import numpy as np

import pandas as pd

from sklearn import linear\_model

df = pd.read\_csv('Concrete\_samples.csv')

pd.options.display.max\_columns = None

# pd.options.display.max\_rows = None

dfcols = df.columns

print(df)

print(df.info())

from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()

df\_scaled = scaler.fit\_transform(df)

df\_scaled = pd.DataFrame(df\_scaled, columns = dfcols)

print(df\_scaled)

# SEPARATE X and y, and drop MPA

X = df.drop('MPA', axis=1)

y = df.MPA

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X,y,test\_size=.4, random\_state=0)

regr\_linear = linear\_model.LinearRegression()

regr\_Lasso = linear\_model.Lasso(alpha=0.001) # set ridge lasso

regr\_Ridge = linear\_model.Ridge(alpha=0.001) # set ridge alpha

# train the models

regr\_linear.fit(X\_train, y\_train)

regr\_Lasso.fit(X\_train, y\_train)

regr\_Ridge.fit(X\_train, y\_train)

# RUN TYPE

print('Run 20%-80%, alpha = 0.1')

# print the coefficients LINEAR

print('\n\nLinear model coefficients:')

for idx, col\_name in enumerate(X\_train.columns):

print('The coefficient for {} is {}'.format(col\_name, regr\_linear.coef\_[idx]))

print('The intercept is {}'.format(regr\_linear.intercept\_))

# print the coefficients LASSO

print('\n\nLasso model coefficients:')

for idx, col\_name in enumerate(X\_train.columns):

print('The coefficient for {} is {}'.format(col\_name, regr\_Lasso.coef\_[idx]))

print('The intercept is {}'.format(regr\_linear.intercept\_))

# print the coefficients RIDGE

print('\n\nRidge model coefficients:')

for idx, col\_name in enumerate(X\_train.columns):

print('The coefficient for {} is {}'.format(col\_name, regr\_Ridge.coef\_[idx]))

print('The intercept is {}'.format(regr\_linear.intercept\_))

# accuracy LINEAR

print('\n\nLinear model accuracy:')

print(regr\_linear.score(X\_train,y\_train))

print(regr\_linear.score(X\_test,y\_test))

# accuracy Lasso

print('\n\nLasso model accuracy:')

print(regr\_Lasso.score(X\_train,y\_train))

print(regr\_Lasso.score(X\_test,y\_test))

# accuracy Ridge

print('\n\nRidge model accuracy:')

print(regr\_Ridge.score(X\_train,y\_train))

print(regr\_Ridge.score(X\_test,y\_test))