Çeyrekler Açıklığı Metodu

İngilizce orijinal ismi **Interquartile Range Method** olan ve **IQR** harfleriyle kısaltılan, istatistik alanında aykırı değer bulmak için kullanılan matematiksel metottur (Barnes, 2021). Şöyle ki; öncelikle veriler artan sırada düzenlenmiştir. Birinci çeyrek (q1) ve üçüncü çeyrek (q3) içine giren kayıtlar tespit edilmiştir. Bu çeyrekler arasındaki mesafe $IQR = q3 - q1$ işlemiyle bulunmuştur. Bu durumda alt sınır $Q1 - 1.5(IQR)$ ve üst sınır $Q1 + 1.5(IQR)$ olarak belirlermiştir. Bu aralığın dışındaki değerler aykırı değer olarak sınıflandırılır.

### Makine Öğrenmesi

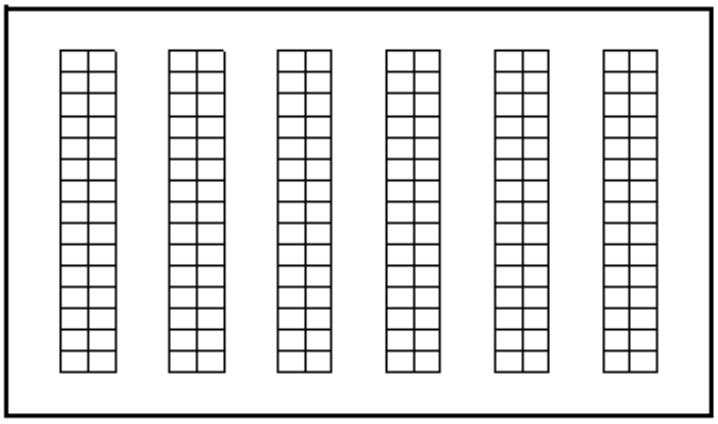
Bilgisayarların, çok net bir biçimde programlanmadan, görevlerini nasıl yerine keşfetmelerine makine öğrenmesi (Machine Learning – ML) ismi verilir. Belli bir grup veriden, yeni gelen veriler hakkında tahmin yürüten çeşitli algoritmalar makine öğrenmesi yönteminde kullanılırlar. Makine öğrenmesi yöntemi, bilgisayarlara verileri nasıl daha iyi işleyeceğini öğretmek için kullanılır. Bazen insanlar, veriler arttıkça karar vermekte zorluk yaşamaya başlarlar. Oysa gelişen teknolojiyle birlikte, bilgisayarların çok sayıda veriyle çalışması kolaylaşmış bu da makine öğrenmesine olan talebi arttırmıştır (Mahesh, 2018).

# BÖLÜM 2. ARAŞTIRMA ve PLANLAMA

## Depoların Genel Fiziki Yapısı

Depolardaki yerleşim problemlerini incelemek için öncelikle depoların genel yerleşim yapısını incelemekte fayda bulunur.

Günümüzün modern depolarında depoların fiziki yapısı; depo için ayrılmış bir alan ve bunların içine yerleştirilmiş raf sistemleridir. Bu raf sistemleri çeşitli yapılarda kurgulanabileceği gibi, fayda/maliyet analizleri neticesinde günümüz şartlarında en çok tercih edilen yöntem sırt sırta dayalı raf sistemleridir. Bu yöntemin kolay montaj, uygun maliyet, kolay bakım, verimli toplama gibi çeşitli avantajları bulunmaktadır (Özçakar, Görener ve Arıkan, 2012). Bu raf yerleşim sisteminin genel yapısı **Şekil 1**'de gösterilmiştir.



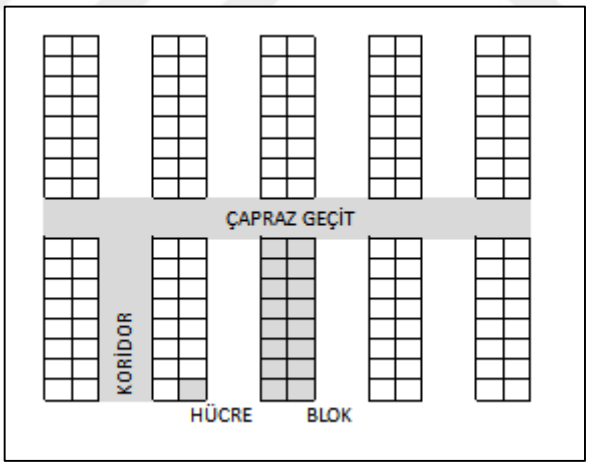
**Şekil 1.** Sırt sırta raf sisteminin temel yapısı

Bu raf sistemi, birbirine dayalı iki raf bölümünün dış taraflarında geçiş için boşluklar bırakarak depo alanını raflar ve koridorlar ile kaplama mantığına dayalıdır. Bu sistemin canlı bir örneği **Şekil 2**'de gösterilmiştir.



**Şekil 2.** Sırt sırta raf sistemi örneği

Bu raf sisteminin temel prensipleri aynı kalmakla birlikte, günümüzde bazı değişimler geçirmiştir. Orijinal haliyle tek bloklu olarak tasarlanan bu raflar, daha verimli kulanım ihtiyacı nedeniyle çapraz geçişli versiyonlara da evirilmiştir (Özçakar, Görener ve Arıkan, 2012). Bu sistemin temel yapısı **Şekil 3**'de gösterilmiştir.



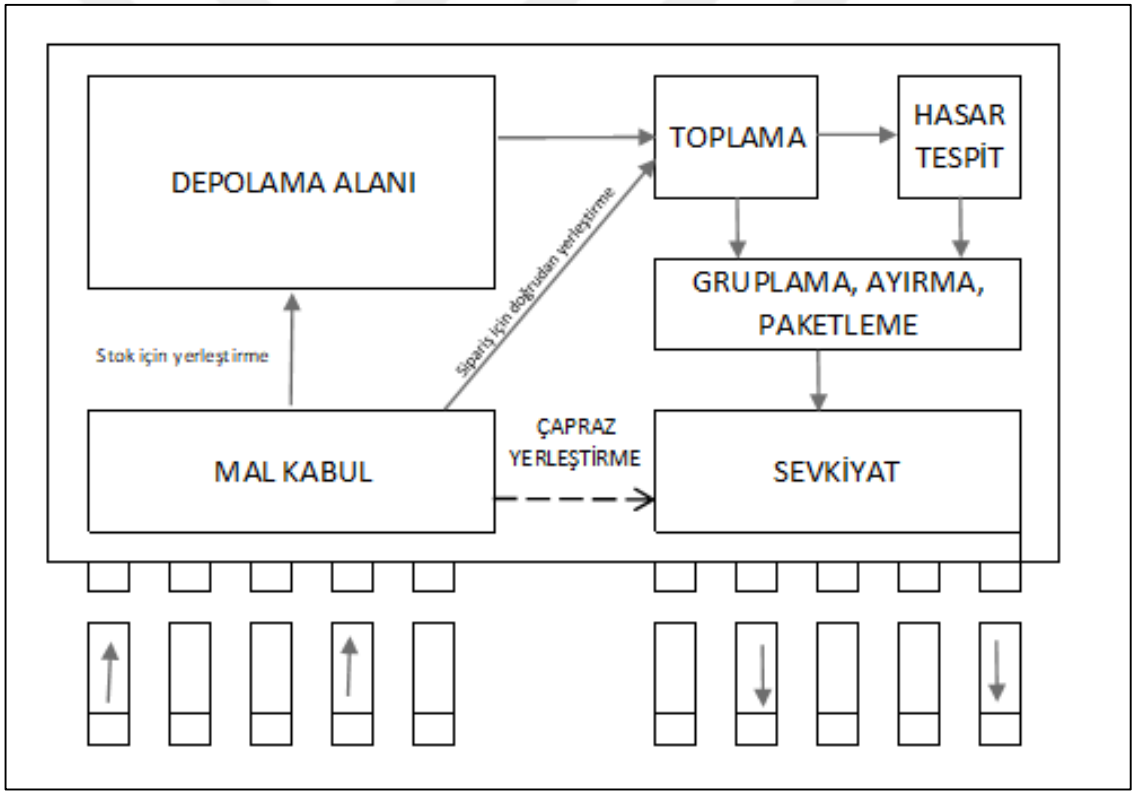

**Şekil 3.** Çapraz geçitli sırt sırta raf sisteminin temel yapısı

Depolama yönetiminde gerek envanter yönetiminde gerekse de depolama yeri belirleme ilgili yapılacak bütün çalışmalarda, üzerinde çalışılacak deponun fiziki yapısı ve yerleşim planının temel özellikleri dikkate alınmalıdır.

**Yerleştirme ve Toplama Döngüsü**

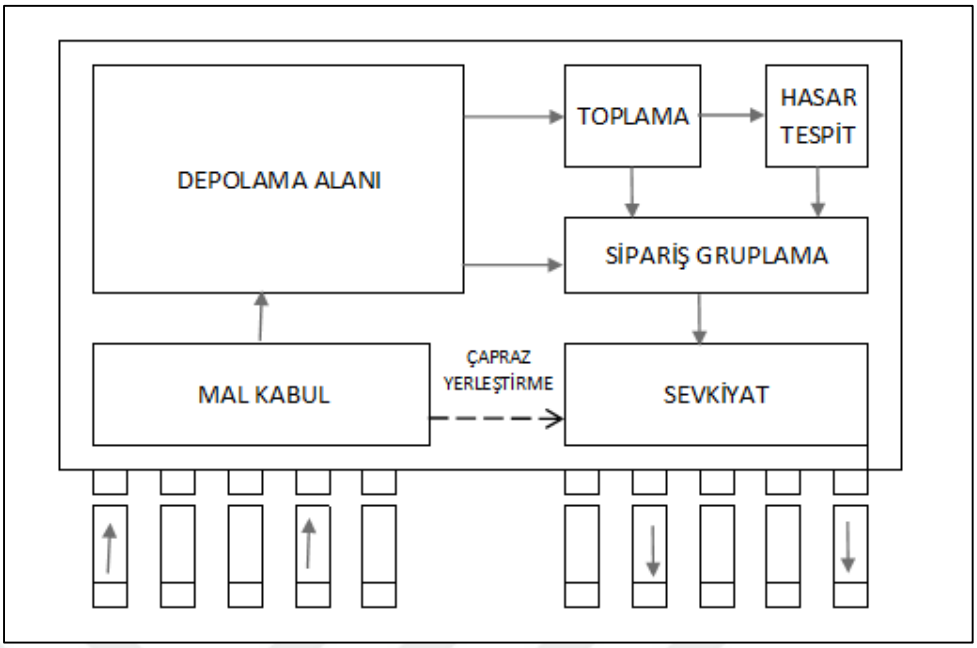
Modern depoların ürünlerin kısa veya uzun süreli saklandığı alanlar olduğundan bahsedilmişti. Kuşkusuz bu durum, depoların sadece ürün saklanan alanlardan oluştuğu sonucunu ortaya çıkarmaz. Depolarda devam eden materyal sirkülasyonu ve ek operasyonlara göre, depoların yerleşim planında ürün saklama alanları dışında ek bölümlerin de gerekliliğini doğurmuştur. Deponun fiziki yapısı, saklanan ürünün içeriği, ürünün saklanma süresi, yapılan operasyonların içeriği gibi konulara göre depodaki yerleştirme ve toplama döngüsünün tipi de değişmektedir.

Tompkins vd (2010) yazdıkları kitapta; tipik bir ticari deponun akış diyagramını, "Tipik depo fonksiyonları ve akışı" ismiyle yayınlamışlardır. Bu şema da **Şekil 4**'de gösterilmektedir (Tompkins, White, Bozer ve Tanchoco, 2010).



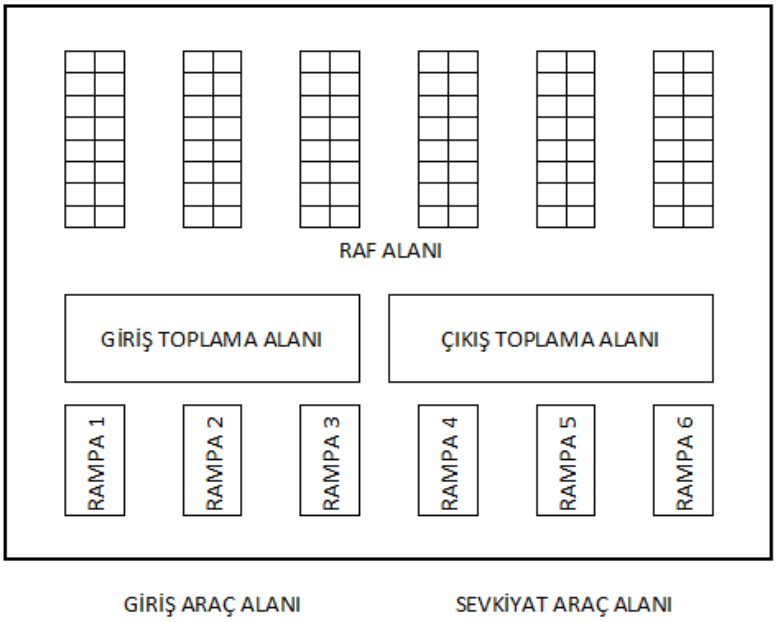

**Şekil 4.** Tompkins ve arkadaşlarına göre tipik depo fonksiyonları ve akışı

Frazelle (2015) de tipik bir ticari deponun akış diyagramını çok benzer bir şekilde bazı revizyonlarla yayınlamıştır. Frazelle'in hazırladığı diyagram **Şekil 5**'de gösterilmektedir (Frazelle E. , 2015).




**Şekil 5.** Frazelle'e göre tipik bir ticari deponun akış şeması

Bu akış diyagramlarına bağlı olarak tipik bir deponun yerleşim planı **Şekil 6**'da gösterilmiştir. Bu planda kabul edilen akışlara uygun olarak bir yerleşim örnek alınmış; yerleşim içerisinde yardımcı ekipmanlara ayrılan alanlar, ofis eşyaları için gerekli oda ve dolaplar, idari kadro için ayrılan ofis bölümleri gibi akış dışında kalan fiziki bölümler gösterilmemiştir.



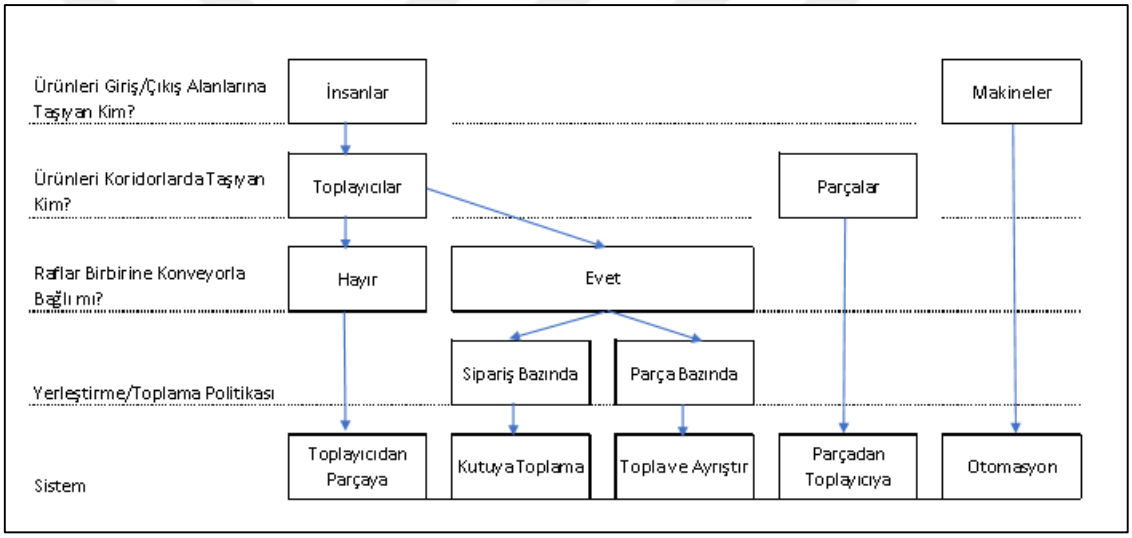

**Şekil 6.** Modern bir ticari deponun örnek fiziki yerleşim planı

Bu akış planının ve yerleşimin; depoların özel durumlarına göre çeşitli varyasyonları oluşturulsa da genel hatlarıyla bu model halen ticari depolarda çoğunlukla uygulanmaya devam etmektedir (Tunç, Kutlu, Zincidi ve Atmaca, 2008).

### Depo Operasyonlarının İcra Biçimleri

Depo akışı incelenen problemin temelini oluştursa da akış içindeki operasyonların icrası da problemin tanımı ve çözümü açısından öneme sahiptir.

Operasyonların icrası konusunda incelemeler yapan Dallari vd (2018), operasyonun çeşitli aşamalarında kullanılan yöntemlere göre bir sınıflandırma tablosu yapmışlardır (Dallari, Marchet ve Melacini, 2008). Bu icra biçimlerine göre oluşturulan sınıflandırma **Şekil 7**'de gösterilmiştir.




**Şekil 7.** Sipariş toplama sistemlerinin sınıflandırılması

Fark edilebileceği gibi akışın hangi aşamasında operasyonun hangi biçimde yapıldığı ürün yerleştirme ve toplama sistemlerinin etkin çalışması konusundaki problemin önemli bir parçasıdır.

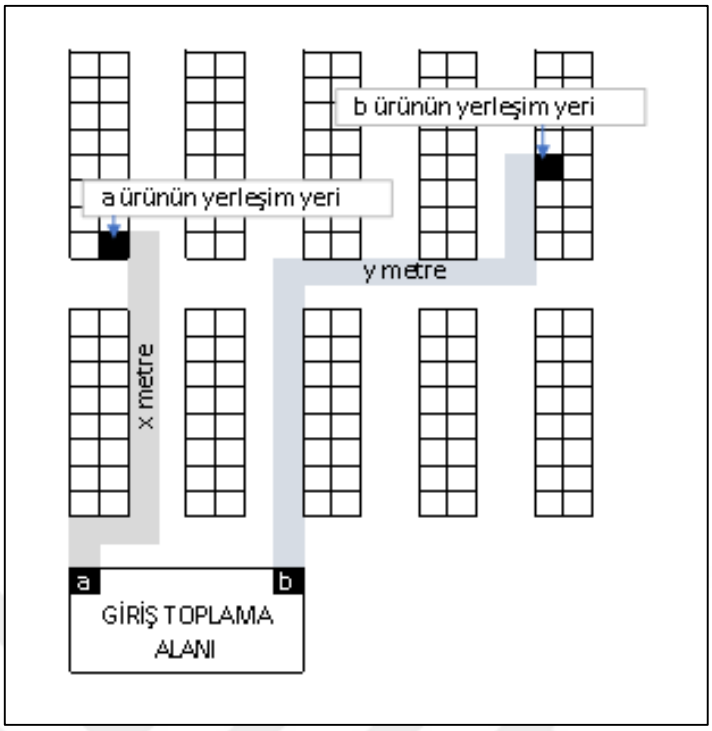
Operasyonların tamamının kontrolü makinelerde ise, makinelerin türü, çalışma algoritmaları, karşılaştıkları fiziki şartlar farklı olabileceği gibi; insanların operasyonu kontrol ettiği senaryolarda ise, insanın makineyi ne kadar kullandığı, hangi tip araçlarla operasyonları icra ettikleri de dikkate alınması gereken parametreler arasındadır.

Ancak gerek insan kontrollü sistemlerde gerekse makinelerin iş üzerinde daha fazla paya sahip olduğu sistemlerde, toplama alanları ile ürünlerin saklandığı alanlar arasındaki

mesafeler, bu mesafeleri oluşturan rotalar önemini kaybetmemekte, sadece bu rotalar arasındaki hız veya rotaların üzerinde hareket eden aktörler farklılaşmaktadır.
Bu rotaları oluşturan ihtiyaçları anlayabilmek için rotaların kullanıldığı operasyonların detaylarını ve rota yapılarını incelemek gerekecektir.

**Mal Kabul Operasyonu**

Çalışmada incelediğimiz depo akışı, mal/ürün kabul operasyonu ile başlamaktadır. Bu operasyon, bir materyalin depodaki bir rampaya, bir araç (kamyon, tır, konteyner gibi) ile getirilip mal kabul alanına indirilmesiyle başlamaktadır.
Ürün giriş toplama alanına yerleştirildikten sonra ürünün cinsine, deponun kullandığı icra biçimine gire toplama alanından ürün alınır. Daha önceden de belirtildiği gibi ürünü toplama alanından alıp taşıma işlemini yürüten bir robot, bir konveyör veya bir insan olabilir.
Ürün toplama alanından alındıktan sonra, bu alandan yerleştireceği rafa hareket ettirilir. Hangi rafa yerleştireceği ise bu çalışmanın genel konusu olduğu için, rafın belirlenme yöntemi bu aşamada dikkate alınmamıştır.
Ürünün yerleştirileceği raf belirlendikten sonra ürün, şartların izin verdiği, en kısa rota kullanılarak belirlenen rafa hareket ettirilir. Bu işlem, giriş toplama alanında ürün var olduğu sürece tekrarlanır.
Toplama alanında bulunan a ürünü için belirlenen yerleştirme noktasının giriş toplama alanına mesafesini x kabul edelim. Yine toplama alanındaki b ürünü için belirlenen yerleştirme noktasının toplama alanına mesafesini ise y kabul edelim. Bu durumda yerleştirici, a ürününü alacak x kadar mesafe kat edecek, sonrasında yine x kadar mesafe kat edip toplama alanına geri dönecektir. Bu esnada b ürününü alacak y kadar mesafe kat edip ürünü rafa götürecek, ardından yine y kadar mesafe kat edip toplama alanına geri dönecektir. Bu durumda a ve b ürünlerini yerleştirip tekrar toplama alanına dönmek için kat edilen mesafe 2x + 2y kadar olacaktır. Bunla ilgi bir örnek **Şekil 8**'de gösterilmiştir.




**Şekil 8.** Ürün yerleştirme esnasında kat edilen mesafelere bir örnek

Örnekte de görüldüğü gibi toplama alanındaki ürünlerin sayısı arttıkça kat edilmesi gereken mesafe de artacaktır. Verilen örnek insana dayalı toplama ve yerleşim yapan depolar için de robotlarla bu işlemleri yapan depolar için de geçerlidir. İki durumda da toplama alanı ile yerleşim alanı arasındaki mesafe en kısa yoldan kat edilmelidir. Bu esnada depo içinde başka yerleştirme ve toplama operasyonları yapan öğelerle karşılaşma gibi durumlar göz ardı edilmiştir. Bu durumlarda alternatif rotalar belirlenecek, bu rotalar en kısa mesafe olmayabilecektir. Ancak x ve y'nin değeri değişmekle birlikte 2x + 2y formülü değişmeyecektir.

İnsan ve robotlar dışında, konveyör kullanan depolarda da benzer bir formül söz konusudur. Bu durumda rota, konveyörün şartlarına bağlı olacaktır. Yine x ve y'nin aldığı değerler değişebilir. Ancak konveyörlerin tek yönlü işlemlerinde bir geri dönüşten söz edilemeyeceği için toplam kat edilmesi gereken mesafe 2x + 2y olarak değil, x + y olarak gerçekleşecektir. Lakin bu mesafenin kısalmasına bağlı olarak zaman kısalsa da ürün sayısına göre yerleştirme zamanları yine de artmaya devam edecektir. Belli tek bir eleman (robot veya insan) kullanan robotlara göre, konveyörler yarı zamanda mesafede işlemi yapacaktır ama toplamda a ürünün bulunduğu nokta ile yerleştireceği nokta arasındaki mesafe işlem zamanının ana etkeni olmaya devam edecektir. Bu mesafe depolama faaliyetlerindeki ana maliyet kalemlerinden de biridir.

## Mal Kabul Problemi Literatür Taraması

Mal kabul operasyonunda ortaya çıkan mesafe problemi ve buradaki kaynakların etkin kullanımı ile ilgili yapılmış çalışmalar bulunmaktadır. Bu çalışmalara göre mal kabul ve yerleştirme problemine belli başlı yaklaşımlar aşağıda listelenmiştir.

Heskett vd (1964) tarafından yapılan araştırmalarda **devir bazlı yerleştirme** yöntemi incelenmiştir. Bu yöntemde bir ürünün depolanması için gerekli olan alan, belirli bir periyot boyunca gerçekleşen talep miktarına orantılanmış ve bir indeks değeri bulunmuştur. Bu indeks değeri küçük olan ürünler giriş ve çıkış toplama alanlarına en yakın yerlere stoklanmıştır. Bu yöntemle devir hızı yüksek ürünler kolay alanlara yerleştirilerek mesafeler düşürülmeye çalışılmıştır (Heskett, Ivie ve Glaskowsky, 1964).

Hausman vd (1976) yaptıkları çalışmada **en yakın alana yerleştirme** yöntemini incelemişlerdir. Bu yöntemde, depoya giriş yapılan ürünler, toplayıcının karşılaştığı ilk boş alana yerleştirilmiş, bu şekilde mal kabul işlemleri esnasında kat edilen mesafe azaltılmıştır. Ancak depoların ön bölümlerinin doluluğu artarken, arka bölümlerindeki boşluklar da artmıştır (Hausman, Schwarz ve Graves, 1976).

Frazelle (1989) yaptığı çalışmada **gruba dayalı yerleştirme** yöntemini ortaya atmış, bu yöntemdeki yerleştirme işlemini ürünlerin birbirleri arasındaki ilişkiye dayandırarak gerçekleştiren bir model üzerinde çalışmıştır. Müşteriden gelen bir ürün siparişi geldiğinde, bu ürün ile başka ürünlerin sipariş etme olasılığı incelenmiş, bu olasılığın yüksek olduğu durumlarda ürünlerin birbirine yakın yerlere yerleştirilmesi önerilmiştir (Frazelle E. H., 1989).

Petersen (1997) yaptığı çalışmalarda, geçmiş araştırmaları da derleyerek, gelen ürünlerin boş depo lokasyonlarına eşit olasılıklara yerleştirme yöntemini incelemiştir. **Rastgele yerleştirme** olarak isimlendirebileceğimiz bu yöntemle depo alanlarının çok etkin kullanılabildiği ortaya konurken, rota mesafelerinin alternatif yöntemlere göre daha uzun olabildiği tespit edilmiştir. Bilgisayarlar tarafından yönetilen ve otomasyon destekli depolar için verimli olabileceği sonucuna ulaşılmıştır (Petersen, An evaluation of order picking routeing policies, 1997).

De Koster vd (2001) **ürüne lokasyon tahsis etme** yöntemini incelemiştir. Bu yönteme göre bütün ürünler için bir veya birden fazla lokasyon önceden ayrılmakta, her gelen ürün, kendisine ayrılan alana yerleştirilmektedir. Bu yöntem mesafe problemi ile ilgilenmemektedir. Yöntem ürünlerin bulunmasını ve envanter yönetimini kolaylaştırsa

da hiç gelmeyebilecek ürünlere bile alan tahsisi yapıldığı için depo alanının verimli kullanım oranını düşürebilmektedir (De Koster ve Neuteboom, 2001).

Petersen vd (2004) yaptıkları çalışmada daha önce yapılan çalışmaları incelemiş ve bu yöntemlerin bir karışımı olan **sınıfa dayalı yerleştirme** yöntemini ortaya atmışlardır. Bu yöntemde ürünler ABC sınıflandırmasına tabi tutulur, bu sınıflandırmaya ayrılan ürünler, sınıfına göre bir alan tahsis edilerek kendi bölgesine yerleştirilmesi önerilmiştir. Bu çalışma ürüne lokasyon tahsis etme ve devir bazlı yerleştirme yöntemlerinin birleştirilmesiyle ortaya çıkmıştır (Petersen, Aase ve Heiser, 2004).

Buraya kadar incelenen çalışmalardan elde edilen ürün yerleştirme yöntemleri **Tablo 2**'de özellikleriyle birlikte listelenmiştir.

**Tablo 2.** Ürün yerleştirmede kullanılan çeşitli yöntemler ve bunların özellikleri

| Yöntem | Temel Kriter | Yerleştirme Mesafesi | Envanter Yönetimi | Hacim | Toplama Mesafesi |
|---|---|---|---|---|---|
| **Devir Bazlı Yerleştirme** | Talep | Etkisiz | Etkisiz | Etkisiz | Olumlu |
| **En Yakın Alana Yerleştirme** | Mesafe | Olumlu | Olumsuz | Etkisiz | Etkisiz |
| **Gruba Dayalı Yerleştirme** | Ürün Grup | Etkisiz | Etkisiz | Etkisiz | Olumlu |
| **Rastgele Yerleştirme** | Yok | Etkisiz | Etkisiz | Olumlu | Etkisiz |
| **Ürüne Lokasyon Tahsis Etme** | Ürün | Etkisiz | Olumlu | Olumsuz | Etkisiz |
| **Sınıfa Dayalı Yerleştirme** | Devir Hızı | Etkisiz | Olumlu | Etkisiz | Olumlu |

Bu yöntemlerden devir bazlı yerleştirme, gruba dayalı yerleştirme ve sınıfa dayalı yerleştirme modelleri bu çalışmada da detaylı şekilde incelenecek ve ilerleyen aşamalarda kullanılacaktır. Bu yöntemler, diğerlerinden farklı olarak ürünlerin sadece yerleştirme aşamasıyla değil, çıkışıyla da ilgilenmektedir. Ürünlerin yerleşim metodu, çalışmanın asıl konusu olan sipariş toplama operasyonunu, direkt olarak etkilediği için, bu metotlardan çalışmanın ilerleyen aşamalarında da bahsedilecektir.

**Sipariş Toplama Operasyonu**

Sipariş toplama operasyonu, depoda ürünlere yönelik müşteri taleplerine cevap vermek için, ürünlerin raf bloğu içerisinde yerleştirilmiş olduğu yerlerden alınarak, çalışmada daha önceden aktarılan çıkış toplama alanına götürülmesi işlemidir.

Bu operasyon iki temel aşamadan oluşmaktadır. Birinci aşama, müşteri tarafından oluşturulan siparişlerin belirli şekilde gruplanıp toplama listelerinin hazırlanmasıdır. İkinci aşama ise, oluşturulan toplama listelerine göre toplayıcı birimlerin belirtilen ürünleri belirtilen adresten alarak, fiziki olarak çıkış toplama alanına getirmesidir.

Operasyonun ilk aşaması olan siparişlerin belirli şekillerde gruplanıp toplama listesi oluşturma aşaması, incelenen mesafe problemi açısından önemlidir. Toplama listesi hazırlanırken belirli bir sıraya göre hazırlanıp, toplayıcıların daha az mesafe kat etmesi sağlanabilir. Ancak bu listenin hazırlanması için ikinci aşamanın icra biçimi de belirleyici rol oynayacaktır.

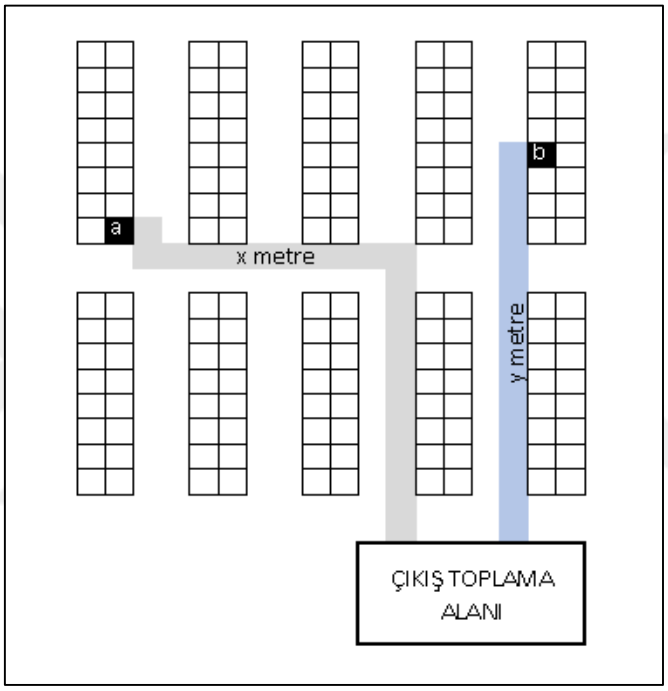
İkinci aşama olan toplayıcı birimlerin ürünleri adresten alarak toplama alanına getirmesi daha önce de belirtilmiş olduğu gibi üç farklı biçimde yapılabilir (Dallari, Marchet ve Melacini, 2008). Bu biçimler aşağıda listelenmiştir:

1. **Toplayıcıdan parçaya:** Ürün toplama işini yapacak insan, yaya olarak veya forklift gibi bir araçla ürünün adreslenmiş olduğu rafa giderek ürünü alır ve toplama alanına getirir.
2. **Parçadan toplayıcıya:** Konveyör veya başka bir yarı otomatik sistem aracılığıyla ürün bulunduğu raftan alınarak toplayıcının bulunduğu noktaya ulaştırılır.
3. **Tamamen otomasyona dayalı:** Ürün bulunduğu adresten bilgisayar kontrollü bir donanım sistemi aracılığıyla alınıp direkt olarak toplama alanına ulaştırılır.

Çalışma kapsamında, direkt olarak incelenen sistem toplayıcıdan parçaya sistemi olmasına rağmen, mesafeye dayalı problem parçadan toplayıcıya gidilen veya tamamen otomasyona dayalı sistemde de ortaya çıkmaktadır. Ancak araya bilgisayar kontrollü ek sistemler girdiği için mesafeyi etkileyecek değişkenlerin sayısı da bu son iki sistemde artmaktadır.

Sipariş toplama aşamasında harcanan zaman dolayısıyla maliyetlerin de ana kaynağını çalışmamızda konu alınan mesafe problemi oluşturmaktadır. Bu mesafeler, mal kabul operasyonundaki benzer problemlerdir. Çünkü temelde mal kabul operasyonunda yapılan

işlemlerin tersi, ürün bazında, sipariş toplama operasyonunda da yapılmaktadır. Örneğin a ürünün bulunduğu rafın, toplama alanına olan mesafesini x kabul edelim. Toplayıcının ürüne gittiği sistemde toplayıcı toplama alanından a ürünün bulunduğu rafa x mesafeyi kat edecek, ürünü alıp tekrar x mesafeyi kat ederek ürünü toplama alanına getirecektir. Ardından b ürünün bulunduğu rafa y mesafeyi kat edip ürünü alarak, ürünü tekrar y kadar mesafe kat ederek toplama alanına getirecektir. Bu durumda toplayıcının parçaya gittiği sistemde kat edilen mesafe 2x + 2y olacaktır. Bununla ilgili örnek **Şekil 9**'da gösterilmiştir.




**Şekil 9.** Sipariş toplama işlemi esnasında kat edilen mesafelere bir örnek

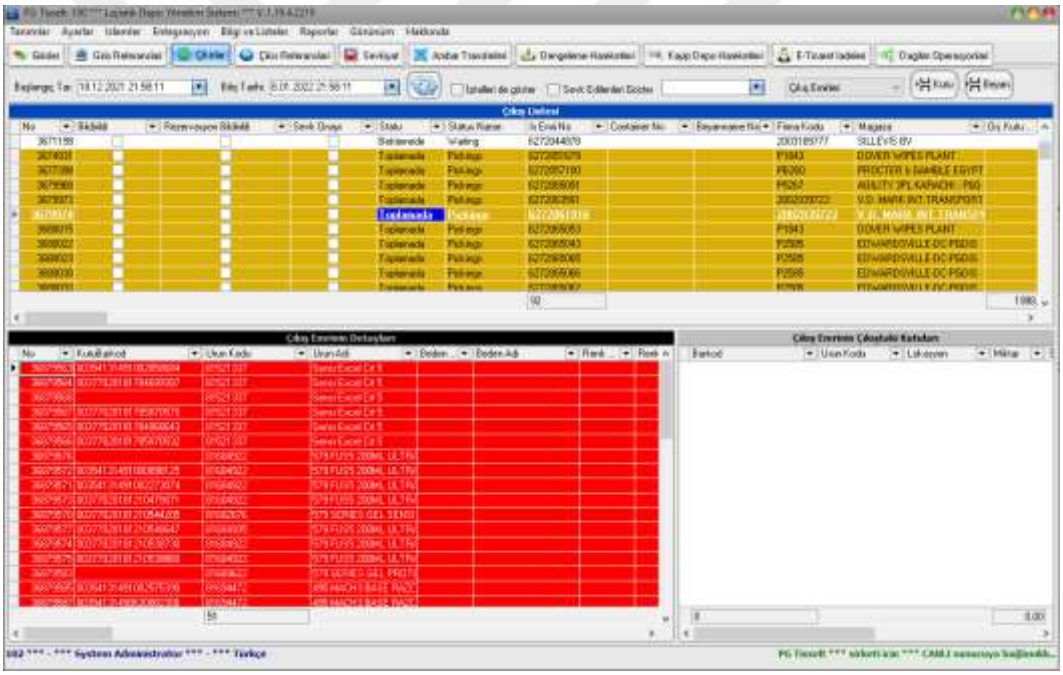
Parçanın toplayıcıya gittiği veya tamamen otomasyona dayalı sistemlerde ise yine ürünlerin x ve y mesafesini kat etmesi gerekecektir. Bu sistemlerde yer alan donanım sistemlerinin yapısından dolayı en kısa mesafeler olan x ve y'nin değerleri değişse de x ve y'nin ürün tarafından kat edilmesi gereken mesafeler olduğu gerçeği değişmeyecektir. Ancak bu sistemlerde toplayıcının iki yönlü (gidiş geliş) şeklinde yaptığı yolculuk yerine ürün tek yönlü olarak (sadece gidiş) yolculuk yapacak ve kat edilen mesafe formülü bu durumda 2x + 2y olarak değil, x + y olarak gerçekleşecektir.

### Toplama Listelerinin Hazırlanması

Daha önceden de belirtildiği gibi, depodan ürün çıkışı müşteri ihtiyaçları ile başlar. Müşteri kendi ihtiyaçlarına göre depodan ürünlerin çıkması için bir talep oluşturur. Çalışmada bu taleplerden sipariş diye bahsedilmiştir.

Bir deponun çalışma biçimine göre müşteri talepleri depoya farklı biçimlerde gelebilir. Bir günde bir müşteriden sadece bir ürün çeşidinden yüzlerce ürün talep edilebileceği gibi, birçok farklı müşteriden bir saat içinde birçok farklı ürün çeşidi talep edilebilir. Sipariş toplama optimizasyonun sağlanması için, belirli zaman aralıklarında bu siparişlerdeki ürünler bir araya getirilip toplama alanlarına ürünlerin biriktirilmesi gerekmektedir.

Örnek olarak sektördeki firmalardan Barsan Global Lojistik'in Tisselt, Belçika'daki bir deposuna gelen siparişlerin ekran görüntüsü **Şekil 10**'da verilmiştir.



**Şekil 10.** Bir depoya gelen müşteri sipariş örnekleri

Toplama listelerinin hazırlanması süreci eldeki müşteri taleplerinin bir araya getirilmesi ile başlar. Bir araya getirilen bu farklı taleplerin depo şartları ve rota optimizasyonu göz önüne alınarak sıralanması gerekmektedir.

Bu toplama listeleri manuel hazırlanabileceği gibi, günümüzde genelde bilgisayar yazılımları, çeşitli algoritmalarla bu listeleri hazırlamaktadır. İncelenen Barsan Global Lojistik Tisselt tesislerinde firmanın kendi depo yönetim sistemi (WMS) yazılımı olan

LDS isimli bilgisayar programı, kendine has bir algoritma ile bu toplama listelerine bir örnek **Şekil 11**'de gösterilmiştir.



**İŞ EMRİ TOPLAMA LİSTESİ**

23.12.2021 / 22:04:41

| İş Emri | Ürün Kodu | Ürün Adı | Lokasyon | İstenen Miktar | Kutu İçi Miktar | Toplama Detayı | Koli Sayısı | Palet Sayısı/No |
|---|---|---|---|---|---|---|---|---|
| BRS369 1506 | 0035413149 1 | | TAMAM | 24 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 6 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 7 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 6 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 4 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 7 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 1 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 1 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 80 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 12 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 3 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 8 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 2 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 1 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 4 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 1 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 8 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 5 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 1 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 1 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 1 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 1 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 2 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 6 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 5 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 4 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 1 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 2 | | | | |
| BRS369 1506 | 0035413149 1 | | TAMAM | 1 | | | | |

Yükleyici Operatör: Tarih: Müşteri:

SAYFA SAYILARI: 3 / 1

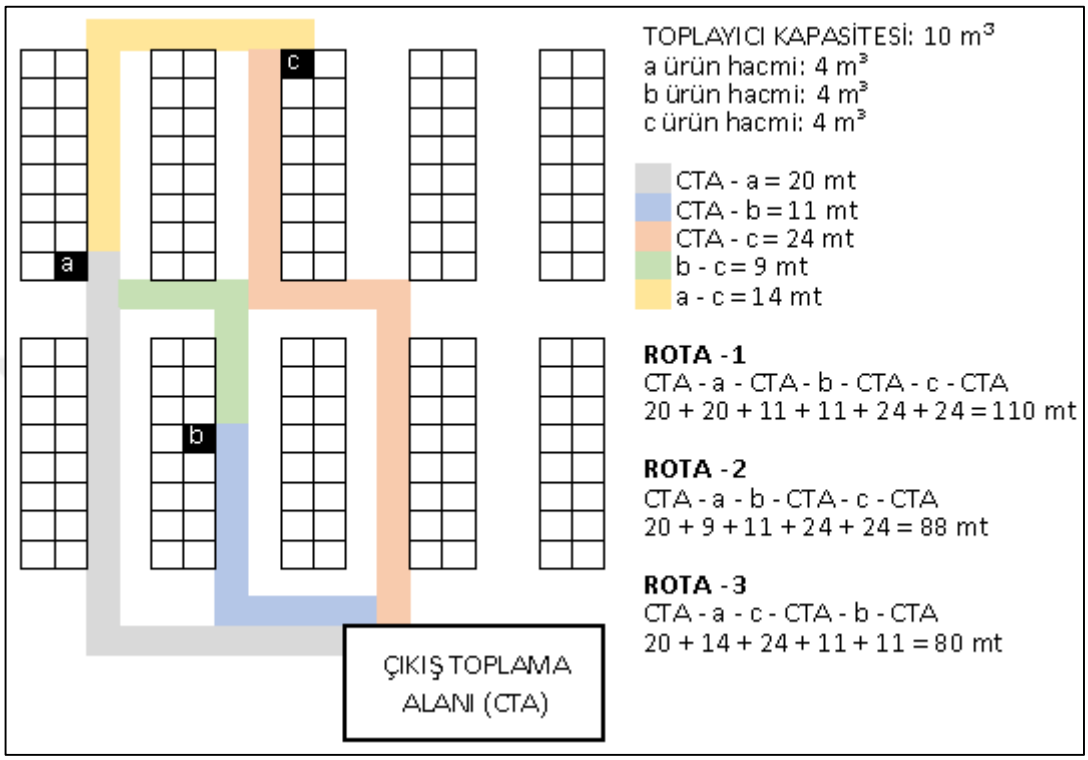
**Şekil 11.** LDS sisteminde hazırlanmış örnek bir toplama listesi

Bu toplama listeleri hazırlandıktan sonra, toplayıcı elektronik veya yazılı olarak kendisine ulaşan bu liste aracılığıyla toplama işlemlerini gerçekleştirir. Toplayıcının kendisine verilen bu adreslere ulaşımında kullandığı yollar, genelde ilgili lokasyonun adresine ulaşabileceği en kısa yol olacaktır. Bu aşamada sipariş toplama rotalarını, toplama listesi oluştururken belirlemek ve toplayıcıları baştan belirlenen bu bütün toplama listesine göre yönlendirmek gerekliliği ortadadır. Operasyon esnasında rotaların belirlenmesi, bütünlüklü ve efektif olmayacaktır.

### Sipariş Toplama Operasyonu Üzerinde Toplama Listelerinin Etkisi

Sipariş toplama operasyonun ilk aşaması olan toplama listelerinin hazırlanması işlemi de toplayıcının ürüne gittiği ilk sistem için daha önemlidir. Örnekte toplayıcılar ürünleri tek tek gidip toplamaktadır. Ancak bu örneklerde bir toplayıcının bir seferde bir adet ürün alabildiği ön kabulü yapılmıştır. Toplayıcının kullandığı ekipmanlara ve siparişlerdeki ürünlerin biçimine göre, bir toplayıcı bir turunda birden fazla ürünü alıp toplama

noktasına getirebilir. Bu durumda mesafeler için uygulanacak formülün değişmesi gibi, ürünlerin toplama sırası da bu formülün oluşmasında büyük öneme sahip olacaktır. Toplayıcının kapasitesinin 10 m³, ürünlerin hacminin 4 m³ olduğu bir örnek ve çeşitli rota senaryoları **Şekil 12**'de verilmiştir.




**Şekil 12.** Örnek bir toplama senaryosu

Verilen örnekte 3 farklı rota senaryosu oluşturulmuştur. Eğer daha önce verilen örneklerdeki gibi her ürün tek tek alınıp çıkış toplama alanına yerleştirilseydi, verilen örnekte toplayıcının 110 metre (Rota 1) yol kat etmesi gerekecekti.

Şekil 10'daki örnekte ise, toplayıcı birim 10 metreküp ürün taşıyabildiği ve ürünler de 4 metreküp hacme sahip olduğu için toplayıcı tüm ürünleri tek tek taşımak zorunda değil. Aynı anda iki ürünü birden alabilir. Bu durumda ilk ürünü aldıktan sonra ikinci ürüne giderek ikisini birden toplama alanına getirebilir, ardından üçüncü ürünü alıp toplama alanına getirebilir.

Bu bilgiye sahip olarak toplayıcıya a, b ve c sırasıyla bir toplama listesi verildiğinde toplayıcı önce a ürününe gidip 20 metrelik bir mesafe kat edecekti. Ardından a ürününden b ürününe dönerek 9 metrelik bir mesafeyi daha kat edecek ve iki ürünü birden alıp 9 metre daha mesafe kat ederek çıkış toplama alanına getirecekti. Son olarak c ürününü almaya 24 metre gidip 24 metre mesafeyi de geri dönerek üç ürünün toplama işlemini

tamamlayacaktı. Örneğimizde Rota 2 olarak isimlendirilen bu senaryoda toplayıcının kat etmesi gereken toplam mesafe 88 metre olacaktı. Ürünlerin tek tek toplandığı Rota 1 senaryosuna göre 22 metre daha az yol kat edilecek, bu 22 metrelik mesafeden zaman, emek ve forklift gibi bir araç kullanılıyorsa yakıt tasarrufu sağlanacaktı.

Bunun yanında, ürünlerin fiziki toplama operasyonlarının başlamasından önce oluşturulan toplama listesinin daha etkin oluşturulması durumunda toplama sıralarına müdahale edilebilir ve daha farklı kazançlar elde edilebilir. Verilen örnekte, toplayıcıya ürünleri a, b, c sırasıyla değil de a, c, b sırasıyla toplaması önerilen bir Rota 3 senaryosu da bulunmaktadır. Bu senaryoda toplayıcı, toplama alanından 20 metre mesafe kat edip a ürününe ulaşacak, ardından 14 metre daha ilerleyip c ürününe ulaşacak, sonra da iki ürünü de alıp 24 metre mesafe kat edip toplama alanına bırakacak. Son olarak 11 metre gidip b ürününü alıp 11 metre geri dönecek ve toplama alanına ulaşacak. Bu senaryoda toplayıcının kat ettiği mesafe toplamda 80 metre olacak.

Görüldüğü gibi, ürünlerin boyutları, durduğu yerler, toplayıcının niteliği tamamen aynı olmasına rağmen, toplama işleminden önce yapılan bir küçük müdahale ile c ürünün b'den önce toplanması sağlanarak bile 8 metrelik bir mesafe azaltılabiliyor. Sadece üç ürünün toplandığı bu senaryoda bile yaklaşık yüzde onluk bir mesafe tasarrufu ortaya çıkarılabiliyor.

Bu durumda sipariş toplama probleminin sadece ürüne ulaşım yolunun optimize edilmesiyle ele alınmasının yetersiz olduğu görülebilir. Aynı ürünlerin, aynı adreslerin ve aynı rotaların yer aldığı bir senaryoda, bir ürünün toplama sırasının yerinin değiştirilmesi ile bile kayda değer bir tasarruf sağlandığı göz önüne alındığında, problemin operasyonun birinci aşamasında toplama listesi hazırlanırken ele alınması gerekliliği açıktır. Çünkü oluşturulan toplama listesindeki sıralamaya göre rotalar belirlenecektir.

# BÖLÜM 3. YÖNTEM

Bu bölümde araştırma modeli, çalışılan depo ve raf tipleri, üzerinde çalışılan ürün tipleri ayrıntılı olarak açıklanacaktır.

Daha önceki yapılan çalışmalardan farklı olarak, sipariş yerleştirme ve toplama süreçleri bir bütün olarak değerlendirilecek, sonuçlar da buna göre incelenecektir. Yine ortaya çıkan yeni algoritma biçimleri uygulamada sınanacak, buna göre yeni sonuçlara ulaşılmaya çalışılacaktır.

Seçilen araştırma modelinin özellikleri açıklanacak, araştırma modelinin oluşturduğu evren içinden seçilen örneklemin özellikleri ile karşılaştırılacaktır. Ardından örneklem üzerinden ham veriler toplanıp, detaylı şekilde çözümlenecektir. Devamında yorumlanıp, uygulamada kullanılmaya hazır hale getirilecek ve nihayetinde uygulama tamamlanacaktır. Tamamlanan uygulama, başka bir uygulamanın çağrılmasına hazır hale getirilip test sürecine sokulacak ve saha testleri yapılacaktır.

## Araştırma Modeli

Bu araştırmada veri setlerinin toplanması, çeşitlendirilmesi ve yorumlanması için Barsan Global Lojistik işletmesinin lojistik depoları kullanılmıştır. **Şekil 13**'de görüldüğü gibi Barsan Global Lojistik, dünyanın 48 ülkesinde organizasyonu tamamlamış ve 3 kıtada 43 lojistik depo tesisi ile depolama alanında faaliyet göstermektedir. 1982 yılından beri sektörde olan şirket, 1990 yılından beri de depolama faaliyetlerinde bulunmaktadır (BGL, 2022).




**Şekil 13**. Barsan Global Lojistik'in depolama tesisleri

Çalışmadaki verilere kaynak olarak firmanın kullandığı LDS (Lojistik Depo Sistemi) yazılımının veritabanı kullanılmıştır. Veritabanı çalışma kapsamında bir kopyası alınarak incelenmiştir. Bu çalışmada Barsan Global'in tercih edilmesinin ana nedenleri arasında aşağıdaki maddeler sıralanabilir.

### Depo Sayısının Fazlalığı ve Yaygınlığı

Barsan Global Lojistik (BGL) hali hazırda 43 depolama tesisine sahiptir. Bu tesisler bölge, metrekare ve coğrafi koordinat bilgileri ile **Tablo 3**'te listelenmiştir.

**Tablo 3.** Çalışmada incelenen BGL'nin depo tesisleri

| Tesis Adi | Bölge | Metrekare | Enlem | Boylam |
|---|---|---|---|---|
| **Orhanli Logistics & Bonded Warehouse** | TR | 37.500 | 40,9104210 | 29,3390780 |
| **Samandıra Logistics Center** | TR | 41.857 | 40,9637370 | 29,2144940 |
| **Hadımköy Logistics Center** | TR | 9.000 | 41,0543022 | 28,6401005 |
| **Pelitli Logistics & Bonded Warehouse** | TR | 65.000 | 40,8724470 | 29,4627090 |
| **Los Angeles DC** | USA | 4.800 | 33,7902760 | -118,2201000 |
| **Manisa Logistics & Bonded Warehouse** | TR | 10.250 | 38,6149270 | 27,3663520 |
| **İzmir Logistics & Bonded Warehouse** | TR | 5.000 | 38,2739420 | 27,2226940 |
| **Hadımköy Logistics & Bonded Warehouse** | TR | 25.000 | 41,1519160 | 28,6044180 |
| **Halkalı Logistics & Bonded Warehouse** | TR | 24.000 | 41,0139360 | 28,7643550 |
| **Çaka Bonded Warehouse** | TR | 8.000 | 40,7533690 | 30,0232160 |
| **MAGNA Suadiye Plant** | TR | 1.500 | 40,7004757 | 30,0491070 |
| **MAGNA Gölcük Plant** | TR | 3.100 | 40,7241592 | 29,8456099 |

| | | | | |
|---|---|---|---|---|
| **Çerkezköy Logistics & Bonded Warehouse** | TR | 16.097 | 41,2828080 | 27,9689830 |
| **Bursa Logistics & Bonded Warehouse** | TR | 7.500 | 40,2528280 | 28,9317960 |
| **Paris Logistics & Bonded Warehouse** | FR | 5.000 | 49,0271200 | 2,4492510 |
| **New Jersey DC** | USA | 4.500 | 40,9244850 | -74,1173290 |
| **Romania DC** | ROM | 6.000 | 45,7680210 | 21,1702150 |
| **Ireland DC** | IR | 3.500 | 53,2545660 | -6,1760740 |
| **Belarus DC** | BEL | 1.250 | 53,9932787 | 27,5242475 |
| **Erbil DC** | Iraq, Erbil | 13.500 | 36,2010660 | 44,0551760 |
| **Houston DC** | Houston | 10.000 | 29,6472210 | -95,0348550 |
| **Chicago DC** | Chicago | 7.500 | 41,7788905 | -87,9891795 |
| **Almata DC** | Almata | 7.846 | 43,4501720 | 77,0309820 |
| **Poland Pabianice DC** | Polonya | 8.200 | 51,7000000 | 19,4200000 |
| **Frankfurt Logistics Center** | Frankfurt | 1.860 | 50,0580210 | 8,5384370 |
| **Nuremberg Logistics Center** | Nuremberg | 5.888 | 49,5199290 | 11,0018020 |
| **Bağdat DC** | Iraq,Baghdad | 800 | 33,2555960 | 44,4341460 |
| **Gaziantep MAIN** | Gaziantep | 1.100 | 37,0612170 | 37,3751993 |
| **Erzurum MAIN** | Erzurum | 1.700 | 39,9140010 | 41,1923193 |
| **Adana MAIN** | Adana | 1.200 | 36,9742680 | 35,5917287 |
| **Diyarbakır MAIN** | Diyarbakır | 550 | 38,0749090 | 40,0557493 |
| **P&G Gebze Plant** | Kocaeli | 1.500 | 40,8541420 | 29,4224028 |
| **P&G Manchester Plant** | Manchester | 1.000 | 53,4776295 | -2,3224604 |
| **Munich Logistics Center** | Munich | 1.000 | 48,1957319 | 11,6030327 |
| **Rotterdam Logistics Center** | Rotterdam | 5.000 | 51,9170551 | 4,1837185 |
| **Budapest Logistics Center** | MAC | 3.500 | 47,4979124 | 19,0402350 |
| **Berlin Logistics Center** | Berlin | 2.400 | 52,5075071 | 13,2417016 |
| **Hannover Logistics Center** | Hannover | 3.500 | 52,3796457 | 9,6914319 |
| **Shanghai Logistics Center** | China, Shanghai | 2.400 | 31,1312742 | 120,9703940 |
| **Modena Logistics Center** | Modena | NULL | 44,6377325 | 10,8516846 |
| **Reimsheld Logistics Center** | Reimsheld | 846 | 51,2039690 | 7,2465620 |
| **Samandıra Bonded Warehouse** | TR | 3.400 | 40,9637370 | 29,2144940 |
| **Willebroek DC** | BEL | 20.000 | 51,0340863 | 4,3569169 |

Dünyanın çeşitli yerlerine yayılmış ve oldukça çeşitlilik gösteren bu tesislerin tamamı bu çalışma kapsamında incelenmiştir.

**Depo Sirkülasyon Hızının Yüksekliği**

BGL'ye ait 43 tesise 2020 yılında yapılan toplam ürün giriş adetleri **Tablo 4**'te listelenmiştir.

**Tablo 4.** BGL'ye ait depolara 2020 yılında aylık giriş sayıları

| Ay | Yıl | Giriş |
|---|---|---|
| **1** | 2020 | 600.089.042 |
| **2** | 2020 | 537.912.483 |
| **3** | 2020 | 496.837.326 |
| **4** | 2020 | 549.355.689 |
| **5** | 2020 | 493.434.510 |
| **6** | 2020 | 560.477.683 |
| **7** | 2020 | 533.418.627 |
| **8** | 2020 | 592.391.070 |
| **9** | 2020 | 726.106.782 |
| **10** | 2020 | 744.752.181 |
| **11** | 2020 | 657.564.098 |
| **12** | 2020 | 858.374.213 |

Sayılarda görüldüğü gibi BGL bünyesindeki depolara 2020 yılında her ay en az 490 milyon adet ürün girişi gerçekleşmiştir.

**Depolama Hizmeti Verilen Müşteri Sayısı ve Sektör Çeşitliliği**

BGL'nin 43 tesisinde 256 farklı müşteriye hizmet verdiği görülmüştür. İncelemede hizmet verilen bu 256 müşterinin 20 farklı sektörde toplandığı **Tablo 5**'de gösterilmiştir.

**Tablo 5.** BGL'nin hizmet verdiği müşteriler ve bu müşterilerin sektörleri

| Sektör Adi | Müşteri Sayısı |
|---|---|
| **Automotive After Market** | 16 |
| **Bonded Operations** | 6 |
| **Computer Hardware** | 1 |
| **Container&Truck Transfer Operations** | 45 |
| **Electrical Hardware Materials** | 4 |
| **Finished Goods Distribution** | 18 |
| **Food Raw Material** | 7 |
| **General Warehouse Operastions** | 2 |
| **Glassware Products** | 2 |
| **High Technology Products** | 50 |
| **Home Consumer Durables** | 2 |
| **Installation Materials** | 8 |
| **Line Feeding** | 22 |
| **Line Feeding -In Factory** | 13 |
| **Line Feeding -In Warehouse** | 26 |
| **Medical and Pharmaceutical** | 7 |
| **Mobile Technology Products** | 2 |
| **Robotic Technology** | 1 |
| **Textile Retail** | 23 |
| **Toys Products** | 1 |

Bu müşteri çeşitliliği ve sektör zenginliği yapılan çalışmanın genelleşmesi ve datasetlerin toplanması açısından önemlidir. Çünkü farklı sektörlerde ve farklı müşterilerde farklı depo koşuları oluşabilmekte bu da ürünlerin ve kutuların sirkülasyon hızlarının sektör ve müşteriye göre farklılaşmasına neden olabilmektedir.

**Evren ve Örneklem**

Araştırma esnasında, veri araştırmalarında ve uygulama testlerinde, BGL'nin 43 tesisinden Belçika'daki Willebroek tesisi incelenmiştir. Tesislerin bir görüntüsü **Şekil 14**'de gösterilmiştir.

+




**Şekil 14.** BGL Willebroek tesisleri

Willebroek tesisleri, Brüksel'in 20 km kuzeyindeki Willebroek kasabasına bağlı Tisselt köyünde bulunan 20 bin metrekarelik lojistik tesisleridir. Tesis, BGL'nin müşterisi olan uluslararası bir tüketim malzemesi firmasına ait ürünlerin giriş çıkış yapıldığı bir depolama merkezidir.

Örneklem olarak bu deponun seçilmesinin nedenleri arasında ilki direkt olarak tüketiciye yönelik ürünler bulunması nedeniyle sirkülasyon hızının yüksekliği olmuştur. Sirkülasyon hızının yüksekliği, sipariş raflama ve toplama süreçlerinde ortaya çıkan mesafe ve zaman problemlerini doğrudan etkilemekte, sirkülasyon yükseldikçe bu problemlerin etkisi artmaktadır. 2020 yılı için Willebroek tesislerinin tüm BGL içindeki sirkülasyona oranı **Tablo 6**'da gösterilmiştir.

**Tablo 6.** Willebroek tesislerinin sirkülasyon hızının BGL içindeki oranı

| Ay | Yıl | Giriş | Willlebroek Giriş | Yüzde |
|---|---|---|---|---|
| **1** | 2020 | 600.089.042 | 33.459.810 | 5,58% |
| **2** | 2020 | 537.912.483 | 29.667.287 | 5,52% |
| **3** | 2020 | 496.837.326 | 36.072.958 | 7,26% |
| **4** | 2020 | 549.355.689 | 31.195.903 | 5,68% |
| **5** | 2020 | 493.434.510 | 24.568.360 | 4,98% |
| **6** | 2020 | 560.477.683 | 22.319.249 | 3,98% |
| **7** | 2020 | 533.418.627 | 35.750.285 | 6,70% |
| **8** | 2020 | 592.391.070 | 34.705.576 | 5,86% |
| **9** | 2020 | 726.106.782 | 32.300.937 | 4,45% |
| **10** | 2020 | 744.752.181 | 30.790.146 | 4,13% |
| **11** | 2020 | 657.564.098 | 30.257.663 | 4,60% |
| **12** | 2020 | 858.374.213 | 31.324.801 | 3,65% |

Depolarda ürünlerin miktarı iş yükü ve fiziki şartlar için önemli olsa da asıl sirkülasyon miktarını belirleyen ise kutu miktarlarıdır. Çünkü ürünler depolar içinde kendi başlarına değil, bir kutu içinde, kendi türünden belirli bir sayıda ürünle hareket eder. Bu yüzden **Tablo 7**'de tüm BGL ölçeğinde ve sadece Willebroek tesisleri ölçeğinde 2020 yılında giriş yapılan kutu sayıları da gösterilmiştir.

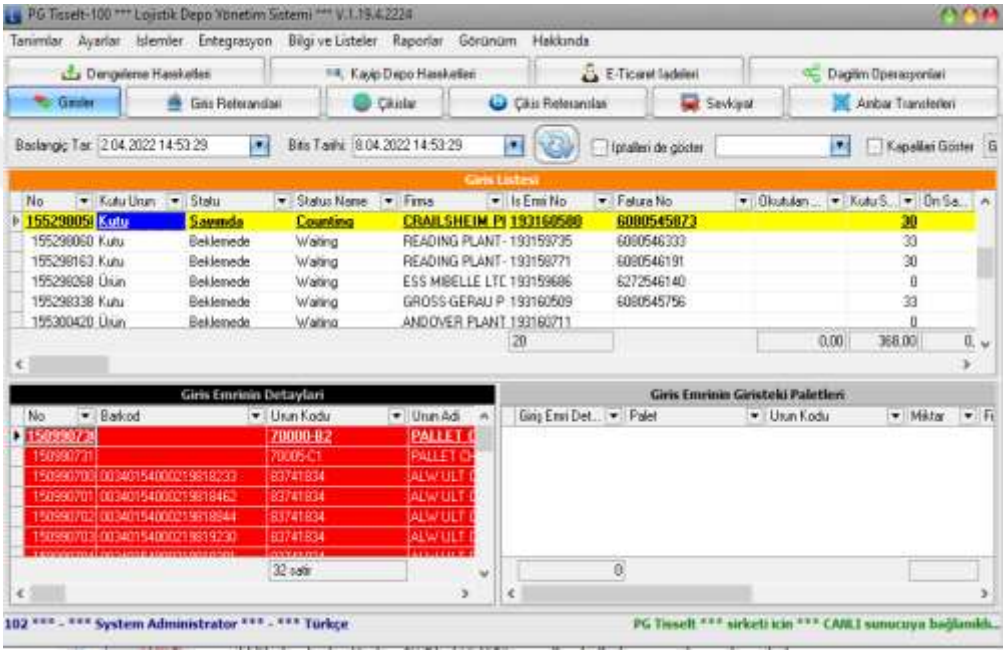
**Tablo 7.** Tüm BGL ve Willebroek ölçeğinde 2020 yılı için giriş yapılan kutu sayıları

| Ay | Yıl | Giren Kutu | Willlebroek Giren Kutu | Yüzde |
|---|---|---|---|---|
| **1** | 2020 | 440.529 | 42.687 | 9,69% |
| **2** | 2020 | 432.663 | 41.538 | 9,60% |
| **3** | 2020 | 452.008 | 47.551 | 10,52% |
| **4** | 2020 | 297.043 | 45.782 | 15,41% |
| **5** | 2020 | 319.198 | 39.773 | 12,46% |
| **6** | 2020 | 396.747 | 38.627 | 9,74% |
| **7** | 2020 | 499.380 | 42.321 | 8,47% |
| **8** | 2020 | 525.483 | 43.781 | 8,33% |
| **9** | 2020 | 602.658 | 46.501 | 7,72% |
| **10** | 2020 | 655.745 | 46.159 | 7,04% |
| **11** | 2020 | 582.504 | 45.634 | 7,83% |
| **12** | 2020 | 671.967 | 39.470 | 5,87% |

Tablo 7'de görüldüğü gibi 2020 yılında Willebroek'te yapılan kutu giriş işlemlerinin, tüm BGL tesislerinin içindeki oranı ortalaması %9,39'dur.

**Veriler ve Toplanması**

Araştırmada kullanılan veriler BGL bünyesindeki tüm depolarda kullanılan LDS (Lojistik Depo Sistemi) programından alınmıştır. **Şekil 15**'te programın masaüstü versiyonun bir ekran görüntüsü bulunmaktadır.




**Şekil 15.** LDS programının ana ekranından bir görüntü

Program birkaç modülden oluşmaktadır. Veri toplamak için masaüstü uygulamasından da faydalanan programın, ana saha verileri ise taşınabilir el terminallerinde çalışan ET versiyonundan akmaktadır. LDS'nin çalıştığı bir el terminalinin görüntüsü **Şekil 16**'da yer almaktadır.

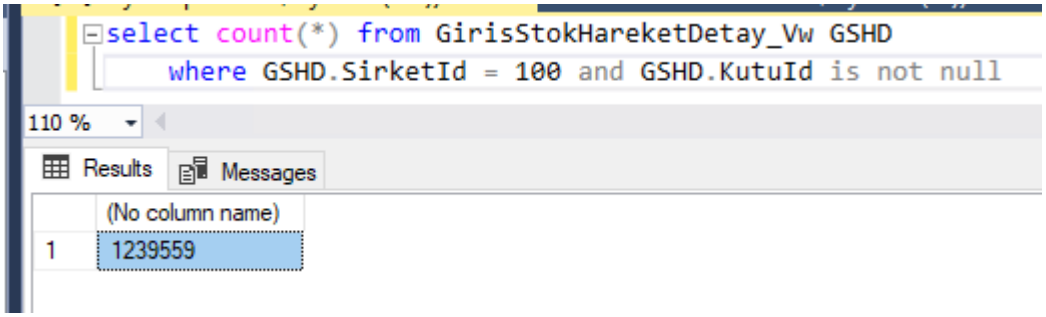



**Şekil 16.** LDS'nin çalıştığı mobil cihazlar ve LDS'nin el terminali versiyonunun bir görüntüsü

El terminali uygulamasındaki genel prensip, ürün veya kutunun depo içindeki her hareketinin uygulama tarafından kayıt alınmasına dayanmaktadır. Uygulamada RFID veya barkod teknolojisiyle girişi yapılan hareketler, merkezi bir sunucuda bulunan Microsoft SQL Server tabanlı veri tabanında kayıt altına alınır. Oluşturulan merkezi veri tabanları sayesinde her türlü uygulama, donanım, sistem araçları üzerinden elde edilen veriler kaydedilir, işlenir talep edildiğinde hızlı ve güvenli bir biçimde iletilir. Anlık tutulan kayıtlar zamanla artar ve büyük kapasiteli veri tabanları haline dönüşür (Şahinaslan ve Şahinaslan, 2022).

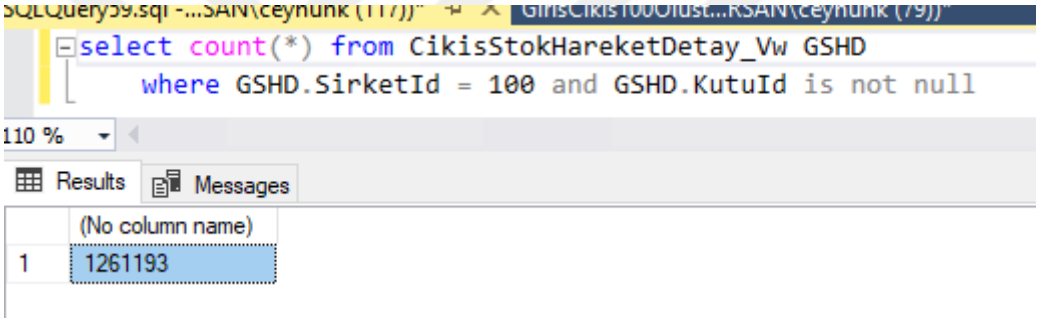
Araştırmada kullanılmak üzere, Willebroek kayıtlarının tutulduğu veri tabanının 7 Aralık 2018'den 3 Mart 2022'ye kadar olan bütün kayıtları, başka bir veri tabanına yedeklenmiş ve buradaki verilerden faydalanılmıştır.

Örnek alınan bu veri tabanındaki giriş okutmalarının kutu düzeyinde toplam sayısı 1,239,559 adettir. Aşağıda direkt veritabanından çekilen SQL sorgusu ve dönen sonuç **Şekil 17**'de gösterilmiştir.




**Şekil 17.** Örnek veri tabanındaki kutu giriş okutmalarının sayısı

Yine örnek veri tabanımızda çıkış kutu okutmalarının toplam sayısının 1.261.193 olduğu **Şekil 18**'de görülebilir.




**Şekil 18.** Örnek veri tabanındaki kutu çıkış okutmalarının sayısı

Çıkış yapılan kutu sayısının giriş yapılan kutu sayısından yüksek olması, anormal bir veri gibi görünse de girişte tek parça halinde gelen kutuların, bazen içeride birden fazlaya bölünme gibi operasyonlardan geçirildiği göz önüne alındığında anlaşılabilir olmaktadır. Bu örnek veriler üzerinden depoya yaklaşık 3 yıl içinde giren yaklaşık 1,2 milyon kutunun hangi lokasyonlarda, ne kadar süre beklediği gözlemlenmeye çalışılmıştır. Bu kutuların hangi özelliklerinin kutuların bekleme süreleri üzerinde etkili olduğu incelenmiştir.

Kutuların depolardaki bekleme süresinin çalışma açısından önemi, kullanılacak algoritmanın kutuları giriş ve çıkış toplama alanlarına hangi mesafedeki lokasyonlara yerleştireceğine karar vermesi açısından önemlidir.

Buradaki temel prensip, depoda daha kısa süre duracak kutuların düzenli olarak daha kısa mesafe kat edilerek ulaşılabilecek lokasyonlara yerleştirilmesi, uzun süre saklanacak kutuların ise daha uzun mesafedeki lokasyonlara yerleştirilmesini sağlamaktır.

Ürünün ne kadar süre depoda kaldığı, yani ürünün sirkülasyon hızının toplam mesafe ve zaman problemi üzerindeki etkisini bir örnekle incelemek gerekirse, **Şekil 19**'daki örneğe bakılabilir.




**Şekil 19.** Sirkülasyon hızının mesafe üzerindeki etkisine 1. örnek

Örnekte depoya 1 hafta içinde 11 giriş ve 11 çıkış yapılabildiği kabul edilmiştir. Bu 11 giriş çıkışın 3 tanesi a ürününden, 1 tanesi b ürününden ve 7 tanesi c ürününden yapılmaktadır. Tüm ürünlerden aynı miktarda da çıkış yapılmaktadır. Dolayısıyla depomuza 1 hafta toplam 22 giriş çıkış işlemi yapılmıştır.

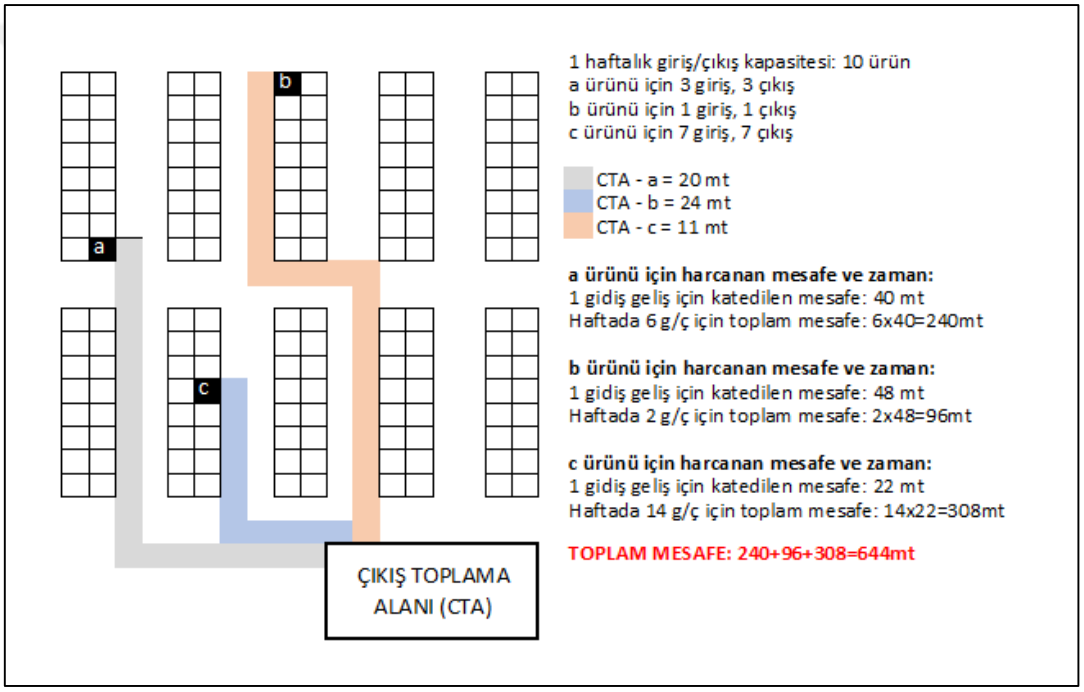
İlk ürün olan a'yı yerleştirdiğimiz lokasyon giriş-çıkış toplama alanımıza 20 metre mesafededir. Her yerleştirmede toplayıcımız ürünü alıp lokasyona yerleştirmekte, sonra bir sonraki ürün için toplama alanına geri dönmektedir. Dolayısıyla her giriş işleminde 40, her çıkış işleminde de 40 metre mesafe kat edilmektedir. 3 giriş işlemi için 120, 3 çıkış işlemi için de 120 metre yol kat edildiğinden, a ürünü için haftada 240 metre yol kat edildiği ortaya çıkmaktadır.

İkinci ürün olan b'yi yerleştirdiğimiz lokasyon giriş-çıkış toplama alanımıza 11 metre mesafededir. Bu üründen haftada 1 giriş ve 1 çıkış işlemi yapılmaktadır. Her giriş işlemi için 22 metre, her çıkış işlemi için de 22 metre yol gidildiğinden haftada alınan toplam mesafe 44 metredir.

Üçüncü ürün olan c'yi yerleştirdiğimiz lokasyon ise giriş-çıkış toplama alanımıza 24 metre uzaklıktadır. 1 kez yerleştirme için 48 metre harcadığımıza göre ve c ürünü için hafta 7 giriş yaptığımıza göre bu ürünün girişleri için haftada 336 metre yol kat edilmektedir. Yine haftada 7 çıkış yapıldığına göre toplam giriş çıkış için alınan mesafe c ürünü için 672 metredir.

Bu üç ürün için 1. örneğimizde toplam kat edilen mesafe 956 metre olmuştur. Bu mesafeye göre zaman da harcanmıştır. Yine toplayıcı ünitede forklift gibi yakıt kullanan bir araç bulunması durumunda bu mesafeye göre enerji sarfiyatı ortaya çıkmıştır.

Aynı depoda, aynı ürünlerin, aynı sirkülasyon sayıları ile farklı bir yerleşime tabi tutulduğu 2. örnek **Şekil 20**'de incelenebilir.




**Şekil 20.** Sirkülasyon hızının mesafe üzerindeki etkisine 2. Örnek

Örnekteki senaryo birebir aynıdır. Tek fark, 1. örnekteki b ve c ürünlerinin yerleşim yerlerinin birbirleriyle değiştirilmesidir. Yani c ürünü, b'nin yerine; b ürünü ise, c'nin yerine yerleştirilmiştir.

Bu durumda şekil 20'de de görüldüğü gibi, toplayıcı ünitenin 1 haftada kat ettiği mesafe toplam 644 metre olmuştur. Örnek 1'de 956 metre mesafe kat edilirken, sadece b'yi yerleştirilen yere c'yi, c'nin yerine de b'yi tercih ederek toplamda bir haftada 312 metre kazanılmıştır. Bu %32,63'lük bir kazanç anlamına gelmektedir. Sadece iki ürünün yerini birbiri ile değiştirerek bu oranda bir mesafe, zaman ve enerji tasarrufu ortaya çıkmıştır.

Sirkülasyon hızının mesafe problemleri üzerindeki etkisinin önemi bu basit örnekte net biçimde ortada durmakta. Bu önemden dolayı, verilerin toplanması esnasında sirkülasyon hızı dikkate alınacak, bu hızı etkileyen faktörler üzerinde durulacak, araştırma buna göre şekillendirilecektir.

**Verilerin Çözümlenmesi ve Yorumlanması**

7 Aralık 2018'den 3 Mart 2022'ye Willebroek tesislerinin veritabanından toplanan 1.239.559 adet giriş kaydının ham verileri incelendiğinde bu kayıtlara ait 144 veriye ulaşılabildiği görülmüştür.

**Ham Verinin Çözümlenmesi**

Ham veri üzerinde tespit edilen 144 veri çeşidi tek tek incelenmiş ve aşağıda açıklanmıştır:

**StokHareketDetayId**: Giriş kaydına ait sıralama numarası.

**StokHareketId**: Giriş kaydının referansına ait sıralama numarası.

**SirketId**: Müşterinin referansı.

**GirisEmriId**: Müşteri siparişinin referansı.

**GirisEmriDetayId**: Müşteri sipariş detaylarının referansı.

**LokasyonId**: Yerleştirilen lokasyon bilgisi.

**KutuId**: Ürünün ait olduğu kutu bilgisi.

**Barkod**: Ürünün kutusunun barkod numarası.

**Kilo**: Ürünün ağırlığı.

**StatuID**: Ürünün anlık statü referansı.

**StatuAdi**: Ürünün anlık statü adı.

**UrunKodu**: Ürünün kodu.

**UrunAdi**: Ürünün adı.

**LokasyonKodu**: Yerleştirilen lokasyonun kod bilgisi.

**AmbarId**: Yerleştirilen ambarın referansı.

**AmbarKodu**: Yerleştirilen ambarın kod bilgisi.

**AMbarAdi**: Yerleştirilen ambarın adı.

**KutuAcik**: Kutunun açık mı kapalı mı olduğu bilgisi.

**KutuPatladi**: Kutunun içinden ürün alınıp alınmadığı bilgisi.

**GirisMiktar**: Ürünün adedi.

**IsEmriNo**: Ürünün müşteri tarafından verilen iş emri numarası.

**BeyannameNo**: Ürünün müşteri tarafından verilen beyanname numarası.

**FaturaNo**: Ürünün müşteri tarafından verilen fatura numarası.

**Gizli**: Ürünün envanterde gösterim bilgisi.

**CikisYasak**: Ürünün çıkışına izin verilip verilmediği bilgisi.

**Tipi**: Ürünün sipariş tipi.

**InsUserId**: Ürünü giriş yapan kullanıcı referansı.

**InsTime**: Ürünün siparişinin sisteme kayıt tarihi.

**DengelemeId**: Dengeleme hareketi referans bilgisi.

**AdjustmentId**: Düzeltme hareketi referans bilgisi.

**GirisEmriDetayKutuId**: Ürünün ilk girdiği kutu referans bilgisi.

**IslemTarihi**: Ürünün girişin okutma tarihi.

**RezervMiktari**: Ürünün çıkış için ayrılan miktarı.

**SirketKodu**: Müşterinin kod bilgisi.

**Bildirildi**: Girişin müşteriye bildirilip bildirilmediği bilgisi.

**BildirilmeZamani**: Girişin müşteriye bildirilme zamanı.

**Marka**: Ürünün markası.

**HareketInsTime**: Ürünün giriş emri kayıt tarihi.

**HareketInsUserId**: Ürünü giriş emrine kaydeden kullanıcı referans bilgisi.

**Koridor**: Ürünün yerleştirildiği koridor bilgisi.

**UrunGrubu**: Ürünün grup bilgisi.

**LokasyonTipId**: Ürünün yerleştirildiği lokasyonun tip referansı.

**GirisEmriAdet**: Ürünün giriş emrinde bildirilen adedi.

**BedenKodu**: Ürünün beden bilgisi.

**BedenAdi**: Ürünün beden bilgisinin adı.

**RenkKodu**: Ürünün renk bilgisi.

**RenkAdi**: Ürünün renk bilgisinin adı.

**AlternatifAdi**: Ürün için varsa kullanılan alternatif isim.

**Boy**: Kutunun boy bilgisi.

**En**: Kutunun en bilgisi.

**Derinlik**: Kutunun derinlik bilgisi.

**KisaAdi**: Ürünün kısa adı.

**ModelKodu**: Ürünün model kodu.

**TipKisaltma**: Ürün tipinin kısa adı.

**TipDeger**: Ürün tipi adı.

**GirisLokasyonTipId**: Girişin yapıldığı lokasyonun tipinin referans değeri.

**LokasyonTipi**: Girişin yapıldığı lokasyonun tipi.

**KonteynerNo**: Varsa ürünün gemide hangi konteynırla taşındığı.

**TakipNotu**: Takip işlemleri için not yazısı.

**SayimTamamlanmaTarihi**: Ürünün depoda sayım okutmasının tamamlanma tarihi.

**Lot**: Ürünün lot numarası.

**SonKullanmaTarihi**: Ürünün son kullanma tarihi.

**SeriNo**: Ürünün seri numarası.

**ETicaretIadeDetayId**: E-ticaret işlemleri için iade detay referansı.

**ETicaretIadeId**: E-ticaret işlemleri için iade referansı.

**FirmaId**: Ürünün tedarikçi referansı.

**FirmaKodu**: Ürünün tedarikçi kodu.

**FirmaAdi**: Ürünün tedarikçi adı.

**SeferNo**: Ürünün ulaştığı seferin numarası.

**Note**: Ürünün girişiyle ilgili not bilgisi.

**Level1**: Ürünün müşteri sınıflandırma bilgisi.

**RampaTarihi**: Ürünün geldiği aracın rampaya giriş tarihi.

**LokasyonKontrolde**: Lokasyonun kontrolde olup olmadığı bilgisi.

**Agirlik**: Ürünün ağırlık bilgisi.

**CBM**: Ürünün hacimsel ölçüsü.

**SealNo**: Ürünün deniz taşımalarında kullanılan mühür numarası.

**TesisId**: Hareketin tesis referansı (Willebroek tesislerinin kodu)

**Kat**: Ürünün yerleştirildiği lokasyonun kat bilgisi.

**Sira**: Ürünün yerleştirildiği lokasyonun sıra bilgisi.

**Bloke**: Ürünün yerleştirildiği lokasyonun bloke bilgisi.

**AmbarAktif**: Ürünün yerleştirildiği ambarın aktif olup olmadığı.

**LokasyonGrupId**: Ürünün yerleştirildiği lokasyonun grup bilgisi.

**SevkPalet**: Ürün sevkedilmişse sevk edilirken kullanılan palet numarası.

**SuperMarket**: Ürünün yerleştirildiği lokasyonun sınıflandırma bilgisi.

**RezervYasak**: Ürünün yerleştirildiği lokasyonun rezerve açık olup olmadığı bilgisi.

**CikisMiktar**: Üründen çıkış varsa, çıkış miktarı.

**KalanMiktar**: Ürünün şu an lokasyonda mevcut miktarı.

**VarsayilanAmbarId**: Ürün için tanımlanmış varsayılan ambar bilgisi.

**StokGirisTarihi**: Stok giriş işlemlerinin tamamlanma tarihi.

**ProformaTarihi**: Ürün için düzenlenen proforma fatura tarihi.

**ProformaNo**: Ürün için düzenlenen proforma fatura no.

**KutuFirmaId**: Kutunun tedarikçi bilgisi.

**LIFO**: Ürünün last in first out bir ürün olup olmadığı.

**KayipLokasyonu**: Ürünün yerleştirildiği lokasyonun kayıp lokasyonu olup olmadığı.

**KutuStatuId**: Kutunun statü referansı.

**ReferansBirimCarpani**: Ürün miktar çevrimi gerekiyorsa çarpanı.

**FaturaBirim**: Ürünün faturalarda kullanılan birimi.

**ReferansBirim**: Ürünün stok hareketlerinde kullanılan birimi.

**KutuTipId**: Kutunun tip referansı.

**KutuTipKodu**: Kutunun tip kodu.

**KutuTipAdi**: Kutunun tip adı.

**KutuIcMiktar**: Kutunun toplam miktarı.

**BlokeAciklama**: Kutu blokeye alınmışsa sebebi.

**FirmaReferansNo**: Müşteri tarafından kutuya verilmiş numara.

**KutuStatuAdi**: Kutunun statü adı.

**Lbs**: Kutunun libre olarak miktarı.

**Storage**: Stoklanması talep edilen bir kutu olup olmadığı.

**AmbarTipId**: Ambarın tip referansı.

**LokasyonPasif**: Lokasyonun pasif olup olmadığı.

**SirketPasif**: Müşterinin pasif olup olmadığı.

**HavuzAlani**: Ürünün yerleştirildiği lokasyonun havuz alanı olup olmadığı.

**VasIslemAlani**: Ürünün yerleştirildiği lokasyonun elleçleme alanı olup olmadığı.

**KutuKayitTarihi**: Kutunun sisteme kayıt tarihi

**ParentKutuBarkod**: Kutu başka bir kutunun içindeki kutuysa ana kutunun barkodu.

**JenerikKod**: Kutu için genel kod numarası.

**UretimTarihi**: Ürünün üretim tarihi.

**TedarikciFirmaId**: Ürünün tedarikçi firma referansı.

**StorageCariId**: Kutu stoklandıysa stok haline getiren firma referansı.

**StorageCariKodu**: Kutu stoklandıysa stok haline getiren firma kodu.

**StorageCariAdi**: Kutu stoklandıysa stok haline getiren firma adı.

**VarsayilanLokasyon**: Ürün için varsayılan lokasyon bilgisi.

**LokasyonHacim**: Ürünün yerleştirildiği lokasyonun hacim bilgisi.

**BeyannameKapatmaZamani**: Ürünün giriş yapıldığı gümrük beyannamesinin kapatılma tarihi.

**KutuMusteriDepo**: Kutunun müşteri tarafında saklandığı depo bilgisi.

**KutuTipGrup**: Kutunun tipinin grup bilgisi.

**KutuMusteriLokasyon**: Kutunun müşteri tarafında saklandığı lokasyon bilgisi.

**BeyannameKapatildi**: Ürünün giriş yapıldığı gümrük beyannamesinin kapatılıp kapatılmadığı.

**AileGrupKodu**: Ürünün aile grup kodu.

**Bloklanmis**: Ürünün blokeye alınıp alınmadığı.

**OrderIndicator**: Ürünün sıralama numarası.

**PO**: Ürünün purchase order belgesinin numarası.

**KutuBatchNo**: Kutuya müşteri tarafından verilen batch numarası.

**KutuIscilikNo**: Kutuya müşteri tarafından verilen işçilik numarası.

**KutuTasimaBirimi**: Kutuya müşteri tarafından verilen taşıma birimi.

**KutuManserNo**: Kutuya müşteri tarafından verilen manşer numarası.

**GirisFaturaTarihi**: Girişin fatura tarihi.

**GirisMusteriKodu**: Girişin müşteri tarafından verilen kodu.

**DengeFaturaNo**: Dengeleme faturası kesildiyse numarası.

**Koordinat**: Kutunun yerleştirildiği lokasyonun x y koordinat sistemine göre koordinat değeri.

**NumericKoridor**: Kutunun yerleştirildiği lokasyonun koridor bilgisinin sayısal karşılığı.

**NumericSira**: Kutunun yerleştirildiği lokasyonun sıra bilgisinin sayısal karşılığı.

**NumericKat**: Kutunun yerleştirildiği lokasyonun kat bilgisinin sayısal karşılığı.

**CikisTarihi**: Kutunun çıkışı yapıldıysa çıkışın yapıldığı tarih.

**IrsaliyeTarihi**: Kutunun çıkış için irsaliyesi kesildiyse irsaliyenin kesildiği tarih.

Ham veri içerisinde yer alan bu 144 alan tek tek incelendiğinde, bazılarının çalışmada odaklanılan sirkülasyon hızı üzerinde önemli bir etkiye sahip olabileceği, bazılarının ise bu hızı etkilemeyeceği varsayılmıştır.

Mevcut HamVeri tablosunun bir örneği **Tablo 8**'de gösterilmiştir.

**Tablo 8.** HamVeri tablosu örneği

| StokHareketDetayId | StokHareketId | SirketId | GirisEmriId | … |
|---|---|---|---|---|
| 118906340 | 63988176 | 100 | 134792189 | … |
| 118869976 | 63942550 | 100 | 134800351 | … |
| 118870662 | 63942597 | 100 | 134800351 | … |
| 118820687 | 63942604 | 100 | 134800351 | … |
| 118870221 | 63942589 | 100 | 134800351 | … |
| 118818966 | 63943056 | 100 | 134726373 | … |
| 118831066 | 63946828 | 100 | 134780455 | … |
| 118873038 | 63942559 | 100 | 134800351 | … |
| 118877945 | 63944765 | 100 | 134746407 | … |
| 118869987 | 63942551 | 100 | 134800351 | … |

Tabloda toplamda 144 alan ve 1.239.559 kayıt bulunmaktadır. 144 alandan ekrana sadece 4 tanesi sığdırılabilmiştir. Sütunların devam ettiğini göstermek için tabloda üç nokta işareti (…) kullanılmıştır.

**Uzman Grubunun Oluşturulması**

Çalışmanın ham verilerin toplanmasının ardından yorumlama aşamasına geçmeden önce, verilerin yorumlanması için Delphi tekniğinin seçilmesine karar verilmiştir. Bu yöntemin tercih edilmesindeki ana nedenler; verilerin sektörel olarak uzmanlık gerektiren birçok alan içermesi ve uzman gözlemine ihtiyaç duyulmasıdır.

Bu doğrultuda; Delphi tekniğinin temel prensiplerine uygun olarak, Willebroek tesislerindeki 1 adet tesis yöneticisi, 1 adet operasyon yöneticisi ve 3 adet vardiya amirinden oluşan bir "uzman grubu" kurulmuştur.

Çalışmanın devamındaki ham verilerin yorumlanması, yorumlanmış verinin gruplanması ve filtrelenmesi gibi işlemler esnasında ihtiyaç duyulan veri alanları için bu ekibe sorular gönderilmiş ve isimsiz olarak yanıtlar toplanmış, daha sonra istatistiksel olarak verilen cevaplar arasından seçimler yapılmıştır.

**Ham Verinin Yorumlanması**

Ham verinin yorumlanması esnasında LDS yazılımının kendi altyapısı gereği ihtiyaç duyduğu ve yorumlamada kullanılmayacak alanlar öncelikle tespit edilmiştir. Bu alanlar arasında yer alan **StokHareketDetayId, StokHareketId, SirketId, InsUserId, InsTime, Bildirildi, BildirilmeZamani, TesisId** alanları yorumlanmış veri setinden kaldırılmıştır. Bu alanlardan 5 tanesiyle ilgili örnek kayıtlar **Tablo 9**'da listelenmiştir.

**Tablo 9.** LDS sisteminin çalışmasına özgü alanlara örnek tablo

| StokHareketDetayId | StokHareketId | SirketId | InsUserId | TesisId |
|---|---|---|---|---|
| 118906340 | 63988176 | 100 | 3063 | 90 |
| 118869976 | 63942550 | 100 | 3053 | 90 |
| 118870662 | 63942597 | 100 | 3376 | 90 |
| 118820687 | 63942604 | 100 | 333 | 90 |
| 118870221 | 63942589 | 100 | 3376 | 90 |
| 118818966 | 63943056 | 100 | 333 | 90 |
| 118831066 | 63946828 | 100 | 3063 | 90 |
| 118873038 | 63942559 | 100 | 3376 | 90 |
| 118877945 | 63944765 | 100 | 3376 | 90 |
| 118869987 | 63942551 | 100 | 3053 | 90 |

Ham verinin incelenmesi esnasında ID, kodu ve adı şeklinde aynı alanların farklı şekilleri olduğu görülmüş bu alanlar sırasıyla ID, kodu ve adı önceliğine göre kullanılmaya karar verilmiş ve geriye kalanlar alanlar yorumlanmış veri setinden kaldırılmıştır. Bu doğrultuda; Barkod, **StatuAdi, UrunAdi, AlternatifAdi, KisaAdi, LokasyonKodu, AmbarKodu, AmbarAdi, SirketKodu, FirmaKodu, FirmaAdi, KutuTipKodu, KutuTipAdi, StorageCariKodu, StorageCariAdi, LokasyonTipi, KutuStatuAdi, TipKisaltma** alanları yorumlanmış veri setine dahil edilmemiştir. Bu alanlardan 5 tanesine ait örnek kayıtların bulunduğu 10 kayıtlık bir liste **Tablo 10**'da gösterilmiştir.

**Tablo 10.** Mükerrer alanlara örnek tablo

| StatuAdi | AmbarKodu | LokasyonKodu | KutuTipAdi | TipKisaltma |
|---|---|---|---|---|
| Depoda | BE44 | 30017005 | B1 | GOO |
| Depoda | BE44 | 242013003 | B2 | GOO |
| Depoda | BE44 | 245015004 | B2 | GOO |
| Depoda | BE44 | 233034001 | B2 | GOO |
| Depoda | BE44 | 246005003 | B2 | GOO |
| Depoda | BE44 | 235010004 | B2 | GOO |
| Depoda | BE44 | 27016003 | B2 | GOO |
| Depoda | BE44 | 246037001 | B2 | GOO |
| Depoda | BE44 | 237006001 | C2 | GOO |
| Depoda | BE44 | 242013003 | C2 | GOO |

Veriler yorumlanırken Willebroek tesislerinin özel durumu incelenmiş, bu tesislerde yapılan operasyonlarda kullanılmamış, sürekli boş veya tekrar eden değere sahip alanlar elenmiştir. Bu tipte görülen **Kilo, BedenKodu, BedenAdi, RenkKodu, RenkAdi, KisaAdi, SeriNo, ETicaretIadeDetayId, ETicaretIadeId, SeferNo, Note, Level1, SealNo, SuperMarket, VarsayilanAmbarId, LIFO, ReferansBirimCarpani, FaturaBirim, ReferansBirim, Lbs, SirketPasif, ParentKutuBarkod, JenerikKod, TedarikciFirmaId, StorageCariId, VarsayilanLokasyon, KutuMusteriDepo, KutuMusteriLokasyon, OrderIndicator, PO, KutuBatchNo, KutuIscilikNo, KutuTasimaBirimi, KutuManserNo, GirisMusteriKodu, DengeFaturaNo, AmbarTipId, KutuIcMiktar** alanları veri setinin son halinden kaldırılmıştır. Bu alanlardan 7 tanesinin yer aldığı örnek 10 kayıt **Tablo 11**'de gösterilmiştir.

**Tablo 11.** İçeriği sürekli tekrarlayan veya hep boş olan kayıtlara örnek tablo

| Kilo | BedenKodu | BedenAdi | RenkKodu | RenkAdi | KisaAdi | SeriNo |
|---|---|---|---|---|---|---|
| NULL | NULL | NULL | NULL | NULL | NULL | NULL |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL |

Veri seti yorumlanırken bazı alanların ürün veya kutunun ilk depoya girişiyle ilgili olmadığı, depo içindeki operasyonlar esnasında belirlendiği görülmüştür. Çalışma kapsamında ürünlerin yerleşim yeri araştırıldığı için, yerleştirildikten sonra oluşacak ve belirlenecek alanların yorumlanmış veri setinde bulunmamasına karar verilmiştir. Bu veriler depoda devam eden operasyon için gerekli alanlar olup, kutunun ilk girişinde belirlenmesi mümkün değildir. Bu alanlar; **LokasyonId, KutuAcik, KutuPatladi, Gizli, CikisYasak, DengelemeId, AdjustmentId, GirisEmriDetayKutuId, RezervMiktari, HareketInsTime, HareketInsUserId, Koridor, LokasyonTipId, GirisLokasyonTipId, SayimTamamlanmaTarihi, LokasyonKontrolde, RampaTarihi, Kat, Sira, Bloke, AmbarAktif, LokasyonGrupId, SevkPalet, RezervYasak, CikisMiktar, KalanMiktar, KayipLokasyonu, KutuStatuId, LokasyonPasif, HavuzAlani, VasIslemAlani, KutuKayitTarihi, BeyannameKapatmaZamani, BeyannameKapatildi, Bloklanmis, GirisFaturaTarihi, Koordinat, NumericKoridor, NumericSira, NumericKat** olarak belirlenmiştir. Bu alanların 6 tanesinin yer aldığı, 10 kayıtlık örnek **Tablo 12**'de gösterilmiştir.

**Tablo 12.** Depo içinde değiştirilen alanlarla ilgili örnek tablo

| LokasyonId | KutuAcik | KutuPatladi | Gizli | CikisYasak | DengelemeId |
|---|---|---|---|---|---|
| 1345019 | 0 | 0 | 0 | 0 | 16960741 |
| 1293394 | 0 | 0 | 0 | 0 | 17013438 |
| 1296787 | 0 | 0 | 0 | 0 | 17010391 |
| 1345347 | 0 | 0 | 0 | 0 | 16960740 |
| 1293434 | 0 | 0 | 0 | 0 | 18096151 |
| 1296230 | 0 | 0 | 0 | 0 | 18007669 |
| 1412259 | 0 | 0 | 0 | 0 | 18150835 |
| 1412639 | 0 | 0 | 0 | 0 | 18150834 |
| 1536578 | 0 | 0 | 1 | 1 | 17013437 |
| 1296418 | 0 | 0 | 0 | 0 | 18321824 |

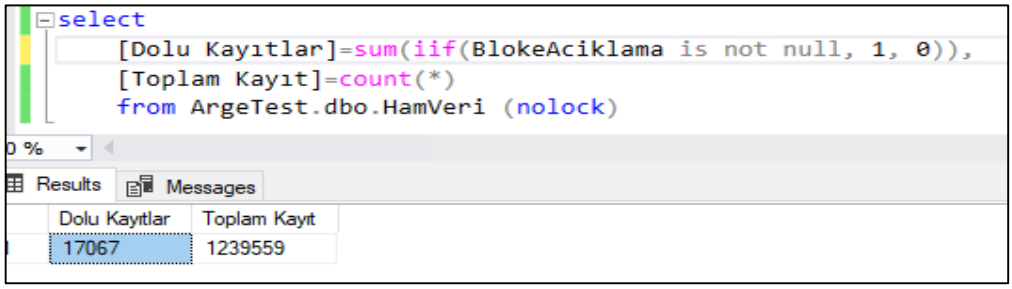
Çalışmanın bu aşamasında veri seti PCA analizine sokulmuştur. **Python** programlama dili içinde bulunan **Sklearn** isimli kütüphanenin hazır kodları kullanılan bu çalışmada bazı alanlar temel bileşenler dışında bırakılmıştır. **AileGrupKodu** ve **BlokeAciklama** alanları bu kapsamda ayrı olarak incelenmiştir.

Ham veri setinde mevut alanlardan **AileGrupKodu** alanının sadece 8.456 kayıtta dolu olduğu görülmüş, toplam 1.239.559 kaydın yaklaşık binde 6'sına denk düşen bu oran dikkate alınarak veri setinden **AileGrupKodu** alanı kaldırılmıştır. Çekilen sorgunun sonucu **Tablo 13**'de gösterilmiştir.

**Tablo 13.** AileGrupKodu alanının toplama göre doluluk durumu

| Dolu Kayıtlar | Toplam Kayıt |
|---|---|
| 8,546 | 1,239,545 |

Ham veri setindeki alanlardan **BlokeAciklama** alanının yalnızca 17.067 kayıtta var olduğu görülmüş, toplam verinin yaklaşık binde 12'sine denk gelmesi nedeniyle bu alan veri setinden kaldırılmıştır. Veri seti üzerinde çekilen örnek sorgu ve sonucu **Şekil 21**'de gösterilmiştir.




**Şekil 21.** BlokeAciklama alanıyla ilgili veri seti üzerinden çekilen sorgu ve sonucunun ekran görüntüsü

Yapılan analizde, ürünün geldiği giriş emri, sipariş, beyanname gibi belge ve kayıtların numaralarının ürünün sirkülasyon hızlarında bir değişikliğe neden olup olmadığı araştırılmıştır. Bu alfa numerik veya nümerik alanlar içinde herhangi bir yorumlanabilir veriye rastlanmamıştır. Bu nedenle **GirisEmriId, GirisEmriDetayId, IsEmriNo, BeyannameNo, FaturaNo, ProformaNo, KonteynerNo, ProformaTarihi, FirmaReferansNo, Storage, UretimTarihi** alanları veri setinden kaldırılmıştır. Alanlarla ilgili 20 kayıtlık örnek veriş **Tablo 14**'de gösterilmiştir.

**Tablo 14.** Giriş emri ile ilgili bilgilerin 20 kayıtlık örnek tablosu

| IsEmriNo | BeyannameNo | FaturaNo | KonteynerNo | FirmaReferansNo | Storage |
|---|---|---|---|---|---|
| 68856214 | 80549662 | 80924926 | NBG1820 | 80924926 | Goodset |
| 68856214 | 80549662 | 80924926 | NBG1820 | 80924926 | Goodset |
| 68856214 | 80549662 | 80924926 | NBG1820 | 80924926 | Goodset |
| 69027703 | 80549918 | 80925288 | OHAKY855 | 80925288 | Goodset |
| 69027703 | 80549918 | 80925288 | OHAKY855 | 80925288 | Goodset |
| 69027703 | 80549918 | 80925288 | OHAKY855 | 80925288 | Goodset |
| 68856214 | 80549662 | 80924926 | NBG1820 | 80924926 | Goodset |
| 69027703 | 80549918 | 80925288 | OHAKY855 | 80925288 | Goodset |
| 69027703 | 80549918 | 80925288 | OHAKY855 | 80925288 | Goodset |
| 69027703 | 80549918 | 80925288 | OHAKY855 | 80925288 | Goodset |
| 68856214 | 80549662 | 80924926 | NBG1820 | 80924926 | Goodset |
| 69027703 | 80549918 | 80925288 | OHAKY855 | 80925288 | Goodset |
| 69027703 | 80549918 | 80925288 | OHAKY855 | 80925288 | Goodset |
| 69027703 | 80549918 | 80925288 | OHAKY855 | 80925288 | Goodset |
| 69027703 | 80549918 | 80925288 | OHAKY855 | 80925288 | Goodset |
| 69027703 | 80549918 | 80925288 | OHAKY855 | 80925288 | Goodset |
| 69027703 | 80549918 | 80925288 | OHAKY855 | 80925288 | Goodset |
| 69048300 | 80549919 | 80925289 | qaff441 | 80925289 | Goodset |
| 69027703 | 80549918 | 80925288 | OHAKY855 | 80925288 | Goodset |
| 69027703 | 80549918 | 80925288 | OHAKY855 | 80925288 | Goodset |

Çalışma kapsamında kutuların ölçüleri, adetleri, yerleştirdiği lokasyon hacimleri gibi fiziksel şartların sirkülasyon hızları üzerinde kayda değer bir etkisi gözlemlenmemiş, verilerin gelişigüzel olduğu sonucuna varılmıştır. Bu sonuçtan kaynaklı olarak **GirisMiktar, GirisEmriAdet, Boy, En, Derinlik, Agirlik, CBM, LokasyonHacim** alanları veri setinden kaldırılmıştır. Analizler esnasında kullanılan verilerden örnek alınmış 30 adet kayıt **Tablo 15**'de gösterilmiştir.

**Tablo 15.** Kutunun fiziki ölçüleriyle sirkülasyon hızı arasında bağlantı kurulamayan analiz çalışmasından örnek 30 kayıt

| Boy | En | Derinlik | Ağırlık | CBM | Depoda Kalınan Gün |
|---|---|---|---|---|---|
| 0,29 | 0,21 | 0,32 | 6,573 | 19,274 | 293 |
| 0,1 | 0,02 | 0,09 | 0,023 | 0,223 | 296 |
| 0,11 | 0,05 | 0,1 | 0,078 | 0,51 | 297 |
| 0,21 | 0,04 | 0,1 | 0,108 | 0,875 | 297 |
| 0,21 | 0,05 | 0,1 | 0,049 | 1,028 | 297 |
| 0,2 | 0,05 | 0,1 | 0,066 | 0,886 | 297 |
| 0,25 | 0,06 | 0,16 | 0,304 | 2,171 | 301 |
| 0,2 | 0,07 | 0,26 | 0,641 | 3,585 | 302 |
| 0,18 | 0,17 | 0,19 | 1,584 | 6,016 | 302 |
| 0,28 | 0,29 | 0,49 | 3,862 | 39,923 | 303 |
| 0,31 | 0,29 | 0,49 | 3,997 | 44,469 | 303 |
| 0,1 | 0,15 | 0,27 | 0,493 | 3,784 | 304 |
| 0,17 | 0,18 | 0,34 | 0,744 | 10,228 | 304 |
| 0,11 | 0,02 | 0,1 | 0,053 | 0,276 | 309 |
| 0,22 | 0,05 | 0,07 | 0,521 | 0,855 | 313 |
| 0,16 | 0,05 | 0,05 | 0,152 | 0,343 | 317 |
| 0,24 | 0,05 | 0,12 | 0,072 | 1,303 | 318 |
| 0,25 | 0,17 | 0,24 | 2,25 | 9,927 | 323 |
| 0,15 | 0,04 | 0,05 | 0,095 | 0,294 | 335 |
| 0,28 | 0,06 | 0,14 | 0,21 | 2,202 | 338 |
| 0,24 | 0,31 | 0,41 | 2,45 | 30,394 | 338 |
| 0,19 | 0,2 | 0,29 | 3,699 | 10,907 | 339 |
| 0,25 | 0,1 | 0,18 | 0,416 | 4,503 | 341 |
| 0,21 | 0,2 | 0,27 | 3,141 | 10,916 | 346 |
| 0,2 | 0,07 | 0,18 | 0,35 | 2,305 | 356 |
| 0,2 | 0,05 | 0,09 | 0,034 | 0,91 | 359 |
| 0,22 | 0,04 | 0,16 | 0,088 | 1,542 | 361 |
| 0,18 | 0,17 | 0,24 | 1,08 | 7,128 | 361 |
| 0,25 | 0,06 | 0,11 | 0,26 | 1,545 | 370 |
| 0,24 | 0,23 | 0,38 | 7,92 | 21,145 | 371 |

Ham veride bulanan 144 alandan geriye elemeler sonucunda 20 alan kalmıştır. Bu 20 alan için tek tek ayrıntılı bir saha analizi yapılmış, veriler incelenmiş ve durum aşağıda belirtilmiştir.

**KutuId:** Kutuların veri tabanındaki kimlik numarasıdır. Veri setinde kutu kimliklendirmesi için kullanılacak ancak verilerin gruplaması aşamasında tekil olduğu için kullanılmayacaktır.

**StatuID:** Girişlerin o andaki giriş durumudur. Kutuların girişlerinin tamamlanması durumunda aldığı değer 30 olmaktadır. Analiz çalışmasında sadece 30 statüsündeki kayıtlar kullanılacaktır. Dolayısıyla veri setinin oluşturulmasında kullanılacak ancak veri setinin içinde yer almayacaktır.

**UrunKodu:** Ürünün kimliğidir. Sirkülasyon hızı üzerinde doğrudan etkisi görülmüştür. Veri setinde kullanılacaktır.

**AmbarId:** Ürünün yerleşeceği ambar bilgisi. Müşteriler ürünlerini kullanım veya saklama amaçlarına göre kendi stok yazılımları üzerinde farklı ambarlarda muhafaza edebilmektedir. İncelenen ham veride BE44, BTRX, BE4S, CLP gibi farklı ambar kodlarına rastlanmış, saha çalışmasında bu ambarlardaki ürünlerin sirkülasyon hızlarında farklılıklar görülmüştür. Veri setinde kullanılacaktır.

**Tipi:** Müşteri tarafından gönderilen ürünün içinde bulunduğu siparişin tip sınıflandırmasıdır. Ham veri içinde SHPMNT, XDOCK, SHUTTLE, RTW, PUTAWAY tiplerinde siparişlere rastlanmıştır. Bu ürünün depoya ne amaçla geldiği ve ne kadar süre depoda kalacağı konusunda etkili bir faktördür. Veri setinde kullanılacaktır.

**IslemTarihi:** Islem tarihi kutunun giriş okutma zamanıdır. Kutunun depoya girdiği zaman olarak okutma zamanı değil, giriş emrinin tamamının işleminin bitip depoya kabul yapıldığı tarih kabul edilecektir. Dolayısıyla veri setinde kullanılmayacaktır.

**Marka:** Ürünün son kullanıcıya yönelik markası. Markaların satış hızlarının değişken olabildiği ve bunun da ürünün sirkülasyon hızı üzerinde etkili olabileceği düşünüldüğünde veri setinde kullanılacaktır.

**UrunGrubu:** Ürünün müşteri tarafından belirlenmiş ana grubudur. Baby Care, Discontinued Professional Salon, Appliances, Skin and Personal Care, Hair Care, Oral Care, Fabric Care, Home Care, Shave Care, Feminine Care, Personal Health Care verileri ham veride rastlanan değerlerdir. Marka gibi ürün sirkülasyon hızında önemli etkisi olabilecek bir alan olduğu için veri setinde kullanılacaktır.

**ModelKodu:** Ürüne müşteri tarafından verilen ikincil bir gruplama kodudur. Sayısal bir alan olduğu için ana detayları müşteri tarafından verilmektedir. Ancak incelemelerde ürünün sınıflandırması üzerinde önemli olduğu görülmüştür. Bu yüzden veri setinde kullanılacaktır.

**TipDeger:** Ürünün hasarlı veya hasarsız olmasını ifade eder. Good ve Damage olarak iki ayrı değer alabilmektedir. Ürünün hasarlı olup olmaması depoda tabi tutulacağı işlemleri ve dolayısıyla kalınan gün sayısını direkt etkileyeceğinden veri setinde kullanılacaktır.

**TakipNotu:** Müşteri tarafından girilen ve müşteri ERP'sindeki bir kodlamayı ifade eden sayısal bir değerdir. Analizde gruplaması belirlenememiş ve sirkülasyon hızına bir etkisi tespit edilememiştir. Veri setinde kullanılmayacaktır.

**Lot:** Ürünün üretim sırasında verilen bir takip numarasıdır. Veri setinde kullanılmayacaktır.

**SonKullanmaTarihi:** Ürünün depodan çıkış hızı üzerinde direkt etkili bir alandır. Veri setinde kullanılacaktır.

**FirmaId:** Ürünün tedarikçi firmasının kodudur. Tedarikçilerden gelen ürünlerin farklı statülerde değerlendirildiği ve sirkülasyon hızı üzerinde etkiye sahip olduğu görülmüştür. Veri setinde kullanılacaktır.

**StokGirisTarihi:** Ürünün giriş emrinin tamamlanıp depoya kabul edildiği tarihtir. Veri setinde giriş tarihi olarak kullanılacaktır.

**KutuFirmaId:** Kutunun kaynak firmasını temsil etmektedir. Yapılan analizde kayıtların büyük bölümünde FirmaId alanı ile aynı olduğu görülmüştür. Karşılaştırmalı yapılan çalışmada veri setinde FirmaId alanının kullanılmasına karar verilmiş, benzer verilere sahip bu alan mükerrerlik oluşmaması adına veri setine dahil edilmemiştir.

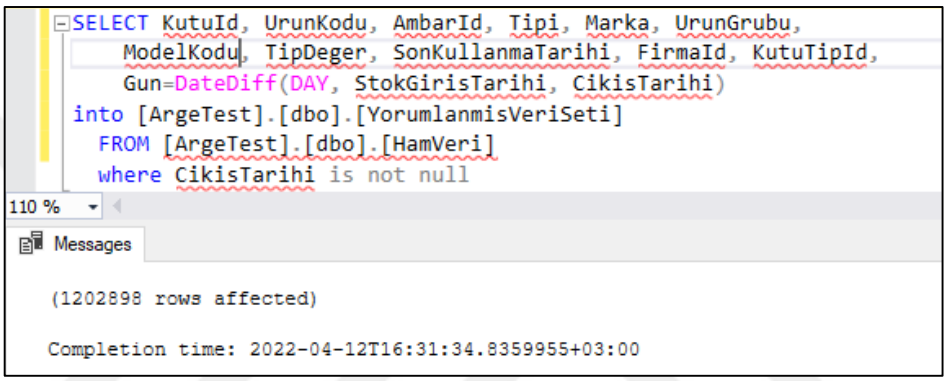
**KutuTipId:** Ürünün muhafaza edildiği kutunun tipini ifade etmektedir. C2, C1, 100X120, B2, P2, PRS Palet, U1, E17, CDock, E14, 99P007, ASRT, P1, OrjinalKutu, K2, K1, OUTL, 998220, B1, P10, StandartPalet, KLT 2020, 0002PAL, EuroPalet şeklinde 24 farklı tipine rastlanmıştır. Kutuların sirkülasyon hızlarına etki edebileceği nedeniyle veri setine dahil edilmiştir.

**KutuTipGrup:** Kutu tiplerin gruplandırılmasıyla oluşturulmuştur. UK ve EURO şeklinde iki varyasyonuna rastlanmıştır. KutuTipId alanının zaten veri setine dahil edilmesi nedeniyle veri setine bu alan dahil edilmemiştir.

**CikisTarihi:** Ürünün depodan çıkış tarihidir. Sirkülasyon hızının belirlenmesinde kullanılacağı için veri setine dahil edilmiştir.

**IrsaliyeTarihi:** Ürünün depodan çıktıktan sonra kesilen irsaliye belgesinin tarihidir. CikisTarihi alanı veri setine dahil edildiği için bu alan dahil edilmemiştir.

Yapılan analiz çalışmasının ardından ham veri üzerindeki **KutuId, UrunKodu, AmbarId, Tipi, Marka, UrunGrubu, ModelKodu, TipDeger, SonKullanmaTarihi, FirmaId, StokGirisTarihi, KutuTipId, CikisTarihi** alanlarının yorumlanmış veri seti üzerinde kullanılmasına karar verilmiştir. Alanlardan **StokGirisTarihi** ve **CikisTarihi** ayrı ayrı kullanılmamış, ikisinin farkından **Gun** isimli yeni bir alan oluşturularak kutunun depoda kaldığı süre bulunmuştur. Yukarıda sayılan 11 alan ve **Gun** isimli alandan, **HamVeri** tablosundan depodan çıkmış bütün kayıtların aktarıldığı **YorumlanmisVeriSeti** isimli yeni bir tablo oluşturulmuştur. Bu tabloyu oluşturmak için kullanılan kod ve sonuç **Şekil 22**'de gösterilmiştir.




**Şekil 22.** Ham veri setinden yorumlanmış veri seti oluşturulurken kullanılan kod ve sonuç

Oluşturulan **YorumlanmisVeriSeti** isimli tablonun örnek kayıtları **Tablo 16**'de gösterilmiştir.

**Tablo 16.** YorumlanmisVeriSeti tablosu örneği

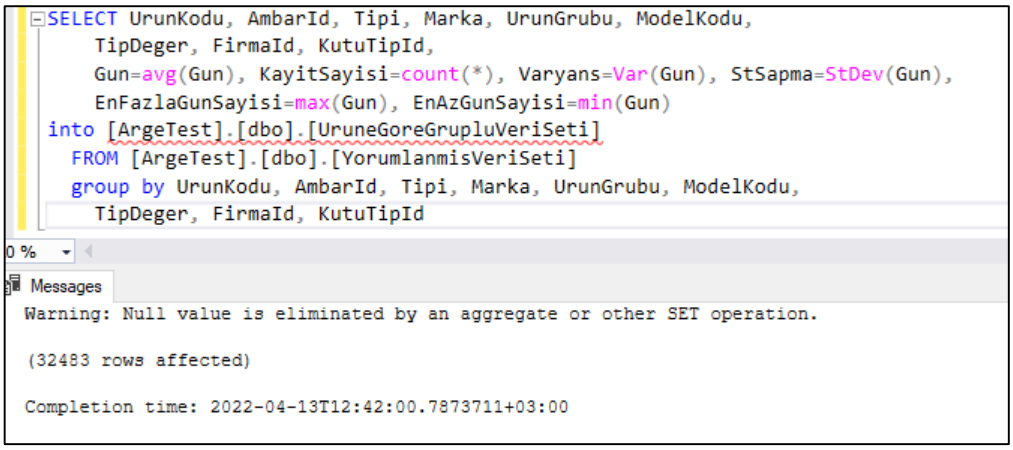
| KutuId | UrunKodu | AmbarId | Tipi | … |
|---|---|---|---|---|
| 14114695 | 81690819 | 518 | PUTAWAY | … |
| 14114688 | 81690819 | 518 | PUTAWAY | … |
| 14114670 | 81690819 | 518 | PUTAWAY | … |
| 14114716 | 81690819 | 518 | PUTAWAY | … |
| 14118094 | 81690819 | 518 | PUTAWAY | … |
| 14119213 | 81690682 | 518 | PUTAWAY | … |
| 14118093 | 81690819 | 518 | PUTAWAY | … |
| 14114719 | 81702279 | 518 | PUTAWAY | … |
| 14114669 | 81690819 | 518 | PUTAWAY | … |
| 14114681 | 81690819 | 518 | PUTAWAY | … |

Ham veri üzerinden henüz depodan çıkılmamış kutuların elenmesiyle birlikte yorumlanmış **YorumlanmisVeriSeti** tablosundaki kayıt sayısı 1.202.898'e düşmüştür.

**Yorumlanmış Verinin Gruplanması**

Yorumlanmış veri seti üzerindeki 12 alandan girişte sorgulamalarda kullanılmak üzere bazı alanları çıkartıp, kalan alanlara göre yeni bir gruplu tablo yapılması için bir çalışma yapılmıştır.

Alanlardan **KutuId**, kutunun kimlik numarası olması nedeniyle, gruplu çalışmada göz ardı edilmiştir. Bu sayede her yeni karşılaşılan kutuda, **KutuId** olmadan kaydın tipini belirlemek için kullanılabilecek **UruneGoreGrupluVeriSeti** isimli yeni bir tablo elde edilmiştir. Bu hesaplamadan ayrıca her gelen üründe farklılaşabilecek olan **SonKullanmaTarihi** alanı da çıkartılmıştır. Ürünlerin depoda kaldığı gün sayısı hesaplanıp **Gun** alanında ortalamaları alınarak tabloya koyulmuştur. **KayitSayisi** alanı içinde bu veri grubuna giren kayıt sayısı saklanmıştır. Hesaplamalarda kullanılmak için Gun alanının varyans değeri **(Varyans)** ve standart sapma değeri **(StSapma)** de tabloya eklenmiştir. **EnFazlaGunSayisi** alanında **Gun** değerleri içindeki en büyük değer, **EnAzGunSayisi** alanında da **Gun** değerleri içindeki en küçük değer saklanmıştır. Tabloyu oluşturmak için gerekli kod ve sonuç **Şekil 23**'te gösterilmiştir.




**Şekil 23.** UruneGoreGrupluVeriSeti tablosunun oluşturulması

Yeni oluşturulan **UruneGoreGrupluVeriSeti** tablosu, **Tablo 17**'de gösterilmiştir.

**Tablo 17.** UruneGoreGrupluVeriSeti tablosu örneği

| UrunKodu | AmbarId | Tipi | Marka | … |
|---|---|---|---|---|
| 64703700 | 518 | SHUTTLE | Oral-B Power | … |
| 64703700 | 518 | XDOCK | Oral-B Power | … |
| 64703709 | 518 | SHUTTLE | Oral-B Power | … |
| 64708764 | 518 | SHUTTLE | Oral-B Power | … |
| 64711702 | 518 | SHUTTLE | Oral-B Power | … |
| 64711704 | 518 | PUTAWAY | Oral-B Power | … |
| 64711706 | 518 | SHUTTLE | Oral-B Power | … |
| 70000-B2 | 518 | XDOCK | Always | … |
| 70005-C2 | 518 | XDOCK | Always | … |
| 70128-K2 | 518 | XDOCK | Always | … |

**Tablo 17**'de veritabanındaki **UruneGoreGrupluVeriSeti** tablosundaki alanlardan sadece **UrunKodu, AmbarId, Tipi, Marka** gösterilebilmiş, diğer alanlar sığmadığından dolayı üç nokta ile ifade edilmiştir. **UruneGoreGrupluVeriSeti** içinde bu 4 alan dışında **UrunGrubu, ModelKodu, TipDeger, FirmaId, KutuTipId, Gun, KayitSayisi, Varyans, StSapma, EnFazlaGunSayisi, EnAzGunSayisi** alanları da yer almaktadır.

Yeni oluşturulan tablo sayesinde **KutuId**'ye ihtiyaç olmadan yapılacak sorgulamalarda, 1.202.898 adetlik **YorumlanmisVeriSeti** tablosuna yerine 32.348 kayıtlık **UruneGoreGrupluVeriSeti** tablosu kullanılabilecektir. Bu tablo aynı zamanda yorumlanmış veri seti üzerindeki standart sapma dışındaki kalan kayıtların temizlenmesi için kullanılacaktır.

**Yorumlanmış Verinin Yeniden Filtrelenmesi**

**UruneGoreGrupluVeriSeti** tablosu hazırlandıktan sonra, gruplanmış her kaydın depoda kaldığı ortalama gün ve gün değerleri için standart sapmalar belirlenmiştir. Standart sapmanın yüksek olduğu ürünler bulunmasından dolayı, aykırı (outlier) olabilecek kayıtlar olabileceği sonucuna varılmış ve ham veri bir aykırı değer taramasına tabi tutulmuştur. Bu taramada **uç değer belirleme yöntemi** kullanılmıştır. Uç değer bulmak içinse **çeyrekler açıklığı metodu** kullanılmıştır. Çeyrekler açıklığı metodunun denklemi **Denklem 1**'de gösterilmiştir.

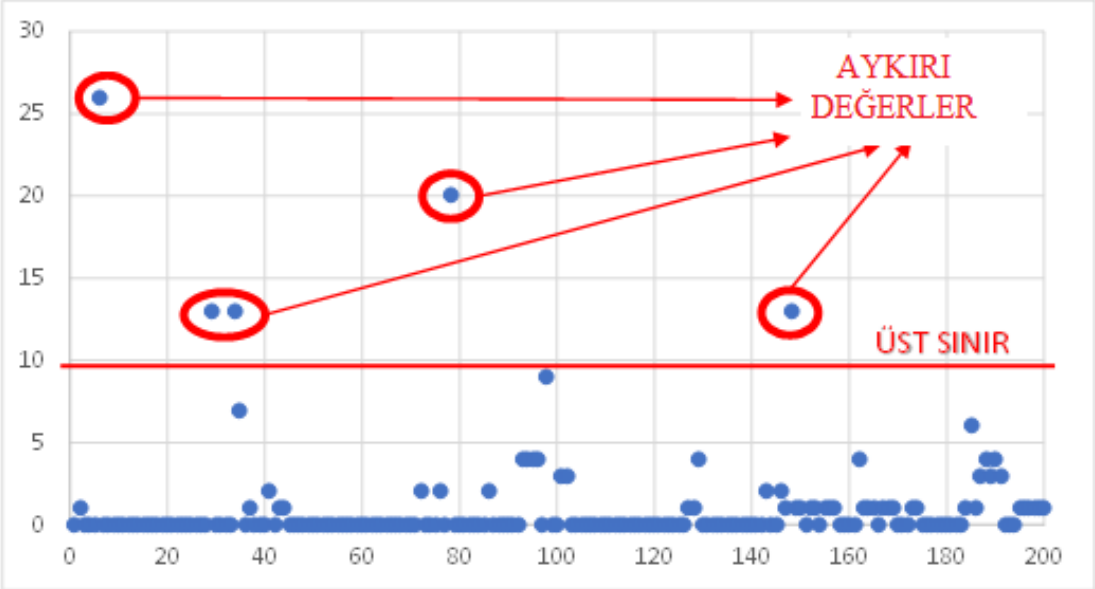
**Denklem 1.** Çeyrekler açıklığı denklemi

$$Q = Q3 - Q1$$

Çeyrekler açıklığı metodunda veriler artan sırada düzenlenmiştir. Birinci çeyrek (q1) ve üçüncü çeyrek (q3) içine giren kayıtlar tespit edilmiştir. Bu çeyrekler arasındaki mesafe *Q3 – Q1* işlemiyle bulunmuştur.

Metodun SQL sunucularında kullanımı ile ilgili Barnes (2021) çalışmasından faydalanılmıştır (Barnes, 2021). **UruneGoreGrupluVeriSeti** içindeki her kayıt için alt sınır ve üst sınır bu yöntemle belirlenmiştir. O grupla ilgili **YorumlanmisVeriSeti** içinde, grubun alt ve üst sınırı dışında olan kayıtlar bulunup tablodan silinmiştir.

Örnek olarak 183 sıra numaralı kayıt seçilmiş ve bu kaydın dağılım grafiği üzerinde, bulunan üst sınır ve sınır dışında kalan kayıtları **Şekil 24**'te gösterilmiştir.



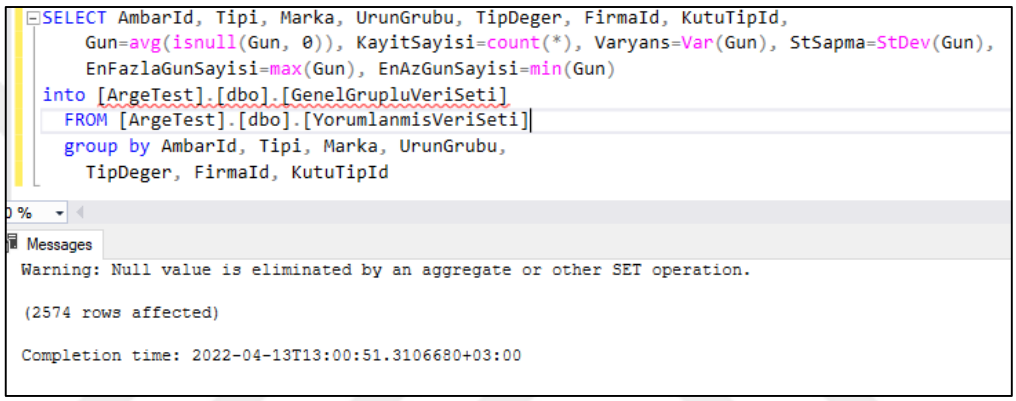

**Şekil 24.** Standart dışı kayıtların 183 sıra numaralı örnek üzerinden grafik gösterimi

Silme işleminin ardından **YorumlanmisVeriSeti** tablosundaki kayıt sayısı 1.029.856 olmuştur.

### Filtrelenmiş Verinin Yeniden Gruplanması

Veriler ürünlere göre filtrelendikten sonra kutu ve üründen bağımsız olarak yeniden gruplanarak bir tabloya aktarılmıştır. Hem kutu ile ilgili hem de ürünle ilgili özel bilgilere ihtiyaç olmaması durumunda az kayıt arasından hızlı seçim yapılabilmesi için; kutu ile ilgili **KutuId** ve ürünle ilgili **UrunKodu, ModelKodu, SonKullanmaTarihi** bilgilerinin

çıkarıldığı **GenelGrupluVeriSeti** isimli yeni bir tablo yaratılmıştır. **AmbarId, Tipi, Marka, UrunGrubu, TipDeger, FirmaId** ve **KutuTipId** alanlarına göre gruplanan bu tabloda, gün sayısının tutulduğu **Gun** alanı, ortalama alınarak belirlenmiştir. **KayitSayisi** alanı içinde bu veri grubuna giren kayıt sayısı saklanmıştır. Hesaplamalarda kullanılmak için **Gun** alanının varyans değeri **(Varyans)** ve standart sapma değeri **(StSapma)** de tabloya eklenmiştir. **EnFazlaGunSayisi** alanında **Gun** değerleri içindeki en büyük değer, **EnAzGunSayisi** alanında da **Gun** değerleri içindeki en küçük değer saklanmıştır. Tablonun Microsoft SQL veritabanı üzerinde oluşturulma süreci **Şekil 25**'de gösterilmiştir.




**Şekil 25.** GenelGrupluVeriSeti tablosunun oluşturulma süreci ve sonucu

Yeni oluşturulan **GenelGrupluVeriSeti** tablosu, **Tablo 18**'te gösterilmiştir.

**Tablo 18.** GenelGrupluVeriSeti tablosu örneği

| AmbarId | Tipi | Marka | UrunGrubu | … |
|---|---|---|---|---|
| 518 | SHPMNT | Always | Feminine Care | … |
| 518 | SHUTTLE | Always | Feminine Care | … |
| 518 | SHUTTLE | Ambi Pur | Home Care | … |
| 518 | SHUTTLE | Ambi Pur | Home Care | … |
| 518 | PUTAWAY | Antikal | Home Care | … |
| 837 | PUTAWAY | Antikal | Home Care | … |
| 837 | XDOCK | Ariel | Fabric Care | … |
| 837 | XDOCK | Ariel | Fabric Care | … |
| 837 | XDOCK | Ariel | Fabric Care | … |
| 851 | RTW | Ariel Professional | Fabric Care | … |

**Tablo 18**'de veri tabanındaki **GenelGrupluVeriSeti** tablosundaki alanlardan sadece **AmbarId, Tipi, Marka, UrunGrubu** gösterilebilmiş, diğer alanlar sığmadığından

dolayı üç nokta ile ifade edilmiştir. **GenelGrupluVeriSeti** içinde bu 4 alan dışında **UrunGrubu, TipDeger, FirmaId, KutuTipId, Gun, KayitSayisi, Varyans, StSapma, EnFazlaGunSayisi, EnAzGunSayisi** alanları da yer almaktadır.

**GenelGrupluVeriSeti** tablosu 2.574 kayıt saklamaktadır ve hazırlanacak uygulamalarda, alanların yeterli olduğu durumlarda hızlı sorgular çekebilmek için kullanılacaktır. Aynı zamanda filtrelenmiş olması nedeniyle, ortalamalar ve standart sapmalar ilk gruplandırmaya göre daha standart hale getirilmiş durumdadır.

# BÖLÜM 4. UYGULAMA

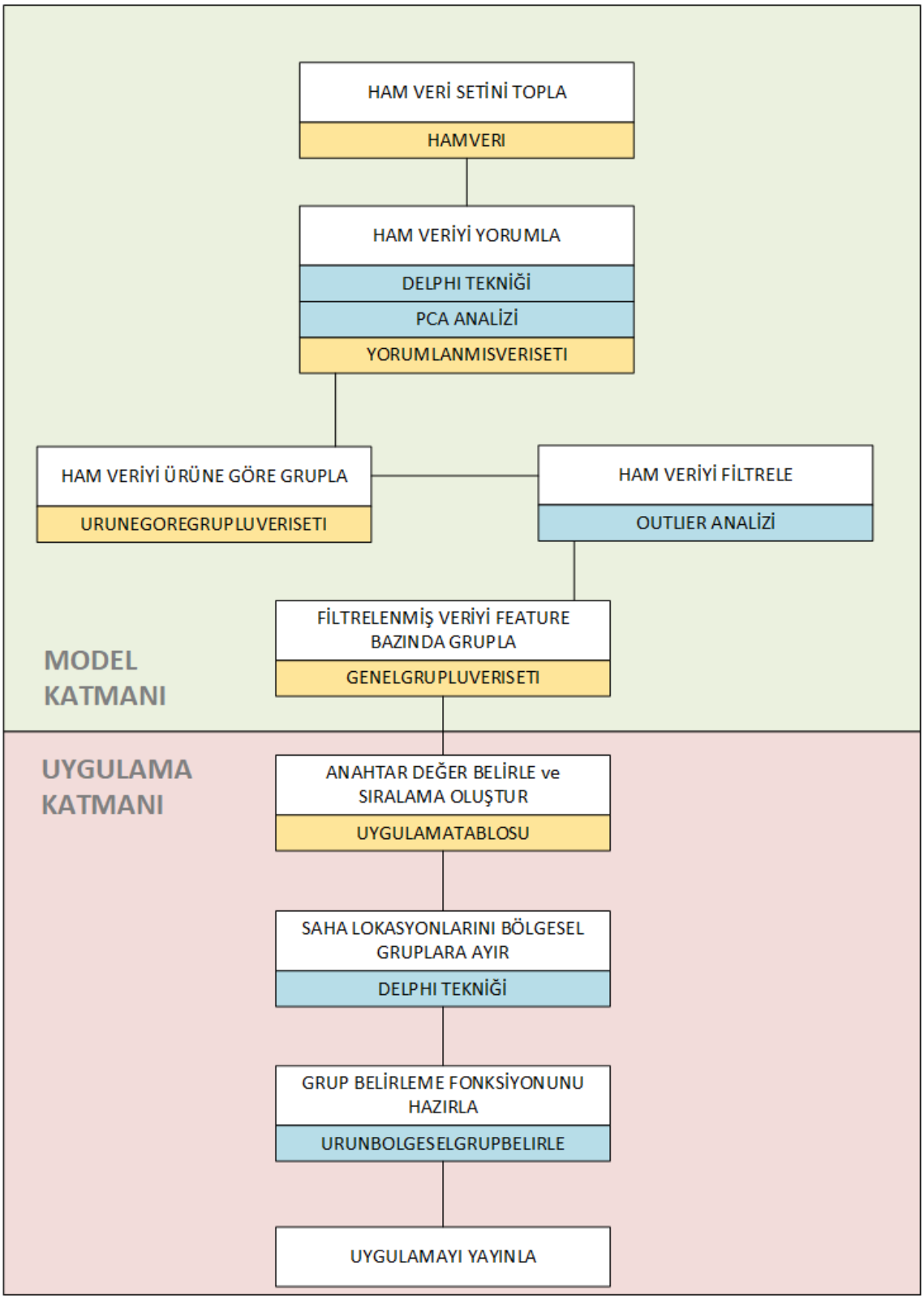
Uygulamanın temel amacı, depoya gelen bir ürünün, geliş anında bu ürün için bir **sirkülasyon hızı anahtar değeri** hesaplanması ve bu anahtar değerine göre giriş ve çıkış toplama alanına yerleştirileceği lokasyonun uzaklığının tespit edilmesidir. Bu esnada özel olarak bir mesafe hesaplanmayacak, mesafelere göre lokasyon gruplarını oluşturulup bu ürünün kendi anahtar değerine göre bir lokasyona yerleştirilmesi sağlanacaktır. Bu sayede, depoda kısa süre duran ürünler toplama alanlarına en yakın lokasyonlara, uzun süre duran ürünler ise en uzak lokasyona yerleştirilecektir.

Uygulama aşamasında daha önceden oluşturulan **Microsoft SQL Server** tabanlı veritabanı kullanılmıştır. Barsan Global Lojistik'in Frankfurt'ta bulunan test SQL sunucusu üzerinde bulunan bu veritabanı üzerinde **T-SQL** sorgulama dili kullanılarak sorgular oluşturulmuştur.

Veri seti üzerinde temel bileşenlerin bulunmasında ve aykırı değer bulma gibi işlemlerde **Python** programlama dilinde kodlar yazılmıştır. Bu kodların yazılması için **Spyder** editörü kullanılmış ve Python 3.6 versiyonu tercih edilmiştir. Yine bu işlemler esnasında Python programla dili için hazırlanılmış **Sklearn** ve **MatPlotLib** kütüphanelerinden de faydalanılmıştır.

Sahadaki test çalışmaları esnasında hazırlanan uygulamanın Willebroek tesislerinde kullanılan LDS yazılımına entegre edilmesi için, programın yazıldığı **Object Pascal** programlama dilinde kodlar yazılmış, bu kodların yazılması için **Delphi 2007** tümleşik geliştirme ortamı kullanılmıştır.

Uygulamanın bu aşamadan önceki ve sonraki süreçlerinin akış diyagramı **Şekil 26**'da gösterilmiştir. Akış diyagramının üst yarısındaki **yeşil** renkli bölüm **model oluşturma katmanını,** alt yarısındaki **kırmızı** renkli bölüm ise **uygulama geliştirme katmanını** göstermektedir. Akış diyagramı üzerinde **beyaz** renkli kutular aşama açıklamalarını, **mavi** renkli kutular kullanılan tekniği, **sarı** renkli kutular ise aşama sonunda oluşan veritabanı tablosunun ismini ifade etmektedir.




**Şekil 26.** Uygulama akış diyagramı

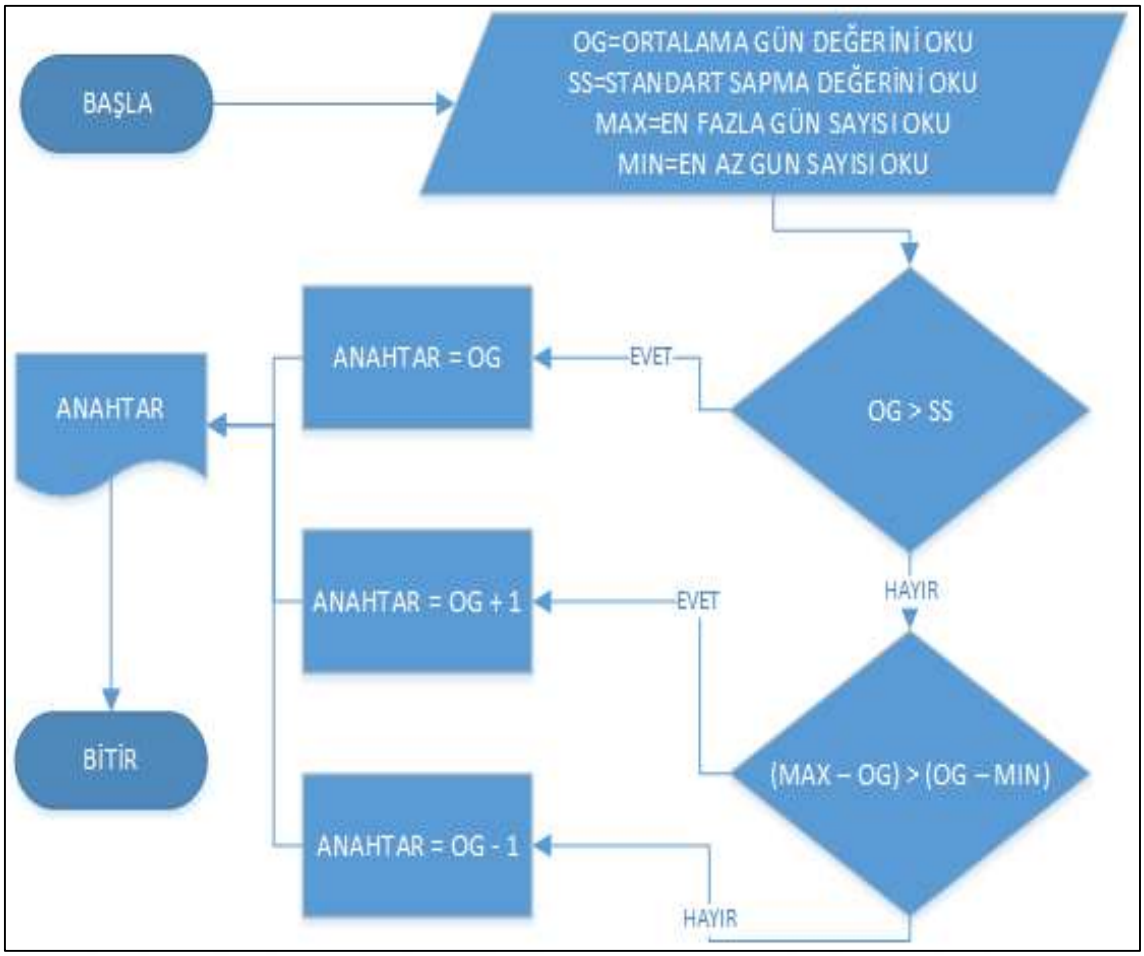
**Şekil 26**'da görülebileceği üzere; uygulamanın model katmanı, ham veri setinin toplanması ve **HAMVERI** isimli veri setinin ortaya çıkmasıyla başlamakta; daha sonra

bu oluşturulan veri seti Delphi Tekniği ve PCA Analizi yöntemleriyle yorumlanıp **YORUMLANMISVERISETI** oluşturulmaktadır. Daha sonra oluşturulan bu yeni veri seti ise ürün gruplama işlemine tabi tutulmakta ve ortaya **URUNEGOREGRUPLUVERISETI** çıkmakta, bu veri seti ise Outlier Analizi ile filtrelenmektedir. Aykırı kayıtlar bu veri setinden temizlendikten sonra daha önceden belirlenen feature'lar bazında gruplama işlemi yapılıp **GENELGRUPLUVERISETI** oluşturulmaktadır.

Akış diyagramının uygulama katmanı, ilerleyen bölümlerde de anlatılacağı gibi, anahtar değer belirlenmesi işleminin ardından **UYGULAMATABLOSU** isimli son veri seti oluşturulup uygulamanın besleneceği verilerin bu veri setine yerleştirilmesiyle başlamaktadır. Daha sonra saha lokasyonların bölgesel gruplara ayrılması işlemi sırasında yine Delphi Tekniği kullanılmıştır. Uygulamanın ana fonksiyonu olan grup belirleme fonksiyonu hazırlanıp bu fonksiyona **URUNBOLGESELGRUPBELIRLE** ismi verilmiştir. Son aşama ise uygulamanın yayınlanmasıdır.

**Anahtar Değerinin Belirlenme Yöntemi**

Oluşturulan veri setinde, kaydın uygulamadaki karşılaştırmalarda kullanılması için bir anahtar değerine ihtiyaç duyulmuştur. Uzun haliyle sirkülasyon hızı anahtar değeri denebilecek bu değer, ürün grubunun depoda kaldığı ortalama gün sayılarına, aynı grubun standart sapma değeri ile karşılaştırılması sonucu bulunmuştur. Eğer gün sayısı standart sapmanın altında ise anahtar değeri olarak gün sayısı kullanılmıştır. Eğer standart sapma gün sayısından büyükse, bu durumda en az gün değeri ve en yüksek gün değerinin ortalama ile farkına bakılmıştır. En büyük değer, ortalamaya daha uzaksa anahtar değeri için gün sayısına 1 eklenmiş, en düşük değer ortalamaya daha uzaksa gün sayısından 1 çıkarılarak anahtar değer bulunmuştur. Bu sayede, standart sapması yüksek kayıtlarda, bir aşağı veya bir yukarı kayılarak beklenti aşağı veya yukarı yönlü şekilde belirlenmiştir. Algoritmanın akış diyagramı **Şekil 27**'de gösterilmiştir.




**Şekil 27.** Anahtar değeri oluşturma algoritmasının akış diyagramı

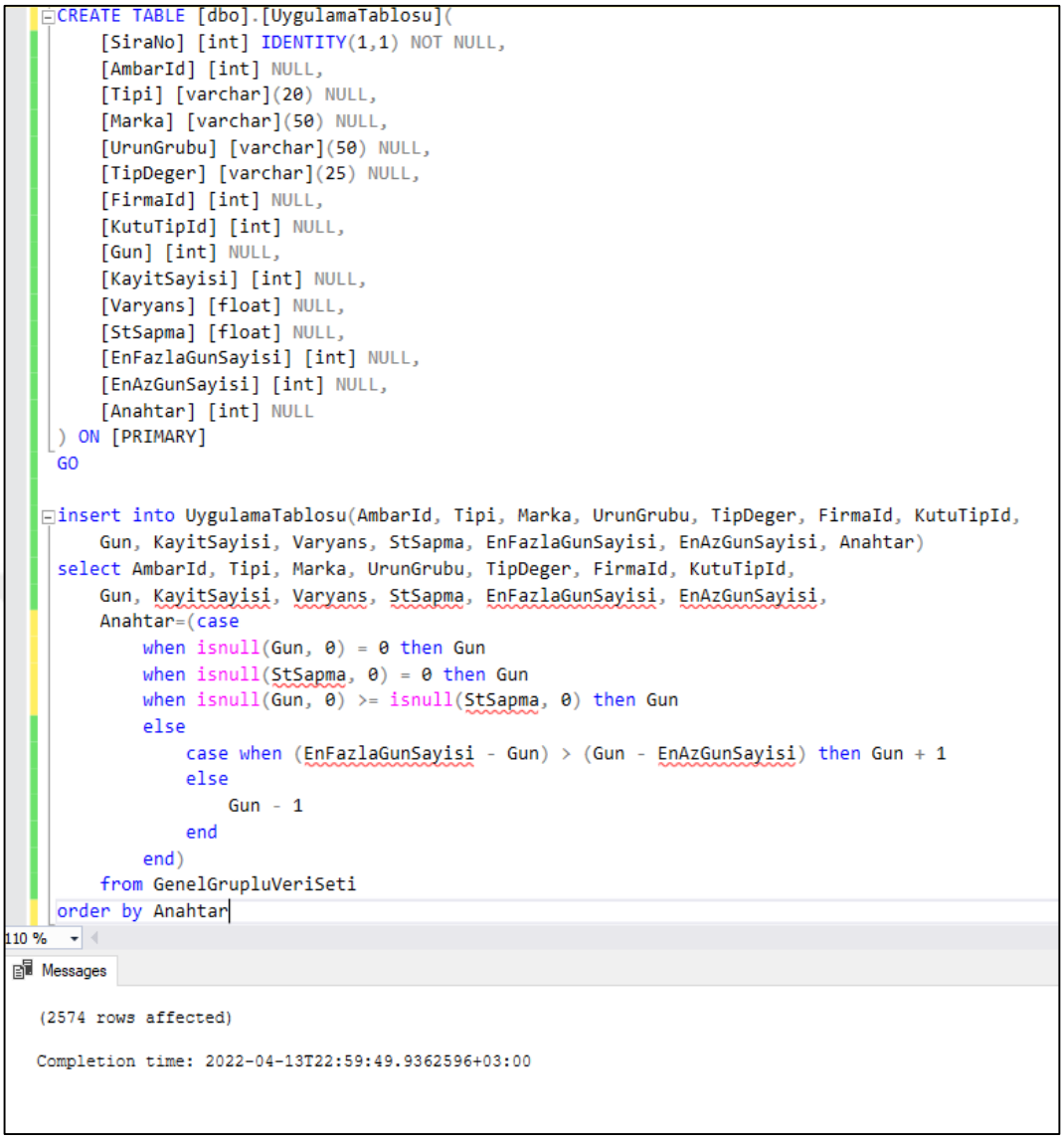
Bu doğrultuda 2.574 kayıtlı gruplanmış veri setimizin her bir kaydı için ortalama gün sayıları üzerinden yeni bir anahtar değer oluşturulmuştur. Anahtar değerleri oluşturulmuş kayıtlardan 20 tane örnek kayıt **Tablo 19**'da gösterilmiştir.

**Tablo 19.** Anahtar değer hesaplamalarına örnek 20 kayıtlık tablo

| Marka | Ürün Grubu | Gün | Kayıt | St, Sapma | En Fazla Gün | En Az Gün | Anahtar |
|---|---|---|---|---|---|---|---|
| Lenor | Fabric Care | 17 | 3 | 26,56 | 48 | 2 | 18 |
| Head & Shoulders | Hair Care | 17 | 752 | 50,21 | 506 | 0 | 18 |
| Flash | Home Care | 17 | 6 | 19,84 | 55 | 0 | 18 |
| Ariel Professional | Fabric Care | 19 | 2 | 19,80 | 33 | 5 | 18 |
| Old Spice | Skin and Personal Care | 17 | 25 | 51,56 | 226 | 3 | 18 |
| Lenor Professional | Fabric Care | 18 | 24 | 5,44 | 32 | 13 | 18 |
| Tampax | Feminine Care | 18 | 1 | 0 | 18 | 18 | 18 |
| Gillette Series | Shave Care | 18 | 5 | 3,29 | 21 | 15 | 18 |
| Dodot | Baby Care | 17 | 601 | 18,90 | 59 | 0 | 18 |
| Old Spice | Skin and Personal Care | 17 | 222 | 39,58 | 288 | 0 | 18 |
| Always | Feminine Care | 17 | 85 | 34,06 | 172 | 0 | 18 |
| Fairy | Fabric Care | 17 | 67 | 38,87 | 272 | 0 | 18 |
| Gillette Series | Shave Care | 17 | 32 | 19,68 | 68 | 0 | 18 |
| Pantene | Hair Care | 17 | 432 | 33,58 | 269 | 0 | 18 |
| Olay | Skin and Personal Care | 19 | 2 | 26,87 | 38 | 0 | 18 |
| Lenor | Fabric Care | 17 | 473 | 31,77 | 249 | 0 | 18 |
| Fixodent | Oral Care | 18 | 5 | 0,00 | 18 | 18 | 18 |
| Fusion | Shave Care | 17 | 18 | 20,15 | 64 | 0 | 18 |
| Head & Shoulders | Hair Care | 19 | 3 | 8,02 | 27 | 11 | 19 |
| Head & Shoulders | Hair Care | 18 | 89 | 20,12 | 82 | 0 | 19 |

Uygulama, ürün yerleştirmeleri esnasında ortalama gün sayıları ve standart sapmalar bir arada değerlendirilmiş olan bu anahtar değerlerini referans alarak çalışacaktır.

Uygulamanın kullanacağı son tabloda, bu anahtar değerleri kayıtlara atanmış ve bu anahtar değerlerine göre kayıtlar sıralanıp bir sıra numarası (**SiraNo**) verilerek kimlik kazandırılmıştır. İlgili SQL kodu **Şekil 28**'de gösterilmiştir.




**Şekil 28.** Uygulama için anahtar değeri belirlenmiş ve bu değere göre sıralanmış UygulamaTablosu isimli tablonun yaratılması

**UygulamaTablosu** isimli tablo, **Tablo 20**'de gösterilmiştir. Uygulama tablosunda **SiraNo, AmbarId, Tipi, Marka, UrunGrubu, TipDeger, FirmaId, KutuTipId, Gun, KayitSayisi, Varyans, StSapma, EnAzGunSayisi, EnFazlaGunSayisi, Anahtar** şeklinde 15 alan olmasına rağmen örnek tabloda yer sorunundan dolayı sadece **SiraNo, AmbarId, Tipi, Marka** ve **UrunGrubu** alanları listelenmiş, diğer alanlar üç nokta (…) ile gösterilmiştir.

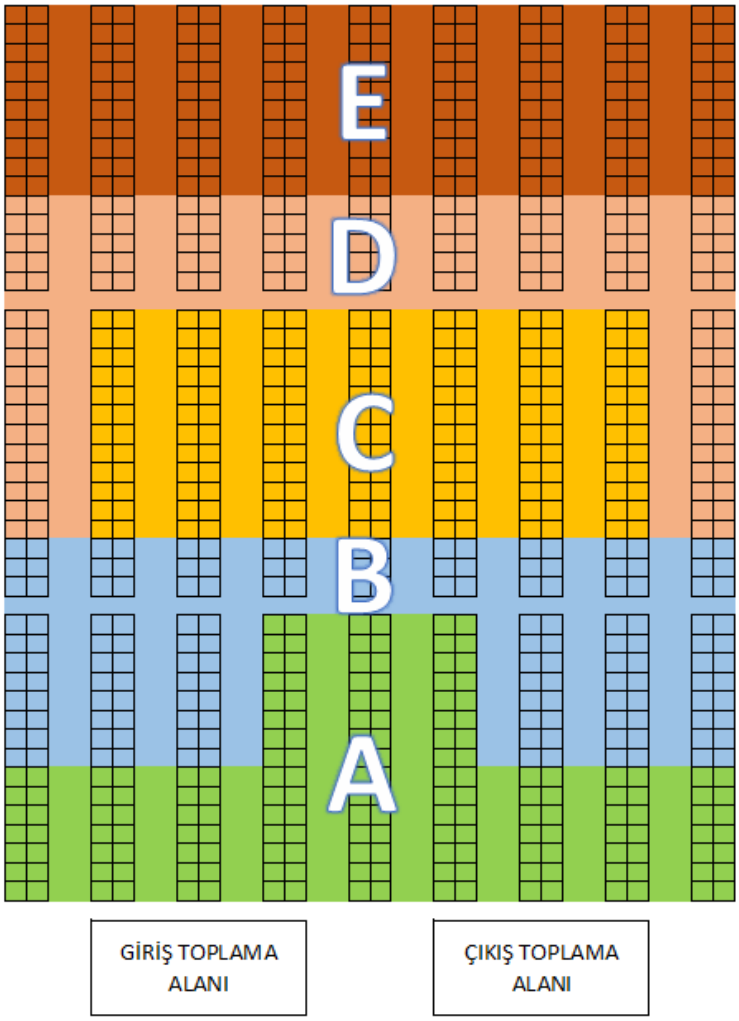
**Tablo 20.** UygulamaTablosu tablosu örneği

| SiraNo | AmbarId | Tipi | Marka | UrunGrubu | … |
|---|---|---|---|---|---|
| **1** | 518 | RTW | Ariel Professional | Fabric Care | … |
| **2** | 518 | RTW | Ariel Professional | Fabric Care | … |
| **3** | 518 | RTW | AUSSIE | Hair Care | … |
| **4** | 518 | RTW | Ariel | Fabric Care | … |
| **5** | 518 | RTW | Aussie | Hair Care | … |
| **6** | 518 | RTW | Blue 3 | Shave Care | … |
| **7** | 837 | XDOCK | Cascade | Home Care | … |
| **8** | 837 | XDOCK | Crest | Oral Care | … |
| **9** | 837 | XDOCK | Dreft | Home Care | … |
| **10** | 518 | RTW | Fairy | Fabric Care | … |

Eldeki ham veri çözümlenmiş, yorumlanmış, gruplanmış, standart sapmalara göre filtrelenmiş, filtrelenmiş haliyle tekrar gruplanmış ve son aşamada 2.574 kayıtlı, anahtar değerine sahip ve bu değere göre sıralı bir **UygulamaTablosu** oluşturulmuştur. Çalışmanın ilerleyen bölümlerinde bu tablo kullanılacaktır.

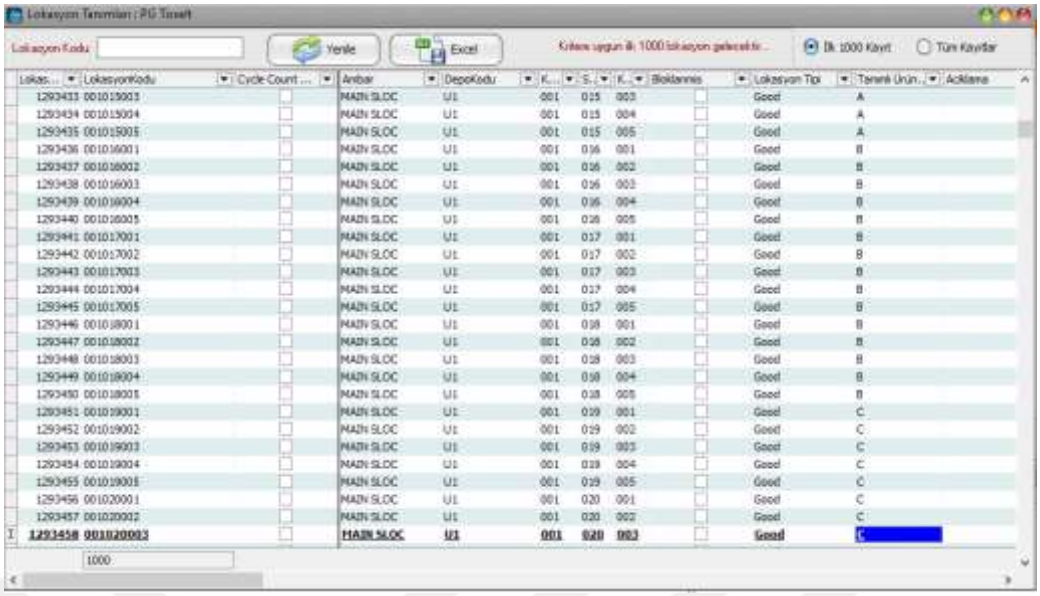
**Lokasyon Gruplamalarının Belirlenmesi**

Çalışma kapsamında incelenen Willebroek tesislerindeki lokasyonlama sistemi 9 koridora ayrılmış 45 sıra çift taraflı raftan oluşmaktadır. Her bir sırada 5 kat ayrı lokasyonlama bölmesi bulunmaktadır. Bu toplamda 4.050 ayrı lokasyon adresine denk gelmektedir. Yapılan çalışmada her biri yaklaşık 810 adet lokasyondan toplam 5 adet bölge belirlenmiştir. Bu 5 bölge toplama alanlarına uzaklığına göre A, B, C, D ve E bölgeleri olarak isimlendirilmiştir. A bölgesi, toplama alanlarına en yakın bölgedir. Ve bu bölgeye sirkülasyon hızı en yüksek ürünlerin konması, toplayıcıların kat edeceği mesafelerin daha kısa olmasını sağlayacaktır. C bölgesi, ortalama uzaklıktaki bölgedir. Bu bölgeye uygulamanın bölgesini belirleyemediği ürünlerin yerleştirilmesi planlanacaktır. Geriye kalan A, B, D ve E bölgelerine sirkülasyon hızına göre uygulama tarafından yerleştirme yapılacak, yerleştirme önerisi bulunamayan ürünler ise giriş esnasında C bölgesine yönlendirilecektir. Willebroek tesislerinin lokasyon planı ve belirlenen bölgeler **Şekil 29**'de gösterilmiştir.




**Şekil 29.** Willebroek tesisleri için oluşturulan A, B, C, D, E bölgeleri

Tesis içinde belirlenen bölgeler, tek tek adresleri ile tesiste kullanılan LDS programı tarafından tanınabilmekte ve uygulamada kullanılan A, B, C, D ve E grupları da bu lokasyon tanımlarına eklenebilmektedir. LDS programında yapılan grup tanımları **Şekil 30**'da gösterilmiştir.



**Şekil 30.** LDS programı üzerinde yapılan bölge grup tanımları

Bu 5 gruptan bir tanesi uygulamanın karar sürecinden yanıt alınamayan ürünlere ayrıldığı için, uygulama verilerine göre ürün gruplarını dörde ayırmak durumunda. Ancak bu işlemi yaparken, **UygulamaTablosu** isimli tablodaki 2.574 kaydı 4'e bölerek işlem yapmak istenen sonucu vermeyecektir. Çünkü veri setimizi yorumlarken, depoya 1 kez veya 1.000 kez girmiş olsun, aynı grubun içine giren ürünler, veri setimizin son halinde tek bir kayıt olarak yorumlanmıştı. İkisini aynı şekilde değerlendirmek, depoya çok ender giren ürünleri, defalarca kez giren ürünlerle aynı statüde değerlendirme yanılgısına yol açacaktır. Bu da aşağı yukarı aynı sayılarda lokasyon ayrılan grupların envanter yönetimini zorlaştıracak, bazı bölümlerin erken dolmasına, bazı bölümlerin ise boş kalmasına neden olacaktır.

Bu nedenle, 2.574 kayıt 4'e bölünüp her 643 grup için bir bölge atamak yerine, her grubun kendi içindeki kayıt sayıları dikkate alınarak bölge gruplandırması yapılacaktır. Bu yüzden **UygulamaTablosu**'ndaki 2.574 kayıt değil, **YorumlanmisVeriSeti** tablosundaki 1.029.856 kayıt esas alınacak, bu kayıt sayısına göre ürünler 4'e ayrılarak ürünlerin depoya giriş çıkış hızlarına göre, depo içinde belirlenmiş 4 bölgenin yaklaşık aynı sayılarda ürünü muhafaza etmesi sağlanıp, bölgelerin doluluk oranları birbirine yakın tutulacaktır.

Bu nedenle tablo üzerine sıralı bir şekilde o sıraya kadar kayıtların, toplam yorumlanmış grupsuz veri seti üzerindeki kayıt sayıları **KumulatifKayitSayisi** diye bir alan açılıp bu tabloya yazılmıştır. Sonrasında bu sayı, tablonun tamamındaki kayıt sayısı dörde

bölündüğünde ortaya çıkan sayıya göre içine dahil olduğu grup belirlenmiş, bu grup A, B, D ve E harfleri ile isimlendirilerek **BolgeGrup** isimli bir alanın içine yazılmıştır. Bu işlemlere ait yazılan kod **Şekil 31**'da gösterilmiştir.

```
alter Table UygulamaTablosu
drop column BolgeGrup;

alter Table UygulamaTablosu
add KumulatifKayitSayisi int;

update U1
    set KumulatifKayitSayisi = (select sum(KayitSayisi) from UygulamaTablosu U2 where U2.SiraNo <= U1.SiraNo)
from UygulamaTablosu U1

alter Table UygulamaTablosu
add BolgeGrup varchar(1);

update U1
    set BolgeGrup =
    case
        when KumulatifKayitSayisi < ((select sum(KayitSayisi) from UygulamaTablosu U2) / 4) * 1 then 'A'
        when KumulatifKayitSayisi < ((select sum(KayitSayisi) from UygulamaTablosu U2) / 4) * 2 then 'B'
        when KumulatifKayitSayisi < ((select sum(KayitSayisi) from UygulamaTablosu U2) / 4) * 3 then 'D'
        else 'E' end
from UygulamaTablosu U1

select U.BolgeGrup, Min=min(Anahtar), Max=max(Anahtar), Sayi=count(*)
    into #AnahtarTablosu
from UygulamaTablosu U
group by U.BolgeGrup
order by U.BolgeGrup

update U1
    set BolgeGrup = A.BolgeGrup
    from UygulamaTablosu U1
    outer apply (select * from #AnahtarTablosu A
        where U1.Anahtar > A.Min
            and U1.Anahtar <= A.Max) A
    where A.BolgeGrup is not null

drop table #AnahtarTablosu
```

**Şekil 31.** UygulamaTablosu'na BolgeGrup isimli bir alan eklenerek bölgesel gruplandırma yapılması

Bölge gruplandırma esnasında C grubu kullanılmamıştır. Daha önce de bahsedildiği gibi C bölgesi, uygulama tarafından grup atanamayan ürünlere ayrılmıştır. Ayrıca aynı değere sahip kayıtların farklı bölgelerde sınıflandırılmaması için son olarak geçici bir **#AnahtarTablosu** isimli tablo yaratılarak, o tablodan alınan en yüksek ve en düşük değerlere göre kayıtlar tekrar güncellenmiştir.

### Veri Setinin Son Hali ve Yapısı

Daha önceki aşamalarda elde edilen **HamVeri** tablosu, çeşitli yöntemlerle filtreleme ve gruplama işlemlerinden geçirilip **GenelGrupluVeriSeti** isimli veri seti oluşturulmuştu. Anahtar değerlerin belirlenmesi ve saha içindeki kullanılacak bölge gruplarının belirlenip UygulamaTablosu isimli tablonun güncellenmesi ile kullanılacak veri setinin son hali

oluşturulmuş ve uygulama için kullanılmaya hazır hale getirilmiştir. Tabloda toplamda 17 alan ve 2.574 kayıt oluşmuştur.

**Tablo 21**'de **UygulamaTablosu** isimli tabloda yer alan bütün alanlar listelenmiş, tipleri belirtilmiş, açıklamaları eklenmiş ve bazı örnek değerler koyulmuştur.

**Tablo 21.** UygulamaTablosu tablosunun içinde bulunan alanlar listesi

| Alan Adı | Alan Tipi | Açıklaması | Örnek Değerler |
|---|---|---|---|
| **SiraNo** | Integer | Kayıt sıra numarası | 1, 2, 3, 4 |
| **AmbarId** | Integer | Ambar kod numarası | 518, 851 |
| **Tipi** | Varchar | Sipariş tipi | SHPMNT, XDOCK, SHUTTLE |
| **Marka** | Varchar | Ürün markası | Pantene, Gilette |
| **UrunGrubu** | Varchar | Ürün üretici grubu | Home Care, Hair Care |
| **TipDeger** | Varchar | Ürünün kalite değer tipi | Good, Damaged |
| **FirmaId** | Integer | Ürün tedarikçisinin kodu | 99679, 99681 |
| **KutuTipId** | Integer | Kutu tipi kod numarası | 45, 109, 110 |
| **Gun** | Integer | Bu tipteki kayıtların depoda kaldığı ortalama gün sayısı | 1, 2, 3, 4 |
| **KayitSayisi** | Integer | Bu tipteki kayıtların toplam sayısı | 1, 2, 3, 4 |
| **Varyans** | Float | Gun alanının varyans değeri | 1065.50, 1443.81 |
| **StSapma** | Float | Gun alanının standart sapma değeri | 32.64, 37.99 |
| **EnFazlaGunSayisi** | Integer | Gun alanının maksimum değeri | 50, 173, 254 |
| **EnAzGunSayisi** | Integer | Gün alanının minimum değeri | 0, 1, 2 |
| **Anahtar** | Integer | Kayıt için özel algoritma ile belirlenen anahtar değeri | 1, 2, 3, 4 |
| **KumulatifKayitSayisi** | Integer | Kayıt sayılarının o kayda kadar ki toplamı | 1029840, 1029895 |
| **BolgeGrup** | Varchar | Anahtar değer için belirlenmiş bölge grubu | A, B, D, E |

Bu 17 alanın içerikleri hakkında daha fazla bilgi açısından **Tablo 22** hazırlanmış, bu tabloda bütün alanların veri seti üzerinde farklı içerikli kayıt sayıları, en düşük ve en yüksek değerleri listelenmiştir.

**Tablo 22**. UygulamaTablosu alanlarının veri çeşitliliği

| Alan Adı | Farklı Kayıt Sayısı | Minimum Değer | Maksimum Değer |
|---|---|---|---|
| **SiraNo** | 2.574 | 1 | 2.574 |
| **AmbarId** | 4 | 518 | 851 |
| **Tipi** | 6 | PUTAWAY | XDOCK |
| **Marka** | 103 | 7 OClock | Yes |
| **UrunGrubu** | 11 | Appliances | Skin and Personal Care |
| **TipDeger** | 2 | Damaged | Good |
| **FirmaId** | 91 | 99.676 | 257093 |
| **KutuTipId** | 25 | 1 | 130 |
| **Gun** | 230 | 0 | 460 |
| **KayitSayisi** | 590 | 1 | 54.410 |
| **Varyans** | 1.626 | 0 | 197.820,50 |
| **StSapma** | 1.624 | 0 | 444,77 |
| **EnFazlaGunSayisi** | 475 | 0 | 935 |
| **EnAzGunSayisi** | 180 | 0 | 460 |
| **Anahtar** | 229 | 0 | 460 |
| **KumulatifKayitSayisi** | 2.574 | 2 | 1.029.845 |
| **BolgeGrup** | 4 | A | E |

UygulamaTablosu üzerindeki **AmbarId, Tipi, Marka, UrunGrubu, TipDeger, FirmaId** ve **KutuTipId** alanları veri seti üzerindeki belirleyici alanlardır. **SiraNo, Gun, KayitSayisi, Varyans, StSapma, EnFazlaGunSayisi, EnAzGunSayisi, Anahtar, KumulatifKayitSayisi** ve **BolgeGrup** alanları ise belirleyici alanlara göre değer alan, buradaki verilere göre içeriği değişmesi beklenen alanlardır.

**Bölge Grup Bulma Fonksiyonunun Hazırlanması**

Oluşturulan **UygulamaTablosu** üzerinde verilerin hızlı bir biçimde çekilebilmesi ve ihtiyaç olan alanların sonuç olarak döndürülmesi için bir SQL fonksiyonu hazırlanması uygun görülmüştür.

Fonksiyon daha önceden veri setleri üzerinde yorumlanarak sayısı 7'ye kadar indirilmiş değeri kendisine parametre olarak almakta, sonucunda 9 adet veriyi sonuç olarak döndürmektedir. Fonksiyonun aldığı parametreler **AmbarId, Tipi, Marka, UrunGrubu, TipDeger, FirmaId** ve **KutuTipId** şeklindedir. Buna yanıt olarak ise **SiraNo, Gun, KayitSayisi, Varyans, StSapma, EnFazlaGunSayisi, EnAzGunSayisi, Anahtar, KumulatifKayitSayisi** ve **BolgeGrup** alanlarını döndürmektedir. Çalışmanın ana amacı

doğrultusunda bunlardan sadece **BolgeGrup** alanı yer belirlemesi için yeterli olacaktır. Ancak diğer uygulamaların da oluşturulan veri setini yorumlayabilmesi adına ek bilgiler de yanıt olarak fonksiyondan geriye döndürülmüştür. Fonksiyonun SQL kodu **Şekil 32**'de gösterilmiştir.

```
CREATE FUNCTION [dbo].[UrunBolgeselGrupBelirle]
(
    @AmbarId int,
    @Tipi varchar(20),
    @Marka varchar(50),
    @UrunGrubu varchar(50),
    @TipDeger varchar(25),
    @FirmaId int,
    @KutuTipId int
)
RETURNS TABLE
AS
RETURN
(
    select SiraNo, Gun, KayitSayisi, Varyans, StSapma,
        EnFazlaGunSayisi, EnAzGunSayisi, Anahtar, KumulatifKayitSayisi,
        BolgeGrup = isnull(BolgeGrup, 'C')
    from UygulamaTablosu U
    where isnull(U.AmbarId, 0) = isnull(@AmbarId, 0)
        and isnull(U.Tipi, '') = isnull(@Tipi, '')
        and isnull(U.Marka, '') = isnull(@Marka, '')
        and isnull(U.UrunGrubu, '') = isnull(@UrunGrubu, '')
        and isnull(U.TipDeger, '') = isnull(@TipDeger, '')
        and isnull(U.FirmaId, 0) = isnull(@FirmaId, 0)
        and isnull(U.KutuTipId, 0) = isnull(@KutuTipId, 0)
)
```

**Şekil 32.** UrunBolgeselGrupBelirle isimli fonksiyonun kodları

Fonksiyondan sorgulanan veriler, hazırlanmış veri setimizde mevcutsa, ilgili kayıt bulup dönülecektir. Eğer kayıt bulunamazsa, belirlenemeyen grup için kullanılan C grup değeri yanıt olarak **BolgeGrup** alanında döndürülecektir.

Fonksiyonun çalışan uygulamasına bir adet örnek aşağıda verilmiştir:

**Fonksiyonun çağrılması için örnek**

İlk örnekte fonksiyona rastgele seçilen aşağıdaki değerler gönderilmiştir:

**AmbarId**: 518

**Tipi**: SHUTTLE

**Marka**: Pantene

**UrunGrubu**: Hair Care

**TipDeger**: Good

**FirmaId**: 99679

**KutuTipId**: 44

Fonksiyonun cevap olarak hazırlanan veri setinden 1722 numaralı kaydı bulduğu ve bölge olarak E yanıtını döndürdüğü **Şekil 33**'de görülebilir.



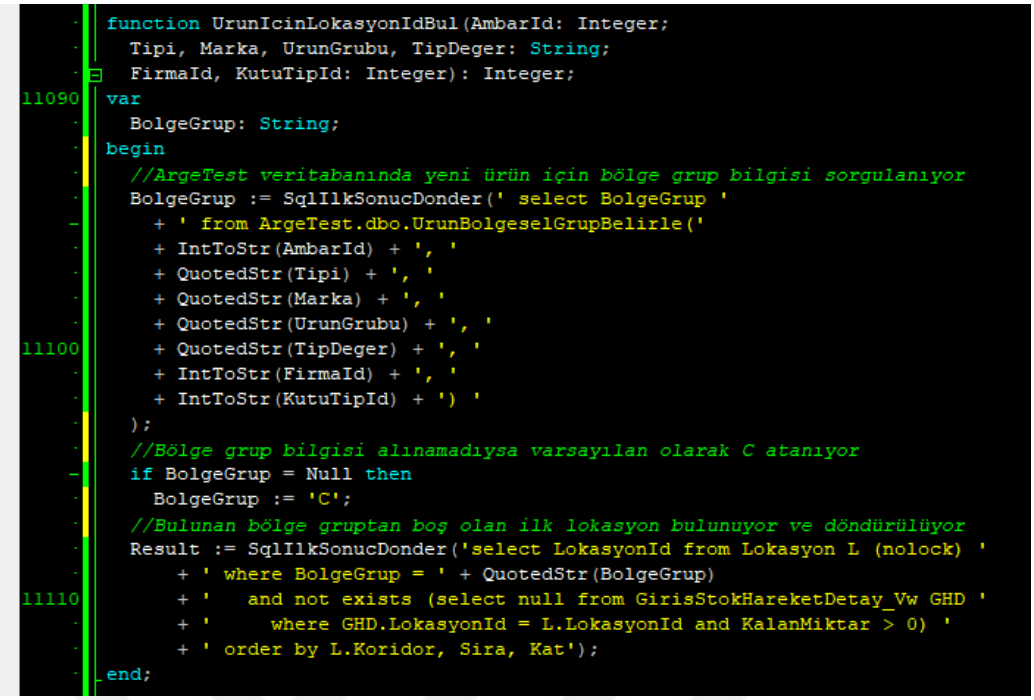

**Şekil 33.** Fonksiyonun çalışma zamanı örneği 1

Verilen kıstaslara göre bugünü kadar depoya 609 girişin yapıldığı, bunların ortalama depoda 14 gün durduğu, en çabuk çıkış yapanın aynı gün, en fazla duran ürünün ise 473 gün depoda geçirdiği fonksiyon tarafından verilen cevaplar arasındadır. Ayrıca ürünün için hesaplanan anahtar değerinin 56 olduğu ve bu ürünün diğer ürünler arasında en çok sirkülasyon hızı hesaplanan ilk 858.730 kayıt arasında bulunduğu da fonksiyonun yanıt olarak döndürdüğü veriler arasındadır.

**Hazırlanan Fonksiyonun Depo Yönetim Yazılımı İçine Yerleştirilmesi**

Hazırlanan veritabanı ve son kullanıma yönelik fonksiyon tamamlandıktan sonra, Barsan Global Lojistik Willebroek tesislerinde kullanılan LDS uygulamasının test sunucusuna yüklenmiştir. Datasetlerin ve fonksiyonun bulunduğu veritabanı LDS'nin test sunucusuna "**ArgeTest**" ismiyle gönderilmiştir.

Bundan sonra saha çalışmalarında program kullanıcıları tarafından test edilmesi için LDS yazılım ekibi ile programın kaynak kodlarına müdahale edilmiştir. **Delphi** programlama dilinde yazılan LDS'nin, el terminallerinde çalışan versiyonunda, depo kabul işlemlerinin yapıldığı bölümdeki kaynak koduna **UrunIcinLokasyonIdBul** isimli yeni bir fonksiyon eklenmiştir. Fonksiyonun Delphi kaynak kodları **Şekil 34**'de mevcuttur.



```
function UrunIcinLokasyonIdBul(AmbarId: Integer;
  Tipi, Marka, UrunGrubu, TipDeger: String;
  FirmaId, KutuTipId: Integer): Integer;
var
  BolgeGrup: String;
begin
  //ArgeTest veritabanında yeni ürün için bölge grup bilgisi sorgulanıyor
  BolgeGrup := SqlIlkSonucDonder(' select BolgeGrup '
    + ' from ArgeTest.dbo.UrunBolgeselGrupBelirle('
    + IntToStr(AmbarId) + ', '
    + QuotedStr(Tipi) + ', '
    + QuotedStr(Marka) + ', '
    + QuotedStr(UrunGrubu) + ', '
    + QuotedStr(TipDeger) + ', '
    + IntToStr(FirmaId) + ', '
    + IntToStr(KutuTipId) + ') '
  );
  //Bölge grup bilgisi alınamadıysa varsayılan olarak C atanıyor
  if BolgeGrup = Null then
    BolgeGrup := 'C';
  //Bulunan bölge gruptan boş olan ilk lokasyon bulunuyor ve döndürülüyor
  Result := SqlIlkSonucDonder('select LokasyonId from Lokasyon L (nolock) '
      + ' where BolgeGrup = ' + QuotedStr(BolgeGrup)
      + '   and not exists (select null from GirisStokHareketDetay_Vw GHD '
      + '     where GHD.LokasyonId = L.LokasyonId and KalanMiktar > 0) '
      + ' order by L.Koridor, Sira, Kat');
end;
```

**Şekil 34.** UrunIcinLokasyonIdBul ismiyle hazırlanan Delphi fonksiyonunun kodları

Bu fonksiyon ürün yerleştirme yapılan ve lokasyon seçimi yapılan bütün ekranlardan çağrılarak kullanılmaya başlanmıştır.

Yine program içinden kullanıcılara önerilen lokasyonları değiştirme imkânı tanınmaktadır. Bunun nedeni, lokasyona varıldığında, ortaya bir veri uyumsuzluğunun çıkması, gidilen lokasyonun fiziki olarak müsait olmaması gibi durumlara karşı önlemdir. Ancak, kullanıcılara ürünü farklı lokasyonlara yerleştirme imkânı tanınırken, farklı bölgedeki bir lokasyona yerleştirme imkânı tanınmamıştır. Uygulama tarafından önerilen lokasyona yerleştirme yapılmasa dahi, yerleştirilen lokasyon uygulamanın belirlediği bölge dahilindeki bir lokasyon olmalıdır. Bunu sağlamak için yine Delphi kaynak kodları içine **UrununLokasyonunBolgeselGrubunuKontrolEt** isimli bir fonksiyon hazırlanmış, bu fonksiyon lokasyon yerleştirme yapılan ekranlardan çağrılmıştır. Fonksiyonun kaynak kodları **Şekil 35**'de gösterilmiştir.

```
function UrununLokasyonunBolgeselGrubunuKontrolEt(LokasyonId, AmbarId: Integer;
  Tipi, Marka, UrunGrubu, TipDeger: String;
  FirmaId, KutuTipId: Integer): Boolean;
var
  LokasyonBolgeGrup: String;
  BolgeGrup: String;
begin
  //Yerleştirme yapılan lokasyonun bölge grubu belirleniyor
  LokasyonBolgeGrup := SqlIlkSonucDonder(' select BolgeGrup '
    + ' from Lokasyon where LokasyonId = ' + IntToStr(LokasyonId));
  if LokasyonBolgeGrup = Null then
    LokasyonBolgeGrup := 'C';

  //Ürün için ArgeTest veritabanında önerilen bölge grubu sorgulanıyor
  BolgeGrup := SqlIlkSonucDonder(' select BolgeGrup '
    + ' from ArgeTest.dbo.UrunBolgeselGrupBelirle('
    + IntToStr(AmbarId) + ', '
    + QuotedStr(Tipi) + ', '
    + QuotedStr(Marka) + ', '
    + QuotedStr(UrunGrubu) + ', '
    + QuotedStr(TipDeger) + ', '
    + IntToStr(FirmaId) + ', '
    + IntToStr(KutuTipId) + ') '
  );
  if BolgeGrup = Null then
    BolgeGrup := 'C';

  //Seçilen lokasyon ile önerilen lokasyonun bölge grupları uyumsuzsa
  //false değeri döndürülüyor
  if (LokasyonBolgeGrup <> BolgeGrup) then
    Result := False;
end;
```

**Şekil 35.** UrununLokasyonunBolgeselGrubunuKontrolEt ismiyle hazırlanan Delphi fonksiyonunun kodları

Delphi kodlarına müdahale edilmesi ile, yeni hazırlanan uygulama, Willebroek tesislerinde önce test ortamında, testlerin tamamlanmasının ardından da canlı ortamda devreye alınmıştır.

# BÖLÜM 5. BULGULAR VE YORUMLAR

Bu bölümde bulgular iki farklı kategoride değerlendirilmiştir. Öncelikle ham veri üzerine, yeni yazılan fonksiyon çalıştırılarak üretilen sonuçlar ham veri ile karşılaştırılmıştır. Burada amaç, yapılan çalışmanın geçmiş veriler üzerindeki sonuçların ölçümlenmesidir. İkinci olarak yapılan uygulamanın canlı sonuçları incelenmeye çalışılmış, mevcut durumun analizi yapılmaya çalışılmıştır. Yorumlar bölümünde ise yapılan çalışmanın sonuçlarının geçmiş çalışmalarla karşılaştırılması ve yorumlanması hedeflenmiştir.

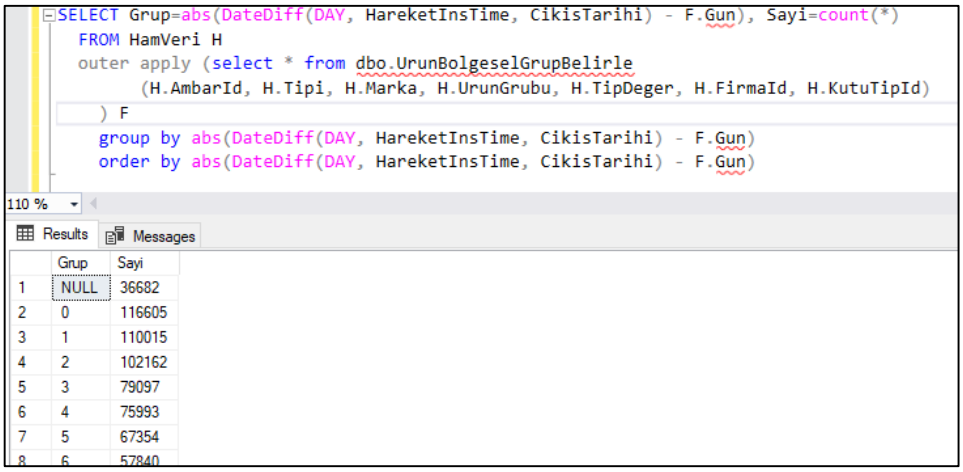
## Bulgular

Bulgular değerlendirilirken öncelikle veri setinde kullanılan ham veri, yeni yapılan uygulama ile teste tabi tutulmuş, sonuçlar karşılaştırmalı olarak ortaya konulmuştur. Daha sonra Willebroek tesislerinde halihazırda çalışan uygulamanın ortaya çıkardığı sonuçlar incelenmiştir.

### Ham Datanın Yeni Uygulama Aracılığıyla Analizi

Araştırmada kullanılmak üzere, Willebroek kayıtlarının tutulduğu veri tabanının 7 Aralık 2018'den 3 Mart 2022'ye kadar olan bütün kayıtları, bir tabloya aktarılmıştı. **HamVeri** olarak isimlendirilen bu tabloda 1.239.559 kayıt bulunmakta. Bu kayıtların yeni hazırlanan fonksiyondan geçirilip ulaşılan sonuçlar detaylı bir şekilde analiz edildi.

Yazılan bir kodla, ham veri üzerinde bulunan verilerin gerçekleşen depoda kalma gün sayıları ile **UrunBolgeselGrupBelirle** tarafından verilen gün sayıları karşılaştırıldı. Ham veri üzerinde çalıştırılan kodun işlenmemiş hali **Şekil 36**'de gösterilmiştir.



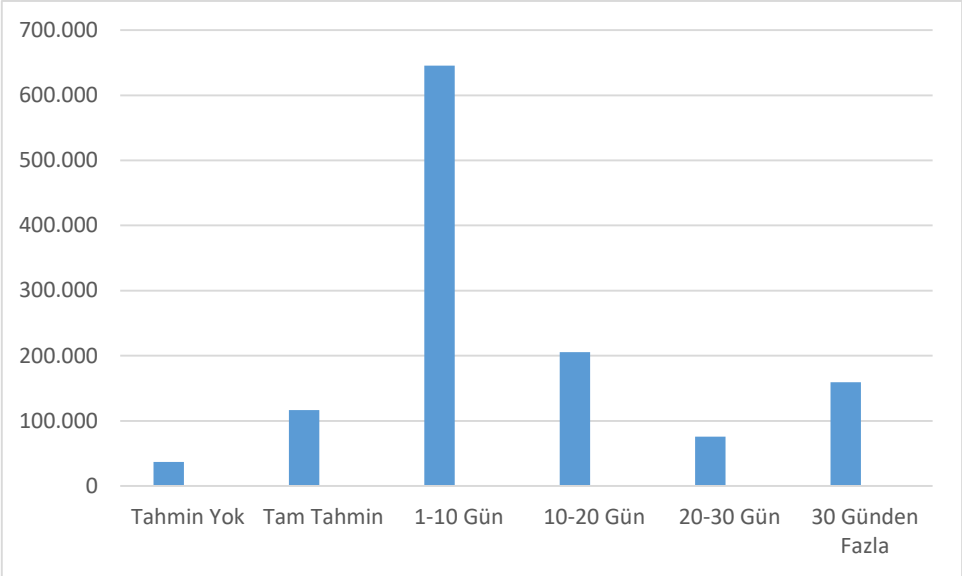

**Şekil 36.** Ham veri üzerinde günlük analiz için çalıştırılan kod

Ham veri üzerinden günlük sapmaları analiz edebilmek için çalıştırılan kod sonucu üretilen 638 satırlık veri daha iyi analiz edebilmek için tekrardan gruplandı. Veri alınamayan kayıtlar "**Tahmin Yok**" adı altında toplanırken, 0 gün sapmalı kayıtlar "**Tam Tahmin**" başlığı altında incelendi. Diğer sapmalar onar günlük gruplar altında toplanırken, 30 gün üstündeki sapmalar ise "**30 Günden Fazla**" başlığı altında bir grup altına toplanarak ele alındı. Elde edilen sayısal veriler **Tablo 23**'de gösterilmiştir.

**Tablo 23.** Ham veri üzerinde yapılan gün tahminlerinin sapma sayıları

| Grup | Sayı | Oran |
|---|---|---|
| **Tahmin Yok** | 36.682 | 2,96 |
| **Tam Tahmin** | 116.605 | 9,41 |
| **1-10 Gün** | 645.519 | 52,08 |
| **10-20 Gün** | 205.730 | 16,60 |
| **20-30 Gün** | 75.780 | 6,11 |
| **30 Günden Fazla** | 159.229 | 12,85 |

Bu analiz esnasında 36.682 kayıt için sonuç üretilemediği görüldü. Bu toplam kayıt sayısının %2,95'e denk gelen bir oran. Kayıtların %9,41'inde tamı tamına gün sayısının doğru tahmin edildiği görüldü. Bu milyonlarca kayıt üzerinden ortalama tahminleriyle yapılan bir çalışmada, yüksek kabul edilebilecek bir oran olarak değerlendirildi. Sapmaların büyük bölümünün 1-10 gün aralığında olduğu, 30 günü geçen sapmaların ise kayıtların sadece %12,85'inde gerçekleştiği görülmüştür. Bu durumun grafik üzerinde gösterimine **Şekil 37**'da yer verilmiştir.




**Şekil 37.** Ham veri üzerindeki tahmin sapmalarının grafiksel gösterimi

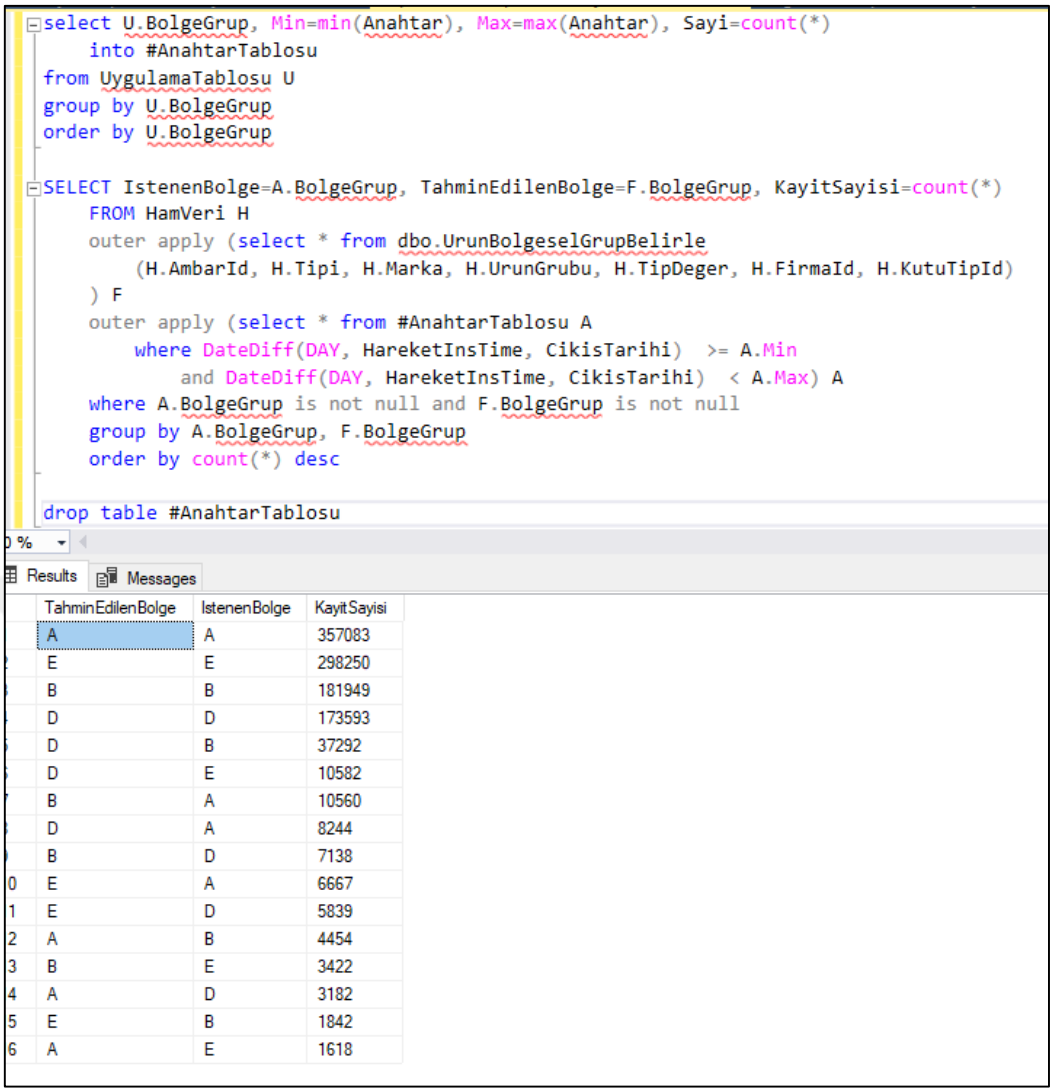
Elde edilen veriler fikir verici olsa bile, çalışmanın temel amacı, ürünün gün bazında tam olarak kaç gün depoda kalacağını hedeflemek değil, ürünleri kaldığı günlere göre bölgesel gruplara dağıtmak olduğundan, günlük sapmalar, bölgesel grup sapmaları altında tekrar incelendi.

Bu inceleme için, ham verinin ortalama günler yerine, uygulamanın kullandığı anahtar değer ölçülerine göre tekrar sınıflandırılması gerekti. Bu amaçla, öncelikle son oluşturulan veri seti içinden, bölgesel grupların anahtar aralıklarının çıkarılmasına ihtiyaç duyuldu. UygulamaTablosu tüm bölge gruplarının sahip olduğu en küçük anahtarlar, en büyük anahtarlar ve barındırdıkları kayıt sayıları **Tablo 24**'de gösterilmiştir.

**Tablo 24.** İşlenmiş uygulama verisi üzerinde bölge gruplarının anahtar aralıkları

| Bölge Grubu | Min. Anahtar | Max. Anahtar | Kayıt Sayısı |
|---|---|---|---|
| **A** | 0 | 6 | 901 |
| **B** | 7 | 10 | 222 |
| **D** | 11 | 18 | 315 |
| **E** | 19 | 460 | 1.136 |

Oluşturulan bu tabloya göre, ham veri seti üzerinde ürünlerin gerçekleşen dataya göre yerleştirmesi gereken gruplar belirlenmiş ve bu gruplar, bu çalışmada hazırlanan fonksiyonun ürettiği sonuçlar ile karşılaştırılmıştır. Hazırlanan kod **Şekil 38**'de gösterilmiştir.



```
select U.BolgeGrup, Min=min(Anahtar), Max=max(Anahtar), Sayi=count(*)
    into #AnahtarTablosu
from UygulamaTablosu U
group by U.BolgeGrup
order by U.BolgeGrup

SELECT IstenenBolge=A.BolgeGrup, TahminEdilenBolge=F.BolgeGrup, KayitSayisi=count(*)
    FROM HamVeri H
    outer apply (select * from dbo.UrunBolgeselGrupBelirle
        (H.AmbarId, H.Tipi, H.Marka, H.UrunGrubu, H.TipDeger, H.FirmaId, H.KutuTipId)
    ) F
    outer apply (select * from #AnahtarTablosu A
        where DateDiff(DAY, HareketInsTime, CikisTarihi)  >= A.Min
            and DateDiff(DAY, HareketInsTime, CikisTarihi)  < A.Max) A
    where A.BolgeGrup is not null and F.BolgeGrup is not null
    group by A.BolgeGrup, F.BolgeGrup
    order by count(*) desc

drop table #AnahtarTablosu
```

Results | Messages

| | TahminEdilenBolge | IstenenBolge | KayitSayisi |
|---|---|---|---|
| | A | A | 357083 |
| | E | E | 298250 |
| | B | B | 181949 |
| | D | D | 173593 |
| | D | B | 37292 |
| | D | E | 10582 |
| | B | A | 10560 |
| | D | A | 8244 |
| | B | D | 7138 |
| 0 | E | A | 6667 |
| 1 | E | D | 5839 |
| 2 | A | B | 4454 |
| 3 | B | E | 3422 |
| 4 | A | D | 3182 |
| 5 | E | B | 1842 |
| 6 | A | E | 1618 |

**Şekil 38.** Anahtar bazında ham veri ile yapılan karşılaştırmalı analizin kodları

Alınan sonuçların sayısal karşılıkları **Tablo 25**'de gösterilmiştir.

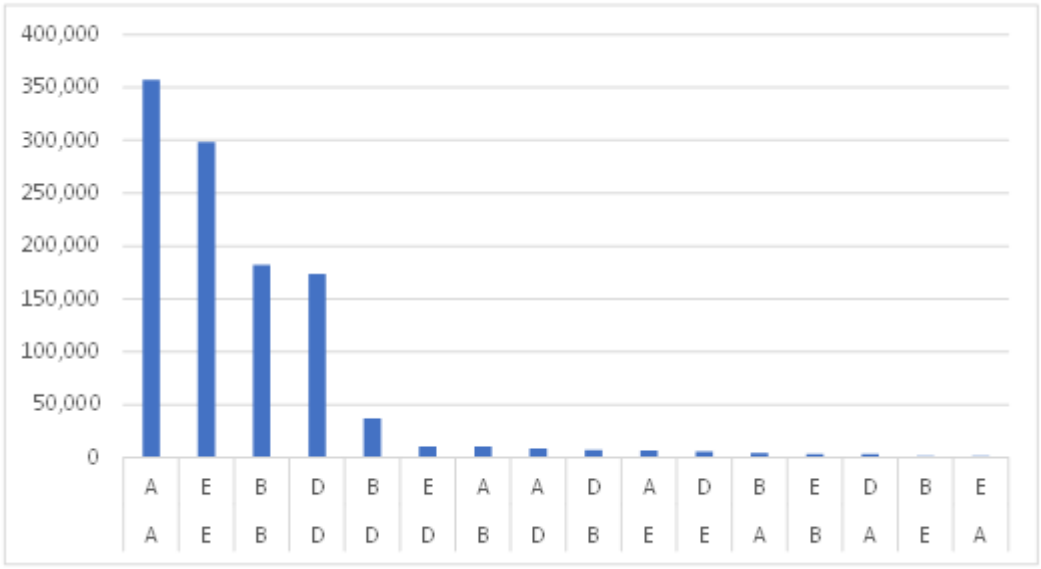
**Tablo 25.** Ham veri üzerindeki kayıtlara göre yerleştirilmesi gereken bölgeler ve uygulamanın tahmin ettiği bölgeler

| Olması Gereken Bölge | Tahmin Edilen Bölge | Kayıt Sayısı | Oran |
|---|---|---|---|
| A | A | 357.083 | 32,12 |
| E | E | 298.250 | 26,83 |
| B | B | 181.949 | 16,37 |
| D | D | 173.593 | 15,61 |
| D | B | 37.292 | 3,35 |
| D | E | 10.582 | 0,95 |
| B | A | 10.560 | 0,95 |
| D | A | 8.244 | 0,74 |
| B | D | 7.138 | 0,64 |
| E | A | 6.667 | 0,60 |
| E | D | 5.839 | 0,53 |
| A | B | 4.454 | 0,40 |
| B | E | 3.422 | 0,31 |
| A | D | 3.182 | 0,29 |
| E | B | 1.842 | 0,17 |
| A | E | 1.618 | 0,15 |

Tabloda "**Olması Gereken Bölge**" sütununda gösterilen bölge numarası, ham veriye göre ürünün depoda durduğu gün sayısının yeni oluşturulan anahtar tablosuna göre isabet ettiği bölgesel grup kodunu göstermektedir. "**Tahmin Edilen Bölge**" sütunundaki bölge kodu ise, aynı ürün için o an çalıştırılan **UrunBolgeselGrupBelirle** fonksiyonu ile ürünün yerleştirilmesi istenen bölgeyi ifade etmektedir.

Ham verilerde karşılaştırmaya giren kayıt sayısının toplam adedi 1.239.559'dur. Bunlardan 1.111.715 adedi tahminde bulunabilmiştir. Kalan yaklaşık 130 bin kayıt içinse program tahmin yürütememiş, bu yüzden bölge grubu olarak 'C' seçeneğini önermiştir. Buna göre programın tahmin yürütme oranı yaklaşık %90 olarak gerçekleşmiştir.

Tabloda da görülebileceği gibi program kayıtların yaklaşık %91'ine birebir yerleşmesi gereken bölgesel grubu önermiştir. Bu 800 binin üzerinde kayıt anlamına gelmektedir. Kayıtların %7'sinde ise bir bölge sapma gerçekleşmiştir. Geriye kalan %2'lik oranda ise program, kayıt için yerleşim yapılması gereken bölgeyi veya bir yanındaki bölgeyi önerememiştir. Yerleşim önerilerini tablosunun grafik üzerinde gösterimi **Şekil 39**'de bulunmaktadır.




**Şekil 39.** Ham veri üzerindeki kayıtlara göre yerleştirilmesi gereken bölgeler ve uygulamanın tahmin ettiği bölgeler tablosunun grafik gösterimi

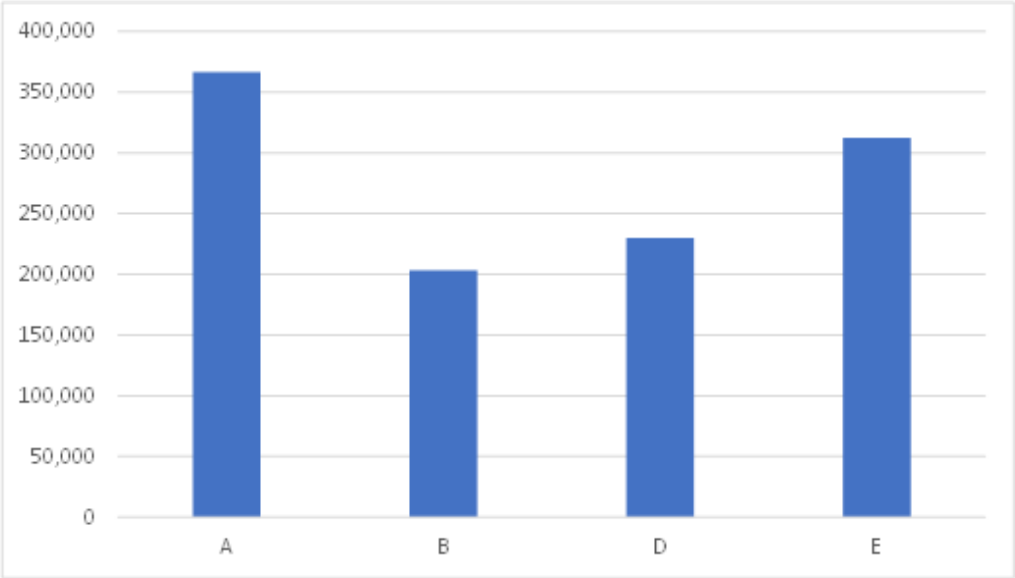
Depo yerleşim önerilerindeki bir diğer kritik konu da gelen ürünlerin eşit sayıdaki bölge grupları arasında birbirlerine yakın sayılarda dağılımıdır. Bölgeler arasındaki kayıt sayıları arasında oluşacak uçurumlar, belli bölgelerin kapasitesinin üzerinde doluluk oranına ulaşmasına, belli bölgelerin ise boş kalarak verimsiz kullanılmasına neden olacaktır.

Programın önerilerine göre bölgelere düşen ürün kayıt sayıları **Tablo 26**'de gösterilmiştir.

**Tablo 26.** Programın bölgelere göre önerdiği kayıt sayıları tablosu

| Tahmin Edilen Bölge | Kayıt Sayısı |
|---|---|
| **A** | 366.337 |
| **B** | 203.069 |
| **D** | 229.711 |
| **E** | 312.598 |

Dağılımın daha rahat anlaşılabilmesi için görselleştirildiği grafik **Şekil 40**'da gösterilmiştir.




**Şekil 40.** Program tarafından önerilen dağılım sayılarının grafik gösterimi

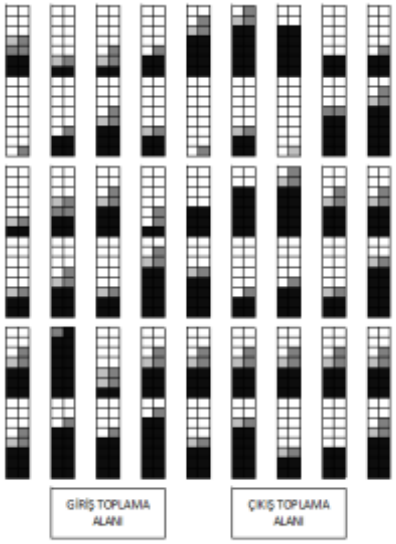
Tablo ve grafikte de görülebileceği gibi, program tarafından 4 bölge arasında önerilen sayılarda çok ciddi farklılık bulunmamıştır. Bu da oluşturulan veri seti üzerine kurgulanan algoritmanın envanter yönetimi üzerinde ciddi bir olumsuzluk yaratma durumuna yönelik kullanılabilecek bir veridir.

**Yeni Uygulamanın Saha Sonuçlarının Analizi**

Willebroek tesislerinde 15 Mart 2022 tarihine kadar, **ürün gruplarına göre ürünlerin bir araya yerleştirilmesi sistemi** kullanılmaktaydı. Bu çalışma kapsamındaki veri setlerinde de kullanılan ve müşteri tarafından bildirilen **UrunGrubu** alanına göre ürünlere depoda yer tahsis edilmesi ve o ürün grubundan her ürün geldiğinde o tahsis edilen alanda bir lokasyona rastgele yerleştirilmesi üzerine kurulu bir sistemdi.

**Yeni Uygulamanın Envanter Yönetime Üzerine Etkilerinin Analizi**

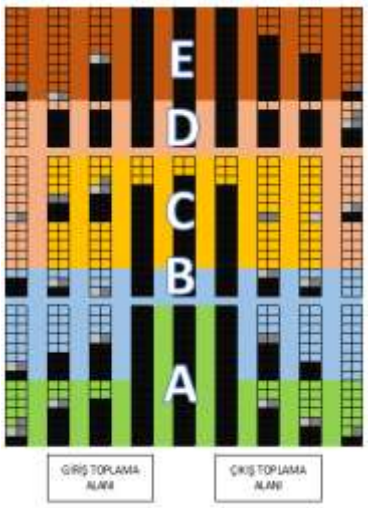
**Ürün gruplarına göre ürünlerin bir araya yerleştirilmesi sistemi**, ürün gruplarının benzer şekilde hareket edeceği varsayımına dayanmaktaydı ve ayrıca ürünlerin bulunmasını kolaylaştıran bir sistemdi. Ancak ürün gruplarına yer tahsis edilmesi, depo içerisinde envanter yönetimi anlamında olumsuz etkilere sebep olmakta, belli ürün grupları için ürün gelse de gelmese de belli bir alan ayrılmak zorunda kalınmaktaydı. 15 Mart 2022 tarihindeki Willebroek tesisindeki lokasyon doluluk şeması **Şekil 41**'da gösterilmiştir.




**Şekil 41.** 15 Mart 2022'de Willebroek tesisleri lokasyon doluluk şeması

Çalışma kapsamında hazırlanan "Depolarda Sipariş Toplama İşlemlerine Yönelik Akıllı Lokasyonlama Uygulaması" (DSTİYALU) fonksiyonlarının Willebroek tesislerinde kullanılan LDS sistemine entegrasyonun canlıya geçişi 15 Nisan 2022 itibariyle tamamlanmış ve tesiste canlıya alınmıştır.

Tesiste ilk göze çarpan sonuç, 30 günlük bir sürede bile tesis içerisindeki lokasyonlarına ara bölgelerinde oluşan boşlukların azalması olmuştur. Boşluklar, ürünlerin zamanla yerleri değişmeye başladıkça, bölgesel olarak birleşmeye ve 5 ana bölgenin ön sıralarında toplanmaya başlamıştır. 15 Nisan 2022 itibariyle Willebroek tesislerindeki lokasyon doluluk şeması **Şekil 42**'de görülebilir.




**Şekil 42.** 15 Nisan 2022'de Willebroek tesisleri lokasyon doluluk şeması

Şemalarda da görülebileceği gibi, geçen 30 günlük zaman diliminde ürünler 5 bölgenin merkez noktalarında toplanmış, deponun çeşitli yerlerinde küçük küçük miktarlardaki boş alanların sayısı azalmıştır. Bu da envanter yönetimini, ürün toplamayı ve deponun verimli kullanılmasını arttırabilecek bir faktördür.

Görsel durumu verilen depodaki ısı haritasının, bazı sayısal verileri de **Tablo 27**'da sunulmuştur.

**Tablo 27.** Depodaki lokasyonların doluluk bilgileriyle ilgili bazı sayısal veriler

| **Lokasyon Durumu** | **15.Mar.22** | **15.Nis.22** |
|---|---|---|
| Depoda Bulunan Lokasyon Sayısı | 4.050 | 4.050 |
| Ürün Bulunan Lokasyon Sayısı | 2.190 | 2.255 |
| Dolu Lokasyon Oran | 54,07 | 55,68 |
| Ürün Bulunan Lokasyon Grubu Sayısı | 54 | 51 |
| Boşluk Grubu Sayısı | 54 | 33 |
| Boşluk Grubunun Dolu Grubuna Oranı | 100,00 | 64,71 |
| %5'i Boş Lokasyon Grubu Sayısı | 3 | 5 |
| %5'i Boş Lokasyonların Grup Sayısına Oranı | 5,56 | 9,80 |
| %95'i Dolu Lokasyon Grubu Sayısı | 3 | 13 |
| %95'i Dolu Lokasyonların Grup Sayısına Oranı | 5,56 | 25,49 |

Sayısal değerlerde de ilk göze çarpan durum, deponun doluluk oranında 1 ay içerisinde önemli bir değişiklik olmamasına rağmen (%54,07'den, %55,68'e yükselmiştir) depodaki boşluk grubu sayısının 54'ten 33'e inmesidir. Bu da ürünlerin daha çok bir araya gelmeye başladığı şeklinde yorumlanabilir. Yine 1 ay öncesinde oluşan boşluk grubu sayısı, ürünlerin yan yana geldiği grupların sayısıyla eşitken, 1 ay sonra boşluk gruplarının sayısı diğerinin %64'ü seviyesine gerilemiştir. Yine %95'i dolu olan lokasyon gruplarının, toplam dolu lokasyon gruplarına oranı %5,56'dan, %25,49'a yükselmiştir. Bu veri de dolu lokasyonların bir araya toplanmaya başladığının işaretidir. 1 ay içerisinde kendiliğinden ortaya çıkan bu boşluk grubu azalmasının ilerleyen zamanlarda depo içinde oluşan boş alanların giderek azalması sonucunu ortaya çıkarması beklenebilir. Bu da depo içinde gereksiz yer kullanımı ve envanter yönetimi üzerinde olumlu etkiler oluşturabilecektir. Ayrıca bölgelerde ortaya çıkan "merkeze toplanma" eğilimi gereksiz mesafe kat etmelerin önüne geçebilecek, depo içinde ürün yerleştirme ve toplama işlemlerinde mesafe ve zaman kazanımı üzerinde olumlu bir gelişmeye vesile olabilecektir.

**Yeni Uygulamanın Mesafeler Üzerinde Etkilerinin Analizi**

Depolarda envanter yönetimi önemli olmakla birlikte, bu çalışmanın ana amacı olan akıllı lokasyonlama konusundaki sonuçlar da incelenmiştir. Geçen 1 aylık süre içindeki kazanılan mesafe, zaman ve yakıt gibi kalemler ilerleyen zamanlarda daha detaylı olarak hesaplamalara tabi tutulacaktır. Ancak ilk bakışta 15 Mart 2022 ve 15 Nisan 2022 arasındaki durumu analiz etmek için örnek bir veri kümesi seçimiyle bir çalışma yapılmıştır. Bu iki tarih arasındaki 1 aylık dönemde depoya toplam girişi yapılan kutu sayısı 41.893 olarak gerçekleşmiştir. Sağlıklı bir analiz için tesiste eskiden kullanılan **ürün gruplarına göre ürünlerin bir araya yerleştirilmesi sistemi** yöntemine göre 36. sıra ve ilerisine yerleşecek 8.656 kutu örnek küme olarak seçilmiştir. Bu kümeye X kümesi adı verilmiştir. X kümesinin elemanlarının yerleşim bölgesi ve çıkış durumu incelenip **Tablo 28**'de gösterilmiştir.

**Tablo 28.** Örnek olarak seçilen X kümesinin 1 aylık durumu

| | |
|---|---|
| **X Kümesi Eleman Sayısı** | 8.656 |
| **X Kümesinden Çıkış Yapılan Kutu Sayısı** | 2.311 |
| **A Bölgesine Yerleşen Eleman Sayısı** | 1.422 |
| **A Bölgesinden Çıkış Yapılan Eleman Sayısı** | 1.203 |

Örnek olarak X kümesi seçilirken, bu dönem içinde giren kutulardan, tesiste 15 Mart 2022 öncesi kullanılan **ürün gruplarına göre ürünlerin bir araya yerleştirilmesi sistemi** yöntemine göre 36. sıra ve ilerisine yerleşecek kutuların seçilmesinin ana nedeni bu sıra numarasının, yeni sistemde "E Bölgesi" olarak ifade edilen, toplama alanlarına en uzak bölümü ifade etmesidir. E bölgesindeki lokasyonlarına, yani 36. sıra ve ilerisinin toplama alanlarına mesafesi yaklaşık 300 metredir. A bölgesi olarak ifade edilen 15. sıra ve gerisinin ise toplama alanlarına mesafesi ortalama 100 metre civarındadır.

Tabloda görüldüğü gibi X kümesinin 8.656 elemanından 1.422 tanesi, eskiden E bölgesine yerleşecek iken, yeni sistemle birlikte A bölgesine yerleşmiştir. X kümesinin elemanlarından 1 aylık dönem içinde çıkışı yapılan 2.311 kutunun yarıdan fazlasının A bölgesine yerleşen kutular arasından olması, bu yerleşimin önemini göstermektedir. Eğer bu kutular E bölgesine yerleşseydi, yerleşim ve toplama için kutu başına ortalama 600 metre mesafe kat edilecekti. Ancak A bölgesine yerleştirilerek kutu başına gidiş gelişte ortalama 200 metre mesafe kat edilmesi sağlandı. 1.203 kutuda toplam tasarruf edilen mesafe miktarı ise tam olarak 481 kilometre olarak gerçekleşti. Sadece X kümesindeki

A'ya taşınan 1.203 kutu için tasarruf edilen 481 kilometrelik mesafe, personele zaman, işletmeye de yakıt tasarrufu olarak yansıdı.

Sağlıklı bir analiz için tersi durumun da dikkate alınması gerektiği düşünülerek yeni bir örnek küme seçildi. Bu kez 1 aylık dönemde depoya girişi yapılan 41.893 kutudan, 15 Mart 2022 öncesi kullanılan sisteme göre 15. sıra ve gerisine yerleştirilmesi gereken 7.815 kutu Y kümesi olarak seçilmiştir. 15. sıra ve daha küçük numaralı sıralar yeni sistemde "A bölgesi" olarak isimlendirilen bölgedir. Bu bölge giriş ve çıkış toplama alanlarına tesisin en yakın bölgesidir ve bu alanlara ortalama uzaklığı 100 metre mesafededir. Bu alanın örnek küme olarak seçilmesinin sebebi, bu alana yerleştirilmesi gerekirken yerleştirilmemiş çıkışı da bu 1 aylık dönem içinde yapılmış kutular için fazladan mesafe kat edilmiş olmasıdır. Y kümesinin durumu **Tablo 29**'de gösterilmiştir.

**Tablo 29.** Örnek olarak seçilen Y kümesinin 1 aylık durumu

| | |
|---|---|
| **Y Kümesi Eleman Sayısı** | 7.815 |
| **Y Kümesinden Çıkış Yapılan Kutu Sayısı** | 2.203 |
| **E Bölgesine Yerleşen Eleman Sayısı** | 2.744 |
| **E Bölgesinden Çıkış Yapılan Eleman Sayısı** | 121 |

Tabloda görüldüğü gibi, eski sisteme göre A bölgesine yerleştirilecek kutulardan 2.744 tanesi yeni sistem tarafından E bölgesine yerleştirilmiştir. Bu kutulardan 121 tanesi bu 1 aylık dönem içinde depodan çıkmıştır. Bu çıkış sayısının bu küme içindeki toplam çıkış sayısına oranı ise %5'tir. Bu 121 kutu için kutu başına giriş çıkış işleminde fazladan 400 metre kat edildiği göz önüne alındığında toplam maliyet fazladan harcanan 48 kilometredir.

Tam tersi örnekte tasarruf edilen 481 kilometrenin %10'u kadar fazladan yol kat edilmiştir. X kümesi örneğinde tasarruf edilen mesafenin %90'ı muhafaza edilmiştir. Bunun yanında eski sisteme göre A bölgesine yerleşecek olan ama yeni sistemin E'ye yerleştirdiği 2.744 kutudan 2.621 tanesi depoda kalmaya devam etmiştir. Eski sisteme göre bu 2.621 kutu, deponun toplama alanlarına en yakın bölgesini 1 ay boyunca boşu boşuna işgal edecekken, yeni sistem bu 2.621 kutuyu E bölgesine yerleştirerek bu gereksiz işgali önlemiştir.

**Yorumlar**

İncelenen 1 aylık veri içinde birbirine en uzak iki bölge olan A ve E bölgeleri ile ilgili iki ayrı örnek küme alınıp sonuçlar ortaya koyulduğunda, yapılan çalışmanın Willebroek tesislerindeki depo ekibine envanter yönetiminde başarı ve maliyetlerden tasarruf sağlayacağı ortaya çıkmaktadır. Bölgeler arasındaki geçişler daha detaylandırıldığında veya daha fazla veri toplandığında da benzer bir tablo ile karşılaşmak şaşırtıcı olmayacaktır.

Bu sistemin tesiste eskiden kullanılan yönteme göre ciddi avantajlar sağladığı ortada bulunmakla birlikte, bu bölümde eski yöntemle değil, daha önce yapılan araştırmalarda kullanılan bazı yöntemlerle karşılaştırması yapılacaktır.

Karşılaştırılan yöntemler arasında makine öğrenmesi algoritmaları, ABC analizi yöntemi gibi genel yöntemlerle birlikte, bu çalışmanın literatür taramasında geçen sipariş toplama veya mal yerleştirmeye özelleştirilmiş yöntemler de bulunmaktadır.

**Makine Öğrenmesi Algoritmaları**

Makine öğrenmesi kavramı, günümüzün petrolü olarak ifade edilmektedir. Her geçen gün farklı veri kaynaklarından gelen verilerin anlamlandırılmasının önemi artmaktadır. Ayrıca gelen verilerin özelliklerine göre oluşturulacak modeller ile birçok problem çözülebilmektedir. Bu bağlamda elde edilen verilerinden anlamlı sonuçlar çıkartmak ya da modeller oluşturmak için Makine öğrenmesi algoritmaları kullanılmaktadır. Her sektör elindeki büyük veriyi kullanarak öğrenme temelli daha zeki sistemler, mimariler kurmak durumundadır (Günerkan, Şahinaslan ve Şahinaslan, 2022).

Çalışma, içerisinde yüksek sayılı bir ham verinin bulunması, bu verinin yorumlanmış olması, temel bileşenlerine ayrılmış olması ve özünde bölge gruplandırma üzerine bir tahmin yapılıyor olması ile, makine öğrenmesi yöntemi için potansiyel bir uygulama alanı olarak görülebilir.

Bu nedenle başlı başına yeni bir çalışma konusu olan, **makine öğrenmesi yöntemlerinin bölge grup tahminlemede kullanılması** ile ilgili de ön bir araştırma yapılmıştır. Makine öğrenmesi yöntemlerinin bu veri seti üzerinde vereceği sonuçlar, belirlenen temel bileşenlere dayanılarak yeni gelecek verilerde makine öğrenme algoritmalarının bölge tahminindeki başarı düzeyleri, ölçümlenerek bu veri seti üzerinde kullanılması hakkında bir araştırma yapılmıştır.

Çalışmanın temel yapısı itibariyle, ham verilerin yorumlanıp bölge grup sınıflandırması üzerine bir tahminleme yapılması yönüyle, makine öğrenme yöntemleri içindeki **sınıflandırma algoritmaları** üzerinde durulmuştur. Araştırma yapılan algoritmalar aşağıda açıklanmıştır.

**Naive Bayes Sınıflandırması**

Bu algoritma 17. Yüzyılda yaşayan matematikçi Thomas Bayes'in geliştirdiği Bayes teoremini kullanır. Bu teorimin denklemi **Denklem 2**'de gösterilmiştir.

**Denklem 2.** Bayes teoreminin denklemi

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Formülde P(A|B), B olayı gerçekleştiği durumda A olayının meydana gelme olasılığıdır. P(B|A), A olayı gerçekleştiği durumda B olayının meydana gelme olasılığıdır. P(A) ve P(B), A ve B olaylarının önsel olasılıklarıdır. Bu algoritmaya sunulan ham verilerin sınıfı, bir dizi olasılık hesaplamasının ardından, belirlenmeye çalışılır (Theorem, 2003).

**Karar Ağacı (Decision Tree) Algoritması**

Karar ağacı algoritması, veriyi düğümlere ayırıp her düğümün birbiriyle karşılaştırılıp bir teste tabi tutulup sonuç üretilerek düğümleri birleştirerek tek bir karar noktasına varma üzerine kurulu bir algoritmadır (Kamiński, Jakubczyk ve Szufel, 2018).

**En Yakın Komşuluk (k Nearest Neighbours) Algoritması**

En yakın komşuluk (k) algoritması, 1967 yılında Cover ve Hart tarafından ortaya atılan, veri setine her yeni katılan verinin, eldeki verilere olan uzaklığının hesaplanıp bir sınıflandırmaya tabi tutulduğu bir algoritmadır (Triguero, García-Gil, Maillo, Luengo ve Salvador García, 2019).

**Rastgele Orman (Random Forest) Algoritması**

Rastgele ormanlar veya rastgele karar ormanları, sınıflandırma, regresyon ve diğer görevler için, eğitim aşamasında çok sayıda karar ağacı oluşturarak problemin tipine göre sınıf (sınıflandırma) veya sayı (regresyon) tahmini yapan bir toplu öğrenme yöntemidir (Ho, 1995).

**Ridge Sınıflandırma Algoritması**

Ridge sınıflandırma algoritması, belirli bir sınıfın örneklerinin doğrusal bir alt uzayda bulunduğunu ve bir kategoriye yeni bir test örneğinin ilgili sınıfın eğitim örneklerinin

doğrusal bir kombinasyonu olarak tanımlanacağını belirten alt uzay varsayımına dayanır (Behera ve Kumaravelan, 2020).

**Torbalama (Bagging) Yöntemi**

Torbalama (bootstrap aggregating — önyükleme toplaması) yöntemi 1996 yılında Breiman tarafından geliştirilmiştir. Orijinal veri setinden elde edilen önyükleme örneklerine tahminciler uygulanarak bir topluluk oluşturulur. Burada önyükleme uygulaması, iadeli rastgele seçim yapıp alt örneklemler oluşturmak için kullanılır (Ay, 2019).

**Yükseltme (Boosting) Yöntemi**

Torbalama Yükseltme yöntemindeki temel fikir, veri setine farklı ağırlıklar verilmesi sonucu elde edilen ağaçlar topluluğundan çıkarsamalar yapılmasıdır. Başlangıçta tüm gözlemler eşit ağırlığa sahiptir. Ağaç topluluğu büyümeye başladıkça, problem bilgisine kurulu olarak ağırlıklandırmalar düzenlenir. Yanlış sınıflandırılan gözlemlerin ağırlığı arttırılırken, nadiren yanlış sınıflandırılan gözlemlerin ağırlığı azaltılır. Bu sayede ağaçlar zor durumlar karşısında kendini düzenleyebilme yeteneği kazanır (Ay, 2019).

**Çok Katmanlı Algılayıcılar (Multi-Layer Perceptron)**

MLP ağlarında nöronlar katmanlar şeklinde organize edilmiştir. MLP'de ilk katman girdi katmanıdır. Girdi katmanı, çözülmesi istenilen probleme ilişkin bilgilerin YSA(yapay sinir ağları)'ya alınmasını sağlar. Diğer katman ise ağ içerisinde işlenen bilginin dışarıya iletildiği çıktı katmanıdır. Girdi ve çıktı katmanlarının arasında yer alan katmana ise gizli katman adı verilir. MLP ağlarında birden fazla gizli katman da bulunabilir (Kaynar, Taştan ve Demirkoparan, 2011). Bir diğer öğrenme yöntemi ise makine öğrenimidir. Makine öğrenmesi derin öğrenme algoritmaları kullanılarak insan zekâsını taklit eden bir bilgisayar teknolojisidir (Şahinaslan, Dalyan ve Şahinaslan, 2022).

**Makine Öğrenmesi Yöntemleri İçin Yeni Bir Veri Seti Hazırlanması**

Makine öğrenmesi yöntemlerinin doğru analizi için ham veri seti üzerinden gidilmesi ihtiyaç duyulmuştur. Ancak bu çalışmada kullanılan ham veri seti üzerinde ulaşılmaya çalışılan çıktı olan Bölge Grubu bilgisi bulunmaktadır. Bu yüzden çalışma içinde oluşturulan yorumlanmış veri setimize, bir algoritma yardımıyla Bölge Grubu bilgisi eklenmiş ve **MLVeriSeti** ismiyle yeni bir tablo oluşturulmuştur. Bu işlemin SQL kodu **Şekil 43**'de gösterilmiştir.

```sql
drop table MLVeriSeti
select U.BolgeGrup, Min=min(Anahtar), Max=max(Anahtar), Sayi=count(*)
    into #AnahtarTablosu
from UygulamaTablosu U
group by U.BolgeGrup
order by U.BolgeGrup

SELECT H.*, Bolge=A.BolgeGrup
    into MLVeriSeti
    FROM YorumlanmisVeriSeti H
    outer apply (select * from dbo.UrunBolgeselGrupBelirle
        (H.AmbarId, H.Tipi, H.Marka, H.UrunGrubu, H.TipDeger, H.FirmaId, H.KutuTipId)
    ) F
    outer apply (select * from #AnahtarTablosu A
        where H.Gun >= A.Min
            and H.Gun < A.Max) A
    where A.BolgeGrup is not null and F.BolgeGrup is not null

drop table #AnahtarTablosu
```

**Şekil 43.** MLVeriSeti isimli tablonun hazırlanması için gerekli SQL kodu

İncelenen makine öğrenmesi yöntemleri için yapılacak analizlerde bu yeni oluşturulan tablo kullanılacaktır.

**Karşılaştırma Uygulamasının Hazırlanması**

Oluşturulan veri setine incelenen uygulamalarının uygulanması ve karşılaştırılması için **Python** programlama dili için üretilen **Sckit Learn** kütüphanesi kullanılmıştır. Bu kütüphane içinde bu algoritmaların da dahil olduğu birçok algoritmayı teste tabi tutmak ve test sonuçlarını elde etmek mümkündür (Varoquaux ve Müller, 2022).

Veri setini incelenen 8 farklı algoritmada teste tabi tutmak için Python dilinde bir program yazılmıştır. Bu programın Python kaynak kodlarından bir örnek **Şekil 44**'de gösterilmiştir.

```
c = connection.cursor()
c.execute('''
          SELECT isnull(AmbarId, 0), Tipi=isnull(Tipi,''), Marka=isnull(Marka,''), UrunGrubu=isnull(UrunGrubu,''),
              TipDeger, FirmaId=isnull(FirmaId,0), KutuTipId=isnull(KutuTipId,0), Bolge
          FROM MLVeriSeti
          ''')

dataframe = pd.DataFrame(c.fetchall(), columns = ['AmbarId', 'Tipi', 'Marka', 'UrunGrubu', 'TipDeger',
                                                  'FirmaId', 'KutuTipId', 'Bolge'])

le = LabelEncoder()
dataframe.Tipi=le.fit_transform(dataframe.Tipi)
dataframe.Marka=le.fit_transform(dataframe.Marka)
dataframe.UrunGrubu=le.fit_transform(dataframe.UrunGrubu)
dataframe.TipDeger=le.fit_transform(dataframe.TipDeger)

array=dataframe.values
X = array[:,0:7]
Y = array[:,7]

models = []
models.append(('NB', GaussianNB()))
models.append(('CART', DecisionTreeClassifier()))
models.append(('KNN', KNeighborsClassifier()))
models.append(('RFC', RandomForestClassifier()))
models.append(('RC', RidgeClassifier()))
models.append(('BC', BaggingClassifier()))
models.append(('ABC', AdaBoostClassifier()))
models.append(('MLPC', MLPClassifier()))
```

**Şekil 44.** ML yöntemleri karşılaştırma programından bir görüntü

Uygulamaya çalışmada kullanılan temel alanlar olan **AmbarId, Tipi, Marka, UrunGrubu, TipDeger, FirmaId, KutuTipiId** birer özellik (feature) olarak verilmiş ve **Bolge** alanı sonuç olarak istenmiştir. Teste tabi tutulan veri setinde 1 milyon üzerinde kayıt bulunmasına rağmen, analizdeki doğruluk oranı arttırılması için örnekleme boyutu olarak 0.5, yani eldeki kayıt sayısının yarısı kullanılmıştır. Testi gerçekleştiren kod **Şekil 45**'da verilmiştir.

```
test_size = 0.5
seed = 7
results = []
names = []
for name, model in models:
    X_train, X_test, Y_train, Y_test = train_test_s
    model.fit(X_train, Y_train)
    result = model.score(X_test, Y_test)

    results.append(result)
    names.append(name)
    msg = "%s: %f" % (name, result)
    print(msg)

c.close()
```

**Şekil 45.** Testi gerçekleştiren ve sonuçları ekrana yazdıran Python kodu

Kodların hazırlanmasının ardından uygulama çalıştırılmış, uygulamanın çalışma esnasında ürettiği çıktılardan bir örnek **Şekil 46**'de gösterilmiştir.

```
In [9]: runfile('C:/Users/Ceyhunk/OneDrive - BARS
GLOBAL LOJISTIK A.S/Tez/Kod')
NB: 0.557780
CART: 0.641760
KNN: 0.585560
RFC: 0.641500
RC: 0.543480
BC: 0.641560
ABC: 0.612700
MLPC: 0.551360

In [10]: runfile('C:/Users/Ceyhunk/OneDrive - BAR
BARSAN GLOBAL LOJISTIK A.S/Tez/Kod')
NB: 0.557780
CART: 0.641760
KNN: 0.585560
RFC: 0.641800
RC: 0.543480
BC: 0.641520
ABC: 0.612700
MLPC: 0.541120
```

**Şekil 46.** Uygulamanın çıktılarının örnek gösterimi

Uygulama, verilerin daha sağlıklı değerlendirilebilmesi için 10 kez çalıştırılmış, her çalışma esnasında üretilen sonuçlar not alınmış ve elde edilen 10 farklı sonucun her yöntem için ortalaması bulunarak, sonuç analizinde bu ortalamalar dikkate alınmıştır.

**Karşılaştırma Sonuçlarının Analizi**

Uygulamadan elde edilen sonuçlar teste tabi tutulan her yöntem için ulaşılan ortalama sonuçlar **Tablo 30**'da gösterilmiştir.

**Tablo 30.** Makine öğrenmesi yöntemlerinin ulaşılan ortalama başarı oranları

| Yöntem | Ortalama Başarı Oranı |
|---|---|
| **Naive Bayes Sınıflandırması** | 0,55 |
| **Karar Ağacı Algoritması** | 0,64 |
| **En Yakın Komşuluk (k) Algoritması** | 0,58 |
| **Rastgele Orman Algoritması** | 0,64 |
| **Ridge Sınıflandırma Algoritması** | 0,54 |
| **Torbalama Yöntemi** | 0,64 |
| **ADA-Boost Yükseltme Yöntemi** | 0,61 |
| **Çok Katmanlı Algılayıcılar Yapay Sinir Ağı** | 0,55 |

Elde edilen sonuçlara göre, incelenen makine öğrenmesi algoritmalarının, verinin ham haliyle, bu çalışmada belirlenen özellik sütunlarını (feature) da dikkate alarak %54 ile %64 oranlarında başarı elde ettiği görülmüş, çalışma için belirlenen koşullarda başarı oranları yeterli bulunmamıştır. Depolardaki farklı örneklerde aynı ana özelliklerle veya farklı özellikler belirlenerek bu algoritmaların tekrar denenebileceği, ancak bu çalışmada kullanılamayacağı sonucuna varılmıştır.

**ABC Analizi Yöntemi**

Depo yönetiminde en fazla kullanılan yöntemlerden biri olan ABC yönteminin kökleri 50'li yıllara dayanmaktadır. Her ne kadar bu çalışmada geliştirilen uygulamaya hem yöntem olarak hem de kullanılan harflerle benzerlik gösterse de bu çalışma ABC analiz uygulamalarından oldukça farklıdır. Öncelikle ABC analizi yöntemi temelde 20-80 kuralına dayanmaktadır. Depodaki ürünler maliyet ve sayılarına göre en değerli %20'si ayrılarak değerlendirilmekte, buna göre de üç gruba ayrılmaktadır. Bu yöntemin depolarda kullanılmasına yönelik bir çalışma Atmaca (2020) tarafından demir-çelik sektörünün depolarına göre dizayn edilerek araştırılmıştır (Atmaca, 2020). Ancak

"**Depolarda Sipariş Toplama İşlemlerine Yönelik Akıllı Lokasyonlama Uygulaması**" (**DSTİYALU**) çalışmasında depodaki ürünler 20-80 yöntemiyle sınıflandırılmamış, sınıflandırma esnasında da sayı ve maliyetler kullanılmamıştır. Ayrıca DSTİYALU'da kullanılan grup sayıları da farklı olabilmektedir. Ek olarak bölgesel grup mantığı, bu çalışmaya özgü olarak geliştirilmiştir. Yine ürün ve bölge gruplarının sınıflandırıldığı algoritma da DSTİYALU'ya özgüdür.

**Devir Bazlı Yerleştirme Yöntemi ve Gruba Dayalı Yerleştirme Yöntemi**

Bu çalışmanın sirkülasyon hızını temel aldığı düşünüldüğünde, aynı temelde literatürde daha önce yapılan bir çalışma mevcuttur. Bu da Heskett vd (1964) yılında ortaya attığı **devir bazlı yerleştirme** yöntemidir. Bu çalışma da sirkülasyon hızını temel almış ve yine DSTİYALU'daki gibi bir indeks değeri kullanılmıştır. Ancak ürün sayısı çok fazla olan ve sürekli yeni ürünler gelen depolarda yeterince veriye ulaşılması mümkün olmamıştır (Heskett, Ivie ve Glaskowsky, 1964).

DSTİYALU'da ise sirkülasyon hızını temel alan anahtar değerleri ürünlere değil, 7 alandan oluşan bir gruba atanmıştır. Bu yönüyle DSTİYALU Frazelle (1989) yaptığı çalışmada **gruba dayalı yerleştirme** yöntemine benzemektedir. Ancak sirkülasyon hızı ve anahtar değerler bu çalışmada kullanılmamıştır (Frazelle E. H., 1989).

Ek olarak mevcut depoyu veri modelindeki gruplandırmalara uygun olarak fiziki olarak bölgesel gruplara ayırma mantığı da bu iki çalışmada yer almamaktadır.

**Sınıfa Dayalı Yerleştirme Yöntemi**

Petersen vd (2004) yazdıkları bir makalede devir bazlı yerleştirme yöntemi ile gruba dayalı yerleştirme yönteminin birleştirilebileceğini ve deponun belli bölgelere ayrılıp (zone) yerleştirmeler yapılarak verimin arttırabileceğini iddia etmişlerdir (Petersen, Aase ve Heiser, 2004).

Temel prensiplerde Petersen vd (2004) tarafından yazılan makalede önerilen yöntemlerden ana kriter olarak sirkülasyon hızının kullanılması, ürünlerin gruplara ayrılması ve deponun bölgelere ayrılması fikirleri DSTİYALU içinde de kullanılmıştır. Ancak çalışmada ürünlerin sınıflandırılması esnasında ABC analizi yöntemi benzeri bir yöntem önerilmiştir. Çalışmada veri setleri, gruplandırma için gerekli kriterler, bölgesel gruplandırmalar için kullanılacak algoritmalar gibi DSTİYALU'da yer alan veriler bulunmamaktadır. Çalışma, incelenen çalışmalar için de DSTİYALU'da en fazla

faydalanılan ve prensipte en çok benzeşen çalışma olmasına rağmen, anahtar değer belirleme algoritmaları, ürün gruplandırma algoritmaları ve bölge belirleme algoritmaları DSTİYALU'da farklılaşmaktadır.

**Sipariş Toplama Problemine Yönelik Yapılmış Çalışmalar**

Sipariş toplama problemine yönelik yapılan bazı çalışmalarla DSTİYALU'nun karşılaştırılmasında temel farkın, bu probleme yönelik çalışmaların genelde yerleştirilmiş ürün ile ilgilenmeleri, DSTİYALU'nun ise toplama hızını belirlemek için yerleştirme sürecine müdahale etmesi olduğu görülmektedir. Örneğin Yine sipariş toplama problemine yönelik bazı çalışmaların ise sipariş toplama problemini "rota hesaplama" olarak ele aldıkları görülmektedir. Örneğin Caron vd (1998) sezgisel rota belirleme yöntemleri önermişlerdir (Caron ve Marchet, 1998). De Koster vd (1999) yaptıkları çalışmada çekirdek algoritmalarını ürün toplamada kullanmışlardır (De Koster, Van der Poort ve Wolters, 1999). Yine Henn vd (2012) tabu arama yöntemi ile meta sezgisel gruplama yöntemiyle toplama önerisinde bulunmuştur (Henn ve Wäscher, 2012). Bu çalışmaların aksine DSTİYALU yerleştirilen ürünler üzerinden çalışmamakta, toplama problemini ortadan kaldırmak için yerleştirme aşamasına müdahale önermektedir.

**Literatürdeki Geçmiş Çalışmaların DSTİYAU ile Karşılaştırılması**

Yorumlar bölümünde incelenen makine öğrenmesi tabanlı algoritmalar direkt olarak başarı oranlarına göre sınıflandırılmış ve ilgili bölümde sonuçlar paylaşılmıştır. Bu algoritmaların mevcut veri seti üzerindeki sonuçları ve başarı oranları hakkında ilgili bölüme bakılabilir.

Çalışmanın literatür taramasındaki bahsedilen bazı yöntemlerle ile bu çalışmanın konusu olan yöntem, devamında yazılı olarak anlatılarak aradaki farklar detaylandırılmıştır. Her yöntemin kendi özellikleri ilgili bölümden takip edilebilir. Yine bölüm içlerinde DSTİYALU ile farklarından da bahsedilmiştir. Ancak bu çalışmada önerilen yöntem ile yorumlar bölümünde bahsedilen, makine öğrenmesi bölümü dışındaki, çalışmaların farkları ayrı bir sınıflandırmaya da tabi tutulmuştur. Bu aradaki farkların bir tablo şeklinde daha anlaşılır bir gösterimi **Tablo 31**'da mevcuttur.

**Tablo 31.** Literatürdeki bazı yöntemlerin DSTİYALU ile karşılaştırılması

| Yöntem | Veri Gruplama | Grup Sayısı | Bölge Gruplama | Gruplama Anahtarı | Rota Hesaplama |
|---|---|---|---|---|---|
| **ABC Analizi Yöntemi** | ABC | 3 | Yok | Sirkülasyon | Yok |
| **Devir Bazlı Yerleştirme** | Ürün | Sınırsız | Yok | Sirkülasyon | Yok |
| **Gruba Dayalı Yerleştirme** | Batch gruplaması | Sınırsız | Yok | Batch değeri | Yok |
| **Sınıfa Dayalı Yerleştirme** | ABC | 3 | Var | Sirkülasyon | Yok |
| **Sezgisel Rota Belirleme** | Yok | Yok | Yok | Yok | Var |
| **Çekirdek Algoritma Destekli Yöntem** | Yok | Yok | Yok | Yok | Var |
| **Meta Sezgisel Gruplama Yöntemi** | Özel algoritma | Sınırsız | Yok | Tabu arama | Var |
| **DSTİYALU** | Özel algoritma | Sınırsız | Var | Sirkülasyon | Yok |

Tabloda bahsedilen **Veri Gruplama** başlığı, incelenen algoritmanın verileri gruplara ayrılan bir algoritma kullanıp kullanmadığı, kullandığı ise genel olarak bu algoritmanın şeklidir. **Grup Sayısı** başlığı altında, oluşturulan bir gruplandırma algoritması varsa, verilerin kaç gruba ayrıldığıdır. **Bölge Gruplama** başlığı altında deponun lokasyonlarının bölgesel gruplara ayrılıp ayrılmadığı gösterilmiştir. **Anahtar** başlığı ise, veriler gruplandırılırken hangi temel kriterden hareket edildiğidir. **Rota Hesaplama** başlığı altında yöntemin taşıyıcı rotalarının mesafelerini hesaplayıp buna göre toplamayı düzenleyip düzenlemediği bilgisi incelenmiştir.

# BÖLÜM 6. SONUÇ

Bu bölümde, araştırmanın kimliği, süreci ve elde edilen sonuçların özetlendiği bir özet, elde edilen bulguların yorumlandıktan sonra araştırma ile varılan nokta anlamında bir yargı ile araştırılan problemin çözümü için başka hangi çalışmaların yapılmasının gerekli görüldüğüne ve mevcut araştırmanın alandaki egemen kuram ve uygulamalara muhtemel katkısının neler olabileceğine ilişkin araştırmacının özgün değerlendirmelerini içeren bir öneriler altbölümleri yer almıştır.

## Özet

Bu çalışma kapsamında önce incelenen konunun kökeni olan "sipariş toplama problemi" hakkında literatür taraması yapılmış, sipariş toplama süreçlerini daha önce inceleyen çalışmalar toparlanmıştır. Ardından problemin kendisinden özet olarak bahsedilmiştir. Çalışmanın amacı ve önemi, varsayımlar, sınırlılıklar yine giriş bölümünde incelenmiştir. Giriş bölümü altında son alarak çalışma için önemli kavramların tanımları yapılmıştır.

Bir sonraki bölüm olan araştırma ve planlama bölümünde, problemin detaylarına inilmiş, problemin ortaya çıktı depolar, depolarda yapılan operasyonlar, bu operasyonların icra biçimleri detaylı şekilde incelenmiştir. Devamında depo içinde gerçekleştirilen mal kabul operasyonu detaylı biçimde açıklanmış, bu operasyonun icrasıyla ilgili daha önce yapılan çalışmalardan bir literatür taraması yapılmıştır. Ardından sipariş toplama operasyonu detaylı şekilde açıklanmış, bu operasyonun başlangıç aşaması olan toplama listeleri hazırlanma sürecinin operasyon üzerindeki etkisi analiz edilmiştir.

Devamında 3. bölüm olan yöntem bölümüne geçilmiştir. Araştırma modeli detaylı şekilde anlatılmış, model hakkında sayısal bilgilere yer verilmiş, neden seçildiği örneklerle ortaya koyulmuştur. Seçilen model içinde kullanılacak örneklem belirlenmiş, bu örneklemin, evren üzerindeki durumu anlatılmıştır. Ardından seçilen örnekler üzerinden ham verilerin toplanması aşamasına geçilmiş, bu bölümde ham verinin hangi araçlardan toplandığı açıklanmıştır. Seçilen ham veri üzerindeki bütün alanlar tek tek incelenmiş ve çözümlenmiş, ayrıntılı biçimde açıklanmıştır. Verilerin yorumlanmasına geçilmeden önce yorumlama için kullanılan yöntemden ve oluşturulan uzman grubundan

bahsedilmiştir. Ardından ham veri tekrar yorumlanmış, gruplanmış, filtrelenmiş ve ikinci kez temel bileşenlerine indirgenerek gruplanmıştır.

Bir sonraki bölüm olan uygulama bölümünde çalışmada kullanılacak ana parametrelerden birisi olan anahtar değerinin nasıl oluşturulduğu açıklanmış, bu değerinin oluşmasıyla ilgili örnekler verilmiştir. Ardından oluşturulan veri seti üzerinde nasıl bir gruplama mantığı oluşturulacağı anlatılmış, bu mantığın fiziki olarak depolarda yansıması olan bölgesel gruplamanın nasıl yapıldığı açıklanmıştır. Devamında, ulaşılan veri ve sonuçların başka uygulamalar tarafından da kullanılabilmesi için oluşturulan fonksiyondan bahsedilmiş, kodları paylaşılmıştır. Uygulama bölümünün sonunda, yapılan uygulamanın bir saha test çalışmasında kullanımı gösterilmiştir.

Bulgular bölümünde ise, programın ham data üzerindeki teorik testleri yapılmış, ardından belli süredir çalışmakta olduğu canlı ortamda ortaya çıkan sonuçlar incelenmiştir.

Yorumlar bölümünde ise çalışmanın, daha önce yapılan benzer çalışmalar ile karşılaştırması detaylı biçimde yapılmıştır.

## Yargı

Yapılan ham veri testlerinde geliştirilen DSTİYALU algoritması, belirlenen bölgesel gruplara yerleştirme tahminlerinde %91'lik bir başarıyla olması gereken grubu doğru tahmin etmiştir. Bu oranının gösterdiği gibi, yapılan çalışma "uygulanabilir" bir çalışma olarak nitelenebilir. Ancak bu oranın aynı saha ortamında "sürdürülebilir" olduğunun kanıtlanması için daha uzun süreye ihtiyaç duyulabilir. Yine çalışmanın farklı saha testlerine de tabi tutulması da elzemdir. Yine de çalışmanın sınırlı saha testinde gösterdiği başarı oranı, dikkate değer görülebilir.

Bu çalışma her ne kadar farklı çalışmalardan faydalanıyor olsa da kendi içerisinde kullandığı veri seti biçimlendirme yöntemleri ve gruplama algoritmaları ile incelenen probleme farklı bir bakış açısı sunmuştur.

Yapılan çalışma bu haliyle, sahada kullanılmaya değer olarak görülmüş ve kullanımına devam etme kararı alınmıştır.

## Öneriler

Çalışma kapsamında farklı bölümlerde farklı yöntemler kullanılmıştır. Veri modeli oluşturulurken ve yorumlanırken, uygulama yazılırken farklı algoritmalar kullanılmış

veya yenileri geliştirilmiştir. Elbette bu kadar yöntemin bir araya geldiği bir çalışma farklı şekillerde de yorumlanabilir. Özellikle veri seti üzerinde temel bileşenlerin belirlendiği bölüm farklı bakış açılarına açıktır. Ayrıca **aykırı değer** belirlenen bölümlerde de farklı yöntemler kullanılabilir. Buralara yapılan müdahalelerin başarı oranları üzerindeki etkisi ölçümlenerek çalışma geliştirilebilir.

Yine çalışma içinde **anahtar değerin** belirlendiği bölüm de tekrar elden geçirilerek sonuçlar incelenebilir, anahtar değer belirlemede farklı algoritmalar denenebilir.

**Bölgesel gruplama mantığı** çalışmanın ana karakterlerinden biri olmasına rağmen istenirse sadece oluşturulan veri setlerinden veya veri seti oluşturma yöntemlerinden de faydalanabilir. Veri seti farklı makine öğrenmesi algoritmaları ile tekrar yorumlanarak daha farklı tahminlerde bulunulup sonuçları bu çalışmayla karşılaştırılabilir. Özellikle algoritmaların her geçen gün geliştiği ve yeni algoritmaların ortaya çıktığı düşünüldüğünde veri setinin yeni algoritmalarla tekrar değerlendirilmesi ortaya farklı sonuçlar çıkarabilecektir.

# KAYNAKÇA


Atmaca, A. D. (2020). *Lojistik Ve Tedarik Zincirinde Stok Yönetimi: Demir Çelik Sektöründe Bir Uygulama.* İstanbul: T.C. Maltepe Üniversitesi.

Ay, Ş. (2019, Aralık 16). *Ensemble Learning — Bagging Ve Boosting*. Deep Learning Türkiye: Https://Medium.Com/Deep-Learning-Turkiye/Ensemble-Learning-Bagging-Ve-Boosting-50643428b22b Adresinden Alındı

Aydin, O., Tasabat, S. E. (2017). Outlier Detection Method By Using Deep Neural Networks. *Pressacademia Procedia (5, 1),* 96-101.

Baray, A., Çakmak, E. (2015). Design Methodology For A Multiple-Level Warehouse Layout Based On Particle Swarm Optimization Algorithm. *İstanbul Üniversitesi İşletme Fakültesi İşletme İktisadı Enstitüsü Yönetim Dergisi , 25 (77),* 13-38.

Barnes, R. (2021, Ağustos 9). *How To Find Outliers With Sql*. The Data Scholl: Https://Dataschool.Com/How-To-Teach-People-Sql/How-To-Find-Outliers-With-Sql/ Adresinden Alındı

Barnes, R. (2021, Ağustos 9). *What Is The Interquartile Range?* The Data School: Https://Dataschool.Com/Fundamentals-Of-Analysis/What-Is-The-İnterquartile-Range/ Adresinden Alındı

Bartholdi, J. J., Hackman, S. T. (2014, 08 19). *Warehouse & Distribution Science Release 0.96.* Warehouse-Science.Com: Https://Www.Warehouse-Science.Com/ Adresinden Alındı

Baruffaldi, G., Accorsi, R., Manzini, R. (2019). Warehouse Management System Customization And İnformation Availability İn 3pl Companies. *Industrial Management & Data Systems*, 251-273.

Behera, B., Kumaravelan, G. (2020). Performance Evaluation Of Machine Learning Algorithms İn Biomedical Document Classification. *International Journal Of Advanced Science And Technology (29, 5),* 5704-5716.

Bgl. (2022, Mart 14). *Hakkımızda*. Barsan.Com: Https://Www.Barsan.Com/Kurumsal/Hakkimizda/ Adresinden Alındı

Caron, F., Marchet, G. (1998). Routing Policies And Coı-Based Storage Policies İn Picker-To-Part Systems. *International Journal Of Production Research 36(3),* 713-732.

Chen, M.-C., Wu, H.-P. (2005). An Association-Based Clustering Approach To Order Batching Considering Customer Demand Patterns. *Omega 33(4),* 333-343.

Dallari, F., Marchet, G., Melacini, M. (2008). Design Of Order Picking System. *The International Journal Of Advanced Manufacturing Technology 42(1),* 1-12.

De Koster, M. B., Neuteboom, A. J. (2001). *The Logistics Of Supermarket Chains : A Comparison Of Seven Chains İn The Netherlands.* Elsevier Business Information.

De Koster, M. B., Van Der Poort, E., Wolters, M. (1999). Efficient Orderbatching Method İn Warehouses. *International Journal Of Production Research Volume 37,* 1479-1504.

De Koster, R., Le-Duc, T., Roodbergen, K. (2007). Design And Control Of Warehouse Order Picking: A Literature Review. *European Journal Of Operational Research (182-2),* 481-507.

Dursun, E., Gürgen, E. (2020). Konteyner Terminal Stok Yönetiminde Abc Analizi Ve Bulanık Sınıflandırma. *Mustafa Kemal University Journal Of Social Sciences Institute (17, 46),* 563-583.

Frazelle, E. (2015). *World-Class Warehousing And Material Handling Second Edıtıon.* New York: Mcgraw-Hill Education,.

Frazelle, E. H. (1989). *Stock Location Assignment And Order Picking Productivity.* Georgia : Georgia Institute Of Technology.

Gu, J., Goetschalckx, M., Mcginnis, L. F. (2007). Research On Warehouse Operation: A Comprehensive Review. *European Journal Of Operational Research, Volume* 177, Issue 1, 16 February 2007, 1-21.

Günerkan, M., Şahinaslan, E., Şahinaslan, Ö. (2022). Gümrük Beyannamesi Sürecinde Öğrenmeye Dayalı Algoritmaların Etkinliğinin İncelenmesi . Acta Infologica,. DOI: 10.26650/acin.1057060

Hausman, W., Schwarz, L. B., Graves, S. C. (1976). Optimal Storage Assignment In Automatic Warehousing Systems. *Management Science 22(6),* 629-638.

Henn, S., Wäscher, G. (2012). Tabu Search Heuristics For The Order Batching Problem In Manual Orderpicking Systems. *European Journal Of Operational Research 222(3)*, 484-494.

Heskett, J. L., Ivie, R. M., Glaskowsky, N. A. (1964). *Business Logistics : Management Of Physical Supply And Distribution.* New York: Ronald Press Co.

Ho, T. K. (1995). Random Decision Forests. *Proceedings Of The 3rd International Conference On Document Analysis And Recognition* (S. 278–282). Montreal: Qc.

Hwang, H. S., Cho, G. S. (2006). A Performance Evaluation Model For Order Picking Warehouse Design. *Computers & Industrial Engineering 51(2)*, 335-342.

Kamiński, B., Jakubczyk, M., Szufel, P. (2018). A Framework For Sensitivity Analysis Of Decision Trees. *Central European Journal Of Operations Research (26, 1)*, 135–159.

Kaynar, O., Taştan, S., Demirkoparan, F. (2011). Yapay Sinir Ağları İle Doğalgaz Tüketim Tahmini. *Atatürk Ü. İibf Dergisi*, 463-474.

Kılıç, E., Özgönenel, O., Usta, Ö., Thomas, D. (2009). Pca Based Protection Algorithm For Transformer Internal Faults. *Turkish Journal Of Electrical Engineering And Computer Science (17, 2)*, 125-142.

Mahesh, B. (2018). Machine Learning Algorithms - A Review. *International Journal Of Science And Research (Ijsr)*, 381-386.

Özçakar, N., Görener, A., Arıkan, M. V. (2012). Depolama Sistemlerinde Sipariş Toplama İşlemlerinin Genetik Algoritmalarla Optimizasyonu. *Isletme Iktisadi EnstitüSü YöNetim Dergisi (71)*, 118-144.

Pan, J. C.-H., Shih, P.-H. (2008). Evaluation Of The Throughput Of A Multiple-Picker Order Picking System With Congestion Consideration. *Computers & Industrial Engineering 55(2)*, 379-389.

Petersen, C. G. (1997). An Evaluation Of Order Picking Routeing Policies. *International Journal Of Operations & Production Management*, 1098-1111.

Petersen, C. G., Aase, G. R., Heiser, D. R. (2004). Improving Order-Picking Performance Through The İmplementation Of Class-Based Storage. *International Journal Of Physical Distribution & Logistics Management*, 534-544.

Ramaa, A., Subramanya, K., Rangaswamy, T. (2012). Impact Of Warehouse Management System. *International Journal Of Computer Applications, Volume 54*, 14-20.

Roodbergen, K. J., Koster, R. B. (2001). Routing Methods For Warehouses With Multiple Cross Aisles. *International Journal Of Production Research 39(9)*, 1865-1883.

Roodbergen, K. J., Vis, I. F. (2009). A Survey Of Literature On Automated Storage And Retrieval Systems. *European Journal Of Operational Research 194(2)*, 343-362.

Shadkam, E., Bijari, M. (2014). Evaluation The Efficiency Of Cuckoo Optimization Algorithm. *International Journal On Computational Science & Applications 4(2)*, 39-47.

Shiau, J.-Y., Lee, M.-C. (2010). A Warehouse Management System With Sequential Picking For Multi-Container Deliveries. *Computers & Industrial Engineering (58, 3)*, 382-392.

St-Vincent, M., Denis, D., Imbeau, D., Laberge, M. (2005). Work Factors Affecting Manual Materials Handling İn A Warehouse Superstore. *International Journal Of Industrial Ergonomics (35, 1)*, 33-46.

Şahin, A. E. (2001). Eğitim Araştırmalarında Delphi Tekniği Ve Kullanımı. *Hacettepe Üniversitesi Eğitim Fakültesi Dergisi (20)*, 215-220.

Şahin, Y., Kulak, O. (2013). Depo Operasyonlarının Planlanması İçin Genetik Algoritma Esaslı Modeller. *Uluslararası Alanya İşletme Fakültesi Dergisi 5(3)*, 141-153.

Şahinaslan, E. Şahinaslan, Ö. (2022). Microsoft SQL Sunucusunda Veritabanı Kurtarma Teknikleri . International Journal of Innovative Engineering Applications , 6 (1) , 158-169 . DOI: 10.46460/ijiea.1070325

Şahinaslan, Ö., Dalyan, H., Şahinaslan, E., 2022. Naive Bayes Sınıflandırıcısı Kullanılarak YouTube Verileri Üzerinden Çok Dilli Duygu Analizi. Bilişim Teknolojileri Dergisi, 15(2), 221-229. doi: 10.17671/gazibtd.999960

Tanyaş, M. (2015). *İstanbul Lojistik Sektör Analizi.* İstanbul: Müsiad Araştırma Raporları.

Theorem, A. R. (2003). A Rough Set View On Bayes' Theorem. *International Journal Of Intelligent Systems (18, 5)*, 487-498.

Tompkins, J. A., White, J. A., Bozer, Y. A., Tanchoco, J. (2010). *Facilities Planning, 4e.* New Jersey: John Wiley & Sons, Inc.

Triguero, I., García-Gil, D., Maillo, J., Luengo, J., Salvador García, F. H. (2019). Transforming Big Data Into Smart Data: An İnsight On The Use Of The K-Nearest Neighbors Algorithm To Obtain Quality Data. *Wıres Data Mining And Knowledge Discovery*, 1-24.

Tuna, G., Tunçel, G. (2012). Depo Yönetiminde Sipariş Toplama Sistemleri: Bir Literatür Araştırması. *Deü Mühendislik Fakültesi Mühendislik Bilimleri Dergisi (14-2)*, 15-31.

Tunç, S., Kutlu, B., Zincidi, A., Atmaca, E. (2008). Depo Sisteminde Sipariş Toplama Sürecinin İyileştirilmesi. *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi (23-2)*, 357-364.

Van Den Berg, J., Zijm, W. (1999). Models For Warehouse Management: Classification And Examples. *Uluslararası Üretim Ekonomisi Dergisi, Cilt 59, Sayı 1-3*, 519-528.

Varoquaux, G., Müller, A. (2022, Nisan 16). *Classifier Comparison*. Scikit Learn: Https://Scikit-Learn.Org/Stable/Auto_Examples/Classification/Plot_Classifier_Comparison.Html Adresinden Alındı

Yazgan, H. R., Cömert, S. E., Kılıç, E. N. (2020). Araç Rotalama Probleminin Sezgisel Algoritmalar İle Çözülmesi: Bir Boya Fabrikasında Uygulama. *Journal Of Turkish Operations Management (4, 2)*, 549-563.

# ÖZGEÇMİŞ

**Ceyhun Karataş**

**Eğitim**

| Derece | Yıl | Üniversite, Enstitü, Anabilim/Anasanat Dalı |
|---|---|---|
| Y.Ls. | 2020 | Maltepe Üniversitesi, Lisansüstü Eğitim Enstitüsü<br>Bilgisayar Mühendisliği Anabilim Dalı |
| Ls. | 2011 | Anadolu Üniversitesi, İşletme Fakültesi<br>İşletme Anabilim Dalı |
| Ön Ls. | 2005 | İstanbul Üniversitesi, TBMYO Bilgisayar Programlama |
| Lise | 2000 | Çorum Anadolu Ticaret Meslek Lisesi |

**İş/İstihdam**

| Yıl | Görev |
|---|---|
| 2018 - | AR-GE Direktörü, Barsan Global Lojistik A.Ş. |
| 2011 - 2018 | Yazılım Müdürü, Barsan Global Lojistik A.Ş. |
| 2003 - 2011 | Yazılım Müdürü, Egemen Yazılım Ltd. Şti. |