Ecommerce Customer Retention with Machine Learning

Margaret Catherman

Masters in Analytics Candidate

Georgia Institute of Technology

April 10, 2022

Overview

A. Introduction
B. The Data: Preparation & Exploration
C. Methods
D. Findings and Results
E. Conclusions
F. Future Study
G. Appendix

A. Introduction

1. Abstract
2. Objective
3. Domain Knowledge

1. Abstract

E-commerce websites have seen significant growth over the past ten years

Customer retention is integral to online retailers' success. Used:

- In business metrics
- Target marketing campaigns

E-commerce websites = big data; Machine Learning Algorithms well suited.

Challenges of data:

- irrelative
- collinearity
- skewed
- missing values

Model Selection Consideration

- data attributes
- parameter tuning
- feature selection
- optimization

2. Objective

To determine the machine learning model with the lowest error rate for predicting churn among the ecommerce data set, using feature selection and parameter tuning for optimization.

3. Domain Knowledge: Churn & the E-commerce site

Churn is a count of a customer or subscriber that has un-subscribed.

Churn has a direct impact on several business metrics:

- monthly recurring revenue (MRR)
- customer lifetime value
- customer acquisition costs

Reasons for Churn:

- Poor onboarding, difficulty navigating website
- Weak customer relations
- Poor customer service
- Metrics of these elements might correlate with churn

Machine learning and e-commerce analysis.

- By identifying customers that might churn, promotional efforts may be extended in an effort to prevent churn.
- E-commerce sites are rich with data ideally suited for machine learning predictive analysis.

Source: (https://onix-systems.com/blog/customer-churn-rate-and-its-impact-on-business-performance).

B. The Data: Preparation & Exploration

1. The Original Data Set
2. Data Preparation
3. Exploratory Data Analysis (EDA)
4. Standardize Data

1. The Data Set This is an **E-Commerce** data set, reflecting customer behaviors and characteristics on an online e-commerce site.

- Dependent variable: Churn, and 19 predictors
- 5630 rows and 20 columns.
- 11 categorical, 9 numerical
- Missing Values in 1856 rows

Variable <chr>	Discerption <chr>
CustomerID	Unique customer ID
Churn	Churn Flag
Tenure	Tenure of customer in organization
PreferredLoginDevice	Preferred login device of customer
CityTier	City tier
WarehouseToHome	Distance in between warehouse to home of customer
PreferredPaymentMode	Preferred payment method of customer
Gender	Gender of customer
HourSpendOnApp	Number of hours spend on mobile application or website
NumberOfDeviceRegistered	Total number of deceives is registered on particular customer
PreferedOrderCat	Preferred order category of customer in last month
SatisfactionScore	Satisfactory score of customer on service
MaritalStatus	Marital status of customer
NumberOfAddress	Total number of added added on particular customer
Complain	Any complaint has been raised in last month
OrderAmountHikeFromlastYear	Percentage increases in order from last year
CouponUsed	Total number of coupon has been used in last month
OrderCount	Total number of orders has been places in last month
DaySinceLastOrder	Day Since last order by customer
CashbackAmount	Average cashback in last month

Link to data: https://www.kaggle.com/datasets/ankitverma2010/ecommerce-customer-churn-analysis-and-prediction

2. Data Preparation

i. Missing Values: were omitted

ii. Dummy variables created: adding 16 new variables; 36 total.

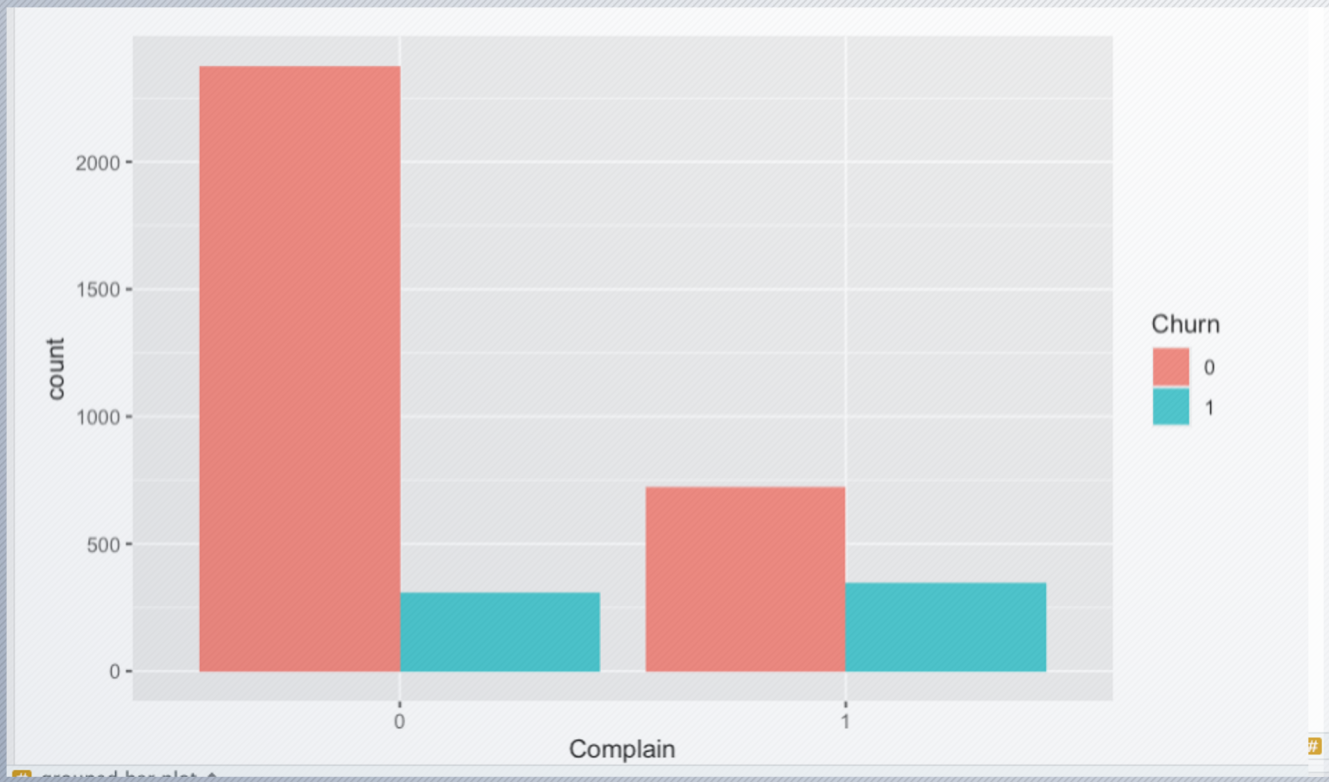
iii. Train/Test random split: train (2/3) & test (1/3); prior to data prep & analysis, to prevent “leakage.”

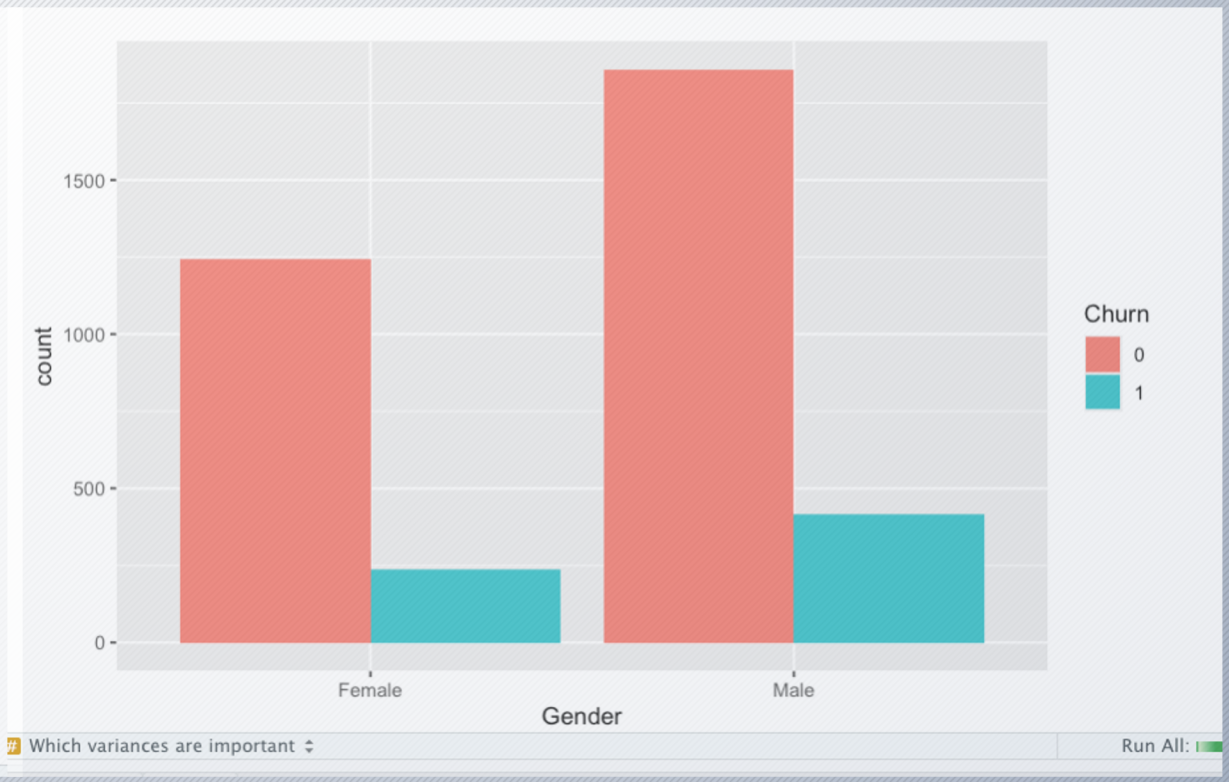
iv. Standardize Data (after exploration)

3. Exploratory Data Analysis (EDA) of training set

i. Visualization
ii. Collinearity
iii. Skewed Features

i. Visualization: Churn is generally a small fraction of all groups. The bar chart for Complain, at left, shows a larger percent of customers that complain churn.





ii. Collinearity

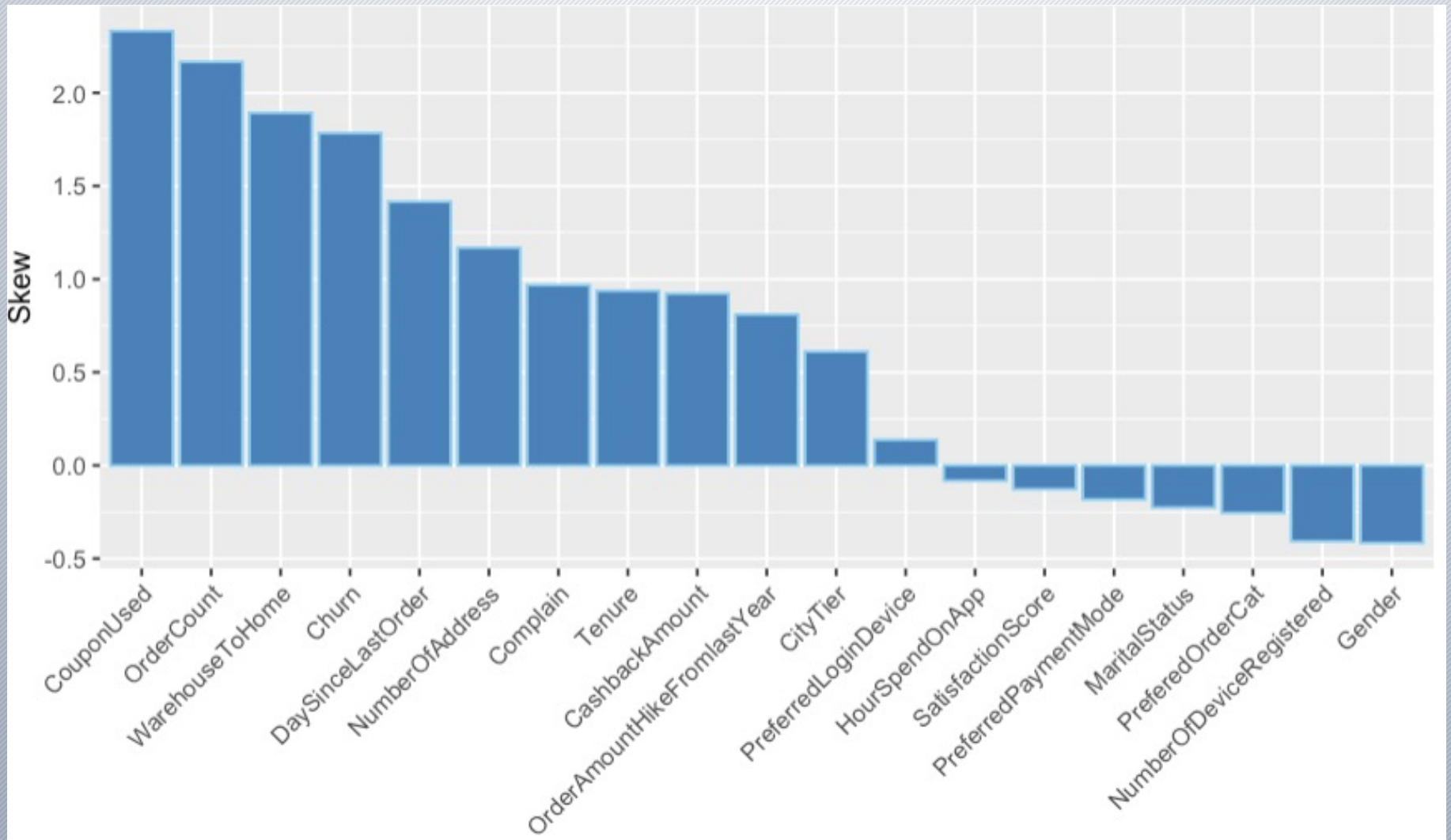
The chart identifies several predictors highly corelated with each other: “Order Count” & “Coupon Used”; Marital Status “Single” & “Married’; & Preferred Device “Mobile” & “Computer”.
For each of these pairs, one member should be removed. We will keep this in mind as we perform additional feature analysis.

var1 <chr>	var2 <chr>	value <dbl>	abs_cor <dbl>
OrderCount	CouponUsed	0.7735315	0.7735315
MaritalStatus.Single	MaritalStatus.Married	-0.7330222	0.7330222
PreferredLoginDevice.Mobile.Phone	PreferredLoginDevice.Computer	-0.6303958	0.6303958
PreferedOrderCat.Others	CashbackAmount	0.5629242	0.5629242
PreferredPaymentMode.E.wallet	CityTier	0.5304556	0.5304556
PreferedOrderCat.Grocery	CashbackAmount	0.5257094	0.5257094
DaySinceLastOrder	OrderCount	0.5209692	0.5209692
PreferredLoginDevice.Phone	PreferredLoginDevice.Mobile.Phone	-0.5188943	0.5188943
PreferredPaymentMode.Debit.Card	PreferredPaymentMode.Credit.Card	-0.5029775	0.5029775

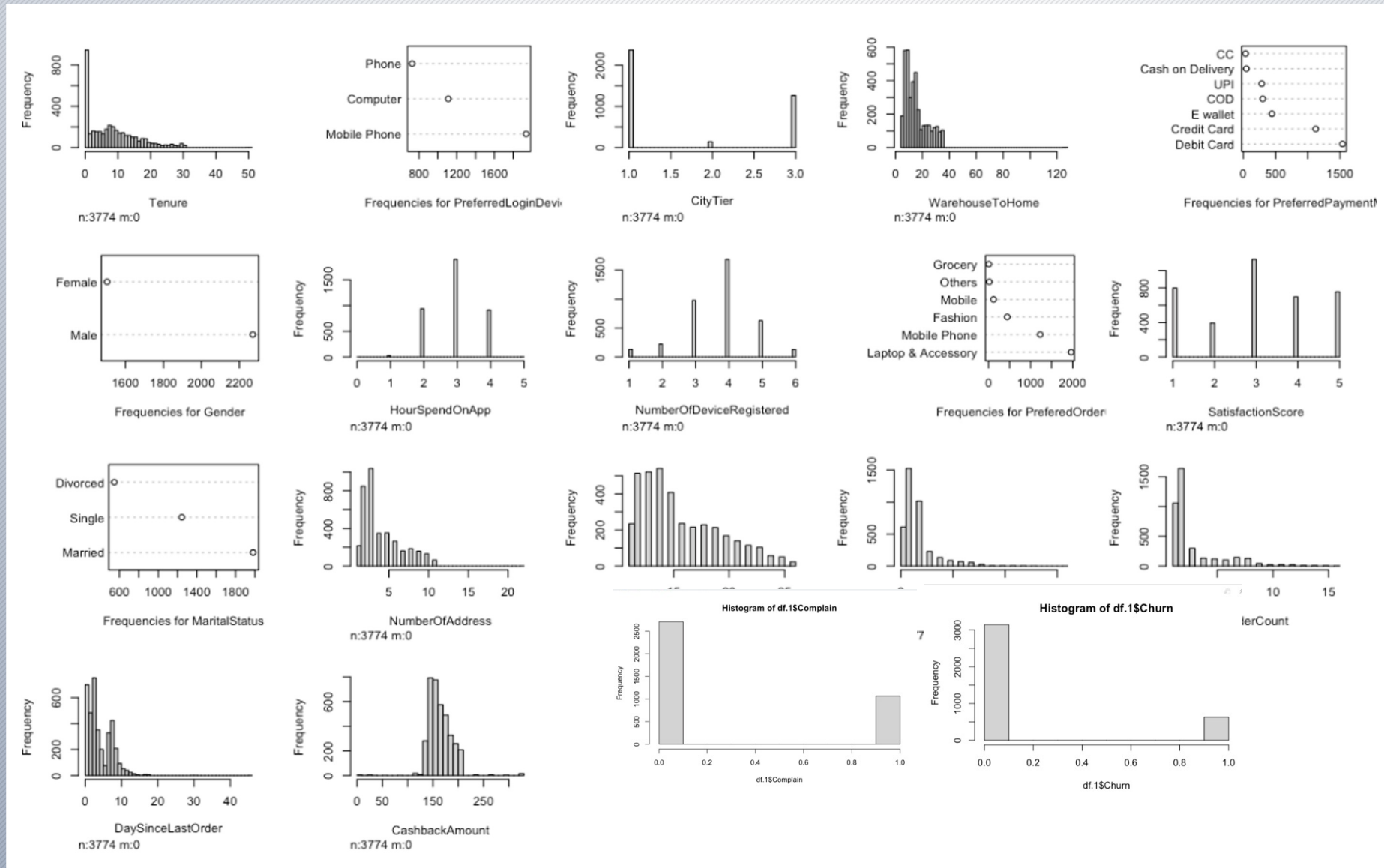
9 rows

iii. Skewed Feature Ranking.

Positively skewed features have values greater than 0; negatively skewed have values less than 0. The farther the absolute value is from 0, the greater it is skewed. A skew value of 0 indicates a normal distribution. This is relevant to certain models, as some do not work well with skewed data.

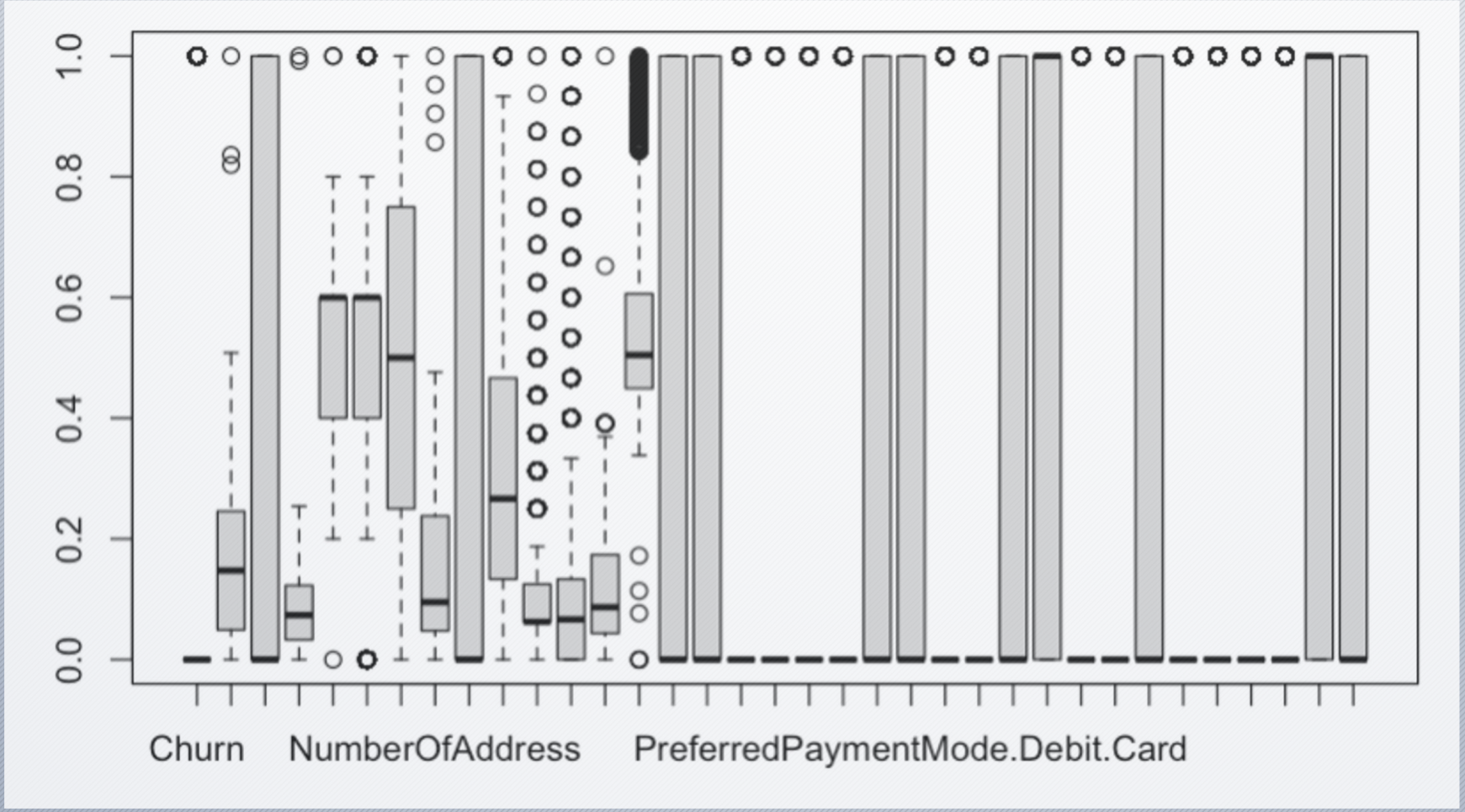


iii. Skewed Features, visualized in histograms.



4. Standardize the E-commerce data.

The data is scaled to a fixed range of 0 to 1. This results in smaller standard deviations; helps to suppress effect of outliers, and prevents the model from assigning heavier importance to variables whose values are at a much larger scale than others.



C. Methods

1. Models Used
2. Parameter Tuning
3. Feature Selection

1. Models Used

Random Forest is an ensemble of decision trees; the algorithm predicts new data by aggregating the predictions of the trees. It picks a bootstrap sample of input variables from the training data set. From these, it makes a tree of size square root of p variables for classification or p/3 for regression (default values).

The **Boosting Algorithm** applies logistic regression techniques to the AdaBoost method by minimizing the logistic loss. Unlike random forests, that builds ensembles deep independent trees, here the ensembles are of weak, shallow and successive, with the trees improving and learning from the previous. However, together, these trees provide a strong committee that out performs many other algorithms. (http://uc-r.github.io/gbm_regression)

.

1. Models Used (continued)

Linear Discriminant Analysis (LDA) A Linear Classification method developed by Fisher (1936), LDA tries to approximate the Naïve Bayes method. It assigns an observation to the class for which the posterior probability *pk*(X) is greatest. So, for a binomial response such as Churn, the observations will be assigned to the default class if Pr(Chrun =Yes|X = x) > 0.5

Naive Bayes Based on Bayes theorem, this is a probabilistic classifier with emphasis on the assumption of independence between the features.

Classification Tree algorithm recursively partitions the predictor space into subsets in which the distribution of the dependent variable is successively more homogeneous.

2. Parameter Tuning:

Steps taken to arrive at optimal parameter tuning using a hyperparameter grid, the ranger library & a loop.

A. Random Forest optimal parameters

- mtry of ?
- number of trees is ?
- terminal node size of observations ?
- sample size of ?

% of the training data.

- B. Boosting optimal parameters

shrinkage at ?

Interaction depth at ?

n.minobsinnode at ?

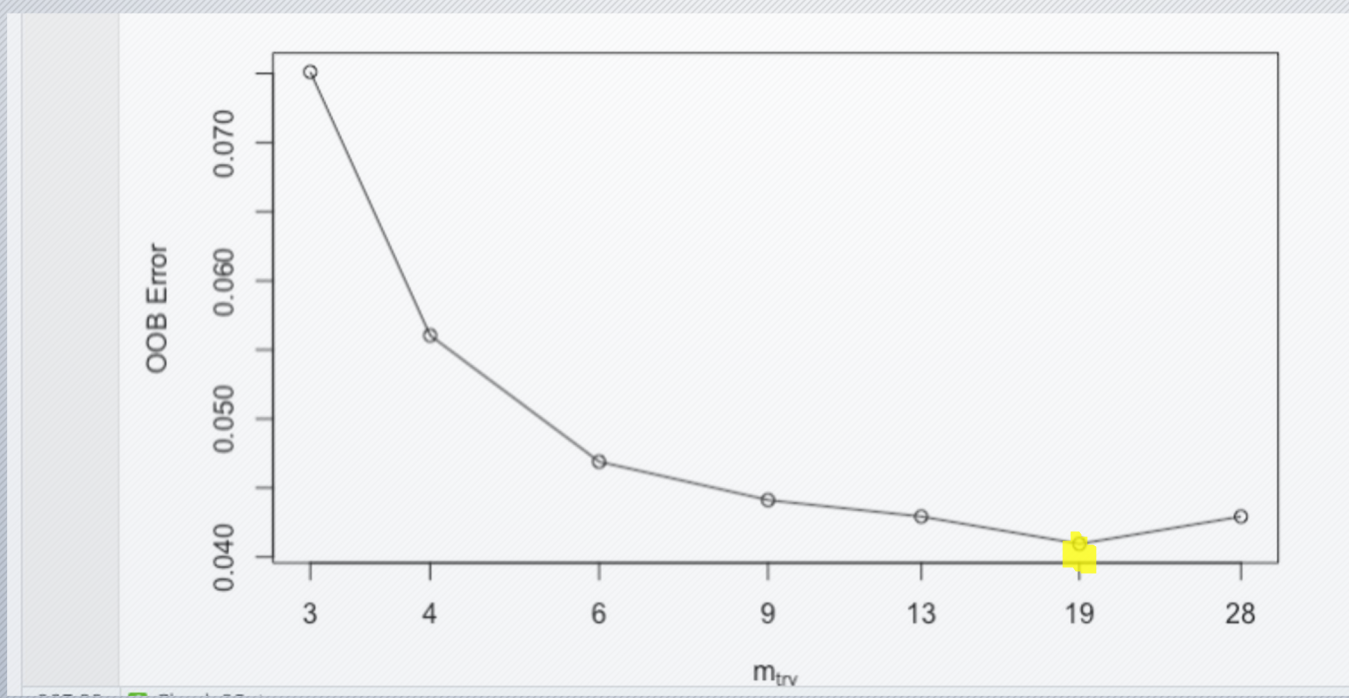
bag fraction at ?

iterations at ?

10 rounds of cross validation.

A. Random Forest Optimization Step 1: Mtry Values for Optimal Error Rate.

Will enter the suggested optimal mtry value, in this case 19, as a starting value for Step 2.

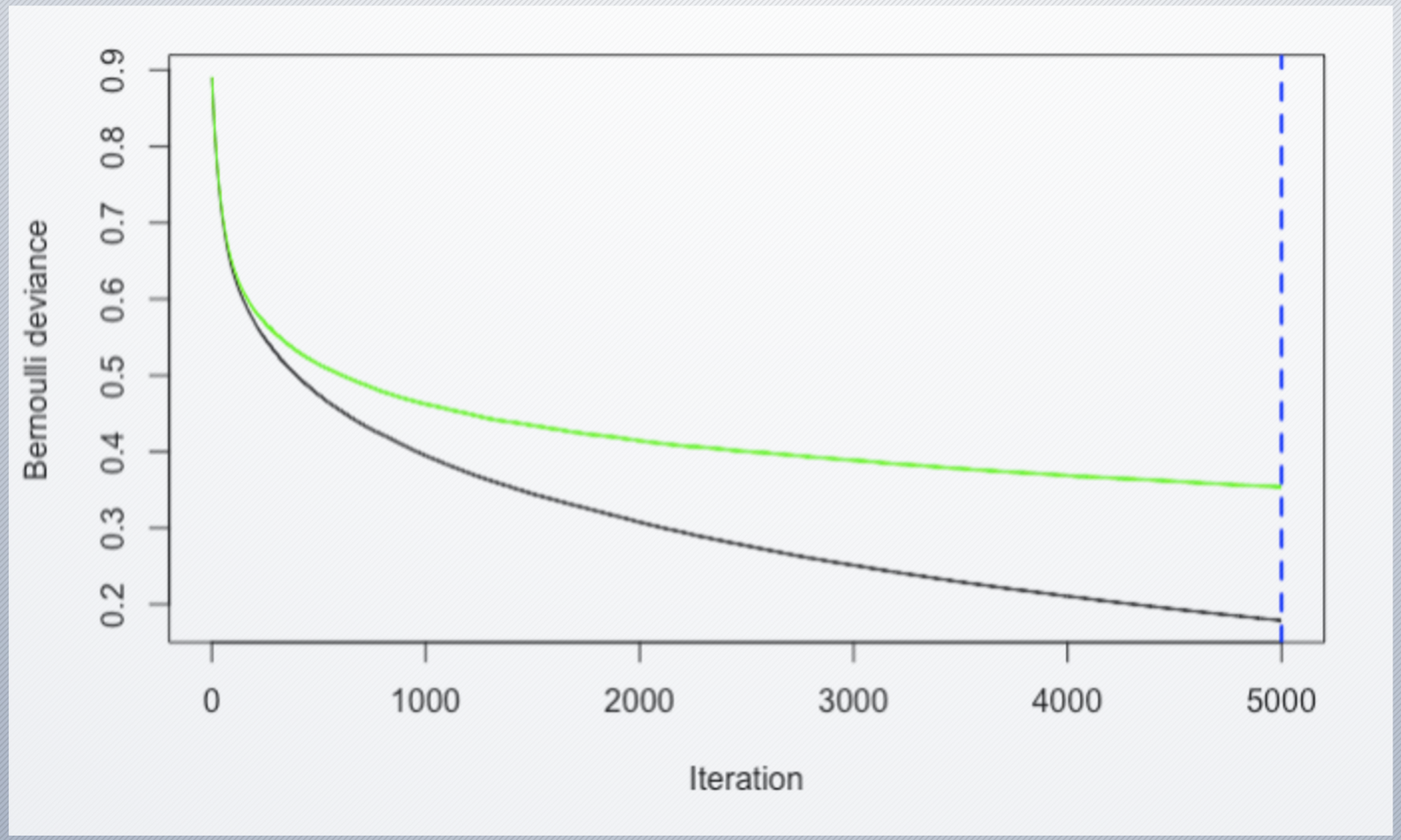


Step 2: Parameter Tuning for Random Forest using a hyperparameter grid with the ranger library in r and a loop, to find the optimal parameters for Random Forest. A range of values are entered for each parameter in the grid; then the values are looped thru the ranger mode. The chart shows the models with optimal parameter tuning combinations; ranked by the 10 lowest error rates. I selected the values in row 1; see three slides ahead

Description: df [10 × 5]

	num.trees <dbl>	mtry <dbl>	node_size <dbl>	sampe_size <dbl>	OOB_RMSE <dbl>
1	1500	19	3	0.8	0.2228947
2	1500	21	3	0.8	0.2228947
3	1500	17	3	0.8	0.2237845
4	1000	19	3	0.8	0.2237845
5	1000	17	3	0.8	0.2246708
6	1000	21	3	0.8	0.2255535
7	500	21	3	0.8	0.2264329
8	500	23	3	0.8	0.2264329
9	1500	23	3	0.8	0.2264329
10	500	19	3	0.8	0.2273089

Step 1: Boosting Optimal Iterations. A similar process to Random Forest. WilB. Boosting Optimization l enter the suggested value, in this case 5000, as a starting value for Step 2.



Step2: Parameter Tuning for Boosting, similar to the approach used for RF, using a hyperparameter grid, the ranger library in r and a loop. The chart shows the models with optimal parameter tuning combinations; ranked by the 10 lowest error rates. I selected the values in row 1; see next slide.

Description: df [10 × 6]

	shrinkage <dbl>	interaction.depth <dbl>	n.minobsinnode <dbl>	bag.fraction <dbl>	optimal_trees <dbl>	min_RMSE <dbl>
1	0.3	5	5	1.0	312	0.4958292
2	0.1	5	5	1.0	1221	0.5055278
3	0.1	5	10	1.0	857	0.5063090
4	0.3	3	5	1.0	528	0.5069493
5	0.1	5	5	0.8	1063	0.5187142
6	0.1	3	5	1.0	1475	0.5226770
7	0.3	3	10	1.0	552	0.5326193
8	0.1	3	5	0.8	1343	0.5328580
9	0.1	5	10	0.8	848	0.5367445
10	0.1	3	10	1.0	1642	0.5370613

1-10 of 10 rows

The resulting optimal parameters from Steps 1 & 2.

- Optimal parameters Random Forest

• mtry of	19
• number of trees is	1500
• terminal node size of observations	3
• sample size of of the training data.	80%

- Optimal parameters Boosting

shrinkage at	.3
Interaction depth at	5
n.minobsinnode at	5
bag fraction at	1
iterations at	312

10 rounds of cross validation.

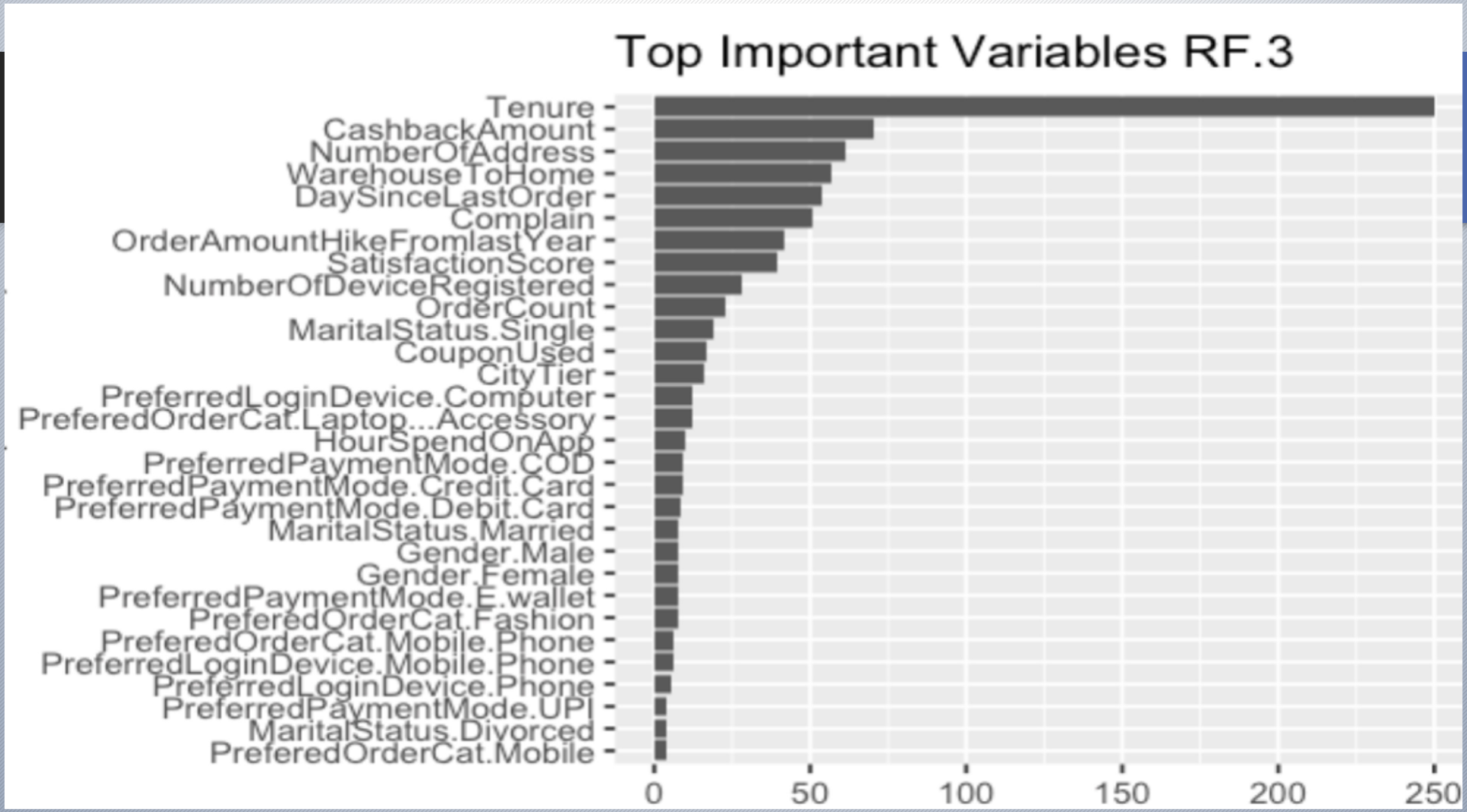
3. Feature Selection

Feature selection is a critical component of model optimization. Two areas of particular importance are:

1. Omitting week, irrelevant predictors
2. Collinearity: Identifying, and removing, a parameter that is highly corelated with another.

How feature importance was assessed for the E-commerce data:

1. Consensus of Feature importance ranking by Random Forest & Boosting.


Top Important Variables RF.3
Tenure
CashbackAmount
NumberOfAddress
WarehouseToHome
DaySinceLastOrder
Complain
OrderAmountHikeFromlastYear
SatisfactionScore
NumberOfDeviceRegistered
OrderCount
MaritalStatus.Single
CouponUsed
CityTier
PreferredLoginDevice.Computer
PreferedOrderCat.Laptop...Accessory
HourSpendOnApp
PreferredPaymentMode.COD
PreferredPaymentMode.Credit.Card
PreferredPaymentMode.Debit.Card
MaritalStatus.Married
Gender.Male
Gender.Female
PreferredPaymentMode.E.wallet
PreferedOrderCat.Fashion
PreferedOrderCat.Mobile.Phone
PreferredLoginDevice.Mobile.Phone
PreferredLoginDevice.Phone
PreferredPaymentMode.UPI
MaritalStatus.Divorced
PreferedOrderCat.Mobile
0
50
100
150
200
250

Feature Selection: Features ranked by influence. There is a consensus on nine of the top ten features, so we will (generally) select these nine and re-run the models with the objective of increasing performance. Omitted will be: Order Count, as it showed colinearly with Days Since Last Order, and Marital Status, as it did not appear in both lists. Number of Device Registered was left in to make 9.

Random Forest	
Tenure	249.94130
CashbackAmount	70.09870
NumberOfAddress	61.09667
WarehouseToHome	57.07053
DaySinceLastOrder	53.77676
Complain	50.84332
OrderAmountHikeFromlastYear	41.67930
SatisfactionScore	39.49538
NumberOfDeviceRegistered	27.81674
OrderCount	22.87587

Boosting	
Tenure	37.05453919
CashbackAmount	9.06248470
Complain	7.88826591
NumberOfAddress	7.30042381
DaySinceLastOrder	5.46938052
WarehouseToHome	5.22657380
SatisfactionScore	4.16949403
OrderAmountHikeFromlastYear	3.37558393
OrderCount	2.69099570
MaritalStatus.Single	2.47959887

Feature Selection: Top 9 Features

Tenure

Number of Address

Cash back Amount

Warehouse to Home

Day Since Last Order

Complain

Order Amount Hike From Last Year

Satisfaction Score

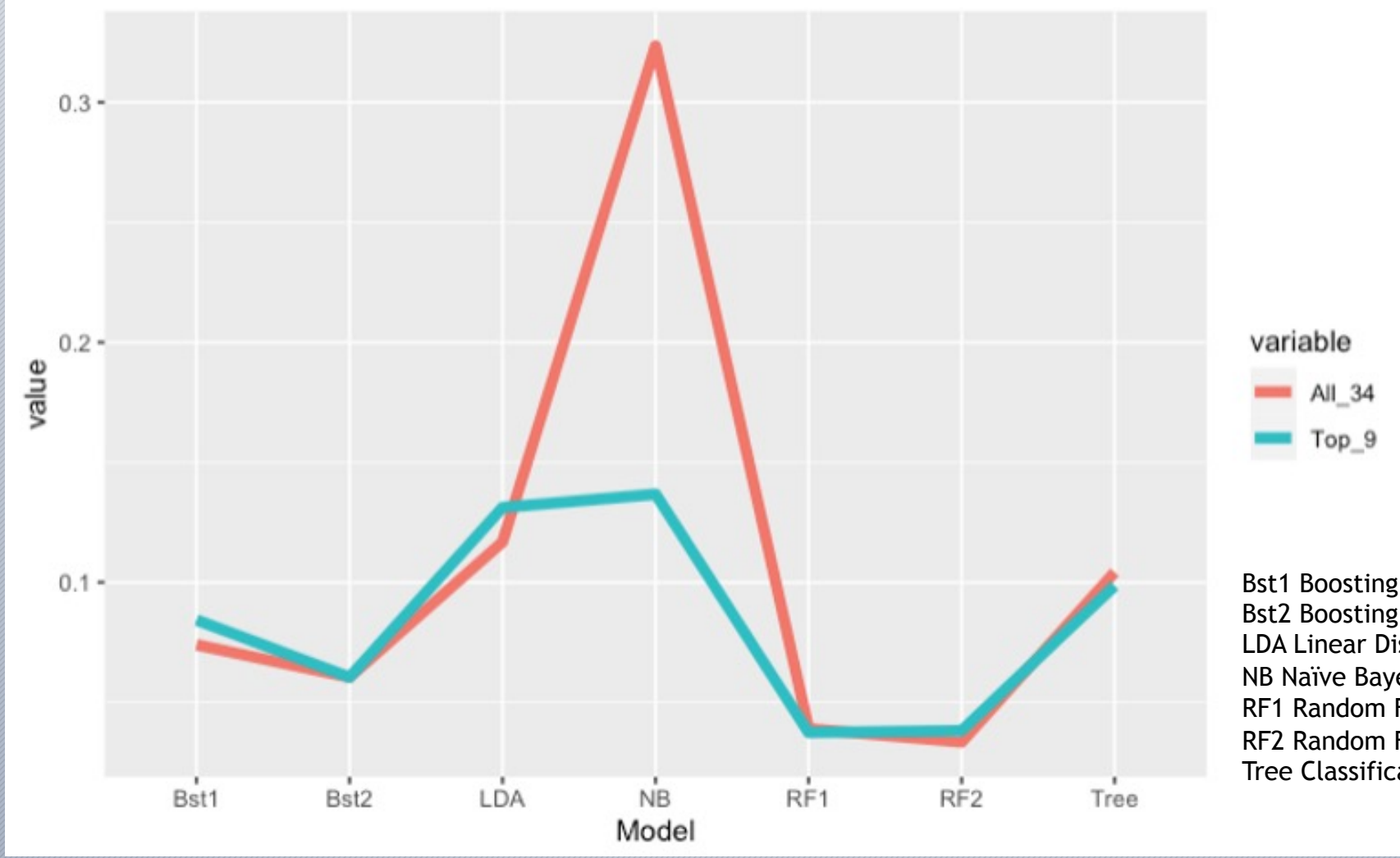
Number of Devises Registered.

Note: All features appeared on both the Random Forest & Boosting top 10 Relevant List, except number of Devises Registered. Order Count was excluded, because it has a high collinearity with Day Since Last Order. Marital status Single was omitted because it was the lowest ranked on the Boosting list, and was not on the Random Forest list.

D. Findings & Results

D. Findings & Results Graph: Test errors for each model.

Best performing model: Optimized Random Forest 2 (3% error rate), on all 34 features.
Analysis of parameter tuning: Boosting showed a slight improvement. Random Forest was virtually unchanged.
Analysis of feature reduction from 34 to Top 9 (red vs. teal lines): Naïve Bayes showed the greatest improvement. Tree & Random Forest1 showed slight improvement. Boosting 2 was unchanged. Random Forest 2, Boosting 1 and LDA each performed worse.



Bst1 Boosting default
Bst2 Boosting Optimized
LDA Linear Discriminant Analysis
NB Naïve Bayes
RF1 Random Forest default
RF2 Random Forest Optimized
Tree Classification Tree

D. Findings & Results Table: Feature reduction's impact on model error rate. The 34 features were reduced to the top 9.

Naïve Bayes showed the greatest improvement, or error reduction (-57%). Tree (-5%) & Random Forest1 (-4%) showed slight error reduction. Boosting 2 was unchanged.

Several models performed slightly worse; that is, reducing features *increased* error rates: Random Forest 2 (14%), Boosting 1 (13%) and LDA (12%).

X <chr>	All_34 <dbl>	Top_9 <dbl>	Delta <dbl>	Delta_Percent <dbl>
Bst1	0.07392687	0.08426073	0.0103	0.1393
Bst2	0.06041335	0.06041335	0.0000	0.0000
LDA	0.11685215	0.13116057	0.0143	0.1224
NB	0.32352941	0.13672496	-0.1868	-0.5774
RF1	0.03895072	0.03736089	-0.0016	-0.0411
RF2	0.03338633	0.03815580	0.0048	0.1438
Tree	0.10413355	0.09856916	-0.0056	-0.0538

All_34: All 34 predictors
Top_9: Top 9 predictors
Delta: Change in error rate due to feature reduction.

Bst1 Boosting default
Bst2 Boosting Optimized
LDA Linear Discriminant Analysis
NB Naïve Bayes

RF1 Random Forest default
RF2 Random Forest Optimized
Tree Classification Tree

E. Conclusions

E. Conclusions

- Random Forest models are known for their superior performance for classification, and here they did not disappoint, as the two Random Forest Models held the lowest error rate on the test data, in all four scenarios: before and after tuning, with all predictors and with just the top 9 best predictors, when compared to the four other methods used. Boosting, also known to be a strong prediction, proved to be so with this analysis, coming in second to Random Forest for lowest error prediction.
- It is not surprising that reducing the features had minimal to negative impact on Random Forest. The randomness of the selection makes each tree less corelated; highly corelated variables play a nearly equivalent role (unlike some other models, that may give slightly more predictive variables preference.) This decorrelation of the tree lowers the predictive error. Thus it stands to reason that reducing the number of feature would be less significance on this models performance; as by its nature, Random Forest has a built in method of dealing with predictors of low relevance or high correlation.

E. Conclusions (continued)

- Finally, and perhaps most importantly, how could these findings be used by an e-commerce business to reduce customer attrition and help improve the bottom line?
- The identification of the highly influential features on Churn can be used to guide an e-commerce business in metrics to measure and threshold values to set to trigger identification of customers likely to Churn.
- This customer segment can then be contacted in various ways in an effort to retain their business, such as with special promotions, coupons, or even be contacted by phone by customer service.
- These top features can also be incorporated in analysis used to compute broader business metrics, such as forecasting, monthly recurring revenue (MRR), customer lifetime value, and customer acquisition costs.

F. Future Study

- Research causes of possible collinearity between Tenure and Churn.

G. Appendix

- 1. The Ecommerce Data set: https://www.kaggle.com/datasets/ankitverma2010/ecommerce-customer-churn-analysis-and-prediction
- 2. The Ecommerce Data Analysis on Kaggle
- https://www.kaggle.com/datasets/ankitverma2010/ecommerce-customer-churn-analysis-and-prediction
-
- 3. Overview of Random Forest parameter tuning, including hyperparameter grid loop in ranger: https://uc-r.github.io/random_forests
- 4. Overview of Boosting parameter tuning
- http://uc-r.github.io/gbm_regression