**Using Credit Approval Data to find a good classifier(kknn) of loan applicant**

**& Used kmeans clustering to predicts flower type**

**Shafagh Yazdani**

***Data***

*The files credit\_card\_data.txt (without headers) and credit\_card\_data-headers.txt (with headers) contain a dataset with 654 data points, 6 continuous and 4 binary predictor variables.  It has anonymized credit card applications with a binary response variable (last column) indicating if the application was positive or negative. The dataset is the “Credit Approval Data Set” from the UCI Machine Learning Repository (*[*https://archive.ics.uci.edu/ml/datasets/Credit+Approval*](https://archive.ics.uci.edu/ml/datasets/Credit+Approval)*) without the categorical variables and without data points that have missing values.*

***Project Overview***

Using Credit Approval Data Set (654 data points, predictor variables: income, credit history,age, family size, assets, liabilities ,) we run the kknn classification model(in the R package kernlab) to find a good classifier to find which loan applicant will pay the loan.

We shuffle the data set first and then split the data set to 2 part , data training , data test, and thenrun the leave one out cross validation for KKnn.

**model<-train.kknn(V11~., data\_training,kmax=20, scale=TRUE)**

We shuffle the data set first and then split the data set to 3 part , data training . data validation, data test,

Applying 10 fold cross validation method and then test our best model with the test data we have found that

Using the kknn model the best accuracy with splitting data was

K= 22 with model accuracy of 0.9191919k

Here the output for different K value accuracy:

k= 1 modelaccuracy= 0.8484848

k= 2 modelaccuracy= 0.8484848

k= 3 modelaccuracy= 0.8484848

k= 4 modelaccuracy= 0.8484848

k= 5 modelaccuracy= 0.8989899

k= 6 modelaccuracy= 0.8989899

k= 7 modelaccuracy= 0.8989899

k= 8 modelaccuracy= 0.8989899

k= 9 modelaccuracy= 0.8989899

k= 10 modelaccuracy= 0.9090909

k= 11 modelaccuracy= 0.9090909

k= 12 modelaccuracy= 0.9090909

k= 13 modelaccuracy= 0.9090909

k= 14 modelaccuracy= 0.9090909

k= 15 modelaccuracy= 0.9090909

k= 16 modelaccuracy= 0.9090909

k= 17 modelaccuracy= 0.9090909

k= 18 modelaccuracy= 0.9090909

k= 19 modelaccuracy= 0.9090909

k= 20 modelaccuracy= 0.9090909

k= 21 modelaccuracy= 0.9090909

k= 22 modelaccuracy= 0.9191919

k= 23 modelaccuracy= 0.9191919

k= 24 modelaccuracy= 0.9191919

k= 25 modelaccuracy= 0.9191919

we did the same 10 fold cross validation model for ksvm with different c value

Setting default kernel parameters

c= 1e-06 modelaccuracy= 0.8181818 Setting default kernel parameters

c= 1e-05 modelaccuracy= 0.8181818 Setting default kernel parameters

c= 1e-04 modelaccuracy= 0.8181818 Setting default kernel parameters

c= 0.001 modelaccuracy= 0.9292929 Setting default kernel parameters

c= 0.01 modelaccuracy= 0.9393939 Setting default kernel parameters

c= 0.1 modelaccuracy= 0.9393939 Setting default kernel parameters

c= 1 modelaccuracy= 0.9393939 Setting default kernel parameters

c= 2 modelaccuracy= 0.9393939 Setting default kernel parameters

c= 4 modelaccuracy= 0.9393939 Setting default kernel parameters

c= 6 modelaccuracy= 0.9393939 Setting default kernel parameters

c= 8 modelaccuracy= 0.9393939 Setting default kernel parameters

c= 10 modelaccuracy= 0.9393939 Setting default kernel parameters

c= 50 modelaccuracy= 0.9393939 Setting default kernel parameters

c= 100 modelaccuracy= 0.9393939 Setting default kernel parameters

**Data (Iris dataset)**

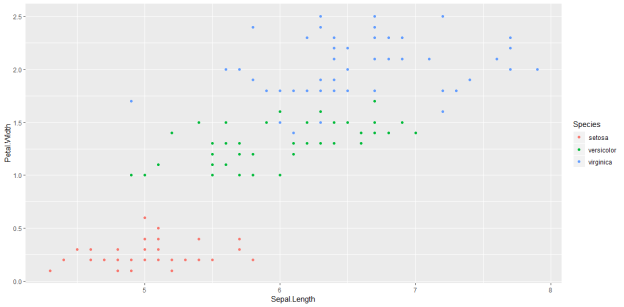
*The iris data set iris.txt contains 150 data points, each with four predictor variables and one categorical response. The predictors are the width and length of the sepal and petal of flowers and the response is the type of flower. The data is available from the R library datasets and can be accessed with iris once the library is loaded. It is also available at the UCI Machine Learning Repository (*[*https://archive.ics.uci.edu/ml/datasets/Iris*](https://archive.ics.uci.edu/ml/datasets/Iris) *).*

**Project Overview:**

Used kmeans clustering to predicts flower type.

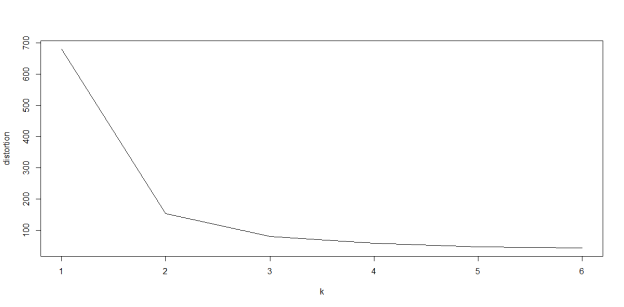
We Use the R function kmeans to cluster the points as well as possible. we report the best combination of predictors, our suggested value of k, and how well our best clustering predicts flower type.

This is plot shows our data set of iris flowers, we see that 2 species of flower have some overlap.

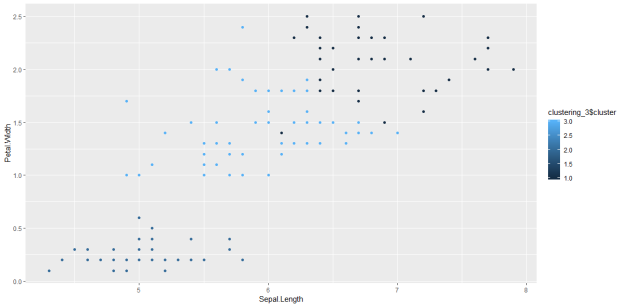


In the elbow diagram of number of k shows that both K=2 and K=3 are good candidate. However, the best k mean clustering model is the K=3. Since the whitin sum of square does not reduce significantly after K=3

We used whitin sum of square(distortion) as metric for choosing our clustering model.



As we see in this two ggplots with 3 clusters and 2 cluster. We notice that 3 clustering model have classified most of these species correctly.



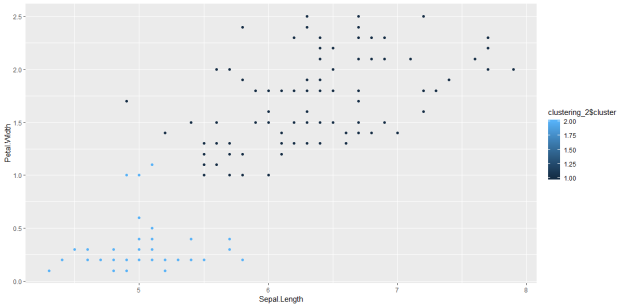


Table for species with 3 cluster shows that most of our species have been classified correctly.

  setosa versicolor virginica

  1      0          2        36

  2     50          0         0

  3      0         48        14