ISE 5103

Intelligent Data Analytics Homework #7

Instructor: Charles Nicholson

(a) (75 points) Build at least 5 different classes of model's **logistic regression**, **MARS** (for classification), **decision tree**, **random forest**, **boosted trees**, **SVM**, **neural nets**. Each of your models with hyper -parameters should be tuned using a re-sampling method of your choice.

We tried logistic regression, MARS, decision tree, random forest, boosted trees.

The deliverable for this part has three components:

- **(20 points)** Choose one model (of your choice) and provide at least 3 potential "insights" relating to hospital readmits that might be of some use to hospitals, insurance companies, doctors, patients, and/or government administration.

From this XGBOOST prediction from which we got good predictions,

Let's see what kind of information helps more from this model.

```
xgbTree variable importance

  only 20 most important variables shown (out of 910)

                       Overall
visits                 100.000
num_lab_procedures      56.864
number_inpatient        44.960
num_medications         44.612
time_in_hospital        32.070
number_diagnoses        30.022
num_procedures          17.662
raceCaucasian            9.032
admission_source7        8.962
payer_code(missing)      8.610
diabetesMedYes           8.196
number_outpatient        8.074
age9                     7.884
diagnosis428             7.601
genderMale               7.593
medical_specialty73      7.170
age8                     7.110
discharge_disposition3   7.042
discharge_disposition14  6.741
number_emergency         6.731
```

I calculated visits from *number_outpatient, number_emergency, number_inpatient*

```
total_data$visits = total_data$number_outpatient +
total_data$number_emergency + total_data$number_inpatient
```

From here we can say what factors make more impact on readmitting. We can see the more time the patients are visiting the hospitals makes them to readmit more. Other than that, medications, number of diagnoses and procedures also impacting a lot. this

- helps hospitals, doctors to take intensive on patients who has special type of problems,
- helps government to get a review about hospitals, doctors and evaluate ranking based on their performances. Like if they're going to be are a greater number of visits/ readmitting where we can assume hospital is trying to make money.
- It also helps insurance companies to get information of people who has more probability of getting readmitting and doesn't have insurance and to get them insure.
-

- **(25 points)** Using at least 8 different types of performance evaluation techniques, quantify (and/or visualize) the predictive quality of the above model.

For Xgboost: Below information gives how we tuned and resampled with 3 folds

```
> modelxgboost
eXtreme Gradient Boosting

57855 samples
   42 predictor
    2 classes: 'no', 'yes'

No pre-processing
Resampling: Cross-Validated (3 fold)
Summary of sample sizes: 38570, 38571, 38569
Resampling results across tuning parameters:

  max_depth  colsample_bytree  Accuracy   Kappa
   1         0.75              0.6268777  0.2408520
   1         1.00              0.6263246  0.2397872
   2         0.75              0.6315100  0.2514941
   2         1.00              0.6320631  0.2529056
   3         0.75              0.6349842  0.2595585
   3         1.00              0.6346731  0.2589944
   5         0.75              0.6388559  0.2684120
   5         1.00              0.6395128  0.2697947
  10         0.75              0.6401177  0.2727306
  10         1.00              0.6401177  0.2730819

Tuning parameter 'nrounds' was held constant at a value of 200
Tuning parameter
 constant at a value of 0
Tuning parameter 'subsample' was held constant at a
 value of 0.75
Accuracy was used to select the optimal model using the largest value.
The final values used for the model were nrounds = 200, max_depth = 10, eta =
 0.05, gamma = 0, colsample_bytree = 0.75, min_child_weight = 0 and subsample = 0.75.
```

Confusion Matrix

One way to assess a classification model's performance is to use a "confusion matrix", which compares actual values (from the test set) to predicted values. Be careful though, the figures are highly dependent on the probability cutoff chosen to classify a record. Depending on your use case, you might want to adjust the cutoff to optimize a specific metric.

Keeping threshold as 0.5 and taking 0 and 1 and finding matrix

```
Confusion Matrix and Statistics

          Reference
Prediction    no   yes
       no  24958  8808
       yes  5674 18415

               Accuracy : 0.7497
                 95% CI : (0.7461, 0.7532)
    No Information Rate : 0.5295
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.4944

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.8148
            Specificity : 0.6765
         Pos Pred Value : 0.7391
         Neg Pred Value : 0.7645
             Prevalence : 0.5295
         Detection Rate : 0.4314
   Detection Prevalence : 0.5836
      Balanced Accuracy : 0.7456

       'Positive' Class : no


> cm$byClass
         Sensitivity          Specificity       Pos Pred Value       Neg Pred Value
           0.8147689            0.6764501            0.7391459            0.7644568
           Precision               Recall                   F1           Prevalence
           0.7391459            0.8147689            0.7751172            0.5294616
      Detection Rate Detection Prevalence    Balanced Accuracy
           0.4313888            0.5836315            0.7456095
```

Metrics definitions

-Accuracy

Proportion of correct predictions (positive and negative) in the sample. Which we got it as 0.74

-Precision

Proportion of correct “positive” predictions in the sample. Which we got it as 0.739

-Recall

Proportion of “positive” actual records correctly predicted as "positive". Which we got it as 0.81

-F1-score

Harmonic mean between precision and recall. More informative than Accuracy for unbalanced datasets. Which we got it as 0.77511

Kappa

The Kappa statistic (or value) is a metric that compares an Observed Accuracy with an Expected Accuracy (random chance). The kappa statistic is used not only to evaluate a single classifier, but also to evaluate classifiers amongst themselves. We got 0.4944, which is moderate.

Accuracy SD (Standard Deviation): 0.00410

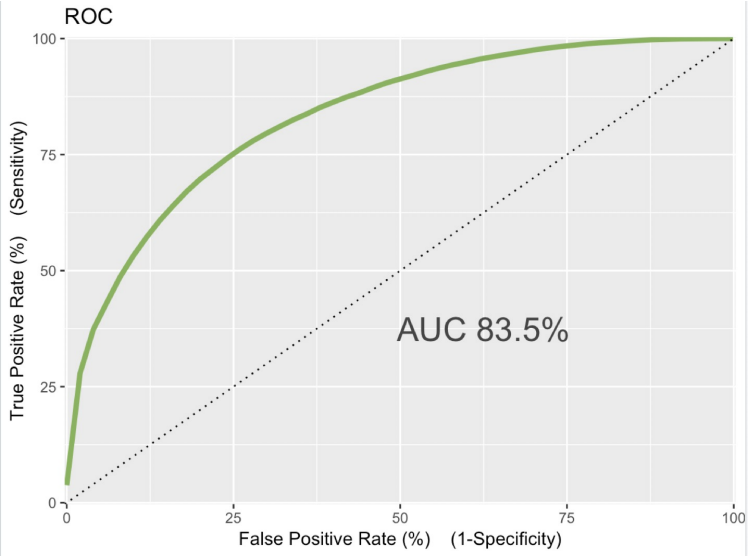
Kappa SD (Standard Deviation): 0.00824

ROC curve

The Receiver Operating Characteristic (or ROC) curve shows the true positive rate vs. the false positive resulting from different cutoffs in the predictive model. The "faster" the curve climbs, the better it is.

On the contrary, a curve close to the diagonal line is worse.

The AUC (Area Under the Curve) for this model is 0.83, which is **fair**.



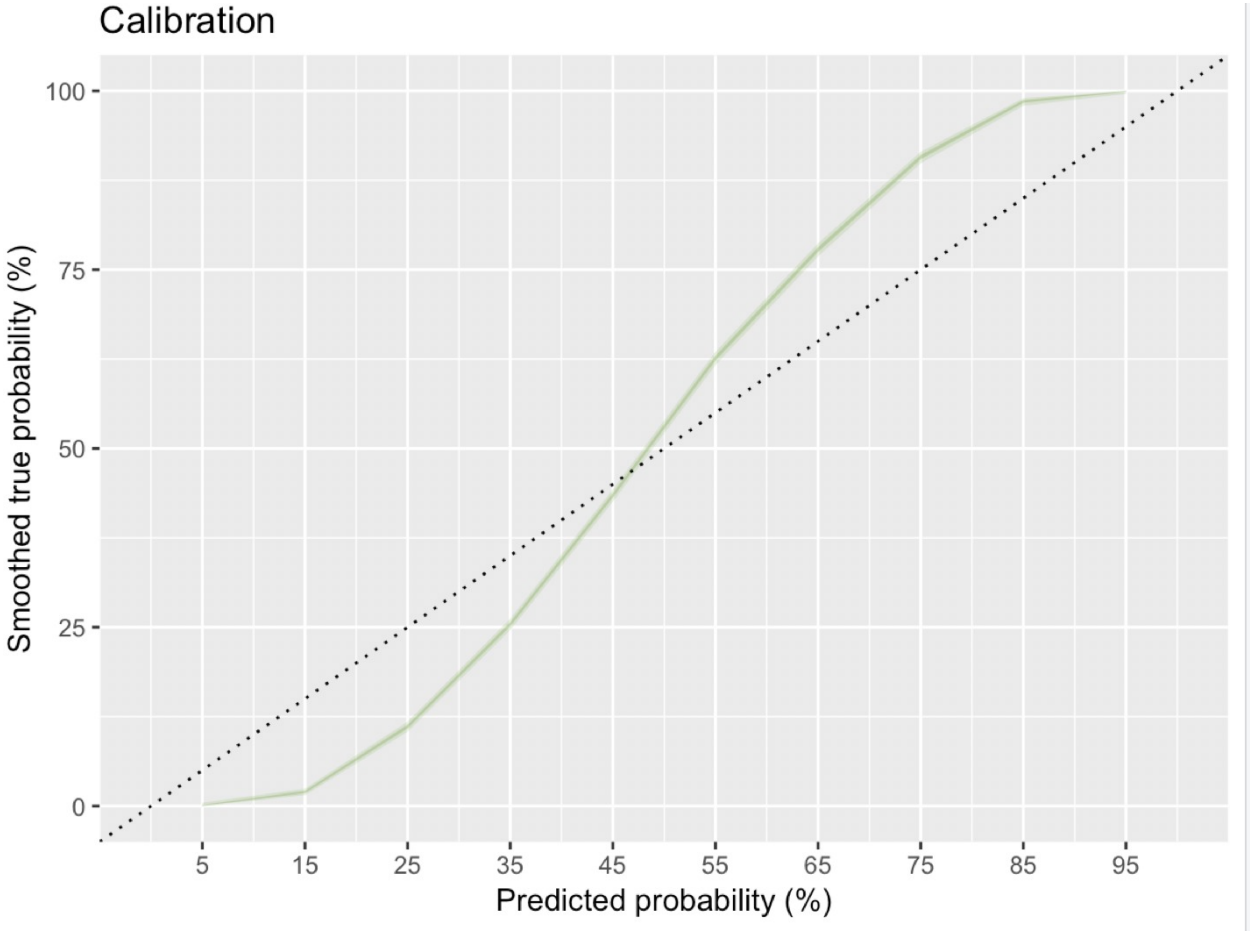
Log Loss

Error metric that considers the predicted probabilities (the lower the better). We got 0.5345

Calibration curve

Calibration denotes the consistency between predicted probabilities and their actual frequencies observed on a test dataset.

A perfectly calibrated model, should have a calibration curve that is exactly on the diagonal line.



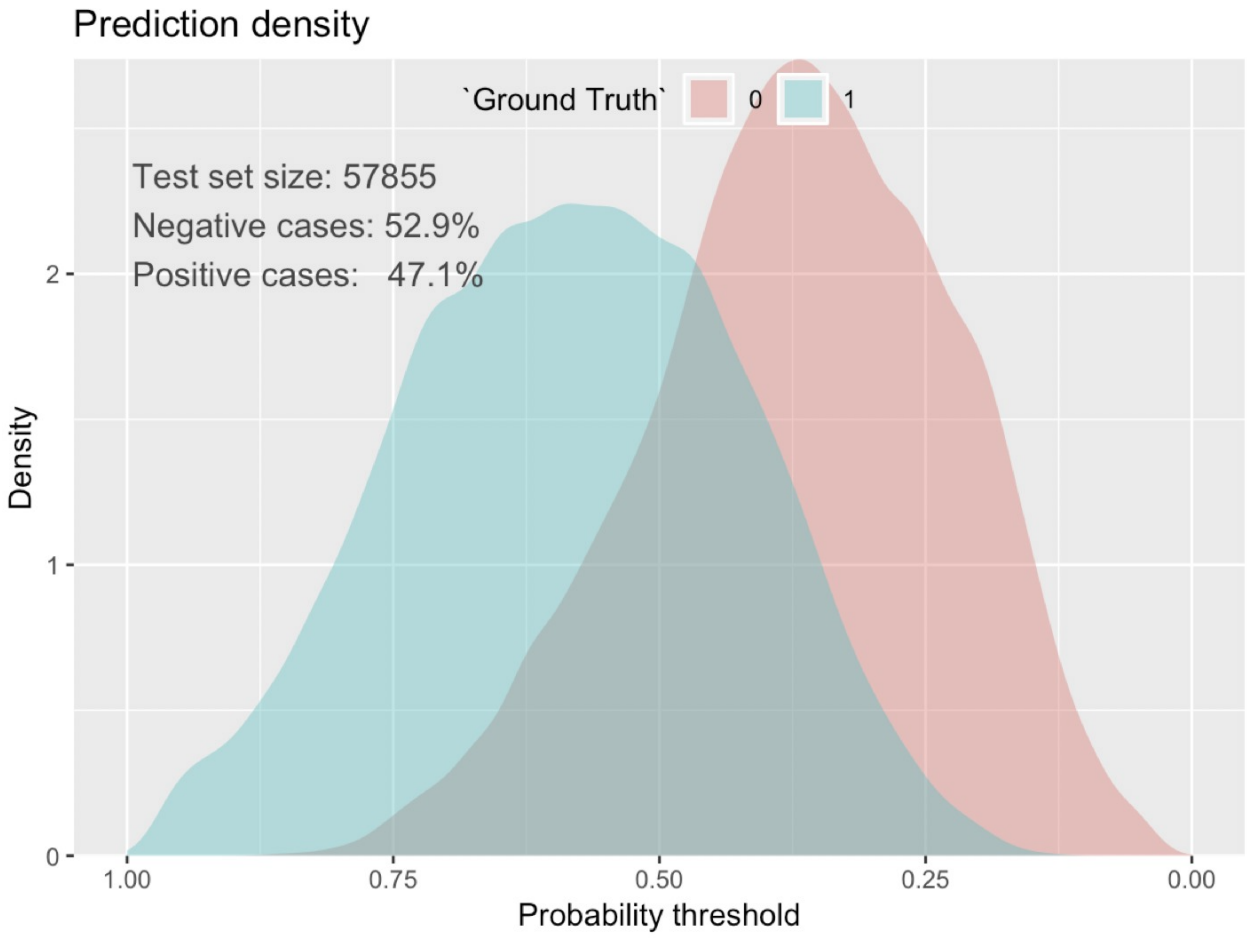
Density chart

This density chart illustrates how the model succeeds in recognizing (and separating) the classes (e.g. 1 and 0 for binary classification). It shows the repartition of the actual classes in the validation set according to the predicted probability of being of said class learnt by the model.

The two density functions show the probability density of rows in the validation set that actually belong to the observed class vs rows that don't.

A perfect model fully separates the density functions:

- the colored areas should not overlap
- the density function of `0` should be entirely on the left
- the density function of `1` should be entirely on the right



- **(30 points)** Summarize all model performances in a table that identifies:

Models	Method	Package	Hyperparameter	Selection	CV performance	
					Accuracy	Kappa
Logistic regression	glm	stats	NA	NA	0.634	0.260
MARS	Earth	earth	Degree	2	0.629	0.249
Decision trees	rpart	rpart	Cp factor	0.0008	0.625	0.242
Random forests	rf	randomForest	Mtry	1	0.628	0.247
Xgboost	Xgbtree	Caret	Max Depth factor	10	0.640	0.273

We got better score form Xgboost when compared to other modeling techniques. But still it is not a great score. We just 64% of accuracy which gives fair prediction.

(b) (25 points) Build the best possible classification model(s) to predict the target value. Submit your model predictions to the Kaggle.com competition website and outperform your peers in high quality predictions on the test data.

Our best model is boosted trees – XgBoost. Submitted our model predictions to the Kaggle. Please find out team as **(C) AS - 06**