T.C.

SAKARYA ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR VE BİLİŞİM MÜHENDİSLİĞİ ANABİLİM DALI

BİLİŞİM TEKNOLOJİLERİ PR. (YL) (UZAKTAN EĞİTİM)

PHYTON İLE VERİ ANALİZİ

ÖDEV1

**Hazırlayan**

SEDAT ÖZTÜRK

E235013168

**Öğretim Üyesi**

Prof. Dr. İSMAİL HAKKI CEDİMOĞLU

EKİM 2024

### Veri setini Pandas DataFrame ile bilgisayara almak

import pandas as pd

file\_path = 'OZTURK\_DS.csv'

DATA = pd.read\_csv(file\_path)

### 2. DataFrame verileri ile ilgili özet bilgileri ekranda gösteriniz.

DATA.describe(include='all')

### 3. Pandas kütüphanesi versiyonunu bulunuz.

pd.\_\_version\_\_

### 4. Install edilen kütüphaneleri listeleyiniz.

!pip list

### 5. yazılımının versiyonunu bulunuz.

### import sys

### print("Python versiyonu:", sys.version)

### 6. Veri setindeki ilk 5 veriyi listeleyiniz.

DATA.head()

### 7. Veri setindeki son 5 veriyi listeleyiniz.

DATA.tail()

### 8. Veri setindeki bütün verileri listeleyiniz.

DATA

### 9. Veri setindeki nümerik alanları listeleyiniz.

DATA.select\_dtypes(include='number').head()

### 10. Veri setindeki nümerik olmayan alanların frekans dağılımını analiz ediniz.

columns = DATA.select\_dtypes(exclude='number').columns

frequency = {col: DATA[col].value\_counts() for col in columns}

for column, distribution in frequency.items():

print(f"--- {column} ---")

print(distribution)

print("\n")

### 11. Histogram diyagramını çiziniz.

import matplotlib.pyplot as plt

DATA.hist(figsize=(10, 10))

plt.show()

### 12. Veri setindeki toplam veri sayısını bulunuz.

DATA.size

### 13. Veri setindeki toplam sütun sayısını bulunuz.

DATA.shape[1]

### 14. Veri setindeki sütun isimlerini bulunuz.

DATA.columns

### 15. Veri setindeki eksik verileri bulunuz.

missing\_values = DATA.isnull().sum()

missing\_values = missing\_values[missing\_values > 0]

print(missing\_values)

### 16. Her bir sütundaki eksik veri sayısını bulunuz.

missing\_values = DATA.isnull().sum()

print(missing\_values)

### 17. Bütün sütunlardaki toplam eksik veri sayısını bulunuz.

total\_missing\_values = DATA.isnull().sum().sum()

print("Bütün sütunlardaki toplam eksik veri sayısı:", total\_missing\_values)

### 18. Eksik verilerin yerine "0 - sıfır" yazınız.

data\_filled = DATA.fillna(0)

print(data\_filled)

### 19. Eksik verileri DataFrame'den çıkarınız.

data\_dropped = DATA.dropna()

print(data\_dropped)

### 20. Tekrarlı verilerin sayısını bulunuz.

duplicate\_count = DATA.duplicated().sum()

print("Tekrarlı verilerin sayısı:", duplicate\_count)

### 21. info() fonksiyonunun değişik varyantlarını DataFrame üzerinde uygulayınız.

print("Belirli sayıda sütun gösterimi (max\_cols=5):")

DATA.info(max\_cols=5)

print("\n")

print("Bellek kullanımı dahil info() (memory\_usage='deep'):")

DATA.info(memory\_usage='deep')

### 22. CSV dosyasını herhangi bir dizine kaydedip karşılaştırınız.

DATA.to\_csv('OZTURK\_DS\_copy.csv')

import filecmp

filecmp.cmp(file\_path, 'OZTURK\_DS.csv')

### 23. DataFrame'deki herhangi bir sütunu ekrana alınız ve veri tipini sorgulayınız.

DATA['Exam\_Score'].dtype

### 24. Sütunun Class'ını sorgulayınız.

type(DATA['Exam\_Score'])

### 25. DataFrame’de "loc" deyimi kullanarak indeks etiketi ile sorgu yapınız.

DATA.loc[0]

### 26. DataFrame’de "iloc" deyimi kullanarak indeks numarası ile sorgu yapınız.

DATA.iloc[0]

### 27. Satır ve sütunu birlikte seçerek "loc" deyimi ile sorgu yazınız.

DATA.loc[0, 'Exam\_Score']

### 28. "loc" deyimini kullanarak DataFrame’de "Slicing" işlemi yapınız.

DATA.loc[0:5, ['Exam\_Score', 'Hours\_Studied']]

### 29. DataFrame’de filtreleme işlemi yapınız.

DATA[DATA['Exam\_Score'] > 70]

### 30. DataFrame’de indeksi sıfırlayınız.

data\_reset = DATA.reset\_index(drop=True)

print(data\_reset)

### 31. DataFrame’e yeni bir sütun ilave ediniz.

DATA['New\_Column'] = DATA['Exam\_Score'] \* 1.1

### 32. DataFrame’den bir sütunu geçici olarak siliniz.

data\_temp = DATA.drop(columns=['New\_Column'])

data\_temp

### 33. DataFrame’den bir sütunu kalıcı olarak siliniz.

DATA.drop(columns=['New\_Column'], inplace=True)

DATA

### 34. DataFrame hakkındaki genel bilgiyi ekranda listeleyiniz.

DATA.info()

### 35. System kütüphanesi versiyonunu bulunuz.

import sys

sys.version

### 36. Pandas kütüphanesinin versiyonunu bulunuz.

pd.\_\_version\_\_

### 37. Matplotlib kütüphanesinin versiyonunu bulunuz.

import matplotlib

matplotlib.\_\_version\_\_

### 38. Veri setindeki min, max, mean, count değerlerini bulunuz.

DATA.describe()

### 39. pd.pivot\_table() metodunu kullanarak, DataFrame'deki bir sütundan tek indeksli "pivot table" oluşturunuz.

pd.pivot\_table(DATA, values='Exam\_Score', index='Gender')

### 40. pd.pivot\_table() metodunu kullanarak, birden fazla sütundan çok indeksli "pivot table" oluşturunuz.

pd.pivot\_table(DATA, values='Exam\_Score', index=['Gender', 'School\_Type'])

### 41. Pivot table'da aggfunc=np.sum fonksiyonunu kullanınız.

import numpy as np

pd.pivot\_table(DATA, values='Exam\_Score', index=['Gender'], aggfunc=np.sum)

### 42. DataFrame'i CSV dosyası olarak kaydediniz.

DATA.to\_csv('DATA.csv', index=False)

### 43. DataFrame'i Excel dosyası olarak kaydediniz.

DATA.to\_excel('DATA.xlsx', index=False)

### 44. DataFrame'i HTML dosyası olarak kaydediniz.

DATA.to\_html('DATA.html', index=False)

### 45. DataFrame'i JSON dosyası olarak kaydediniz.

DATA.to\_json('DATA.json')

### 46. DataFrame'i txt dosyası olarak kaydediniz.

DATA.to\_csv('DATA.txt', index=False, sep='\t')

### 47. DataFrame'deki bir sütunu küçükten büyüğe sıralayınız.

DATA.sort\_values('Exam\_Score', ascending=True)

### 48. DataFrame'deki bir sütunu büyükten küçüğe sıralayınız.

DATA.sort\_values('Exam\_Score', ascending=False)

### 49. DataFrame'deki iki sütunu küçükten büyüğe sıralayınız.

DATA.sort\_values(['Exam\_Score', 'Hours\_Studied'], ascending=True)

### 50. DataFrame'deki iki sütunu büyükten küçüğe sıralayınız.

DATA.sort\_values(['Exam\_Score', 'Hours\_Studied'], ascending=False)

### 51. DataFrame'deki 5 veriyi tesadüfi olarak listeleyiniz.

DATA.sample(5)

### 52. Veri seti ile ilgili olarak Scatter plot

DATA.plot.scatter(x='Hours\_Studied', y='Exam\_Score')

plt.show()

### 53. Histogram

DATA['Exam\_Score'].plot.hist()

plt.show()

### 54. Bar chart

DATA['Gender'].value\_counts().plot.bar()

plt.show()

### 55. Pie chart

DATA['Gender'].value\_counts().plot.pie(autopct='%1.1f%%')

plt.show()

### 56. Heat Map grafiklerini çiziniz.

import seaborn as sns

correlation\_matrix = DATA.select\_dtypes(include='number').head().corr()

plt.figure(figsize=(10, 8))

sns.heatmap(correlation\_matrix, annot=True, cmap='viridis', linewidths=0.5)

plt.title('Heatmap of Correlation Matrix')

plt.show()

### 57. Seaborn Kütüphanesi kullanarak "Stripplot grafiğini" çiziniz.

import seaborn as sns

plt.figure(figsize=(10, 6))

sns.stripplot(x='Gender', y='Exam\_Score', data=DATA, jitter=True)

plt.title('Stripplot of Exam Score by Gender')

plt.xlabel('Gender')

plt.ylabel('Exam Score')

plt.show()

### 58. Seaborn Kütüphanesi kullanarak "Swarmplot grafiğini" çiziniz.

import seaborn as sns

sampled\_data = DATA.sample(frac=0.1, random\_state=1)

plt.figure(figsize=(10, 6))

sns.swarmplot(x='Gender', y='Exam\_Score', data=sampled\_data)

plt.title('Swarmplot of Exam Score by Gender (Sampled Data)')

plt.xlabel('Gender')

plt.ylabel('Exam Score')

plt.show()

### 59. Keras kütüphanesini yükleyiniz.

!pip install keras

### 60. Keras kütüphanesinin versiyonunu bulunuz.

import keras as k

print("Keras versiyonu:", k.\_\_version\_\_)

### 61. CSV dataset'den sadece iki sütunu DataFrame'e import ediniz.

DATA\_subset = DATA[['Hours\_Studied', 'Exam\_Score']]

DATA\_subset

### 62. İki sayıyı toplamak, çıkartmak, bölmek ve çarpmak için .PY uzantılı bir SCRIPT veya MODÜL oluşturunuz.

import mathop

a = float(input("Birinci sayıyı giriniz: "))

b = float(input("İkinci sayıyı giriniz: "))

toplam = mathop.add(a, b)

cikarma = mathop. subtract (a, b)

carpim = mathop.multiply(a, b)

bolme = mathop. divide(a, b)

print(f"{a} + {b} = {toplam}")

print(f"{a} - {b} = {cikarma}")

print(f"{a} \* {b} = {carpim}")

print(f"{a} / {b} = {bolme}")

### 63. Sweetviz Kütüphanesini, Keşifsel Veri Analizi yapmak için kullanınız.

!pip install sweetviz

import sweetviz as sv

report = sv.analyze(DATA)

report.show\_html('DATA\_report.html')

### 64. ydata\_profiling kütüphanesi kullanılarak DataFrame'deki veriler hakkında "html" rapor hazırlayınız.

!pip install ydata-profiling

from ydata\_profiling import ProfileReport

profile = ProfileReport(DATA, title="Öztürk DF Profil Raporu")

profile.to\_file("DATA\_profile.html")

### 65. yfinance kütüphanesini kullanınız.

!pip install yfinance

import yfinance as yf

data = yf.download("MSFT", start="2024-01-01", end="2024-06-01")

data.head()

### 66. DASK büyük veri kütüphanesini kullanınız.

!pip install dask

import dask.dataframe as dd

ddf = dd.from\_pandas(DATA, npartitions=3)

ddf.head()

### 67. Vaex kütüphanesini kullanınız.

!pip install vaex

import vaex

df\_vaex = vaex.from\_pandas(DATA)

df\_vaex.head()

### 68. Bir metni farklı dillere çeviriniz.

!pip install googletrans==4.0.0-rc1

from googletrans import Translator

translator = Translator()

text = "Merhaba Dünya"

languages = ['en', 'fr', 'it', 'de', 'zh-cn']

for lang in languages:

translated = translator.translate(text, dest=lang)

print(f"{lang} diline çeviri: {translated.text}")

### 69. Bir metnin özetini çıkarınız.

!pip install scipy==1.12

!pip install "gensim==3.8.1"

!pip install gensim

from gensim.summarization import summarize

text = """Hayaller herkesin ulaşmak istediği hedeflerdir.Çoğu insan ulaşır çoğu insan ulaşmaz.Ama en önemlisi hayallerimizden vazgeçmemek.Bir insan hayallerinden vazgeçerse onlara ulaşma ihtimali düşer.

Ben birazcık hayalimden bahsedeyim:

Benim en büyük hayalim öğretmen olmak.Öğretmen olup öğrencilerimle vakit geçirmek en büyük hayalim diyebilirim.

Bu hayalimden vazgeçmek istemiyorum.Ama büyükçe fikirlerimizde değişebilir.Benim kararımda değişebilir.İnsanlar her zaman istedikleri her şeyi yaparlar yeter ki o şeyden vazgeçmesinler.İnsan bir şeyi yapmak istese önünde hiç bir engel duramaz eğer engeller varsa o engelleri aşabilir."""

summary = summarize(text)

print(summary)

### 70. Bir metnin Anahtar Kelimelerini bulunuz.

from gensim.summarization import keywords

text = """Hayaller herkesin ulaşmak istediği hedeflerdir.Çoğu insan ulaşır çoğu insan ulaşmaz.Ama en önemlisi hayallerimizden vazgeçmemek.Bir insan hayallerinden vazgeçerse onlara ulaşma ihtimali düşer.

Ben birazcık hayalimden bahsedeyim:

Benim en büyük hayalim öğretmen olmak.Öğretmen olup öğrencilerimle vakit geçirmek en büyük hayalim diyebilirim.

Bu hayalimden vazgeçmek istemiyorum.Ama büyükçe fikirlerimizde değişebilir.Benim kararımda değişebilir.İnsanlar her zaman istedikleri her şeyi yaparlar yeter ki o şeyden vazgeçmesinler.İnsan bir şeyi yapmak istese önünde hiç bir engel duramaz eğer engeller varsa o engelleri aşabilir."""

key\_words = keywords(text)

print(key\_words)