텍스트, 서류, 영수증이(가) 표시된 사진

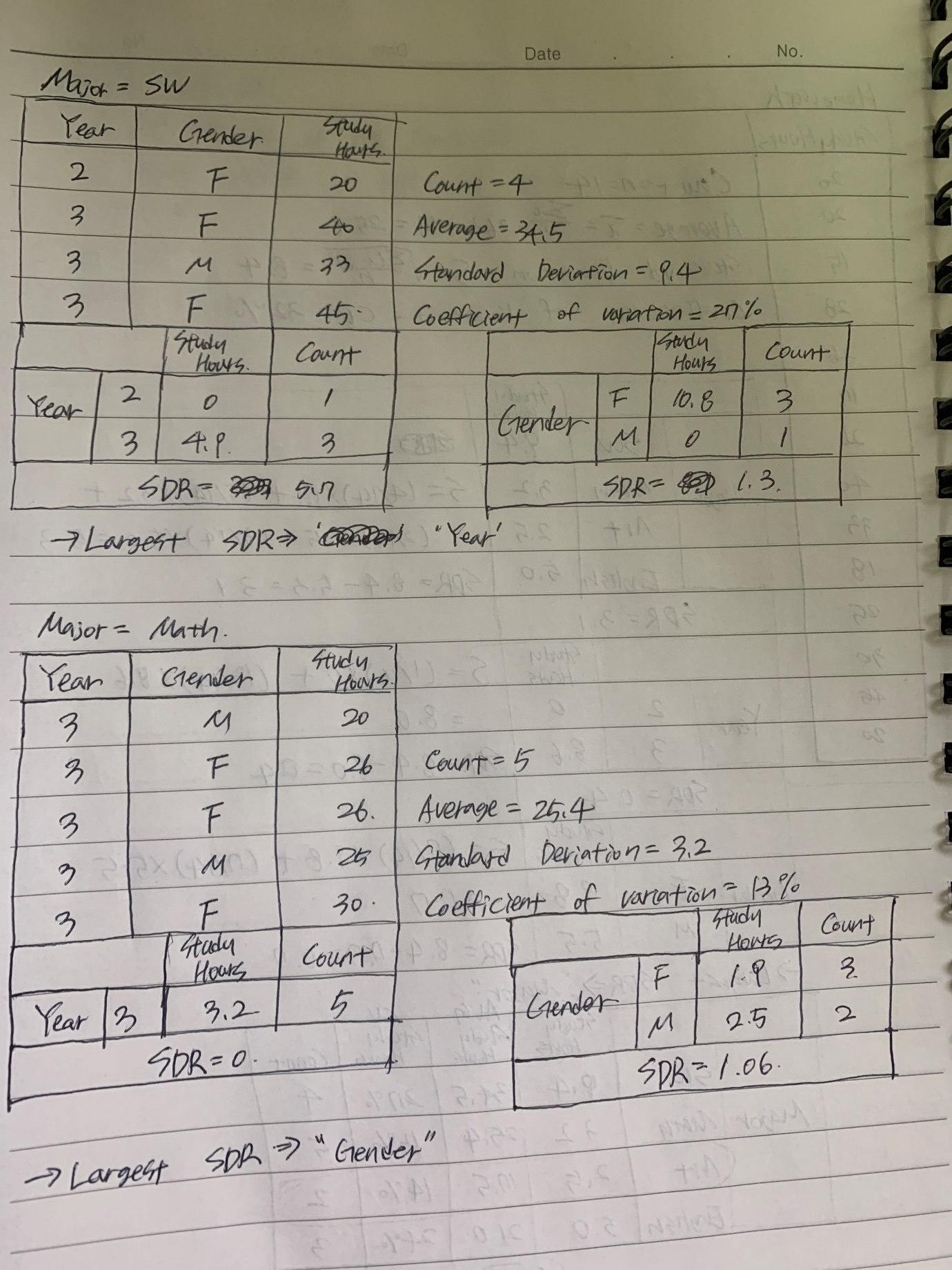
자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명텍스트이(가) 표시된 사진

자동 생성된 설명텍스트이(가) 표시된 사진

자동 생성된 설명

from matplotlib import pyplot as plt  
from sklearn.model\_selection import train\_test\_split  
from sklearn import datasets  
from sklearn.tree import DecisionTreeRegressor  
import pandas as pd  
import numpy as np  
  
dataset = np.array(  
[['SW', 2, 'F', 20],  
['Math', 3, 'M', 20],  
['Art', 3, 'F', 15],  
['English', 3, 'M', 28],  
['Math', 3, 'F', 26],  
['English', 3, 'M', 17],  
['Math', 3, 'F', 26],  
['SW', 3, 'F', 40],  
['SW', 3, 'M', 33],  
['English', 3, 'M', 18],  
['Math', 3, 'M', 25],  
['Math', 3, 'F', 30],  
['SW', 3, 'F', 45],  
['Art', 3, 'M', 20],  
])  
  
print(dataset)  
  
X = dataset[:, 1:2].astype(int)  
  
# print X  
print(X)  
  
y = dataset[:, 3].astype(int)  
  
# print y  
print(y)  
  
# create a regressor object  
regressor = DecisionTreeRegressor(random\_state=0)  
  
# fit the regressor with X and Y data  
regressor.fit(X, y)  
  
  
# arange for creating a range of values  
X\_grid = np.arange(min(X), max(X), 0.1)  
  
# reshape for reshaping the data  
X\_grid = X\_grid.reshape((len(X\_grid), 1))  
  
# scatter plot for original data  
plt.scatter(X, y, color='red')  
  
# plot predicted data  
plt.plot(X\_grid, regressor.predict(X\_grid), color='blue')  
  
  
# show the plot  
plt.show()  
  
# import export\_graphviz  
from sklearn.tree import export\_graphviz  
  
# export the decision tree to a tree.dot file  
# for visualizing the plot easily anywhere  
export\_graphviz(regressor, out\_file='tree.dot',  
 feature\_names=['HW1'])