

머신러닝 이론과 실전 HW1

2021321148 조인식 (Insik Cho)

연습을 위해 교수님이 제공한 harris.dat 데이터를 이용했고, 첫 번째 열을 종속변수로 사용하라고 하신 점을 바탕으로 다음과 같이 진행했습니다.

```
data_name=input("Enter the name of data file [(ex) harris.dat] : ") # data name
coding_fm=int(input("Select the data coding format(1 = 'a b c' or 2 = 'a,b,c') : ")) # data separator
separator_fm={coding_fm ==1 : " ".get(True, ",")
res_pos=int(input("Enter the column position of the response variable : [from 1 to p] : "))
header=input("Does the data have column header? (y/n) : ")
if(header=="y") : trdata=pd.read_csv(data_name, sep=separator_fm) # loading data
else : trdata=pd.read_csv(data_name, sep=separator_fm, header=None) # loading data
out_name=input("Enter the output file name to export [(ex) result.txt] : ")
```

```
Enter the name of data file [(ex) harris.dat] : harris.dat
Select the data coding format(1 = 'a b c' or 2 = 'a,b,c') : 1
Enter the column position of the response variable : [from 1 to p] : 1
Does the data have column header? (y/n) : n
Enter the output file name to export [(ex) result.txt] : result.txt
```

전체적인 틀은 교수님이 예시로 제공한 코드를 바탕으로 진행했습니다.

```
# Extracting X and y
data = pd.DataFrame(trdata)
C = pd.DataFrame(np.ones(shape = (data.shape[0],)))
data = pd.concat([C,data], axis = 1, ignore_index= True)
X= data.drop([data.columns[res_pos]], axis=1 )
Y= data.iloc[:, res_pos ].values

#Implementing Multiple Linear regression

beta = np.linalg.inv(X.T@ X)@(X.T @ Y)
haty = X @ beta
residual = (Y- haty )
RSS = ((residual**2).mean()*(data.shape[0])
TSS = (((Y - (Y.mean()))**2).mean()*(data.shape[0])
Rsquared = 1- (RSS/TSS )
MSE = RSS / (data.shape[0] - data.shape[1]+1)
```

일반적으로 데이터들이 상수항을 포함하지는 않고, 주어진 예시 데이터도 상수항이 포함되지 않았기에 상수항을 따로 포함해 X (독립변수)를 만들고, y (종속변수)는 위에서 넣은 res_pos 열을 사용하도록 했습니다.

학부에서 배운 $\hat{\beta} = (X'X)^{-1}X'y$ 식을 이용해 OLS 추정치를 계산했고, 이를 바탕으로 $\hat{y} = X\hat{\beta}$ 를 계산해 잔차항 $\hat{u} = y - \hat{y}$ 을 구해 RSS (Residual sum of squares) 및 MSE , R^2 를 계산했습니다.

```

import sys
sys.stdout = open(out_name, 'w')

print("coefficients")
print("-----")

for i in range(beta.size):
    if i == 0 :
        print('constant:', beta[0])
    else:
        print('beta%d :'%i, beta[i])

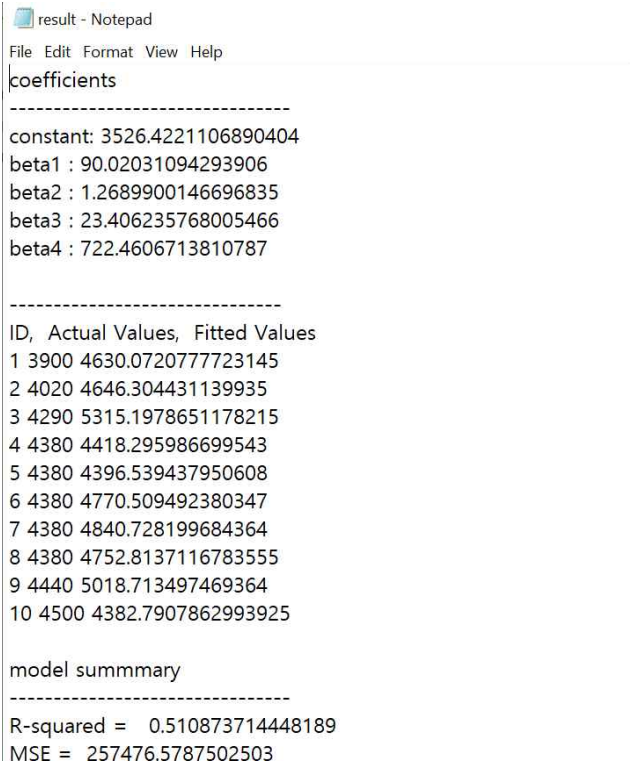
print(" ")
print("-----")
print("ID, Actual Values, Fitted Values")
for i in range(10):
    k = i + 1
    print( k, Y[i], haty[i])

print(" ")
print("model summary")
print("-----")
print("R-squared = ", Rsquared)
print("MSE = ", MSE)

```

sys module을 이용해 교수님이 기본 예시로 제시하신 result.txt 이름으로 결과물이 형성도록 만들었습니다.

harris.dat 데이터로 만든 결과물은 다음과 같습니다.



```

result - Notepad
File Edit Format View Help
coefficients
-----
constant: 3526.4221106890404
beta1 : 90.02031094293906
beta2 : 1.2689900146696835
beta3 : 23.406235768005466
beta4 : 722.4606713810787
-----
ID, Actual Values, Fitted Values
1 3900 4630.0720777723145
2 4020 4646.304431139935
3 4290 5315.1978651178215
4 4380 4418.295986699543
5 4380 4396.539437950608
6 4380 4770.509492380347
7 4380 4840.728199684364
8 4380 4752.8137116783555
9 4440 5018.713497469364
10 4500 4382.7907862993925
-----
model summary
-----
R-squared = 0.510873714448189
MSE = 257476.5787502503

```