Abelardo Riojas, Jeret McCoy, and Gustavo Flores
Applied Machine Learning Spring 2022
Homework 2

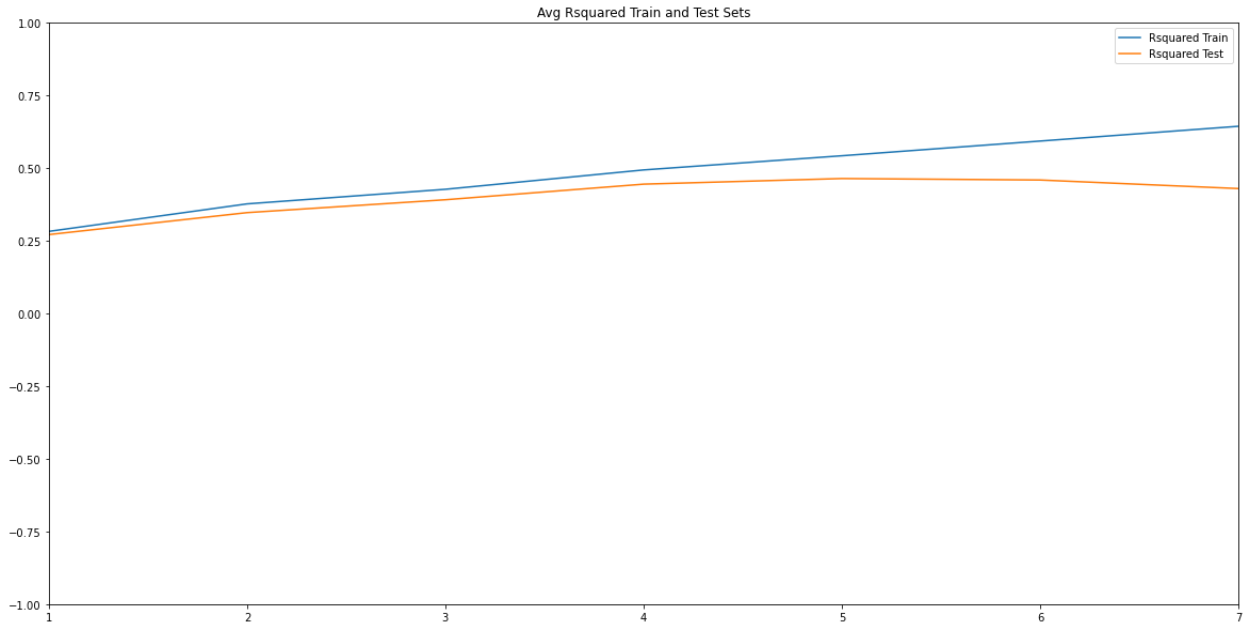
Part A.

a) Average Train R2 = 0.5296659962039154 Average Test R2 = 0.49031147068942094

```
import csv
import numpy as np
import sklearn.model_selection as ms
import matplotlib as plt
import sklearn.metrics as met

def read_csv(filename):
    with open(filename, newline='') as f_input:
        return [list(map(float, row)) for row in csv.reader(f_input)]

abalone = read_csv("abalone.csv")
abalone = np.array(abalone)

X = abalone[:,0:7]
y = abalone[:,7]

lamb = .0001 * np.ones(8)


r2_train = list()
r2_test = list()

for i in range(0, 10):
    trainX, testX, trainy, testy = ms.train_test_split(X, y, test_size=0.10)

    input_train_X = np.zeros([trainX[:,0].size, 8])
    input_train_X[:,0] = 1
    input_train_X[:,1:8] = trainX[:,:]

    N_theta = np.linalg.inv(input_train_X.T.dot(input_train_X) +
lamb).dot(input_train_X.T).dot(trainy)


    input_test_X = np.zeros([testX[:,0].size, 8])
    input_test_X[:,0] = 1
    input_test_X[:,1:8] = testX[:,:]
```

```
    p0 = input_train_X * N_theta
    p = input_test_X * N_theta


    pred_train_val = np.ones(p0[:,1].size)
    pred_train_val[:] = p0[:,0] + p0[:,1] + p0[:,2] + p0[:,3] + p0[:,4] + p0[:,5] + p0[:,6] + p0[:,7]


    pred_test_val = np.ones(p[:,1].size)
    pred_test_val[:] = p[:,0] + p[:,1] + p[:,2] + p[:,3] + p[:,4] + p[:,5] + p[:,6] + p[:,7]


    r2_train.append(met.r2_score(trainy, pred_train_val))
    r2_test.append(met.r2_score(testy, pred_test_val))

print("Average Train R2 = {} Average Test R2 =
{}".format(sum(r2_train)/len(r2_train),sum(r2_test)/len(r2_test)))
```

Part B.

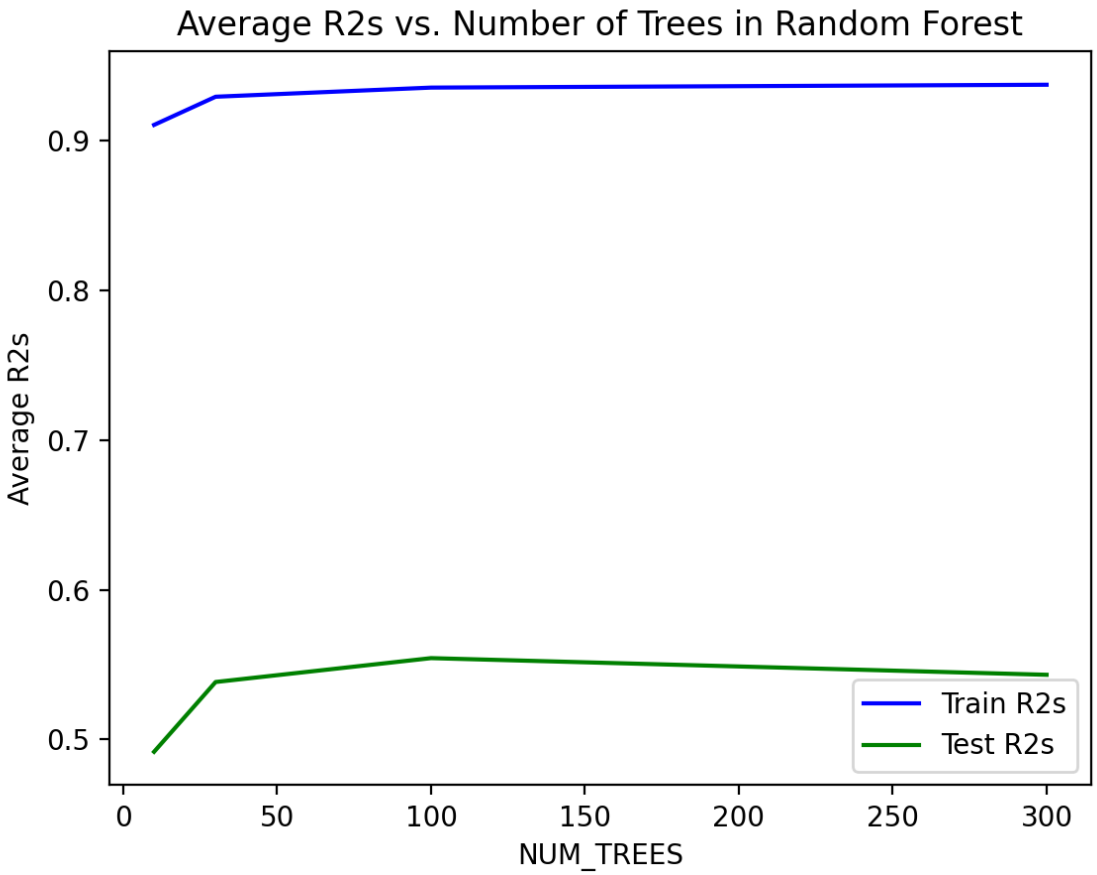


```
from csv import reader
import numpy as np
from random import shuffle
from sklearn.tree import DecisionTreeRegressor as tree
from matplotlib import pyplot as plt
path = 'abalone.csv'

openfile = open(path)
rf = reader(openfile)
data = list(rf)

abalone = []
for row in data:
    abalone.append([float(i) for i in row])

def prepare_dataset(ds):
    shuffle(ds)
    X = []
    Y = []
    X_test = []
    Y_test = []
    split = int(len(ds)*.9)
    for row in ds[:split]:
        X.append(row[:-1])
        Y.append(row[-1])
    for row in ds[split:]:
        X_test.append(row[:-1])
        Y_test.append(row[-1])
    return X, Y, X_test, Y_test

X, Y, X_test, Y_test = prepare_dataset(abalone)

avg_rsq_train = [0]*7
avg_rsq_test = [0]*7

depth = list(range(1,8))

for i in range(10):
    X, Y, X_test, Y_test = prepare_dataset(abalone)
    for k in depth:
        regr = tree(max_depth=k)
        regr.fit(X,Y)
        avg_rsq_train[k-1] += regr.score(X,Y)
        avg_rsq_test[k-1] += regr.score(X_test,Y_test)

avg_rsq_train = [i/10 for i in avg_rsq_train]
```

```
avg_rsq_test = [i/10 for i in avg_rsq_test]


plt.title('Avg Rsquared Train and Test Sets')
plt.plot(depth, avg_rsq_train)
plt.plot(depth, avg_rsq_test)
plt.xlim([1, 7])
plt.ylim([-1, 1])
plt.legend(['Rsquared Train', 'Rsquared Test'])
plt.show()
```

Part C.

c) Random forest regression with 10, 30, 100 and 300 trees. Report the average training and test R2in each case. (3 points)




```
NUM_TREES = 10 Average Train R2 = 0.9105566339006831 Average Test R2 = 0.49172730524524944
NUM_TREES = 30 Average Train R2 = 0.9293810574264636 Average Test R2 = 0.5383503216098813
NUM_TREES = 100 Average Train R2 = 0.9354821119627192 Average Test R2 = 0.5542994753856377
NUM_TREES = 300 Average Train R2 = 0.9373920835302446 Average Test R2 = 0.5431750222397621
```

Part C Code:

```
from sklearn.ensemble import RandomForestRegressor
import csv
from sklearn.metrics import r2_score
import numpy as np
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt


def read_csv(filename):
    with open(filename, newline='') as f_input:
        return [list(map(float, row)) for row in csv.reader(f_input)]

abalone = read_csv("abalone.csv")
abalone = np.array(abalone)
X = abalone[:,0:7]
y = abalone[:,7]


NUM_TREES_LIST = [10, 30, 100, 300]
trainAvg = list()
testAvg = list()
for NUM_TREES in NUM_TREES_LIST:
    r2test = list()
    r2train = list()
    for _ in range(10):
        trainX, testX, trainy, testy = train_test_split(X, y,
test_size=0.10)
        model = RandomForestRegressor(n_estimators=NUM_TREES)
        model.fit(trainX, trainy)
        trainpreds = model.predict(trainX)
        testpreds = model.predict(testX)
        r2train.append(r2_score(trainy, trainpreds))
        r2test.append(r2_score(testy, testpreds))
    print("NUM_TREES = {} Average Train R2 = {} Average Test R2 =
{}".format(NUM_TREES,sum(r2train)/len(r2train),sum(r2test)/len(r2test)))
    trainAvg.append(sum(r2train)/len(r2train))
    testAvg.append(sum(r2test)/len(r2test))

plt.plot(NUM_TREES_LIST,trainAvg, c='b', label = "Train R2s")
plt.plot(NUM_TREES_LIST,testAvg, c= 'g', label = "Test R2s")
```

```
plt.legend()
plt.xlabel("NUM_TREES")
plt.ylabel("Average R2s")
plt.title("Average R2s vs. Number of Trees in Random Forest")
plt.show()
```