

1. Veriyi Bulmak

Verileri veri tabanlarından, veri ambarlarından alıp kullanabileceğiniz gibi kendi yapay veri setlerinizi de oluşturabilirsiniz.

2. Büyük Resmi Görmek

Büyük resimden kasıt sistemin ana hatlarını oluşturabilmektir.

3. Veriyi Almak

Probleme uygun veri setini seçmeliyiz. Ne kadar veriye ihtiyaç olduğu önceden belirlenmeli. Çalışma alanını oluştururuz ardından veriyi yükleriz veriyi kolay kullanabileceğimiz biçime dönüştürürüz devamında verilerin boyutunu kontrol edip test setini oluştururuz.

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

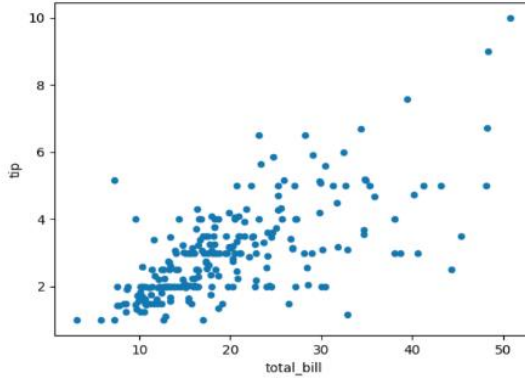
```
veri_seti = sns.load_dataset('tips')
```

4. Veriyi Anlamak

Veri araştırması, verilerin istatistiksel ve görselleştirme teknikleriyle tanımlanmasıdır. Girdi verilerinin kalitesi çıktı verilerinin kalitesini etkiler.

Veriyi Görselleştirme örneği

```
veri_seti.plot(x='total_bill', y='tip', kind='scatter')
plt.show()
```



5. Veriyi Hazırlamak

Veri setimizi hazırlamak için aşağıdaki adımlar yapılmalıdır.

1=Veri Temizleme

2=Özellik Seçimi

3=Gerekirse, Özellik Mühendisliği

4=Özellik Ölçeklendirme

6. Modeli Seçmek ve Eğitmek

Model eğitimi, veri seti üzerinde makine öğrenmesi algoritmasının kullanılmasıyla gerçekleştirilir. Bu süreç, veri setindeki desenleri ve ilişkileri yakalamak için modelin parametrelerinin ayarlanmasını içerir. Eğitim süreci, iteratif olarak gerçekleştirilir ve modelin performansı, belirli bir ölçüt (örneğin, hata oranı) kullanılarak değerlendirilir. Eğitim süreci, genellikle kayıp fonksiyonunun minimize edilmesi amacıyla gerçekleştirilir.

```
# Gerekli kütüphaneleri yükleyelim
import pandas as pd
import numpy as np
from sklearn.datasets import load_boston
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score

# Boston Housing veri setini yükleyelim
boston = load_boston()
X = boston.data
y = boston.target

# Veriyi eğitim ve test setlerine bölelim
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Linear regresyon modelini oluşturalım ve eğitelim
lr = LinearRegression()
lr.fit(X_train, y_train)

# Modeli test seti üzerinde tahmin yaparak test edelim
y_pred = lr.predict(X_test)

# Modelin performansını değerlendirelim
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
print(f'Ortalama Kare Hata (MSE): {mse:.2f}')
print(f'R^2 Skoru: {r2:.2f}')
```

7. Sonuçları Sunmak

Ortalama Kare Hata (MSE): 24.29

R^2 Skoru: 0.67

8. Sistem Takibi Bakımı

Sistemin performansını yeni gelen verilerle düzenli olarak kontrol etmeliyiz. Eğer modeller düzenli olarak yeni verilerle eğitilmezse çürümeye doğru gider.