ROBERT H. SMITH
SCHOOL OF BUSINESS

DATA MINING PROJECT REPORT

PROFESSOR: Courtney Paulson

BUDT 758T Spring 2019

GROUP 14

Zheming Chen

Yefei Fan

Yingjie Gao

Bing Wu

May - 15 - 2019

1 EXECUTIVE SUMMARY

This report is commissioned by the Maryland Department of Health. The hospital's emergency department plays a vital role in healthcare operations and serves as the front line of care for the injured and severely ill. The emergency department is often the only accessible source of providing 24-hour urgent treatment. Thus, it's important to know the potential availability and make better room arrangement to avoid overcrowding. As a consulting team, we utilize patient information to predict whether a patient will return to a hospital within 30 days of being discharged. With return prediction information, hospital or emergency center could prepare in advance to meet further room requirements and make recommendations to alleviate overcrowding. Our project results on patient return prediction will offer operational decision support to the hospital to improve the work efficiency in the emergency department and the quality of healthcare services.

We explore the hospital data using a number of data mining methods. There are 27 variables in hospital dataset, with 1 target variable and 26 predictors. In exploratory data analysis phase, we first performed some preprocessing on the dataset, such as handling missing values, combining multiple levels of categorical variables, removing current variables, creating new variables, as well as scaling variables and creating dummy variables for particular models to avoid potential errors in modeling phase. Also, we explore the potential relationships among variables using clustering and visualize the possible relationships among predictors.

In modeling stage, we utilize a number of the classification analysis methods to predict whether a patient will return to a hospital within 30 days of being discharged. We choose accuracy as the performance evaluation method for comparison among models and we also pay attention to True Positive Rate because we think the patients who will return are more important for hospital to figure out and manager of the emergency center would like to predict all patients who will return accurately. The ultimate goal is to demonstrate a classification prediction model that can be used on future patients to predict whether or not we expect them to return with a high accuracy.

Since our goal is for prediction, we will focus on model prediction power instead of inference power. We could get some sense that some variables, such as the primary method of payment the patient used and what happened to the patient after leaving the emergency room, are important to predict target variable from the result of models. In addition, bagging method with trees gives us the highest accuracy in our testing set, which is 78.05%.

From this analysis, emergency department could use our analysis report to predict the returning patients based on the patients information in hospital database. After knowing how many patients would return within 1 month of being discharged, emergency department could be more prepared in advance for the upcoming patients by allocating resource according to needs. Emergency center could even know which patients would return. Knowing this information, emergency center could send appointment request reminder or message to those potentially returning patients and it would decrease the variance of patient arrival rate, improving operation efficiency in emergency department.

2 EXPLORATORY DATA ANALYSIS

Data Preprocessing

We first check the information of variables one by one in training dataset, including the type of variables, the number of missing values, and the levels of categorical variables. We find that some variables, such as *GENDER* and *CHARGES*, only have a few missing values, lower than 5% of total observations. We then delete missing values directly because we will not lose significant information. However, some categorical variables, such as *ACUITY_ARR*, have considerable missing values. We first discuss whether these missing values are unintentional or intentional missed and then assign new levels to missing values. In addition, we find that some levels of categorical variables, such as *DC_RESULT*, only have a few data points, which may result in errors: for example, those levels with a few data points would appear in train set but don't occur in validation or testing set. Also, too many levels in one variable would make modeling unnecessarily complicated. Therefore, we combine multiple levels into one level to reduce possible errors and complexity. Finally, we remove *INDEX* variable because it represents unique identity and doesn't give much useful information for prediction. After discussing the meaning of *CONSULT_ORDER* , *CONSULT_CHARGE* and *CONSULT_ED*, we decide to combine three variables to two variables to represent the same information with fewer variables to reduce dimensionality of the dataset.

In addition, we also analyze patient departure time and arrival time. We first transform six time variables into numerical variables and then calculate the correlation of departure time and arrival time in each time unit. We find that there is a strong correlation of arrival time and departure time (almost or exactly equal to 1) in each time unit. Thus, we decide to keep three arrival time variables and remove departure time variables to prevent multicollinearity between independent variables. **(See Appendix 1.1 for the detailed data preprocessing introduction)** We clean data in unseen testing dataset using the same methods. Besides, we use mean value to impute missing values in *CHARGES*.

Variables Relationship

We then want to explore the possible relationships between variables. We choose unsupervised methods, such as k-means and hierarchical clustering, to discover potential groups and analyze the characteristics of groups. Hierarchical clustering provides us a dendrogram to visualize the clusters and k-means method provides statistical information about clusters. These methods could give us insights on characteristics of different groups. We also utilize visualization techniques to explore the possible relationships of variables that can well explain our data.

We use complete linkage method to make hierarchical clustering with a subset of size 1000 with all variables we have. From the dendrogram (**See Appendix 2.1**), we probably want to choose 2 clusters. However, to be sure about the clustering quality, we use K-means clustering and draw elbow plot (**See Appendix 2.2**) to gain statistical information about the clusters. Through 1 to 15,

we find eight centroids is a good choice to explore meaningful variables for us to understand our data, and get 51.9 % of the possible 100% variation. Comparing cluster means of different variables in the 8 clusters (**See Appendix 2.3**), we have the following interesting findings about a group.

Cluster 3 has more unknown race patients who were in the self-pay category, and we assume they were not covered by health insurance. More cluster 3 patients left without finishing the treatment, some discharge and some left without being seen. We assume that there is a possible reason to explain this phenomenon. Patients in cluster 3 came to the emergency department with less urgent situation, when faced with expensive charges, they could not afford to pay by themselves and have no insurance to cover. Considering the disease was not that urgent, they chose to leave without finishing the treatment. After researching relevant information, we find that the Emergency Medical and Treatment Labor Act (EMTLA) passed by Congress in 1986 explicitly forbids the denial of care to indigent or uninsured patients based on a lack of ability to pay. This may be the reason why more patients in cluster 3, who were not covered by insurance, came to the emergency room than a regular department with non-emergent reasons, and then left without being seen for not being able to afford the payments. Thus, we assume these patients are more dependent on emergency rooms and may have a higher possibility to come back again. We can further assume that our data is from public hospitals.

We then use visualization methods to show the variable relationships observed in our findings. From the relationship between *FINANCIAL_CLASS* and *ED_RESULT* (**See Appendix 2.4**), we can see that patients who paid by themselves, apart from those formally covered by insurance, are more likely to left without being seen than other categories. This is consistent with our presumptions that self-pay patients are more likely to left without being seen. While visualizing the *FINANCIAL_CLASS* and *ACUITY_ARR* (**See Appendix 2.5**), only a few self-pay patients have immediate or emergent acuity. Most of the self-pay patients clustered in the unknown, less urgent and urgent acuity category. We could conclude that self-pay patients are more likely to come to the emergency room with less urgent situations.

Therefore, we find that the patients who come to the emergency room without insurance and with less emergent reasons. It may not only because they have a higher risk to elope, causing losses to the hospital, but also because they tend to come back to hospital for not emergent reasons again in the future, which prohibits the hospital from making full use of emergency rooms and serving those who are really in need of this limited resource. Also, we think the hospital might need to pitch more fund for the low-income group in terms of health care coverage to help alleviate the overuse of emergent rooms.

3 MODEL BUILDING AND MODEL EVALUATION

Data Partitioning

After preprocessing, the dimensionality of our Hospital_Train dataset is 22 columns and 37909 rows. We randomly partition the data set into 15% testing set and 85% rest set. We then split the rest dataset into 15% validation set and the 85% training set. In modeling phase, we will use training set to build models, use validation set to choose best parameters or cutoff, which can balance the tradeoff of accuracy and true positive rate, and use testing set to compare models with baselines and choose the model with best performance in practice.

Evaluation Method and Baseline Illustration

Our goal is to predict whether a patient would return or not in one month after the patient is discharged. Therefore, we classify it as a classification problem. The most appropriate evaluation method for this problem is accuracy because there is a clear baseline in each dataset, most common baseline, to evaluate model performance. In addition to accuracy, we pay attention to the True Positive Rate (TPR). For an emergency department, it cares more about not missing any patient who would return because emergency center needs to make sure they have allocated enough resource in advance.

Model Building and Evaluation

Overall, the models we have tried are logistic regression (backward elimination and forward selection with or without stepwise procedure), LASSO regression, Ridge regression, KNN, classification tree, neural network, random forest, bagging, and boosting **(See Appendix 2 to check out our model outputs)**.

Logistic Regression

We first try logistic regression model because it is designed for classification problem and provides interpretable outputs that clearly explain the relationship of predictors and target variable. We have the highest validation accuracy of 76.74% with a cutoff of 0.4. The accuracy in the testing set is 76.8%. We also try stepwise logistic regression, LASSO and Ridge to see whether removing variables or introducing bias could improve the accuracy. However, the result doesn't improve significantly. We also try principle component analysis (PCA) to combine variables and reduce dimensions, but the outcome is not very satisfying. The PCs don't explain variance efficiently. **(See Appendix 3.5 for PCA plot)**

KNN

KNN is a data-driven method, useful for prediction with a large dataset. Thus, we decide to try KNN model to see whether we can get higher accuracy. Because KNN is using distance between data points to perform classification and the value of charges in our dataset is very large so that it would affect KNN to measure the distance and perform classification, we converting categorical variables to dummy variables and scaling variables between 0 and 1. We compare different k

values and cutoffs in validation set. The best accuracy we get in validation set is 76.7%, when k is 25 and cutoff is 0.5 and the TPR is only 17.1%. The accuracy in testing set is 76.8%. **(See Appendix 3.6 for spider plot)**

Neural Network

We also try fully connected feed forward neural network. Neural network is a black box algorithm which has been proven to be good in prediction with almost zero possibility to interpret. When we set only one hidden layer of neural network, the best accuracy in validation set is 77.2%. In testing data, the accuracy is only 65%. If we set two hidden layers and four nodes, the model failed to converge. We don't continue with a higher stepmax value because we believe the reason is not the hyperparameters but the fact that neural network is not a very good model for sparse data. There are almost 80 dummy variables after transformation, which makes the whole data set sparse. Neural network can easily get stuck in a local minimum no matter what hyperparameters we set. **(See Appendix 3.8 for network structure plot)**

Classification Tree

We also try classification tree. According to the plot **(See Appendix 3.7 for tree plot)**, with tree size of 4, chosen from validation set, *ED_RESULT*, *GENDER* and *FINANCIAL_CLASS* are important for building the tree. The pruned tree has a 76.3% accuracy and 57.62% TPR in validation set, and a 76.91% accuracy and 55.75% TPR in testing set.

Ensemble Methods

Ensemble method is a good way to reduce variance without increasing bias, reducing the model's reliability on original dataset and improve the prediction power to unseen dataset. Therefore, we try bagging trees with all variables and run 1000 trees. We use the validation set to choose the best cutoff. With cutoff 0.6, the testing accuracy of bagging is 0.7805 and the TPR is 0.1918.**(See Appendix 3.9 for variable importance plot)** We can also see from the variable importance plot that the payment method, age and what happened to the patient after leaving the emergency room are the top 3 important variables for the prediction in bagging method.

However, we are still concerned that some variables would be very important for prediction and would dominate splits if we use bagging method. Therefore, we decide to run random forest to de-correlate the bagged trees. We run random forest with all variables, 1000 trees, and we set the model to try 10 variables in each split. The highest accuracy occurs when cutoff is 0.5 in validation set, and we get corresponding testing accuracy 0.7796 and the TPR 0.2815 that is slightly better than bagging method.**(See Appendix 3.10 for spider plot)** There are also some interesting findings: some variables are very important for building the tree but not important for prediction.**(See Appendix 3.11 for variables importance plot)**

We want to get a higher accuracy and we analyze that prediction might not perform well on this dataset because this data set is unbalanced. There is about 75% patient who would not return to the emergency center within one month. Therefore, boosting might be a good choice for this prediction task because it's designed for focusing on wrong predicted data points and adding

more weights on those data points to predict those data correctly. However, after running boosting methods with 10000 trees, the testing accuracy is only 0.7738.

Model Performance Evaluation

After running all of the models, we choose that the best performance model was the model has the highest accuracy for prediction a patient who would return or not in testing set. In our case, bagging method has the highest accuracy on both testing set and actual test data (78.05% and 77.49% respectively). In addition, if emergency center takes true positive rate into consideration, tree performs best with 0.5575 TPR in testing set. Therefore, it makes sense for emergency center to use several models to analyze the result and explore the usage of the analysis.

4 CONCLUSION

Emergency Department is one of the most limited and vital elements in healthcare industry. It's very important for ED to accurately prediction how many patients would be in their department in the upcoming time period. Predicting the returned patients is especially important because it not only shows the quality of healthcare service that the emergency center provides, but also helps emergency department to know in advance the information about how many patient will return in the future and the healthcare information about those patients.

From this analysis, our recommendations to Emergency Department is that when choosing which model is the best for this prediction task, they not only need to compare the accuracy, but also take true positive rate into consideration since the cost of misclassifying a returning patient as not returning patient is high. In addition, initializing programs which can help patients who don't have insurance to learn necessary knowledge about the benefits of getting proper insurance and how use the insurance to help them keep healthy with lower cost could help emergency department to reduce the return rate and have more control on how many patients would return in the future.

In total, with our prediction, emergency department can know in advance that how many patients would return within 1 month of being discharged and prepare appropriate amount of resource for these patients in upcoming month. Also, emergency center could even know which patients would return. With this information, emergency center could send appointment request or reminder message to those potentially returning patients in the beginning of the following month. This would decrease the variance of patient arrival rate, improving operation efficiency in emergency department and the quality of the service in emergency department.

5 APPENDIX

Appendix 1: Exploratory Data Analysis

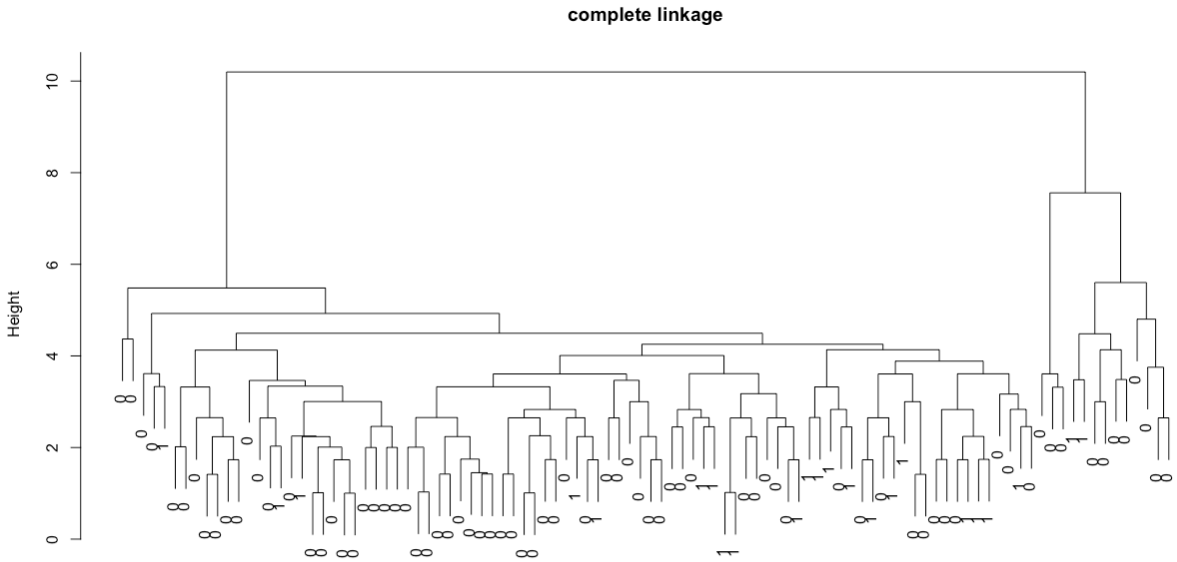
1. Data Processing in Hospital_Train.csv

Variable Name	Type	Missing Value		REGROUP	
		#	how to handle	REGROUP	COMMENT
HOSPITAL	Factor	0	-	-	-
GENDER	Factor	2	Delete	-	-
AGE	Numeric	0	-	-	-
RACE	Factor	177	to 'Unknown'	Declined to Answer' to 'Unknown'; Level whose number of values less than 100 into 'Other'	-
ETHNICITY	Factor	373	to 'Unknown'	Declined to Answer' to 'Unknown'	-
FINANCIAL_CLASS	Factor	0	-	Medicaid Pending', 'Out of State Medicaid', 'Global Contracts' into 'Other'	-
WEEKDAY_ARR	Factor	0	-	into 'Weekday' and 'Weekend'	weekday: 1~5; weekend: 6~7
HOUR_ARR	Factor	0	-	into 'day' and 'night'	day: 8:00~20:00; night: 20:00~8:00
MONTH_ARR	Factor	0	-	into four seasons	spring: 3~5; summer: 6~8; autumn: 9~11; winter: 12~2
WEEKDAY_DEP	Factor	0	-	into 'Weekday' and 'Weekend'	Delete
HOUR_DEP	Factor	0	-	into 'day' and 'night'	Delete
MONTH_DEP	Factor	0	-	into four seasons	Delete
SAME_DAY	Factor	0	-	-	-
ED_RESULT	Factor	73	Delete	Level whose number of values less than 100 into 'Other'	-
ACUITY_ARR	Factor	3263	to 'Unknown'	-	-

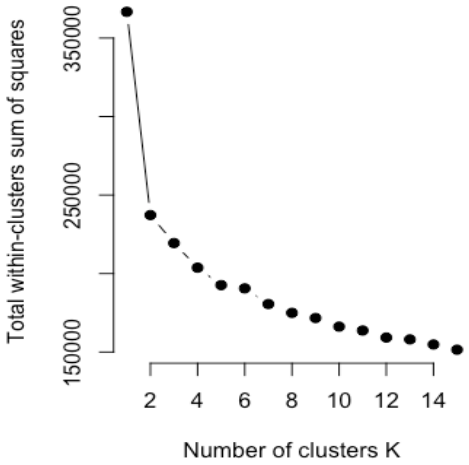
DC_RESULT	Factor	1	Delete	Level whose number of values less than 100 into 'Other'	-
ADMIT_RESULT	Factor	30610	to 'Not Admitted'	-	-
CONSULT_ORDER	Factor	0	-	-	Delete
CONSULT_CHARGE	Factor	0	-	-	Delete
CONSULT_IN_ED	Factor	37321	to '0'	-	Delete
DIAGNOSIS	Factor	0	-	-	-
DIAG_DETAILS	Numeric	0	-	-	-
RISK	Factor	33045	to 'Unknown'	-	-
SEVERITY	Factor	33045	to 'Unknown'	-	-
CHARGES	Numeric	100	Delete	-	-
CONSULT_ED	Factor	created	0	0 = no consultation, 1 = consultation not in ED, 2 = consultation in ED	-
CONSULT_C	Factor	created	0	0 = no consultation, 1 = free consultation, 2 = charged consultation	-
RETURN	Factor	141	Delete	-	-

Appendix 2. Clustering:

1 Dendrogram



2. Elbow plot of k-means clustering



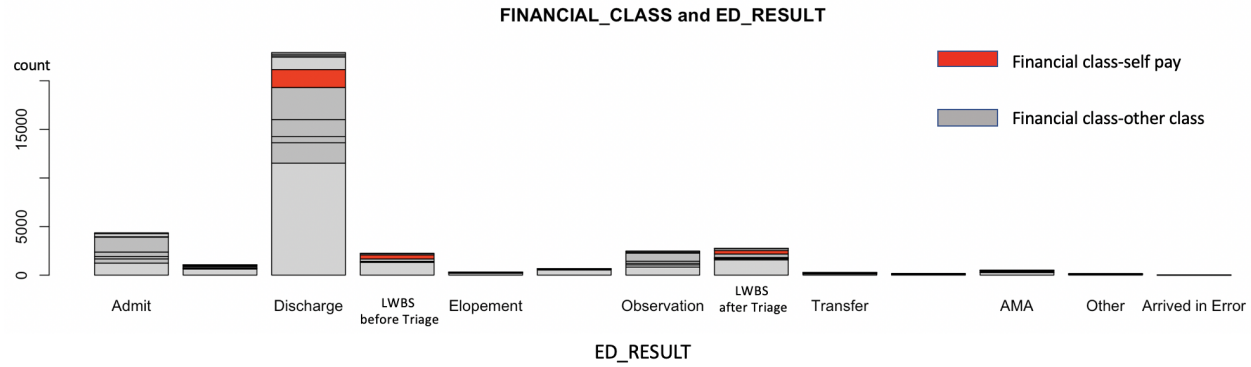
3. Result of K-Means clustering

```
Within cluster sum of squares by cluster:
[1] 18031.35 25458.47 22015.65 26610.72 15073.53 17739.56 30211.17 21194.70
 (between_SS / total_SS =  51.9 %)
```

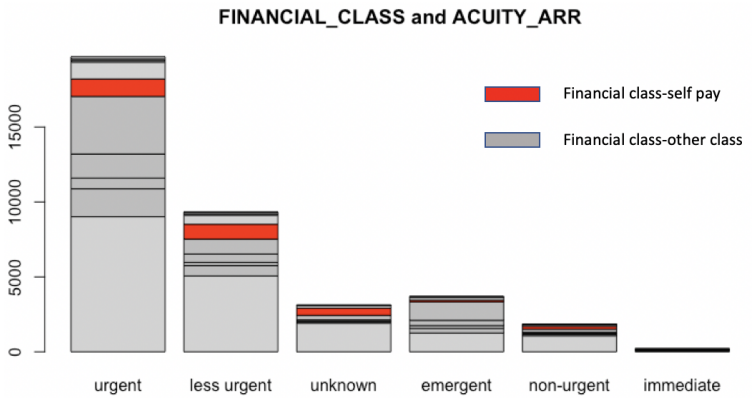
Clusters	RACE Unknown	FINANCIAL _CLASS. Self.pay	ED_RESULT Left.prior.To. Completing. treatment	ED_RESULT LWBS. Before. Triage	ACUITY_ARR Unknown	ACUITY_ARR Emergent	ACUITY_ARR Immediate	charge	Return
1	0.0080	0.0765	0.0000	0.0014	0.0050	0.0878	0.0030	0.0013	0.2289
2	0.0086	0.0796	0.0001	0.0020	0.0097	0.0615	0.0036	0.0011	0.2017

3	**0.0265**	**0.1482**	**0.3804**	**0.4617**	**0.4609**	**0.0094**	**0.0009**	**0.0002**	**0.4278**
4	0.0033	0.0109	0.0000	0.0000	0.0156	0.3216	0.0185	0.0186	0.1608
5	0.0168	0.0974	0.2414	0.2878	0.2885	0.0713	0.0025	0.0014	0.3638
6	0.0033	0.0084	0.0000	0.0000	0.0659	0.3820	0.0302	0.0141	0.1822
7	0.0066	0.0732	0.0001	0.0032	0.0123	0.0706	0.0032	0.0013	0.2241
8	0.0048	0.0808	0.0000	0.0003	0.0086	0.0534	0.0017	0.0009	0.2236

4. Relationship of FINANCIAL_CLASS and ED_RESULT



5. Relationship of FINANCIAL_CLASS and ACUITY_ARR

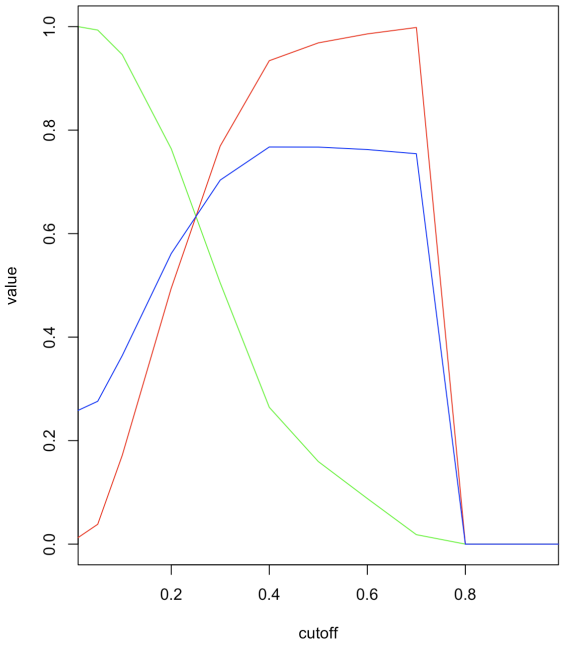


Appendix 3: Modeling and Model Evaluation

1. Summary of Original Logistic Regression:

```
> log_acc
 [1] 0.2584316 0.2760190 0.3641630 0.5615560 0.7034968 0.7674322 0.7672253 0.7624664 0.7543969
[10] 0.0000000 0.0000000 0.0000000 0.0000000
> log_TPR
 [1] 1.00000000 0.99334996 0.94596841 0.76392352 0.50457190 0.26433915 0.15960100 0.08811305
 [9] 0.01828761 0.00000000 0.00000000 0.00000000 0.00000000
> log_TNR
 [1] 0.01267218 0.03829201 0.17134986 0.49449036 0.76942149 0.93415978 0.96859504 0.98595041
 [9] 0.99834711 0.00000000 0.00000000 0.00000000 0.00000000
```

2. Spider plot of TPR, TNR and accuracy:



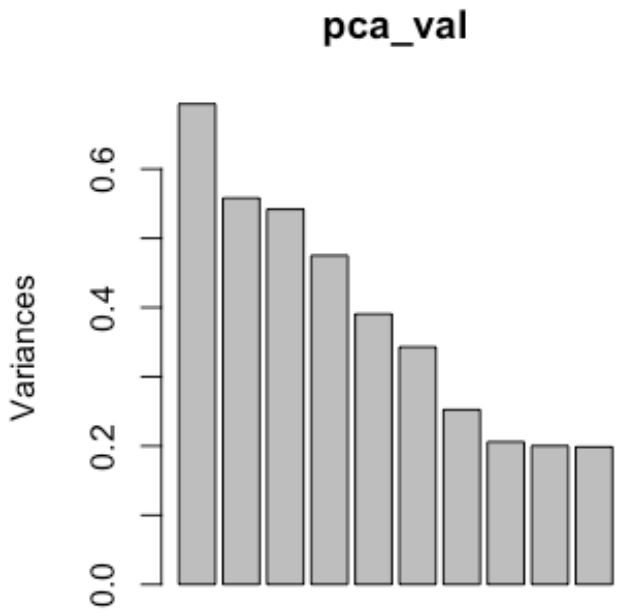
3. Summary of Ridge Logistic Regression:

```
> rid_acc
 [1] 0.2434049 0.2629265 0.3441787 0.5504749 0.7061203 0.7687302 0.7745339 0.7682026
 [9] 0.7590573 0.0000000 0.0000000 0.0000000 0.0000000
> rid_TPR
 [1] 1.000000000 0.996353027 0.956236324 0.782640408 0.511305616 0.251641138
 [7] 0.169948942 0.078045222 0.004376368 0.000000000 0.000000000 0.000000000
[13] 0.000000000
> rid_TNR
 [1] 0.003012746 0.029895713 0.149710313 0.476709154 0.768018540 0.933024334
 [7] 0.966628042 0.987485516 0.998841251 0.000000000 0.000000000 0.000000000
[13] 0.000000000
```

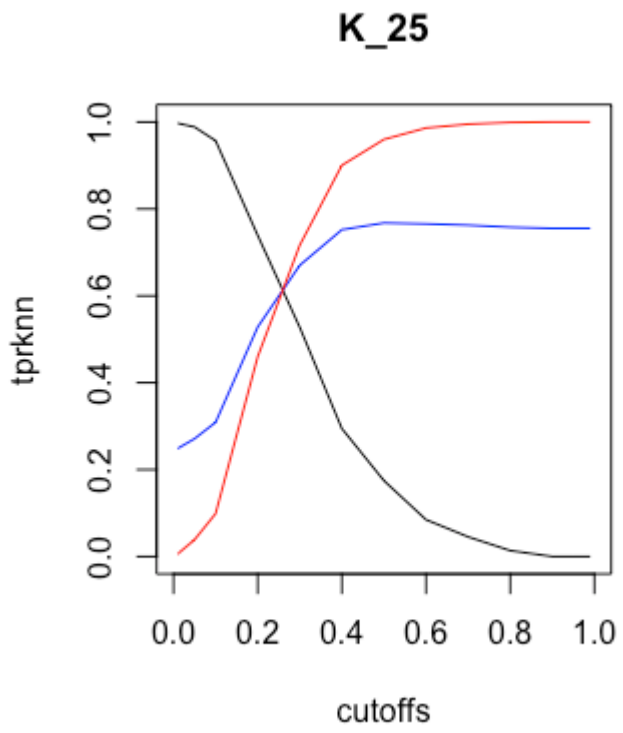
4. Summary of LASSO Logistic Regression:

```
> las_acc
 [1] 0.2485051 0.2653887 0.3561379 0.5529370 0.7026029 0.7682026 0.7747098 0.7699613
 [9] 0.7599367 0.0000000 0.0000000 0.0000000 0.0000000
> las_TPR
 [1] 1.000000000 0.995623632 0.951859956 0.775346462 0.528811087 0.255288111
 [7] 0.174325310 0.086068563 0.007293946 0.000000000 0.000000000 0.000000000
[13] 0.000000000
> las_TNR
 [1] 0.009733488 0.033371958 0.166859791 0.482271147 0.757821553 0.931170336
 [7] 0.965469293 0.987253766 0.999073001 0.000000000 0.000000000 0.000000000
[13] 0.000000000
```

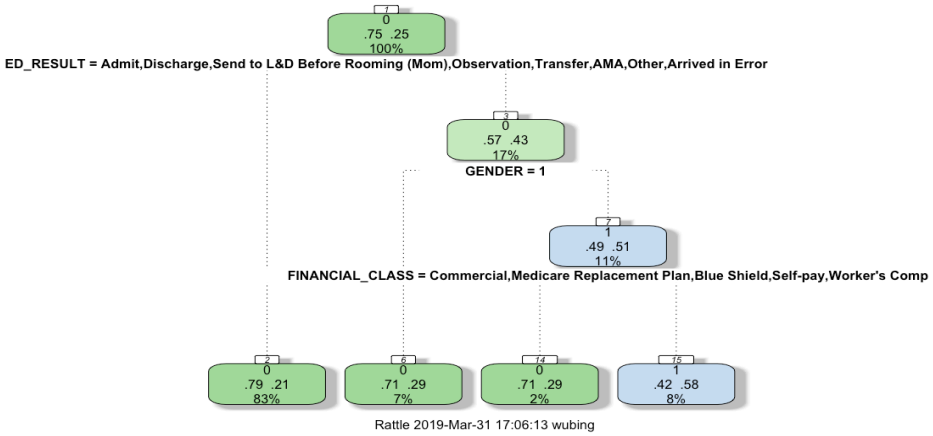
5. Principal Component Analysis Plot:



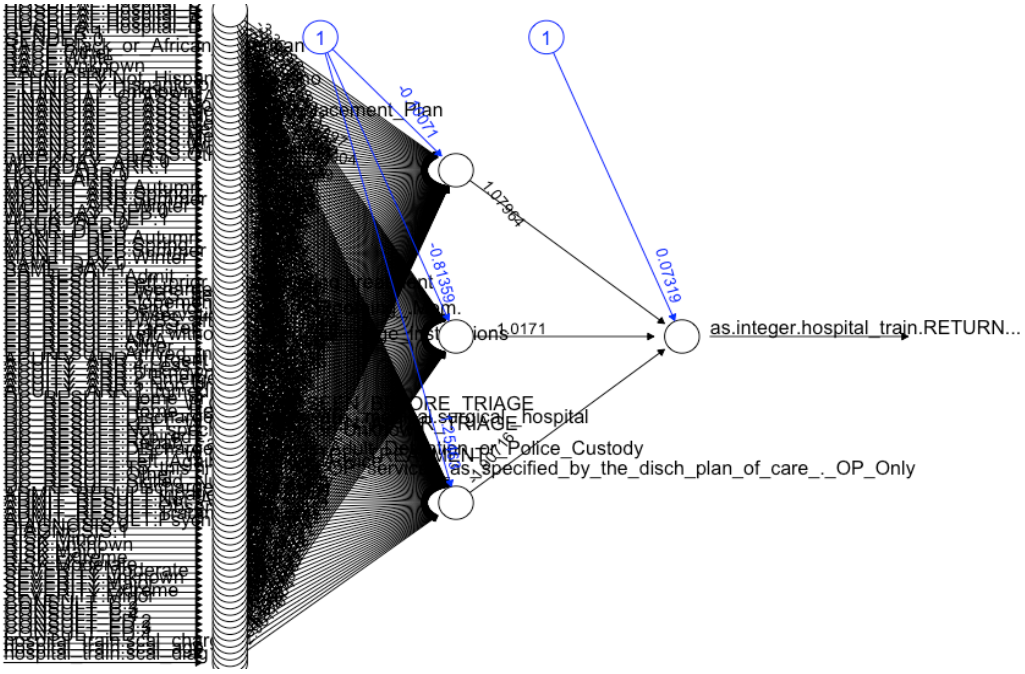
6. KNN - spider plot of TPR, TNR and accuracy:



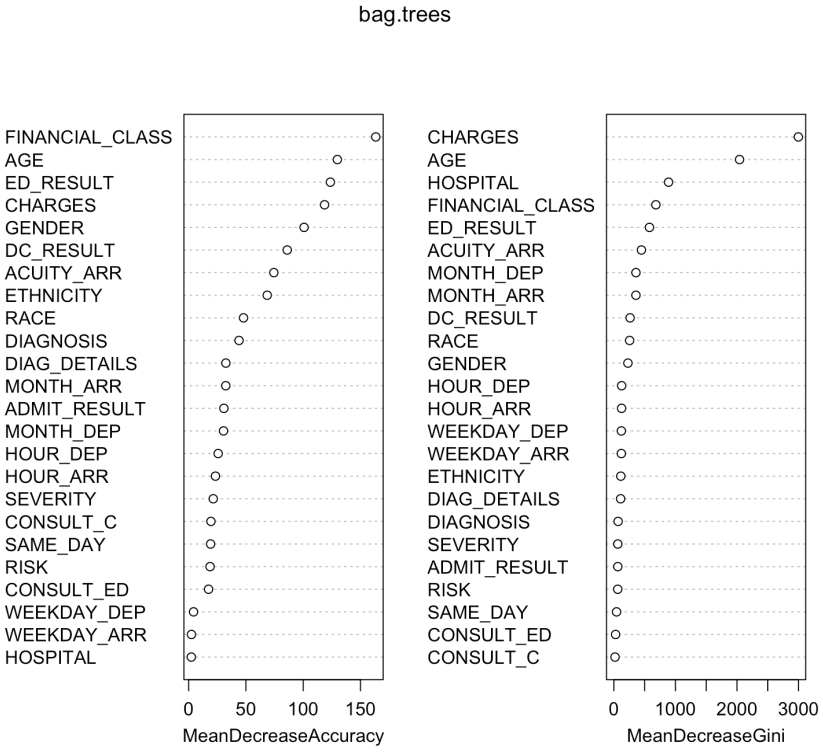
7. Tree Plot:



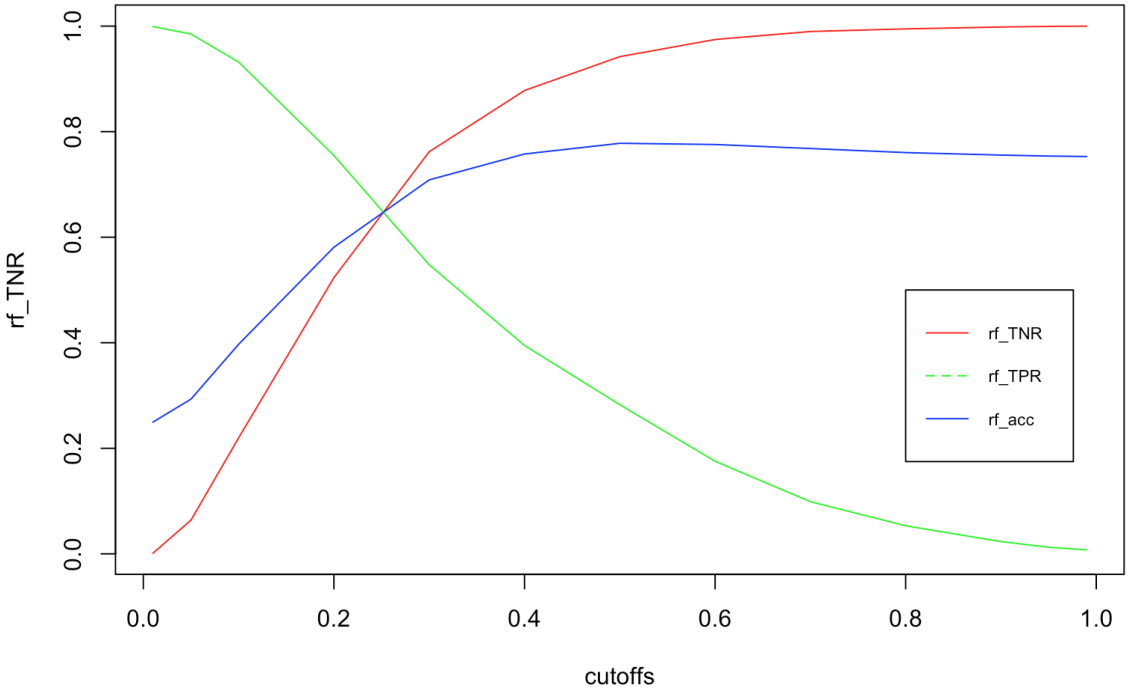
8. Neural Network:



9. Bagging(25 variables, 1000 tree, mtry = 25): varImportant plot:

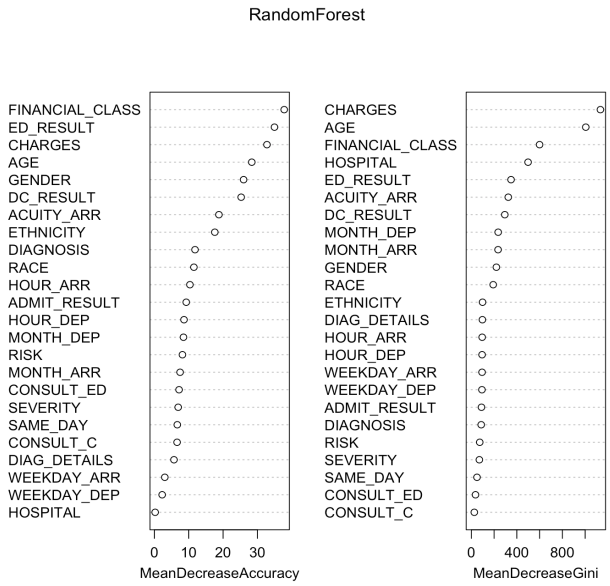


10. Random Forest spider plot:



Based on this result, I choose cutoff 0.5 for later use.

11. Random Forest (all 25 variables, 1000 trees, mtry=10):
Variable Importance plot:



Appendix 4: Model Evaluation

Model	Accuracy training set	Accuracy validation/ cross-validation set	Accuracy testing set	TPR training set	TPR validations/ cross-validation set	TPR testing set
Base line	0.7540	0.7510	0.7530	-	-	-
Logistic Regression	0.7653	0.7674	0.7680	0.2519	0.2643	0.2611
LASSO	-	0.7701	0.7747	-	0.2594	0.1743
Ridge	-	0.7688	0.7745	-	0.1571	0.1699
LDA	0.7650	0.7659	0.7662	0.2494	0.2660	0.2582
Tree	0.7662	0.7637	0.7691	0.5762	0.5721	0.5575
KNN	0.7714	0.7670	0.768	0.1866	0.1710	0.1746
RF(all variables cutoff 0.5)	0.9990	0.7780	0.7796	0.9979	0.2826	0.2815
bag.trees(all variables cutoff 0.6)	0.9994	0.7788	0.7805	0.9978	0.1978	0.1918
Boosting 1000trees	0.7675	-	0.7727	-	-	0.2113
Boosting 10000trees	0.7684	-	0.7738	-	-	0.1921