```
#importing all necessary packages

import pandas as pd
import numpy as np
import random
import matplotlib.pyplot as plt
import itertools
import math
import statistics
%matplotlib inline
from scipy.stats import pearsonr
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression

def warn(*args, **kwargs):
  pass
import warnings
warnings.warn = warn


from google.colab import files

train_data = files.upload()
test_data = files.upload()

import io
df = pd.read_csv(io.BytesIO(train_data['hw2__question1_train.csv']))
testdf = pd.read_csv(io.BytesIO(test_data['hw2__question1_test.csv']))
df.head()
df.shape[0]
```

Choose Files hw2__question1_train.csv

- **hw2__question1_train.csv**(application/vnd.ms-excel) - 4133 bytes, last modified: 2/21/2020 - 100% done

```
Saving hw2__question1_train.csv to hw2__question1_train.csv
```

Choose Files hw2__question1_test.csv

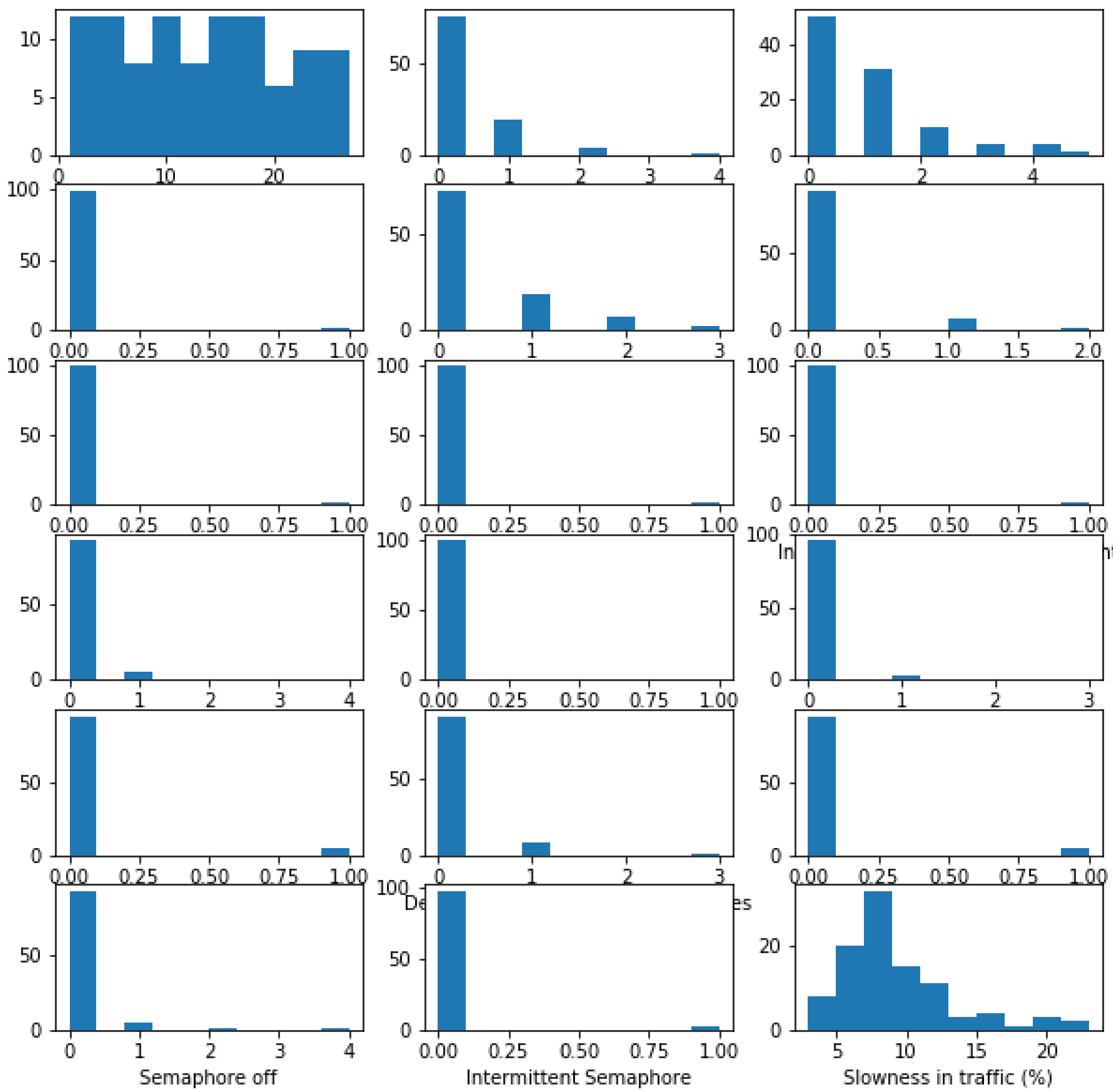
- **hw2__question1_test.csv**(application/vnd.ms-excel) - 1686 bytes, last modified: 2/21/2020 - 100% done

```
Saving hw2__question1_test.csv to hw2__question1_test.csv
```

Q1(i)

```
fig, axs = plt.subplots(6, 3, figsize=(10,10))

for i,keys in enumerate(df.keys()[:18]):
    #print(df[i])
    plt.subplot(6,3,i+1)
    plt.xlabel(keys)
    df3 = df[keys]
    plt.hist(df3)
```



Q1(ii)

```
for i in df.keys()[:17]:
    r,p = pearsonr(df[i],df['Slowness in traffic (%)'])
    print("Pearson's correlation for " + i + " is " + format(r))
```

```
Pearson's correlation for Hour (Coded) is 0.6707105739042726
Pearson's correlation for Immobilized bus is 0.15510125658984805
Pearson's correlation for Broken Truck is 0.14719368440730546
Pearson's correlation for Vehicle excess is -0.14646404002322472
Pearson's correlation for Accident victim is 0.1267848369630505
Pearson's correlation for Running over is -0.012205336668602058
Pearson's correlation for Fire vehicles is 0.18409716141808105
Pearson's correlation for Occurrence involving freight is 0.056958237786809605
Pearson's correlation for Incident involving dangerous freight is 0.03153045306055531
Pearson's correlation for Lack of electricity is 0.5737326666240945
Pearson's correlation for Fire is -0.04475290111820755
Pearson's correlation for Point of flooding is 0.45561796521556935
Pearson's correlation for Manifestations is -0.055721212206178296
Pearson's correlation for Defect in the network of trolleybuses is -0.16783420054885045
Pearson's correlation for Tree on the road is -0.07893838395875262
Pearson's correlation for Semaphore off is 0.42866617010993074
Pearson's correlation for Intermittent Semaphore is -0.13589889585548645
```

Q1(iii)

```
#building data matrix
X = []
df_X = pd.read_csv('hw2__question1_train.csv')
df_X.insert(0, "Bias", 1)#inserting bias
df_X = df_X.drop(['Slowness in traffic (%)'],axis = 1)#deleting the last colum
df_X.head()

for i in range((df_X.shape[0])):
    X.append(list(df_X.iloc[i, :]))

#print(len(X),len(X[0]))



X_trans = np.transpose(X) #18x100 matrix
x_dot_xtans = X_trans.dot(X) #18x18
X_t_X_inv = np.linalg.inv(x_dot_xtans)
w = X_t_X_inv.dot(X_trans).dot(df['Slowness in traffic (%)']) #weight vector
w
```

```
array([ 4.88163779,  0.26349179,  0.44771762, -0.14543905, -2.14512958,
       -0.03559434, -0.29705992,  4.26438321,  0.74721543,  0.38048825,
        2.09501546, -0.37111665,  1.81826542,  0.74351787, -0.56926429,
       -0.59828605,  0.51841252, -0.80646839])
```

Q1(iv)

```
df_X_test = pd.read_csv('hw2__question1_test.csv')
df_X_test.insert(0, "Bias", 1)#inserting bias
```

```
df_X_test = df_X_test.drop(['Slowness in traffic (%)'],axis = 1)
y_pred = df_X_test.dot(w)
#print(y_pred)
r,_ = pearsonr(y_pred,testdf['Slowness in traffic (%)'])

rss = 0

for i in range(len(y_pred)):
    rss += (y_pred[i] - testdf['Slowness in traffic (%)'][i])**2
    #print(y_pred[i],testdf['Slowness in traffic (%)'][i])

print("Pearson’s correlation coefficient = "+ format(r))
print("Rss = " + format(rss))
```

```
Pearson’s correlation coefficient = 0.8197758586024665
Rss = 501.8642602243592
```

Q1(v)

```
pearsons_coeff_dict ={}
for i,j in enumerate(df.keys()[:17]):
    for k,m in enumerate(df.keys()[:17]):
        if((j!=m) and (i<k)):
          print(j,m,pearsonr(df[j],df[m])[0])
          pearsons_coeff_dict[(j,m)] = pearsonr(df[j],df[m])[0]

for i in pearsons_coeff_dict.keys():
    if pearsons_coeff_dict[i] > 0.6:
      print(i,pearsons_coeff_dict[i])
```

```
Hour (Coded) Immobilized bus 0.0853377955041556
Hour (Coded) Broken Truck 0.22391505607501003
Hour (Coded) Vehicle excess -0.16250899875150268
Hour (Coded) Accident victim 0.18841399536931955
Hour (Coded) Running over 0.04222047516583157
Hour (Coded) Fire vehicles 0.156136096839679
Hour (Coded) Occurrence involving freight 0.023367307010019985
Hour (Coded) Incident involving dangerous freight -0.0031864509559118175
Hour (Coded) Lack of electricity 0.28388585372563585
Hour (Coded) Fire -0.043017087904809514
Hour (Coded) Point of flooding 0.2344892677080247
Hour (Coded) Manifestations -0.15274511507206298
Hour (Coded) Defect in the network of trolleybuses -0.17500014544566006
Hour (Coded) Tree on the road -0.04970277553932209
Hour (Coded) Semaphore off 0.19327301180300366
Hour (Coded) Intermittent Semaphore -0.18381227527775795
Immobilized bus Broken Truck 0.221496644375117
Immobilized bus Vehicle excess -0.048427983228424176
Immobilized bus Accident victim -0.024010860336515324
Immobilized bus Running over 0.182776581862117
Immobilized bus Fire vehicles 0.10779131750842799
Immobilized bus Occurrence involving freight 0.10779131750842799
Immobilized bus Incident involving dangerous freight 0.26401061824528
Immobilized bus Lack of electricity 0.1005048728521751
Immobilized bus Fire -0.04842798322842417
Immobilized bus Point of flooding 0.006378283699682306
Immobilized bus Manifestations 0.17473164531508992
Immobilized bus Defect in the network of trolleybuses 0.10131202263522487
Immobilized bus Tree on the road 0.03209356750685326
Immobilized bus Semaphore off 0.05067015001831877
Immobilized bus Intermittent Semaphore 0.04218983702383181
Broken Truck Vehicle excess -0.07537783614444092
Broken Truck Accident victim 0.3952955627914515
Broken Truck Running over 0.1507556722888818
Broken Truck Fire vehicles 0.014357683075131588
Broken Truck Occurrence involving freight 0.10409320229470409
Broken Truck Incident involving dangerous freight 0.10409320229470409
Broken Truck Lack of electricity 0.11420215254830426
Broken Truck Fire 0.014357683075131588
Broken Truck Point of flooding 0.07746351473264397
Broken Truck Manifestations 0.03277367626722311
Broken Truck Defect in the network of trolleybuses 0.10695420527776878
Broken Truck Tree on the road -0.04916051440083466
Broken Truck Semaphore off 0.19696861341086808
Broken Truck Intermittent Semaphore -0.04336734693877552
Vehicle excess Accident victim -0.055548961815628996
Vehicle excess Running over -0.030303030303031
Vehicle excess Fire vehicles -0.010101010101010097
Vehicle excess Occurrence involving freight -0.010101010101010097
Vehicle excess Incident involving dangerous freight -0.010101010101010097
Vehicle excess Lack of electricity -0.024857885581689738
Vehicle excess Fire -0.010101010101010097
Vehicle excess Point of flooding -0.017674908041006725
Vehicle excess Manifestations -0.023057148795535817
Vehicle excess Defect in the network of trolleybuses -0.02782172303192223
Vehicle excess Tree on the road -0.023057148795535817
Vehicle excess Semaphore off -0.022666155653645097
```

```
Vehicle excess Intermittent Semaphore -0.014357683075131602
Accident victim Running over 0.004272997062740695
Accident victim Fire vehicles 0.08688427360906073
Accident victim Occurrence involving freight -0.055548961815628996
Accident victim Incident involving dangerous freight 0.08688427360906073
Accident victim Lack of electricity 0.11366808307501351
Accident victim Fire -0.055548961815628996
Accident victim Point of flooding 0.0274155133420664
Accident victim Manifestations -0.06177400193221929
Accident victim Defect in the network of trolleybuses 0.06098653822823959
Accident victim Tree on the road -0.06177400193221928
Accident victim Semaphore off 0.10779673959972517
Accident victim Intermittent Semaphore -0.0789578845012374
Running over Fire vehicles -0.030303030303030310
Running over Occurrence involving freight -0.03030303030303031
Running over Incident involving dangerous freight -0.03030303030303031
Running over Lack of electricity -0.021306759070019776
Running over Fire -0.030303030303030318
Running over Point of flooding 0.03534981608201347
Running over Manifestations 0.06917144638660751
Running over Defect in the network of trolleybuses 0.06828968380562729
Running over Tree on the road 0.0691714463866075
Running over Semaphore off -0.06799846696093534
Running over Intermittent Semaphore 0.1722921969015793
Fire vehicles Occurrence involving freight -0.010101010101010097
Fire vehicles Incident involving dangerous freight -0.010101010101010097
Fire vehicles Lack of electricity -0.024857885581689738
Fire vehicles Fire -0.010101010101010097
Fire vehicles Point of flooding -0.017674908041006725
Fire vehicles Manifestations -0.023057148795535817
Fire vehicles Defect in the network of trolleybuses -0.02782172303192223
Fire vehicles Tree on the road -0.023057148795535817
Fire vehicles Semaphore off -0.022666155653645097
Fire vehicles Intermittent Semaphore -0.014357683075131598
Occurrence involving freight Incident involving dangerous freight -0.010101010101010097
Occurrence involving freight Lack of electricity -0.024857885581689734
Occurrence involving freight Fire -0.010101010101010097
Occurrence involving freight Point of flooding -0.017674908041006725
Occurrence involving freight Manifestations 0.4380858271151806
Occurrence involving freight Defect in the network of trolleybuses -0.02782172303192223
Occurrence involving freight Tree on the road -0.023057148795535817
Occurrence involving freight Semaphore off 0.18338980483403766
Occurrence involving freight Intermittent Semaphore -0.014357683075131602
Incident involving dangerous freight Lack of electricity -0.024857885581689734
Incident involving dangerous freight Fire -0.010101010101010097
Incident involving dangerous freight Point of flooding -0.017674908041006732
Incident involving dangerous freight Manifestations 0.4380858271151806
Incident involving dangerous freight Defect in the network of trolleybuses -0.0278217230
Incident involving dangerous freight Tree on the road -0.023057148795535817
Incident involving dangerous freight Semaphore off -0.022666155653645097
Incident involving dangerous freight Intermittent Semaphore -0.014357683075131602
Lack of electricity Fire -0.024857885581689734
Lack of electricity Point of flooding 0.5261032250786362
Lack of electricity Manifestations -0.056742044693343034
Lack of electricity Defect in the network of trolleybuses -0.0684673315734855
Lack of electricity Tree on the road -0.05674204469334305
Lack of electricity Semaphore off 0.6686336386559811
Lack of electricity Intermittent Semaphore -0.03533326266687868
```

```
Fire Point of flooding -0.017674908041006725
Fire Manifestations -0.023057148795535817
Fire Defect in the network of trolleybuses -0.02782172303192223
Fire Tree on the road -0.023057148795535817
Fire Semaphore off -0.022666155653645097
Fire Intermittent Semaphore -0.014357683075131602
Point of flooding Manifestations -0.04034576548024156
Point of flooding Defect in the network of trolleybuses -0.04868289321702686
Point of flooding Tree on the road -0.040345765480241574
Point of flooding Semaphore off 0.3809917281970773
Point of flooding Intermittent Semaphore -0.025123302075452113
Manifestations Defect in the network of trolleybuses 0.2828969169935907
Manifestations Tree on the road 0.15789473684210523
Manifestations Semaphore off 0.04233197080583852
Manifestations Intermittent Semaphore -0.032773676267223106
Defect in the network of trolleybuses Tree on the road 0.05196065822331256
Defect in the network of trolleybuses Semaphore off -0.06243053897462107
Defect in the network of trolleybuses Intermittent Semaphore -0.039546092707746464
Tree on the road Semaphore off -0.05173907542935821
Tree on the road Intermittent Semaphore -0.03277367626722312
Semaphore off Intermittent Semaphore -0.03221791446125724
('Lack of electricity', 'Semaphore off') 0.6686336386559811
```

Q1(vi)

```
mu = np.mean(np.hstack((df['Slowness in traffic (%)'],testdf['Slowness in traffic (%)'])))
mu
```

```
9.622222222222222
```

```
y_new_train = (df['Slowness in traffic (%)'] > mu).astype(int)
y_new_test = (testdf['Slowness in traffic (%)'] > mu).astype(int)
```

```
X_train = StandardScaler().fit_transform(df_X)
X_test = StandardScaler().fit_transform(df_X_test)
clf = LogisticRegression(penalty = 'none').fit(df_X,y_new_train)
y_logis_pred = clf.predict(df_X_test)
count = 0
for i,val in enumerate(y_new_test):
    if (val == y_logis_pred[i]):
        count += 1
acc = count/len(y_new_test)
acc
```

```
0.6285714285714286
```

Q1(vii)

```
acc_dict = {}
c_list = [1000,100,10,1,0.1,0.01,0.001]

for k in c_list:
  clf = LogisticRegression(penalty = 'l2',C = k).fit(df_X,y_new_train)
  y_logis_pred = clf.predict(df_X_test)
  count = 0
  for i,val in enumerate(y_new_test):
      if (val == y_logis_pred[i]):
          count += 1
  acc = count/len(y_new_test)
  acc_dict[k] = acc

acc_dict
```

```
{0.001: 0.4857142857142857,
 0.01: 0.7142857142857143,
 0.1: 0.7142857142857143,
 1: 0.6857142857142857,
 10: 0.6857142857142857,
 100: 0.6285714285714286,
 1000: 0.6285714285714286}
```

```
from sklearn.model_selection import cross_val_score
c_list = [1000,100,10,1,0.1,0.01,0.001]
score_list = []
for i in c_list:
  clf = LogisticRegression(penalty = 'l2',C = i).fit(df_X,y_new_train)
  scores = cross_val_score(clf, df_X,y_new_train , cv=5)
  print(scores,np.mean(scores))
```

```
[0.8  0.65 0.85 0.95 0.65] 0.78
[0.8  0.65 0.85 0.95 0.65] 0.78
[0.85 0.65 0.85 0.95 0.65] 0.7899999999999999
[0.85 0.7  0.85 0.95 0.7 ] 0.8099999999999999
[0.85 0.65 0.85 1.   0.7 ] 0.8099999999999999
[0.85 0.7  0.85 1.   0.7 ] 0.82
[0.8  0.75 0.8  0.8  0.7 ] 0.7700000000000001
```

```
clf = LogisticRegression(penalty = 'l2',C = 0.01).fit(df_X,y_new_train)
y_logis_pred = clf.predict(df_X_test)
count = 0
for i,val in enumerate(y_new_test):
    if (val == y_logis_pred[i]):
        count += 1
count/len(y_new_test)
```

```
0.7142857142857143
```