นายสิทธิพงษ์ แปลมูลตรี 62011212109

4. [Multiple linear regression]

Source code

```
import numpy as np
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
import matplotlib.pyplot as plt
import pandas as pd
plt.style.use('seaborn')

df = pd.read_csv('data/multi_run.csv')
print(df.head())

def caculator(data,plt,subplot):
    y_train = df['distance']
    x_train = df[[data]]

    model = LinearRegression()
    model.fit(x_train,y_train)
    # Y = b1X1 + b2X2 + ... + a
    b = model.coef_
    a = model.intercept_

    y_pred = model.predict(x_train)
    r2 = r2_score(y_train,y_pred)
    mse = mean_squared_error(y_train,y_pred)
    print("Linea Equation x0\t: Y = %.2f(%s)+%.2f"%(b,data,a))
    print('R2: %.2f'%(r2))
    print('MSE: %.2f'%(mse))

    plt.subplot(subplot)
    plt.scatter(x_train,y_train,color ="green")
    plt.plot(x_train,y_pred,color = "blue",label="Y =%.2f(%s) + %.2f"%(b,data,a))
    plt.xlabel('X = %s'%(data))
    plt.ylabel('Y = Distance')
    plt.legend()

y_train = df['distance']
x_train = df[['time','steep']]

model = LinearRegression()
model.fit(x_train,y_train)
# Y = b1X1 + b2X2 + ... + a
b = model.coef_
a = model.intercept_
y_pred = model.predict(x_train)
r2 = r2_score(y_train,y_pred)
mse = mean_squared_error(y_train,y_pred)


print("\nMultiple linear regression equation:\tY = %.2f(time) + %.2f(steep) + %.2f "%(b[0],b[1],a))
print('R2: %.2f'%(r2))
print('MSE: %.2f\n'%(mse))

x_new = np.array([[10,20],[15,2],[20,10]])
y_pred_new = model.predict(x_new)
print("Predicted response of X:")
print("10,20\t%.2f"%(y_pred_new[0]))
print("15,2\t%.2f"%(y_pred_new[1]))
print("20,10\t%.2f\n"%(y_pred_new[2]))

sp=121
xno=0
data = ['time','steep']
for i in range(len(data)):
    caculator(data[i],plt,sp)
    sp=sp+1
    xno=xno+1
plt.show()
```

นายสิทธิพงษ์ แปลมูลตรี 62011212109

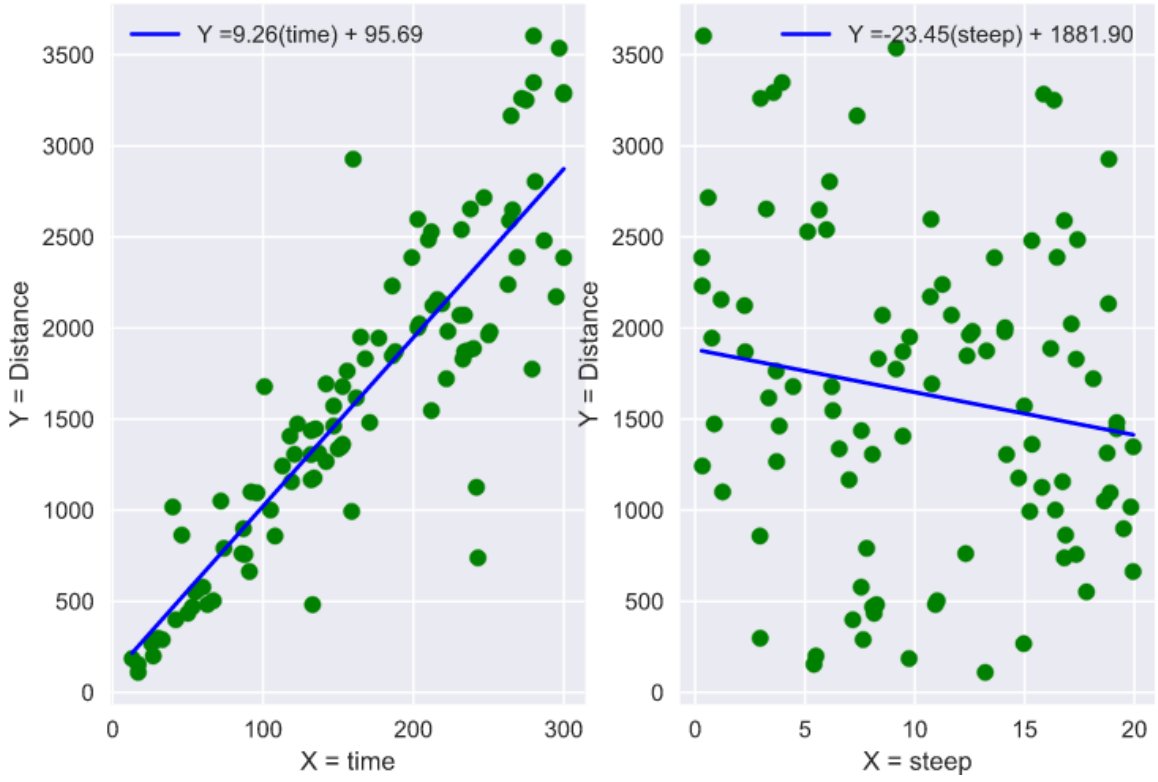
Output

```
   distance  time  steep
0    109.58    17  13.21
1    153.77    17   5.41
2    267.12    26  14.96
3    199.50    27   5.50
4    297.04    30   2.96

Multiple linear regression equation:    Y = 9.19(time) + -16.06(steep) + 274.36
R2: 0.78
MSE: 161332.22

Predicted response of X:
10,20   45.07
15,2    380.12
20,10   297.59

Linea Equation x0       : Y = 9.26(time)+95.69
R2: 0.77
MSE: 170443.57
Linea Equation x0       : Y = -23.45(steep)+1881.90
R2: 0.03
MSE: 723421.19
```



นายสิทธิพงษ์ แปลมูลตรี 62011212109

a. แสดง multiple linear regression equation

เส้นสมการ

```
Multiple linear regression equation:    Y = 9.19(time) + -16.06(steep) + 274.36
```

b.แสดงค่า R^2 และ MSE ของ Training data

```
y_train = df['distance']
x_train = df[['time','steep']]

model = LinearRegression()
model.fit(x_train,y_train)
# Y = b1X1 + b2X2 + ... + a
b = model.coef_
a = model.intercept_
y_pred = model.predict(x_train)
r2 = r2_score(y_train,y_pred)
mse = mean_squared_error(y_train,y_pred)
```

```
R2: 0.78
MSE: 161332.22
```

R^2 เท่ากับ 0.78 นั้นหมายความ ข้อมูลจะเข้าใกล้กับเส้นที่คาดการณ์อยู่ระดับความถูกต้องที่ 78% เลยทีเดียว

MSE = 161332.22 เป็นค่าเฉลี่ยของข้อมูลจริงและค่าคาดการณ์ ซึ่งเฉลี่ยออกมาแล้วมีค่าที่สูงมาก

ดังนั้น จากข้อมูลชุดนี้จึงไม่เหมาะในการหาด้วยวิธี Multiple regression equation

d. แยกการวิเคราะห์เป็น simple linear regression โดยวิเคราะห์เป็นtime --> distance และ Steep --> distance และแสดงกราฟของทั้ง 2 ตัวแปร

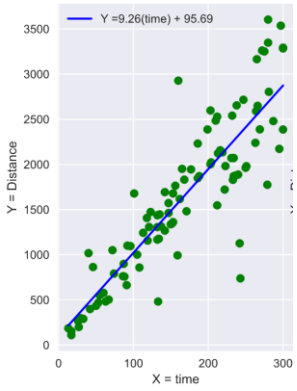
1. time--> distance

```
Linea Equation x0       : Y = 9.26(time)+95.69
R2: 0.77
MSE: 170443.57
```

R^2 เท่ากับ 0.77 นั้นหมายความ ข้อมูลจะเข้าใกล้กับเส้นที่คาดการณ์อยู่ระดับความถูกต้องที่ 77% เลยทีเดียว

MSE = 170443.57 เป็นค่าเฉลี่ยของข้อมูลจริงและค่าคาดการณ์ ซึ่งเฉลี่ยออกมาแล้วมีค่าที่สูงมาก

ดังนั้น จากข้อมูล time--> distance จึงไม่เหมาะในการหาด้วยวิธี simple linear regression



นายสิทธิพงษ์ แปลมูลตรี 62011212109

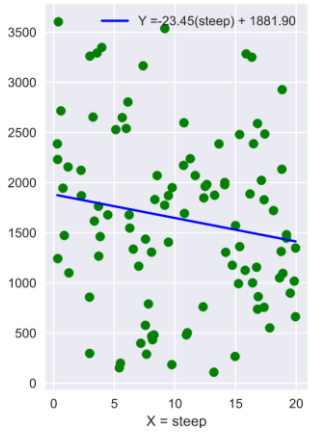
2. Steep--> distance

```
Linea Equation x0        : Y = -23.45(steep)+1881.90
R2: 0.03
MSE: 723421.19
```

R^2 เท่ากับ 0.03 นั้นหมายความ ข้อมูลจะเข้าใกล้กับเส้นที่คาดการณ์อยู่ระดับความถูกต้องที่ 0.03%

MSE = 170443.57 เป็นค่าเฉลี่ยของข้อมูลจริงและค่าคาดการณ์ ซึ่งเฉลี่ยออกมาแล้วมีค่าที่สูงมาก

ดังนั้น จากข้อมูล Steep--> distance จึงไม่เหมาะในการหาด้วยวิธี simple linear regression



e. ทำนายว่า ถ้าวิ่งด้วยข้อมูลต่อไปนี้จะได้ระยะทางเท่าใด โดยใช้ multiple regression

time	steep
10	20
15	2
20	10

จากสมการ

```
Multiple linear regression equation:    Y = 9.19(time) + -16.06(steep) + 274.36
```

เมื่อแทนค่าจะได้คำตอบตามนี้

```
Predicted response of X:
10,20   45.07
15,2    380.12
20,10   297.59
```