Predictive Modeling with Medxoom

ISYE 6754

Mary Munro and Greg Schreiter

Introduction

Medxoom is an Atlanta-based startup that aims to save customers, such as employer health plans, money based on a comparison of pricing schemes at hospitals in the United States. For this project, they wanted us to predict either the relative cost of inpatient and outpatient care (RPIO), or the relative price of outpatient care (RPO) for these hospitals. Modeling and analysis was performed separately so that effective models for each variable could be built. Medxoom recommended binning these continuous predictors into groups, which allows for more tangible comparisons. After building models, an error analysis was performed to allow us to gain insight into model performance. The final section has outcomes and recommendations for future work.

Data Description and Cleaning

Items 1-2 below describe the datasets we chose to use from those provided by Medxoom.
Item 3 describes the derived dataset used in Greg's analysis
Items 4-8 describe the datasets used in Mary's Analysis. In essence, RAND, hospital rating, and census data were used as predictors. The R file, *1_joining_and_cleaning.Rmd*, performs the joining and cleaning of these predictors. Curated RAND and Label data (**4,5**) were inner joined on Medicare provider number and filtered to include only years 2015-2017 since the labels were derived during this time. This data was then aggregated by year, taking the median of continuous variables and mode of discrete variables. Continuous NA's were replaced by the variable's median, and discrete NA's were replaced by the variable's mode. Rating data (**6**) was then left joined to this set, with NA's replaced by the mode. Finally census data by county (**7**) was left joined to this, replacing NA's with the median of the county's parent state. This decision was made since label dependency on state was discovered.

1. **Rand_hcris_cy_hosp_a_2020_05_01.csv.zip**
    a. The above file will be referred to as the "RAND dataset" going forward.
    b. The RAND dataset contains information about the financial performance, accounting, and other hospital operations data for a large number of hospitals in the United States. In total there are roughly 1596 hospitals per year, with an associated 1247 columns of variables for each hospital. Since there are a very high number of variables in the RAND dataset, a method had to be devised to reduce this to a tractable number. Several things were apparent about the RAND dataset:
    c. Many columns had 50-90% of values that were missing. The decision was made to omit these columns as they would not provide much information
    d. Some of the columns, such as "Inpatient Costs," "Outpatient Costs," "Total Costs," "Total Assets," and "Total Salaries," are combinations of many other variables. These variables can be included while omitting their constituent parts, while still yielding effectively the same information.
    e. Geographical or location identifiers were mostly omitted, however a few of these a few of them like zip code, hospital name, hospital street address, state, were

kept - not necessarily for predictive purposes, but they could help make sense of regional variations or other patterns.

Using the above guidelines, a subset of 45 of the 1247 variables in RAND was identified that still had the majority of information that was available. This file is included as CuratedData.csv. A full list of these 45 variables is also included in the appendix.

2. **Detailed_Data.xlsx**
   The Hospitals tab of the excel document contains the “labels” that Medxoom wants us to predict. The variables “Relative Prices of Inpatient and Outpatient Care” and “Relative Prices of Outpatient Care” were chosen to focus on for our analysis. This data was joined with the RAND dataset using the Medicare provider number as a primary key to produce the dataset used for Greg’s analysis.

3. **GregsData.csv**
   This dataset contains the results of joining the Hospitals tab of Detailed Data and the RAND dataset by the medicare provider number, along with a few other modifications: There were several Y/N responses that were changed to 0-1 encodings. Also the column names were edited to reduce their length, for ease of model building in R. Lastly, the data were filtered so that only the relevant years of 2015, 2016 and 2017 were included. There are 4754 observations and 66 predictors total. This file had been included in the zip folder.

4. **Hospitals_labeled.csv**
   This contains the provided label column, “Relative Prices of Inpatient and Outpatient Care”, for given hospital names and Medicare provider numbers. This is extracted from the Hospitals tab of Detailed_Data.xlsx. Hospital medicare provider number was kept for joining.

5. **CuratedData.csv**
   The aforementioned selected columns from RAND. Hospital medicare provider number was kept for joining.

6. **HCAHPS_Hospital_clean.csv**
   This contains hospital ratings. Hospital medicare provider number was kept for joining.

7. **Census_county.csv**
   This contains census data for all counties of the US downloaded from census.gov. State and County columns were kept for joining.

8. **census_columns.csv**
   This describes the columns of Census_county.csv. A column has also been added with y/n entries in order to identify variables for extraction for the analysis.

Mary’s model creation and Error Analysis

Exploratory Data Analysis

EDA hinted at a strong dependency on state (see figs 1-3 below), and so state and census data was closely monitored during iterative variable selection and model formulation. In the final models, it was indeed proven to be the case.

Figure 1: Average relative price across states of the US (using 726 labeled data points)

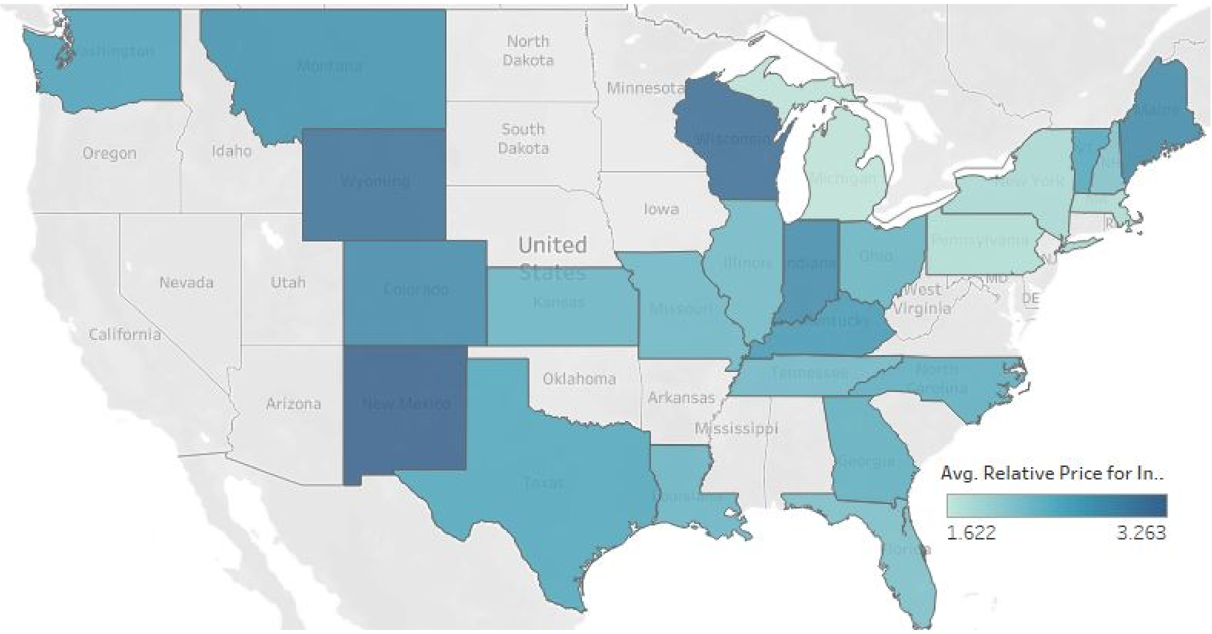


Figure 2: Variance of relative price across states of the US

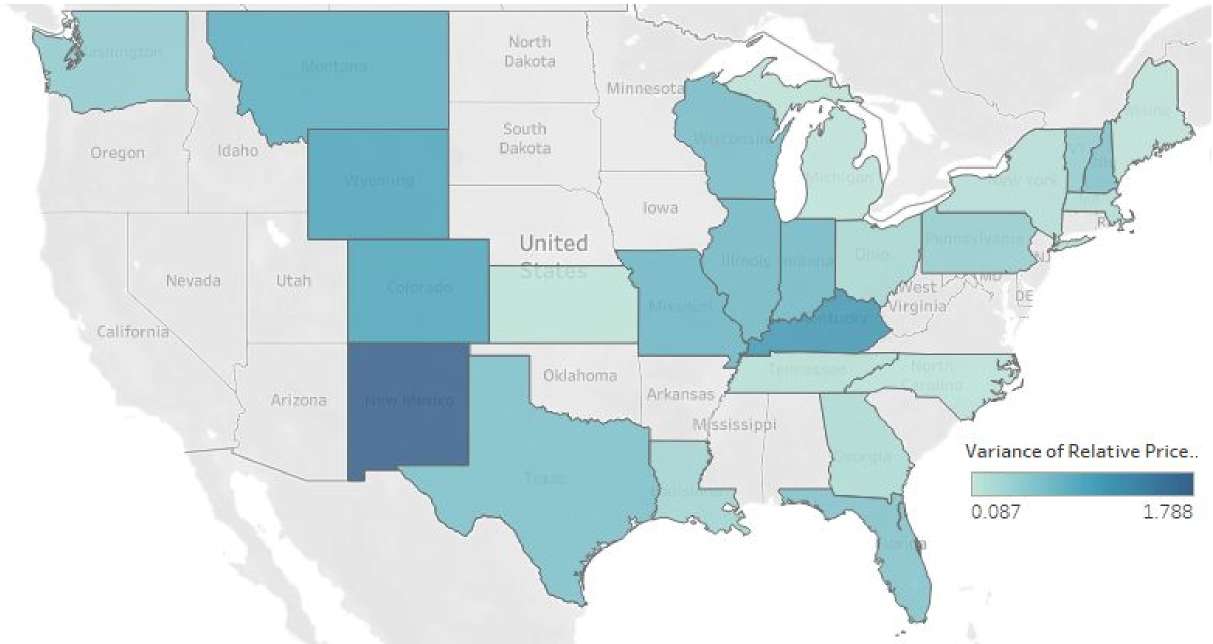
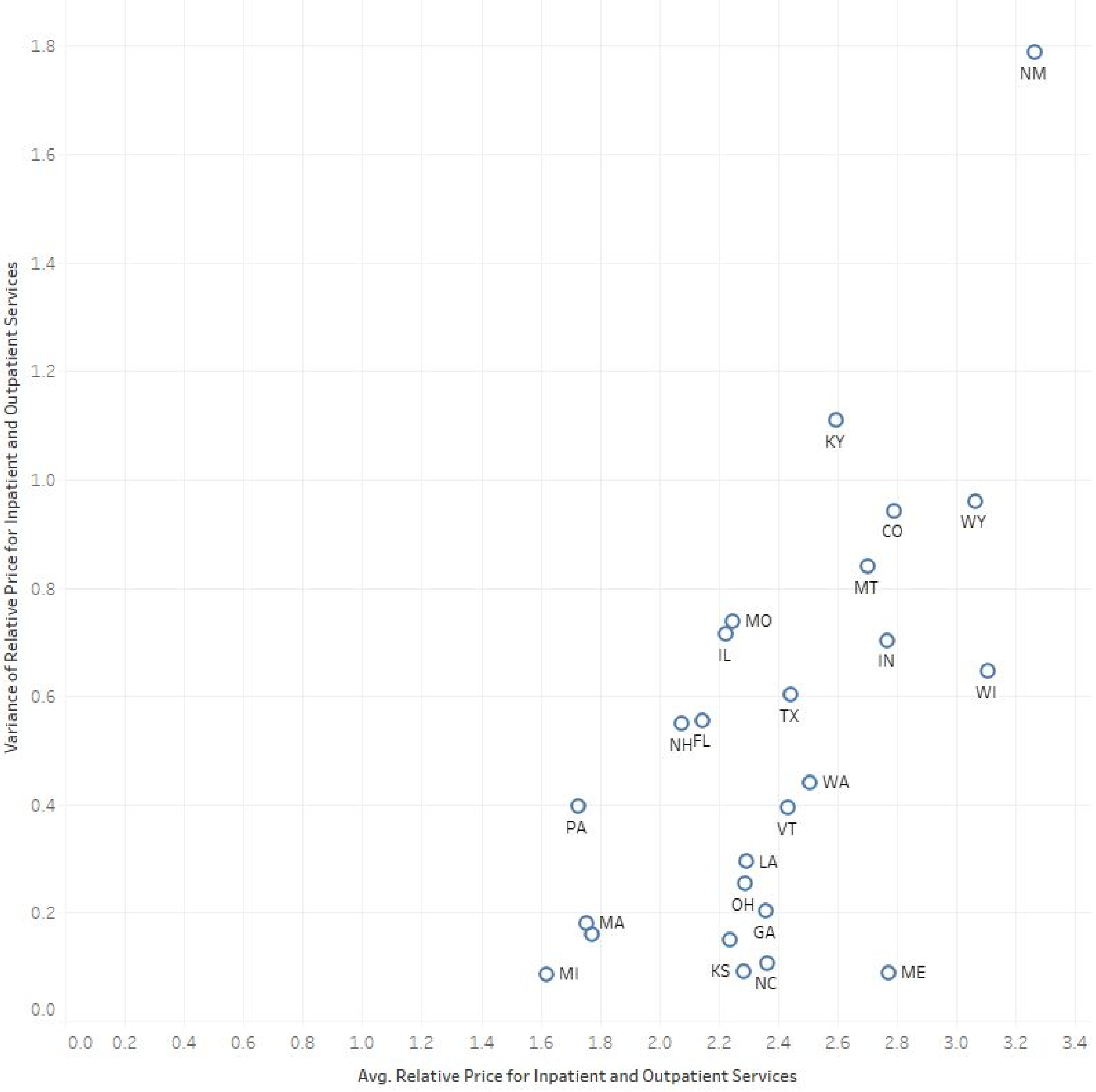


Figure 3: Variance of relative price VS Average relative price across states of the US



Variable selection/dimensionality reduction

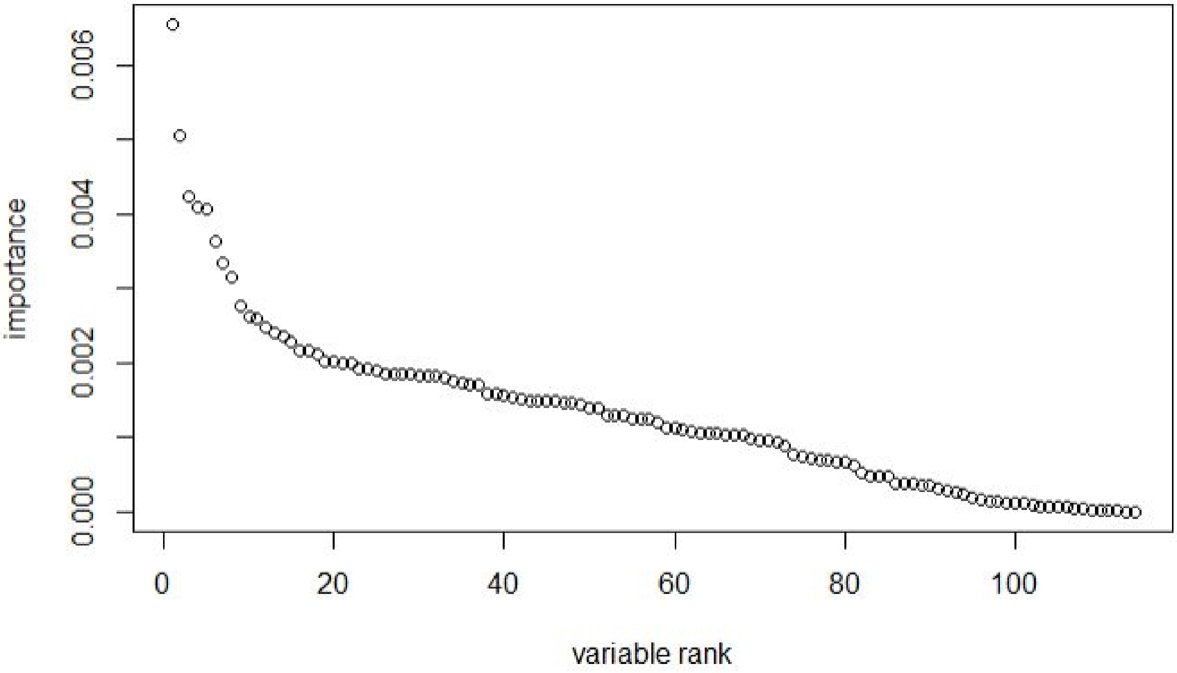
One of the challenges of dealing with this data was dealing with the mix of categorical and non categorical data. My first approach was to separate them, perform PCA on the non categorical to extract features, then to form a matrix of these principal components and the categorical. A random forest model predicting continuous cost yielded promising results with a significant plot of predicted vs given labels. This approach was friendly to R's RandomForest package since it nicely dealt with categorical variables. For 5-bin classification, it yielded an accuracy of 33% and *FlipError* of 20%. The code is located at *2_PCA_on_non_cat.Rmd.*

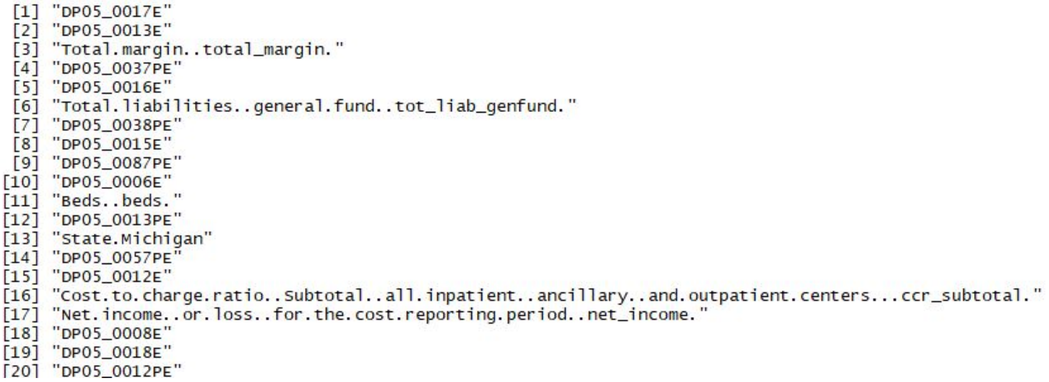
In order to use other models, I needed to experiment with different methods of categorical encoding. This led me to use dummy variables, then form a matrix

[*categorical* | *dummy_encoded_non_categorical*].

Further winnowing of these variables was then performed by training a Random forest model on cleaned data to predict Relative Inpatient/Outpatient cost. The model kept track of variable importance, and this is shown in figure 4 below. With this plot, an initial cutoff for selection of the most important variables was chosen(around 60), then the model iteratively updated until the adjusted R^2 of a re-trained random forest model stayed the same. The top 20 variables are shown as well. (code: 4_no_PCA.Rmd)

Figure 4: Variable Importance





```
 [1] "DP05_0017E"
 [2] "DP05_0013E"
 [3] "Total.margin..total_margin."
 [4] "DP05_0037PE"
 [5] "DP05_0016E"
 [6] "Total.liabilities..general.fund..tot_liab_genfund."
 [7] "DP05_0038PE"
 [8] "DP05_0015E"
 [9] "DP05_0087PE"
[10] "DP05_0006E"
[11] "Beds..beds."
[12] "DP05_0013PE"
[13] "State.Michigan"
[14] "DP05_0057PE"
[15] "DP05_0012E"
[16] "Cost.to.charge.ratio..Subtotal..all.inpatient..ancillary..and.outpatient.centers...ccr_subtotal."
[17] "Net.income..or.loss..for.the.cost.reporting.period..net_income."
[18] "DP05_0008E"
[19] "DP05_0018E"
[20] "DP05_0012PE"
```

Worst_cols was defined as those columns which didn't improve the R^2 of the random forest model. Similarly, *Best_cols* were the minimum columns needed. So as to not lose any information, *Worst_cols* was not discarded. More information was squeezed out of it.

Dimensionality reduction was then achieved by separately performing PCA on *worst_cols* and *best_cols*, then binding their principal components to form

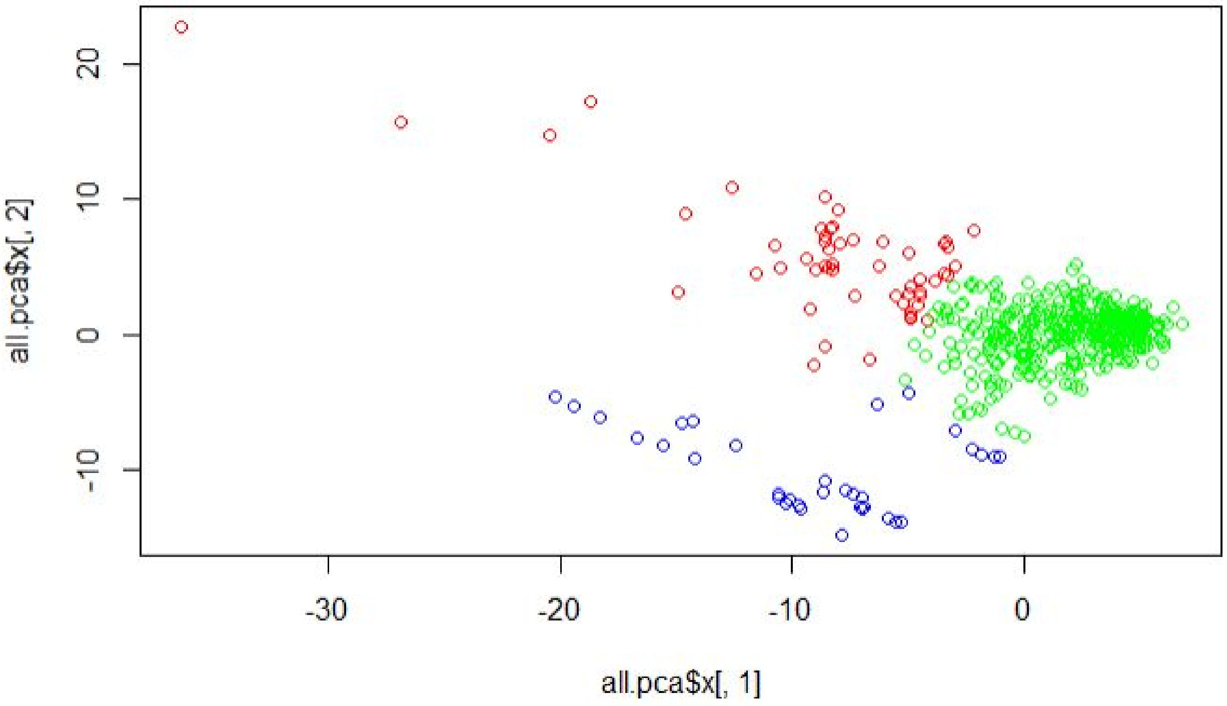
predictor_matrix= [bestPC's worstPC's].

The numbers of PC's used were later tuned to optimize hospital ranking during modeling. (Code: 5_PCA_on_BEST_and_WORST_cols.Rmd)

Setting *predictor_matrix* aside, another feature generated as follows. PCA was performed on all columns of the initial dataset, then k-means clustering was used to extract groups of *similar* hospitals. The number of groups was initialized then later tuned to optimize hospital ranking. This grouping column was then bound to the other predictors, so

predictor_matrix = [bestPC's worstPC's groups].

Figure 5: A plot of the first 2 PC's showing the 3 groups discovered by kmeans



Model Creation

With *predictor_matrix,* various models were trained to classify cost into 5 equally-sized buckets, and evaluated by ranking-accuracy of test data (for code please refer to *6_kmeans.rmd*). For this purpose the metric, *FlipError,* was defined as the fraction of test points which were classified further than an adjacent bucket.

$$FlipError = \sum_{i=1}^{n} \{|predict_i - ref_i| > 1\}$$

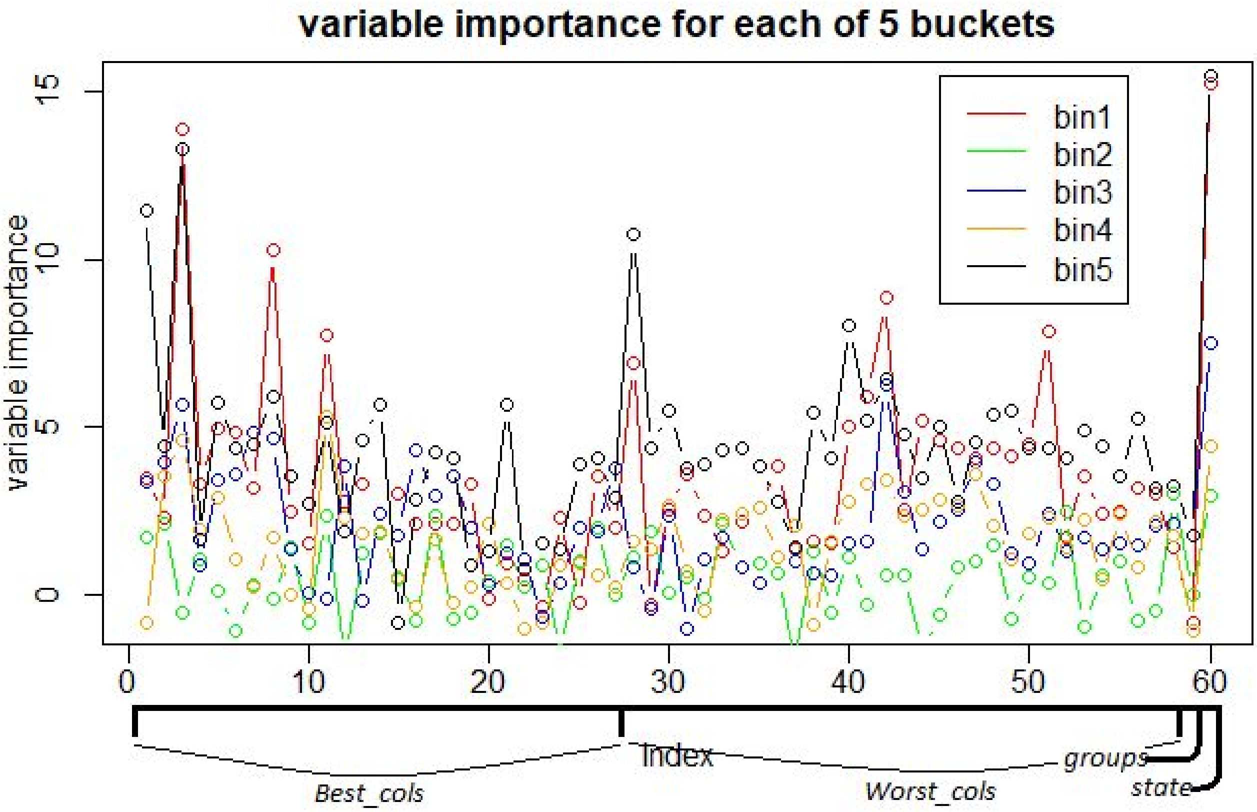
Table 1: Summary of results

	Model	Test *FlipError* (%)	Average bin classification accuracy (%)
1	Random Forest	21	36
2	Ridge (cross validated)	22	40
3	Lasso (cross validated)	25	41
4	KNN (k=7)	25	36

Interesting observations about these models

1. In the Random Forest model, a plot of variable importance to each bucket is shown below.The spike on the right at index=60 corresponds to the state column which was added to *predictor_matrix* = [bestPC's worstPC's groups *state*] only for this random forest model. Other models were content to gain state information from PCA. At index=59, one may observe that the *groups* variable has most benefited bin 5 in this model. It is also interesting to note that *Worst_cols* was important to some bins.

Figure 6: A comparison of predictor importance for classification



2. These models use as predictors, PC's of *Best_cols* and *Worst_cols*. In a previous strategy, PCA was performed on the entire data-set. The result was a higher random forest average classification accuracy, but lower ranking efficiency (higher *FlipError).* This was rejected since the business problem prioritized hospital ranking over high mean classification accuracy. (Code: 3_PCA_on_all.Rmd)

Stacking with H2O

Stacking, a useful ensemble modeling technique when base models are not highly correlated, reduced both *FlipError* and mean classification error by 2%. The 4 base models were selected based on differences in underlying math to achieve minimum prediction correlation. These were: Generalized Linear Models (GLM), Gradient Boosting Machine (GBM), Naive Bayes (NB), and Random Forest (RF). Table 2 below presents the correlation matrix of base model predictions, with coefficients in the 80's showing an opportunity for a boost in predictions through stacking. For code, please see *7_stacking.Rmd.*

Table 2: Correlation matrix of base model predictions

	my_glm_pred	my_rf_pred	my_gbm_pred	my_nb_pred
my_glm_pred	1.00	0.93	0.91	0.87
my_rf_pred	0.93	1.00	0.96	0.86
my_gbm_pred	0.91	0.96	1.00	0.85
my_nb_pred	0.87	0.86	0.85	1.00

Table 3: Performance of Base models and Stacked Ensemble

Model	**Mean Classification Error (%)**	***FlipError (%)***
glm	59	23
rf	60	22
gbm	60	21
nb	65	32
ensemble	**57**	**21**

It is interesting to note that while glm has the lowest classification error, it does not claim the lowest *FlipError*.

Error Analysis of Stacked Model

So far only accuracy and *FlipError* have been discussed. While we have already seen that 20% of the test data has been classified beyond an adjacent bucket, here we will take a look at the distribution of degree of misclassification. Figure 7 presents a summary of this distribution, where degree can take the values of integers [0,1,2,3,4]. A test point of Degree 0 has been correctly classified into its bucket. A test point of Degree 1 misclassification has been placed in a bucket adjacent to its true bucket. *FlipError* gives the proportion of data of degrees 2,3 and 4. There is no bar for degree 4 in figure 7 since no test points were misclassified to this extreme.

Figure 7: The distribution of misclassification degree ranging 0-4

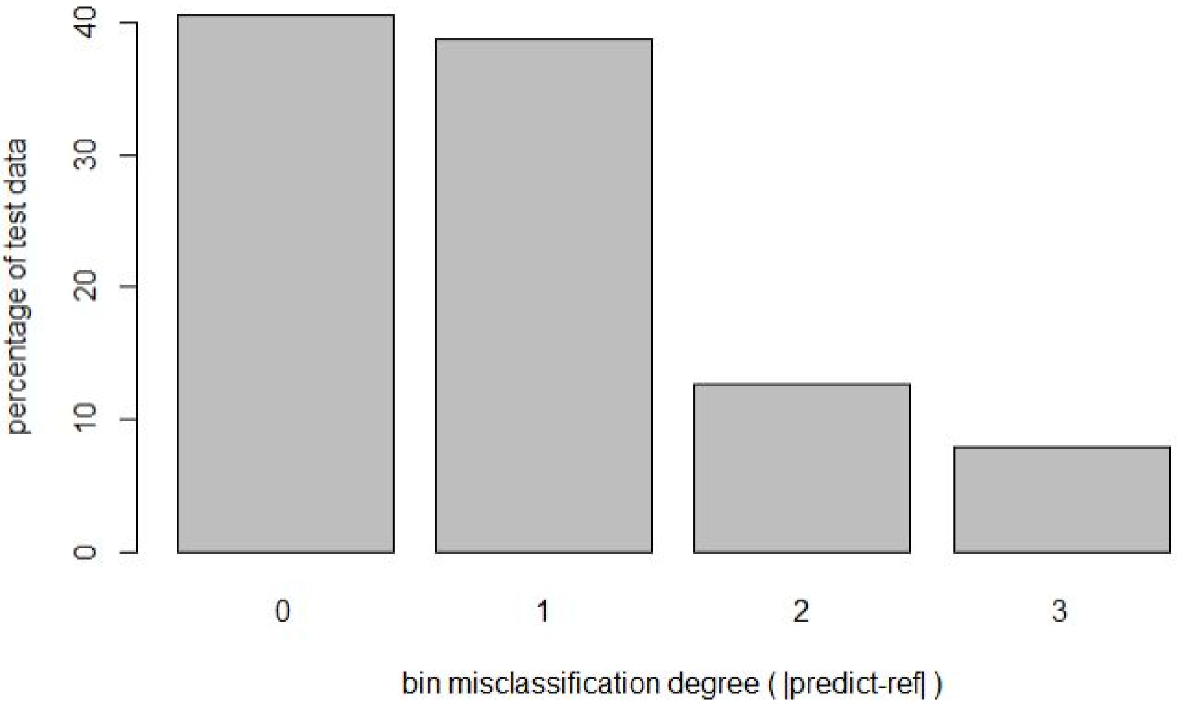
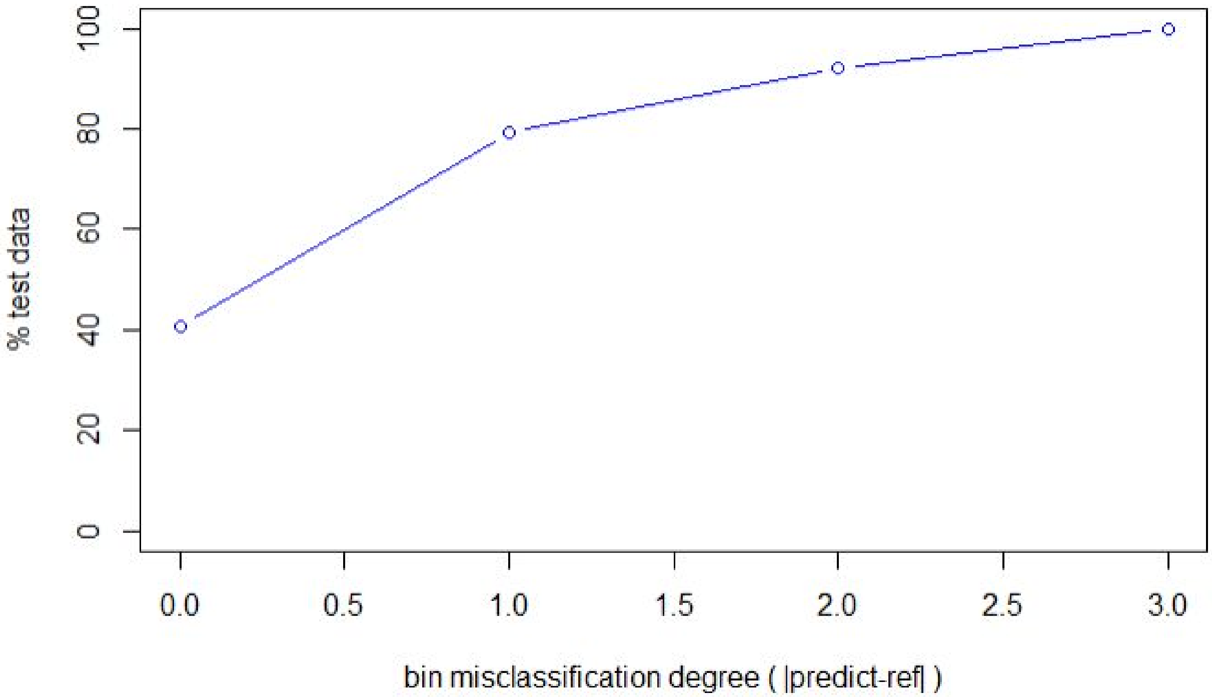


Figure 8 provides an alternate way of accounting for flip errors in test data. We see that 40% of data was correctly classified, and that 80% was at most 1 bucket away from home.

Figure 8: Cumulative accountability of test data misclassification degree



We have remarked several times on the clues leading to the conclusion that there is a high dependency on state. Figure 9 summarizes each state's contribution to the *FlipError* for the purpose of adding to these clues. Texas and Indiana seem particularly problematic. However, when a state's contribution to error is normalized by the number of available points in test data, this seems to explain these anomalies with TX and IN. Instead NC emerges as a problematic state. In order to rule out that a poor state-wise train/test pseudorandom split was not to blame, Figure 11 presented the state-wise contribution to flip error versus n_{test}/n_{train} ratio. As there were no states in the top right corner, this proposition was rejected.

Figure 9: Contribution of each state to *FlipError*

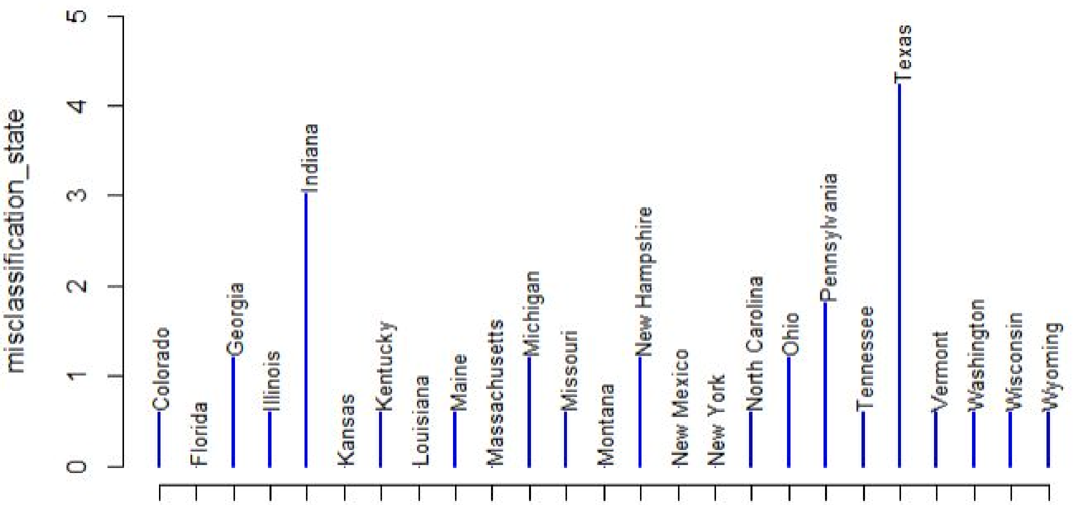


Figure 10: Contribution of each state to *FlipError,* normalized by the number of available state test points

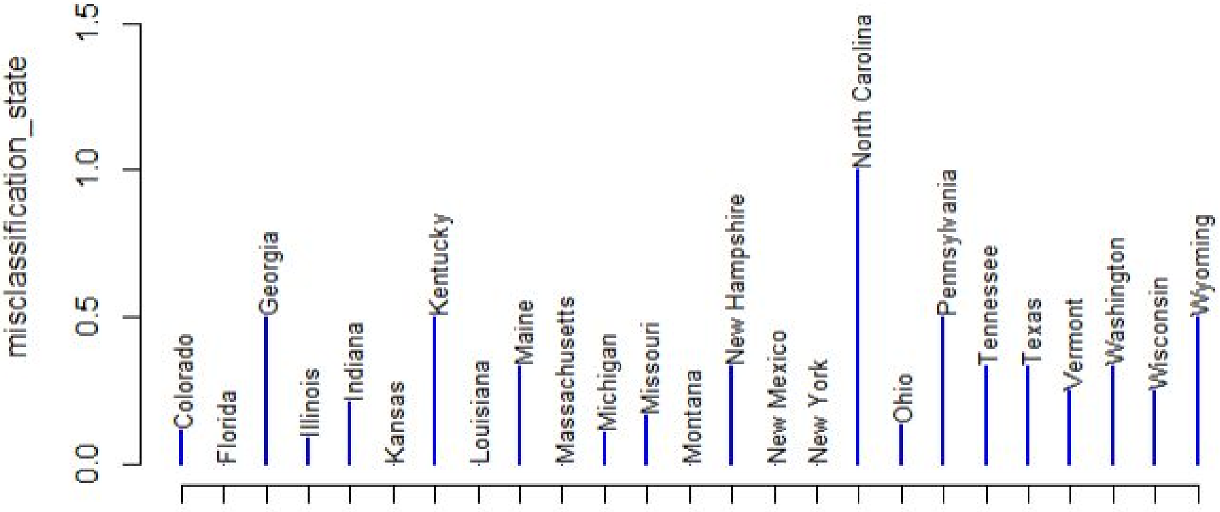
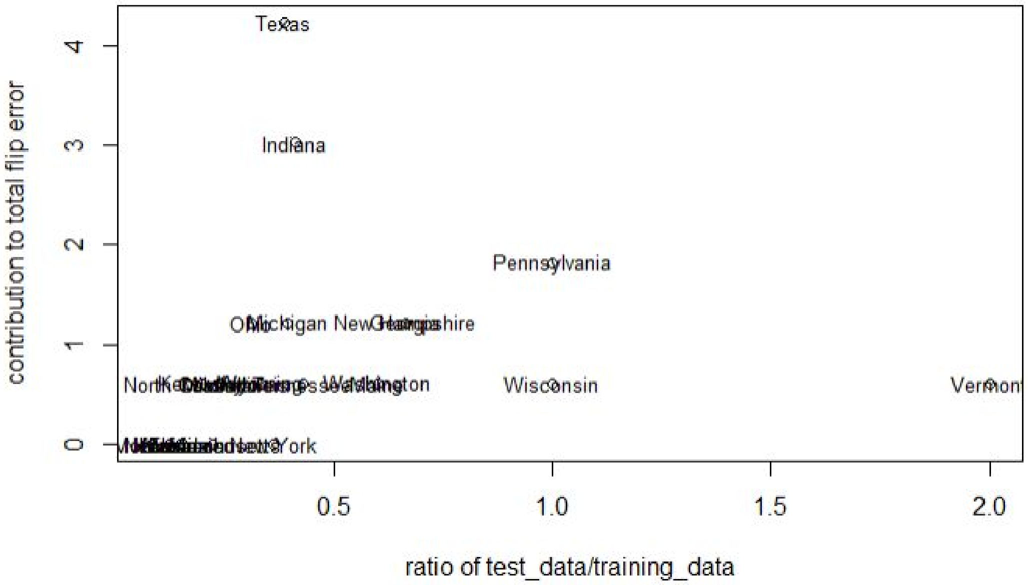


Figure 11: $\frac{n_{test}}{n_{train}}$ ratio is not the culprit of any state's high error contribution



Greg’s model creation and Error Analysis

Starting with GregsData.csv, an attempt was made to find the best subset of predictors for the Relative Price for Outpatient Services (RPO). The reason this variable was chosen instead of Relative Price for Inpatient and Outpatient Services (RPIO) was there exists more labeled data points to train on - 1577 for RPO as opposed to 798 for RPIO. The reason for this is that we hoped to obtain better predictions for more hospitals overall.

Variable Selection

There are 66 variables to work with in GregsData.csv. However, not all of these are useful in the modeling process. Some automated feature selection methods were performed to help us choose a subset of these variables. First, correlation analysis was used to see if any of the numeric predictors were highly correlated with RPO. The results of this are displayed visually:

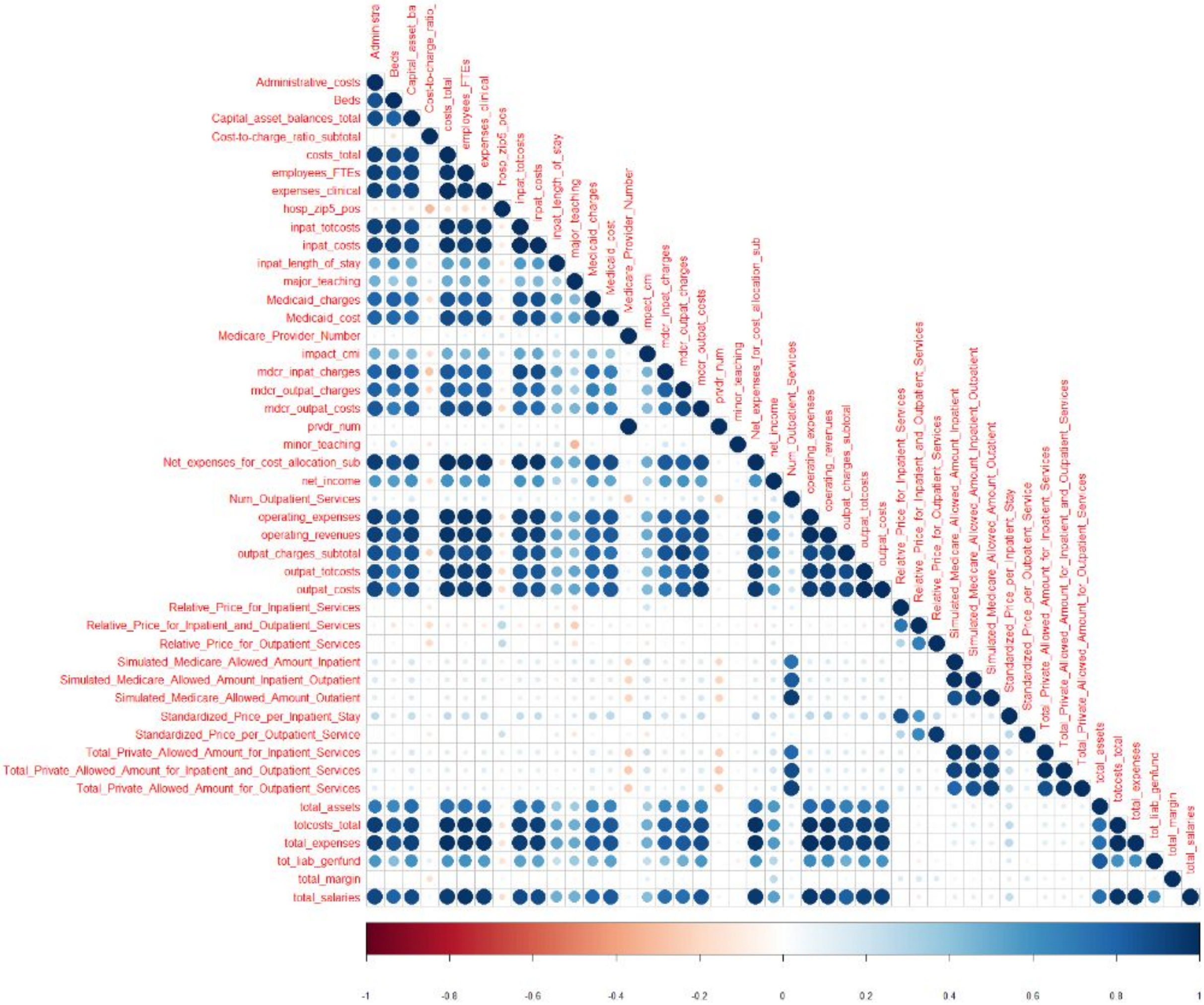


Figure 1: Correlation Plot of Continuous Variables

In general it is best to avoid features with a high correlation to RPO, as this may lead to problems in the modelling process such as multicollinearity. The predictors that are highly correlated with RPO are RPIO, Standardized Price per Inpatient Stay and Standardized Price per Outpatient Service. The choice was made to eliminate these variables from consideration.

Next, automated feature selection available within the R Caret package was applied. With RPO as the response and all other variables as predictors, the output of the feature importance rating is as follows:

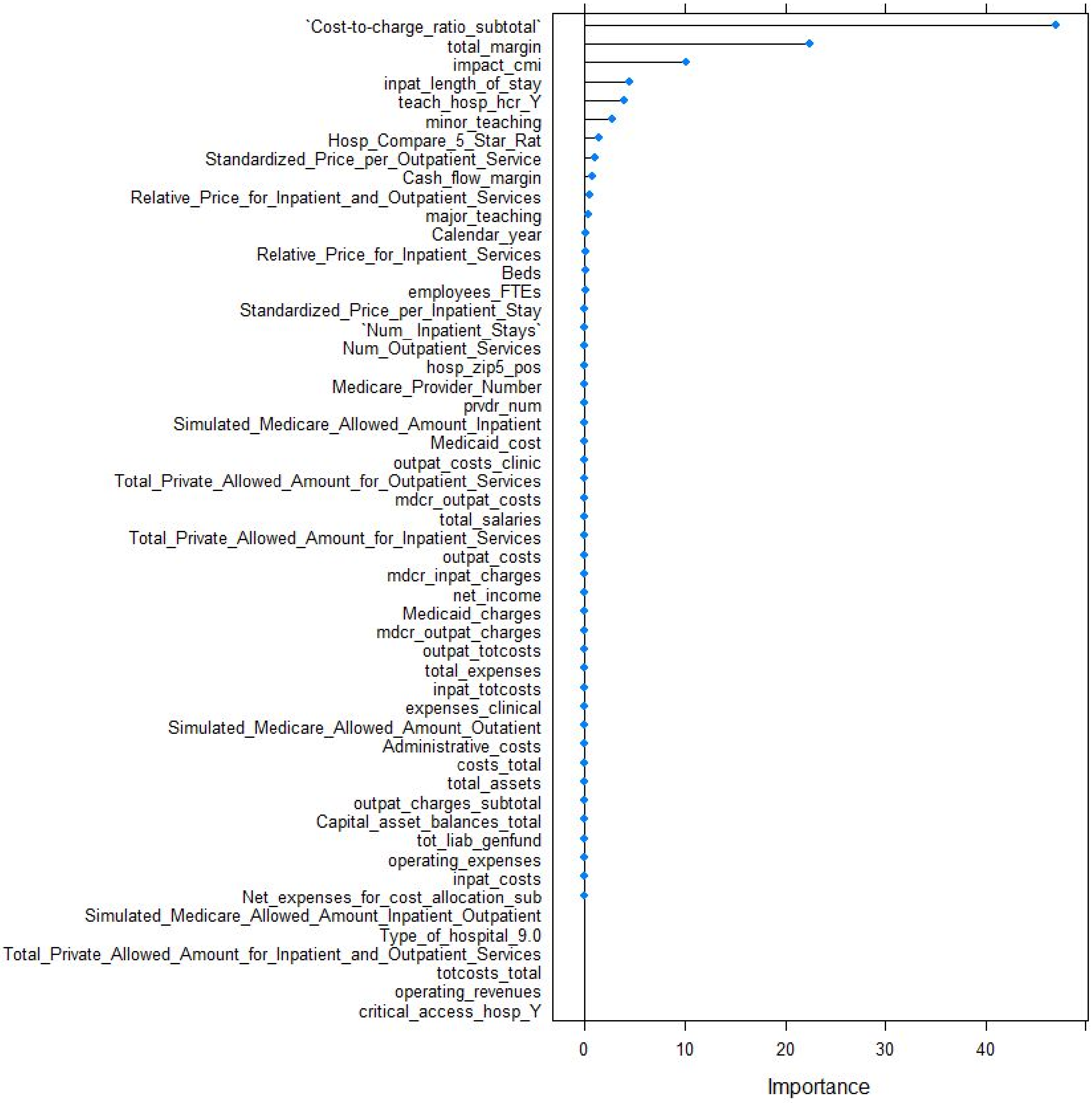


Figure 2: Automated Variable Importance Test from the Caret R Package

The first seven “important” predictors for inclusion in our variable list were selected, and the rest of the variables have been deemed not important. However, there is one caveat here: To run both of the above analyses, all missing values in our dataset needed to be discarded, and also the non-numerical features were discarded. To perform the final variable selection for use in modeling, the rest of the variables in question were studied and a subset of variables was selected based on the amount of data present and whether it was thought to be a valuable predictor. Through this process 17 additional predictors were identified, to bring the grand total to 24. The specific list of variables chosen is as follows:

Final Variables Chosen for Modeling

1. "Administrative_costs"
2. "Beds"
3. "Capital_asset_balances_total"
4. "costs_total"
5. "employees_FTEs"
6. "Medicaid_charges"
7. "Medicaid_cost"
8. "impact_cmi"
9. "mdcr_inpat_charges"
10. "mdcr_outpat_charges"
11. "net_income"
12. "operating_expenses"
13. "operating_revenues"
14. "outpat_costs"
15. "total_assets"
16. "total_expenses"
17. "tot_liab_genfund"
18. "teach_hosp_hcr_Y"
19. "Hosp_Compare_5_Star_Rat"
20. "Cost.to.charge_ratio_subtotal"
21. "total_margin"
22. "inpat_length_of_stay"
23. "minor_teaching"
24. "Cash_flow_margin"

Initial Model Creation

Using the above variables as predictors, models were constructed with Relative Price for Outpatient Services (RPO) as the response variable. RPO was binned into both 3 and 5 buckets such that each bucket had roughly the same number of observations, and models were created for both bucket versions. Additionally, the data was subset to only include 2017, and it was further split 70/30 into train/test sets.The 24 predictors used are listed in the appendix, and

10-fold cross-validation was used in each of the models.The initial models chosen were as follows:

1. Multinomial logistic regression (GLM)
2. Random Forest, with 2 different sets of parameters (RF and RF2)
3. Gradient Boosted Machine (GBM)

These models were then combined into one ensemble which would then give the predictions. Aside from accuracy, it was important to keep track of the serious misclassifications, i.e. classifying bin 3 as bin 1 and vice versa, and for 5 bins serious misclassifications were considered as being anything more than 1 bin away from the actual value. We recall the *FlipRate* from before: this is defined as the fraction of test points which were classified further than one adjacent bucket.

3 bins:

Model	**Accuracy**	**Flip Rate**
GLM	57.9%	8.1%
RF	59.0%	7.2%
RF II	59.0%	6.1%
GBM	59.0%	7.4%

5 bins:

Model	**Accuracy**	**Flip Rate**
GLM	40.1%	28.2%
RF	35.7%	22.9%
RF II	39.0%	21.8%
GBM	35.7%	22.9%

First Error Analysis and Model Refinement

Next, the flip errors were analyzed geographically to see if there were any improvements that can be made. In the following plot, the percent of data points that each state contributes to the training set is compared against the percentage of flip rate errors that are produced by the state. States on or above the black line are considered “normal,” and states under the red line are considered “anomalous” since they are responsible for more errors than they should be:

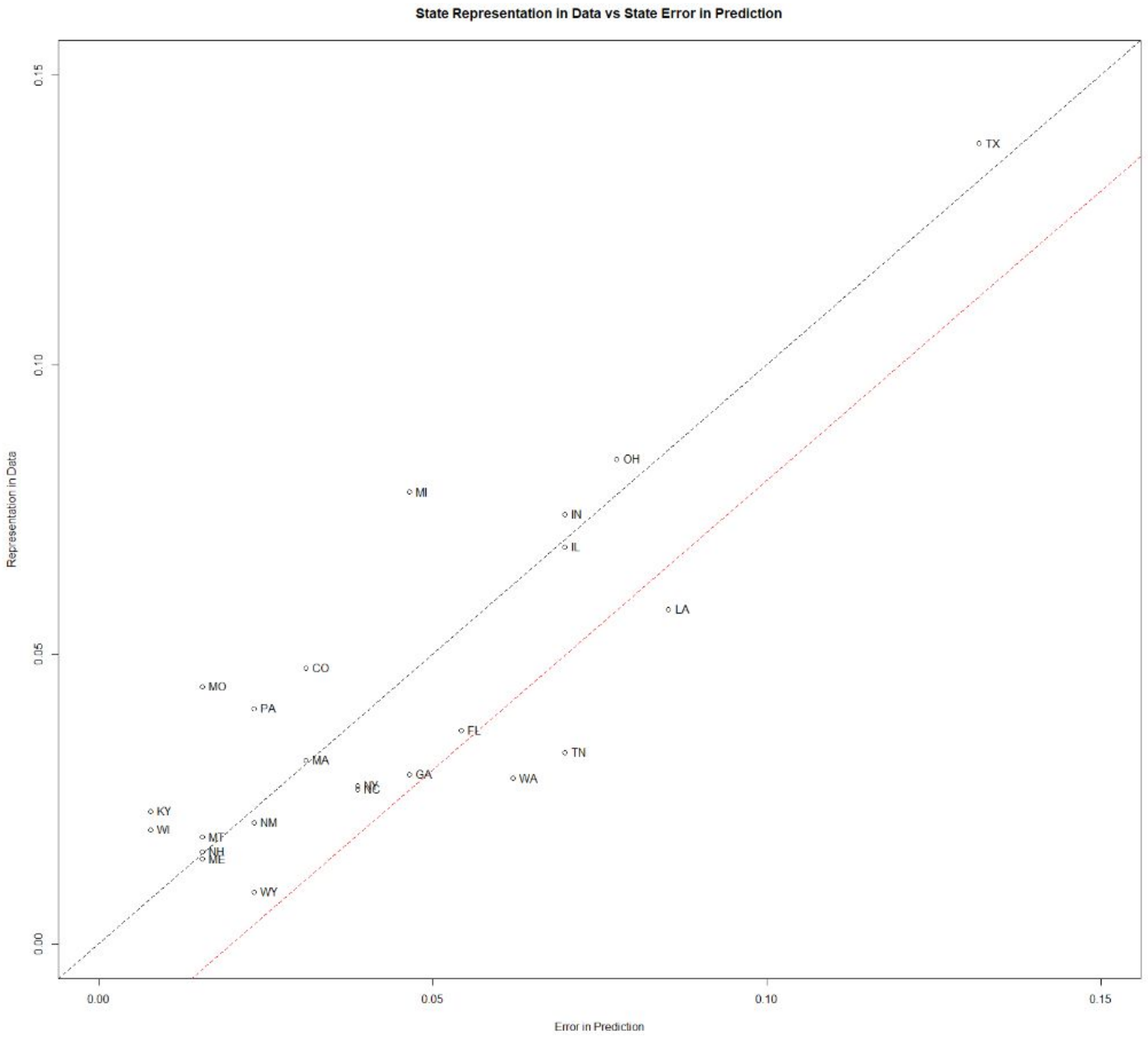


Figure 3: Percent Representation in Data vs Error in Prediction for each US State (5 bins)

Final Model

Using the above framework, it was determined that Louisiana, Tennessee and Washington were anomalous, and hence two separate models were built: one with just LA, TN, and WA data, and then the rest of the data with those states removed. The predictions for these two models were then combined, and better performance overall was obtained .

Using the same general set of models as in the initial modeling round, the following results were obtained:

3 bins, with LA, TN and WA data:

Model	**Accuracy**	**Flip Rate**
GLM	44.3%	16.4%
RF	55.7%	6.6%
RF II	55.7%	4.9%
GBM	55.7%	1.6%

3 bins, with the remaining data:

Model	**Accuracy**	**Flip Rate**
GLM	56.4%	8.5%
RF	60.0%	7.2%
RF II	58.6%	6.3%
GBM	60.0%	7.8%

It is clear which models should be combined: the GBM classifier for the first set of data is either equal to or better than the other models, and the first RF model for the second data is similar. Combining these together yields a 6.99% flip error rate.

5 bins, with LA, TN and WA data:

Model	**Accuracy**	**Flip Rate**
GLM	27.9%	20.0%
RF	45.9%	18.0%
RF II	42.6%	20.0%
GBM	45.9%	18.0%

5 bins, with the remaining data:

Model	**Accuracy**	**Flip Rate**
GLM	33.8%	26.8%
RF	33.6%	21.7%
RF II	36.0%	22.1%
GBM	33.6%	22.9%

In the 5 bucket case, it is less clear which models to select. For the LA, TN and WA data the best performing model on both metrics is RF. In the remaining data case, we have chosen to minimize the flip error rate instead of maximize accuracy. Thus again, the RF model is selected. Combining these predictions together yields a flip error rate of 21.2%. This is considered the most promising model overall, as nearly 79% of hospitals are labeled correctly to within one bin.

Conclusions

1. All of our models placed high importance on the following RAND variables: Total margin, Cost to charge ratio, beds. More details about these variables:
    - Total_margin is a derived variable, coming from the ratio of net income to the sum of net patient revenue and all other income. It is a measure of profitability of the hospitals.
    - ‘Beds’ measures the inpatient capacity of the hospital.
    - The cost-to-charge ratio is the costs of the procedure divided by the amount charged to patients, for all inpatient, ancillary, and outpatient centers. This is a measure of the markup associated with hospital procedures.
2. Despite somewhat different approaches, both of our best-performing models obtained around 21% flip error on our test data for the 5-bin case.
3. Incorporating State-level differences when creating models. Both of our results showed that State was an important factor in model performance, so when building future models it is recommended to incorporate State in some way.
4. Here are the variables from the census that proved most important.

DP05_0013E	Estimate!!SEX AND AGE!!55 to 59 years
DP05_0015E	Estimate!!SEX AND AGE!!65 to 74 years
DP05_0016E	Estimate!!SEX AND AGE!!75 to 84 years
DP05_0017E	Estimate!!SEX AND AGE!!85 years and over
DP05_0037PE	Percent!!RACE!!One race!!White
DP05_0038PE	Percent!!RACE!!One race!!Black or African American

5. Various forms of imputation (median, MICE) were used to deal with the missing values but they did not lead to a meaningful increase in accuracy or decrease in the flip error rate.

Further areas for exploration:

1. Model stacking: Attempts were made to use model stacking in an effort to improve the overall performance of the individual models, but this ended up yielding very marginal performance gains. This may either be due to our inexperience with how stacking works, or our inexperience with the h2o package in R which the predictive models were built. More time could be spent in this area to determine models that are more appropriate for stacking, potentially leading to an increase in performance.
2. The original plan was to stack our models together to harness the power of diversity in approaches for yielding less correlated results. However, we ran into problems stacking models built from different datasets. Ideally we would like to figure out how to make this work.

3. More Ranking Algorithms. We ranked the hospitals by classifying into 5 buckets with equivalent numbers of points, then allowing misclassification into adjacent buckets. It would be beneficial to experiment with more complex algorithms, such as LambdaMART.
4. All of our models exhibited high ranking errors for a few states. While for some states we attributed this to their high contribution to test data, it would be ideal to investigate these states individually. Perhaps there are more explanatory variables, or perhaps there are discrepancies in the data for these States.
5. Incorporate recently released RAND data.

Appendix

Columns of RAND Dataset chosen for analysis

1. Hospital is a Critical Access Hospital (CAH) (Y/N) (as reported in Provider of Services file) [critical_access_hosp_pos]
2. Hospital name (as reported in Provider of Services file) [hosp_name_pos]
3. Hospital street address (as reported in Provider of Services file) [hosp_street_address_pos]
4. Hospital 5-digit zip code (as reported in Provider of Services file) [hosp_zip5_pos]
5. Medicare provider number [prvdr_num]
6. Calendar year [cy]
7. Compendium of US Health Systems 2018 health system name [health_sys_name]
8. Type of hospital (as reported in the hospital cost report) (1=general short-term, 2=general long-term, 3=cancer, 4=psych, 5=rehab, 6=religious nonmedical, 7=childrens, 8=alcohol and drug, 9=other) [type_hosp]
9. Beds [beds]
10. Critical access hospital? (Y/N) (as reported in the hospital cost report) [critical_access_hosp_hcr]
11. Employees on payroll (full-time equivalents) [employees_FTEs]
12. Medicare casemix index (CMI) from CMS hospital impact files [impact_cmi]
13. Is this a hospital involved in training residents in approved GME programs? [teach_hosp_hcr]
14. Inpatient length of stay [inpat_length_of_stay]
15. Major teaching hospital (interns and residents-to-bed>0.25) [major_teaching]
16. Minor teaching hospital (interns and residents-to-bed between 0 and 0.25) [minor_teaching]
17. Administrative costs (admin and general + nursing admin + medical records) [admin_costs]
18. Cash flow margin [cash_flow_margin]
19. Total margin [total_margin]
20. Total liabilities, general fund [tot_liab_genfund]
21. Total liabilities, plant fund [tot_liab_plantfund]
22. Total liabilities, special fund [tot_liab_specfund]
23. Total assets [total_assets]
24. Medicare outpatient charges [mdcr_outpat_charges]
25. Medicare outpatient costs [mdcr_outpat_costs]
26. Medicare inpatient charges [mdcr_inpat_charges]
27. Net income (or loss) for the cost reporting period [net_income]
28. Operating expenses [operating_expenses]
29. Operating revenues [operating_revenues]
30. Total salaries [total_salaries]

31. Expenses for clinical activities [expenses_clinical]
32. Total expenses [total_expenses]
33. Medicaid charges [mdcd_charges]
34. Medicaid cost [mdcd_costs]
35. Costs,Total (all inpatient, ancillary, and outpatient centers) [costs_total]
36. Net expenses for cost allocation, Subtotal (sum of general service, inpatient routine, ancillary, outpatient, other reimbursable, and special purpose cost centers) [net_expenses_subtotal]
37. Total costs, reimbursable and nonreimbursable cost centers [totcosts_total]
38. Outpatient costs, Clinic [outpat_costs_clinic]
39. Outpatient charges, Subtotal (all Outpatient, ancillary, and outpatient centers) [outpat_charges_subtotal]
40. Cost-to-charge ratio, Subtotal (all inpatient, ancillary, and outpatient centers) [ccr_subtotal]
41. Inpatient costs (sum of 63 cost centers) [inpat_costs]
42. Inpatient costs (incl Medicare disallowed) (sum of 63 cost centers) [inpat_totcosts]
43. Outpatient costs (sum of 52 cost centers) [outpat_costs]
44. Outpatient costs (incl Medicare disallowed) (sum of 52 cost centers) [outpat_totcosts]
45. Capital asset balances, total (subtotal minus reconciling), beginning balance [capasset_total_beginning]

Mary chose a subset of the above variables with the following additional variables:

1. County name [county_name]
2. Ownership: Proprietary (for-profit) [ownership_forprofit]
3. Ownership: Voluntary (non-profit) [ownership_nonprofit]
4. State name [state_name]

File structure of project folder

```
Medxoom_teamA
                    MedxoomFinal_teamA.pdf
                    Code_Greg-------------------> medxoom_clean.R
                                                   GregsData.csv
                    Code_Mary-------------------> 1_joining_and_cleaning.Rmd
                                                   2_PCA_on_non_cat.Rmd
                                                   3_PCA_on_all.Rmd
                                                   4_no_PCA.Rmd
                                                   5_PCA_on_BEST_and_WORST_cols.Rmd
                                                   6_kmeans.Rmd
                                                   7_stacking.Rmd
                                                   census_columns.csv
                                                   census_county.csv
                                                   CuratedData.csv
                                                   HCAHPS_Hospital_clean.csv
                                                   hospitals_labeled.csv
```

Note: There is an .html document accompanying every R markdown notebook (.Rmd)

How to Run Mary's code

From within folder Medxoom_teamA\Code_Mary, run each of the 7 Rmd files in numerical order.

How to run Greg's code

Currently the entire analysis can be run as one large script in R version 3.6.3 (2020-02-29) -- "Holding the Windsock", but segments of the code can also be run individually (e.g. for the 3 bin, 5 bin, 3 bin separate, 5 bin separate cases).

The R packages that are needed to reproduce the analysis are as follows:

1. readr
2. mice
3. Hmisc
4. caret
5. corrplot
6. h2o