O’ng’arov Sherzodjon 471-guruh

[ ]

import numpy as np

import pandas as pd

import seaborn as sns

import matplotlib.pyplot as plt

%matplotlib inline



[ ]

import matplotlib.pyplot as plt

import seaborn as sns

%matplotlib inline



[ ]

#1-modul

df3=pd.read\_csv("/content/Ice Cream Sales - temperatures.csv")

df3



account\_circle

[ ]

df3.corrwith(df3["Temperature"])

account\_circle

Temperature 1.000000

Ice Cream Profits 0.988446

dtype: float64

[ ]

plt.figure(figsize=(8,4))  
sns.scatterplot(data=df3, x='Temperature', y='Ice Cream Profits')  
plt.show()

account\_circle

[ ]

plt.figure(figsize=(8,4))  
sns.regplot(data=df3, x='Temperature', y='Ice Cream Profits', line\_kws={"color": "red"})  
plt.show()

account\_circle

[ ]

X=df3["Temperature"].to\_numpy()  
X

account\_circle

array([ 39, 40, 41, 42, 43, 43, 44, 44, 45, 45, 45, 46, 46,

47, 48, 48, 48, 48, 48, 48, 49, 49, 50, 50, 50, 50,

50, 51, 51, 52, 52, 52, 52, 52, 53, 53, 53, 53, 54,

54, 54, 54, 54, 54, 54, 54, 54, 55, 55, 55, 55, 56,

56, 57, 57, 58, 58, 58, 58, 58, 58, 59, 59, 59, 59,

59, 59, 59, 59, 60, 60, 60, 60, 60, 61, 61, 61, 61,

61, 61, 61, 61, 61, 62, 62, 62, 62, 63, 63, 63, 63,

63, 64, 64, 64, 64, 64, 64, 64, 64, 64, 65, 65, 65,

65, 65, 65, 65, 65, 65, 66, 66, 66, 66, 66, 66, 66,

66, 66, 66, 66, 66, 66, 66, 67, 67, 67, 67, 67, 67,

68, 68, 68, 68, 68, 68, 68, 68, 68, 68, 68, 68, 69,

69, 69, 69, 69, 69, 70, 70, 70, 70, 70, 70, 70, 70,

70, 71, 71, 71, 71, 71, 71, 71, 71, 71, 72, 72, 72,

72, 72, 72, 72, 72, 72, 72, 73, 73, 73, 73, 73, 73,

73, 73, 73, 74, 74, 74, 74, 74, 74, 74, 74, 74, 74,

74, 74, 75, 75, 75, 75, 75, 75, 76, 76, 76, 76, 76,

76, 76, 76, 76, 76, 76, 77, 77, 77, 77, 77, 77, 77,

77, 77, 77, 77, 77, 77, 77, 77, 78, 78, 78, 78, 78,

78, 78, 78, 78, 78, 78, 78, 79, 79, 79, 79, 79, 79,

79, 79, 79, 79, 79, 79, 80, 80, 80, 80, 80, 80, 80,

80, 80, 81, 81, 81, 81, 81, 81, 81, 81, 81, 82, 82,

82, 82, 82, 82, 82, 82, 82, 82, 82, 83, 83, 83, 83,

83, 83, 84, 84, 84, 84, 84, 84, 84, 84, 84, 84, 84,

84, 84, 85, 85, 85, 85, 85, 85, 85, 85, 85, 85, 85,

86, 86, 86, 86, 86, 86, 86, 86, 87, 87, 87, 87, 87,

87, 88, 88, 88, 89, 89, 89, 89, 90, 90, 90, 90, 90,

90, 90, 91, 91, 91, 92, 92, 92, 92, 93, 93, 94, 94,

95, 95, 95, 95, 96, 96, 96, 97, 98, 99, 99, 99, 101,

101])

[ ]

Y=df3["Ice Cream Profits"].to\_numpy()  
Y

account\_circle

array([13.17, 11.88, 18.82, 18.65, 17.02, 15.88, 19.07, 19.57, 21.62,

22.34, 19.23, 21.25, 19.81, 22.12, 24.22, 24.68, 23.78, 26.41,

25.01, 22.29, 27.81, 23.54, 22.89, 25.68, 27.29, 27.64, 27.31,

21.93, 32.18, 30.67, 28.05, 28.82, 27.87, 29.39, 32.6 , 31.62,

25.71, 28.48, 30.09, 33.58, 29.75, 31.94, 33.71, 28.37, 27.41,

27.99, 30.37, 27.68, 29.53, 33.91, 34.19, 33.22, 34.47, 30.89,

35.8 , 33.44, 36.79, 31.56, 35.13, 36.11, 32.39, 38.18, 29.69,

38.47, 37.74, 36.71, 32.29, 37.5 , 35.33, 35.06, 36.25, 40.25,

39.69, 40.95, 37.96, 38.1 , 38.21, 37.3 , 39.53, 37.42, 39.42,

38.16, 37.66, 39.04, 41.44, 40.19, 37.93, 50.17, 44.15, 41.58,

40.59, 39.17, 40.57, 40.28, 41.21, 44.85, 40.94, 40.14, 38.57,

44.07, 44.1 , 47.36, 45.38, 41.09, 43.78, 42.72, 42.1 , 43.28,

44.31, 42.71, 43.03, 42.16, 46.74, 47.68, 44.48, 47.52, 44.98,

45.07, 45.42, 47.36, 48.26, 51.75, 45.05, 40.65, 48.65, 45.26,

46.04, 44.85, 42.94, 50.62, 45.65, 49.37, 45.89, 50.74, 47.17,

49.6 , 41.68, 46.9 , 47.35, 47.73, 43.73, 47.47, 51.38, 41.74,

49.88, 47.78, 42.5 , 48.77, 49.46, 50.87, 49.12, 49.95, 50.31,

49.32, 52.67, 52.05, 48.82, 53.33, 54.59, 53.77, 49.6 , 52.17,

46.74, 53.04, 49.34, 55.04, 57.18, 51.26, 53.78, 51.55, 50.01,

53.59, 52.47, 48.96, 53.57, 50.79, 52.13, 52.42, 54.67, 51.82,

53.21, 54.4 , 55.01, 54.08, 53.97, 55.28, 54.36, 53.62, 50.65,

55.52, 58.61, 50.64, 54.28, 53.95, 53.44, 57.1 , 54.26, 55.34,

53.71, 57.84, 55.91, 58.62, 58.85, 52.84, 56.59, 59.43, 59.69,

53.83, 59.41, 53.17, 53.48, 59.94, 60.31, 60.33, 53.82, 53.07,

59.48, 54.1 , 56.33, 59.87, 60.75, 56.43, 60.86, 55.07, 58.39,

58.72, 57.52, 56.33, 57.47, 58.13, 60.46, 60.33, 60.89, 62.58,

61.22, 59.62, 58.31, 59.12, 57.93, 57.25, 62.2 , 59.7 , 64.82,

57.06, 62.52, 59.93, 61.71, 59.49, 67.42, 56.34, 59.69, 57.44,

64.63, 55.47, 61.22, 62.79, 59.91, 61.59, 63.46, 64.45, 65.42,

61.82, 64.36, 58.11, 59.47, 65.86, 61.52, 62.12, 64.23, 62.36,

62.32, 64.97, 66.15, 64.02, 63.41, 61.85, 65.49, 64.39, 66.06,

64.86, 62.85, 66.57, 65.54, 62.58, 63.29, 64.38, 60.78, 65.66,

66.61, 65.12, 63.13, 63.35, 65.4 , 65.41, 68.28, 64.1 , 66.26,

63.63, 67.58, 68.54, 65.2 , 67.93, 67.88, 69.71, 64.22, 61.82,

68.28, 62.99, 64.96, 65.99, 70.3 , 64.31, 69.59, 68.35, 69.66,

71.46, 69.9 , 69.19, 67.97, 64.85, 70.43, 68.48, 70.29, 65.19,

68. , 70.64, 69.67, 74.69, 69.78, 73.16, 71.51, 73.32, 74.09,

71.12, 67.58, 77.39, 75.11, 74.8 , 73.94, 75.94, 79.31, 81.81,

75.58, 78.2 , 75.6 , 75.04, 77.41, 79.76, 77.18, 80.94, 75.7 ,

78.2 , 80.75, 80.97, 80.98, 80.02, 82.83, 80.95, 82.5 , 84.12,

85.13, 87.08, 89.29, 81.91, 85.02])

[ ]

Xmean=X.mean()  
Xmean

account\_circle

71.98082191780821

[ ]

Ymean=Y.mean()  
print(Ymean)

account\_circle

52.10361643835618

[ ]

theta1=sum((X-Xmean)\*(Y-Ymean))/sum((X-Xmean)\*\*2)  
print(f"{theta1=}")

account\_circle

theta1=1.1920089289728377

[ ]

theta0 = Ymean - theta1\*Xmean  
print(f"{theta0=}")

account\_circle

theta0=-33.698166002474956

[ ]

x\_test=df3.sample(100,random\_state=42)["Temperature"].to\_numpy()  
print(x\_test)  
y\_test=df3.sample(100,random\_state=42)["Ice Cream Profits"].to\_numpy()  
print(y\_test)

account\_circle

[74 52 48 85 58 73 61 66 70 67 90 54 85 58 68 84 39 82 89 65 45 71 60 80

79 74 77 84 94 99 50 65 54 95 74 66 59 50 91 89 79 92 52 84 65 78 63 58

43 54 78 92 83 81 73 90 64 42 98 61 78 93 61 54 81 60 77 67 76 61 64 74

87 76 91 75 61 62 95 48 59 82 44 66 66 87 86 80 85 66 87 58 70 80 68 77

68 48 70 50]

[53.95 29.39 24.68 65.99 31.56 54.08 38.21 47.36 50.31 46.04 73.94 33.58

64.22 33.44 46.9 65.2 13.17 61.85 73.16 47.36 22.34 55.04 39.69 55.47

67.42 57.1 60.75 65.4 80.94 87.08 27.64 44.31 33.71 80.75 54.26 44.48

38.47 22.89 75.58 73.32 64.82 75.04 28.05 63.63 42.71 61.22 40.59 36.79

15.88 27.99 62.58 77.41 62.58 64.36 52.13 74.8 40.28 18.65 84.12 37.3

59.12 79.76 39.53 30.37 65.86 40.95 58.39 48.65 52.84 37.66 41.21 54.28

70.64 59.94 79.31 58.85 38.1 41.44 80.97 23.78 32.29 64.97 19.57 47.68

44.98 65.19 67.97 62.79 67.93 43.03 68.48 32.39 52.05 61.82 45.89 60.86

43.73 26.41 49.46 27.29]

[ ]

y\_pridect=theta0 +theta1\*x\_test  
  
arr1 = np.round(y\_pridect,decimals = 2)  
print(arr1)

account\_circle

[54.51 28.29 23.52 67.62 35.44 53.32 39.01 44.97 49.74 46.17 73.58 30.67

67.62 35.44 47.36 66.43 12.79 64.05 72.39 43.78 19.94 50.93 37.82 61.66

60.47 54.51 58.09 66.43 78.35 84.31 25.9 43.78 30.67 79.54 54.51 44.97

36.63 25.9 74.77 72.39 60.47 75.97 28.29 66.43 43.78 59.28 41.4 35.44

17.56 30.67 59.28 75.97 65.24 62.85 53.32 73.58 42.59 16.37 83.12 39.01

59.28 77.16 39.01 30.67 62.85 37.82 58.09 46.17 56.89 39.01 42.59 54.51

70.01 56.89 74.77 55.7 39.01 40.21 79.54 23.52 36.63 64.05 18.75 44.97

44.97 70.01 68.81 61.66 67.62 44.97 70.01 35.44 49.74 61.66 47.36 58.09

47.36 23.52 49.74 25.9 ]

[ ]

data={  
    "Asil qiymatlar": y\_test,  
    "Bashorat qiymatlar":arr1  
}  
df4=pd.DataFrame(data)  
df4

account\_circle

[ ]

MAE = np.sum(np.absolute(arr1-y\_test))/len(y\_test)  
print(f"{MAE=}")  
MAE = np.sum(np.absolute(arr1-y\_test))/len(y\_test)  
MAE

account\_circle

MAE=1.8577999999999997

1.8577999999999997

[ ]

RMSE = np.sqrt(np.sum((arr1-y\_test)\*\*2)/len(y\_test))  
print(f"{RMSE=}")

account\_circle

RMSE=2.3052726519871785

[ ]

#2-usul  
r=pd.read\_csv("/content/Ice Cream Sales - temperatures.csv")  
r

account\_circle

[ ]

r.isnull().sum()

account\_circle

Temperature 0

Ice Cream Profits 0

dtype: int64

[ ]

r.shape

account\_circle

(365, 2)

[ ]

from sklearn.model\_selection import train\_test\_split  
train\_set, test\_set = train\_test\_split(r, test\_size=0.10, random\_state=42)

[ ]

train\_set.shape

account\_circle

(328, 2)

[ ]

test\_set.shape

account\_circle

(37, 2)

[ ]

from sklearn import linear\_model  
LinearR = linear\_model.LinearRegression()

[ ]

x\_train = np.asanyarray(train\_set[['Temperature']])  
y\_train = np.asanyarray(train\_set[['Ice Cream Profits']])

[ ]

LinearR.fit(x\_train, y\_train)

account\_circle

[ ]

theta1 = LinearR.coef\_[0][0]  
theta0 = LinearR.intercept\_[0]  
print (f'{theta1=}')  
print (f'{theta0=}')

account\_circle

theta1=1.195487550758665

theta0=-34.022468641507395

[ ]

plt.figure(figsize=(10,6))  
sns.scatterplot(data=r, x='Temperature', y='Ice Cream Profits')  
plt.plot(x\_train, theta1\*x\_train + theta0, '-r')  
plt.xlabel("Temperature")  
plt.ylabel("Ice Cream Profits")

account\_circle

[ ]

x\_test = np.asanyarray(test\_set[['Temperature']])  
y\_test = np.asanyarray(test\_set[['Ice Cream Profits']])

[ ]

y\_predict = LinearR.predict(x\_test)  
y\_predict

account\_circle

array([[54.44361011],

[28.142884 ],

[23.36093379],

[67.59397317],

[35.3158093 ],

[53.24812256],

[38.90227195],

[44.87970971],

[49.66165991],

[46.07519726],

[73.57141093],

[30.5338591 ],

[67.59397317],

[35.3158093 ],

[47.27068481],

[66.39848562],

[12.60154584],

[64.00751052],

[72.37592338],

[43.68422216],

[19.77447114],

[50.85714746],

[37.7067844 ],

[61.61653542],

[60.42104787],

[54.44361011],

[58.03007277],

[66.39848562],

[78.35336113],

[84.33079888],

[25.7519089 ],

[43.68422216],

[30.5338591 ],

[79.54884868],

[54.44361011],

[44.87970971],

[36.51129685]])

[ ]

arr = np.round(y\_pridect,decimals = 2)  
print(arr)

account\_circle

[54.51 28.29 23.52 67.62 35.44 53.32 39.01 44.97 49.74 46.17 73.58 30.67

67.62 35.44 47.36 66.43 12.79 64.05 72.39 43.78 19.94 50.93 37.82 61.66

60.47 54.51 58.09 66.43 78.35 84.31 25.9 43.78 30.67 79.54 54.51 44.97

36.63 25.9 74.77 72.39 60.47 75.97 28.29 66.43 43.78 59.28 41.4 35.44

17.56 30.67 59.28 75.97 65.24 62.85 53.32 73.58 42.59 16.37 83.12 39.01

59.28 77.16 39.01 30.67 62.85 37.82 58.09 46.17 56.89 39.01 42.59 54.51

70.01 56.89 74.77 55.7 39.01 40.21 79.54 23.52 36.63 64.05 18.75 44.97

44.97 70.01 68.81 61.66 67.62 44.97 70.01 35.44 49.74 61.66 47.36 58.09

47.36 23.52 49.74 25.9 ]

arrow\_upwardarrow\_downward

link

comment

settings

delete

more\_vert

[ ]

from sklearn.metrics import mean\_absolute\_error, mean\_squared\_error

MAE = mean\_absolute\_error(y\_test, y\_predict)

RMSE = np.sqrt(mean\_squared\_error(y\_test, y\_predict))

print(f"{MAE=}")

print(f"{RMSE=}")



account\_circle

MAE=1.98778284492892

RMSE=2.53063048832881

https://colab.research.google.com/drive/1UHUUGiL-ojMXo2uqqyvTYTp-msziEtUt?usp=sharing