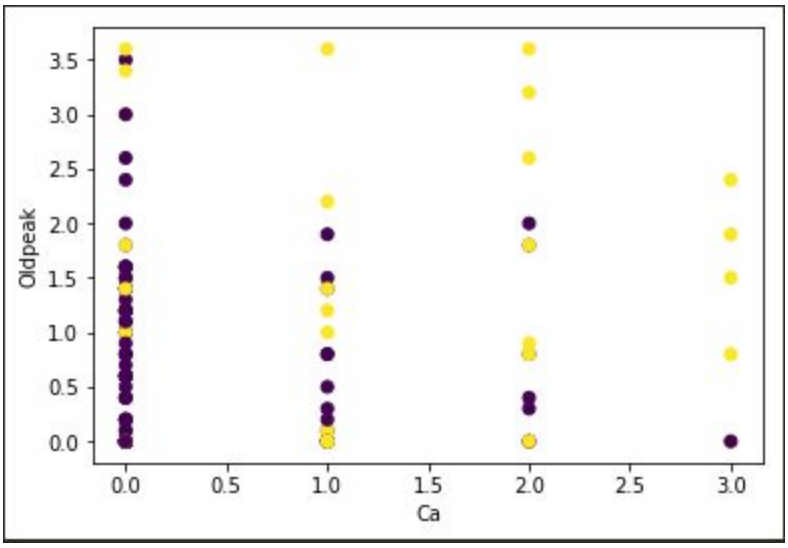
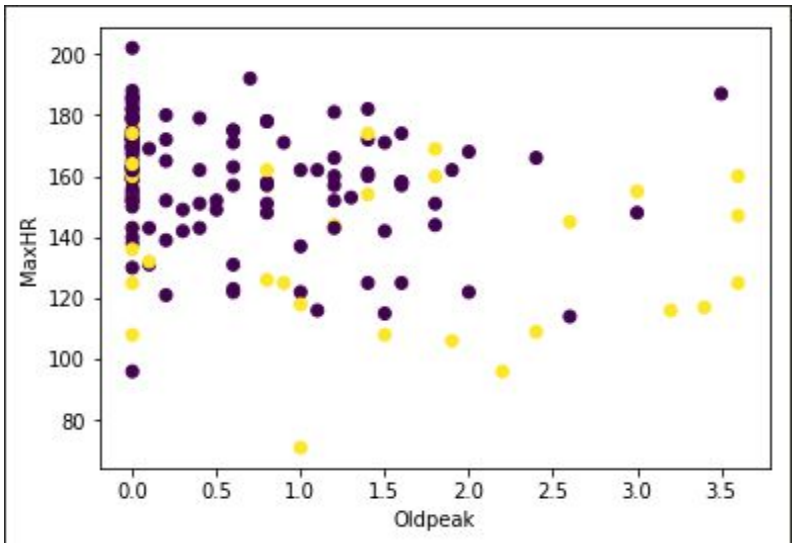
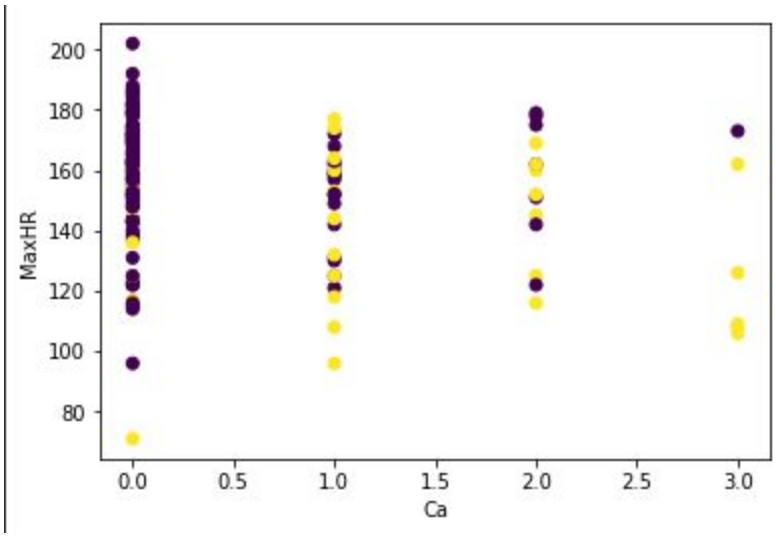
Edward Eustachon
epe245
HW5 Output

1) The axis labels will change based on the plot displayed so don't mind the last 2 lines

```
import numpy as np #general library, will come in handy later
import pandas as pd #another nice library for storing matrices, it rely's on numpy
import matplotlib.pyplot as plt #this library is for graphing things
from sklearn import svm #These libraries have the necessary models
from sklearn.model_selection import GridSearchCV, train_test_split

df = pd.read_csv('C:/Users/eddie/Desktop/School/ECO 348K Advanced Econometrics/Homework\HW5\Hearts_Dummy.csv')
df = df.dropna(how='any', axis=0)

#create subset of observations
X = df.loc[df['Thal_normal'] == 1, :'Thal_normal']
Y = df.loc[df['Thal_normal'] == 1, 'AHD_Yes':]

#scatter plots
plt.scatter(X['Ca'], X['Oldpeak'], c=Y['AHD_Yes'])
plt.scatter(X['Oldpeak'], X['MaxHR'], c=Y['AHD_Yes'])
plt.scatter(X['Ca'], X['MaxHR'], c=Y['AHD_Yes'])
plt.xlabel('Ca')
plt.ylabel('MaxHR')
```







No separating hyperplanes exist for any scatter plot

Edward Eustachon
epe245
HW5 Output

2)

```
#create subset of observations
X = df.loc[df['Thal_normal'] == 1, :'Thal_normal']
Y = df.loc[df['Thal_normal'] == 1, 'AHD_Yes':]

#create training set and test set from subset
Xtrain, Xtest, Ytrain, Ytest = train_test_split(X, Y, test_size=0.5, random_state=0)

#create test observation from assignment
obs = [[0, 1, 2.5, 150, 1]]

#begin model creation, fitting, and cross-validation
model = svm.SVC()
model.fit(Xtrain, Ytrain.values.ravel())
print()
print('------------------------------------------------------------')
print('Default parameters are C=1, kernel=RBF, and gamma=auto')
print('Training data score w/ default parameters:', model.fit(Xtrain, Ytrain.values.ravel()).score(Xtrain, Ytrain.values.ravel()))
print('Test data score w/ default parameters:', model.fit(Xtrain, Ytrain.values.ravel()).score(Xtest, Ytest.values.ravel()))
print('------------------------------------------------------------')
parameters = [
  {'C': [0.01, .1, 1, 10, 100], 'kernel': ['linear']},
  {'C': [0.01, .1, 1, 10, 100], 'gamma': [0.01, 0.1, 1, 10, 100], 'kernel': ['rbf']}]
clf = GridSearchCV(model, parameters, cv=5)
clf.fit(Xtrain, Ytrain.values.ravel())
print()
print('------------------------------------------------------------')
print('CV parameters are:', clf.best_params_)
print('Training data w/ CV parameters:', clf.best_score_)
print('Test data score w/ CV parameters:', clf.score(Xtest, Ytest.values.ravel()))
print('------------------------------------------------------------')

#use test observation on cross-validated model
model = svm.SVC(C=1, kernel='linear')
model.fit(X, Y.values.ravel())
yhat = model.predict(obs)
print()
print('------------------------------------------------------------')
print('Test observation is class:', yhat)
print('------------------------------------------------------------')
```

```
------------------------------------------------------------
Default parameters are C=1, kernel=RBF, and gamma=auto
Training data score w/ default parameters: 1.0
Test data score w/ default parameters: 0.7560975609756098
------------------------------------------------------------

------------------------------------------------------------
CV parameters are: {'C': 1, 'kernel': 'linear'}
Training data w/ CV parameters: 0.8780487804878049
Test data score w/ CV parameters: 0.7926829268292683
------------------------------------------------------------

------------------------------------------------------------
Test observation is class: [1]
------------------------------------------------------------
```

The test observation is classified to have heart disease.

C is the tolerance for which we allow observations to cross the margin or be on the wrong side. Lower values of C allow less error on the soft margin meaning the model will try to overfit the data resulting in low bias. As C increases, the bias will increase but there will be less variance.

In a sense, gamma determines how far an observation has influence. Lower values of gamma will result in data that has higher peaks to adjust for observations further away. As gamma increases, the bias will increase but there will be less variance.