Bubble Slot

```
# bubble chart using matplotlib and pandas

# place legend outside plot


import pandas

import matplotlib.pyplot as plt

import matplotlib.patches as mpatches


%matplotlib inline


pandas.set_option('max_columns', 10)


# put in path/filename (see csv data below)

# data can also be found in data section of world bank site

df = pandas.read_csv('C:/Users/Ubaid-ur-Rehman/Desktop/Sem06/data science/project/Project
Task/bubble slot/ad12.CSV')

plt.scatter(x=df['AA1'],

y=df['AA3'],

s=df['AA2']/0.5,

alpha=0.1,

c=df['BubbleColor'])

# chart title, axis labels

plt.title('AA1, Advertisement (bubble size) Colors (cyan=add, purple=nonadd)')

plt.xlabel('AA1')

plt.ylabel('AA3')
```

KNN 10 Fold

```
import pandas as pd
```

```
#read in the data using pandas

UbaidFile = pd.read_csv('C:/Users/Ubaid-ur-Rehman/Desktop/Sem06/data science/project/Project Task
final/Speraman/ad13.CSV')



#check data has been read in properly

UbaidFile.head()


#create a dataframe with all training data except the target column

X = UbaidFile.drop(columns=['AA1559'])



#check that the target variable has been removed

X.head()



#separate target values

y = UbaidFile['AA1559'].values



#view target values

y[0:5]


from sklearn.model_selection import train_test_split


#split dataset into train and test data

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=1, stratify=y)
```

```
from sklearn.neighbors import KNeighborsClassifier
# Create KNN classifier
knn = KNeighborsClassifier(n_neighbors = 3)
# Fit the classifier to the data
knn.fit(X_train,y_train)

#show first 5 model predictions on the test data
knn.predict(X_test)[0:5]



#check accuracy of our model on the test data
knn.score(X_test, y_test)

#check accuracy of our model on the test data
knn.score(X_test, y_test)

from sklearn.model_selection import cross_val_score
import numpy as np



#create a new KNN model
knn_cv = KNeighborsClassifier(n_neighbors=3)

#train model with cv of 5
cv_scores = cross_val_score(knn_cv, X, y, cv=10)

#print each cv score (accuracy) and average them
print(cv_scores)
print('cv_scores mean:{}'.format(np.mean(cv_scores)))


knn_cv.fit(X_train, y_train)

y_pred = knn_cv.predict(X_test)



from sklearn.metrics import classification_report, confusion_matrix, accuracy_score

result = confusion_matrix(y_test, y_pred)



print("Confusion Matrix:")

print(result)


result1 = classification_report(y_test, y_pred)

print("Classification Report:",)

print (result1)



result2 = accuracy_score(y_test,y_pred)

print("Accuracy:",result2)
```

Speraman

```
#pandas (all lowercase) is a popular Python-based data analysis toolkit which can be imported using import pandas as pd.

#It presents a diverse range of utilities, ranging from parsing multiple file formats to converting an

#entire data table into a NumPy matrix array
```

```
import pandas as pd


#the numpy package is bound to the local variable numpy. The import as syntax simply allows you to bind the import to the

#local variable name of your choice (usually to avoid name collisions, shorten verbose module names, or standardize access to modules with compatible APIs).

import numpy as np


#flow maintain if error occur in in middle of

#code then this show at the end of execution

from scipy.stats import chi2_contingency


from scipy.stats import spearmanr



ubaidFile = pd.read_csv('C:/Users/Ubaid-ur-Rehman/Desktop/Sem06/data science/project/Project Task final/Speraman/ad13.CSV')



import matplotlib.pyplot as plt


fig, ax = plt.subplots()

ax.scatter(ubaidFile.iloc[:,0],ubaidFile.iloc[:,1])

ax.set_xlabel('AA1')

ax.set_ylabel('AA2')

ax.set_title('Advertise Website')

plt.show()


#calculate Spearman Rank correlation and corresponding p-value

rho = spearmanr(ubaidFile['AA1'], ubaidFile['AA2'])
```

```
#print Spearman rank correlation and p-value

print(rho)
```