**ASSIGNMENT 4**

**Assumptions**

1. Header is assumed to be everything above the first blank line.
2. Stop words present in the NLTK corpus are considered.

**Pre-processing Steps**

**Removal of Header**

(All the lines before the first blank line are removed)

**Removal of Punctuation marks, comma, etc**

(They are removed through regular expression)

**Tokenization**

(Tokens are formed using word\_tokenize and special symbols are removed)

**Removal of Stop Words**

(Stop words are removed using NLTK stop words)

**Normalization**

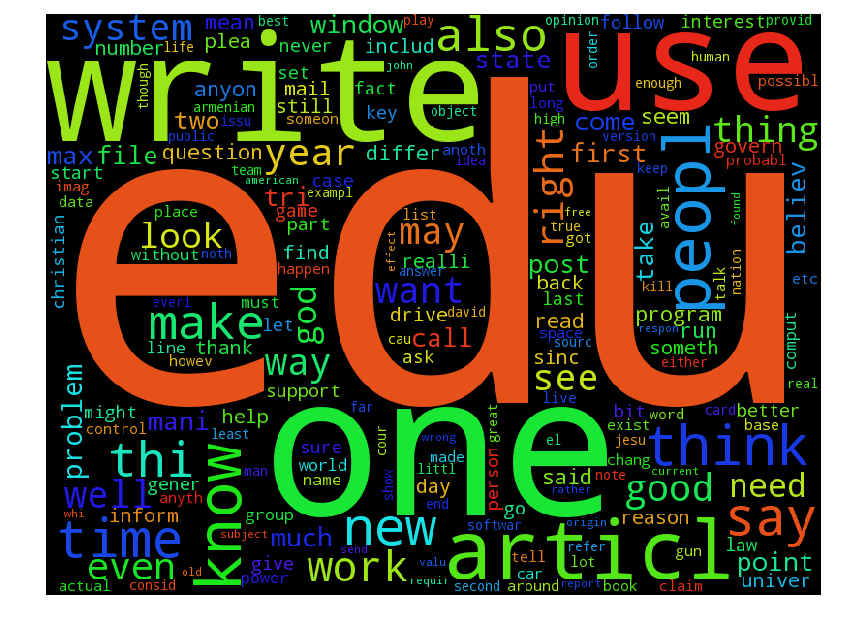
(All token are converted into lower case)

**Stemming**

(Stemming is performed using Porter algorithm to get the root word)

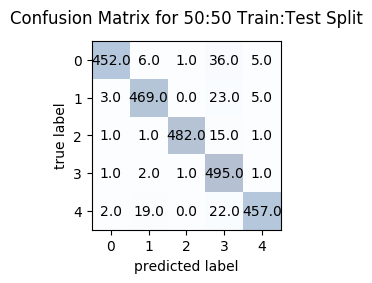
**Number of Documents:** 5000

**WordCloud**



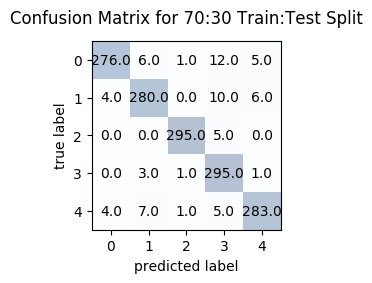
**Rocchio Classification Algorithm**

**50:50 Train Test Split**



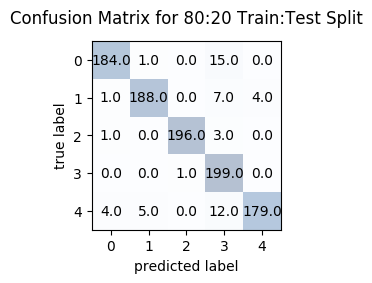
**Accuracy = 94.2%**

**70:30 Train Test Split**



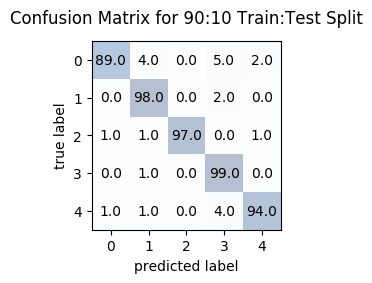
**Accuracy= 95.2666666667%**

**80:20 Train Test Split**

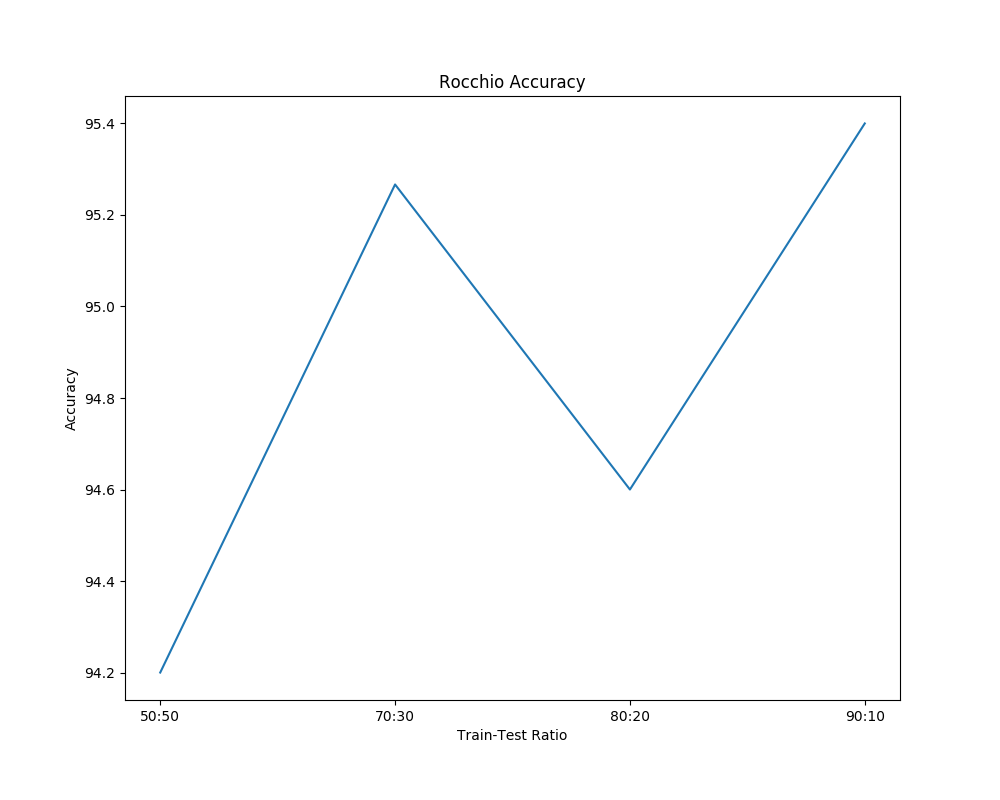


**Accuracy = 94.6%**

**90:10 Train Test Split**



**Accuracy= 95.4%**

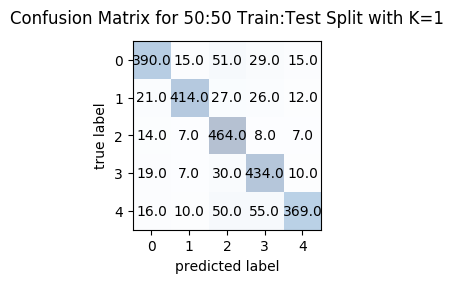


**Accuracy vs Train-Test Ratio**

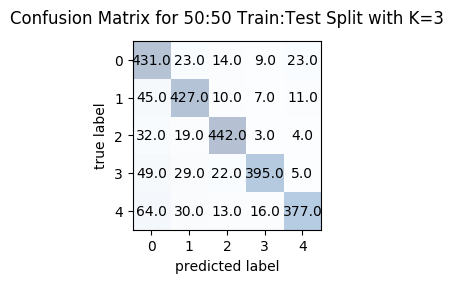
From the graph, we can infer that the accuracy of the Rocchio classification algorithm increases with the increase in the Train Ratio (Train Data) (decrease in Test Ratio) (with a slight decrease in 80:20 ratio).

**KNN Classification Algorithm**

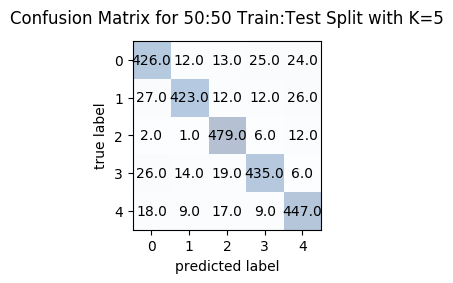
**50:50 Train Test Split**



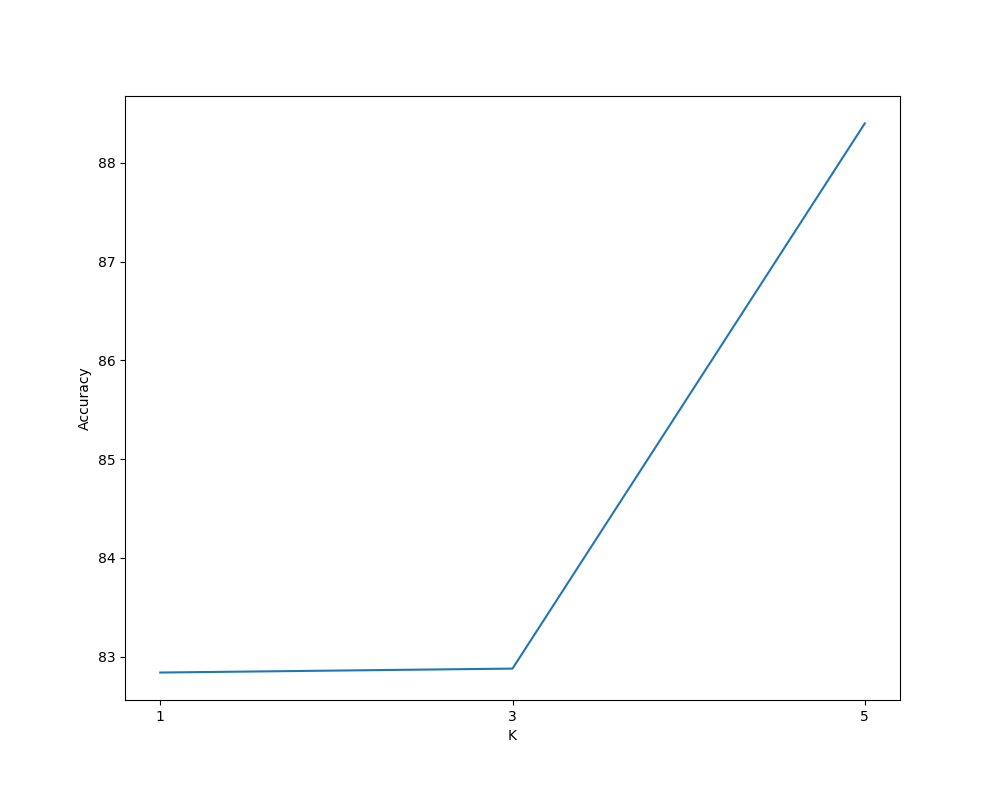
**Accuracy= 82.84%**



**Accuracy= 82.88%**



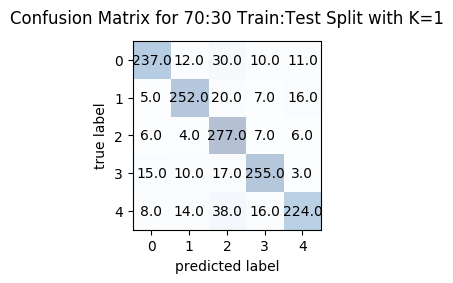
**Accuracy= 88.4%**



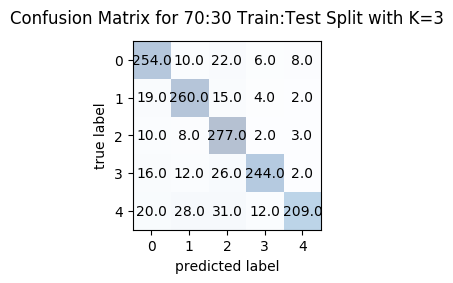
**K vs Accuracy**

From the graph, we can infer that the accuracy of the KNN increases with the increase in the K Value.

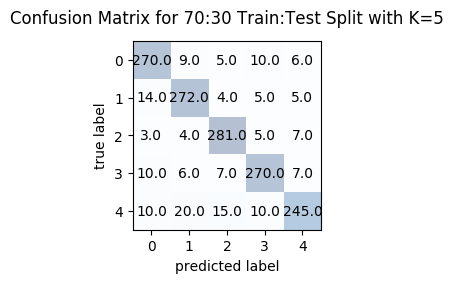
**70:30 Train Test Split**



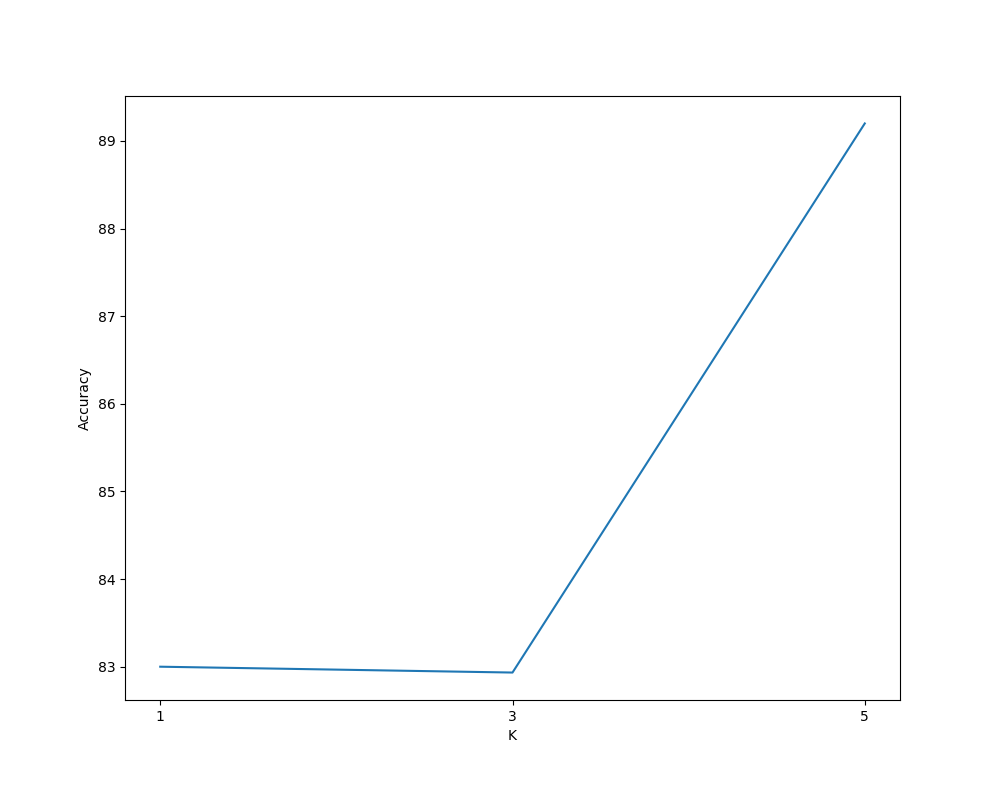
**Accuracy= 83.00%**



**Accuracy= 82.9333333%**



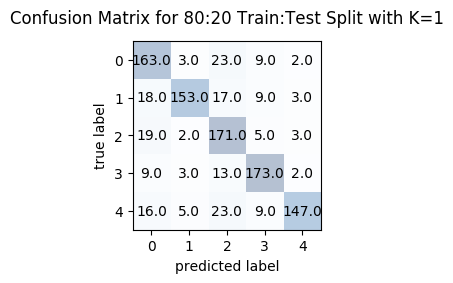
**Accuracy= 89.2%**



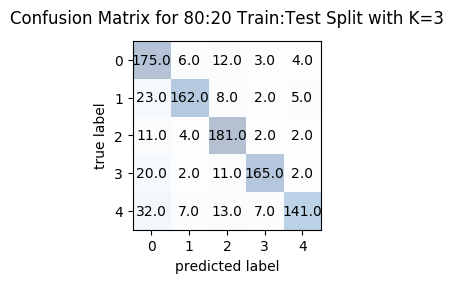
**K vs Accuracy**

From the graph, we can infer that the accuracy of the KNN increases with the increase in the K Value.

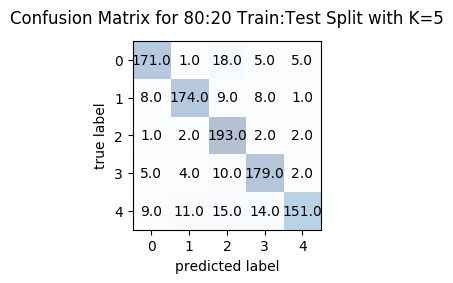
**80:20 Train Test Split**



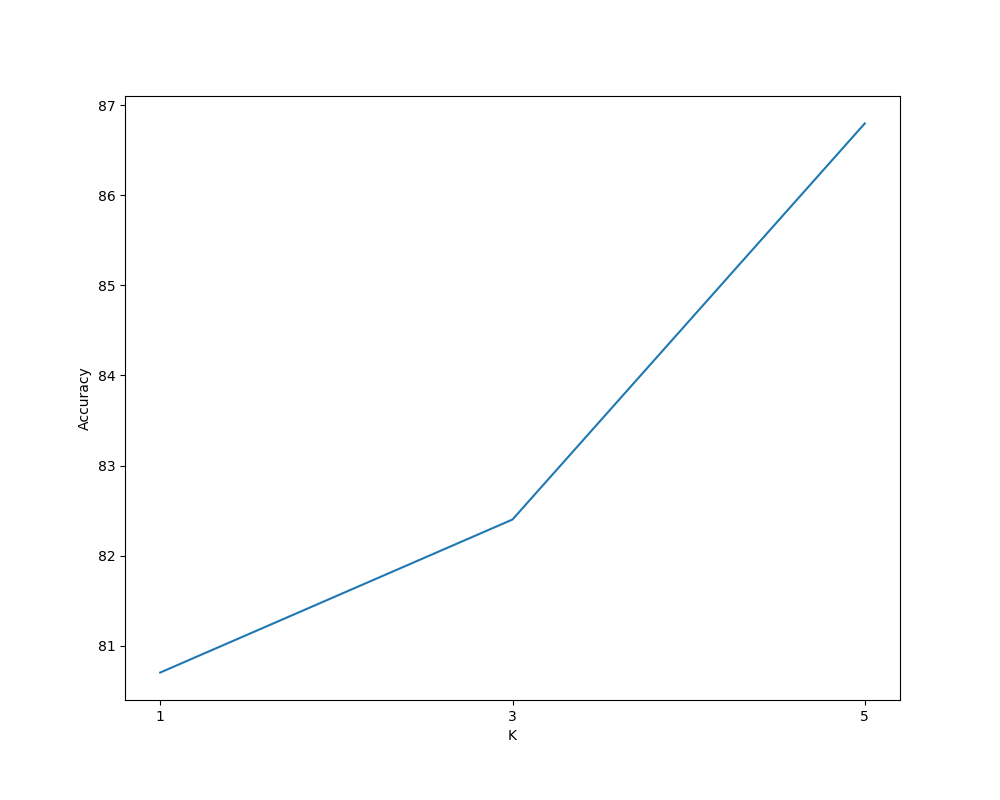
**Accuracy= 80.7%**



**Accuracy= 82.4%**



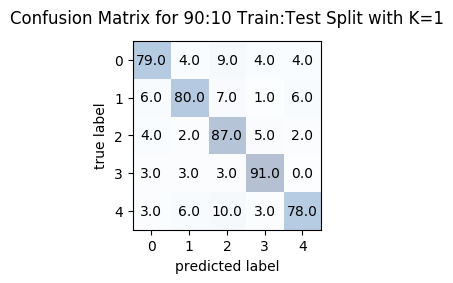
**Accuracy= 86.8%**



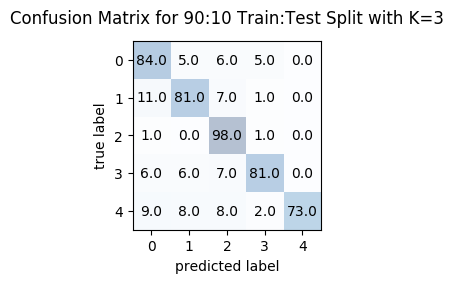
**K vs Accuracy**

From the graph, we can infer that the accuracy of the KNN increases with the increase in the K Value.

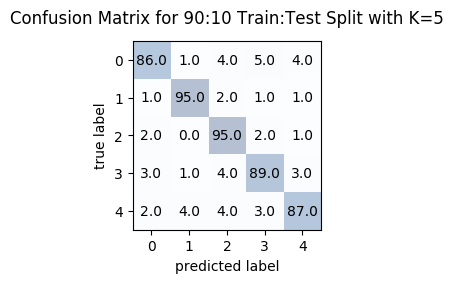
**90:10 Train Test Split**



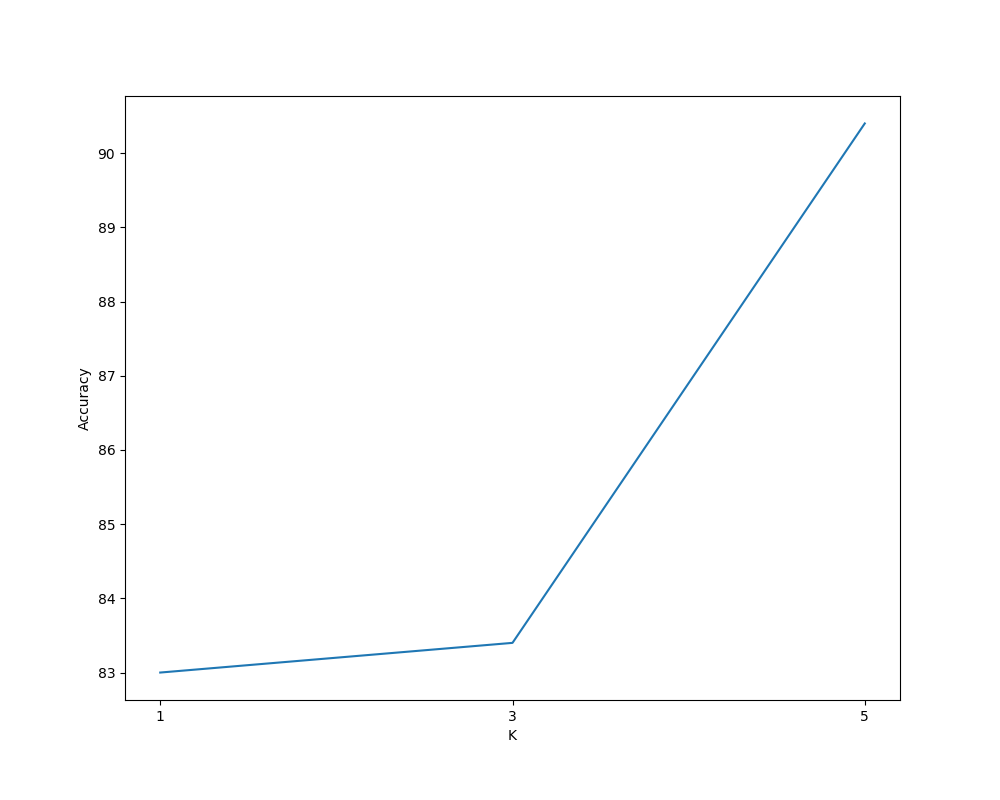
**Accuracy= 83.0%**



**Accuracy= 83.4%**

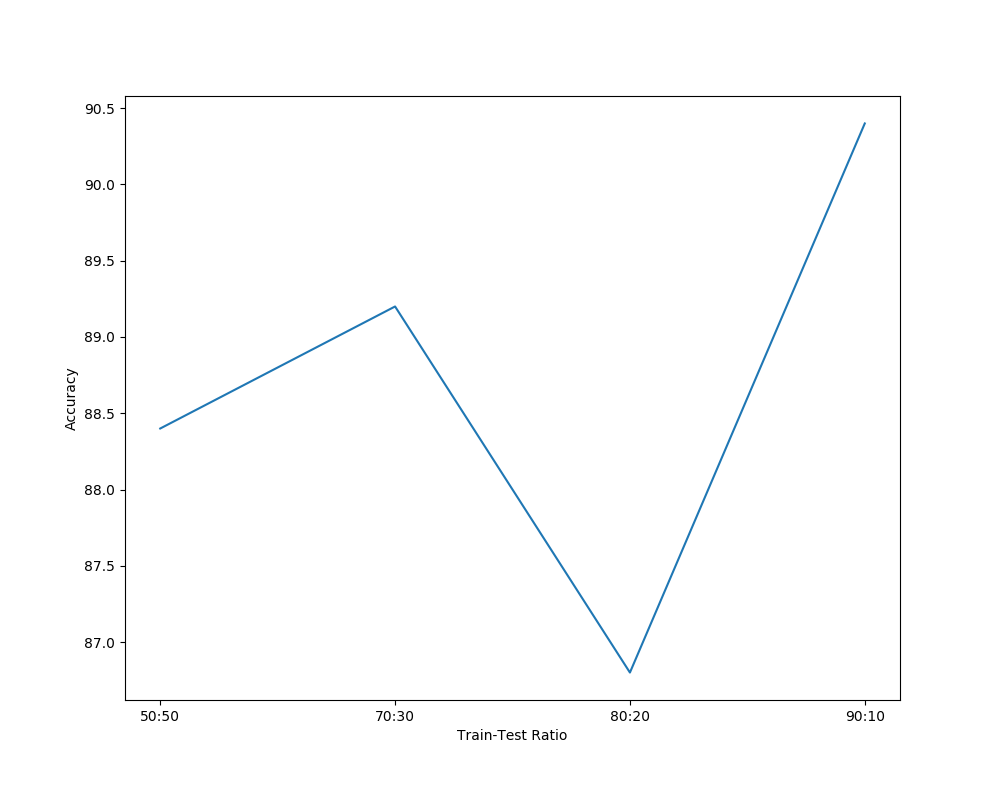


**Accuracy= 90.4%**



**K vs Accuracy**

From the graph, we can infer that the accuracy of the KNN increases with the increase in the K Value.



**Accuracy vs Train-Test Ratio (at best value of K)**

We can infer that the best accuracy occurs at 90:10 train test ratio with a value of **90.4%**

**Comparison**

|  |  |
| --- | --- |
| **KNN Classification** | **90.4%** |
| **Rocchio Classification** | **95.4%** |
| **Naïve Bayes Classification** | **96.8%** |

Naïve Bayes Classification achieves an accuracy of 96.8% which is higher than the Rocchio and KNN classification. KNN achieves only 90.4% because the train dataset available is small. Rocchio classification achieves 95.4% which is quite close to Naïve Bayes classification. We can see that Naïve Bayes classification outperforms the KNN and Rocchio classification.