Name: Vigneshwaarar CR

CWID: A20392185

KNN Classifier Implementation

**KNN Algorithm:**

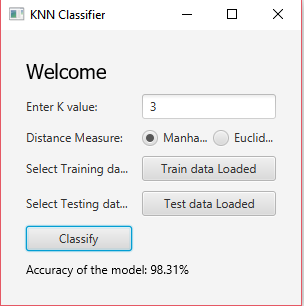
* Before we use the data to build the model, we normalize the dataset on an equal scale and discretization must be done on the classification features, other than the feature which we use for prediction.
* Once data preprocessing is done, split the dataset into training dataset (80%) and testing dataset (20%).  
  we use training dataset to build the model and testing dataset to predict and find accuracy of the model.
* In KNN algorithm, we find the K number of nearest neighbor for the data using the distance value. Distance can be measured using different metrics and here, we use “Manhattan” distance and “Euclidean” distance.
* Manhattan distance = abs|X1-Y1| + … + abs|Xn-Yn|
* Euclidean distance = sqrt((X1-Y1)2 + … + (Xn-Yn)2)

Now, for each data in testing dataset, we find the distance to each of the training dataset. Let’s assume if K is 3, then top 3 training data with lesser distance value will be selected and most occurring value of the classification feature which we are predicting, will be the predicted value for the test data. Similarly, we predict for all the data in test dataset. As we already know the truth, we compare it with the predicted values and find accuracy. We repeat this process for different K values.

Note: Use odd numbers for K-value

**Execution steps:**

* Split the dataset into training and testing dataset
* Run the .jar file
* Enter odd number to K-value
* Select training and testing dataset by clicking on file browser controls
* Click in classify button to see the accuracy of the model for that k-value
* The predicted output file is saved to the folder path where the .jar exists



**Accuracy graph:**

