Project 2, Kaggle Porto Seguro Safe Driver Prediction Challenge

Group 1

5/30/2021

## Introduction to dataset

We selected the Porto Seguro’s Safe Driver Prediction competition from the Kaggle website (<https://www.kaggle.com/c/porto-seguro-safe-driver-prediction/overview/evaluation>). The goal of this analysis was to predict whether or not a driver will file an insurance claim next year based on the given predictors. The “target” columns within both data sets, indicate whether or not a claim was filed by the policy holder and takes binary values of 0 for no and 1 for yes. This dataset does contain unbalanced classes. After first attempting random forests and bagging methods in the first interim report, this second interim report focuses on boosting methods.

## Feature extraction and description

Within both the training and testing datasets, there are several variables included however, they are not named. Instead, the features that belong to the same groups are identified as “ind”, “reg”, “car”, and “calc” and have prefixes of bin for binary features or cat for categorical features. There are 3 “reg” variables, 15 “car” variables with 11 being categorical, 18 “ind” variables, 3 of which are binary and 11 being categorical, and 20 “calc” variables with 6 being binary variables. All predictor variables were included when building the XGBoost classifier.

## Random Forest

We tested a random forest model for classification. The response variable is "target", a factor variable with two classes (0 - meaning an insurance claim was not filed, 1- meaning an insurance claim was file). We specified the model to include all predictor variables. In our random forest model, the number of trees used was 500, and the number of variables tried at each split was 7. Our Out of Bag (OOB) estimate of error rate was 3.4%. The error was used to measure the prediction error of the random forest approach. The confusion matrix output showed that most of the observations were grouped at 0,0. A small portion of the observations were grouped at 0,1. There was only 1 observation grouped in 1,0 and no observations were in group 1,1 (Please keep in mind that values in each grouping may increase as sample size increases. The case of our output was because we used a small random subset of the data for running the model). The results of the random forest model indicated that there was the potential of overfitting the observations into one group/tree. The groups with very small or no observations indicated the potential for missing structure. The model was also used to test the performance of the validation dataset, where we found that the results were slightly similar. The grouping of observations were similar to the train model.

## Bagging

The bagging model used for the random subset of our training dataset was composed of classification and regression training (CART) approaches. We computed the model with all the predictor variables (a total of 58 predictors). The resampling method used for this model was bootstrapped, and there were a total of 25 repetitions that were computed. The sample size of each repetition stayed the same throughout. The resampling results from the bagged model output gave an accuracy of 96% with a very low Kappa value that is below zero. This indicated that the high accuracy was likely due to chance rather than being based on actual data values. One can think of it as high accuracy and low precision. The results from the bagging model were used to plot the variable importance plot below.

Chart

Description automatically generated

The plot indicated a rank of importance between each predictor variables in the data set according to the bagging model (ps\_reg\_03 being the highest of importance, and ps\_ind\_10\_bin being the lowest of importance). If this classification model is deemed to be a good fit, then the variables with higher level of importance would be used for building a prediction model to answer for the response variable. The model was also used to test the performance of the validation dataset. The performance of the validation dataset gave the same accuracy, but a higher Kappa value compared to the training dataset.

## Hierarchical Clustering

An attempt at hierarchical clustering was performed using the training data. The hierarchical clustering technique used average linkage to compare the distances between clusters. The resulting dendrogram appeared to show an imbalanced data set where a smaller portion of observations belonged to one hierarchy while a larger portion of observations belong to the larger hierarchy. This coincides with the true response values found in the dataset. However, the hierarchical clustering technique did not result in any useful features or data which could be used in the XGBoost model.

## XGBoost

We used XGBoost with cross-validation to find a model that maximized the Precision Recall Area Under the Curve (PR AUC) metric, which is especially effective for imbalanced data sets. This is because the PR AUC metric looks at maximizing both precision and recall, which helps us balance the two goals of minimizing wrong guesses (precision) and making sure to catch as many of the rare positive cases as we can (recall.)

Our dataset is unbalanced, with a ratio of negative to positive values of ‘target’ of about 26:1. To improve our initial model, we weighted the positive observations of ‘target’ by this ratio, as recommended in the xgboost documentation. After weighting the observations with positive values, our PR AUC values were:

xgb.data.cv$evaluation\_log[sel\_rounds]

## iter train\_aucpr\_mean train\_aucpr\_std test\_aucpr\_mean test\_aucpr\_std  
## 1: 43 0.1012588 0.002051856 0.0631388 0.0017634

To be thorough, we looked at a few different weightings, and assessed the precision and recall for each. The original weighting ratio was about 26. We also evaluated weightings of 24 and 28. The resulting PR AUC metric decreased in each case. The recommended weighting provided the best PR AUC.

We looked at several values of eta (learning rate) and found that an eta of 0.05 provided a very slight improvement in test PR AUC over an eta of 0.1:

xgb.data.cv4$evaluation\_log[sel\_rounds4]

## iter train\_aucpr\_mean train\_aucpr\_std test\_aucpr\_mean test\_aucpr\_std  
## 1: 85 0.1034818 0.002259056 0.0632758 0.002101165

We then looked at whether reducing max.depth from 6 to 3 could avoid overfitting in the training data and ultimately improve the test PR AUC. This gave substantial improvement in test PR AUC but required over 300 rounds and a lot of modeling time.

## iter train\_aucpr\_mean train\_aucpr\_std test\_aucpr\_mean test\_aucpr\_std  
## 1: 239 0.0721008 0.0004772678 0.064601 0.001954284

The selected XGBoost model was then used to predict the values of the training data set, with the following results:

## Precision   
## 0.6298011

## Recall   
## 0.9739356

## [1] "Importance Matrix Top 10"

Since the Kaggle competition uses the gini coefficient to score entries, our team also used the gini coefficient to score our training model. The model predictions resulted in a gini coefficient which would place our group near the top of the leaderboard if the model predicts the test data with the same efficiency. The gini coefficient scores entries between 0.0 and 0.5. A score of 0.0 suggests our model performs no better than random guessing. A score of 0.5 suggests our model perfectly predicts the test data. With a coefficient of 0.2751364, our model fits the data reasonably well.

## Obstacles

The largest issue facing our group was the imbalance of the data set. Because the positive values in our training data occurred so infrequently, the trained models did not predict the response variable exactly as we thought we would be able to during the initial stages of the model building. Since the models predicted the probability that the response would be positive (0.00 – 1.00) we could not completely rely on the confusion matrix to identify the correctly predicted positive and negative values. Instead, we utilized the gini coefficient to measure the distribution of the predicted response variables between 0.00 and 1.00.

Additionally, the variables which made up the data set did not give us any indication of what they represented. Due to this fact, we could not engineer additional features beyond being able to convert a few variables to factors.

## Conclusion

The model which performed the best on the held out data set was the XGBoost model. Ultimately, this model was the most successful due to the properties of this type of boosted tree algorithm. XGBoost attempts to predict the errors from its previous iteration, which allows the model to improve through additional iterations. Because the trees were kept shallow, we were able to limit the amount of over fitting in the data.

This Kaggle competition utilized data which was somewhat difficult to work with as the data labels did not give us any useful indication of which variables might be related. Because of this, feature extraction and feature engineering was vastly limited in this project. If we were to complete a similar project again in the future, we would ideally like to work with a data set which could be engineered to better extract more useful features. Feature engineering in XGB models is powerful as the model is a greedy model which chooses the next node split based on the feature which best explains most of the error from the previous tree. If engineered features are not useful, the model will not select them, however, if the engineered features are useful, the XGB model will select them to explain most of the error in the previous iteration.

In the future, we would like to utilize an XGBoost model to compete in another Kaggle classification competition. However, we would prefer to compete in a competition where we could better understand the variables in the data set.

## Appendix

library(ggplot2)  
library(tree)  
library(Boruta)  
library(tidyverse)

## Registered S3 method overwritten by 'cli':  
## method from  
## print.tree tree

## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v tibble 3.1.1 v dplyr 1.0.5  
## v tidyr 1.1.3 v stringr 1.4.0  
## v readr 1.4.0 v forcats 0.5.1  
## v purrr 0.3.4

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(randomForest)

## randomForest 4.6-14

## Type rfNews() to see new features/changes/bug fixes.

##   
## Attaching package: 'randomForest'

## The following object is masked from 'package:dplyr':  
##   
## combine

## The following object is masked from 'package:ggplot2':  
##   
## margin

library(caret)

## Loading required package: lattice

##   
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':  
##   
## lift

library(lime)

##   
## Attaching package: 'lime'

## The following object is masked from 'package:dplyr':  
##   
## explain

#Importing datasets#

data <- read.csv("sample\_submission.csv")  
train <- read.csv("train.csv")

#Data#

head(train)

## id target ps\_ind\_01 ps\_ind\_02\_cat ps\_ind\_03 ps\_ind\_04\_cat ps\_ind\_05\_cat  
## 1 7 0 2 2 5 1 0  
## 2 9 0 1 1 7 0 0  
## 3 13 0 5 4 9 1 0  
## 4 16 0 0 1 2 0 0  
## 5 17 0 0 2 0 1 0  
## 6 19 0 5 1 4 0 0  
## ps\_ind\_06\_bin ps\_ind\_07\_bin ps\_ind\_08\_bin ps\_ind\_09\_bin ps\_ind\_10\_bin  
## 1 0 1 0 0 0  
## 2 0 0 1 0 0  
## 3 0 0 1 0 0  
## 4 1 0 0 0 0  
## 5 1 0 0 0 0  
## 6 0 0 0 1 0  
## ps\_ind\_11\_bin ps\_ind\_12\_bin ps\_ind\_13\_bin ps\_ind\_14 ps\_ind\_15 ps\_ind\_16\_bin  
## 1 0 0 0 0 11 0  
## 2 0 0 0 0 3 0  
## 3 0 0 0 0 12 1  
## 4 0 0 0 0 8 1  
## 5 0 0 0 0 9 1  
## 6 0 0 0 0 6 1  
## ps\_ind\_17\_bin ps\_ind\_18\_bin ps\_reg\_01 ps\_reg\_02 ps\_reg\_03 ps\_car\_01\_cat  
## 1 1 0 0.7 0.2 0.7180703 10  
## 2 0 1 0.8 0.4 0.7660777 11  
## 3 0 0 0.0 0.0 -1.0000000 7  
## 4 0 0 0.9 0.2 0.5809475 7  
## 5 0 0 0.7 0.6 0.8407586 11  
## 6 0 0 0.9 1.8 2.3326487 10  
## ps\_car\_02\_cat ps\_car\_03\_cat ps\_car\_04\_cat ps\_car\_05\_cat ps\_car\_06\_cat  
## 1 1 -1 0 1 4  
## 2 1 -1 0 -1 11  
## 3 1 -1 0 -1 14  
## 4 1 0 0 1 11  
## 5 1 -1 0 -1 14  
## 6 0 -1 0 0 14  
## ps\_car\_07\_cat ps\_car\_08\_cat ps\_car\_09\_cat ps\_car\_10\_cat ps\_car\_11\_cat  
## 1 1 0 0 1 12  
## 2 1 1 2 1 19  
## 3 1 1 2 1 60  
## 4 1 1 3 1 104  
## 5 1 1 2 1 82  
## 6 1 1 0 1 104  
## ps\_car\_11 ps\_car\_12 ps\_car\_13 ps\_car\_14 ps\_car\_15 ps\_calc\_01 ps\_calc\_02  
## 1 2 0.4000000 0.8836789 0.3708099 3.605551 0.6 0.5  
## 2 3 0.3162278 0.6188165 0.3887158 2.449490 0.3 0.1  
## 3 1 0.3162278 0.6415857 0.3472751 3.316625 0.5 0.7  
## 4 1 0.3741657 0.5429488 0.2949576 2.000000 0.6 0.9  
## 5 3 0.3160696 0.5658315 0.3651027 2.000000 0.4 0.6  
## 6 2 0.4459821 0.8790491 0.4062019 3.000000 0.7 0.8  
## ps\_calc\_03 ps\_calc\_04 ps\_calc\_05 ps\_calc\_06 ps\_calc\_07 ps\_calc\_08 ps\_calc\_09  
## 1 0.2 3 1 10 1 10 1  
## 2 0.3 2 1 9 5 8 1  
## 3 0.1 2 2 9 1 8 2  
## 4 0.1 2 4 7 1 8 4  
## 5 0.0 2 2 6 3 10 2  
## 6 0.4 3 1 8 2 11 3  
## ps\_calc\_10 ps\_calc\_11 ps\_calc\_12 ps\_calc\_13 ps\_calc\_14 ps\_calc\_15\_bin  
## 1 5 9 1 5 8 0  
## 2 7 3 1 1 9 0  
## 3 7 4 2 7 7 0  
## 4 2 2 2 4 9 0  
## 5 12 3 1 1 3 0  
## 6 8 4 2 0 9 0  
## ps\_calc\_16\_bin ps\_calc\_17\_bin ps\_calc\_18\_bin ps\_calc\_19\_bin ps\_calc\_20\_bin  
## 1 1 1 0 0 1  
## 2 1 1 0 1 0  
## 3 1 1 0 1 0  
## 4 0 0 0 0 0  
## 5 0 0 1 1 0  
## 6 1 0 1 1 1

str(train)

## 'data.frame': 595212 obs. of 59 variables:  
## $ id : int 7 9 13 16 17 19 20 22 26 28 ...  
## $ target : int 0 0 0 0 0 0 0 0 0 1 ...  
## $ ps\_ind\_01 : int 2 1 5 0 0 5 2 5 5 1 ...  
## $ ps\_ind\_02\_cat : int 2 1 4 1 2 1 1 1 1 1 ...  
## $ ps\_ind\_03 : int 5 7 9 2 0 4 3 4 3 2 ...  
## $ ps\_ind\_04\_cat : int 1 0 1 0 1 0 1 0 1 0 ...  
## $ ps\_ind\_05\_cat : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ ps\_ind\_06\_bin : int 0 0 0 1 1 0 0 1 0 0 ...  
## $ ps\_ind\_07\_bin : int 1 0 0 0 0 0 1 0 0 1 ...  
## $ ps\_ind\_08\_bin : int 0 1 1 0 0 0 0 0 1 0 ...  
## $ ps\_ind\_09\_bin : int 0 0 0 0 0 1 0 0 0 0 ...  
## $ ps\_ind\_10\_bin : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ ps\_ind\_11\_bin : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ ps\_ind\_12\_bin : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ ps\_ind\_13\_bin : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ ps\_ind\_14 : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ ps\_ind\_15 : int 11 3 12 8 9 6 8 13 6 4 ...  
## $ ps\_ind\_16\_bin : int 0 0 1 1 1 1 1 1 1 0 ...  
## $ ps\_ind\_17\_bin : int 1 0 0 0 0 0 0 0 0 0 ...  
## $ ps\_ind\_18\_bin : int 0 1 0 0 0 0 0 0 0 1 ...  
## $ ps\_reg\_01 : num 0.7 0.8 0 0.9 0.7 0.9 0.6 0.7 0.9 0.9 ...  
## $ ps\_reg\_02 : num 0.2 0.4 0 0.2 0.6 1.8 0.1 0.4 0.7 1.4 ...  
## $ ps\_reg\_03 : num 0.718 0.766 -1 0.581 0.841 ...  
## $ ps\_car\_01\_cat : int 10 11 7 7 11 10 6 11 10 11 ...  
## $ ps\_car\_02\_cat : int 1 1 1 1 1 0 1 1 1 0 ...  
## $ ps\_car\_03\_cat : int -1 -1 -1 0 -1 -1 -1 0 -1 0 ...  
## $ ps\_car\_04\_cat : int 0 0 0 0 0 0 0 0 0 1 ...  
## $ ps\_car\_05\_cat : int 1 -1 -1 1 -1 0 1 0 1 0 ...  
## $ ps\_car\_06\_cat : int 4 11 14 11 14 14 11 11 14 14 ...  
## $ ps\_car\_07\_cat : int 1 1 1 1 1 1 1 1 1 1 ...  
## $ ps\_car\_08\_cat : int 0 1 1 1 1 1 1 1 1 1 ...  
## $ ps\_car\_09\_cat : int 0 2 2 3 2 0 0 2 0 2 ...  
## $ ps\_car\_10\_cat : int 1 1 1 1 1 1 1 1 1 1 ...  
## $ ps\_car\_11\_cat : int 12 19 60 104 82 104 99 30 68 104 ...  
## $ ps\_car\_11 : int 2 3 1 1 3 2 2 3 3 2 ...  
## $ ps\_car\_12 : num 0.4 0.316 0.316 0.374 0.316 ...  
## $ ps\_car\_13 : num 0.884 0.619 0.642 0.543 0.566 ...  
## $ ps\_car\_14 : num 0.371 0.389 0.347 0.295 0.365 ...  
## $ ps\_car\_15 : num 3.61 2.45 3.32 2 2 ...  
## $ ps\_calc\_01 : num 0.6 0.3 0.5 0.6 0.4 0.7 0.2 0.1 0.9 0.7 ...  
## $ ps\_calc\_02 : num 0.5 0.1 0.7 0.9 0.6 0.8 0.6 0.5 0.8 0.8 ...  
## $ ps\_calc\_03 : num 0.2 0.3 0.1 0.1 0 0.4 0.5 0.1 0.6 0.8 ...  
## $ ps\_calc\_04 : int 3 2 2 2 2 3 2 1 3 2 ...  
## $ ps\_calc\_05 : int 1 1 2 4 2 1 2 2 1 2 ...  
## $ ps\_calc\_06 : int 10 9 9 7 6 8 8 7 7 8 ...  
## $ ps\_calc\_07 : int 1 5 1 1 3 2 1 1 3 2 ...  
## $ ps\_calc\_08 : int 10 8 8 8 10 11 8 6 9 9 ...  
## $ ps\_calc\_09 : int 1 1 2 4 2 3 3 1 4 1 ...  
## $ ps\_calc\_10 : int 5 7 7 2 12 8 10 13 11 11 ...  
## $ ps\_calc\_11 : int 9 3 4 2 3 4 3 7 4 3 ...  
## $ ps\_calc\_12 : int 1 1 2 2 1 2 0 1 2 5 ...  
## $ ps\_calc\_13 : int 5 1 7 4 1 0 0 3 1 0 ...  
## $ ps\_calc\_14 : int 8 9 7 9 3 9 10 6 5 6 ...  
## $ ps\_calc\_15\_bin: int 0 0 0 0 0 0 0 1 0 0 ...  
## $ ps\_calc\_16\_bin: int 1 1 1 0 0 1 1 0 1 1 ...  
## $ ps\_calc\_17\_bin: int 1 1 1 0 0 0 0 1 0 0 ...  
## $ ps\_calc\_18\_bin: int 0 0 0 0 1 1 0 0 0 0 ...  
## $ ps\_calc\_19\_bin: int 0 1 1 0 1 1 1 1 0 1 ...  
## $ ps\_calc\_20\_bin: int 1 0 0 0 0 1 0 0 1 0 ...

ncol(train)

## [1] 59

nrow(train)

## [1] 595212

#Data Partition into training and validation sets#

set.seed(123)  
ind <- sample(2, nrow(train), replace = TRUE, prob = c(0.5, 0.5))  
  
traindata <- train[ind == 1,]  
valdata <- train[ind == 2,]

#View both training and validation data sets#

head(traindata)

## id target ps\_ind\_01 ps\_ind\_02\_cat ps\_ind\_03 ps\_ind\_04\_cat ps\_ind\_05\_cat  
## 2 9 0 1 1 7 0 0  
## 4 16 0 0 1 2 0 0  
## 5 17 0 0 2 0 1 0  
## 7 20 0 2 1 3 1 0  
## 8 22 0 5 1 4 0 0  
## 9 26 0 5 1 3 1 0  
## ps\_ind\_06\_bin ps\_ind\_07\_bin ps\_ind\_08\_bin ps\_ind\_09\_bin ps\_ind\_10\_bin  
## 2 0 0 1 0 0  
## 4 1 0 0 0 0  
## 5 1 0 0 0 0  
## 7 0 1 0 0 0  
## 8 1 0 0 0 0  
## 9 0 0 1 0 0  
## ps\_ind\_11\_bin ps\_ind\_12\_bin ps\_ind\_13\_bin ps\_ind\_14 ps\_ind\_15 ps\_ind\_16\_bin  
## 2 0 0 0 0 3 0  
## 4 0 0 0 0 8 1  
## 5 0 0 0 0 9 1  
## 7 0 0 0 0 8 1  
## 8 0 0 0 0 13 1  
## 9 0 0 0 0 6 1  
## ps\_ind\_17\_bin ps\_ind\_18\_bin ps\_reg\_01 ps\_reg\_02 ps\_reg\_03 ps\_car\_01\_cat  
## 2 0 1 0.8 0.4 0.7660777 11  
## 4 0 0 0.9 0.2 0.5809475 7  
## 5 0 0 0.7 0.6 0.8407586 11  
## 7 0 0 0.6 0.1 0.6174545 6  
## 8 0 0 0.7 0.4 0.6072479 11  
## 9 0 0 0.9 0.7 0.9013878 10  
## ps\_car\_02\_cat ps\_car\_03\_cat ps\_car\_04\_cat ps\_car\_05\_cat ps\_car\_06\_cat  
## 2 1 -1 0 -1 11  
## 4 1 0 0 1 11  
## 5 1 -1 0 -1 14  
## 7 1 -1 0 1 11  
## 8 1 0 0 0 11  
## 9 1 -1 0 1 14  
## ps\_car\_07\_cat ps\_car\_08\_cat ps\_car\_09\_cat ps\_car\_10\_cat ps\_car\_11\_cat  
## 2 1 1 2 1 19  
## 4 1 1 3 1 104  
## 5 1 1 2 1 82  
## 7 1 1 0 1 99  
## 8 1 1 2 1 30  
## 9 1 1 0 1 68  
## ps\_car\_11 ps\_car\_12 ps\_car\_13 ps\_car\_14 ps\_car\_15 ps\_calc\_01 ps\_calc\_02  
## 2 3 0.3162278 0.6188165 0.3887158 2.449490 0.3 0.1  
## 4 1 0.3741657 0.5429488 0.2949576 2.000000 0.6 0.9  
## 5 3 0.3160696 0.5658315 0.3651027 2.000000 0.4 0.6  
## 7 2 0.3162278 0.6396829 0.3687818 3.162278 0.2 0.6  
## 8 3 0.4469899 0.9005737 0.3748333 3.316625 0.1 0.5  
## 9 3 0.4000000 0.7806413 0.4074310 2.828427 0.9 0.8  
## ps\_calc\_03 ps\_calc\_04 ps\_calc\_05 ps\_calc\_06 ps\_calc\_07 ps\_calc\_08 ps\_calc\_09  
## 2 0.3 2 1 9 5 8 1  
## 4 0.1 2 4 7 1 8 4  
## 5 0.0 2 2 6 3 10 2  
## 7 0.5 2 2 8 1 8 3  
## 8 0.1 1 2 7 1 6 1  
## 9 0.6 3 1 7 3 9 4  
## ps\_calc\_10 ps\_calc\_11 ps\_calc\_12 ps\_calc\_13 ps\_calc\_14 ps\_calc\_15\_bin  
## 2 7 3 1 1 9 0  
## 4 2 2 2 4 9 0  
## 5 12 3 1 1 3 0  
## 7 10 3 0 0 10 0  
## 8 13 7 1 3 6 1  
## 9 11 4 2 1 5 0  
## ps\_calc\_16\_bin ps\_calc\_17\_bin ps\_calc\_18\_bin ps\_calc\_19\_bin ps\_calc\_20\_bin  
## 2 1 1 0 1 0  
## 4 0 0 0 0 0  
## 5 0 0 1 1 0  
## 7 1 0 0 1 0  
## 8 0 1 0 1 0  
## 9 1 0 0 0 1

head(valdata)

## id target ps\_ind\_01 ps\_ind\_02\_cat ps\_ind\_03 ps\_ind\_04\_cat ps\_ind\_05\_cat  
## 1 7 0 2 2 5 1 0  
## 3 13 0 5 4 9 1 0  
## 6 19 0 5 1 4 0 0  
## 10 28 1 1 1 2 0 0  
## 12 35 0 2 1 3 1 0  
## 15 46 0 5 1 11 0 0  
## ps\_ind\_06\_bin ps\_ind\_07\_bin ps\_ind\_08\_bin ps\_ind\_09\_bin ps\_ind\_10\_bin  
## 1 0 1 0 0 0  
## 3 0 0 1 0 0  
## 6 0 0 0 1 0  
## 10 0 1 0 0 0  
## 12 0 1 0 0 0  
## 15 0 0 0 1 0  
## ps\_ind\_11\_bin ps\_ind\_12\_bin ps\_ind\_13\_bin ps\_ind\_14 ps\_ind\_15 ps\_ind\_16\_bin  
## 1 0 0 0 0 11 0  
## 3 0 0 0 0 12 1  
## 6 0 0 0 0 6 1  
## 10 0 0 0 0 4 0  
## 12 0 0 0 0 9 1  
## 15 0 0 0 0 10 0  
## ps\_ind\_17\_bin ps\_ind\_18\_bin ps\_reg\_01 ps\_reg\_02 ps\_reg\_03 ps\_car\_01\_cat  
## 1 1 0 0.7 0.2 0.7180703 10  
## 3 0 0 0.0 0.0 -1.0000000 7  
## 6 0 0 0.9 1.8 2.3326487 10  
## 10 0 1 0.9 1.4 2.3166517 11  
## 12 0 0 0.9 0.1 0.3783186 11  
## 15 0 1 0.8 0.6 1.0529720 11  
## ps\_car\_02\_cat ps\_car\_03\_cat ps\_car\_04\_cat ps\_car\_05\_cat ps\_car\_06\_cat  
## 1 1 -1 0 1 4  
## 3 1 -1 0 -1 14  
## 6 0 -1 0 0 14  
## 10 0 0 1 0 14  
## 12 0 -1 0 -1 11  
## 15 1 -1 0 -1 11  
## ps\_car\_07\_cat ps\_car\_08\_cat ps\_car\_09\_cat ps\_car\_10\_cat ps\_car\_11\_cat  
## 1 1 0 0 1 12  
## 3 1 1 2 1 60  
## 6 1 1 0 1 104  
## 10 1 1 2 1 104  
## 12 1 0 0 1 36  
## 15 1 1 2 1 103  
## ps\_car\_11 ps\_car\_12 ps\_car\_13 ps\_car\_14 ps\_car\_15 ps\_calc\_01 ps\_calc\_02  
## 1 2 0.4000000 0.8836789 0.3708099 3.605551 0.6 0.5  
## 3 1 0.3162278 0.6415857 0.3472751 3.316625 0.5 0.7  
## 6 2 0.4459821 0.8790491 0.4062019 3.000000 0.7 0.8  
## 10 2 0.4472136 1.4581844 0.3902563 3.605551 0.7 0.8  
## 12 3 0.3741657 0.9544913 0.3793415 3.741657 0.3 0.7  
## 15 1 0.3162278 0.5665249 0.3583295 2.645751 0.4 0.4  
## ps\_calc\_03 ps\_calc\_04 ps\_calc\_05 ps\_calc\_06 ps\_calc\_07 ps\_calc\_08 ps\_calc\_09  
## 1 0.2 3 1 10 1 10 1  
## 3 0.1 2 2 9 1 8 2  
## 6 0.4 3 1 8 2 11 3  
## 10 0.8 2 2 8 2 9 1  
## 12 0.5 2 2 8 2 10 1  
## 15 0.1 3 1 8 2 6 3  
## ps\_calc\_10 ps\_calc\_11 ps\_calc\_12 ps\_calc\_13 ps\_calc\_14 ps\_calc\_15\_bin  
## 1 5 9 1 5 8 0  
## 3 7 4 2 7 7 0  
## 6 8 4 2 0 9 0  
## 10 11 3 5 0 6 0  
## 12 8 9 2 1 10 0  
## 15 12 4 1 3 9 0  
## ps\_calc\_16\_bin ps\_calc\_17\_bin ps\_calc\_18\_bin ps\_calc\_19\_bin ps\_calc\_20\_bin  
## 1 1 1 0 0 1  
## 3 1 1 0 1 0  
## 6 1 0 1 1 1  
## 10 1 0 0 1 0  
## 12 1 0 1 0 0  
## 15 0 0 0 1 0

#Take a small sample from training and val set for analysis#

set.seed(123)  
train\_s2 <- sample\_frac(traindata, 0.05)  
train\_s2$target <- as.factor(train\_s2$target)  
  
val\_s2 <-sample\_frac(valdata, 0.05)  
val\_s2$target <- as.factor(val\_s2$target)  
nrow(val\_s2)

## [1] 14897

#Random Forest Model#

set.seed(123)  
train\_rf\_model <- randomForest(target~., data = train\_s2)  
train\_rf\_model

##   
## Call:  
## randomForest(formula = target ~ ., data = train\_s2)   
## Type of random forest: classification  
## Number of trees: 500  
## No. of variables tried at each split: 7  
##   
## OOB estimate of error rate: 3.86%  
## Confusion matrix:  
## 0 1 class.error  
## 0 14290 0 0  
## 1 574 0 1

#Bagging#

bag\_train <- train(target~.,  
 data=train\_s2,  
 method="treebag",  
 importance=TRUE)  
bag\_train

## Bagged CART   
##   
## 14864 samples  
## 58 predictor  
## 2 classes: '0', '1'   
##   
## No pre-processing  
## Resampling: Bootstrapped (25 reps)   
## Summary of sample sizes: 14864, 14864, 14864, 14864, 14864, 14864, ...   
## Resampling results:  
##   
## Accuracy Kappa   
## 0.9612285 0.002113822

#Variable important plot#

plot(varImp(bag\_train))

Chart

Description automatically generated

#Test on validation data#

set.seed(123)  
val\_rf\_model <- randomForest(target~., data = val\_s2)  
val\_rf\_model

##   
## Call:  
## randomForest(formula = target ~ ., data = val\_s2)   
## Type of random forest: classification  
## Number of trees: 500  
## No. of variables tried at each split: 7  
##   
## OOB estimate of error rate: 3.77%  
## Confusion matrix:  
## 0 1 class.error  
## 0 14335 1 6.975446e-05  
## 1 561 0 1.000000e+00

bag\_val <- train(target~.,  
 data=val\_s2,  
 method="treebag",  
 importance=TRUE)  
bag\_val

## Bagged CART   
##   
## 14897 samples  
## 58 predictor  
## 2 classes: '0', '1'   
##   
## No pre-processing  
## Resampling: Bootstrapped (25 reps)   
## Summary of sample sizes: 14897, 14897, 14897, 14897, 14897, 14897, ...   
## Resampling results:  
##   
## Accuracy Kappa   
## 0.9610896 0.001837796

# Project 2 - Kaggle Porto Seguro Safe Driver Prediction Challenge  
# Project 2 - Kaggle Porto Seguro Safe Driver Prediction Challenge  
  
library(tinytex)  
library(MASS)  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following object is masked from 'package:MASS':  
##   
## select

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.3 v purrr 0.3.4  
## v tibble 3.1.1 v stringr 1.4.0  
## v tidyr 1.1.3 v forcats 0.5.1  
## v readr 1.4.0

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()  
## x dplyr::select() masks MASS::select()

library(gbm)

## Loaded gbm 2.1.8

library(caret)

## Loading required package: lattice

##   
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':  
##   
## lift

library(xgboost)

##   
## Attaching package: 'xgboost'

## The following object is masked from 'package:dplyr':  
##   
## slice

library(ggplot2)  
library(corrplot)

## corrplot 0.88 loaded

library(mvtnorm)  
library(DiagrammeR)  
library(ROSE)

## Loaded ROSE 0.0-3

library(cluster)  
library(plotrix)  
library(graphics)  
library(data.table)

##   
## Attaching package: 'data.table'

## The following object is masked from 'package:purrr':  
##   
## transpose

## The following objects are masked from 'package:dplyr':  
##   
## between, first, last

data <- fread("train.csv")  
  
data = data[ ,2:length(names(data))]  
  
  
#HC analysis  
  
  
  
data <- data %>%   
 mutate\_at(vars(ends\_with("bin")), list(factor)) %>%   
 mutate\_at(vars(ends\_with("cat")), list(factor))  
  
# create predictors and response data frames  
data\_resp <- data[,1]  
data\_pred <- data[,-1]  
  
  
# Create index to split training and validation sets  
set.seed(1)  
idx.hc <- sample(c(0,1), nrow(data), replace = TRUE, prob = c(.95, .05))  
  
# Using the index separate the training set into training and validation sets  
resp\_tr <- data\_resp[idx.hc == 0]  
pred\_tr <- data\_pred[idx.hc == 0,]  
  
resp\_val <- data\_resp[idx.hc == 1]  
pred\_val <- data\_pred[idx.hc == 1,]  
  
  
# hc <- hclust(dist(hc.data), method = "average")  
# plot(hc)  
  
# cl <- cutree(hc, k=2)  
# plot(data.frame(hc.data[,1], data.frame(hc.data[,2]), col=cl))  
  
  
  
# XGBoost Classifier  
data <- fread("train.csv")  
data = data[ ,2:length(names(data))]  
#   
#   
# data <- data %>%  
# mutate\_at(vars(ends\_with("bin")), list(factor)) %>%  
# mutate\_at(vars(ends\_with("cat")), list(factor))  
  
# create predictors and response data frames  
data\_resp <- data[,1]  
data\_pred <- data[,-1]  
  
  
# Create index to split training and validation sets  
set.seed(1)  
idx <- sample(c(0,1), nrow(data), replace = TRUE, prob = c(.8, .2))  
  
# Using the index separate the training set into training and validation sets  
resp\_tr <- as.matrix((data\_resp[idx == 0]))  
pred\_tr <- as.matrix(data\_pred[idx == 0,])  
  
resp\_val <- as.matrix(data\_resp[idx == 1])  
pred\_val <- as.matrix(data\_pred[idx == 1,])  
  
  
# Training data is heavily skewed towards zero  
summary(data\_resp)

## target   
## Min. :0.00000   
## 1st Qu.:0.00000   
## Median :0.00000   
## Mean :0.03645   
## 3rd Qu.:0.00000   
## Max. :1.00000

# train.data = as.matrix(data[,2:length(names(data))])  
# train.labels = as.numeric(data$target)  
# val.data = as.matrix(test[ ,2:length(names(test))])  
# val.labels = as.numeric(test$target)  
  
# xgb.data = xgboost(data = train.data, label = train.labels,  
# max.depth = 6, eta = 1, nrounds = 200,  
# objective = "binary:logistic", eval\_metric = "aucpr")  
#   
# xgb.data$evaluation\_log[200]  
  
  
pos <- sum(resp\_tr == 1)  
neg<- sum(resp\_tr == 0)  
  
neg.rate <- neg / pos  
  
  
#Weighting:  
  
xgb.data.cv = xgb.cv(data = pred\_tr, label = resp\_tr,   
 max.depth = 6, eta = .1, nrounds = 200, min\_child\_weight = 1,  
 nfold = 5, early\_stopping\_rounds = 5, scale\_pos\_weight = neg.rate,  
 objective = "binary:logistic", eval\_metric = "aucpr")

## [1] train-aucpr:0.062067+0.000520 test-aucpr:0.055246+0.002690   
## Multiple eval metrics are present. Will use test\_aucpr for early stopping.  
## Will train until test\_aucpr hasn't improved in 5 rounds.  
##   
## [2] train-aucpr:0.064955+0.000406 test-aucpr:0.056497+0.001884   
## [3] train-aucpr:0.066608+0.000370 test-aucpr:0.057213+0.001936   
## [4] train-aucpr:0.068480+0.000508 test-aucpr:0.058165+0.002260   
## [5] train-aucpr:0.069746+0.000707 test-aucpr:0.058544+0.002276   
## [6] train-aucpr:0.071302+0.000358 test-aucpr:0.059262+0.002467   
## [7] train-aucpr:0.072491+0.000457 test-aucpr:0.059805+0.002498   
## [8] train-aucpr:0.073722+0.000731 test-aucpr:0.059998+0.002317   
## [9] train-aucpr:0.074434+0.000576 test-aucpr:0.060208+0.002198   
## [10] train-aucpr:0.075643+0.000700 test-aucpr:0.060520+0.002133   
## [11] train-aucpr:0.076459+0.000640 test-aucpr:0.060748+0.002083   
## [12] train-aucpr:0.077520+0.000848 test-aucpr:0.061169+0.002050   
## [13] train-aucpr:0.078153+0.000847 test-aucpr:0.061272+0.002082   
## [14] train-aucpr:0.078917+0.000790 test-aucpr:0.061452+0.002033   
## [15] train-aucpr:0.079814+0.001054 test-aucpr:0.061653+0.001982   
## [16] train-aucpr:0.080562+0.001126 test-aucpr:0.061807+0.001958   
## [17] train-aucpr:0.081480+0.000921 test-aucpr:0.061942+0.001970   
## [18] train-aucpr:0.082376+0.000855 test-aucpr:0.062014+0.001926   
## [19] train-aucpr:0.083249+0.000909 test-aucpr:0.062118+0.001917   
## [20] train-aucpr:0.084145+0.000841 test-aucpr:0.062197+0.001869   
## [21] train-aucpr:0.085123+0.000863 test-aucpr:0.062245+0.001864   
## [22] train-aucpr:0.085883+0.000970 test-aucpr:0.062332+0.001674   
## [23] train-aucpr:0.086579+0.001202 test-aucpr:0.062348+0.001732   
## [24] train-aucpr:0.087445+0.001292 test-aucpr:0.062434+0.001792   
## [25] train-aucpr:0.088242+0.001294 test-aucpr:0.062532+0.001779   
## [26] train-aucpr:0.089165+0.001170 test-aucpr:0.062546+0.001775   
## [27] train-aucpr:0.090164+0.001157 test-aucpr:0.062431+0.001837   
## [28] train-aucpr:0.091099+0.001511 test-aucpr:0.062681+0.001752   
## [29] train-aucpr:0.091821+0.001539 test-aucpr:0.062836+0.001619   
## [30] train-aucpr:0.092367+0.001623 test-aucpr:0.062871+0.001573   
## [31] train-aucpr:0.093032+0.001664 test-aucpr:0.062882+0.001621   
## [32] train-aucpr:0.093794+0.001700 test-aucpr:0.062865+0.001628   
## [33] train-aucpr:0.094491+0.001847 test-aucpr:0.062853+0.001564   
## [34] train-aucpr:0.095214+0.001725 test-aucpr:0.062866+0.001635   
## [35] train-aucpr:0.095818+0.001818 test-aucpr:0.062932+0.001631   
## [36] train-aucpr:0.096303+0.001781 test-aucpr:0.062875+0.001629   
## [37] train-aucpr:0.096974+0.001834 test-aucpr:0.062938+0.001767   
## [38] train-aucpr:0.097560+0.001822 test-aucpr:0.062979+0.001707   
## [39] train-aucpr:0.098484+0.002057 test-aucpr:0.063021+0.001660   
## [40] train-aucpr:0.099254+0.002021 test-aucpr:0.063124+0.001719   
## [41] train-aucpr:0.099898+0.001962 test-aucpr:0.063119+0.001722   
## [42] train-aucpr:0.100606+0.001942 test-aucpr:0.063104+0.001800   
## [43] train-aucpr:0.101259+0.002052 test-aucpr:0.063139+0.001763   
## [44] train-aucpr:0.101888+0.002093 test-aucpr:0.063042+0.001735   
## [45] train-aucpr:0.102555+0.002132 test-aucpr:0.063032+0.001743   
## [46] train-aucpr:0.103380+0.002143 test-aucpr:0.062943+0.001789   
## [47] train-aucpr:0.103919+0.002157 test-aucpr:0.063007+0.001710   
## [48] train-aucpr:0.104632+0.002149 test-aucpr:0.062948+0.001727   
## Stopping. Best iteration:  
## [43] train-aucpr:0.101259+0.002052 test-aucpr:0.063139+0.001763

summary(xgb.data.cv)

## Length Class Mode   
## call 12 -none- call   
## params 7 -none- list   
## callbacks 3 -none- list   
## evaluation\_log 5 data.table list   
## niter 1 -none- numeric  
## nfeatures 1 -none- numeric  
## folds 5 -none- list   
## best\_iteration 1 -none- numeric  
## best\_ntreelimit 1 -none- numeric

sel\_rounds <- xgb.data.cv$best\_iteration  
AUC <- xgb.data.cv$evaluation\_log[sel\_rounds]  
AUC

## iter train\_aucpr\_mean train\_aucpr\_std test\_aucpr\_mean test\_aucpr\_std  
## 1: 43 0.1012588 0.002051856 0.0631388 0.0017634

# Assess different weights of the positive class of target.  
# Weight minus two  
  
xgb.data.cv2 = xgb.cv(data = pred\_tr, label = resp\_tr,  
 max.depth = 6, eta = .1, nrounds = 200, min\_child\_weight = 1,  
 nfold = 5, early\_stopping\_rounds = 5, scale\_pos\_weight = neg.rate-2,  
 objective = "binary:logistic", eval\_metric = "aucpr")

## [1] train-aucpr:0.062223+0.000614 test-aucpr:0.055562+0.001095   
## Multiple eval metrics are present. Will use test\_aucpr for early stopping.  
## Will train until test\_aucpr hasn't improved in 5 rounds.  
##   
## [2] train-aucpr:0.065296+0.000844 test-aucpr:0.056971+0.001271   
## [3] train-aucpr:0.067992+0.001119 test-aucpr:0.058348+0.001183   
## [4] train-aucpr:0.069173+0.000887 test-aucpr:0.058629+0.000983   
## [5] train-aucpr:0.070679+0.000942 test-aucpr:0.058886+0.000752   
## [6] train-aucpr:0.071911+0.001013 test-aucpr:0.059004+0.000703   
## [7] train-aucpr:0.073159+0.001038 test-aucpr:0.059416+0.001070   
## [8] train-aucpr:0.074191+0.000815 test-aucpr:0.059719+0.000963   
## [9] train-aucpr:0.075159+0.000775 test-aucpr:0.059908+0.000852   
## [10] train-aucpr:0.076222+0.000616 test-aucpr:0.060227+0.000972   
## [11] train-aucpr:0.077269+0.000843 test-aucpr:0.060547+0.000975   
## [12] train-aucpr:0.077970+0.000958 test-aucpr:0.060562+0.000953   
## [13] train-aucpr:0.078762+0.001053 test-aucpr:0.060697+0.001051   
## [14] train-aucpr:0.079636+0.001074 test-aucpr:0.060955+0.001074   
## [15] train-aucpr:0.080192+0.000898 test-aucpr:0.061059+0.001109   
## [16] train-aucpr:0.081065+0.000951 test-aucpr:0.061063+0.000909   
## [17] train-aucpr:0.081829+0.000904 test-aucpr:0.061176+0.001004   
## [18] train-aucpr:0.082762+0.000704 test-aucpr:0.061299+0.001108   
## [19] train-aucpr:0.083791+0.000860 test-aucpr:0.061336+0.001154   
## [20] train-aucpr:0.084582+0.000788 test-aucpr:0.061396+0.001247   
## [21] train-aucpr:0.085267+0.000625 test-aucpr:0.061440+0.001121   
## [22] train-aucpr:0.086031+0.000668 test-aucpr:0.061631+0.001097   
## [23] train-aucpr:0.086956+0.000526 test-aucpr:0.061667+0.001129   
## [24] train-aucpr:0.087878+0.000591 test-aucpr:0.061687+0.001138   
## [25] train-aucpr:0.088500+0.000579 test-aucpr:0.061765+0.001170   
## [26] train-aucpr:0.089174+0.000650 test-aucpr:0.061676+0.001183   
## [27] train-aucpr:0.089794+0.000563 test-aucpr:0.061752+0.001130   
## [28] train-aucpr:0.090604+0.000583 test-aucpr:0.061876+0.001191   
## [29] train-aucpr:0.091485+0.000651 test-aucpr:0.061852+0.001147   
## [30] train-aucpr:0.092442+0.000694 test-aucpr:0.061848+0.001224   
## [31] train-aucpr:0.093163+0.000587 test-aucpr:0.061822+0.001190   
## [32] train-aucpr:0.093850+0.000630 test-aucpr:0.061855+0.001169   
## [33] train-aucpr:0.094607+0.000512 test-aucpr:0.061964+0.001150   
## [34] train-aucpr:0.095169+0.000453 test-aucpr:0.061955+0.001182   
## [35] train-aucpr:0.095719+0.000360 test-aucpr:0.061975+0.001192   
## [36] train-aucpr:0.096455+0.000350 test-aucpr:0.062088+0.001194   
## [37] train-aucpr:0.097014+0.000372 test-aucpr:0.062058+0.001326   
## [38] train-aucpr:0.097758+0.000392 test-aucpr:0.062141+0.001268   
## [39] train-aucpr:0.098490+0.000463 test-aucpr:0.062143+0.001222   
## [40] train-aucpr:0.099236+0.000676 test-aucpr:0.062080+0.001203   
## [41] train-aucpr:0.099805+0.000629 test-aucpr:0.062116+0.001178   
## [42] train-aucpr:0.100515+0.000724 test-aucpr:0.062101+0.001152   
## [43] train-aucpr:0.101202+0.000745 test-aucpr:0.062191+0.001115   
## [44] train-aucpr:0.101785+0.000634 test-aucpr:0.062200+0.001124   
## [45] train-aucpr:0.102349+0.000626 test-aucpr:0.062285+0.001079   
## [46] train-aucpr:0.102888+0.000537 test-aucpr:0.062261+0.001100   
## [47] train-aucpr:0.103460+0.000663 test-aucpr:0.062276+0.001090   
## [48] train-aucpr:0.104084+0.000756 test-aucpr:0.062298+0.001098   
## [49] train-aucpr:0.104650+0.000848 test-aucpr:0.062266+0.001089   
## [50] train-aucpr:0.105229+0.000786 test-aucpr:0.062303+0.001058   
## [51] train-aucpr:0.105784+0.000880 test-aucpr:0.062283+0.001086   
## [52] train-aucpr:0.106492+0.000826 test-aucpr:0.062291+0.001129   
## [53] train-aucpr:0.107174+0.000923 test-aucpr:0.062265+0.001158   
## [54] train-aucpr:0.107941+0.001047 test-aucpr:0.062345+0.001126   
## [55] train-aucpr:0.108453+0.001122 test-aucpr:0.062336+0.001110   
## [56] train-aucpr:0.108930+0.000861 test-aucpr:0.062323+0.001108   
## [57] train-aucpr:0.109446+0.000750 test-aucpr:0.062271+0.001103   
## [58] train-aucpr:0.109922+0.000638 test-aucpr:0.062193+0.001188   
## [59] train-aucpr:0.110590+0.000851 test-aucpr:0.062192+0.001160   
## Stopping. Best iteration:  
## [54] train-aucpr:0.107941+0.001047 test-aucpr:0.062345+0.001126

sel\_rounds2 <- xgb.data.cv2$best\_iteration  
AUC2 <- xgb.data.cv2$evaluation\_log[sel\_rounds2]  
AUC2

## iter train\_aucpr\_mean train\_aucpr\_std test\_aucpr\_mean test\_aucpr\_std  
## 1: 54 0.1079412 0.001046657 0.062345 0.001126319

best\_xgb <- ifelse(AUC2 > AUC, "AUC2", "AUC")  
best\_auc <- ifelse(AUC2 > AUC, AUC2, AUC)  
  
#  
# # Weight plus 2  
xgb.data.cv3 = xgb.cv(data = pred\_tr, label = resp\_tr,  
 max.depth = 6, eta = .1, nrounds = 200, min\_child\_weight = 1,  
 nfold = 5, early\_stopping\_rounds = 5, scale\_pos\_weight = neg.rate+2,  
 objective = "binary:logistic", eval\_metric = "aucpr")

## [1] train-aucpr:0.061542+0.000649 test-aucpr:0.055775+0.001133   
## Multiple eval metrics are present. Will use test\_aucpr for early stopping.  
## Will train until test\_aucpr hasn't improved in 5 rounds.  
##   
## [2] train-aucpr:0.064058+0.000488 test-aucpr:0.057192+0.001069   
## [3] train-aucpr:0.066319+0.000791 test-aucpr:0.057755+0.001500   
## [4] train-aucpr:0.068240+0.001111 test-aucpr:0.058480+0.001476   
## [5] train-aucpr:0.069644+0.000718 test-aucpr:0.059258+0.001364   
## [6] train-aucpr:0.071170+0.000320 test-aucpr:0.059560+0.001410   
## [7] train-aucpr:0.072608+0.000253 test-aucpr:0.059815+0.001667   
## [8] train-aucpr:0.073766+0.000666 test-aucpr:0.060145+0.001901   
## [9] train-aucpr:0.075085+0.000631 test-aucpr:0.060128+0.002029   
## [10] train-aucpr:0.075944+0.000973 test-aucpr:0.060371+0.002046   
## [11] train-aucpr:0.076690+0.000964 test-aucpr:0.060522+0.001979   
## [12] train-aucpr:0.077591+0.001058 test-aucpr:0.060690+0.001896   
## [13] train-aucpr:0.078478+0.001047 test-aucpr:0.060812+0.002085   
## [14] train-aucpr:0.079003+0.001245 test-aucpr:0.061012+0.002163   
## [15] train-aucpr:0.079880+0.001374 test-aucpr:0.061085+0.002176   
## [16] train-aucpr:0.080548+0.001301 test-aucpr:0.061187+0.002179   
## [17] train-aucpr:0.081395+0.001210 test-aucpr:0.061275+0.002209   
## [18] train-aucpr:0.082219+0.001479 test-aucpr:0.061320+0.002267   
## [19] train-aucpr:0.082955+0.001425 test-aucpr:0.061424+0.002304   
## [20] train-aucpr:0.083551+0.001471 test-aucpr:0.061437+0.002308   
## [21] train-aucpr:0.084227+0.001712 test-aucpr:0.061538+0.002348   
## [22] train-aucpr:0.085148+0.001717 test-aucpr:0.061546+0.002441   
## [23] train-aucpr:0.086015+0.001753 test-aucpr:0.061678+0.002413   
## [24] train-aucpr:0.086888+0.001934 test-aucpr:0.061668+0.002495   
## [25] train-aucpr:0.087621+0.001976 test-aucpr:0.061757+0.002426   
## [26] train-aucpr:0.088483+0.001962 test-aucpr:0.061793+0.002409   
## [27] train-aucpr:0.088992+0.001890 test-aucpr:0.061876+0.002493   
## [28] train-aucpr:0.089654+0.001936 test-aucpr:0.061894+0.002586   
## [29] train-aucpr:0.090529+0.002245 test-aucpr:0.062046+0.002609   
## [30] train-aucpr:0.091202+0.002307 test-aucpr:0.062055+0.002597   
## [31] train-aucpr:0.091985+0.002280 test-aucpr:0.062142+0.002610   
## [32] train-aucpr:0.092748+0.002276 test-aucpr:0.062102+0.002605   
## [33] train-aucpr:0.093279+0.002330 test-aucpr:0.062142+0.002598   
## [34] train-aucpr:0.093916+0.002368 test-aucpr:0.062202+0.002643   
## [35] train-aucpr:0.094553+0.002405 test-aucpr:0.062246+0.002641   
## [36] train-aucpr:0.095433+0.002086 test-aucpr:0.062270+0.002738   
## [37] train-aucpr:0.096241+0.002070 test-aucpr:0.062325+0.002783   
## [38] train-aucpr:0.096801+0.002118 test-aucpr:0.062336+0.002756   
## [39] train-aucpr:0.097421+0.002121 test-aucpr:0.062371+0.002793   
## [40] train-aucpr:0.097997+0.002044 test-aucpr:0.062374+0.002798   
## [41] train-aucpr:0.098704+0.002086 test-aucpr:0.062298+0.002677   
## [42] train-aucpr:0.099519+0.001980 test-aucpr:0.062371+0.002640   
## [43] train-aucpr:0.100308+0.002263 test-aucpr:0.062355+0.002668   
## [44] train-aucpr:0.100945+0.002361 test-aucpr:0.062365+0.002705   
## [45] train-aucpr:0.101543+0.002504 test-aucpr:0.062314+0.002699   
## Stopping. Best iteration:  
## [40] train-aucpr:0.097997+0.002044 test-aucpr:0.062374+0.002798

sel\_rounds3 <- xgb.data.cv3$best\_iteration  
AUC3 <- xgb.data.cv3$evaluation\_log[sel\_rounds3]  
AUC3

## iter train\_aucpr\_mean train\_aucpr\_std test\_aucpr\_mean test\_aucpr\_std  
## 1: 40 0.0979972 0.002043691 0.062374 0.002798124

best\_xgb <- ifelse(AUC3 > best\_auc, "AUC3", best\_xgb)  
best\_auc <- ifelse(AUC3 > best\_auc, AUC3, best\_auc)  
  
# # Looks like the recommended weighting was the best.   
# # Now to try a different value of eta.  
  
xgb.data.cv4 = xgb.cv(data = pred\_tr, label = resp\_tr,  
 max.depth = 6, eta = .05, nrounds = 200, min\_child\_weight = 1,  
 nfold = 5, early\_stopping\_rounds = 5, scale\_pos\_weight = neg.rate,  
 objective = "binary:logistic", eval\_metric = "aucpr")

## [1] train-aucpr:0.062062+0.000961 test-aucpr:0.055245+0.001779   
## Multiple eval metrics are present. Will use test\_aucpr for early stopping.  
## Will train until test\_aucpr hasn't improved in 5 rounds.  
##   
## [2] train-aucpr:0.063962+0.000346 test-aucpr:0.056403+0.002339   
## [3] train-aucpr:0.065590+0.000822 test-aucpr:0.057030+0.002275   
## [4] train-aucpr:0.066382+0.000780 test-aucpr:0.057364+0.002329   
## [5] train-aucpr:0.067294+0.000637 test-aucpr:0.057763+0.002377   
## [6] train-aucpr:0.068461+0.000693 test-aucpr:0.058122+0.002336   
## [7] train-aucpr:0.069498+0.000390 test-aucpr:0.058460+0.002421   
## [8] train-aucpr:0.070299+0.000453 test-aucpr:0.058788+0.002560   
## [9] train-aucpr:0.071070+0.000612 test-aucpr:0.058744+0.002366   
## [10] train-aucpr:0.071740+0.000612 test-aucpr:0.058960+0.002517   
## [11] train-aucpr:0.072501+0.000538 test-aucpr:0.059375+0.002690   
## [12] train-aucpr:0.073084+0.000953 test-aucpr:0.059655+0.002683   
## [13] train-aucpr:0.073748+0.000827 test-aucpr:0.059829+0.002650   
## [14] train-aucpr:0.074479+0.000989 test-aucpr:0.059873+0.002525   
## [15] train-aucpr:0.074990+0.000959 test-aucpr:0.060038+0.002518   
## [16] train-aucpr:0.075577+0.000959 test-aucpr:0.060088+0.002501   
## [17] train-aucpr:0.076181+0.000997 test-aucpr:0.060271+0.002609   
## [18] train-aucpr:0.076720+0.001106 test-aucpr:0.060386+0.002640   
## [19] train-aucpr:0.077202+0.001168 test-aucpr:0.060429+0.002713   
## [20] train-aucpr:0.077707+0.001187 test-aucpr:0.060625+0.002675   
## [21] train-aucpr:0.078072+0.001153 test-aucpr:0.060610+0.002604   
## [22] train-aucpr:0.078332+0.001189 test-aucpr:0.060755+0.002597   
## [23] train-aucpr:0.078799+0.001343 test-aucpr:0.060894+0.002623   
## [24] train-aucpr:0.079310+0.001391 test-aucpr:0.060986+0.002697   
## [25] train-aucpr:0.079683+0.001376 test-aucpr:0.061104+0.002646   
## [26] train-aucpr:0.080145+0.001463 test-aucpr:0.061197+0.002587   
## [27] train-aucpr:0.080606+0.001428 test-aucpr:0.061333+0.002474   
## [28] train-aucpr:0.080947+0.001442 test-aucpr:0.061300+0.002492   
## [29] train-aucpr:0.081424+0.001421 test-aucpr:0.061311+0.002469   
## [30] train-aucpr:0.081853+0.001444 test-aucpr:0.061386+0.002438   
## [31] train-aucpr:0.082268+0.001485 test-aucpr:0.061460+0.002462   
## [32] train-aucpr:0.082710+0.001443 test-aucpr:0.061557+0.002465   
## [33] train-aucpr:0.082986+0.001540 test-aucpr:0.061570+0.002437   
## [34] train-aucpr:0.083317+0.001542 test-aucpr:0.061591+0.002450   
## [35] train-aucpr:0.083799+0.001550 test-aucpr:0.061665+0.002389   
## [36] train-aucpr:0.084398+0.001767 test-aucpr:0.061727+0.002434   
## [37] train-aucpr:0.084890+0.001690 test-aucpr:0.061807+0.002367   
## [38] train-aucpr:0.085319+0.001661 test-aucpr:0.061889+0.002443   
## [39] train-aucpr:0.085810+0.001625 test-aucpr:0.061977+0.002367   
## [40] train-aucpr:0.086264+0.001613 test-aucpr:0.061952+0.002368   
## [41] train-aucpr:0.086785+0.001615 test-aucpr:0.061988+0.002332   
## [42] train-aucpr:0.087226+0.001613 test-aucpr:0.062068+0.002249   
## [43] train-aucpr:0.087635+0.001607 test-aucpr:0.062147+0.002153   
## [44] train-aucpr:0.088115+0.001530 test-aucpr:0.062171+0.002100   
## [45] train-aucpr:0.088517+0.001565 test-aucpr:0.062149+0.002179   
## [46] train-aucpr:0.089018+0.001571 test-aucpr:0.062225+0.002225   
## [47] train-aucpr:0.089437+0.001532 test-aucpr:0.062292+0.002210   
## [48] train-aucpr:0.089907+0.001589 test-aucpr:0.062324+0.002244   
## [49] train-aucpr:0.090380+0.001594 test-aucpr:0.062360+0.002262   
## [50] train-aucpr:0.090833+0.001645 test-aucpr:0.062362+0.002271   
## [51] train-aucpr:0.091232+0.001702 test-aucpr:0.062459+0.002148   
## [52] train-aucpr:0.091580+0.001738 test-aucpr:0.062416+0.002186   
## [53] train-aucpr:0.091942+0.001714 test-aucpr:0.062432+0.002224   
## [54] train-aucpr:0.092261+0.001692 test-aucpr:0.062528+0.002142   
## [55] train-aucpr:0.092567+0.001717 test-aucpr:0.062570+0.002138   
## [56] train-aucpr:0.092932+0.001742 test-aucpr:0.062656+0.002129   
## [57] train-aucpr:0.093383+0.001749 test-aucpr:0.062667+0.002185   
## [58] train-aucpr:0.093734+0.001853 test-aucpr:0.062690+0.002165   
## [59] train-aucpr:0.094083+0.001875 test-aucpr:0.062718+0.002200   
## [60] train-aucpr:0.094391+0.001906 test-aucpr:0.062767+0.002242   
## [61] train-aucpr:0.094747+0.001860 test-aucpr:0.062775+0.002198   
## [62] train-aucpr:0.095113+0.001907 test-aucpr:0.062829+0.002251   
## [63] train-aucpr:0.095578+0.001941 test-aucpr:0.062859+0.002244   
## [64] train-aucpr:0.095945+0.002022 test-aucpr:0.062860+0.002224   
## [65] train-aucpr:0.096288+0.002101 test-aucpr:0.062873+0.002256   
## [66] train-aucpr:0.096616+0.002050 test-aucpr:0.062868+0.002263   
## [67] train-aucpr:0.096924+0.002090 test-aucpr:0.062924+0.002270   
## [68] train-aucpr:0.097327+0.002160 test-aucpr:0.062918+0.002271   
## [69] train-aucpr:0.097674+0.002112 test-aucpr:0.062937+0.002242   
## [70] train-aucpr:0.098116+0.002153 test-aucpr:0.062987+0.002237   
## [71] train-aucpr:0.098487+0.002166 test-aucpr:0.063000+0.002227   
## [72] train-aucpr:0.098946+0.002182 test-aucpr:0.063020+0.002264   
## [73] train-aucpr:0.099296+0.002198 test-aucpr:0.063035+0.002262   
## [74] train-aucpr:0.099752+0.002233 test-aucpr:0.062980+0.002243   
## [75] train-aucpr:0.100033+0.002159 test-aucpr:0.063054+0.002193   
## [76] train-aucpr:0.100330+0.002101 test-aucpr:0.063048+0.002190   
## [77] train-aucpr:0.100680+0.002131 test-aucpr:0.063023+0.002172   
## [78] train-aucpr:0.101019+0.002143 test-aucpr:0.063104+0.002152   
## [79] train-aucpr:0.101373+0.002138 test-aucpr:0.063111+0.002097   
## [80] train-aucpr:0.101779+0.002191 test-aucpr:0.063129+0.002130   
## [81] train-aucpr:0.102097+0.002239 test-aucpr:0.063133+0.002137   
## [82] train-aucpr:0.102392+0.002213 test-aucpr:0.063200+0.002122   
## [83] train-aucpr:0.102805+0.002200 test-aucpr:0.063237+0.002110   
## [84] train-aucpr:0.103094+0.002250 test-aucpr:0.063254+0.002092   
## [85] train-aucpr:0.103482+0.002259 test-aucpr:0.063276+0.002101   
## [86] train-aucpr:0.103866+0.002244 test-aucpr:0.063225+0.002099   
## [87] train-aucpr:0.104179+0.002324 test-aucpr:0.063225+0.002124   
## [88] train-aucpr:0.104497+0.002413 test-aucpr:0.063213+0.002129   
## [89] train-aucpr:0.104794+0.002382 test-aucpr:0.063254+0.002119   
## [90] train-aucpr:0.105098+0.002426 test-aucpr:0.063274+0.002072   
## Stopping. Best iteration:  
## [85] train-aucpr:0.103482+0.002259 test-aucpr:0.063276+0.002101

sel\_rounds4 <- xgb.data.cv4$best\_iteration  
  
AUC4<- xgb.data.cv4$evaluation\_log[sel\_rounds4]  
AUC4

## iter train\_aucpr\_mean train\_aucpr\_std test\_aucpr\_mean test\_aucpr\_std  
## 1: 85 0.1034818 0.002259056 0.0632758 0.002101165

best\_xgb <- ifelse(AUC4 > best\_auc, "AUC4", best\_xgb)  
best\_auc <- ifelse(AUC4 > best\_auc, AUC4, best\_auc)  
  
# # I think this helped improve test auc a bit. Now try lower eta = .01  
#   
  
xgb.data.cv5 = xgb.cv(data = pred\_tr, label = resp\_tr,  
 max.depth = 6, eta = .01, nrounds = 200, min\_child\_weight = 1,  
 nfold = 5, early\_stopping\_rounds = 10, scale\_pos\_weight = neg.rate,  
 objective = "binary:logistic", eval\_metric = "aucpr")

## [1] train-aucpr:0.062361+0.000528 test-aucpr:0.055210+0.001031   
## Multiple eval metrics are present. Will use test\_aucpr for early stopping.  
## Will train until test\_aucpr hasn't improved in 10 rounds.  
##   
## [2] train-aucpr:0.062917+0.000686 test-aucpr:0.055423+0.001053   
## [3] train-aucpr:0.063094+0.000666 test-aucpr:0.055466+0.001120   
## [4] train-aucpr:0.063325+0.000664 test-aucpr:0.055431+0.001039   
## [5] train-aucpr:0.063791+0.000819 test-aucpr:0.055591+0.001098   
## [6] train-aucpr:0.064070+0.000934 test-aucpr:0.055655+0.001067   
## [7] train-aucpr:0.064272+0.000936 test-aucpr:0.055707+0.001068   
## [8] train-aucpr:0.064541+0.001155 test-aucpr:0.055701+0.001127   
## [9] train-aucpr:0.064894+0.001102 test-aucpr:0.055801+0.001095   
## [10] train-aucpr:0.065035+0.001073 test-aucpr:0.055814+0.001120   
## [11] train-aucpr:0.065428+0.001050 test-aucpr:0.055985+0.001188   
## [12] train-aucpr:0.065839+0.001083 test-aucpr:0.056208+0.001088   
## [13] train-aucpr:0.066053+0.001089 test-aucpr:0.056310+0.001164   
## [14] train-aucpr:0.066203+0.001006 test-aucpr:0.056356+0.001022   
## [15] train-aucpr:0.066322+0.000973 test-aucpr:0.056380+0.001060   
## [16] train-aucpr:0.066659+0.001115 test-aucpr:0.056493+0.001091   
## [17] train-aucpr:0.066901+0.000971 test-aucpr:0.056533+0.000961   
## [18] train-aucpr:0.067017+0.000825 test-aucpr:0.056606+0.001108   
## [19] train-aucpr:0.067412+0.001020 test-aucpr:0.056701+0.001075   
## [20] train-aucpr:0.067509+0.000985 test-aucpr:0.056716+0.001083   
## [21] train-aucpr:0.067777+0.000919 test-aucpr:0.056783+0.001059   
## [22] train-aucpr:0.068044+0.001015 test-aucpr:0.056858+0.001141   
## [23] train-aucpr:0.068149+0.000873 test-aucpr:0.056851+0.001139   
## [24] train-aucpr:0.068335+0.000887 test-aucpr:0.056943+0.001149   
## [25] train-aucpr:0.068489+0.000770 test-aucpr:0.057055+0.001239   
## [26] train-aucpr:0.068650+0.000731 test-aucpr:0.057126+0.001264   
## [27] train-aucpr:0.068856+0.000654 test-aucpr:0.057193+0.001279   
## [28] train-aucpr:0.069040+0.000576 test-aucpr:0.057298+0.001300   
## [29] train-aucpr:0.069213+0.000585 test-aucpr:0.057279+0.001319   
## [30] train-aucpr:0.069301+0.000536 test-aucpr:0.057365+0.001304   
## [31] train-aucpr:0.069492+0.000522 test-aucpr:0.057436+0.001340   
## [32] train-aucpr:0.069627+0.000514 test-aucpr:0.057498+0.001320   
## [33] train-aucpr:0.069779+0.000570 test-aucpr:0.057530+0.001344   
## [34] train-aucpr:0.069976+0.000522 test-aucpr:0.057611+0.001382   
## [35] train-aucpr:0.070200+0.000595 test-aucpr:0.057661+0.001370   
## [36] train-aucpr:0.070322+0.000750 test-aucpr:0.057723+0.001330   
## [37] train-aucpr:0.070546+0.000818 test-aucpr:0.057791+0.001351   
## [38] train-aucpr:0.070703+0.000730 test-aucpr:0.057818+0.001343   
## [39] train-aucpr:0.070853+0.000876 test-aucpr:0.057895+0.001329   
## [40] train-aucpr:0.070972+0.000944 test-aucpr:0.057930+0.001345   
## [41] train-aucpr:0.071146+0.000963 test-aucpr:0.058008+0.001339   
## [42] train-aucpr:0.071313+0.000931 test-aucpr:0.057991+0.001321   
## [43] train-aucpr:0.071446+0.000916 test-aucpr:0.058062+0.001334   
## [44] train-aucpr:0.071604+0.000842 test-aucpr:0.058067+0.001333   
## [45] train-aucpr:0.071820+0.000911 test-aucpr:0.058089+0.001308   
## [46] train-aucpr:0.071905+0.000853 test-aucpr:0.058131+0.001340   
## [47] train-aucpr:0.072117+0.000964 test-aucpr:0.058158+0.001316   
## [48] train-aucpr:0.072272+0.001043 test-aucpr:0.058202+0.001324   
## [49] train-aucpr:0.072396+0.001007 test-aucpr:0.058200+0.001306   
## [50] train-aucpr:0.072537+0.001042 test-aucpr:0.058254+0.001293   
## [51] train-aucpr:0.072748+0.001067 test-aucpr:0.058303+0.001303   
## [52] train-aucpr:0.072877+0.001066 test-aucpr:0.058326+0.001285   
## [53] train-aucpr:0.073031+0.001037 test-aucpr:0.058381+0.001311   
## [54] train-aucpr:0.073138+0.000989 test-aucpr:0.058436+0.001305   
## [55] train-aucpr:0.073281+0.001041 test-aucpr:0.058459+0.001308   
## [56] train-aucpr:0.073451+0.000963 test-aucpr:0.058511+0.001291   
## [57] train-aucpr:0.073529+0.001025 test-aucpr:0.058538+0.001257   
## [58] train-aucpr:0.073708+0.000977 test-aucpr:0.058580+0.001261   
## [59] train-aucpr:0.073817+0.000987 test-aucpr:0.058650+0.001246   
## [60] train-aucpr:0.073941+0.000962 test-aucpr:0.058675+0.001237   
## [61] train-aucpr:0.074103+0.000983 test-aucpr:0.058725+0.001214   
## [62] train-aucpr:0.074222+0.000997 test-aucpr:0.058757+0.001214   
## [63] train-aucpr:0.074378+0.000970 test-aucpr:0.058792+0.001195   
## [64] train-aucpr:0.074533+0.000899 test-aucpr:0.058833+0.001163   
## [65] train-aucpr:0.074659+0.000920 test-aucpr:0.058865+0.001145   
## [66] train-aucpr:0.074816+0.000907 test-aucpr:0.058894+0.001138   
## [67] train-aucpr:0.074918+0.000900 test-aucpr:0.058938+0.001125   
## [68] train-aucpr:0.075074+0.000937 test-aucpr:0.058979+0.001116   
## [69] train-aucpr:0.075273+0.000941 test-aucpr:0.059001+0.001105   
## [70] train-aucpr:0.075406+0.000956 test-aucpr:0.059073+0.001115   
## [71] train-aucpr:0.075551+0.000993 test-aucpr:0.059074+0.001095   
## [72] train-aucpr:0.075681+0.000940 test-aucpr:0.059112+0.001089   
## [73] train-aucpr:0.075858+0.000933 test-aucpr:0.059146+0.001071   
## [74] train-aucpr:0.076014+0.000921 test-aucpr:0.059181+0.001065   
## [75] train-aucpr:0.076120+0.000982 test-aucpr:0.059216+0.001066   
## [76] train-aucpr:0.076214+0.000956 test-aucpr:0.059251+0.001070   
## [77] train-aucpr:0.076390+0.000935 test-aucpr:0.059263+0.001067   
## [78] train-aucpr:0.076464+0.000995 test-aucpr:0.059286+0.001061   
## [79] train-aucpr:0.076601+0.000990 test-aucpr:0.059320+0.001083   
## [80] train-aucpr:0.076690+0.000982 test-aucpr:0.059348+0.001074   
## [81] train-aucpr:0.076791+0.000960 test-aucpr:0.059398+0.001084   
## [82] train-aucpr:0.076864+0.000975 test-aucpr:0.059414+0.001090   
## [83] train-aucpr:0.076962+0.000948 test-aucpr:0.059440+0.001101   
## [84] train-aucpr:0.077046+0.000925 test-aucpr:0.059476+0.001108   
## [85] train-aucpr:0.077154+0.000957 test-aucpr:0.059495+0.001126   
## [86] train-aucpr:0.077247+0.000925 test-aucpr:0.059518+0.001108   
## [87] train-aucpr:0.077321+0.000935 test-aucpr:0.059564+0.001094   
## [88] train-aucpr:0.077415+0.000965 test-aucpr:0.059578+0.001095   
## [89] train-aucpr:0.077501+0.000938 test-aucpr:0.059592+0.001092   
## [90] train-aucpr:0.077592+0.000920 test-aucpr:0.059612+0.001094   
## [91] train-aucpr:0.077683+0.000888 test-aucpr:0.059645+0.001102   
## [92] train-aucpr:0.077785+0.000863 test-aucpr:0.059686+0.001104   
## [93] train-aucpr:0.077859+0.000823 test-aucpr:0.059709+0.001101   
## [94] train-aucpr:0.077961+0.000859 test-aucpr:0.059736+0.001095   
## [95] train-aucpr:0.078044+0.000853 test-aucpr:0.059758+0.001087   
## [96] train-aucpr:0.078096+0.000850 test-aucpr:0.059770+0.001071   
## [97] train-aucpr:0.078206+0.000863 test-aucpr:0.059786+0.001072   
## [98] train-aucpr:0.078280+0.000827 test-aucpr:0.059818+0.001055   
## [99] train-aucpr:0.078413+0.000833 test-aucpr:0.059832+0.001033   
## [100] train-aucpr:0.078458+0.000828 test-aucpr:0.059854+0.001033   
## [101] train-aucpr:0.078566+0.000857 test-aucpr:0.059871+0.001015   
## [102] train-aucpr:0.078674+0.000881 test-aucpr:0.059875+0.000986   
## [103] train-aucpr:0.078782+0.000881 test-aucpr:0.059912+0.001003   
## [104] train-aucpr:0.078915+0.000907 test-aucpr:0.059919+0.000984   
## [105] train-aucpr:0.078990+0.000912 test-aucpr:0.059944+0.000981   
## [106] train-aucpr:0.079062+0.000940 test-aucpr:0.059944+0.000976   
## [107] train-aucpr:0.079161+0.001002 test-aucpr:0.059966+0.000974   
## [108] train-aucpr:0.079217+0.000978 test-aucpr:0.059982+0.000984   
## [109] train-aucpr:0.079315+0.000994 test-aucpr:0.060002+0.001003   
## [110] train-aucpr:0.079409+0.000983 test-aucpr:0.059965+0.000970   
## [111] train-aucpr:0.079528+0.001011 test-aucpr:0.059983+0.000954   
## [112] train-aucpr:0.079590+0.001026 test-aucpr:0.060014+0.000963   
## [113] train-aucpr:0.079709+0.001034 test-aucpr:0.060024+0.000981   
## [114] train-aucpr:0.079768+0.001042 test-aucpr:0.060043+0.000963   
## [115] train-aucpr:0.079856+0.001038 test-aucpr:0.060064+0.000963   
## [116] train-aucpr:0.079937+0.001073 test-aucpr:0.060088+0.000967   
## [117] train-aucpr:0.080036+0.001048 test-aucpr:0.060110+0.000956   
## [118] train-aucpr:0.080075+0.001065 test-aucpr:0.060137+0.000956   
## [119] train-aucpr:0.080172+0.001054 test-aucpr:0.060181+0.000955   
## [120] train-aucpr:0.080276+0.001092 test-aucpr:0.060198+0.000955   
## [121] train-aucpr:0.080381+0.001095 test-aucpr:0.060221+0.000968   
## [122] train-aucpr:0.080484+0.001070 test-aucpr:0.060253+0.000962   
## [123] train-aucpr:0.080555+0.001067 test-aucpr:0.060309+0.000966   
## [124] train-aucpr:0.080632+0.001057 test-aucpr:0.060311+0.000957   
## [125] train-aucpr:0.080760+0.001077 test-aucpr:0.060345+0.000968   
## [126] train-aucpr:0.080839+0.001075 test-aucpr:0.060353+0.000974   
## [127] train-aucpr:0.080884+0.001116 test-aucpr:0.060370+0.000956   
## [128] train-aucpr:0.080959+0.001118 test-aucpr:0.060387+0.000959   
## [129] train-aucpr:0.081059+0.001085 test-aucpr:0.060407+0.000964   
## [130] train-aucpr:0.081137+0.001137 test-aucpr:0.060417+0.000956   
## [131] train-aucpr:0.081195+0.001118 test-aucpr:0.060449+0.000964   
## [132] train-aucpr:0.081236+0.001136 test-aucpr:0.060468+0.000960   
## [133] train-aucpr:0.081310+0.001128 test-aucpr:0.060513+0.000981   
## [134] train-aucpr:0.081422+0.001128 test-aucpr:0.060534+0.000986   
## [135] train-aucpr:0.081474+0.001133 test-aucpr:0.060548+0.000991   
## [136] train-aucpr:0.081546+0.001176 test-aucpr:0.060563+0.000997   
## [137] train-aucpr:0.081624+0.001159 test-aucpr:0.060568+0.000980   
## [138] train-aucpr:0.081685+0.001170 test-aucpr:0.060634+0.001018   
## [139] train-aucpr:0.081756+0.001172 test-aucpr:0.060645+0.001023   
## [140] train-aucpr:0.081799+0.001184 test-aucpr:0.060658+0.001016   
## [141] train-aucpr:0.081899+0.001206 test-aucpr:0.060669+0.001019   
## [142] train-aucpr:0.081970+0.001199 test-aucpr:0.060682+0.001013   
## [143] train-aucpr:0.082051+0.001165 test-aucpr:0.060694+0.001024   
## [144] train-aucpr:0.082117+0.001151 test-aucpr:0.060709+0.001031   
## [145] train-aucpr:0.082202+0.001119 test-aucpr:0.060720+0.001035   
## [146] train-aucpr:0.082266+0.001105 test-aucpr:0.060730+0.001045   
## [147] train-aucpr:0.082350+0.001110 test-aucpr:0.060754+0.001049   
## [148] train-aucpr:0.082427+0.001084 test-aucpr:0.060781+0.001054   
## [149] train-aucpr:0.082521+0.001092 test-aucpr:0.060815+0.001057   
## [150] train-aucpr:0.082586+0.001062 test-aucpr:0.060825+0.001074   
## [151] train-aucpr:0.082666+0.001073 test-aucpr:0.060843+0.001070   
## [152] train-aucpr:0.082737+0.001063 test-aucpr:0.060844+0.001069   
## [153] train-aucpr:0.082808+0.001062 test-aucpr:0.060853+0.001079   
## [154] train-aucpr:0.082919+0.001016 test-aucpr:0.060861+0.001088   
## [155] train-aucpr:0.082985+0.001032 test-aucpr:0.060829+0.001075   
## [156] train-aucpr:0.083048+0.001018 test-aucpr:0.060842+0.001087   
## [157] train-aucpr:0.083142+0.001002 test-aucpr:0.060859+0.001091   
## [158] train-aucpr:0.083261+0.001003 test-aucpr:0.060844+0.001098   
## [159] train-aucpr:0.083327+0.001017 test-aucpr:0.060871+0.001104   
## [160] train-aucpr:0.083412+0.001004 test-aucpr:0.060881+0.001109   
## [161] train-aucpr:0.083477+0.000996 test-aucpr:0.060904+0.001104   
## [162] train-aucpr:0.083557+0.000996 test-aucpr:0.060900+0.001113   
## [163] train-aucpr:0.083678+0.000978 test-aucpr:0.060905+0.001120   
## [164] train-aucpr:0.083755+0.001016 test-aucpr:0.060903+0.001130   
## [165] train-aucpr:0.083821+0.001024 test-aucpr:0.060914+0.001134   
## [166] train-aucpr:0.083888+0.001021 test-aucpr:0.060929+0.001143   
## [167] train-aucpr:0.083955+0.000998 test-aucpr:0.060932+0.001153   
## [168] train-aucpr:0.084012+0.001004 test-aucpr:0.060951+0.001142   
## [169] train-aucpr:0.084088+0.001013 test-aucpr:0.060958+0.001136   
## [170] train-aucpr:0.084182+0.001008 test-aucpr:0.060960+0.001142   
## [171] train-aucpr:0.084263+0.000989 test-aucpr:0.060961+0.001146   
## [172] train-aucpr:0.084329+0.000979 test-aucpr:0.060973+0.001157   
## [173] train-aucpr:0.084404+0.000979 test-aucpr:0.060978+0.001153   
## [174] train-aucpr:0.084445+0.000964 test-aucpr:0.060969+0.001154   
## [175] train-aucpr:0.084510+0.000960 test-aucpr:0.060988+0.001151   
## [176] train-aucpr:0.084570+0.000952 test-aucpr:0.060998+0.001159   
## [177] train-aucpr:0.084662+0.000953 test-aucpr:0.061011+0.001162   
## [178] train-aucpr:0.084764+0.000948 test-aucpr:0.061013+0.001152   
## [179] train-aucpr:0.084836+0.000948 test-aucpr:0.061008+0.001155   
## [180] train-aucpr:0.084894+0.000945 test-aucpr:0.061021+0.001153   
## [181] train-aucpr:0.084971+0.000949 test-aucpr:0.061010+0.001166   
## [182] train-aucpr:0.085049+0.000970 test-aucpr:0.061026+0.001167   
## [183] train-aucpr:0.085097+0.000958 test-aucpr:0.061028+0.001161   
## [184] train-aucpr:0.085180+0.000935 test-aucpr:0.061037+0.001162   
## [185] train-aucpr:0.085257+0.000934 test-aucpr:0.061049+0.001155   
## [186] train-aucpr:0.085327+0.000926 test-aucpr:0.061052+0.001158   
## [187] train-aucpr:0.085371+0.000922 test-aucpr:0.061059+0.001156   
## [188] train-aucpr:0.085479+0.000926 test-aucpr:0.061065+0.001153   
## [189] train-aucpr:0.085559+0.000960 test-aucpr:0.061082+0.001150   
## [190] train-aucpr:0.085642+0.000961 test-aucpr:0.061072+0.001169   
## [191] train-aucpr:0.085745+0.000939 test-aucpr:0.061083+0.001175   
## [192] train-aucpr:0.085839+0.000951 test-aucpr:0.061081+0.001176   
## [193] train-aucpr:0.085899+0.000945 test-aucpr:0.061089+0.001171   
## [194] train-aucpr:0.085993+0.000946 test-aucpr:0.061085+0.001179   
## [195] train-aucpr:0.086056+0.000948 test-aucpr:0.061082+0.001179   
## [196] train-aucpr:0.086145+0.000930 test-aucpr:0.061086+0.001192   
## [197] train-aucpr:0.086225+0.000936 test-aucpr:0.061088+0.001197   
## [198] train-aucpr:0.086305+0.000937 test-aucpr:0.061101+0.001198   
## [199] train-aucpr:0.086407+0.000942 test-aucpr:0.061105+0.001197   
## [200] train-aucpr:0.086498+0.000947 test-aucpr:0.061114+0.001195

sel\_rounds5 <- xgb.data.cv5$best\_iteration  
AUC5 <- xgb.data.cv5$evaluation\_log[sel\_rounds5]  
AUC5

## iter train\_aucpr\_mean train\_aucpr\_std test\_aucpr\_mean test\_aucpr\_std  
## 1: 200 0.0864978 0.0009473995 0.061114 0.001194817

best\_xgb <- ifelse(AUC5 > best\_auc, "AUC5", best\_xgb)  
best\_auc <- ifelse(AUC5 > best\_auc, AUC5, best\_auc)  
# Didn't help too much. Now try limiting max.depth to avoid overfitting  
  
xgb.data.cv6 = xgb.cv(data = pred\_tr, label = resp\_tr,   
 max.depth = 3, eta = .05, nrounds = 250, min\_child\_weight = 1,  
 nfold = 5, early\_stopping\_rounds = 10, scale\_pos\_weight = neg.rate,  
 objective = "binary:logistic", eval\_metric = "aucpr")

## [1] train-aucpr:0.051573+0.000414 test-aucpr:0.050334+0.001133   
## Multiple eval metrics are present. Will use test\_aucpr for early stopping.  
## Will train until test\_aucpr hasn't improved in 10 rounds.  
##   
## [2] train-aucpr:0.052270+0.000550 test-aucpr:0.051008+0.000953   
## [3] train-aucpr:0.052898+0.000143 test-aucpr:0.051697+0.001656   
## [4] train-aucpr:0.054105+0.000698 test-aucpr:0.052766+0.001834   
## [5] train-aucpr:0.054987+0.000605 test-aucpr:0.053418+0.002039   
## [6] train-aucpr:0.055663+0.000314 test-aucpr:0.054077+0.002060   
## [7] train-aucpr:0.055983+0.000409 test-aucpr:0.054263+0.001993   
## [8] train-aucpr:0.056601+0.000612 test-aucpr:0.054750+0.001990   
## [9] train-aucpr:0.057053+0.000617 test-aucpr:0.055190+0.002072   
## [10] train-aucpr:0.057502+0.000500 test-aucpr:0.055540+0.001895   
## [11] train-aucpr:0.057934+0.000595 test-aucpr:0.055992+0.001943   
## [12] train-aucpr:0.058400+0.000741 test-aucpr:0.056453+0.001863   
## [13] train-aucpr:0.058699+0.000567 test-aucpr:0.056830+0.001731   
## [14] train-aucpr:0.058990+0.000604 test-aucpr:0.057129+0.001851   
## [15] train-aucpr:0.059234+0.000664 test-aucpr:0.057400+0.001758   
## [16] train-aucpr:0.059589+0.000605 test-aucpr:0.057688+0.001678   
## [17] train-aucpr:0.059812+0.000628 test-aucpr:0.057871+0.001665   
## [18] train-aucpr:0.060002+0.000589 test-aucpr:0.058033+0.001641   
## [19] train-aucpr:0.060274+0.000584 test-aucpr:0.058245+0.001663   
## [20] train-aucpr:0.060538+0.000584 test-aucpr:0.058455+0.001725   
## [21] train-aucpr:0.060702+0.000533 test-aucpr:0.058622+0.001731   
## [22] train-aucpr:0.060915+0.000600 test-aucpr:0.058793+0.001878   
## [23] train-aucpr:0.061148+0.000569 test-aucpr:0.059014+0.001800   
## [24] train-aucpr:0.061304+0.000562 test-aucpr:0.059121+0.001746   
## [25] train-aucpr:0.061494+0.000622 test-aucpr:0.059288+0.001718   
## [26] train-aucpr:0.061610+0.000539 test-aucpr:0.059451+0.001791   
## [27] train-aucpr:0.061757+0.000590 test-aucpr:0.059527+0.001762   
## [28] train-aucpr:0.061876+0.000571 test-aucpr:0.059577+0.001749   
## [29] train-aucpr:0.062005+0.000601 test-aucpr:0.059667+0.001691   
## [30] train-aucpr:0.062164+0.000652 test-aucpr:0.059762+0.001733   
## [31] train-aucpr:0.062309+0.000601 test-aucpr:0.059895+0.001771   
## [32] train-aucpr:0.062459+0.000580 test-aucpr:0.060051+0.001833   
## [33] train-aucpr:0.062539+0.000597 test-aucpr:0.060136+0.001788   
## [34] train-aucpr:0.062657+0.000569 test-aucpr:0.060195+0.001847   
## [35] train-aucpr:0.062748+0.000569 test-aucpr:0.060276+0.001866   
## [36] train-aucpr:0.062855+0.000605 test-aucpr:0.060364+0.001829   
## [37] train-aucpr:0.062939+0.000612 test-aucpr:0.060435+0.001837   
## [38] train-aucpr:0.063007+0.000573 test-aucpr:0.060517+0.001911   
## [39] train-aucpr:0.063082+0.000569 test-aucpr:0.060567+0.001950   
## [40] train-aucpr:0.063135+0.000562 test-aucpr:0.060625+0.001990   
## [41] train-aucpr:0.063245+0.000546 test-aucpr:0.060725+0.002020   
## [42] train-aucpr:0.063357+0.000497 test-aucpr:0.060764+0.002012   
## [43] train-aucpr:0.063409+0.000509 test-aucpr:0.060815+0.002015   
## [44] train-aucpr:0.063532+0.000488 test-aucpr:0.060894+0.002040   
## [45] train-aucpr:0.063632+0.000526 test-aucpr:0.060967+0.001997   
## [46] train-aucpr:0.063722+0.000500 test-aucpr:0.061006+0.002056   
## [47] train-aucpr:0.063802+0.000511 test-aucpr:0.061114+0.002087   
## [48] train-aucpr:0.063874+0.000492 test-aucpr:0.061170+0.002124   
## [49] train-aucpr:0.063929+0.000499 test-aucpr:0.061221+0.002122   
## [50] train-aucpr:0.063979+0.000505 test-aucpr:0.061288+0.002140   
## [51] train-aucpr:0.064024+0.000513 test-aucpr:0.061312+0.002179   
## [52] train-aucpr:0.064083+0.000478 test-aucpr:0.061339+0.002199   
## [53] train-aucpr:0.064165+0.000447 test-aucpr:0.061405+0.002216   
## [54] train-aucpr:0.064252+0.000501 test-aucpr:0.061456+0.002178   
## [55] train-aucpr:0.064354+0.000511 test-aucpr:0.061530+0.002227   
## [56] train-aucpr:0.064418+0.000528 test-aucpr:0.061589+0.002194   
## [57] train-aucpr:0.064509+0.000501 test-aucpr:0.061699+0.002261   
## [58] train-aucpr:0.064573+0.000493 test-aucpr:0.061703+0.002249   
## [59] train-aucpr:0.064633+0.000481 test-aucpr:0.061741+0.002237   
## [60] train-aucpr:0.064695+0.000502 test-aucpr:0.061789+0.002190   
## [61] train-aucpr:0.064801+0.000495 test-aucpr:0.061862+0.002234   
## [62] train-aucpr:0.064861+0.000482 test-aucpr:0.061940+0.002219   
## [63] train-aucpr:0.064929+0.000506 test-aucpr:0.062029+0.002255   
## [64] train-aucpr:0.064973+0.000487 test-aucpr:0.061992+0.002234   
## [65] train-aucpr:0.065044+0.000487 test-aucpr:0.062025+0.002225   
## [66] train-aucpr:0.065078+0.000508 test-aucpr:0.062037+0.002204   
## [67] train-aucpr:0.065158+0.000482 test-aucpr:0.062102+0.002224   
## [68] train-aucpr:0.065228+0.000463 test-aucpr:0.062162+0.002136   
## [69] train-aucpr:0.065272+0.000449 test-aucpr:0.062159+0.002089   
## [70] train-aucpr:0.065314+0.000461 test-aucpr:0.062190+0.002094   
## [71] train-aucpr:0.065393+0.000489 test-aucpr:0.062235+0.002070   
## [72] train-aucpr:0.065452+0.000493 test-aucpr:0.062327+0.002102   
## [73] train-aucpr:0.065486+0.000486 test-aucpr:0.062347+0.002078   
## [74] train-aucpr:0.065555+0.000472 test-aucpr:0.062365+0.002083   
## [75] train-aucpr:0.065623+0.000482 test-aucpr:0.062378+0.002059   
## [76] train-aucpr:0.065693+0.000462 test-aucpr:0.062382+0.002026   
## [77] train-aucpr:0.065734+0.000485 test-aucpr:0.062432+0.002003   
## [78] train-aucpr:0.065779+0.000486 test-aucpr:0.062484+0.002009   
## [79] train-aucpr:0.065829+0.000479 test-aucpr:0.062531+0.002058   
## [80] train-aucpr:0.065880+0.000467 test-aucpr:0.062552+0.002048   
## [81] train-aucpr:0.065929+0.000472 test-aucpr:0.062529+0.002073   
## [82] train-aucpr:0.065993+0.000469 test-aucpr:0.062582+0.002076   
## [83] train-aucpr:0.066016+0.000471 test-aucpr:0.062580+0.002064   
## [84] train-aucpr:0.066074+0.000467 test-aucpr:0.062596+0.002047   
## [85] train-aucpr:0.066142+0.000457 test-aucpr:0.062601+0.002043   
## [86] train-aucpr:0.066196+0.000476 test-aucpr:0.062640+0.002041   
## [87] train-aucpr:0.066234+0.000476 test-aucpr:0.062672+0.002049   
## [88] train-aucpr:0.066291+0.000468 test-aucpr:0.062724+0.002065   
## [89] train-aucpr:0.066348+0.000456 test-aucpr:0.062752+0.002054   
## [90] train-aucpr:0.066390+0.000461 test-aucpr:0.062768+0.002042   
## [91] train-aucpr:0.066425+0.000447 test-aucpr:0.062790+0.002068   
## [92] train-aucpr:0.066485+0.000440 test-aucpr:0.062814+0.002060   
## [93] train-aucpr:0.066556+0.000442 test-aucpr:0.062873+0.002053   
## [94] train-aucpr:0.066594+0.000435 test-aucpr:0.062897+0.002067   
## [95] train-aucpr:0.066630+0.000446 test-aucpr:0.062924+0.002063   
## [96] train-aucpr:0.066677+0.000470 test-aucpr:0.062925+0.002062   
## [97] train-aucpr:0.066737+0.000491 test-aucpr:0.062959+0.002062   
## [98] train-aucpr:0.066759+0.000474 test-aucpr:0.062976+0.002069   
## [99] train-aucpr:0.066807+0.000462 test-aucpr:0.062988+0.002079   
## [100] train-aucpr:0.066851+0.000442 test-aucpr:0.062991+0.002111   
## [101] train-aucpr:0.066903+0.000426 test-aucpr:0.063030+0.002132   
## [102] train-aucpr:0.066954+0.000430 test-aucpr:0.063053+0.002120   
## [103] train-aucpr:0.066994+0.000446 test-aucpr:0.063062+0.002126   
## [104] train-aucpr:0.067038+0.000435 test-aucpr:0.063087+0.002120   
## [105] train-aucpr:0.067068+0.000436 test-aucpr:0.063121+0.002130   
## [106] train-aucpr:0.067120+0.000431 test-aucpr:0.063160+0.002160   
## [107] train-aucpr:0.067155+0.000433 test-aucpr:0.063173+0.002146   
## [108] train-aucpr:0.067201+0.000439 test-aucpr:0.063181+0.002135   
## [109] train-aucpr:0.067243+0.000445 test-aucpr:0.063193+0.002134   
## [110] train-aucpr:0.067293+0.000442 test-aucpr:0.063206+0.002140   
## [111] train-aucpr:0.067338+0.000430 test-aucpr:0.063232+0.002148   
## [112] train-aucpr:0.067381+0.000441 test-aucpr:0.063245+0.002143   
## [113] train-aucpr:0.067420+0.000433 test-aucpr:0.063257+0.002122   
## [114] train-aucpr:0.067451+0.000430 test-aucpr:0.063269+0.002114   
## [115] train-aucpr:0.067504+0.000440 test-aucpr:0.063292+0.002121   
## [116] train-aucpr:0.067566+0.000452 test-aucpr:0.063305+0.002114   
## [117] train-aucpr:0.067618+0.000462 test-aucpr:0.063362+0.002101   
## [118] train-aucpr:0.067647+0.000468 test-aucpr:0.063382+0.002092   
## [119] train-aucpr:0.067673+0.000481 test-aucpr:0.063370+0.002089   
## [120] train-aucpr:0.067713+0.000480 test-aucpr:0.063387+0.002089   
## [121] train-aucpr:0.067751+0.000494 test-aucpr:0.063403+0.002080   
## [122] train-aucpr:0.067788+0.000488 test-aucpr:0.063416+0.002055   
## [123] train-aucpr:0.067849+0.000470 test-aucpr:0.063420+0.002048   
## [124] train-aucpr:0.067881+0.000465 test-aucpr:0.063433+0.002060   
## [125] train-aucpr:0.067912+0.000469 test-aucpr:0.063448+0.002063   
## [126] train-aucpr:0.067970+0.000458 test-aucpr:0.063494+0.002073   
## [127] train-aucpr:0.067998+0.000455 test-aucpr:0.063499+0.002077   
## [128] train-aucpr:0.068030+0.000453 test-aucpr:0.063501+0.002063   
## [129] train-aucpr:0.068075+0.000478 test-aucpr:0.063521+0.002067   
## [130] train-aucpr:0.068110+0.000468 test-aucpr:0.063535+0.002071   
## [131] train-aucpr:0.068132+0.000466 test-aucpr:0.063549+0.002066   
## [132] train-aucpr:0.068182+0.000466 test-aucpr:0.063563+0.002049   
## [133] train-aucpr:0.068215+0.000468 test-aucpr:0.063570+0.002026   
## [134] train-aucpr:0.068244+0.000477 test-aucpr:0.063567+0.002035   
## [135] train-aucpr:0.068287+0.000481 test-aucpr:0.063568+0.002028   
## [136] train-aucpr:0.068341+0.000459 test-aucpr:0.063661+0.002165   
## [137] train-aucpr:0.068376+0.000464 test-aucpr:0.063677+0.002168   
## [138] train-aucpr:0.068412+0.000457 test-aucpr:0.063720+0.002162   
## [139] train-aucpr:0.068456+0.000442 test-aucpr:0.063740+0.002190   
## [140] train-aucpr:0.068490+0.000438 test-aucpr:0.063732+0.002188   
## [141] train-aucpr:0.068514+0.000423 test-aucpr:0.063740+0.002200   
## [142] train-aucpr:0.068533+0.000424 test-aucpr:0.063751+0.002198   
## [143] train-aucpr:0.068564+0.000430 test-aucpr:0.063759+0.002189   
## [144] train-aucpr:0.068596+0.000431 test-aucpr:0.063767+0.002183   
## [145] train-aucpr:0.068649+0.000418 test-aucpr:0.063802+0.002223   
## [146] train-aucpr:0.068683+0.000410 test-aucpr:0.063802+0.002201   
## [147] train-aucpr:0.068722+0.000412 test-aucpr:0.063844+0.002244   
## [148] train-aucpr:0.068784+0.000459 test-aucpr:0.063857+0.002243   
## [149] train-aucpr:0.068816+0.000456 test-aucpr:0.063868+0.002238   
## [150] train-aucpr:0.068873+0.000438 test-aucpr:0.063892+0.002254   
## [151] train-aucpr:0.068896+0.000426 test-aucpr:0.063917+0.002254   
## [152] train-aucpr:0.068915+0.000439 test-aucpr:0.063921+0.002247   
## [153] train-aucpr:0.068946+0.000435 test-aucpr:0.063926+0.002232   
## [154] train-aucpr:0.068985+0.000424 test-aucpr:0.063924+0.002228   
## [155] train-aucpr:0.069007+0.000419 test-aucpr:0.063928+0.002217   
## [156] train-aucpr:0.069046+0.000411 test-aucpr:0.063967+0.002249   
## [157] train-aucpr:0.069075+0.000410 test-aucpr:0.063963+0.002245   
## [158] train-aucpr:0.069116+0.000433 test-aucpr:0.063974+0.002239   
## [159] train-aucpr:0.069142+0.000426 test-aucpr:0.063971+0.002222   
## [160] train-aucpr:0.069194+0.000433 test-aucpr:0.063994+0.002203   
## [161] train-aucpr:0.069259+0.000427 test-aucpr:0.064014+0.002206   
## [162] train-aucpr:0.069326+0.000409 test-aucpr:0.064043+0.002202   
## [163] train-aucpr:0.069364+0.000421 test-aucpr:0.064054+0.002190   
## [164] train-aucpr:0.069402+0.000434 test-aucpr:0.064069+0.002177   
## [165] train-aucpr:0.069430+0.000436 test-aucpr:0.064071+0.002178   
## [166] train-aucpr:0.069452+0.000421 test-aucpr:0.064070+0.002177   
## [167] train-aucpr:0.069489+0.000422 test-aucpr:0.064078+0.002182   
## [168] train-aucpr:0.069521+0.000430 test-aucpr:0.064089+0.002176   
## [169] train-aucpr:0.069573+0.000442 test-aucpr:0.064109+0.002163   
## [170] train-aucpr:0.069606+0.000441 test-aucpr:0.064114+0.002154   
## [171] train-aucpr:0.069649+0.000456 test-aucpr:0.064145+0.002169   
## [172] train-aucpr:0.069705+0.000473 test-aucpr:0.064167+0.002189   
## [173] train-aucpr:0.069741+0.000464 test-aucpr:0.064167+0.002184   
## [174] train-aucpr:0.069783+0.000461 test-aucpr:0.064187+0.002172   
## [175] train-aucpr:0.069806+0.000464 test-aucpr:0.064195+0.002171   
## [176] train-aucpr:0.069858+0.000494 test-aucpr:0.064206+0.002144   
## [177] train-aucpr:0.069888+0.000489 test-aucpr:0.064198+0.002162   
## [178] train-aucpr:0.069943+0.000512 test-aucpr:0.064206+0.002147   
## [179] train-aucpr:0.069972+0.000518 test-aucpr:0.064229+0.002145   
## [180] train-aucpr:0.070011+0.000528 test-aucpr:0.064232+0.002139   
## [181] train-aucpr:0.070036+0.000530 test-aucpr:0.064234+0.002120   
## [182] train-aucpr:0.070091+0.000507 test-aucpr:0.064248+0.002146   
## [183] train-aucpr:0.070128+0.000513 test-aucpr:0.064269+0.002135   
## [184] train-aucpr:0.070195+0.000534 test-aucpr:0.064271+0.002116   
## [185] train-aucpr:0.070229+0.000529 test-aucpr:0.064308+0.002089   
## [186] train-aucpr:0.070286+0.000513 test-aucpr:0.064310+0.002092   
## [187] train-aucpr:0.070318+0.000517 test-aucpr:0.064320+0.002080   
## [188] train-aucpr:0.070346+0.000506 test-aucpr:0.064342+0.002086   
## [189] train-aucpr:0.070380+0.000513 test-aucpr:0.064346+0.002076   
## [190] train-aucpr:0.070411+0.000519 test-aucpr:0.064344+0.002074   
## [191] train-aucpr:0.070436+0.000518 test-aucpr:0.064339+0.002080   
## [192] train-aucpr:0.070487+0.000524 test-aucpr:0.064365+0.002089   
## [193] train-aucpr:0.070548+0.000515 test-aucpr:0.064360+0.002081   
## [194] train-aucpr:0.070594+0.000518 test-aucpr:0.064361+0.002082   
## [195] train-aucpr:0.070621+0.000521 test-aucpr:0.064357+0.002079   
## [196] train-aucpr:0.070652+0.000526 test-aucpr:0.064373+0.002092   
## [197] train-aucpr:0.070708+0.000496 test-aucpr:0.064389+0.002089   
## [198] train-aucpr:0.070740+0.000501 test-aucpr:0.064381+0.002077   
## [199] train-aucpr:0.070760+0.000505 test-aucpr:0.064381+0.002066   
## [200] train-aucpr:0.070788+0.000498 test-aucpr:0.064406+0.002052   
## [201] train-aucpr:0.070837+0.000508 test-aucpr:0.064408+0.002053   
## [202] train-aucpr:0.070897+0.000512 test-aucpr:0.064403+0.002058   
## [203] train-aucpr:0.070936+0.000498 test-aucpr:0.064401+0.002051   
## [204] train-aucpr:0.070960+0.000502 test-aucpr:0.064412+0.002069   
## [205] train-aucpr:0.071022+0.000500 test-aucpr:0.064424+0.002037   
## [206] train-aucpr:0.071055+0.000495 test-aucpr:0.064438+0.002046   
## [207] train-aucpr:0.071084+0.000487 test-aucpr:0.064445+0.002038   
## [208] train-aucpr:0.071111+0.000483 test-aucpr:0.064459+0.002020   
## [209] train-aucpr:0.071136+0.000485 test-aucpr:0.064455+0.002017   
## [210] train-aucpr:0.071172+0.000508 test-aucpr:0.064462+0.002021   
## [211] train-aucpr:0.071208+0.000533 test-aucpr:0.064467+0.002020   
## [212] train-aucpr:0.071240+0.000528 test-aucpr:0.064466+0.002019   
## [213] train-aucpr:0.071274+0.000532 test-aucpr:0.064464+0.002015   
## [214] train-aucpr:0.071295+0.000536 test-aucpr:0.064463+0.002009   
## [215] train-aucpr:0.071326+0.000517 test-aucpr:0.064453+0.002008   
## [216] train-aucpr:0.071350+0.000512 test-aucpr:0.064451+0.002012   
## [217] train-aucpr:0.071372+0.000514 test-aucpr:0.064480+0.001992   
## [218] train-aucpr:0.071410+0.000509 test-aucpr:0.064459+0.001991   
## [219] train-aucpr:0.071467+0.000526 test-aucpr:0.064475+0.001992   
## [220] train-aucpr:0.071502+0.000519 test-aucpr:0.064519+0.001984   
## [221] train-aucpr:0.071529+0.000515 test-aucpr:0.064521+0.001990   
## [222] train-aucpr:0.071554+0.000511 test-aucpr:0.064523+0.001979   
## [223] train-aucpr:0.071579+0.000513 test-aucpr:0.064517+0.001985   
## [224] train-aucpr:0.071605+0.000509 test-aucpr:0.064520+0.001989   
## [225] train-aucpr:0.071666+0.000469 test-aucpr:0.064530+0.001987   
## [226] train-aucpr:0.071692+0.000469 test-aucpr:0.064516+0.001992   
## [227] train-aucpr:0.071720+0.000468 test-aucpr:0.064540+0.001992   
## [228] train-aucpr:0.071750+0.000468 test-aucpr:0.064529+0.002001   
## [229] train-aucpr:0.071775+0.000479 test-aucpr:0.064527+0.002000   
## [230] train-aucpr:0.071815+0.000473 test-aucpr:0.064545+0.001987   
## [231] train-aucpr:0.071854+0.000478 test-aucpr:0.064541+0.001991   
## [232] train-aucpr:0.071885+0.000479 test-aucpr:0.064551+0.001977   
## [233] train-aucpr:0.071929+0.000468 test-aucpr:0.064560+0.001978   
## [234] train-aucpr:0.071980+0.000478 test-aucpr:0.064571+0.001967   
## [235] train-aucpr:0.071988+0.000493 test-aucpr:0.064567+0.001965   
## [236] train-aucpr:0.072020+0.000494 test-aucpr:0.064574+0.001972   
## [237] train-aucpr:0.072049+0.000490 test-aucpr:0.064580+0.001962   
## [238] train-aucpr:0.072082+0.000472 test-aucpr:0.064591+0.001951   
## [239] train-aucpr:0.072101+0.000477 test-aucpr:0.064601+0.001954   
## [240] train-aucpr:0.072131+0.000467 test-aucpr:0.064599+0.001959   
## [241] train-aucpr:0.072169+0.000465 test-aucpr:0.064574+0.001926   
## [242] train-aucpr:0.072204+0.000469 test-aucpr:0.064583+0.001922   
## [243] train-aucpr:0.072231+0.000478 test-aucpr:0.064584+0.001926   
## [244] train-aucpr:0.072264+0.000473 test-aucpr:0.064588+0.001895   
## [245] train-aucpr:0.072303+0.000496 test-aucpr:0.064588+0.001898   
## [246] train-aucpr:0.072350+0.000461 test-aucpr:0.064571+0.001869   
## [247] train-aucpr:0.072378+0.000464 test-aucpr:0.064578+0.001867   
## [248] train-aucpr:0.072418+0.000456 test-aucpr:0.064599+0.001854   
## [249] train-aucpr:0.072439+0.000456 test-aucpr:0.064584+0.001869   
## Stopping. Best iteration:  
## [239] train-aucpr:0.072101+0.000477 test-aucpr:0.064601+0.001954

sel\_rounds6 = xgb.data.cv6$best\_iteration  
  
AUC6 <- xgb.data.cv6$evaluation\_log[sel\_rounds6]  
AUC6

## iter train\_aucpr\_mean train\_aucpr\_std test\_aucpr\_mean test\_aucpr\_std  
## 1: 239 0.0721008 0.0004772678 0.064601 0.001954284

best\_xgb <- ifelse(AUC6 > best\_auc, "AUC6", best\_xgb)  
best\_auc <- ifelse(AUC6 > best\_auc, AUC6, best\_auc)  
  
#better results with smaller trees! Now try trees with even less depth  
  
  
xgb.data.cv7 = xgb.cv(data = pred\_tr, label = resp\_tr,   
 max.depth = 2, eta = .05, nrounds = 400, min\_child\_weight = 1,  
 nfold = 5, early\_stopping\_rounds = 10, scale\_pos\_weight = neg.rate,  
 objective = "binary:logistic", eval\_metric = "aucpr")

## [1] train-aucpr:0.048016+0.000205 test-aucpr:0.048003+0.000818   
## Multiple eval metrics are present. Will use test\_aucpr for early stopping.  
## Will train until test\_aucpr hasn't improved in 10 rounds.  
##   
## [2] train-aucpr:0.048680+0.001009 test-aucpr:0.048123+0.000797   
## [3] train-aucpr:0.048870+0.001025 test-aucpr:0.048367+0.000679   
## [4] train-aucpr:0.049584+0.000671 test-aucpr:0.048908+0.001345   
## [5] train-aucpr:0.051052+0.000342 test-aucpr:0.050440+0.001564   
## [6] train-aucpr:0.051791+0.000596 test-aucpr:0.051184+0.002128   
## [7] train-aucpr:0.052576+0.000410 test-aucpr:0.051769+0.002024   
## [8] train-aucpr:0.053246+0.000278 test-aucpr:0.052503+0.002007   
## [9] train-aucpr:0.053700+0.000403 test-aucpr:0.052780+0.002463   
## [10] train-aucpr:0.054213+0.000318 test-aucpr:0.053344+0.002405   
## [11] train-aucpr:0.054595+0.000402 test-aucpr:0.053730+0.002292   
## [12] train-aucpr:0.054893+0.000127 test-aucpr:0.054031+0.002192   
## [13] train-aucpr:0.055242+0.000395 test-aucpr:0.054314+0.002254   
## [14] train-aucpr:0.055504+0.000247 test-aucpr:0.054698+0.002486   
## [15] train-aucpr:0.055961+0.000275 test-aucpr:0.055003+0.002304   
## [16] train-aucpr:0.056099+0.000247 test-aucpr:0.055115+0.002160   
## [17] train-aucpr:0.056443+0.000375 test-aucpr:0.055555+0.002132   
## [18] train-aucpr:0.056645+0.000317 test-aucpr:0.055700+0.002227   
## [19] train-aucpr:0.056901+0.000430 test-aucpr:0.055900+0.002145   
## [20] train-aucpr:0.056996+0.000444 test-aucpr:0.056014+0.002099   
## [21] train-aucpr:0.057355+0.000398 test-aucpr:0.056341+0.002099   
## [22] train-aucpr:0.057733+0.000438 test-aucpr:0.056603+0.002021   
## [23] train-aucpr:0.057864+0.000404 test-aucpr:0.056735+0.002018   
## [24] train-aucpr:0.058012+0.000268 test-aucpr:0.056889+0.002019   
## [25] train-aucpr:0.058117+0.000343 test-aucpr:0.057016+0.001941   
## [26] train-aucpr:0.058407+0.000276 test-aucpr:0.057256+0.002084   
## [27] train-aucpr:0.058530+0.000271 test-aucpr:0.057404+0.002125   
## [28] train-aucpr:0.058731+0.000335 test-aucpr:0.057615+0.002021   
## [29] train-aucpr:0.058880+0.000322 test-aucpr:0.057707+0.001953   
## [30] train-aucpr:0.059045+0.000270 test-aucpr:0.057872+0.002013   
## [31] train-aucpr:0.059239+0.000262 test-aucpr:0.057997+0.001995   
## [32] train-aucpr:0.059441+0.000254 test-aucpr:0.058221+0.001936   
## [33] train-aucpr:0.059502+0.000264 test-aucpr:0.058242+0.001893   
## [34] train-aucpr:0.059607+0.000278 test-aucpr:0.058330+0.001867   
## [35] train-aucpr:0.059660+0.000276 test-aucpr:0.058356+0.001851   
## [36] train-aucpr:0.059778+0.000341 test-aucpr:0.058447+0.001805   
## [37] train-aucpr:0.059947+0.000310 test-aucpr:0.058631+0.001793   
## [38] train-aucpr:0.059994+0.000317 test-aucpr:0.058689+0.001794   
## [39] train-aucpr:0.060067+0.000277 test-aucpr:0.058794+0.001832   
## [40] train-aucpr:0.060149+0.000312 test-aucpr:0.058870+0.001790   
## [41] train-aucpr:0.060235+0.000318 test-aucpr:0.058953+0.001793   
## [42] train-aucpr:0.060354+0.000296 test-aucpr:0.059079+0.001801   
## [43] train-aucpr:0.060483+0.000286 test-aucpr:0.059282+0.001811   
## [44] train-aucpr:0.060549+0.000282 test-aucpr:0.059431+0.001766   
## [45] train-aucpr:0.060640+0.000347 test-aucpr:0.059523+0.001732   
## [46] train-aucpr:0.060703+0.000320 test-aucpr:0.059586+0.001702   
## [47] train-aucpr:0.060773+0.000334 test-aucpr:0.059643+0.001710   
## [48] train-aucpr:0.060810+0.000333 test-aucpr:0.059591+0.001743   
## [49] train-aucpr:0.060913+0.000308 test-aucpr:0.059650+0.001767   
## [50] train-aucpr:0.060990+0.000278 test-aucpr:0.059744+0.001761   
## [51] train-aucpr:0.061069+0.000279 test-aucpr:0.059911+0.001699   
## [52] train-aucpr:0.061145+0.000260 test-aucpr:0.059971+0.001731   
## [53] train-aucpr:0.061195+0.000265 test-aucpr:0.060021+0.001713   
## [54] train-aucpr:0.061260+0.000195 test-aucpr:0.060020+0.001794   
## [55] train-aucpr:0.061337+0.000231 test-aucpr:0.060060+0.001764   
## [56] train-aucpr:0.061360+0.000245 test-aucpr:0.060096+0.001773   
## [57] train-aucpr:0.061417+0.000251 test-aucpr:0.060198+0.001738   
## [58] train-aucpr:0.061460+0.000251 test-aucpr:0.060176+0.001784   
## [59] train-aucpr:0.061516+0.000247 test-aucpr:0.060264+0.001822   
## [60] train-aucpr:0.061619+0.000291 test-aucpr:0.060434+0.001776   
## [61] train-aucpr:0.061631+0.000273 test-aucpr:0.060437+0.001790   
## [62] train-aucpr:0.061688+0.000261 test-aucpr:0.060483+0.001804   
## [63] train-aucpr:0.061729+0.000298 test-aucpr:0.060487+0.001751   
## [64] train-aucpr:0.061766+0.000286 test-aucpr:0.060543+0.001771   
## [65] train-aucpr:0.061803+0.000287 test-aucpr:0.060592+0.001751   
## [66] train-aucpr:0.061849+0.000275 test-aucpr:0.060592+0.001735   
## [67] train-aucpr:0.061918+0.000241 test-aucpr:0.060643+0.001754   
## [68] train-aucpr:0.061957+0.000249 test-aucpr:0.060676+0.001760   
## [69] train-aucpr:0.062001+0.000249 test-aucpr:0.060763+0.001761   
## [70] train-aucpr:0.062058+0.000244 test-aucpr:0.060825+0.001766   
## [71] train-aucpr:0.062122+0.000225 test-aucpr:0.060856+0.001757   
## [72] train-aucpr:0.062146+0.000238 test-aucpr:0.060867+0.001770   
## [73] train-aucpr:0.062185+0.000248 test-aucpr:0.060899+0.001745   
## [74] train-aucpr:0.062213+0.000220 test-aucpr:0.060907+0.001759   
## [75] train-aucpr:0.062247+0.000241 test-aucpr:0.060947+0.001758   
## [76] train-aucpr:0.062308+0.000241 test-aucpr:0.060978+0.001777   
## [77] train-aucpr:0.062350+0.000229 test-aucpr:0.061007+0.001771   
## [78] train-aucpr:0.062380+0.000242 test-aucpr:0.061008+0.001761   
## [79] train-aucpr:0.062441+0.000230 test-aucpr:0.060986+0.001805   
## [80] train-aucpr:0.062476+0.000246 test-aucpr:0.061085+0.001750   
## [81] train-aucpr:0.062507+0.000230 test-aucpr:0.061127+0.001757   
## [82] train-aucpr:0.062539+0.000235 test-aucpr:0.061145+0.001737   
## [83] train-aucpr:0.062562+0.000230 test-aucpr:0.061136+0.001730   
## [84] train-aucpr:0.062612+0.000211 test-aucpr:0.061099+0.001817   
## [85] train-aucpr:0.062655+0.000240 test-aucpr:0.061136+0.001795   
## [86] train-aucpr:0.062698+0.000257 test-aucpr:0.061199+0.001760   
## [87] train-aucpr:0.062720+0.000259 test-aucpr:0.061220+0.001775   
## [88] train-aucpr:0.062758+0.000267 test-aucpr:0.061215+0.001776   
## [89] train-aucpr:0.062803+0.000236 test-aucpr:0.061271+0.001786   
## [90] train-aucpr:0.062818+0.000235 test-aucpr:0.061307+0.001805   
## [91] train-aucpr:0.062859+0.000262 test-aucpr:0.061332+0.001791   
## [92] train-aucpr:0.062886+0.000268 test-aucpr:0.061354+0.001799   
## [93] train-aucpr:0.062912+0.000257 test-aucpr:0.061373+0.001823   
## [94] train-aucpr:0.062955+0.000253 test-aucpr:0.061385+0.001827   
## [95] train-aucpr:0.062988+0.000253 test-aucpr:0.061439+0.001840   
## [96] train-aucpr:0.063014+0.000243 test-aucpr:0.061471+0.001857   
## [97] train-aucpr:0.063070+0.000223 test-aucpr:0.061499+0.001881   
## [98] train-aucpr:0.063109+0.000234 test-aucpr:0.061540+0.001883   
## [99] train-aucpr:0.063137+0.000249 test-aucpr:0.061579+0.001874   
## [100] train-aucpr:0.063164+0.000243 test-aucpr:0.061594+0.001871   
## [101] train-aucpr:0.063197+0.000246 test-aucpr:0.061618+0.001876   
## [102] train-aucpr:0.063216+0.000256 test-aucpr:0.061644+0.001843   
## [103] train-aucpr:0.063249+0.000267 test-aucpr:0.061657+0.001849   
## [104] train-aucpr:0.063283+0.000269 test-aucpr:0.061674+0.001856   
## [105] train-aucpr:0.063324+0.000285 test-aucpr:0.061700+0.001839   
## [106] train-aucpr:0.063351+0.000281 test-aucpr:0.061725+0.001830   
## [107] train-aucpr:0.063390+0.000261 test-aucpr:0.061750+0.001831   
## [108] train-aucpr:0.063407+0.000270 test-aucpr:0.061814+0.001879   
## [109] train-aucpr:0.063432+0.000280 test-aucpr:0.061861+0.001863   
## [110] train-aucpr:0.063460+0.000266 test-aucpr:0.061887+0.001875   
## [111] train-aucpr:0.063488+0.000277 test-aucpr:0.061910+0.001864   
## [112] train-aucpr:0.063500+0.000283 test-aucpr:0.061932+0.001861   
## [113] train-aucpr:0.063535+0.000279 test-aucpr:0.061956+0.001862   
## [114] train-aucpr:0.063553+0.000268 test-aucpr:0.061960+0.001874   
## [115] train-aucpr:0.063573+0.000272 test-aucpr:0.061989+0.001884   
## [116] train-aucpr:0.063607+0.000267 test-aucpr:0.062010+0.001884   
## [117] train-aucpr:0.063632+0.000273 test-aucpr:0.062016+0.001873   
## [118] train-aucpr:0.063647+0.000275 test-aucpr:0.062039+0.001867   
## [119] train-aucpr:0.063674+0.000283 test-aucpr:0.062061+0.001868   
## [120] train-aucpr:0.063703+0.000277 test-aucpr:0.062037+0.001819   
## [121] train-aucpr:0.063729+0.000281 test-aucpr:0.062048+0.001825   
## [122] train-aucpr:0.063752+0.000283 test-aucpr:0.062064+0.001833   
## [123] train-aucpr:0.063781+0.000268 test-aucpr:0.062091+0.001846   
## [124] train-aucpr:0.063797+0.000262 test-aucpr:0.062131+0.001930   
## [125] train-aucpr:0.063816+0.000271 test-aucpr:0.062143+0.001927   
## [126] train-aucpr:0.063840+0.000273 test-aucpr:0.062155+0.001925   
## [127] train-aucpr:0.063871+0.000273 test-aucpr:0.062162+0.001920   
## [128] train-aucpr:0.063879+0.000272 test-aucpr:0.062172+0.001917   
## [129] train-aucpr:0.063912+0.000265 test-aucpr:0.062199+0.001924   
## [130] train-aucpr:0.063937+0.000253 test-aucpr:0.062203+0.001932   
## [131] train-aucpr:0.063969+0.000255 test-aucpr:0.062220+0.001931   
## [132] train-aucpr:0.063987+0.000258 test-aucpr:0.062241+0.001921   
## [133] train-aucpr:0.063997+0.000260 test-aucpr:0.062251+0.001922   
## [134] train-aucpr:0.064014+0.000259 test-aucpr:0.062255+0.001923   
## [135] train-aucpr:0.064033+0.000262 test-aucpr:0.062257+0.001913   
## [136] train-aucpr:0.064060+0.000255 test-aucpr:0.062263+0.001921   
## [137] train-aucpr:0.064092+0.000257 test-aucpr:0.062284+0.001907   
## [138] train-aucpr:0.064108+0.000266 test-aucpr:0.062293+0.001908   
## [139] train-aucpr:0.064131+0.000271 test-aucpr:0.062299+0.001887   
## [140] train-aucpr:0.064149+0.000268 test-aucpr:0.062316+0.001896   
## [141] train-aucpr:0.064173+0.000270 test-aucpr:0.062326+0.001906   
## [142] train-aucpr:0.064191+0.000269 test-aucpr:0.062349+0.001902   
## [143] train-aucpr:0.064207+0.000263 test-aucpr:0.062373+0.001901   
## [144] train-aucpr:0.064228+0.000279 test-aucpr:0.062380+0.001887   
## [145] train-aucpr:0.064239+0.000275 test-aucpr:0.062369+0.001900   
## [146] train-aucpr:0.064267+0.000265 test-aucpr:0.062400+0.001918   
## [147] train-aucpr:0.064289+0.000277 test-aucpr:0.062428+0.001893   
## [148] train-aucpr:0.064295+0.000278 test-aucpr:0.062415+0.001899   
## [149] train-aucpr:0.064309+0.000274 test-aucpr:0.062426+0.001891   
## [150] train-aucpr:0.064320+0.000272 test-aucpr:0.062433+0.001904   
## [151] train-aucpr:0.064346+0.000267 test-aucpr:0.062435+0.001906   
## [152] train-aucpr:0.064364+0.000265 test-aucpr:0.062452+0.001917   
## [153] train-aucpr:0.064394+0.000270 test-aucpr:0.062471+0.001897   
## [154] train-aucpr:0.064409+0.000271 test-aucpr:0.062481+0.001899   
## [155] train-aucpr:0.064438+0.000249 test-aucpr:0.062508+0.001914   
## [156] train-aucpr:0.064447+0.000250 test-aucpr:0.062515+0.001908   
## [157] train-aucpr:0.064470+0.000243 test-aucpr:0.062520+0.001897   
## [158] train-aucpr:0.064491+0.000250 test-aucpr:0.062517+0.001892   
## [159] train-aucpr:0.064507+0.000248 test-aucpr:0.062538+0.001897   
## [160] train-aucpr:0.064514+0.000255 test-aucpr:0.062555+0.001906   
## [161] train-aucpr:0.064531+0.000249 test-aucpr:0.062571+0.001929   
## [162] train-aucpr:0.064546+0.000253 test-aucpr:0.062590+0.001907   
## [163] train-aucpr:0.064569+0.000239 test-aucpr:0.062598+0.001917   
## [164] train-aucpr:0.064598+0.000247 test-aucpr:0.062596+0.001909   
## [165] train-aucpr:0.064613+0.000244 test-aucpr:0.062598+0.001901   
## [166] train-aucpr:0.064629+0.000244 test-aucpr:0.062607+0.001912   
## [167] train-aucpr:0.064653+0.000260 test-aucpr:0.062618+0.001896   
## [168] train-aucpr:0.064677+0.000255 test-aucpr:0.062629+0.001889   
## [169] train-aucpr:0.064694+0.000254 test-aucpr:0.062656+0.001910   
## [170] train-aucpr:0.064707+0.000256 test-aucpr:0.062656+0.001916   
## [171] train-aucpr:0.064728+0.000257 test-aucpr:0.062676+0.001912   
## [172] train-aucpr:0.064758+0.000233 test-aucpr:0.062706+0.001937   
## [173] train-aucpr:0.064771+0.000239 test-aucpr:0.062719+0.001935   
## [174] train-aucpr:0.064786+0.000234 test-aucpr:0.062723+0.001927   
## [175] train-aucpr:0.064801+0.000240 test-aucpr:0.062717+0.001921   
## [176] train-aucpr:0.064822+0.000229 test-aucpr:0.062714+0.001916   
## [177] train-aucpr:0.064841+0.000240 test-aucpr:0.062723+0.001919   
## [178] train-aucpr:0.064859+0.000244 test-aucpr:0.062722+0.001908   
## [179] train-aucpr:0.064887+0.000237 test-aucpr:0.062741+0.001917   
## [180] train-aucpr:0.064895+0.000232 test-aucpr:0.062749+0.001919   
## [181] train-aucpr:0.064907+0.000233 test-aucpr:0.062773+0.001934   
## [182] train-aucpr:0.064922+0.000233 test-aucpr:0.062782+0.001931   
## [183] train-aucpr:0.064939+0.000233 test-aucpr:0.062786+0.001932   
## [184] train-aucpr:0.064957+0.000221 test-aucpr:0.062791+0.001930   
## [185] train-aucpr:0.064979+0.000221 test-aucpr:0.062803+0.001945   
## [186] train-aucpr:0.064998+0.000216 test-aucpr:0.062822+0.001965   
## [187] train-aucpr:0.065031+0.000199 test-aucpr:0.062813+0.001908   
## [188] train-aucpr:0.065041+0.000201 test-aucpr:0.062812+0.001901   
## [189] train-aucpr:0.065071+0.000198 test-aucpr:0.062826+0.001905   
## [190] train-aucpr:0.065078+0.000198 test-aucpr:0.062828+0.001902   
## [191] train-aucpr:0.065095+0.000204 test-aucpr:0.062825+0.001893   
## [192] train-aucpr:0.065115+0.000221 test-aucpr:0.062848+0.001882   
## [193] train-aucpr:0.065136+0.000223 test-aucpr:0.062850+0.001889   
## [194] train-aucpr:0.065152+0.000216 test-aucpr:0.062865+0.001904   
## [195] train-aucpr:0.065185+0.000211 test-aucpr:0.062884+0.001894   
## [196] train-aucpr:0.065210+0.000204 test-aucpr:0.062895+0.001870   
## [197] train-aucpr:0.065228+0.000203 test-aucpr:0.062915+0.001881   
## [198] train-aucpr:0.065241+0.000209 test-aucpr:0.062914+0.001869   
## [199] train-aucpr:0.065259+0.000222 test-aucpr:0.062915+0.001868   
## [200] train-aucpr:0.065280+0.000212 test-aucpr:0.062926+0.001870   
## [201] train-aucpr:0.065307+0.000196 test-aucpr:0.062937+0.001869   
## [202] train-aucpr:0.065325+0.000198 test-aucpr:0.062944+0.001858   
## [203] train-aucpr:0.065337+0.000203 test-aucpr:0.062950+0.001856   
## [204] train-aucpr:0.065362+0.000199 test-aucpr:0.062969+0.001863   
## [205] train-aucpr:0.065384+0.000198 test-aucpr:0.062986+0.001863   
## [206] train-aucpr:0.065404+0.000199 test-aucpr:0.062987+0.001866   
## [207] train-aucpr:0.065419+0.000199 test-aucpr:0.062996+0.001861   
## [208] train-aucpr:0.065445+0.000220 test-aucpr:0.063007+0.001858   
## [209] train-aucpr:0.065464+0.000218 test-aucpr:0.063025+0.001862   
## [210] train-aucpr:0.065491+0.000202 test-aucpr:0.063029+0.001853   
## [211] train-aucpr:0.065496+0.000199 test-aucpr:0.063021+0.001836   
## [212] train-aucpr:0.065520+0.000193 test-aucpr:0.063027+0.001834   
## [213] train-aucpr:0.065535+0.000186 test-aucpr:0.063042+0.001837   
## [214] train-aucpr:0.065563+0.000195 test-aucpr:0.063065+0.001820   
## [215] train-aucpr:0.065573+0.000198 test-aucpr:0.063074+0.001831   
## [216] train-aucpr:0.065589+0.000201 test-aucpr:0.063078+0.001830   
## [217] train-aucpr:0.065621+0.000210 test-aucpr:0.063086+0.001825   
## [218] train-aucpr:0.065634+0.000211 test-aucpr:0.063083+0.001818   
## [219] train-aucpr:0.065646+0.000208 test-aucpr:0.063082+0.001816   
## [220] train-aucpr:0.065673+0.000215 test-aucpr:0.063099+0.001811   
## [221] train-aucpr:0.065692+0.000199 test-aucpr:0.063100+0.001809   
## [222] train-aucpr:0.065700+0.000196 test-aucpr:0.063104+0.001803   
## [223] train-aucpr:0.065715+0.000199 test-aucpr:0.063117+0.001792   
## [224] train-aucpr:0.065737+0.000220 test-aucpr:0.063133+0.001791   
## [225] train-aucpr:0.065751+0.000214 test-aucpr:0.063140+0.001789   
## [226] train-aucpr:0.065764+0.000217 test-aucpr:0.063143+0.001781   
## [227] train-aucpr:0.065773+0.000222 test-aucpr:0.063139+0.001782   
## [228] train-aucpr:0.065784+0.000217 test-aucpr:0.063153+0.001797   
## [229] train-aucpr:0.065805+0.000213 test-aucpr:0.063172+0.001786   
## [230] train-aucpr:0.065810+0.000223 test-aucpr:0.063173+0.001785   
## [231] train-aucpr:0.065822+0.000224 test-aucpr:0.063172+0.001787   
## [232] train-aucpr:0.065845+0.000228 test-aucpr:0.063179+0.001783   
## [233] train-aucpr:0.065852+0.000231 test-aucpr:0.063187+0.001789   
## [234] train-aucpr:0.065878+0.000215 test-aucpr:0.063189+0.001786   
## [235] train-aucpr:0.065908+0.000217 test-aucpr:0.063204+0.001800   
## [236] train-aucpr:0.065914+0.000216 test-aucpr:0.063200+0.001798   
## [237] train-aucpr:0.065928+0.000205 test-aucpr:0.063207+0.001796   
## [238] train-aucpr:0.065938+0.000206 test-aucpr:0.063211+0.001795   
## [239] train-aucpr:0.065954+0.000203 test-aucpr:0.063212+0.001796   
## [240] train-aucpr:0.065970+0.000192 test-aucpr:0.063213+0.001804   
## [241] train-aucpr:0.065994+0.000205 test-aucpr:0.063232+0.001821   
## [242] train-aucpr:0.066000+0.000207 test-aucpr:0.063239+0.001818   
## [243] train-aucpr:0.066014+0.000207 test-aucpr:0.063243+0.001810   
## [244] train-aucpr:0.066046+0.000207 test-aucpr:0.063253+0.001798   
## [245] train-aucpr:0.066073+0.000200 test-aucpr:0.063245+0.001787   
## [246] train-aucpr:0.066094+0.000210 test-aucpr:0.063252+0.001777   
## [247] train-aucpr:0.066114+0.000226 test-aucpr:0.063254+0.001776   
## [248] train-aucpr:0.066125+0.000221 test-aucpr:0.063256+0.001776   
## [249] train-aucpr:0.066139+0.000206 test-aucpr:0.063262+0.001779   
## [250] train-aucpr:0.066156+0.000210 test-aucpr:0.063290+0.001778   
## [251] train-aucpr:0.066170+0.000213 test-aucpr:0.063293+0.001781   
## [252] train-aucpr:0.066185+0.000201 test-aucpr:0.063345+0.001815   
## [253] train-aucpr:0.066197+0.000197 test-aucpr:0.063348+0.001820   
## [254] train-aucpr:0.066209+0.000199 test-aucpr:0.063357+0.001815   
## [255] train-aucpr:0.066236+0.000214 test-aucpr:0.063386+0.001792   
## [256] train-aucpr:0.066243+0.000216 test-aucpr:0.063394+0.001803   
## [257] train-aucpr:0.066275+0.000198 test-aucpr:0.063383+0.001790   
## [258] train-aucpr:0.066288+0.000192 test-aucpr:0.063383+0.001795   
## [259] train-aucpr:0.066293+0.000201 test-aucpr:0.063380+0.001784   
## [260] train-aucpr:0.066319+0.000193 test-aucpr:0.063378+0.001772   
## [261] train-aucpr:0.066334+0.000187 test-aucpr:0.063383+0.001770   
## [262] train-aucpr:0.066361+0.000198 test-aucpr:0.063401+0.001785   
## [263] train-aucpr:0.066379+0.000213 test-aucpr:0.063408+0.001777   
## [264] train-aucpr:0.066381+0.000211 test-aucpr:0.063406+0.001769   
## [265] train-aucpr:0.066403+0.000195 test-aucpr:0.063416+0.001766   
## [266] train-aucpr:0.066411+0.000197 test-aucpr:0.063415+0.001760   
## [267] train-aucpr:0.066424+0.000198 test-aucpr:0.063423+0.001761   
## [268] train-aucpr:0.066440+0.000196 test-aucpr:0.063428+0.001743   
## [269] train-aucpr:0.066456+0.000195 test-aucpr:0.063446+0.001756   
## [270] train-aucpr:0.066463+0.000201 test-aucpr:0.063441+0.001763   
## [271] train-aucpr:0.066476+0.000201 test-aucpr:0.063446+0.001758   
## [272] train-aucpr:0.066485+0.000216 test-aucpr:0.063467+0.001749   
## [273] train-aucpr:0.066518+0.000195 test-aucpr:0.063467+0.001740   
## [274] train-aucpr:0.066538+0.000202 test-aucpr:0.063483+0.001738   
## [275] train-aucpr:0.066546+0.000199 test-aucpr:0.063489+0.001727   
## [276] train-aucpr:0.066559+0.000195 test-aucpr:0.063490+0.001732   
## [277] train-aucpr:0.066584+0.000182 test-aucpr:0.063511+0.001759   
## [278] train-aucpr:0.066593+0.000186 test-aucpr:0.063516+0.001753   
## [279] train-aucpr:0.066598+0.000184 test-aucpr:0.063514+0.001746   
## [280] train-aucpr:0.066617+0.000196 test-aucpr:0.063518+0.001748   
## [281] train-aucpr:0.066634+0.000200 test-aucpr:0.063521+0.001744   
## [282] train-aucpr:0.066662+0.000194 test-aucpr:0.063544+0.001762   
## [283] train-aucpr:0.066673+0.000189 test-aucpr:0.063559+0.001766   
## [284] train-aucpr:0.066683+0.000189 test-aucpr:0.063569+0.001772   
## [285] train-aucpr:0.066695+0.000192 test-aucpr:0.063561+0.001777   
## [286] train-aucpr:0.066705+0.000191 test-aucpr:0.063563+0.001779   
## [287] train-aucpr:0.066713+0.000191 test-aucpr:0.063566+0.001782   
## [288] train-aucpr:0.066722+0.000199 test-aucpr:0.063548+0.001760   
## [289] train-aucpr:0.066745+0.000199 test-aucpr:0.063571+0.001752   
## [290] train-aucpr:0.066769+0.000191 test-aucpr:0.063591+0.001749   
## [291] train-aucpr:0.066788+0.000205 test-aucpr:0.063598+0.001741   
## [292] train-aucpr:0.066800+0.000204 test-aucpr:0.063600+0.001740   
## [293] train-aucpr:0.066815+0.000207 test-aucpr:0.063614+0.001752   
## [294] train-aucpr:0.066832+0.000195 test-aucpr:0.063622+0.001781   
## [295] train-aucpr:0.066845+0.000194 test-aucpr:0.063615+0.001784   
## [296] train-aucpr:0.066867+0.000182 test-aucpr:0.063620+0.001788   
## [297] train-aucpr:0.066895+0.000184 test-aucpr:0.063612+0.001775   
## [298] train-aucpr:0.066911+0.000181 test-aucpr:0.063624+0.001771   
## [299] train-aucpr:0.066921+0.000180 test-aucpr:0.063630+0.001771   
## [300] train-aucpr:0.066928+0.000181 test-aucpr:0.063629+0.001760   
## [301] train-aucpr:0.066941+0.000180 test-aucpr:0.063627+0.001759   
## [302] train-aucpr:0.066951+0.000183 test-aucpr:0.063625+0.001762   
## [303] train-aucpr:0.066956+0.000183 test-aucpr:0.063621+0.001759   
## [304] train-aucpr:0.066970+0.000180 test-aucpr:0.063622+0.001759   
## [305] train-aucpr:0.066980+0.000182 test-aucpr:0.063623+0.001755   
## [306] train-aucpr:0.066982+0.000184 test-aucpr:0.063639+0.001739   
## [307] train-aucpr:0.066997+0.000187 test-aucpr:0.063657+0.001737   
## [308] train-aucpr:0.067003+0.000188 test-aucpr:0.063658+0.001738   
## [309] train-aucpr:0.067036+0.000210 test-aucpr:0.063665+0.001743   
## [310] train-aucpr:0.067048+0.000216 test-aucpr:0.063668+0.001738   
## [311] train-aucpr:0.067061+0.000220 test-aucpr:0.063681+0.001734   
## [312] train-aucpr:0.067079+0.000219 test-aucpr:0.063679+0.001719   
## [313] train-aucpr:0.067101+0.000220 test-aucpr:0.063677+0.001720   
## [314] train-aucpr:0.067126+0.000228 test-aucpr:0.063682+0.001719   
## [315] train-aucpr:0.067131+0.000225 test-aucpr:0.063683+0.001721   
## [316] train-aucpr:0.067160+0.000227 test-aucpr:0.063700+0.001713   
## [317] train-aucpr:0.067183+0.000218 test-aucpr:0.063700+0.001714   
## [318] train-aucpr:0.067208+0.000204 test-aucpr:0.063713+0.001724   
## [319] train-aucpr:0.067215+0.000207 test-aucpr:0.063719+0.001729   
## [320] train-aucpr:0.067219+0.000205 test-aucpr:0.063721+0.001731   
## [321] train-aucpr:0.067242+0.000208 test-aucpr:0.063735+0.001722   
## [322] train-aucpr:0.067250+0.000208 test-aucpr:0.063729+0.001712   
## [323] train-aucpr:0.067255+0.000203 test-aucpr:0.063729+0.001710   
## [324] train-aucpr:0.067273+0.000207 test-aucpr:0.063733+0.001714   
## [325] train-aucpr:0.067291+0.000197 test-aucpr:0.063741+0.001711   
## [326] train-aucpr:0.067302+0.000200 test-aucpr:0.063744+0.001719   
## [327] train-aucpr:0.067313+0.000197 test-aucpr:0.063755+0.001729   
## [328] train-aucpr:0.067322+0.000189 test-aucpr:0.063760+0.001727   
## [329] train-aucpr:0.067350+0.000197 test-aucpr:0.063765+0.001736   
## [330] train-aucpr:0.067369+0.000196 test-aucpr:0.063776+0.001735   
## [331] train-aucpr:0.067398+0.000182 test-aucpr:0.063791+0.001751   
## [332] train-aucpr:0.067410+0.000183 test-aucpr:0.063798+0.001748   
## [333] train-aucpr:0.067419+0.000183 test-aucpr:0.063801+0.001751   
## [334] train-aucpr:0.067434+0.000184 test-aucpr:0.063803+0.001749   
## [335] train-aucpr:0.067444+0.000183 test-aucpr:0.063810+0.001740   
## [336] train-aucpr:0.067463+0.000188 test-aucpr:0.063819+0.001739   
## [337] train-aucpr:0.067467+0.000189 test-aucpr:0.063816+0.001734   
## [338] train-aucpr:0.067476+0.000193 test-aucpr:0.063825+0.001732   
## [339] train-aucpr:0.067480+0.000184 test-aucpr:0.063814+0.001731   
## [340] train-aucpr:0.067496+0.000193 test-aucpr:0.063816+0.001731   
## [341] train-aucpr:0.067502+0.000197 test-aucpr:0.063820+0.001734   
## [342] train-aucpr:0.067515+0.000198 test-aucpr:0.063822+0.001743   
## [343] train-aucpr:0.067533+0.000191 test-aucpr:0.063839+0.001712   
## [344] train-aucpr:0.067538+0.000187 test-aucpr:0.063837+0.001708   
## [345] train-aucpr:0.067550+0.000178 test-aucpr:0.063817+0.001725   
## [346] train-aucpr:0.067570+0.000189 test-aucpr:0.063810+0.001725   
## [347] train-aucpr:0.067597+0.000211 test-aucpr:0.063795+0.001716   
## [348] train-aucpr:0.067609+0.000213 test-aucpr:0.063797+0.001720   
## [349] train-aucpr:0.067620+0.000213 test-aucpr:0.063805+0.001719   
## [350] train-aucpr:0.067629+0.000217 test-aucpr:0.063804+0.001726   
## [351] train-aucpr:0.067635+0.000215 test-aucpr:0.063810+0.001733   
## [352] train-aucpr:0.067646+0.000216 test-aucpr:0.063827+0.001751   
## [353] train-aucpr:0.067677+0.000222 test-aucpr:0.063822+0.001737   
## Stopping. Best iteration:  
## [343] train-aucpr:0.067533+0.000191 test-aucpr:0.063839+0.001712

sel\_rounds7 = xgb.data.cv7$best\_iteration  
  
AUC7 <- xgb.data.cv7$evaluation\_log[sel\_rounds7]  
AUC7

## iter train\_aucpr\_mean train\_aucpr\_std test\_aucpr\_mean test\_aucpr\_std  
## 1: 343 0.0675326 0.0001914645 0.0638386 0.001712363

best\_xgb <- ifelse(AUC7 > best\_auc, "AUC7", best\_xgb)  
best\_auc <- ifelse(AUC7 > best\_auc, AUC7, best\_auc)  
  
# Not as good! Okay, now let's run the actual model  
  
xgb.data6 = xgboost(data = pred\_tr, label = resp\_tr,   
 max.depth = 3, eta = .05, nrounds = sel\_rounds6, min\_child\_weight = 1,  
 nfold = 5, early\_stopping\_rounds = 10, scale\_pos\_weight = neg.rate,  
 objective = "binary:logistic", eval\_metric = "aucpr")

## [21:48:23] WARNING: amalgamation/../src/learner.cc:573:   
## Parameters: { "nfold" } might not be used.  
##   
## This may not be accurate due to some parameters are only used in language bindings but  
## passed down to XGBoost core. Or some parameters are not used but slip through this  
## verification. Please open an issue if you find above cases.  
##   
##   
## [1] train-aucpr:0.051338   
## Will train until train\_aucpr hasn't improved in 10 rounds.  
##   
## [2] train-aucpr:0.052698   
## [3] train-aucpr:0.052870   
## [4] train-aucpr:0.052915   
## [5] train-aucpr:0.053809   
## [6] train-aucpr:0.055088   
## [7] train-aucpr:0.055543   
## [8] train-aucpr:0.056242   
## [9] train-aucpr:0.056637   
## [10] train-aucpr:0.057150   
## [11] train-aucpr:0.057279   
## [12] train-aucpr:0.057804   
## [13] train-aucpr:0.057943   
## [14] train-aucpr:0.058306   
## [15] train-aucpr:0.058692   
## [16] train-aucpr:0.058839   
## [17] train-aucpr:0.059344   
## [18] train-aucpr:0.059593   
## [19] train-aucpr:0.059723   
## [20] train-aucpr:0.059936   
## [21] train-aucpr:0.060170   
## [22] train-aucpr:0.060429   
## [23] train-aucpr:0.060682   
## [24] train-aucpr:0.060891   
## [25] train-aucpr:0.060907   
## [26] train-aucpr:0.061050   
## [27] train-aucpr:0.061251   
## [28] train-aucpr:0.061425   
## [29] train-aucpr:0.061480   
## [30] train-aucpr:0.061613   
## [31] train-aucpr:0.061820   
## [32] train-aucpr:0.061983   
## [33] train-aucpr:0.062086   
## [34] train-aucpr:0.062201   
## [35] train-aucpr:0.062316   
## [36] train-aucpr:0.062474   
## [37] train-aucpr:0.062570   
## [38] train-aucpr:0.062674   
## [39] train-aucpr:0.062746   
## [40] train-aucpr:0.062881   
## [41] train-aucpr:0.062949   
## [42] train-aucpr:0.062972   
## [43] train-aucpr:0.063029   
## [44] train-aucpr:0.063113   
## [45] train-aucpr:0.063202   
## [46] train-aucpr:0.063355   
## [47] train-aucpr:0.063425   
## [48] train-aucpr:0.063471   
## [49] train-aucpr:0.063488   
## [50] train-aucpr:0.063592   
## [51] train-aucpr:0.063693   
## [52] train-aucpr:0.063766   
## [53] train-aucpr:0.063763   
## [54] train-aucpr:0.063851   
## [55] train-aucpr:0.063933   
## [56] train-aucpr:0.064007   
## [57] train-aucpr:0.064027   
## [58] train-aucpr:0.064155   
## [59] train-aucpr:0.064178   
## [60] train-aucpr:0.064263   
## [61] train-aucpr:0.064356   
## [62] train-aucpr:0.064395   
## [63] train-aucpr:0.064477   
## [64] train-aucpr:0.064496   
## [65] train-aucpr:0.064566   
## [66] train-aucpr:0.064590   
## [67] train-aucpr:0.064729   
## [68] train-aucpr:0.064731   
## [69] train-aucpr:0.064757   
## [70] train-aucpr:0.064845   
## [71] train-aucpr:0.064895   
## [72] train-aucpr:0.064957   
## [73] train-aucpr:0.064985   
## [74] train-aucpr:0.064986   
## [75] train-aucpr:0.065089   
## [76] train-aucpr:0.065182   
## [77] train-aucpr:0.065230   
## [78] train-aucpr:0.065289   
## [79] train-aucpr:0.065293   
## [80] train-aucpr:0.065374   
## [81] train-aucpr:0.065450   
## [82] train-aucpr:0.065484   
## [83] train-aucpr:0.065501   
## [84] train-aucpr:0.065527   
## [85] train-aucpr:0.065577   
## [86] train-aucpr:0.065626   
## [87] train-aucpr:0.065660   
## [88] train-aucpr:0.065684   
## [89] train-aucpr:0.065737   
## [90] train-aucpr:0.065768   
## [91] train-aucpr:0.065836   
## [92] train-aucpr:0.065857   
## [93] train-aucpr:0.065925   
## [94] train-aucpr:0.065988   
## [95] train-aucpr:0.066019   
## [96] train-aucpr:0.066041   
## [97] train-aucpr:0.066072   
## [98] train-aucpr:0.066127   
## [99] train-aucpr:0.066156   
## [100] train-aucpr:0.066213   
## [101] train-aucpr:0.066243   
## [102] train-aucpr:0.066235   
## [103] train-aucpr:0.066293   
## [104] train-aucpr:0.066315   
## [105] train-aucpr:0.066351   
## [106] train-aucpr:0.066416   
## [107] train-aucpr:0.066438   
## [108] train-aucpr:0.066480   
## [109] train-aucpr:0.066540   
## [110] train-aucpr:0.066564   
## [111] train-aucpr:0.066599   
## [112] train-aucpr:0.066619   
## [113] train-aucpr:0.066632   
## [114] train-aucpr:0.066670   
## [115] train-aucpr:0.066680   
## [116] train-aucpr:0.066701   
## [117] train-aucpr:0.066824   
## [118] train-aucpr:0.066869   
## [119] train-aucpr:0.066915   
## [120] train-aucpr:0.066959   
## [121] train-aucpr:0.066987   
## [122] train-aucpr:0.067005   
## [123] train-aucpr:0.067025   
## [124] train-aucpr:0.067066   
## [125] train-aucpr:0.067087   
## [126] train-aucpr:0.067141   
## [127] train-aucpr:0.067184   
## [128] train-aucpr:0.067218   
## [129] train-aucpr:0.067303   
## [130] train-aucpr:0.067342   
## [131] train-aucpr:0.067344   
## [132] train-aucpr:0.067427   
## [133] train-aucpr:0.067438   
## [134] train-aucpr:0.067453   
## [135] train-aucpr:0.067492   
## [136] train-aucpr:0.067537   
## [137] train-aucpr:0.067576   
## [138] train-aucpr:0.067590   
## [139] train-aucpr:0.067611   
## [140] train-aucpr:0.067635   
## [141] train-aucpr:0.067652   
## [142] train-aucpr:0.067757   
## [143] train-aucpr:0.067772   
## [144] train-aucpr:0.067780   
## [145] train-aucpr:0.067794   
## [146] train-aucpr:0.067822   
## [147] train-aucpr:0.067855   
## [148] train-aucpr:0.067870   
## [149] train-aucpr:0.067982   
## [150] train-aucpr:0.068007   
## [151] train-aucpr:0.068038   
## [152] train-aucpr:0.068068   
## [153] train-aucpr:0.068145   
## [154] train-aucpr:0.068167   
## [155] train-aucpr:0.068191   
## [156] train-aucpr:0.068261   
## [157] train-aucpr:0.068295   
## [158] train-aucpr:0.068333   
## [159] train-aucpr:0.068350   
## [160] train-aucpr:0.068365   
## [161] train-aucpr:0.068404   
## [162] train-aucpr:0.068431   
## [163] train-aucpr:0.068508   
## [164] train-aucpr:0.068519   
## [165] train-aucpr:0.068537   
## [166] train-aucpr:0.068571   
## [167] train-aucpr:0.068586   
## [168] train-aucpr:0.068605   
## [169] train-aucpr:0.068614   
## [170] train-aucpr:0.068646   
## [171] train-aucpr:0.068699   
## [172] train-aucpr:0.068745   
## [173] train-aucpr:0.068783   
## [174] train-aucpr:0.068817   
## [175] train-aucpr:0.068836   
## [176] train-aucpr:0.068858   
## [177] train-aucpr:0.068871   
## [178] train-aucpr:0.068892   
## [179] train-aucpr:0.068929   
## [180] train-aucpr:0.068936   
## [181] train-aucpr:0.068971   
## [182] train-aucpr:0.069034   
## [183] train-aucpr:0.069052   
## [184] train-aucpr:0.069149   
## [185] train-aucpr:0.069166   
## [186] train-aucpr:0.069183   
## [187] train-aucpr:0.069245   
## [188] train-aucpr:0.069269   
## [189] train-aucpr:0.069312   
## [190] train-aucpr:0.069347   
## [191] train-aucpr:0.069368   
## [192] train-aucpr:0.069398   
## [193] train-aucpr:0.069480   
## [194] train-aucpr:0.069524   
## [195] train-aucpr:0.069611   
## [196] train-aucpr:0.069661   
## [197] train-aucpr:0.069683   
## [198] train-aucpr:0.069696   
## [199] train-aucpr:0.069765   
## [200] train-aucpr:0.069774   
## [201] train-aucpr:0.069794   
## [202] train-aucpr:0.069810   
## [203] train-aucpr:0.069827   
## [204] train-aucpr:0.069873   
## [205] train-aucpr:0.069874   
## [206] train-aucpr:0.069910   
## [207] train-aucpr:0.069921   
## [208] train-aucpr:0.069931   
## [209] train-aucpr:0.070011   
## [210] train-aucpr:0.070034   
## [211] train-aucpr:0.070073   
## [212] train-aucpr:0.070087   
## [213] train-aucpr:0.070148   
## [214] train-aucpr:0.070173   
## [215] train-aucpr:0.070200   
## [216] train-aucpr:0.070280   
## [217] train-aucpr:0.070301   
## [218] train-aucpr:0.070315   
## [219] train-aucpr:0.070363   
## [220] train-aucpr:0.070425   
## [221] train-aucpr:0.070447   
## [222] train-aucpr:0.070482   
## [223] train-aucpr:0.070483   
## [224] train-aucpr:0.070491   
## [225] train-aucpr:0.070507   
## [226] train-aucpr:0.070527   
## [227] train-aucpr:0.070549   
## [228] train-aucpr:0.070574   
## [229] train-aucpr:0.070586   
## [230] train-aucpr:0.070630   
## [231] train-aucpr:0.070632   
## [232] train-aucpr:0.070654   
## [233] train-aucpr:0.070674   
## [234] train-aucpr:0.070688   
## [235] train-aucpr:0.070700   
## [236] train-aucpr:0.070741   
## [237] train-aucpr:0.070767   
## [238] train-aucpr:0.070790   
## [239] train-aucpr:0.070811

pred = predict(xgb.data6, pred\_val)  
pred.xgb.data = ifelse(pred > 0.5, 1, 0)  
  
  
# Misclassification errors in the training data  
tab <- table(resp\_val, pred.xgb.data)  
  
  
cm.gbm = confusionMatrix(tab)  
cm.gbm$byClass

## Sensitivity Specificity Pos Pred Value   
## 0.9739356 0.0559052 0.6298011   
## Neg Pred Value Precision Recall   
## 0.5653351 0.6298011 0.9739356   
## F1 Prevalence Detection Rate   
## 0.7649457 0.6225169 0.6062914   
## Detection Prevalence Balanced Accuracy   
## 0.9626712 0.5149204

cm.gbm$table

## pred.xgb.data  
## resp\_val 0 1  
## 0 71968 42303  
## 1 1926 2505

# F1 score of classification accuracy for test data  
cm.gbm$byClass[7]

## F1   
## 0.7649457

## Now to calculate gini  
  
print("Gini coefficient equals:")

## [1] "Gini coefficient equals:"

normalizedGini <- function(aa, pp) {  
 Gini <- function(a, p) {  
 if (length(a) != length(p)) stop("Actual and Predicted need to be equal lengths!")  
 temp.df <- data.frame(actual = a, pred = p, range=c(1:length(a)))  
 temp.df <- temp.df[order(-temp.df$pred, temp.df$range),]  
 population.delta <- 1 / length(a)  
 total.losses <- sum(a)  
 null.losses <- rep(population.delta, length(a)) # Hopefully is similar to accumulatedPopulationPercentageSum  
 accum.losses <- temp.df$actual / total.losses # Hopefully is similar to accumulatedLossPercentageSum  
 gini.sum <- cumsum(accum.losses - null.losses) # Not sure if this is having the same effect or not  
 sum(gini.sum) / length(a)  
 }  
 Gini(aa,pp) / Gini(aa,aa)  
}  
  
resp\_val <- data\_resp[idx == 1]  
normalizedGini(resp\_val$target, pred)

## [1] 0.2751364

xgb.data.cv$evaluation\_log[sel\_rounds]

## iter train\_aucpr\_mean train\_aucpr\_std test\_aucpr\_mean test\_aucpr\_std  
## 1: 43 0.1012588 0.002051856 0.0631388 0.0017634

xgb.data.cv4$evaluation\_log[sel\_rounds4]

## iter train\_aucpr\_mean train\_aucpr\_std test\_aucpr\_mean test\_aucpr\_std  
## 1: 85 0.1034818 0.002259056 0.0632758 0.002101165

## iter train\_aucpr\_mean train\_aucpr\_std test\_aucpr\_mean test\_aucpr\_std  
## 1: 239 0.0721008 0.0004772678 0.064601 0.001954284

## Precision   
## 0.6298011

## Recall   
## 0.9739356

## [1] "Importance Matrix Top 10"

## Feature Gain Cover Frequency  
## 1: ps\_car\_13 0.22745522 0.09911149 0.09358128  
## 2: ps\_ind\_03 0.08530166 0.12960714 0.11217756  
## 3: ps\_reg\_03 0.08470981 0.04335991 0.05878824  
## 4: ps\_ind\_05\_cat 0.07851275 0.05782201 0.04799040  
## 5: ps\_ind\_17\_bin 0.07328443 0.02890187 0.02759448  
## 6: ps\_car\_07\_cat 0.04752001 0.02687975 0.01859628  
## 7: ps\_ind\_15 0.04624639 0.07056401 0.06058788  
## 8: ps\_ind\_06\_bin 0.03874686 0.01497341 0.00839832  
## 9: ps\_car\_01\_cat 0.03659037 0.06194511 0.04559088  
## 10: ps\_reg\_01 0.03616844 0.05949489 0.04559088

## [1] "Gini coefficient equals: 0.2751364"