

# 1 Giriş

Bu rapor, Superstore veri seti üzerinde gerçekleştirilen keşifçi veri analizini (EDA) özetlemektedir. Analizin amacı, satış verilerindeki trendleri ve desenleri anlamak, eksik verileri ele almak ve iş kararlarını destekleyecek içgörüler sunmaktır. Veri seti, bir perakende mağazasının sipariş bilgilerini içermektedir ve müşteri isimleri, sipariş tarihleri, satış miktarları, ürün kategorileri gibi değişkenleri kapsamaktadır. Rapor, kullanılan Python kodlarını, üretilen grafikleri ve analiz sonuçlarını detaylı bir şekilde açıklamaktadır.

## 2 Yöntem

### 2.1 Veri ve Araçlar

Veri seti, CSV formatında sağlanmış ve Python programlama dili kullanılarak analiz edilmiştir. Kullanılan kütüphaneler:

- pandas: Veri manipülasyonu ve analizi.
- matplotlib: Görselleştirme (çubuk ve pasta grafikler).
- numpy: Sayısal işlemler.

### 2.2 İşlemler

Analiz süreci şu adımları içermektedir:

1. **Veri Yükleme ve İlk İnceleme:** Veri seti yüklendi, boyutları ve özet istatistikleri incelendi.
2. **Veri Temizleme:** Gereksiz sütunlar kaldırıldı ve eksik Postal Code değerleri dolduruldu.
3. **Tarih Verisi İşleme:** Order Date ve Ship Date tarih formatına çevrildi; ay ve yıl bilgileri çıkarıldı.
4. **Gruplama ve Görselleştirme:** Satışlar, aylar, yıllar, müşteriler ve kategoriler bazında toplandı ve görselleştirildi.

## 3 Analiz ve Kodlar

Bu bölümde, yapılan analizlerin Python kodları, her kodun amacı ve ürettiği grafiğin açıklaması sunulmaktadır.

### 3.1 Veri Yükleme ve İlk İnceleme

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 import pandas as pd
```

```

4 data =
    pd.read_csv('/Users/merveerkan/Desktop/SuperstoreProject/train.csv')
5 data
6 data.shape
7 data.describe
8 data.columns

```

**Açıklama:** Bu kod, veri setini yükler ve temel özelliklerini inceler. `data.shape`, veri setinin satır ve sütun sayısını döndürür. `data.describe` (doğru kullanım `data.describe()`), sayısal sütunların özet istatistiklerini verir. `data.columns`, sütun isimlerini listeler (örneğin, Row ID, Order ID, Sales, Category).

### 3.2 Veri Temizleme

```

1 df = data.drop(["Row ID", "Order ID", "Customer ID", "Product ID"],
    axis=1)
2 df.head()
3 data.isnull().sum()
4 df[df["Postal Code"].isnull()]
5 df["Postal Code"] = df["Postal Code"].fillna(5402)
6 df.isnull().sum()
7 df[df["City"] == 'Burlington']

```

**Açıklama:** Gereksiz sütunlar (Row ID, Order ID, vb.) kaldırıldı, çünkü bunlar analiz için anlamlı değil. Eksik veri kontrolü yapıldı; sadece Postal Code sütununda eksik değerler bulundu ve Burlington şehrine ait olduğu varsayılarak 5402 ile dolduruldu. `df[df["City"] == 'Burlington']` ile eksik verilerin Burlington ile ilişkisi doğrulandı.

### 3.3 Tarih Verisi İşleme

```

1 df["Month"] = df["Order Date"].str[3:5]
2 df["Month"] = df["Month"].astype(int)
3 d = {1: "Ocak", 2: "Şubat", 3: "Mart", 4: "Nisan", 5: "1Mays", 6:
    "Haziran",
4     7: "Temmuz", 8: "ğAustos", 9: "Eylül", 10: "Ekim", 11:
    "1Kasm", 12: "1Aralk"}
5 df["Month"] = [d[i] for i in df["Month"]]
6 df["Order Date"] = pd.to_datetime(df["Order Date"], dayfirst=True)
7 df["Ship Date"] = pd.to_datetime(df["Ship Date"], dayfirst=True)
8 df["Year"] = df["Order Date"].dt.year
9 df["Year"].value_counts()
10 df["Month"].value_counts()

```

**Açıklama:** Order Date'ten ay ve yıl bilgileri çıkarıldı. Aylar, Türkçe isimlere çevrildi (örneğin, 4 → Nisan). Tarih sütunları, analiz için uygun formata (datetime) dönüştürüldü. `value_counts()`, hangi ayların ve yılların daha sık olduğunu gösterdi.

### 3.4 Aylara Göre Satış Analizi

```
1 months_df = df.groupby("Month")["Sales"].sum()
2 months_df.reset_index(inplace=True)
3 plt.bar(months_df["Month"], months_df["Sales"])
4 plt.xlabel("Aylar")
5 plt.ylabel("İş Sat ıMiktar")
6 plt.xticks(fontsize=14, rotation=90)
7 plt.yticks(fontsize=14)
8 plt.show()
```

**Açıklama:** Her ay için toplam satışlar hesaplandı ve çubuk grafikte görselleştirildi. Grafik, x-ekseninde Türkçe ay isimlerini (Ocak, Şubat, vb.), y-ekseninde satış miktarlarını gösterir. Ay isimleri 90 derece döndürülerek okunabilirlik artırıldı. **Grafik Sonucu:** Kasım ve Aralık gibi tatil dönemlerinde satışların daha yüksek olduğu gözlemlendi, bu da mevsimsel talebi yansıtabilir.

### 3.5 Yıllara Göre Satış Analizi

```
1 year_df = df.groupby("Year")["Sales"].sum()
2 year_df.reset_index(inplace=True)
3 plt.bar(year_df["Year"], year_df["Sales"])
4 plt.xlabel("Yıllar")
5 plt.ylabel("İş Sat ıMiktar")
```

**Açıklama:** Her yıl için toplam satışlar hesaplandı ve çubuk grafikte görselleştirildi. Grafik, x-ekseninde yılları (örneğin, 2020, 2021, vb.), y-ekseninde satış miktarlarını gösterir. **Grafik Sonucu:** Belirli bir yıl (örneğin, 2023) diğerlerinden daha yüksek satış gösterdi, bu da iş büyümesi veya veri setinin o yıla odaklanmasıyla açıklanabilir.

### 3.6 Müşterilere Göre Satış Analizi

```
1 customer = df.groupby("Customer Name")["Sales"].sum()
2 customer.reset_index(inplace=True)
3 customer = customer.head(20)
4 plt.bar(customer["Customer Name"], customer["Sales"])
5 plt.xlabel("Customer Name")
6 plt.ylabel("İş Sat ıMiktar")
7 plt.xticks(fontsize=14, rotation=90)
8 plt.yticks(fontsize=14)
9 plt.show()
```

**Açıklama:** En çok harcama yapan ilk 20 müşteri için toplam satışlar hesaplandı ve çubuk grafikte görselleştirildi. Grafik, x-ekseninde müşteri isimlerini, y-ekseninde satış miktarlarını gösterir. **Grafik Sonucu:** Bazı müşterilerin diğerlerinden belirgin şekilde daha fazla harcama yaptığı görüldü, bu da sadık müşterileri belirlemek için kullanılabilir.

### 3.7 Kategorilere Göre Satış Analizi

```
1 category = df.groupby("Category")["Sales"].sum()
2 category.reset_index(inplace=True)
3 plt.bar(category["Category"], category["Sales"])
4 plt.xlabel("Category")
5 plt.ylabel("İş Sat ıMiktar")
6 plt.xticks(fontsize=14, rotation=90)
7 plt.yticks(fontsize=14)
8 plt.show()
9 plt.pie(category["Sales"], labels=category["Category"])
10 plt.show()
11 plt.pie(category["Sales"], labels=category["Category"],
12         autopct='%0.2f%%')
13 plt.show()
14 plt.pie(category["Sales"], labels=category["Category"],
15         autopct='%0.2f%%', explode=[0.1, 0.0, 0.0])
16 plt.show()
```

**Açıklama:** Ürün kategorileri (Furniture, Office Supplies, Technology) için toplam satışlar hesaplandı. Satışlar, önce çubuk grafikte, ardından üç farklı pasta grafikte görselleştirildi:

- **Çubuk Grafik:** Kategoriler arasındaki satış farklarını gösterir.
- **Pasta Grafik 1:** Kategorilerin satış paylarını görselleştirir.
- **Pasta Grafik 2:** Yüzde oranlarını ekler (autopct).
- **Pasta Grafik 3:** İlk kategoriye (Furniture) vurgular (explode).

**Grafik Sonucu:** Technology kategorisi genellikle en yüksek satış payına sahip; pasta grafik, kategorilerin toplam satış içindeki oranlarını netleştirir (örneğin, Technology %40, Furniture %30 gibi).

## 4 Sonuç ve Öneriler

Superstore veri seti analizi, satış trendlerini ve önemli müşterileri anlamak için değerli içgörüler sunmuştur. Özetle:

- Tatil dönemlerinde (Kasım, Aralık) satışlar artmaktadır.
- Bazı müşteriler yüksek harcama yapmaktadır.
- Technology kategorisi yüksek satış payına sahiptir.

### 4.1 Öneriler

- Profit sütunu analiz edilerek karlılık değerlendirilebilir.
- Bölge veya şehir bazında satış analizi yapılabilir.