PRAGYA VERMA – (101276204)

SEIS 763: 01 Machine Learning

Assignment #4

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**1. Load the patient data from “ML\_HW\_Data\_Patients.csv” file (same data file in**

**the last assignment).**

import numpy as np  
import pandas as pd  
import scipy.stats as stats  
from matplotlib import pyplot as plt  
from sklearn.linear\_model import LinearRegression, Lasso, LassoCV  
from sklearn.model\_selection import train\_test\_split  
  
#---------------------------------------------------------------------------------------------  
  
#1 - Load the patient data from “ML\_HW\_Data\_Patients.csv” file (  
df=pd.read\_csv('ML\_HW\_Data\_Patients.csv')  
df.info

**2. Use the following 7 variables Age, Gender, Height, Weight, Smoker, Location,**

**SelfAssessedHealthStatus to build a linear regression model to predict the**

**systolic blood pressure.**

df['Height']=stats.zscore(df['Height'])  
df['Weight']=stats.zscore(df['Weight'])  
df=df.drop(['Diastolic','LastName'], axis=1)  
df  
  
df['Age']=stats.zscore(df['Age'])  
Gender\_dummy=pd.get\_dummies(df.Gender)  
Smoker\_dummy=pd.get\_dummies(df.Smoker)  
Location\_dummy=pd.get\_dummies(df.Location)  
Self\_dummy=pd.get\_dummies(df.SelfAssessedHealthStatus)  
  
new\_df=pd.concat([df,Gender\_dummy,Smoker\_dummy,Location\_dummy,Self\_dummy],axis=1)  
new\_df.drop(labels=['Gender','Smoker','Location',"'Female'",0,'SelfAssessedHealthStatus',  
 "'St. Mary's Medical Center'","'Poor'"],axis=1,inplace=True)  
new\_df.rename(columns={1:'Smoker',"'Male'":'Male',"'county General Hospital'":'countyGenralHospital', "'VA Hospital'":'VAHospital',"'Excellent'":'Excellent',  
 "'Fair'":'Fair',"'Good'":'Good'},inplace=True)  
  
X= new\_df.drop(['Systolic'],axis=1)  
Y=new\_df['Systolic']  
  
X\_train,X\_test,y\_train,y\_test=train\_test\_split(X,Y,test\_size=0.033,random\_state=42)  
reg=LinearRegression()  
reg.fit(X\_train,y\_train)  
y\_predict=reg.predict(X\_test)

3. Use **\*\*lasso regression\*\*** with **\*\*10-fold cross-validation**\*\* to identify useful

predictors.

model=LassoCV(eps=1e-3,cv=10)  
model.fit(X,Y)  
alphas=model.alphas\_  
alpha=model.alpha\_  
mse\_path=model.mse\_path\_  
  
  
print('alpha:',model.alpha\_)  
print('Intercept:',model.intercept\_)  
print('Coefficients:',model.coef\_)  
  
coefs=[]  
for a in alphas:  
 clf=Lasso(alpha=a)  
 clf.fit(X,Y)  
 coefs.append(clf.coef\_)  
coefs\_path=np.mat(coefs)

**4. Which top \*\*TWO\*\* remaining predictors (with non-zero theta values) are you**

**going to select after the lasso analysis?**

**Ans:** Smoker and Height.

**5 - What is the lambda (l) value you choose in order to select the top two predictors you identified in the last question?**

**Ans:** I used 0. 902192616951158 as the lambda (l) value to choose the top two predictors.

**Ans:**

[0.902192616951158, array([ 0. , 0. , 0. , 0. , 5.93853558,

0. , -0. , 0. , -0. , 0. ]), 120.76089790447145]

**6. What are the q values for the two selected predictors at the lambda (l) value you identified in the last question?**

**Ans:**

Coffients: [ 0. 0. 0. 0. 5.93853558 0.

-0. 0. -0. 0. ]

7. **This question can be more difficult in Python; hence optional if you use**

**Python.** Plot a lasso plot with **OR** without cross-validation. Please have readable

tick labels on the X and Y axes in your plot for easy visualization and verification.



