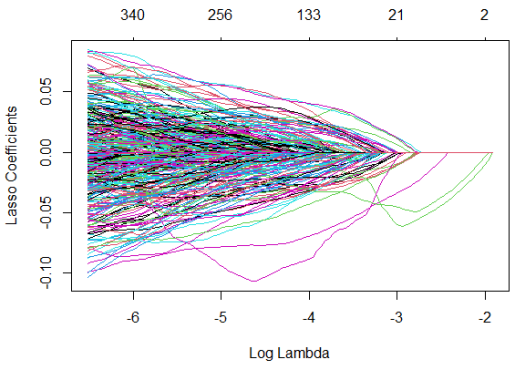
Project Report (STAT 639)
Sreekar Annaluru
832001030

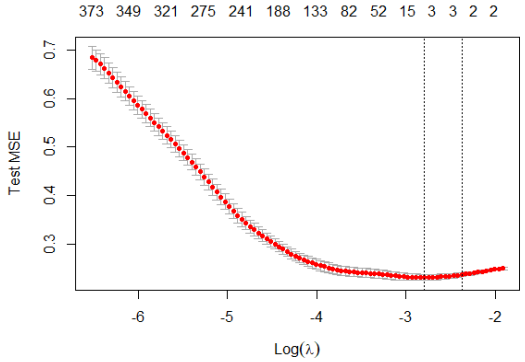
Classification:

First I started by checking the distribution of input data x. All the variables are normally distributed. I installed the “psych” package to check the plots and corresponding correlations. Dimensions of x are (400,500). Here p>n implies this is a higher dimension problem and we have to perform some variable selection and dimension reduction.

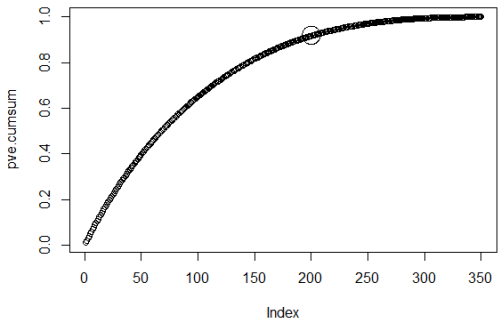
Variable Selection and PCA:

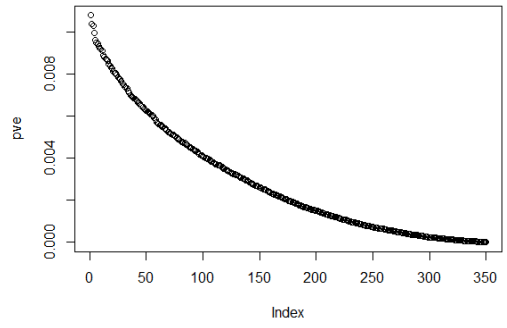
I started with Lasso to perform variable selection using the “glmnet” library. I converted x into a matrix and scaled its values before inputting in the Lasso model. Using cross validation, the best lambda is **0.00234**.





I filtered out non zero coefficients from the model and created a subset of x. Now the x subset contains 400 observations and 350 variables. Now I performed principal component analysis on this subset to capture maximum variability and reduce noise. For this subset the first 200 principal components explained 92% of variability. Hence, I captured only these 200 principal components and converted them back into a dataset using inverse function.





Finally, I deleted the highly collinear variables, more than 0.4, from the dataset. My final dataset named “class_data” has 400 observations and 348 variables!

Neural Networks:

I split the dataset into 75% train and 25% test sets. I performed a simple Neural Network classification and test accuracy of the model is 58%. This is my proposed model.

Predicted/Actual	0	1
0	33	14
1	28	25

Linear Discriminant Analysis:

I split the dataset into 70% train and 30% test sets. I performed a simple LDA classification and test accuracy of the model is 54%

Predicted/Actual	0	1
0	33	18
1	28	21

KNN:

I used KNN classification and iterated over 100 K values to find the optimal K with lowest test MSE. Test accuracy is 61% for the optimal K.

Predicted/Actual	0	1
0	41	31
1	8	20

Random Forest:

I built a random forest model using the original dataset. The OOB error is 40% and the test accuracy is 75% for 75:25 split.

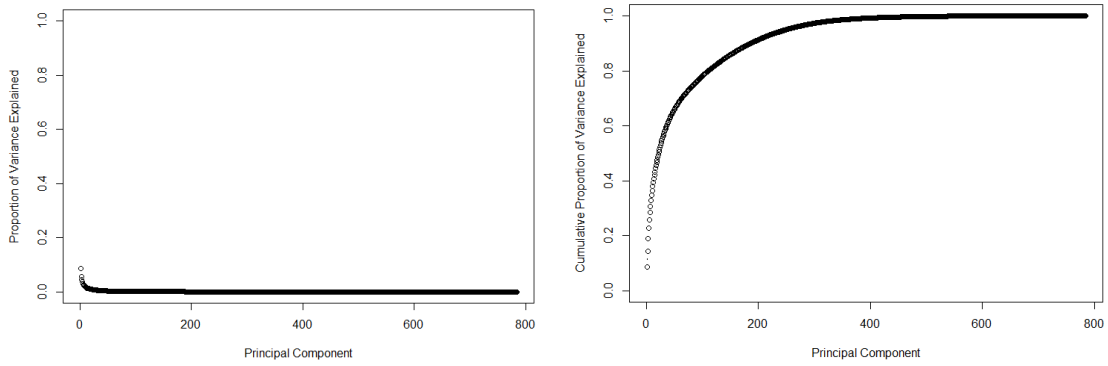
Predicted/Actual	0	1

0	44	8
1	17	31

Note: Please reload the class_data before performing RF.

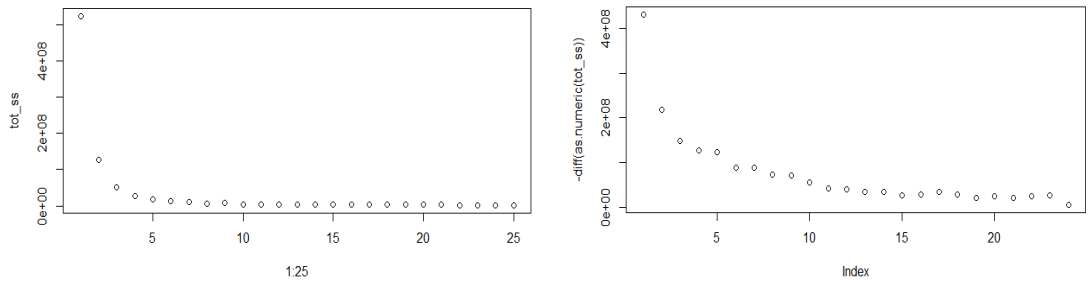
Clustering:

First, started by plotting mean and variance of all columns to check the distribution. Interestingly, all are normally distributed with different means. This helped me in finalizing the number of clusters. As per my analysis there are 4 classes of normal distributions with means 0, 125, 220, 250 (add 150 if 5)!
Now I performed principal component analysis on the variables after scaling the values. First 200 principal components explained 91% of the variability. Hence, I filtered out the first 200 principal components and converted them back to the original format.



K-Means Clustering:

I started with performing k-means clustering on the final dataset. I plotted the total sum of squares within each cluster for each cluster and proposed clusters which correspond to a larger decrease in sum of squares. This is similar to the elbow method.



As you can see maximum reduction of variability is for 2 clusters and the reduction stagnates after 5 clusters. Hence, I am proposing 4 clusters but 5 is very close.

Hierarchical clustering:

I performed all three methods of hierarchical clustering (single, complete, and average). Interestingly, they are very close to the values from k-means clustering. Check the table for more details.

Method/Cluster	1	2	3	4
Single	965	15	5	15
Average	790	105	85	20
Complete	745	185	50	20

Values are pretty similar for 5 clusters.

Gaussian Mixture Models:

I performed a gaussian mixture model using the “Mclust” library. Results are very similar to the above 2 methods for 4 clusters.

Model/Cluster	1	2	3	4
GMM	690	150	100	60

Finally, I propose **K=4** as my final answer for the clustering problem.

References

[Introduction to Statistical Learning]