Predicting Churn in a

Telecommunications Customer Dataset

Alexander Hyman, Spalding Lewis, David Mackey, Olga Perera

[Introduction](#iq0g4obyk) 1

[Project Background and Description](#jdbro0a3nys3) 1

[Project Scope and Context for Analysis](#fr1qdv0mijd) 2

[Business Questions](#xz2gj2s5n1ft) 3

[Data Overview](#ke7qc7g7ymm4) 4

[Data Source](#rz88riuq57lk) 5

[Data Quality](#mgmx7dt7lkht) 5

[Data Selection](#io1soyf4zg7y) 5

[Data Dictionary](#ctmkhdew5dh6) 7

[Data Structure](#yhdnv4foblnv) 9

[Data Findings](#v1l3cm7p4zvw) 10

[Statistical Analysis](#lneezgh96lze) 10

[Summary Statistics](#ytmh87bflfy4) 11

[Descriptive Statistics](#tjnfzh8rdj9r) 14

[Inferential Statistics](#cg3nxr8t384a) 20

[Statistical Analysis Overview](#ypgg5dreke6j) 23

[Linear Modeling](#nfv6rcjgwj0) 24

[All Customers](#ht0ivqsm2tsi) 24

[Internet Customers](#r9a34t8msnls) 26

[Data Mining](#czlimn79jshv) 27

[Support Vector Machines](#nilzv4mce7m6) 28

[Naive Bayes](#to34yy3im95s) 35

[Decision Tree](#xzvwq18ugjx7) 40

[Model Summary](#ix0vtzun9q8x) 44

[Recommendations to Management](#nm92vxf0l18) 45

[Actionable Insights](#xuqn5tmbpzqa) 47

[Future Analysis](#dqa76ae7kv) 48

[Bibliography](#knl624oi3o50) 50

[Appendix A: R Code](#nh3zw6z9f9p8) 51

[Appendix B: Additional Visualizations](#m420obrd1t) 75

Introduction

Project Background and Description

This project examined a tele-communications dataset focused on customer retention and churn rates. Churn, or the rate at which customers leave a subscription or contract service, is a key factor in business health, and a high churn rate can indicate a number of issues requiring resolution in order for subscription rates to remain steady. Customer dissatisfaction, the availability of cheaper or better services elsewhere, and failures in customer service can all be origin points for high churn rates; that the ultimate issue could also be a blend of multiple variables only highlights the potential complexity of analysis needed in order to understand customer behavior and ensure long-term subscription stability.

Historically, research on customer retention has focused on tenure, preferring to model average customer life-cycles rather than examine the many reasons why customers may choose to terminate service:

With an interest in managing retention, influencing it with marketing actions, and understanding the effect of retention on the value of the customer base, the focus has primarily been on when customers terminate a relationship—that is, the approach has been to model the time until churn, or the duration of the customer relationship. In this sense, the reason for which customers decide to discard service is irrelevant (Braun and Schweidel 881).

As Braun and Schweidel note, a key reason for the prioritization of tenure over factors leading to churn is the complexity of reasons that may lead a customer to terminate service, many of which may be out of the control of either the company or the individual. Braun and Schweidel categorize factors like “customer relocation, change in personal circumstances unrelated to the service, or even death” as “uncontrollable” with respect to company retention efforts, and suggest that it may not always be in the company’s best interest to invest in strategies to model the complex relationship between these factors and potential customer churn (Braun and Schweidel 882).

This is not to say, however, that modelling churn with respect to observed factors within a given customer database is without value; while average customer tenure is a powerful tool for company decision-making, data on customer habits and demographics can also be a valuable asset in identifying potential areas for improvement in customer service, marketing and outreach, all of which may lead to improved customer retention.

Project Scope and Context for Analysis

The dataset selected for this examination was initially provided by IBM as a sample data set, which was then acquired for the purposes of this study via Kaggle. The data includes 20 variables linked to 7044 unique customer identification numbers. Among the variables are tenure, average monthly and total charges, and demographic information (whether or not the customer has a partner, dependents, or is a senior citizen for example). Unfortunately, the dataset does not include customer location, customer age, or customer-identified reasons for churn (all of which can provide strong insight for modeling future customer behavior). Within these limitations, our project focused on comparing churn with three major variables: average customer tenure, contract type (including whether or not internet service was used), and demographics (including whether the customer was a senior citizen, has a partner and/or dependents). These variables were selected after an initial examination of both the data and existing literature on the subject, which indicated these as the most likely predictors of churn available within the given dataset.

Our examination prioritized tenure for a number of reasons, the first being the key importance of customer retention to overall business health. Acquiring new customers is significantly more expensive than retaining an existing customer; according to Amy Gallo, writing for the Harvard Business Review in 2014, the average cost of acquiring a new customer (depending on the industry) is “anywhere from five to 25 times more expensive than retaining an existing one” (Gallo 2014). Beyond the impact of the cost of retention vs acquisition, tenure is also a significant factor in ongoing business growth. As Fred Reichheld and Christine Detrick note, “a 5% increase in customer retention [in the financial services industry] produces more than a 25% increase in profit” (Reichheld and Detrick 2003).

Beyond our focus on tenure, we examined the relationship between churn and selected demographic factors in order to identify potential areas for customer retention efforts. While the dataset is limited in terms of the strong predictors of customer churn typically used by analysts (factors including customer location, customer-identified reasons for churn, and customer age are not provided, for example), we used contract type, average monthly and total costs, senior citizen status, and household demographics to model potential areas within the customer base that might necessitate attention with respect to customer retention and marketing efforts.

Business Questions

Based on our preliminary examination of the dataset, we identified four key questions to address during the course of our investigation. These focal questions aim to diagnose potential issues related to customer retention, and will form the basis of our research on churn rates going forward:

* What demographics (if any) are most likely to churn?
* Are there any significant trends in when churned customer choose to leave?
* Does any contract type perform significantly better compared to other options?
* Which customers should be provided additional resources in order to increase retention?

While there are many potential variables that could be the cause of churn, our dataset has a heavy focus on demographic variables, including whether the customer has any dependents, if the customer as a senior citizen, and if the customer has a partner. Exploring these demographic variables may identify customer groups with a high susceptibility to churn; this information can then be used to develop new business approaches to offset high churn rates.

It is important to note that in most cases, a customer will churn at some point during their contract or subscription; identifying the average length of tenure at which churn happens, therefore, is of key importance to maintaining a stable customer base and revenue stream. If we were to discover any significant trends in when a large proportion of our customers leave, additional resources could be spent on customers approaching this tenure threshold in order to increase retention.

Another major variable requiring further evaluation is the performance of contract types relative to each other. If a specific contract type performs either significantly better or significantly worse than the other contract types, the company may wish to pull customers away from the less desirable contract type and diagnose the issues customers are having with this subscription.

Finally, after determining which general segments of the customer base have a higher likelihood of churning, we would want to see if we can develop a personalized model to identify which specific individuals are likely to churn. With this insight, we can then provide additional customer support or target these customers with promotional prices or events.

Data Overview

Data Source

Data acquisition involves collecting and adding to the data holdings. During our data acquisition process we investigated ~~into~~ various interesting datasets from several sources and selected one with most analytical potential. The selected dataset was provided by IBM Sample Data sets, and acquired for this study via following web link: <https://www.kaggle.com/blastchar/telco-customer-churn>.

Data Quality

In our first assessments of the dataset, almost the entire dataset was of high quality and easily understood. There were 11 data points missing from the column “Total Charges”. Further investigation determined that these missing values were due to these customer instances having a tenure of 0 months. Because these customers had not been charged for any amount during their subscription, there could not be any value provided in total charges.

Please note that the data used for this report only captured a single month of customer information. Conclusions drawn from a single month’s worth of data will intrinsically have many limitations, and hidden variables, including month-to-month variation, world events, and factors beyond either customer or company control which cannot be fully accounted for. If we were to do a more in-depth analysis, multiple month’s worth of data, if not years, would be more robust in describing the trends that we attempt to investigate here.

Data Selection

Data selection is defined as the process of determining the appropriate data type and source, as well as suitable instruments to collect data. The dataset in question contains primarily quantitative data described in numerical values, for example customer tenure in months, monthly charges, total charges, and dichotomous data with "yes/no" values, for example, gender and dependents, which help us to measure which proportion of customers has certain demographic attributes.

The customer dataset contains both types of data: discrete and continuous. Continuous variables consist of measured data such as monthly charges and total charges. The majority of variables in our dataset contain discrete data that helps us to identify demographic attributes. These discrete variables include factors like age (senior citizen or not), gender (male or female), dependents (whether customer has dependents or not), marital status (whether the customer has a partner) and some service-related attributes, including type of service (phone, internet) and additional related services. Analysis of demographic and service data help us to identify certain patterns related to customer churn rate.

In order to better answer our business questions, we decided to analyze the entire dataset of customers, with an additional focus on the subset of customers subscribed to internet services. This provided us with more information on different customer segments, allowing for comparison of the demographic behaviors, service types and contract patterns that can affect customer retention rates.

Analysis focused on identifying the ways in which various demographic attributes, types of contracts and services, or independent variables, affect customer churn rate, which in our analysis represents the dependent variable.

Our descriptive statistics analysis focuses primarily on demographics data, type of contract, and type of service limited to top level subscriptions, phone and internet. The following columns were excluded from our analysis:

* MultipleLines
* OnlineSecurity
* OnlineBackup
* DeviceProtection
* TechSupport
* StreamingTV
* StreamingMovies

Data mining analysis takes into consideration all available independent variables.

According to Techopedia, data cleansing is the process of altering data in a given storage resource to make sure that it is accurate and correct. (Data cleansing, n.d., [para.](https://www.techopedia.com/definition/1174/data-cleansing)1). During our initial data quality assessment, we found 11 observations with missing values (NAs) in TotalCharges column. This was potentially due to data unavailability for newly subscribed customers with tenure=0.

Our data cleansing process consisted of subsetting observations with tenure higher than zero. Next, we created a subset of customers with internet services by selecting all observations where the value of Internet Services value was not equal to “Yes”. We then removed the customerID column and unused levels of internet service from subset of customers with internet using the droplevels() function.

During our next step we created a subset of customers subscribed exclusively to phone services by selecting all observations in which the value of InternetServices is equal to “No”. We then removed redundant columns containing the MultiLines variable using droplevels() function.

Data Dictionary

A data dictionary was used for this dataset to keep track of the variables. For each variable in the dataset, the data dictionary contains the index, column name, and the meaning of the variable. The data dictionary is provided in Table 1.

Table 1: Data Dictionary

|  |  |  |
| --- | --- | --- |
| Index | Column Name | Definition |
| 1 | customerID | Customer ID |
| 2 | gender | Customer’s gender: Male, Female |
| 3 | SeniorCitizen | Customer’s senior citizen status, numeric value: 1 - yes, 0 - no |
| 4 | Partner | Identifies whether the customer has a partner or not: Yes, No |
| 5 | Dependents | Identifies whether the customer has dependents or not: Yes, No |
| 6 | tenure | Number of months the customer has stayed with the company |
| 7 | PhoneService | Identifies if customer uses phone services: Yes, No |
| 8 | MultipleLines | Identifies whether the customer has multiple lines or not: Yes, No, No phone service |
| 9 | InternetService | Identifies if customer uses internet services and type of service: DSL, Fiber optics, No |
| 10 | OnlineSecurity | Identifies whether the customer has online security or not: Yes, No, No internet service |
| 11 | OnlineBackup | Identifies whether the customer has online backup or not: Yes, No, No internet service |
| 12 | DeviceProtection | Identifies whether the customer has device protection or not: Yes, No, No internet service |
| 13 | TechSupport | Identifies whether the customer has tech support or not: Yes, No, No internet service |
| 14 | StreamingTV | Identifies whether the customer has streaming TV or not: Yes, No, No internet service |
| 15 | StreamingMovies | Identifies whether the customer has streaming movies or not: Yes, No, No internet service |
| 16 | Contract | Type of contract: One year, Two year, Month-to-month |
| 17 | Paperless Billing | Identifies billing type: Yes, No |
| 18 | PaymentMethod | Identifies type of payment: Electronic check, Mailed check, Bank transfer (automatic), Credit card (automatic) |
| 19 | MonthlyCharges | The amount charged to the customer monthly |
| 20 | TotalCharges | The total amount charged to the customer |
| 21 | Churn | Identifies whether the customer churned or not: Yes, No |

Data Structure

Data structure is defined as a specific form of organizing and storing data. R supports five basic types of data structure: vector, matrix, list, data frame and factor. Our dataset is structured as a data frame where each component is of the same length. It has 7043 observations and 21 variables. The structure of the dataset is provided in Table 2.

Table 2: Structure of Data Frame

|  |  |  |
| --- | --- | --- |
| Variable | Type | Example Data |
| customerID | Character | "7590-VHVEG", "5575-GNVDE" |
| gender | Character | "Female" "Male" |
| SeniorCitizen | Numeric | 0, 1 |
| Partner | Character | "Yes" "No" |
| Dependents | Character | "Yes" "No" |
| tenure | Numeric | 1, 34, 2, 45 |
| PhoneService | Character | "Yes" "No" |
| MultipleLines | Character | "No phone service" "No" “Yes” |
| InternetService | Character | “DSL”, “Fiber optics”, “No” |
| OnlineSecurity | Character | "Yes" "No" |
| OnlineBackup | Character | "Yes" "No" |
| OnlineSecurity | Character | "Yes" "No" |
| OnlineBackup | Character | "Yes" "No" |
| DeviceProtection | Character | "Yes" "No" |
| TechSupport | Character | "Yes" "No" |
| StreamingTV | Character | "Yes" "No" |
| StreamingMovies | Character | "Yes" "No" |
| Contract | Character | "Month-to-month" "One year" "Two year" |
| PaperlessBilling | Character | "Yes" "No" |
| PaymentMethod | Character | "Electronic check" "Mailed check" "Mailed check" "Bank transfer (automatic)" “Credit card (automatic)” |
| MonthlyCharges | Numeric | 29.9, 57, 53.9, 42.3 |
| TotalCharges | Numeric | 29.9, 1889.5, 108.2, 1840.8 |
| Churn | Character | "Yes" "No" |

Data Findings

Our initial data findings are based on overall quality of data, initial data cleansing, transformation and preparing subsets of data. During our first assessment we determined that the entire dataset contained high quality data, with only 11 missing values that are potentially absent due to a lack of data for customers recently subscribed (i.e. less than one year tenure) at the time of data collection. We also noticed that column names were easily understood and did not require renaming. It came to our attention that dataset contains only one month of captured data, which might not take into consideration certain trends and events in the industry.

Our next step was to identify the type of data for each variable, discrete or continuous, as discrete variables can be a detrimental factor in further statistical analysis. We also investigated the dataset structure using str() function, removed rows with zero tenure values, and created subsets of customers subscribed to phone and internet services.

Statistical Analysis

Summary Statistics

Summary statistics are used to synthesize a large set of observations and communicate the information in a simple manner. It was useful to find measures of central tendency such as mean, median, mode and data dispersion such as standard deviation and spread to retrieve first meaningful information about our dataset. We used summary statistics to summarize our dataset’s numerical variables, with results shown in Table 3.

Table 3: Summary Statistics For Continuous Data

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Tenure** | **MonthlyCharges** | **TotalCharges** |
| Mean | 32.42179 | 64.79821 | 2283.3 |
| Median | 29 | 70.35 | 1397.475 |
| Minimum | 1 | 18.25 | 18.8 |
| Maximum | 72 | 118.75 | 8684.8 |
| Standard Deviation | 24.54526 | 30.08597 | 2266.771 |
| Quantile (0.05) | 1 | 19.65 | 49.605 |
| Quantile (0.95) | 72 | 107.4225 | 6923.59 |
| Skewness | 0.2376801 | -0.2220555 | 0.9614374 |

Going further, we summarized churn rate and average tenure for subsets of customers with and without internet services (shown in Table 4).

Table 4: Average Churn for Phone and Internet Customers

|  |  |  |
| --- | --- | --- |
|  | Churn rate, % | Average tenure, months |
| Phone Subscribers | 7.43 | 31 |
| Internet Subscribers | 31.86 | 33 |

Our initial findings have shown that average customer tenure is 32 months, and there is a drastic difference in churn rates for the subset of customer subscribed to phone services and internet subscribers. On the other hand, average tenure for the internet subscribers is only slightly higher than for the customers subscribed to phone services. This raises more questions about reasons for such clear difference. A demographics summary of churned customers for both subsets is provided in Table 5.

Table 5: Demographics Churn Count Data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Churn by Gender | | Churn by Age | |
|  | Male | Female | Senior Citizen | Not Senior Citizen |
| Phone Subscribers | 57 | 56 | 5 | 108 |
| Internet Subscribers | 873 | 883 | 471 | 1285 |

As noted in the above table, the churn rate for female customers is equivalent to male, but churn rate for younger customers is drastically higher than for senior citizens, meaning that senior citizens have better retention rate in both subsets.

Descriptive Statistics

Our early investigations into this data were mostly focused on analyzing the nature of churn in the context of different categorical variables. To get an understanding of the degree of different customers churning we also included tenure in these analyses. We considered that the shorter tenure of a churned customer, the worse it would be for the company.

These statistics would be invaluable to a telecommunications company, providing guidance on subsets of customers requiring more or less attention, on as well as being helpful in knowing how long customers have been subscribed before they leave.

Our first and most basic finding was that our overall churn rate is 26.54%. According to Strategy Analytics, a leading consulting firm that uses statistics to assist companies in business decisions, the average churn rate among the top telecommunication companies in the first quarter of 2018 was 1.93% (Dano). By comparison, the churn rate for our company is significantly worse. This alone would be a significant finding, but it does not provide any actionable insight without more context; the limitations of our dataset, in particular the lack of data beyond one month, prevent further dependable analysis in this instance. Moving forward we have attempted to find information that leads to actionable insights.

Continuing to investigate churn rates overall, we looked into how churn is related to tenure. It is important to note however that as the tenure of a non-churned customer is only their current tenure, it is important to keep in mind that non-churned customers are continuously gaining tenure past this month’s dataset.

Below, a histogram shows the number of customers that churned and did not churn split up by their tenure in months. Clearly, a significant portion of the churned customers left the service within the first year of their subscription, with a majority of those leaving within just the first few months. This trend holds true across different demographics, including both customers with or without dependents and customers with or without partners. See Appendix 1 for the plots showing this trend.

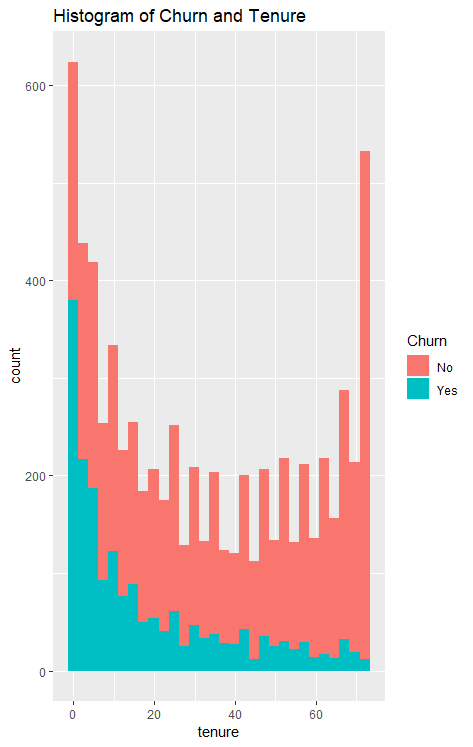


Figure 1: Churn over Tenure

Based on this statistic, once a customer has been subscribed for at least a year, the rate of churn drops significantly. Reasons for this could be differences in service from other companies that have a steep learning curve or possibly a retention program that only pays dividends for longer term subscribers.

This leads to two points for the company. One, more resources should be spent on keeping customers subscribed past the first few months and the first year. And consequently, resources could be taken away from initiatives to keep longer term customers. Although, since this part of the business plan seems to be working well, changing the design of it should be taken slowly if at all.

Demographics

After getting a better idea of how churn is described by itself, we wanted to investigate how different demographics were churning and if we could use this to improve our overall churn.

Gender

The first analysis by a demographic variable was to look at churn by gender. We wanted to see if male and female customers churned differently. Unsurprisingly, male and female customers churn almost identically. Figure 2 displays the count data for both male and female customer, with the color characterizing whether the customer had churned or been retained this month.

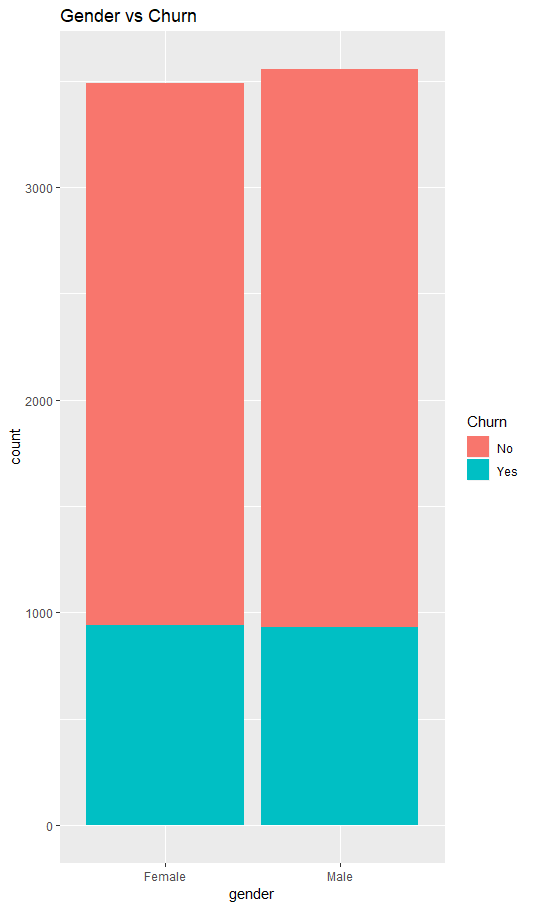


Figure 2: Churn by Gender

With churn rates of 26.92% for female customers and 26.16% for male customers, it is clear that these two demographics have no significant churn difference between them. Even though there are no actionable insights here, it is still valuable to know that gender has little effect on churn. The company can now move forward knowing that no additional resources need to be delegated towards either demographic specifically.

Customers with Dependents

Diving deeper we investigated how customers with dependents vs those without dependents compared to each other. We found that customers with dependents had a churn rate of 15.45% and those without had a churn rate of 31.28%. This can also be seen in Figure 3.

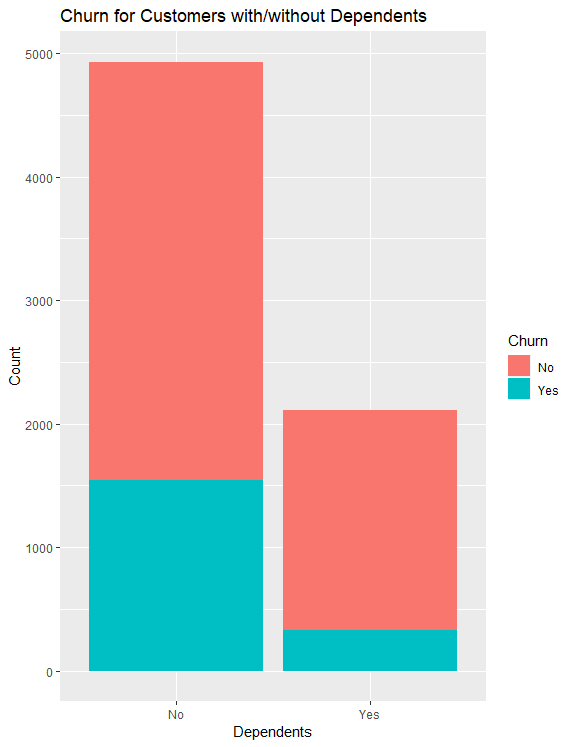


Figure 3: Customers With Dependents vs. Customers Without Dependents

Clearly these populations have different relationships to churn. These findings would indicate that although the customer base without dependents has more than double the number than those with dependents, over a third of them left the service within the last month.

Not only are they leaving, but as seen in Figure 4, customers without dependents have shorter tenures than those with dependents, both those who have churned and those that have not. Average tenure for customers with dependents was 38.7 months and was 29.8 months for those without.

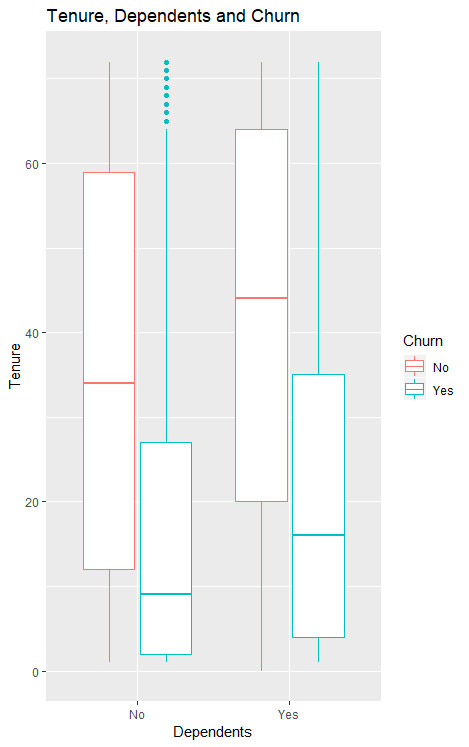


Figure 4: Boxplot for Tenure, Dependents, and Churn

In a business sense we would advise that additional resources should be used to investigate why customers without dependents are churning much more. Possible reasons for this significant difference could be that those customers with dependents are much less likely to move their household, and therefore more likely keep the same telecom provider, or that those customers without dependents have less consistent pay, which causes them to be more likely to cancel or change service providers.

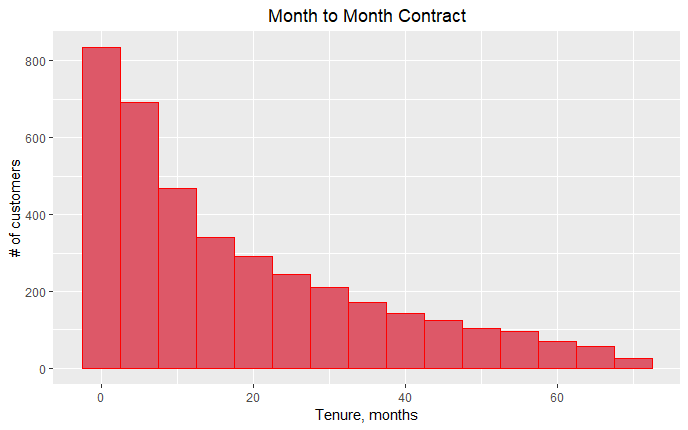
It is interesting to notice though that even though the mean tenures for customers without dependents are lower than those with dependents, three of the four different categories seen in Figure 4 have the same maximum tenure. This would mean that there are customers that do stay for extended periods of time from all categories and that from a business perspective the model is working correctly for those specific customers.

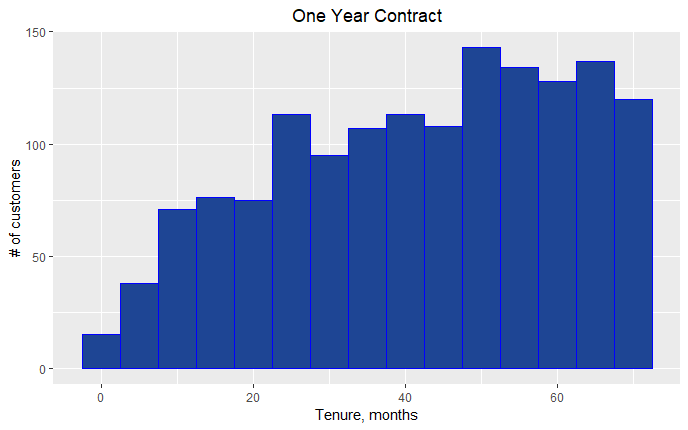
We do need to keep in mind that this maximum tenure (72 months or 6 years) could very easily be a limitation of the dataset. When the dataset was created, only those customers that subscribed within the last 6 years were included or the subscription was only created 6 years go and this dataset was created by the company for an annual review of their methods and business models.

Contract Analysis

Churn rate by the type of contract histograms for both subsets demonstrate higher churn rates for customers on Month-to-month service agreement. Interestingly, customers subscribed to phone services had much lower churn rate than internet customers, meaning that there is an obvious problem with internet services.

To better understand churn rates by type of contract, we created histograms representing tenure for customers for each type of contract (Figure 5).





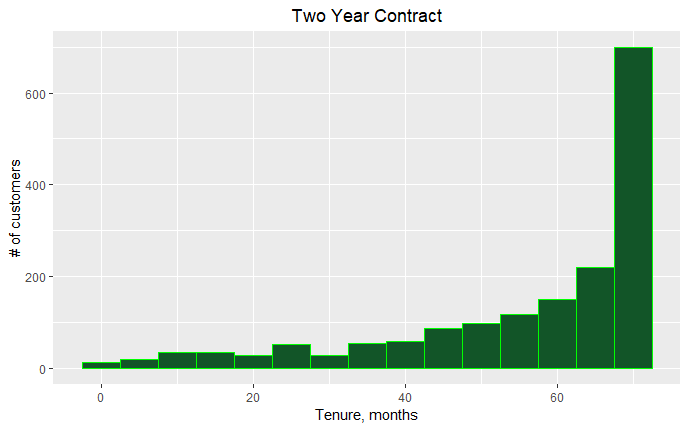


Figure 5: Tenure by Contract Type

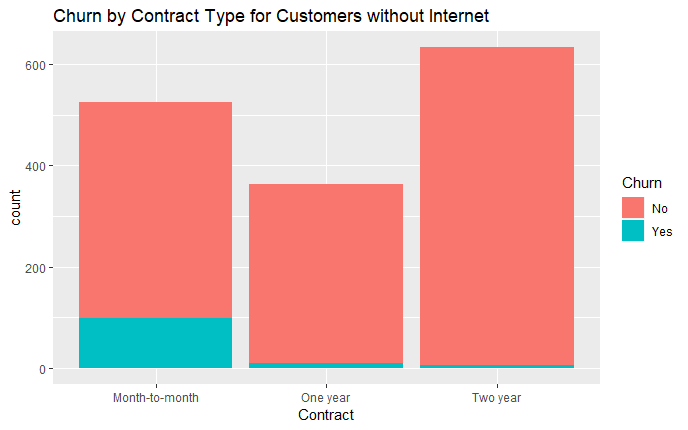
We noticed how tenure changes and becomes longer for customers on a two year agreement. This shows that customers with longer contract are more loyal to the company and tend to stay with it for a longer period of time. If they're willing to make such a big monetary commitment, they're likely more invested in using the company’s services and less likely to churn out after a short period of time.

Going further, we investigated churn rates by type of contract for subsets of customers subscribed to phone services and internet services.

Table 6. Churn rates by contract type

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Churn rate by contract type, % | | | |
|  | Month to month | One Year | Two Year | Total |
| Internet services | 28.20 | 2.85 | 0.78 | 31.83 |
| Phone services | 6.49 | 0.59 | 0.33 | 7.41 |

Our findings are illustrated in Figure 6:



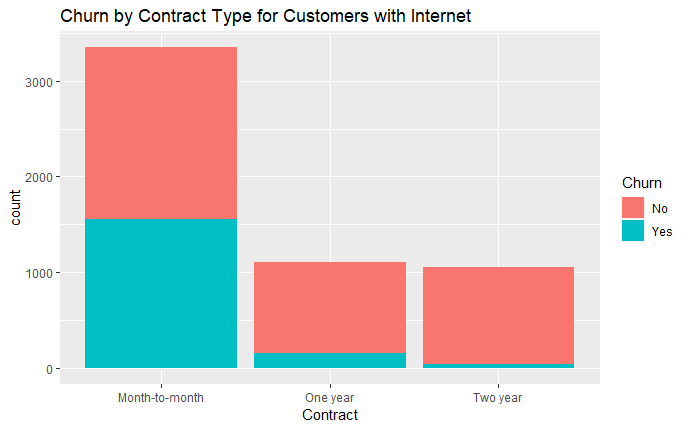


Figure 6: Churn by Contract Type

Churn rates by the type of contract for both subsets demonstrate higher churn rates for customers on Month-to-month service agreement. Interestingly, customers subscribed to phone services had much lower churn rate than internet customers, meaning that there is an obvious problem with internet services.

Inferential Statistics

Demographics Analysis

Customers with Dependents

As discussed above, those customers with dependents churned at a much lower rate and this is confirmed by the following chi squared and t test. We set up this chi squared and t test with a null hypothesis that no significant difference exists between the churn rates of customers with dependents and those without dependents and that any difference that does exist is completely due to random variation. The alternative hypothesis was that a significant difference does exist between the churn rates of customers with dependents and those without. We decided to implement an alpha of .05.

As you can see below the test statistic came to be <2.2x10-16 which is most assuredly small enough to say that the difference between these two populations, customers with and without dependents, is significant and is not caused by randomness.

Table 7: Observed and Expected Dependent (Chi- square test)

|  |  |  |
| --- | --- | --- |
| Chi Squared - Observed | Customer was not churned | Customer was churned |
| Does not have dependents | 3390 | 1543 |
| Has Dependents | 1784 | 326 |

|  |  |  |
| --- | --- | --- |
| Chi Squared - Expected | Customer was not churned | Customer was churned |
| Does not have dependents | 3623.93 | 1309.0696 |
| Has Dependents | 1550.07 | 559.9304 |

T-Test for significance

|  |  |
| --- | --- |
| P Value | <2.2x10-16 |

Senior Citizens

Another customer segment we wanted to analyze further was senior citizens. What we were most interested in was whether senior citizens had a statistically different churn rate than non-senior citizens.

Our initial calculations determined that churn rate for senior citizens was 41.68%, while the churn rate for the rest of the customers was 23.65%. Although it may appear that seniors had a significantly higher churn rate than the rest of the customer base, we wanted to conduct a hypothesis test to be sure.

For the hypothesis test, our null hypothesis was that there was no difference in the churn of senior citizens when compared to the rest of the customer base. We had also set the alpha level at 0.95, so that we would be confident that the two customer groups behave differently 95% of the time. The t-test function for this hypothesis test returned a p-value of < 2.2e-16, meaning that we had to reject the null hypothesis. The count data for senior citizens colored by the churn is compared to the count and churn of the rest of the customers in Figure 7.

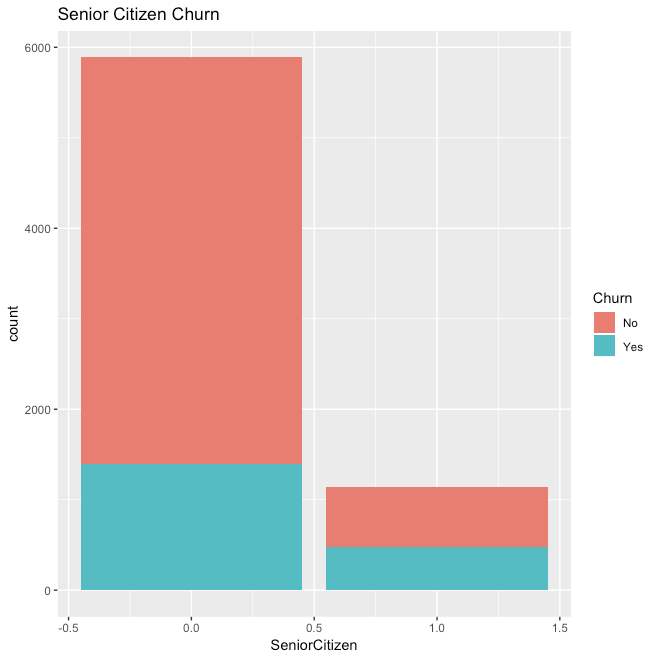


Figure 7: Churn Rate for Senior Citizens

Contract Analysis

One major categorical variable that we believed would be important in determining whether a customer would churn was the contract type the customer was using. While we cannot use a t-test to determine whether one contract has a higher rate of churn than another, we decided it would be appropriate to conduct a chi-square test to see if there was some sort of a relationship between contract type and churn. Table 8 shows the count data that was observed for the for the dataset as well as the expected value.

Table 8: Observed Churn (top) vs. Expected Churn (bottom) by Contract Type

|  |  |  |  |
| --- | --- | --- | --- |
|  | Month-to-month | One year | Two year |
| No Churn | 2220 | 1306 | 1637 |
| Churn | 1655 | 166 | 48 |

|  |  |  |  |
| --- | --- | --- | --- |
|  | Month-to-month | One year | Two year |
| No Churn | 2845.08 | 1080.77 | 1237.15 |
| Churn | 1029.92 | 391.24 | 447.85 |

While it is quite evident that the month-to-month contract type has a higher churn rate than expected, and the other contracts have a lower churn than expected, we still cannot tell the relationship between contract type and churn. However, the chi-square test for independence between contract type and churn returned a p-value of < 2.2e-16, which would mean that contract type and churn are not independent.

Statistical Analysis Overview

While the descriptive and inferential statistics analyzed above cannot provide us with insights on an individual level, we are able to infer general trends among our customers. These general insights include:

* Gender is not a significant factor in whether a customer will churn
* Customers generally churn within the first year of their subscription
* The churn rate for senior citizens is higher than the rest of the customers
* The relationship between churn and contract type is not independent

With this information we can see that marketing campaigns targeting a specific gender would likely not improve customer churn, additional resources should be spent on customers early in their tenure with the service, and and more research should be conducted on why senior citizens have a higher churn rate. Is it due to involuntary circumstances, such as moving to retirement homes, or is the service too expensive for them. Further analysis on the individual level will be conducted in the linear modeling and data mining sections.

Linear Modeling

Linear modeling is primarily used when attempting to predict a continuous candidate variable. Within our telecom dataset, we determined that the most appropriate candidate variable to estimate using linear modeling was tenure. Being able to model tenure would be beneficial to the company because it would allow the telecom company to predict how long a customer will remain subscribed to the service with only very basic information about the customer. Because we would like to estimate total tenure of a customer, we need to account for the fact that we do not know the full tenure of a customer unless they have churned (no longer a customer). For this reason, the linear modeling analysis will be conducted only on a subset of customers that have churned. After attempting to model the tenure of all customers that have churned, we will then attempt to model tenure for the subset of churned customers that had an internet service. This subset was chosen because there are many categorical variables that are dependent on the the customer having an internet service.

All Customers

Typically, linear modeling works best when there is a correlation between two continuous variables. In this dataset, the only variables that were continuous and could be properly evaluated for a linear relationship were average monthly charges and tenure. We decided that total charges were not applicable for linear modeling because total charges is a combined variable calculated by multiplying tenure with average monthly charges.

The first step in evaluating the relationship between tenure and average monthly charges is calculating the correlation between the two variables. In evaluating all customers that had churned, we discovered that there was a correlation of 0.40 between average monthly charges and tenure. While this shows that there is a positive relationship between tenure and monthly charges, it is not a strong relationship. This weak relationship between tenure and average monthly charges becomes even more evident when fitting a line through the average monthly charges versus tenure scatter plot as shown in Figure 8. The R-squared for this model was determined to be 0.161, which could have also been determined by squaring the value we had found for the correlation.

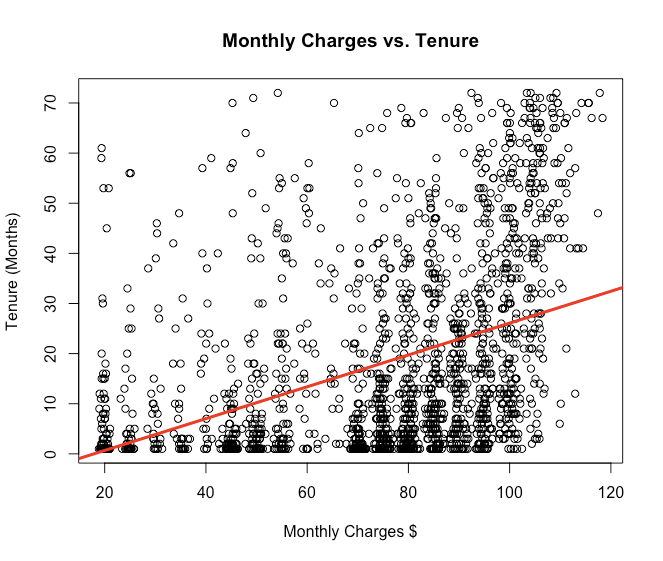


Figure 8: Best Fit Line Monthly Charges vs. Tenure

While the simple linear regression may not be appropriate for modeling tenure, we may be able to create a better model by including all categorical variables in addition to our monthly charges.

The resulting linear model did improve the adjusted R-squared for the model to 0.5684. This value means that all the variables that we had included in our model account for 56.84% of the variation in tenure. This value would also mean that 43.16% of the variation in tenure is due to randomness unknown to our dataset. There also appears to be a trend in the residuals of our model, as seen in Figure 9. This positive correlation in residuals would mean that a linear model is not appropriate for estimating tenure for this dataset.

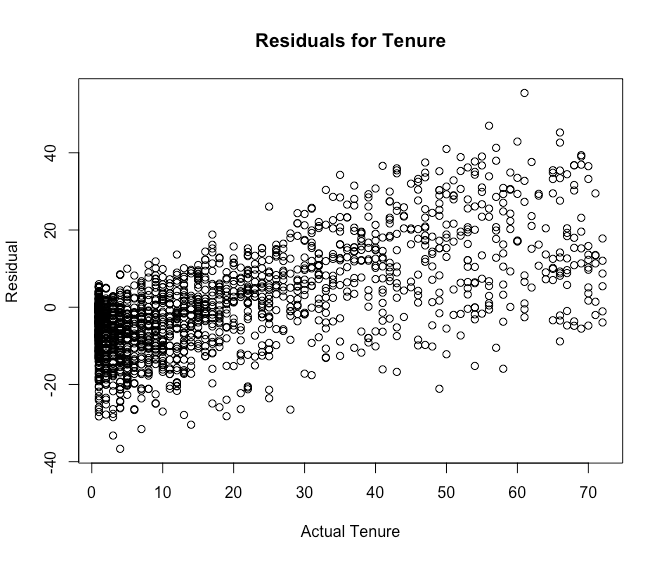


Figure 9: Residual Plot for Tenure

Internet Customers

As we had previously calculated for all customers, the correlation between tenure and average monthly charges for churned internet customers was 0.41, nearly the exact same as the correlation for all customers.

In addition to calculating the correlation between tenure and average monthly charges, we created a linear model using all the variables provided by the dataset for churned customers that had internet. This model performed slightly worse than the model for all churned customers, as it had an adjusted R-squared of 0.5646. Once again, this means that our data can only explain 56.46% of the variation in actual tenure of the customer. Under these conditions, we determined that, with the data we have been provided, it is not appropriate to model tenure for any subset of customers in our dataset without collecting more data.

Data Mining

Unlike tenure, churn is not a continuous variable represented by a number; it a binary, categorical variable that tells us whether a customer was retained, or if they had cancelled their subscription. Because churn is so critical to our business model, being able to predict the likeliness of a current customer leaving would allow the company to allocate resources more efficiently, and potentially retain customers that would have left otherwise.

To model churn rate, we will utilize the supervised learning algorithms of support vector machines (SVM), naive bayes, and decision trees to determine the probability of a customer churning. While it is important to have an accurate model, it is more important that we limit Type II Error, or the false negative rate. We determined that we want to prioritize type II error because it will hurt more to have a customer churn that we thought we would retain than it would to spend extra resources on keeping a customer that was never going to churn. We will be able to adjust the classifications our model makes by creating a probability threshold for classification. Classification will be determined in instances in which the model outputs a probability greater than the specified threshold.

Because of our prioritization of type II error, we will want to calculate the precision (fraction of our positive predictions that are true) and the recall (fraction of positive results that were predicted as true) at multiple probability thresholds, hopefully finding a probability threshold that prioritizes recall, but maintains enough precision to not lose too much accuracy. We will determine the probability threshold that best balances recall and precision by also calculating the F1 Score at each probability threshold. The F1 score is a metric that equally weighs the recall and precision to output a value between 0 and 1. For each model, we will utilize the probability threshold that has the highest F1 score as our final model for that algorithm.

To train the models with the three algorithms, we created a randomly selected training set and a randomly selected test set. The training set will be ⅔’s of the entire dataset, while the test set will be the remaining rows not selected. The algorithm will utilize the training data to learn the parameters that best predicts whether a customer has churned or not, and we will then use the model to output the probability a customer has churning from our test set. We will then loop through a sequence of probability thresholds, and we will calculate the accuracy, precision, recall, and f1 score of the model on the test data at the given probability threshold. We will also plot the calculated model metrics at the given probability threshold and choose the threshold for our model with the highest F1 Score. The training set and test set for each of the models will remain the same for each of the learning algorithms.

In addition to the models created for all customers, we will also train a model to predict churn among customers that have internet as a part of their service. We will follow the steps outlined above for these models as well.

Support Vector Machines

All Customers

A model was created using the SVM machine learning algorithm to predict the probability of a customer churning. The entire set of data had 7032 observations and 19 variables. A training set was randomly selected from the initial data set to be ⅔ the size of the entire data set. The remaining ⅓ of the data was selected as the test data set. This resulted in two data frames, teledf.train with 4,688 observations and teledf.test with 2,344 observations. The variables used to train the model included:

* gender
* SeniorCitizen
* Partner
* Dependents
* Tenure
* PhoneService
* MultipleLines
* InternetService
* OnlineSecurity
* OnlineBackup
* DeviceProtection
* TechSupport
* StreamingTV
* StreamingMovies
* Contract
* PaperlessBilling
* PaymentMethod
* Monthly Charges

Utilizing the ksvm function from the kernlab library, the SVM model was trained with the parameters as follows:

* Formula = Churn ~ .
* data = teledf.train
* kernel = "rbfdot",
* kpar = "automatic"
* C = 1
* cross = 3
* prob.model = T
* probability = T

The first parameter is the formula we used to predict churn, which utilized all variables except for churn in the teledf.train data frame. The Gaussian kernel was chosen after scoring a higher accuracy on the model compared to the polynomial kernel. We also decided to allow the model to select the parameters for us. The SVM model was built to have the default C regularization parameter of 1 in order to have a decent margin between the classifications, but not lose too much information in the process. We also utilized 3 folds for the k-fold cross-validation in training the model. This allows us to keep the model from overfitting during the training process. Additionally we wanted the model to output probabilities that the customer will churn so we can change the probability threshold for predicting churn.

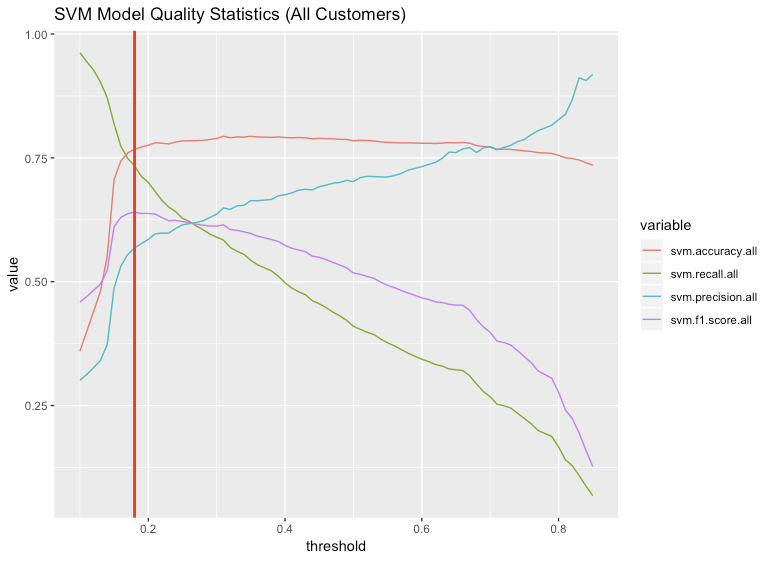
When predicting the churn of customers in the test set, we see that based on the standard probability threshold of 0.5, the SVM model attains an accuracy on the test set of 78.97%, which is only slightly worse than the accuracy during training (80.30%). While this model may seem to be helpful, an accuracy of 71.8% would be attained if we were to predict that we would retain every customer. A confusion matrix of the initial SVM model is provided in Figure 10.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Predicted | |
|  |  | Retained | Churned |
| Actual | Retained | 1515 | 168 |
| Churned | 326 | 335 |

Figure 10: Confusion Matrix for SVM on All Customers

Because we are looking to limit Type II Error, we also calculated the precision, recall, accuracy, and F1 Score for the SVM model with probability thresholds ranging from 0.1 to 0.85. Using this methodology, we were able to determine a maximum F1 score of 0.64 (out of 1.0) occurs at a probability threshold of 0.18. This means we had the best balance of precision and recall when the probability a customer had churned was greater than or equal to 0.18 as determined by the SVM classifier.

At a probability threshold of 0.18, our SVM model had an accuracy of 76.75%. While the accuracy for the adjusted model is worse than the accuracy of the initial SVM model, the new model will correctly signal a customer is going to churn 73.37% of the time when a customer actually churns. However, when this model predicts that a customer will churn, it is only correct 56.79% of the time. While this is not ideal, it is much better to spend additional resources on customers that were not going to churn than it is to completely ignore customers that were going to churn. The final confusion matrix along with the final model’s metrics, and probability threshold metrics plot are provided in Figure 11.



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | Predicted | |  |  | Metric | Value |
|  |  | Retained | Churned |  |  | Accuracy: | 76.74% |
| Actual | Retained | 1314 | 369 |  |  | Recall: | 73.37% |
| Churned | 176 | 485 |  |  | Precision: | 56.79% |
|  |  |  |  |  |  | F1 Score: | 0.6402 |

Figure 11: SVM Classifier (All Customers)

Internet Customers

A second model was created using the SVM machine learning algorithm to estimate the probability of a customer churning from a subset of customers who have internet as a part of their service. This subset of data had 5512 observations and the same 19 variables listed in the All Customers section. A training set from this data was randomly selected to be ⅔’s of the size of all customers with internet. The remaining ⅓ of the data was selected as the test data set. This resulted in two data frames, internet.train with 3,674 observations and internet.test with 1,838 observations. While the variables had the same names, the factor level of “No internet service” was dropped from the dataset for the variables: OnlineSecurity, OnlineBackup, DeviceProtection, TechSupport, StreamingTV, and StreamingMovies. Once again utilizing the ksvm function from the kernlab library, the SVM model was trained with the same parameters used in the All Customers model.

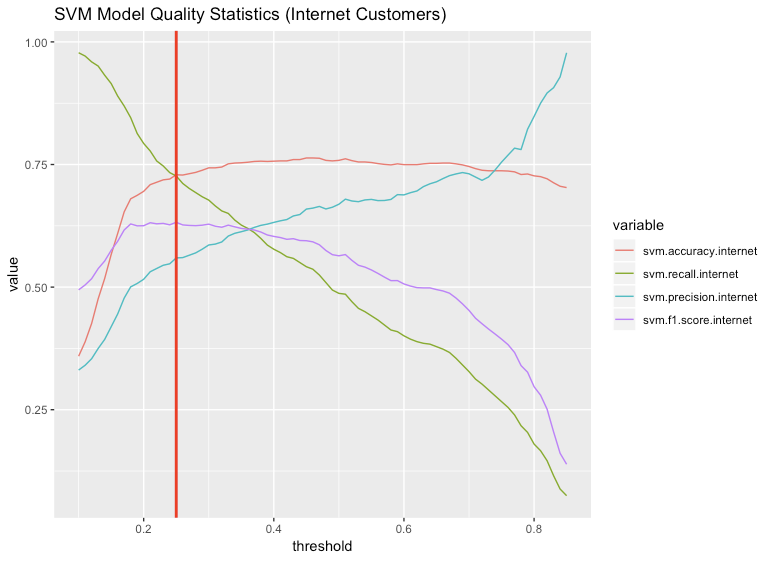
When predicting the churn of customers in the test set, we see that based on a probability threshold of 0.5, the SVM model attains an accuracy on the test set of 76.28%, with this being only slightly less than the cross-validation accuracy during training (76.73%). Once again, this model only slightly improves upon the accuracy of a null hypothesis for each of the instances (68.0% accurate). A confusion matrix of the initial SVM model predicting churn for internet customers is provided in Figure 12.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Predicted | |
|  |  | Retained | Churned |
| Actual | Retained | 1093 | 156 |
| Churned | 280 | 309 |

Figure 12: Confusion Matrix for SVM on Internet Customers

Again, we will look to limit Type II Error by calculating the precision, recall, accuracy, and F1 Score for the SVM model at probability thresholds ranging from 0.1 to 0.85 with the probabilities provided by the SVM model. This process determined that the maximum F1 score of 0.63 (out of 1.0) was achieved at a probability threshold of 0.25. This means we had the best balance of precision and recall when the probability provided by the SVM model was greater than or equal to 0.25.

At a probability threshold of 0.25, the SVM model had an accuracy of 72.91%. The accuracy is worse than the accuracy of the SVM model with a probability threshold of 0.5, but our model will correctly signal a customer is going to churn 72.67% of the time when a customer actually churns. However, when this model predicts that a customer will churn, it is only correct 55.94% of the time. Once again, this is not ideal, but we will be better off utilizing our money to retain customers that would have churned otherwise than attempting to gain new customers. The confusion matrix along, model metrics, and metrics plot are provided in Figure 13.



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | Predicted | |  |  | Metric | Value |
|  |  | Retained | Churned |  |  | Accuracy: | 72.91% |
| Actual | Retained | 1314 | 369 |  |  | Recall: | 72.67% |
| Churned | 176 | 485 |  |  | Precision: | 55.95% |
|  |  |  |  |  |  | F1 Score: | 0.6322 |

Figure 13: SVM Classifier (Internet Customers)

Overall, the model predicting churn for internet customers performs slightly worse in all aspects than the model predicting churn for all customers. However, the performance of the specialized model created for internet customers should be compared to the more generalized model for all customers to truly evaluate the differences between the two models. This is especially important because the average bill for internet customers is about $10 greater than the bill for all customers.

Naive Bayes

All Customers

A second model was created using the naive Bayes machine learning algorithm to predict the probability of a customer churning. The same train and test sets previously used in the SVM classifier were used to train and test the naive bayes model.

Utilizing the naiveBayes function from the e1071 library, the naive Bayes model was trained with the parameters as follows:

* Formula = Churn ~ .
* Data = teledf.train
* Probability = T

The first parameter is the formula that was used to estimate churn for the customers. Churn was used as the dependent variable, and the rest of the variables were chosen as the independent variables. The second parameter shows that we were using the data provided by the teledf.train dataset to train the model. The last parameter tells the algorithm to calculate the probability of customer churn for that instance, so the output will be the probability of a customer churning (according to the model) instead of which class (Churn or No Churn) the model believes is most likely.

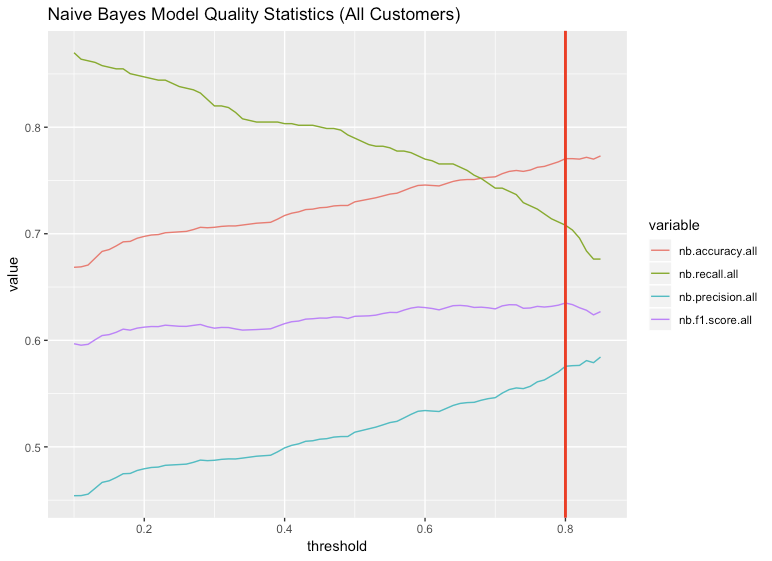
When estimating the churn of customers in the test set, we see that the mode achieves an accuracy of 72.99% at the probability threshold of 0.5. This model performs much worse than the SVM model and is only slightly better than the 71.8% accuracy that would have been attained if we were to predict retention of every customer. A confusion matrix of the initial naive Bayes model is provided in Figure 14.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Predicted | |
|  |  | Retained | Churned |
| Actual | Retained | 1189 | 494 |
| Churned | 139 | 522 |

Figure 14: Confusion Matrix for Naive Bayes on All Customers

Because we are looking to limit Type II Error, we also calculated the precision, recall, accuracy, and F1 Score for the naive bayes classifier with probability thresholds ranging from 0.1 to 0.85. Using this methodology, we were able to determine a maximum F1 score of 0.635 (out of 1.0) could be achieved at the probability threshold of 0.80. This means we had the best balance of precision and recall when the probability a customer had churned was greater than or equal to 0.80.

The probability threshold is much larger than we would expect, especially given our desire to limit false negatives. However, when the model outputs the probability of the customer churn, it is fairly certain that it knows the class. This can be inferred by the relatively high recall given the very high probability threshold. At the probability threshold of 0.80, our naive bayes model had an accuracy of 77.05%, which is an improvement upon the model at the 0.5 probability threshold. We also determined that the model will correctly signal a customer is going to churn 70.80% of the time a customer actually churns. However, when this model predicts that a customer will churn, it is only correct 57.56% of the time. The confusion matrix along, model metrics, and metrics plot for the naive bayes model are provided in Figure 15.



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | Predicted | |  |  | Metric | Value |
|  |  | Retained | Churned |  |  | Accuracy: | 77.05% |
| Actual | Retained | 1338 | 345 |  |  | Recall: | 70.80% |
| Churned | 193 | 468 |  |  | Precision: | 57.56% |
|  |  |  |  |  |  | F1 Score: | 0.6350 |

Figure 15: Naive Bayes Classifier (All Customers)

Internet Customers

A second model was created using the naive bayes machine learning algorithm to predict the probability of a customer churning from a subset of customers who have internet as a part of their service. The same training and test sets utilized by the the SVM classifier was again used to train and test the naive bayes classifier. The naiveBayes function from the e1071 library was again used to train the the model with the same parameters used in the naive bayes model for all customers.

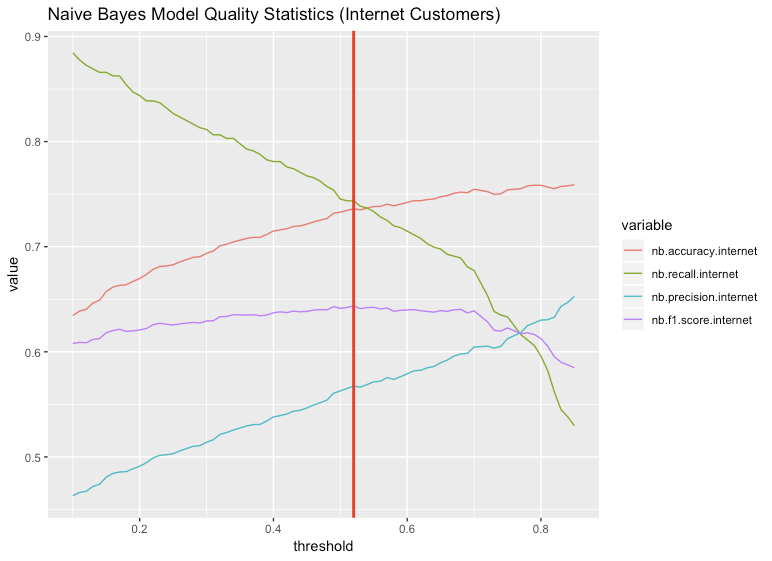
When estimating churn of customers in the test set, we see that the model achieves an accuracy of 73.28% at the probability threshold of 0.5. Once again, this model only slightly improves upon the accuracy a null hypothesis for all instances would generate (68.0%). The confusion matrix of the naive bayes model at the probability threshold of 0.5 for customers with internet is provided in Figure 16.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Predicted | |
|  |  | Retained | Churned |
| Actual | Retained | 908 | 341 |
| Churned | 150 | 439 |

Figure 16: Confusion Matrix for Naive Bayes on Internet Customers

Again, we will look to limit Type II Error by calculating the precision, recall, accuracy, and F1 Score for the SVM model with probability thresholds ranging from 0.1 to 0.85 using the probabilities provided by naive Bayes model on internet customers. This process determined a maximum F1 score of 0.64 (out of 1.0) at a probability threshold of 0.52. This means we had the best balance of precision and recall when the probability provided by the naive bayes model was greater than or equal to 0.52.

At an a probability threshold of 0.52, our naive Bayes model had an accuracy of 73.61%. This model’s accuracy is slightly better than the accuracy of the model with a probability threshold of 0.5, as expected, and our model will correctly signal a customer is going to churn 74.36% of the time a customer actually churns. However, when this model predicts that a customer will churn, it is only correct 56.73% of the time. The confusion matrix along, model metrics, and metrics plot are provided in Figure 17.



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | Predicted | |  |  | Metric | Value |
|  |  | Retained | Churned |  |  | Accuracy: | 73.61% |
| Actual | Retained | 915 | 334 |  |  | Recall: | 74.36% |
| Churned | 151 | 438 |  |  | Precision: | 56.74% |
|  |  |  |  |  |  | F1 Score: | 0.6436 |

Figure 17: Naive Bayes Classifier (Internet Customers)

The naive Bayes model predicting churn for internet customers performs slightly worse than the naive Bayes model for all customers in accuracy and precision, but outperforms the other model in recall and f1 score. While accuracy is great for the model, the increased performance in recall could make this best performing model in classifying churn among any subset of customers so far.

Decision Tree

All Customers

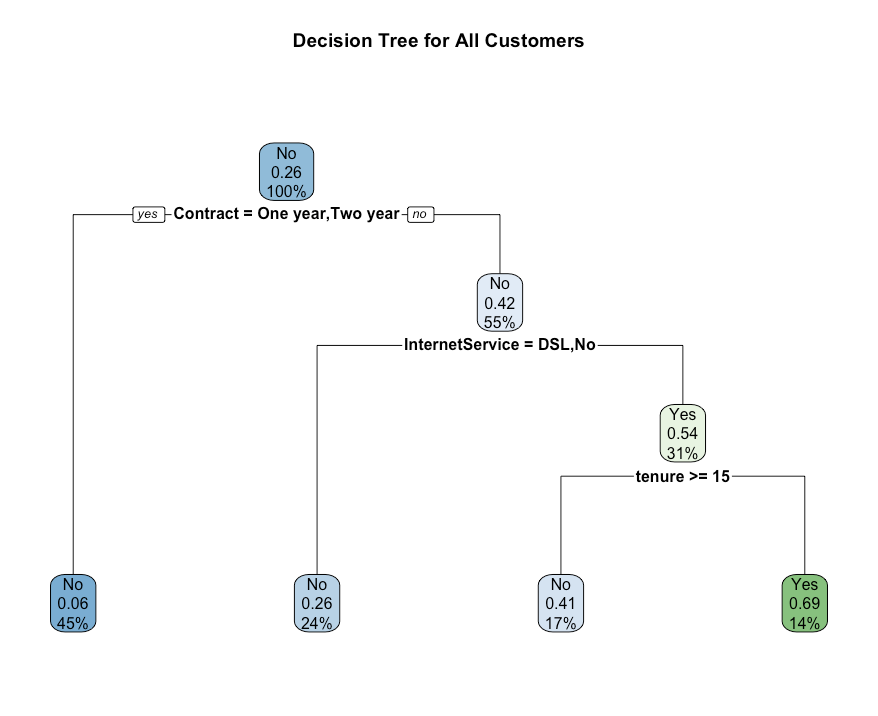
One last model was created to predict whether a customer would churn using the decision tree learning algorithm. This learning algorithm uses the attributes of the customer data set to split up the dataset into separate branches until every instance is in a “leaf” that classifies the the instance. Because the classification is primarily based on categorical variables, and not all variables are chosen by the tree, we will utilize the default probability threshold for the final model.

The decision tree was created by using the rpart function in the rpart library, and was trained with the parameters as follows:

* Formula = Churn ~ .
* Data = Teledf.train
* control = rpart.control(minsplit = 20, cp=0.01)
* method = "class"

The first parameter is the same as we have used for each of the classification models. This formula is telling the algorithm to estimate whether a customer has churned using all of the variables provided in the training data frame. The second parameter is telling the model to utilize all of the rows in the teledf.train dataframe to train the model. The control parameter is where we are inputting some parameters to control the overfitting of the model. We had set minsplit equal to 20, which means a branch must have a minimum of 20 observations in order to attempt a split. We also set the complexity parameter to 0.01, which means any split that does not improve the overall fit of the model by 0.01 is not attempted. The last parameter in the is telling the model we are more interested in the class of the model than the probability of churn the model has detected.

When predicting the churn of customers in the test set, we see that the decision tree model attains an accuracy on the test set of 77.94%. While this model is the most accurate, there is a significant drop off in recall in this model. The decision tree, along with the confusion matrix and the model metrics are provided in Figure 18.



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | Predicted | |  |  | Metric | Value |
|  |  | Retained | Churned |  |  | Accuracy: | 77.94% |
| Actual | Retained | 1589 | 94 |  |  | Recall: | 36.00% |
| Churned | 423 | 238 |  |  | Precision: | 71.69% |
|  |  |  |  |  |  | F1 Score: | 0.4793 |

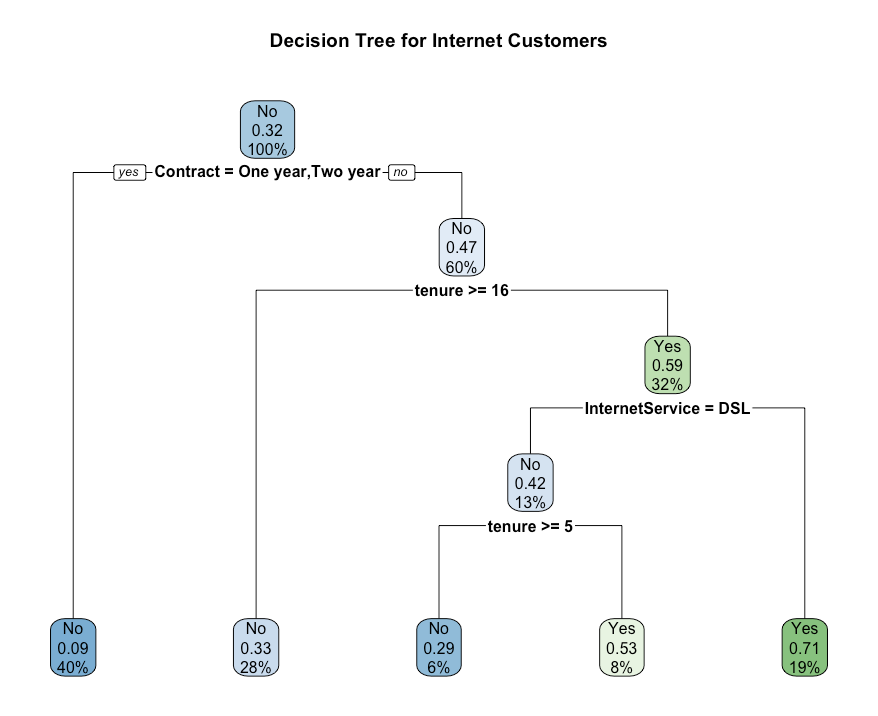
Figure 18: Decision Tree Classifier (All Customers)

As shown in Figure 18, the decision tree classifier is the most accurate, but would be the worst model for our business case. The decision tree model only signals a customer will churn 36% of the time a customer actually churns. However, the decision tree does tell us exactly the variables and the levels it believes are the most important in determining customer churn. For all customers, the contract type, internet service, and tenure provides the most useful information.

Internet Customers

As we had done for the previous learning algorithms, we created a second decision tree classifier for the subset of customers that have internet as a part of their service. The training and test sets used for this model are the same as the training and test sets used for the other models attempting to classify the churn in internet customers. The parameters used in this decision tree model are also the same as the parameters used in the all customers decision tree classifier.

When predicting the churn of internet customers in the test set, we see that the model has an accuracy of 74.26%, which is the most accurate model for the internet customer subset. However, the recall of this model is 51.6%, which would be the worst performing model for this metric. The confusion matrix along, model metrics, and metrics plot are provided in Figure 19.



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | Predicted | |  |  | Metric | Value |
|  |  | Retained | Churned |  |  | Accuracy: | 74.26% |
| Actual | Retained | 1061 | 188 |  |  | Recall: | 51.61% |
| Churned | 285 | 304 |  |  | Precision: | 61.79% |
|  |  |  |  |  |  | F1 Score: | 0.5624 |

Figure 19: Decision Tree Classifier (Internet Customers)

The decision tree for the subset of customers with internet finds that the variables that were useful in predicting customer churn were the same variables that the model found useful in predicting customer churn for all customers. The only real difference is that customers who use DSL are likely to churn if they are still relatively new customers (customer for 5 months or fewer).

Model Summary

While one logical way to choose a model would be to pick the most accurate model, our priority in this business case is to prevent customers from churning. Customers who we believe would be likely to churn would receive additional customer support as well as some additional discounts. While these additional efforts to prevent customers from churning does have an immediate adverse effect on our bottom line (money that would not have been spent otherwise), we have determined that the negative effects of spending resources on customers who were not actually going to churn as our model had predicted (false-positive) are negligible compared to the adverse effect of ignoring customers who were actually going to churn, but our model did not register a positive event (false-negative); therefore, we will want to prioritize a model that has a higher F1-score, with the model performing better in terms of recall than precision. A summary table for the performance metrics of the classification models is provided in Table 9.

Table 9: Metrics for Classification Models

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| All Customers | | | | |
| Model | Accuracy | Recall | Precision | F1 Score |
| SVM | 76.75% | 73.37% | 56.79% | 0.6403 |
| Naive Bayes | 77.05% | 70.80% | 57.56% | 0.6350 |
| Decision Tree | 77.94% | 36.01% | 71.69% | 0.4794 |
| Internet Customers | | | | |
| Model | Accuracy | Recall | Precision | F1 Score |
| SVM | 72.74% | 72.84% | 55.71% | 0.6313 |
| Naive Bayes | 73.61% | 74.36% | 56.74% | 0.6436 |
| Decision Tree | 74.27% | 51.61% | 61.79% | 0.5624 |

The purpose of evaluating a model on both all customers and a subset of customers that have internet is to determine whether the different subsets have radically different behaviors, and to see if the subset of data allowed the classification model to perform better than the entire dataset. Based on the results of the test sets, we determined that we would choose one model from each subset.

We determined that the SVM classifier was the best model to utilize from the training conducted on all customers. While the accuracy is slightly worse than the naive bayes and decision tree model, the SVM model had the best recall and F1-score of the three different models. The boost of 3% to recall has a much more positive effect for customer retention purposes than a 1% improvement in overall accuracy

We also determined that the naive bayes classifier is the best model to use on the subset of customers with internet as a part of their service. We decided this was the best model to use for this subset because it had the highest recall and f1-score. The decision tree for this subset performed better in terms of recall than the decision tree for all customers, but even with this increase in recall, the algorithm is not complex enough to be able to tell the subtle differences between churned and retained customers.

Recommendations to Management

The analysis conducted focused on finding and modeling different ways for a telecommunications company to retain more customers. We used a dataset consisting of customer data, including demographics, subscription services used, payments made, length of subscription, and whether the subscription was cancelled. With our investigation complete, we recommend the following steps to assist the company in increasing customer retention.

First the company should investigate why customers without dependents are canceling their subscription significantly at a higher rate than those with dependents. We found this to be true across the entire dataset and since customers without dependents make up most of the customers subscribed, we believe this information could be used to provide many more retained customers.

Next we would advise devoting resources into early retention programs for new customers. We found that a majority of customers that did cancel their service, did so in the first year of the service if not within the first few months. After this first year, the churn rate drops significantly. This finding was discovered in the initial statistical analysis, and supported by the Decision Tree classifier model. Programs focused on retaining new customers could provide another boon to retaining customers.

We also recommend conducting a survey among customers with an internet package, specifically the fiber optic internet package, to determine the root cause of customer dissatisfaction with these services. It is important to understand all the reasons behind lower customer retention for one particular service comparing to another and develop an appropriate strategy that helps to retain customers. Potential reasons for suppressed retention rates could include the price, quality, or lack of customer service.

We have also completed a rigorous modeling process to provide a way to predict whether a customer is likely to ~~c~~hurn or not. These models will allow the company to target specific customers with personalized retention plans. Three different learning algorithms were trained to model the churn of all the customers in the dataset as well only customers subscribed to an internet service. The models chosen prioritized a low false negative rate, but did not lose too much information via an increase to the false positive rate. This process provided us with two models that provided a signal more than 73% of the time a customer would churn.

When attempting to estimate churn for all customers, we recommend that the company utilizes the SVM model. We also recommend that the company utilizes the naive bayes classifier when estimating the churn of customers with an internet package. These two models will be invaluable to the company in determining where to allocate customer service resources in an attempt to increase retention, and in turn overall revenue.

One final insight we can gather from our classification models is which currently measured variable is having the most impact on customer churn. While we did not choose to use the decision tree classifier to determine whether a customer would be retained or churn, this model provides us with specific breakpoints in the data which the model learned to be critical in determining customer churn.

For all customers in the dataset, the branches the algorithm found to be critical in determining churn were contract type, internet service, and tenure. For internet services, the algorithm once again found that contract type, tenure, and internet type to be the critical variables in determining churn. From this, we can infer that customers who are not committed to long-term contracts are much more likely to churn from the service. We can also tell that fiber optic customers are much more likely to churn than DSL customers, but DSL customers that have been been using our service for less than five months are also susceptible to churning.

These insights inform us that we should prioritize customer retention resources for DSL customers that have been with the company for less than 5 months, and fiber optic customers that have been with the company for less than 16 months. Because these churn rates generally occur early in the subscription, we recommend changes to our onboarding procedure for month-to-month customers. The overall high churn rate for fiber optic customers shows that either the price for the service is too expensive, or there is a quality issue that needs to be corrected.

Actionable Insights

After thoroughly investigating our business questions, we were able to develop a range of actions that should be pursued in order to increase the retention rate and therefore increase company profits. The business questions along with our discoveries and action items are as follows:

* What demographics (if any) are most likely to churn?
  + Finding: Seniors and customers without dependents have a churn rate higher than their counterparts
  + Action Item 1: Conduct a survey with seniors to find if the cause of churn is voluntary or involuntary. If voluntary, investigate if price, complexity, or customer service, are not to satisfaction.
  + Action Item 2: Increase customer service resources towards those without dependents, and potentially offer promotional rate.
* Are there any significant trends in when churned customer choose to leave?
  + Finding: Customers typically leaves within the first year of their subscription, if not within their first month
  + Action Item 3: Look into current onboarding process to see if we can improve the system
* Does any contract type perform significantly better compared to other options?
  + Finding: Customers under the month-to-month contract churn at a significantly higher rate than customers with one year or two year contracts
  + Action Item 4: Provide greater incentives for customers to sign a one year or two year contracts
* Which customers should be provided additional resources in order to increase retention?
  + Finding: The Decision Tree classifier believes customers on month-to-month contracts are at higher risk of churning, especially new customers, and customers using the fiber optic internet package
  + Action Item 5: Conduct a survey to investigate why customers who use fiber optic internet are even more susceptible to churn. Also further investigate the problems with the onboarding process for DSL customers.

Future Analysis

The dataset we had used to conduct this analysis utilized a wide range of demographic and service related variables. With this data we were able to determine key demographics that are depressing the retention rate, as well as issues with key services. If we were to further analyze the issue of churn within the telecom company we would have the following recommendations:

1. Collect data on whether the customer owns their property, or is renting. If the customer is renting, collect data about when lease is ending.
2. Collect this data for multiple months, so an analysis can be conducted on an annual basis. Many people move in August due to seasonal work and school cycles, and could provide more context for a single month’s sample.
3. Collect more continuous data relating to customer income, number of dependents, cost of monthly mortgage or rent. This data could allow us to better model tenure of the customers.
4. Collect geographic data of the customers. This data could provide additional information about income of population, and trends by geographic location could result in more targeted ad campaigns to recruit new customers.

Bibliography

Braun M. and Schweidel D. (2011) Modeling Customer Lifetimes with Multiple Causes of Churn. Marketing Science 30. 881-902.

Gallo, A. (2014). The Value of Keeping the Right Customers.  Harvard Business Review. Retrieved from <https://hbr.org/2014/10/the-value-of-keeping-the-right-customers>

Hardie, B. (2009). Probability Models for Customer-Base Analysis. Annual Advanced Research Techniques Forum. 2-222.  Retrieved from <http://www.brucehardie.com/talks/ho_cba_tut_art_09.pdf>

Lima E., Mues C. and B. Baesens (2009). Domain Knowledge Integration in Data Mining Using Decision Tables: Case Studies in Churn Prediction. The Journal of the Operational Research Society 60. 1096-1106.

Mike, D. (2018). How Verizon, AT&T, T-Mobile, Sprint and More Stacked up in Q1 2018: The Top 7 Carriers. FierceWireless.  Retrieved from [www.fiercewireless.com/wireless/how-verizon-at-t-t-mobile-sprint-and-more-stacked-up-q1-2018-top-7-carriers](http://www.fiercewireless.com/wireless/how-verizon-at-t-t-mobile-sprint-and-more-stacked-up-q1-2018-top-7-carriers).

Office of Research Integrity (2018). Responsible conduct in data management.  Retrieved from [https://ori.hhs.gov/education/products/n\_illinois\_u/datamanagement/dstopic.ht](https://ori.hhs.gov/education/products/n_illinois_u/datamanagement/dstopic.html)

Reichheld, F. and Detrick, C. (2003). Loyalty: A Prescription for Cutting Costs. Marketing Management 12. 24-25.

Techopedia. (2018).  Data Cleansing. Retrieved from<https://www.techopedia.com/definition/1174/data-cleansing>

Appendix A: R Code

####Libraries

library(ggplot2)

library(kernlab)

library(arules)

library(arulesViz)

library(e1071)

library(knitr)

library(moments)

library(reshape2)

library(rpart)

### Reading/cleaning data

######################################################################

#Alex Hyman

teledf <- read.csv("data.csv")

#Looking at summary of dataset

summary(teledf)

#Looking at structure of dataset

str(teledf)

#Check for missing data

(colSums(is.na(teledf)))

#Missing data for total charges.

teledf[is.na(teledf$TotalCharges), ]

#Only ones missing totall charges had a tenure of zero

teledf[teledf$tenure == 0, ]

#Making teledf only customers with tenure greater than or equal to 1

teledf <- subset(teledf, teledf$tenure > 0)

#Missing data for total charges. ####Drop or fill?

#Subsetting Customers with internet

internet <- subset(teledf, teledf$InternetService != "No")

summary(internet)

#Dropping the customer id

internet <- internet[,-1]

#Dropping all unused levels in internet

#Fixing internetservice column

internet$InternetService <- droplevels(internet$InternetService)

#Fixing online security column

internet$OnlineSecurity <- droplevels(internet$OnlineSecurity)

#Fixing online backup column

internet$OnlineBackup <- droplevels(internet$OnlineBackup)

#Fixing device protection

internet$DeviceProtection <- droplevels(internet$DeviceProtection)

#Fixing tech support

internet$TechSupport <- droplevels(internet$TechSupport)

#Fixing streaming tv

internet$StreamingTV <- droplevels(internet$StreamingTV)

#Fixing streaming movies

internet$StreamingMovies <- droplevels(internet$StreamingMovies)

#Subsetting customers without internet

nointernet <- subset(teledf, teledf$InternetService == "No")

#Dropping unused levels from nointernet

summary(nointernet)

#Dropping multiple lines no phone service

nointernet$MultipleLines <- droplevels(nointernet$MultipleLines)

#deleting all unused columns

nointernet <- nointernet[,-6:-14]

######################################################################

###Descriptive statistics

######################################################################

#Alex Hyman

#Function to calculate churn with tapply

churn <- function(col){

per <- sum(col == "Yes") / length(col) \* 100

return(per)

}

#Overall churn rate

churn.all <- sum(teledf$Churn == "Yes") / nrow(teledf) \* 100

#Churn rate by not having any internet service

#True means they have internet

churn.internet <- tapply(teledf$Churn, teledf$InternetService != "No", churn)

#Table containing all churn rates

table.churn <- data.frame("Overall Churn" = churn.all, "Internet Churn" =

churn.internet[2], "No Internet Churn" = churn.internet[1])

#Putting together a nice table of churn rates

knitr::kable(table.churn)

#Looking at churn between the two internet types

tapply(internet$Churn, internet$InternetService, churn)

#Percent of Monthly revenue by internet

tapply(teledf$MonthlyCharges, teledf$InternetService, sum) / sum(teledf$MonthlyCharges) \* 100

#Average cost by internet type

tapply(teledf$MonthlyCharges, teledf$InternetService, mean)

######################################################################

# Olga Perera

library(moments)

# Analyzing subsets of customers without internet and with internet

# Churn by gender for the subset of customers without internet

female <- length(nointernet$gender[which(nointernet$gender=='Female' & nointernet$Churn=='Yes')])

male <- length(nointernet$gender[which(nointernet$gender=='Male' & nointernet$Churn=='Yes')])

female

male

# Churn by age for the subset of customers without internet

nonsenior <- length(nointernet$gender[which(nointernet$SeniorCitizen==0 & nointernet$Churn=='Yes')])

senior <- length(nointernet$gender[which(nointernet$SeniorCitizen==1 & nointernet$Churn=='Yes')])

nonsenior

senior

# Churn by gender for the subset of customers with internet

femaleint <- length(internet$gender[which(internet$gender=='Female' & internet$Churn=='Yes')])

maleint <- length(internet$gender[which(internet$gender=='Male' & internet$Churn=='Yes')])

femaleint

maleint

# Churn by age for the subset of customers with internet

nonseniorint <- length(internet$gender[which(internet$SeniorCitizen==0 & internet$Churn=='Yes')])

seniorint <- length(internet$gender[which(internet$SeniorCitizen==1 & internet$Churn=='Yes')])

nonseniorint

seniorint

#Function for print descriptive stats

printVecInfo <- function (myVector)

{

cat("Mean: ", mean(myVector),"\n")

cat("Median: ", median(myVector),"\n")

cat("Min: ", min(myVector)," ")

cat("Max: ", max(myVector),"\n")

cat("Sd: ", sd(myVector), "\n")

cat("Quantiles (at 0.05 and 0.95): ", quantile(myVector, probs = c(0.05, 0.95)), "\n")

cat("Skewness: ",skewness(myVector))

}# NO INTERNET subset

# Summary stats for numerical variables

printVecInfo(nointernet$tenure)

printVecInfo(nointernet$MonthlyCharges)

printVecInfo(nointernet$TotalCharges)

#INTERNET subset

# Summary stats for numerical variables

printVecInfo(internet$tenure)

printVecInfo(internet$MonthlyCharges)

printVecInfo(internet$TotalCharges)

# Entire datatset

#Summary stats for numercal variables

printVecInfo(teledf$tenure)

printVecInfo(teledf$MonthlyCharges)

printVecInfo(teledf$TotalCharges)

# find percentage of churned customers("churn="yes")

churnYes1 <- internet$Churn[which(internet$Churn=='Yes')]

(length(churnYes1)/length(internet$Churn))\*100

# find percentage of churned customers("churn="yes")

churnYes <- nointernet$Churn[which(nointernet$Churn=='Yes')]

(length(churnYes)/length(nointernet$Churn))\*100

#find percentage of churned customers Internet subset

mtmInternet <- internet$Churn[which(internet$Churn=='Yes' & internet$Contract=='Month-to-month')]

(length(mtmInternet)/length(internet$Churn))\*100

oneInternet <- internet$Churn[which(internet$Churn=='Yes' & internet$Contract=='One year')]

(length(oneInternet)/length(internet$Churn))\*100

twoInternet <- internet$Churn[which(internet$Churn=='Yes' & internet$Contract=='Two year')]

(length(twoInternet)/length(internet$Churn))\*100

allInternet <- internet$Churn[which(internet$Churn=='Yes')]

(length(allInternet)/length(internet$Churn))\*100

#find percentage of churned customers phone subset

mtmNointernet <- nointernet$Churn[which(nointernet$Churn=='Yes' & nointernet$Contract=='Month-to-month')]

(length(mtmNointernet)/length(nointernet$Churn))\*100

oneNointernet <- nointernet$Churn[which(nointernet$Churn=='Yes' & nointernet$Contract=='One year')]

(length(oneNointernet)/length(nointernet$Churn))\*100

twoNointernet <- nointernet$Churn[which(nointernet$Churn=='Yes' & nointernet$Contract=='Two year')]

(length(twoNointernet)/length(nointernet$Churn))\*100

allNoInternet <- internet$Churn[which(nointernet$Churn=='Yes')]

(length(allNoInternet)/length(nointernet$Churn))\*100

######################################################################

# Spalding Lewis

# Demographics: dependents(y/n)

# dataframe: customers with dependents

dependents <- subset(teledf, teledf$Dependents == "Yes", drop = FALSE)

# dataframe: customers without dependents

no\_dependents <- subset(teledf, teledf$Dependents == "No", drop = FALSE)

# churn rate: dependents(y/n)

dep\_churn\_rate <- churn(dependents$Churn)

nodep\_churn\_rate <- churn(no\_dependents$Churn)

# dataframe: average churn, dependents(y) churn, dependents(n) churn

table.churn <- data.frame("Overall Churn" = churn.all, "Churn: Customers with Dependents" =

dep\_churn\_rate, "Churn: Customers without Dependents" = nodep\_churn\_rate)

# table: print table.churn

knitr::kable(table.churn)

# Demographics: partner(y/n)

# dataframe: customers with partners

yes\_partner <- subset(teledf, teledf$Partner == "Yes", drop = FALSE)

# dataframe: customers without partners

no\_partner <- subset(teledf, teledf$Partner == "No", drop = FALSE)

# churn rates partner(y/n)

partner\_churn <- churn(yes\_partner$Churn)

no\_partner\_churn <- churn(no\_partner$Churn)

# dataframe: churn rates with/without partner

table.partner.churn <- data.frame("Overall Churn" = churn.all, "Churn: Customers with Partners" =partner\_churn, "Churn: Customers without Partners" = no\_partner\_churn)

# table: print table.partner.churn

knitr::kable(table.partner.churn)

# Demographics: customers with dependents and partners vs single customers

partner.dep <- subset(teledf,teledf$Dependents=='Yes' & teledf$Partner=='Yes',drop=FALSE)

nopartner.nodep <- subset(teledf,teledf$Dependents=='No' & teledf$Partner=='No', drop=FALSE)

# churn rates: multi-person households vs single customers

churn.family <- churn(partner.dep$Churn)

churn.single <- churn(nopartner.nodep$Churn)

# dataframe: family churn vs single churn

table.churn.household <- data.frame("Overall churn"=churn.all,"Dependents and Partners"=churn.family,"Single"=churn.single)

# print table: churn rates

knitr::kable(table.churn.household)

######################################################################

# David Mackey

# Analyzing what contracts are better

# Attach the dataset

attach(teledf)

# Seeing Churn rates for male and female customers

tapply(Churn, gender, churn)

# Seeing what contracts have the most churn

churn.contracts <- data.frame(tapply(Churn, Contract, churn))

colnames(churn.contracts) <- 'Churn Rate (%)'

# Seeing if paperless billing has any affect on Churn

churn.paperless <- data.frame(tapply(Churn, PaperlessBilling, churn))

colnames(churn.paperless) <- 'Churn Rate (%)'

# Seeing if Payment method has any affect on Churn

churn.payment <- data.frame(tapply(Churn, PaymentMethod, churn))

colnames(churn.payment) <- 'Churn Rate (%)'

######################################################################

###Inferential Statistics

######################################################################

#Alex Hyman

##hypothesis test that seniors have statistically significant

#difference in churn rate

#Churn rates among senior citizens

seniors <- length(teledf$customerID[(teledf$SeniorCitizen == 1 & teledf$Churn == "Yes")]) / length(teledf$customerID[teledf$SeniorCitizen == 1]) \* 100

#Rounding the rate

seniors = round(seniors, digits = 2)

#Print out churn rate for senior citizens

cat("Churn rate among senior citizens: ", seniors, "%", sep = "")

#Churn rate for non-senior citizen customers

allOtherCustomers <- length(teledf$customerID[teledf$Churn == "Yes" & teledf$SeniorCitizen == 0]) / length(teledf$customerID[teledf$SeniorCitizen == 0]) \* 100

allOtherCustomers <- round(allOtherCustomers, 2)

cat("Churn rate of non-Seniors: ", allOtherCustomers, "%", sep = "")

#Creating booleans for churn in t.test application

teledf[teledf$Churn == "Yes", "LogicalChurn"] <- T

teledf[teledf$Churn == "No", "LogicalChurn"] <- F

#Are these groups statistically different

t <- t.test(LogicalChurn ~ SeniorCitizen, data = teledf)

print(t)

##Chi-square test on whether contract type and churn have a relationship

#Chi-square test for independence

chit <- chisq.test(teledf$Contract, teledf$Churn)

#What was actually observed

chit$observed

#What was expected

chit$expected

######################################################################

#Spalding Lewis

# chi-square test: dependents and churn

# chi-square test for independence

dep\_chisq <- chisq.test(teledf$Dependents, teledf$Churn)

# observed

dep\_chisq$observed

# expected

dep\_chisq$expected

# t test: dependents and churn

t <- t.test(LogicalChurn ~ Dependents, data = teledf)

print(t)

# chi-square: partner and churn

partner\_chisq <- chisq.test(teledf$Partner, teledf$Churn)

partner\_chisq$observed

partner\_chisq$expected

t2 <- t.test(LogicalChurn ~ Partner, data = teledf)

print(t2)

######################################################################

###Visualizations

######################################################################

#Alex Hyman

#Churn by contract type bar plot

p <- ggplot(teledf, aes(Contract)) + geom\_bar(aes(fill = Churn))

p <- p + ylab("Count") + ggtitle("Churn by Contract type")

p

#Histogram of tenure for month-to-month contracts with churn fill

p <- ggplot(teledf[teledf$Contract == "Month-to-month", ], aes(tenure, fill = Churn))

p <- p + geom\_histogram(binwidth = 1) + xlab("Tenure (months)")

p <- p + ggtitle("Histogram on Churn and tenure\nfor Month-to-Month Customers")

p

#Plotting tenure vs charge to see if pattern in where churn is

g <- ggplot(teledf, aes(tenure, MonthlyCharges, color = Churn, shape = Contract))

g <- g + geom\_point()

g <- g + ggtitle("Tenure vs. Average Monthly Charges")

g <- g + xlab("Tenure") + ylab("Average Monthly Charges")

g

######################################################################

#Olga Perera

# Plotting churn rates by contract type for subset of customers without internet

ChurnYesMTM <- length(nointernet$Contract[which(nointernet$Churn=='Yes'& nointernet$Contract=='Month-to-month')])

ChurnNoMTM <- length(nointernet$Contract[which(nointernet$Churn=='No'& nointernet$Contract=='Month-to-month')])

ChurnYes2y <- length(nointernet$Contract[which(nointernet$Churn=='Yes'& nointernet$Contract=='Two year')])

ChurnNo2y <- length(nointernet$Contract[which(nointernet$Churn=='No'& nointernet$Contract=='Two year')])

ChurnYes1y <- length(nointernet$Contract[which(nointernet$Churn=='Yes'& nointernet$Contract=='One year')])

ChurnNo1y <- length(nointernet$Contract[which(nointernet$Churn=='No'& nointernet$Contract=='One year')])

# Plot Churn by contract type for customers without internet

g <- ggplot(nointernet, aes(Contract)) + geom\_bar(aes(fill = Churn)) + ggtitle("Churn by Contract Type for Customers without Internet")

g

# Plot Churn by contract type for customers with internet

g1 <- ggplot(internet, aes(Contract)) + geom\_bar(aes(fill = Churn)) + ggtitle("Churn by Contract Type for Customers with Internet")

G1

#Tenure for different types of contract histograms

#Tenure for month-to month option

mtm <- subset(teledf, teledf$Contract=="Month-to-month")

mtmplot1 <- ggplot(mtm, aes(x=tenure)) + geom\_histogram(binwidth=5, color="red", fill="#DD5868") +

ggtitle("Month to Month Contract") + theme(plot.title = element\_text(hjust = 0.5)) + xlab("Tenure, months") + ylab("# of customers")

mtmplot1

#Tenure for one year contract option

oneyear <- subset(teledf, teledf$Contract=="One year")

mtmplot2 <- ggplot(oneyear, aes(x=tenure)) + geom\_histogram(binwidth=5, color="blue", fill="#1E4594") +

ggtitle("One Year Contract") + theme(plot.title = element\_text(hjust = 0.5)) + xlab("Tenure, months") + ylab("# of customers")

mtmplot2

#Tenure for two year contract option

twoyear <- subset(teledf, teledf$Contract=="Two year")

mtmplot3 <- ggplot(twoyear, aes(x=tenure)) + geom\_histogram(binwidth=5, color="green", fill="#125528") +

ggtitle("Two Year Contract") + theme(plot.title = element\_text(hjust = 0.5)) + xlab("Tenure, months") + ylab("# of customers")

mtmplot3

######################################################################

#Spalding Lewis

# visualizations

# plot: churn(y/n) and dependents (y/n)

a <- ggplot(teledf, aes(Dependents)) + geom\_bar(aes(fill = Churn))

a <- a + ylab("Count") + ggtitle("Churn for Customers with/without Dependents")

a

# plot: churn(y/n), dependents(y/n) and other tested variables

# tenure

b <- ggplot(teledf, aes(Dependents, tenure, color = Churn))

b <- b + geom\_boxplot()

b <- b + ggtitle("Tenure, Dependents and Churn")

b <- b + xlab("Dependents") + ylab("Tenure")

b

# monthly charges

c <- ggplot(teledf, aes(Dependents, MonthlyCharges, color = Churn))

c <- c + geom\_boxplot()

c <- c + ggtitle("Monthly Charges, Dependents and Churn")

c <- c + xlab("Dependents") + ylab("Monthly Charges")

c

# total charges

d <- ggplot(teledf, aes(Dependents, TotalCharges, color = Churn))

d <- d + geom\_boxplot()

d <- d + ggtitle("Total Charges, Dependents and Churn")

d <- d + xlab("Dependents") + ylab("Total Charges")

d

# VISUALIZATIONS -- churn, dependents and partner

e <- ggplot(teledf) +

geom\_bin2d(aes(x=Dependents, y=Churn,))

e

f <- ggplot(teledf) +

geom\_bin2d(aes(x=Partner, y=Churn))

f

# Histogram describing churn vs Tenure for Customers with Dependents

g <- ggplot(dependents) +

geom\_histogram(aes(x=tenure,fill=Churn)) + ggtitle('Churn and Tenure for Customers with Dependents')

g

# Histogram describing churn vs Tenure for Customers without Dependents

h <- ggplot(no\_dependents) +

geom\_histogram(aes(x=tenure, fill=Churn)) + ggtitle('Churn and Tenure for Customers without Dependents')

h

# Histogram describing churn vs Tenure for Customers with Partners

j <- ggplot(yes\_partner) +

geom\_histogram(aes(x=tenure, fill=Churn)) + ggtitle('Churn and Tenure for Customers with Partners')

j

# Histogram describing churn vs Tenure for Customers without Partners

k <- ggplot(no\_partner) +

geom\_histogram(aes(x=tenure, fill=Churn)) + ggtitle('Churn and Tenure for Customers without Parteners')

k

# Overall Histogram of Churn vs Tenure

l <- ggplot(teledf) +

geom\_histogram(aes(x=tenure, fill=Churn)) + ggtitle('Histogram of Churn and Tenure')

l

######################################################################

#David Mackey

# Plotting the churn vs gender

Gender\_Plot <- ggplot(teledf) + geom\_bar(aes(x = gender, fill=Churn)) +

ggtitle("Gender vs Churn")

Gender\_Plot

# Subset of customers that have phone service

telephonedf <- teledf[teledf$PhoneService == 'Yes',]

# Boxplots showing if having multiple lines changes either the money made or tenure

ggplot(telephonedf, aes(MultipleLines, TotalCharges)) + geom\_boxplot() +

ggtitle('Multiple Line Phone Service vs Total Charges') +xlab('Multiple Lines') + ylab('Total Charges') +

theme(plot.title = element\_text(hjust = 0.5))

ggplot(telephonedf, aes(MultipleLines, tenure)) + geom\_boxplot() +

ggtitle('Multiple Line Phone Service vs Tenure') +xlab('Multiple Lines') + ylab('Tenure (Mo.)') +

theme(plot.title = element\_text(hjust = 0.5))

# Subset those customers with internet service that were churned

Churned.internet <- internet[internet$Churn == 'Yes',]

# Get the total numbre of churned customers that had internet

internet.churn.rows <- nrow(Churned.internet)

# Percentage of churned customers that had online security

churn.security <- 100 \* nrow(Churned.internet[Churned.internet$OnlineSecurity =="Yes",])/internet.churn.rows

# Percentage of churned customers that had Online Backup

churn.backup <- 100 \* nrow(Churned.internet[Churned.internet$OnlineBackup =="Yes",])/internet.churn.rows

# Percentage of churned customers that had Device Protection

churn.protect <- 100 \* nrow(Churned.internet[Churned.internet$DeviceProtection =="Yes",])/internet.churn.rows

# Percentage of churned customers that had Tech Support

churn.support <- 100 \* nrow(Churned.internet[Churned.internet$TechSupport =="Yes",])/internet.churn.rows

# Percentage of churned customers that had Streaming TV

churn.TV <- 100 \* nrow(Churned.internet[Churned.internet$StreamingTV =="Yes",])/internet.churn.rows

# Percentage of churned customers that had Streaming Movies

churn.movies <- 100 \* nrow(Churned.internet[Churned.internet$StreamingMovies =="Yes",])/internet.churn.rows

# Make a dataframe from above percentages

churned.options <- data.frame(c(churn.security,churn.backup,churn.protect,churn.support, churn.TV, churn.movies))

churned.options$Service <- c('Online Security','Online Backup','Device Protection','Tech Support','Spreaming TV','Streaming Movies')

colnames(churned.options) <- c('Churned\_Percent','Service')

# Create Bar plot to show the churn percentages of each service

ggplot(churned.options, aes(x = Service, y=Churned\_Percent)) +

geom\_bar(stat = 'identity') + ggtitle('Churn Rates of for additional internet services') +

theme(plot.title = element\_text(hjust = 0.5))

######################################################################

###Linear Modeling

######################################################################

#Alex Hyman

#Only two numerical columns, is there a linear relationship?

#Must use the churned customers because we know exactly how long they lasted

#Looking at correlation between monthly charges and tenure

cor(teledf[teledf$Churn == "Yes", "tenure"], teledf[teledf$Churn == "Yes", "MonthlyCharges"])

#Not that great

#Plot to take a look

#Subsetting the churned customers from the teledf

teledf.churn <- subset(teledf, teledf$Churn == "Yes")

#Getting rid of customer ID, total charges and churn

drop <- c("TotalCharges", "Churn", "customerID")

teledf.churn <- teledf.churn[, !names(teledf.churn) %in% drop]

#Creating a linear model directly from monthly charges

lm1 <- lm(tenure ~ MonthlyCharges, data = teledf.churn)

#Look at adjusted R2

summary(lm1)

#0.16 Adjusted R2 is terrible

#plotting Monthly charges vs tenure

plot(teledf.churn$MonthlyCharges, teledf.churn$tenure,

xlab = "Monthly Charges $", ylab = "Tenure (Months)", main = "Monthly Charges vs. Tenure")

#Drawing the model in the plot

abline(lm1, col = "red", lwd = 3)

#Linear model using all values

lm.all <- lm(tenure ~ ., data = teledf.churn)

#Looking at model

summary(lm.all)

#Predicting tenure for all

lm.all.predict <- predict(lm.all, teledf.churn)

#Creating data frame to compare predicted and actual

df.predict.all <- data.frame(actual = teledf.churn$tenure, predicted = lm.all.predict)

#Calculating Residual

df.predict.all$residual <- df.predict.all$actual - df.predict.all$predicted

#Plotting residuals

plot(df.predict.all$actual, df.predict.all$residual, main = "Residuals for Tenure",

xlab = "Actual Tenure", ylab = "Residual")

#What about for only internet customers

#Subsetting churned customers

internet.churn <- subset(internet, internet$Churn == "Yes")

#Getting rid of churn variable

internet.churn <- internet.churn[,!names(internet.churn) %in% c("TotalCharges", "Churn")]

#Is there a correlation for internet users?

cor(internet.churn$tenure, internet.churn$MonthlyCharges)

#Still not great

lm2 <- lm(tenure ~ MonthlyCharges, data = internet.churn)

summary(lm2)

#Linear model for internet customers using all variables

lm3 <- lm(tenure ~ ., data = internet.churn)

summary(lm3)

#Plot to take a look

plot(internet.churn$MonthlyCharges, internet.churn$tenure, main = "Monthly Charges vs Tenure",

xlab = "Monthly Charges $", ylab = "Tenure (Months)")

#Similar shape

abline(lm2, lwd = 3, col = "red")

######################################################################

###Data Mining

######################################################################

#Alex Hyman

##Subsetting train and test for all customers

#Dropping id and Total charges (it is the same as monthly \* tenure)

#teledf.svm <- teledf[,!names(teledf) %in% c("customerID", "TotalCharges")]

#Setting seed to keep our random customers the same

set.seed(1016)

#Creating random indices to index from 1 to nrows in teledf.svm

randindex <- sample(1:nrow(teledf))

#Creating a cutoff point that is 2/3 through the data

cutOff <- floor(2 \* nrow(teledf) / 3)

#Indexing from our full dataset the random indexes from 1 to cutoff

teledf.train <- teledf[randindex[1:cutOff], ]

#Putting the rest in out test set

teledf.test <- teledf[randindex[(cutOff + 1):nrow(teledf)], ]

teledf.train <- teledf.train[,!names(teledf.train) %in% c("customerID", "TotalCharges")]

teledf.test <- teledf.test[,!names(teledf.test) %in% c("customerID", "TotalCharges")]

#Training and test data for internet subset

randindex <- sample(1:nrow(internet))

#Cutoff for train and test

cutoff <- floor(2 \* length(randindex) / 3)

#Getting rid of internet total charges

internet <- internet[, !names(internet) %in% "TotalCharges"]

#Indexing training set

internet.train <- internet[randindex[1:cutoff], ]

#Indexing test set

internet.test <- internet[randindex[(cutoff+1):length(randindex)], ]

####################################

##SVM - All

#Want to minimize Type 2 Error (Predicted Not to Leave, actually leaves)

svm.1 <- ksvm(Churn ~ ., data = teledf.train, kernel = "rbfdot",

kpar = "automatic", C = 1, prob.model = T, cross = 3, probability = T)

#Predicting svm1 on the test set

svm.pred <- predict(svm.1, teledf.test, type = "p")

svm.pred.all <- svm.pred[,2]

svm.original <- table(data.frame(actual = teledf.test$Churn, predicted = predict(svm.1, teledf.test, type = "response")))

cat("Accuracy: ", (svm.original[1,1] + svm.original[2,2]) / sum(svm.original) \* 100, "%", sep = "")

sum(teledf.test$Churn == "No") / length(teledf.test$Churn)

test.threshold <- seq(0.1, 0.85, by = 0.01)

#Creating vectors to hold model statistics

svm.accuracy.all <- numeric(75)

svm.recall.all <- numeric(75)

svm.precision.all <- numeric(75)

svm.f1.score.all <- numeric(75)

for (i in 1:length(test.threshold)){

#If the prediction is greater than the threshold, make it a 1, else 0

temp.pred <- ifelse(svm.pred.all > test.threshold[i], 1, 0)

#Create a data frame with the predictions and the actual

df.temp <- data.frame(actual = teledf.test$Churn, predicted = temp.pred)

#Make it a table

comptable.temp <- table(df.temp)

#put Overall accuracy for this model in the accuracy vector

svm.accuracy.all[i] <- (comptable.temp[1,1] + comptable.temp[2,2]) / sum(comptable.temp)

#put Overall recall for this model in the recall vector

svm.recall.all[i] <- comptable.temp[2,2] / (comptable.temp[2,1] + comptable.temp[2,2])

#put precision accuracy for this model in the precision vector

svm.precision.all[i] <- comptable.temp[2,2] / sum(comptable.temp[,2])

#put f1 score accuracy for this model in the f1 score vector

svm.f1.score.all[i] <- 2 \* ((svm.precision.all[i] \* svm.recall.all[i]) / (svm.precision.all[i] + svm.recall.all[i]))

}

#Creating data frame for model statistics

results.svm.all <- data.frame(threshold = test.threshold, svm.accuracy.all, svm.recall.all , svm.precision.all , svm.f1.score.all)

#Finding maximum f1 score

max.f1.all <- results.svm.all[which.max(results.svm.all$svm.f1.score.all), "threshold"]

#Melting data frame while maintaining the threshold

results.svm.all <- melt(results.svm.all, id.vars = "threshold")

#Plotting the model statistics versus the threshold for probability

g <- ggplot(results.svm.all, aes(x = threshold, y = value, color = variable))

#Drawing the a line for each value and a vertical line at the probability of the maximum f1 score

g <- g + geom\_line() + ggtitle("SVM Model Quality Statistics (All Customers)") + geom\_vline(xintercept = max.f1.all, col = "red", size = 1)

g

svm.pred.final <- ifelse(svm.pred.all > max.f1.all, 1, 0)

#Table for confusion Matrix

compTable.svm <- data.frame(Actual = teledf.test$Churn, Predicted = svm.pred.final)

#Looking at the confusion Matrix

compTable.svm <- table(compTable.svm)

svm.accuracy.final <- (compTable.svm[1,1] + compTable.svm[2,2]) / sum(compTable.svm)

#put Overall recall for this model in the recall vector

svm.recall.final <- compTable.svm[2,2] / (compTable.svm[2,1] + compTable.svm[2,2])

#put precision accuracy for this model in the precision vector

svm.precision.final <- compTable.svm[2,2] / sum(compTable.svm[,2])

#put f1 score accuracy for this model in the f1 score vector

svm.f1.score.final <- 2 \* ((svm.precision.final \* svm.recall.final) / (svm.precision.final + svm.recall.final))

cat("SVM Model quality (All Customers):\n\taccuracy: ", svm.accuracy.final, "\n\trecall: ", svm.recall.final,

"\n\tprecision: ", svm.precision.final, "\n\tF1 score: ", svm.f1.score.final)

##SVM - Internet

#Want to minimize Type 2 Error (Predicted Not to Leave, actually leaves)

svm.2 <- ksvm(Churn ~ ., data = internet.train, kernel = "rbfdot",

kpar = "automatic", C = 1, prob.model = T, cross = 3, probability = T)

#Predicting svm1 on the test set

svm.pred.internet <- predict(svm.2, internet.test, type = "p")

svm.pred.internet <- svm.pred.internet[,2]

svm.original.internet <- table(data.frame(actual = internet.test$Churn, predicted = predict(svm.2, internet.test, type = "response")))

cat("Accuracy: ", (svm.original.internet[1,1] + svm.original.internet[2,2]) / sum(svm.original.internet) \* 100, "%", sep = "")

sum(internet.test$Churn == "No") / length(internet.test$Churn)

test.threshold <- seq(0.1, 0.85, by = 0.01)

#Creating vectors to hold model statistics

svm.accuracy.internet <- numeric(75)

svm.recall.internet <- numeric(75)

svm.precision.internet <- numeric(75)

svm.f1.score.internet <- numeric(75)

for (i in 1:length(test.threshold)){

#If the prediction is greater than the threshold, make it a 1, else 0

temp.pred <- ifelse(svm.pred.internet > test.threshold[i], 1, 0)

#Create a data frame with the predictions and the actual

df.temp <- data.frame(actual = internet.test$Churn, predicted = temp.pred)

#Make it a table

comptable.temp <- table(df.temp)

#put Overall accuracy for this model in the accuracy vector

svm.accuracy.internet[i] <- (comptable.temp[1,1] + comptable.temp[2,2]) / sum(comptable.temp)

#put Overall recall for this model in the recall vector

svm.recall.internet[i] <- comptable.temp[2,2] / (comptable.temp[2,1] + comptable.temp[2,2])

#put precision accuracy for this model in the precision vector

svm.precision.internet[i] <- comptable.temp[2,2] / sum(comptable.temp[,2])

#put f1 score accuracy for this model in the f1 score vector

svm.f1.score.internet[i] <- 2 \* ((svm.precision.internet[i] \* svm.recall.internet[i]) / (svm.precision.internet[i] + svm.recall.internet[i]))

}

#Creating data frame for model statistics

results.svm.internet <- data.frame(threshold = test.threshold, svm.accuracy.internet, svm.recall.internet , svm.precision.internet , svm.f1.score.internet)

#Finding maximum f1 score

max.f1 <- results.svm.internet[which.max(results.svm.internet$svm.f1.score.internet), "threshold"]

#Melting data frame while maintaining the threshold

results.svm.internet <- melt(results.svm.internet, id.vars = "threshold")

#Plotting the model statistics versus the threshold for probability

g <- ggplot(results.svm.internet, aes(x = threshold, y = value, color = variable))

#Drawing the a line for each value and a vertical line at the probability of the maximum f1 score

g <- g + geom\_line() + ggtitle("SVM Model Quality Statistics (Internet Customers)") + geom\_vline(xintercept = max.f1, col = "red", size = 1)

g

svm.pred.final.internet <- ifelse(svm.pred.internet > max.f1, 1, 0)

#Table for confusion Matrix

compTable.svm.internet <- data.frame(Actual = internet.test$Churn, Predicted = svm.pred.final.internet)

#Looking at the confusion Matrix

compTable.svm.internet <- table(compTable.svm.internet)

svm.accuracy.final.internet <- (compTable.svm.internet[1,1] + compTable.svm.internet[2,2]) / sum(compTable.svm.internet)

#put Overall recall for this model in the recall vector

svm.recall.final.internet <- compTable.svm.internet[2,2] / (compTable.svm.internet[2,1] + compTable.svm.internet[2,2])

#put precision accuracy for this model in the precision vector

svm.precision.final.internet <- compTable.svm.internet[2,2] / sum(compTable.svm.internet[,2])

#put f1 score accuracy for this model in the f1 score vector

svm.f1.score.final.internet <- 2 \* ((svm.precision.final.internet \* svm.recall.final.internet) / (svm.precision.final.internet + svm.recall.final.internet))

cat("SVM Model quality (Internet Customers):\n\taccuracy: ", svm.accuracy.final.internet, "\n\trecall: ", svm.recall.final.internet,

"\n\tprecision: ", svm.precision.final.internet, "\n\tF1 score: ", svm.f1.score.final.internet)

##################

## Naive Bayes

#Training a naive bayes classification model predicting churn from all variables

nb <- naiveBayes(Churn ~ ., data = teledf.train, probability = T)

#Predicting based on nb model

nb.pred <- predict(nb, teledf.test, type = "raw")

nb.original <- table(data.frame(actual = teledf.test$Churn, predicted = predict(nb, teledf.test, type = "class")))

cat("Accuracy: ", (nb.original[1,1] + nb.original[2,2]) / sum(nb.original) \* 100, "%", sep = "")

sum(teledf.test$Churn == "No") / length(teledf.test$Churn)

nb.pred.all <- nb.pred[,2]

test.threshold <- seq(0.1, 0.85, by = 0.01)

#Creating vectors to hold model statistics

nb.accuracy.all <- numeric(75)

nb.recall.all <- numeric(75)

nb.precision.all <- numeric(75)

nb.f1.score.all <- numeric(75)

for (i in 1:length(test.threshold)){

#If the prediction is greater than the threshold, make it a 1, else 0

temp.pred <- ifelse(nb.pred.all > test.threshold[i], 1, 0)

#Create a data frame with the predictions and the actual

df.temp <- data.frame(actual = teledf.test$Churn, predicted = temp.pred)

#Make it a table

comptable.temp <- table(df.temp)

#put Overall accuracy for this model in the accuracy vector

nb.accuracy.all[i] <- (comptable.temp[1,1] + comptable.temp[2,2]) / sum(comptable.temp)

#put Overall recall for this model in the recall vector

nb.recall.all[i] <- comptable.temp[2,2] / (comptable.temp[2,1] + comptable.temp[2,2])

#put precision accuracy for this model in the precision vector

nb.precision.all[i] <- comptable.temp[2,2] / sum(comptable.temp[,2])

#put f1 score accuracy for this model in the f1 score vector

nb.f1.score.all[i] <- 2 \* ((nb.precision.all[i] \* nb.recall.all[i]) / (nb.precision.all[i] + nb.recall.all[i]))

}

#Creating data frame for model statistics

results.nb.all <- data.frame(threshold = test.threshold, nb.accuracy.all, nb.recall.all , nb.precision.all , nb.f1.score.all)

#Finding maximum f1 score

max.f1 <- results.nb.all[which.max(results.nb.all$nb.f1.score.all), "threshold"]

#Melting data frame while maintaining the threshold

results.nb.all <- melt(results.nb.all, id.vars = "threshold")

#Plotting the model statistics versus the threshold for probability

g <- ggplot(results.nb.all, aes(x = threshold, y = value, color = variable))

#Drawing the a line for each value and a vertical line at the probability of the maximum f1 score

g <- g + geom\_line() + ggtitle("Naive Bayes Model Quality Statistics (All Customers)") + geom\_vline(xintercept = max.f1, col = "red", size = 1)

g

nb.pred.final <- ifelse(nb.pred.all > max.f1, 1, 0)

#Table for confusion Matrix

compTable.nb <- data.frame(Actual = teledf.test$Churn, Predicted = nb.pred.final)

#Looking at the confusion Matrix

compTable.nb <- table(compTable.nb)

nb.accuracy.final <- (compTable.nb[1,1] + compTable.nb[2,2]) / sum(compTable.nb)

#put Overall recall for this model in the recall vector

nb.recall.final <- compTable.nb[2,2] / (compTable.nb[2,1] + compTable.nb[2,2])

#put precision accuracy for this model in the precision vector

nb.precision.final <- compTable.nb[2,2] / sum(compTable.nb[,2])

#put f1 score accuracy for this model in the f1 score vector

nb.f1.score.final <- 2 \* ((nb.precision.final \* nb.recall.final) / (nb.precision.final + nb.recall.final))

cat("Naive Bayes Model quality (All Customers):\n\taccuracy: ", nb.accuracy.final, "\n\trecall: ", nb.recall.final,

"\n\tprecision: ", nb.precision.final, "\n\tF1 score: ", nb.f1.score.final)

#Table for confusion Matrix for naive bayes model

compTable.nb <- data.frame(Actual = teledf.test$Churn, Predicted = nb.pred.final)

#Looking at the naive bayes confusion Matrix

table(compTable.nb)

###Naive Bayes - internet

#Training a naive bayes classification model predicting churn from all variables

nb.internet <- naiveBayes(Churn ~ ., data = internet.train, probability = T)

#Predicting based on nb model

nb.pred.internet <- predict(nb.internet, internet.test, type = "raw")

nb.original.internet <- table(data.frame(actual = internet.test$Churn, predicted = predict(nb.internet, internet.test, type = "class")))

cat("Accuracy: ", (nb.original[1,1] + nb.original[2,2]) / sum(nb.original) \* 100, "%", sep = "")

nb.pred.internet <- nb.pred.internet[,2]

test.threshold <- seq(0.1, 0.85, by = 0.01)

#Creating vectors to hold model statistics

nb.accuracy.internet <- numeric(75)

nb.recall.internet <- numeric(75)

nb.precision.internet <- numeric(75)

nb.f1.score.internet <- numeric(75)

for (i in 1:length(test.threshold)){

#If the prediction is greater than the threshold, make it a 1, else 0

temp.pred <- ifelse(nb.pred.internet > test.threshold[i], 1, 0)

#Create a data frame with the predictions and the actual

df.temp <- data.frame(actual = internet.test$Churn, predicted = temp.pred)

#Make it a table

comptable.temp <- table(df.temp)

#put Overall accuracy for this model in the accuracy vector

nb.accuracy.internet[i] <- (comptable.temp[1,1] + comptable.temp[2,2]) / sum(comptable.temp)

#put Overall recall for this model in the recall vector

nb.recall.internet[i] <- comptable.temp[2,2] / (comptable.temp[2,1] + comptable.temp[2,2])

#put precision accuracy for this model in the precision vector

nb.precision.internet[i] <- comptable.temp[2,2] / sum(comptable.temp[,2])

#put f1 score accuracy for this model in the f1 score vector

nb.f1.score.internet[i] <- 2 \* ((nb.precision.internet[i] \* nb.recall.internet[i]) / (nb.precision.internet[i] + nb.recall.internet[i]))

}

#Creating data frame for model statistics

results.nb.internet <- data.frame(threshold = test.threshold, nb.accuracy.internet, nb.recall.internet , nb.precision.internet , nb.f1.score.internet)

#Finding maximum f1 score

max.f1 <- results.nb.internet[which.max(results.nb.internet$nb.f1.score.internet), "threshold"]

#Melting data frame while maintaining the threshold

results.nb.internet <- melt(results.nb.internet, id.vars = "threshold")

#Plotting the model statistics versus the threshold for probability

g <- ggplot(results.nb.internet, aes(x = threshold, y = value, color = variable))

#Drawing the a line for each value and a vertical line at the probability of the maximum f1 score

g <- g + geom\_line() + ggtitle("Naive Bayes Model Quality Statistics (Internet Customers)") + geom\_vline(xintercept = max.f1, col = "red", size = 1)

g

nb.pred.final.internet <- ifelse(nb.pred.internet > max.f1, 1, 0)

#Table for confusion Matrix

compTable.nb.internet <- data.frame(Actual = internet.test$Churn, Predicted = nb.pred.final.internet)

#Looking at the confusion Matrix

compTable.nb.internet <- table(compTable.nb.internet)

nb.accuracy.final.internet <- (compTable.nb.internet[1,1] + compTable.nb.internet[2,2]) / sum(compTable.nb.internet)

#put Overall recall for this model in the recall vector

nb.recall.final.internet <- compTable.nb.internet[2,2] / (compTable.nb.internet[2,1] + compTable.nb.internet[2,2])

#put precision accuracy for this model in the precision vector

nb.precision.final.internet <- compTable.nb.internet[2,2] / sum(compTable.nb.internet[,2])

#put f1 score accuracy for this model in the f1 score vector

nb.f1.score.final.internet <- 2 \* ((nb.precision.final.internet \* nb.recall.final.internet) / (nb.precision.final.internet + nb.recall.final.internet))

cat("Naive Bayes Model quality (Internet Customers):\n\taccuracy: ", nb.accuracy.final.internet, "\n\trecall: ", nb.recall.final.internet,

"\n\tprecision: ", nb.precision.final.internet, "\n\tF1 score: ", nb.f1.score.final.internet)

#Table for confusion Matrix for naive bayes model

compTable.nb.internet <- data.frame(Actual = internet.test$Churn, Predicted = nb.pred.final.internet)

#Looking at the naive bayes confusion Matrix

table(compTable.nb.internet)

###Decision Trees - All

##################

###Decision tree - all

dt <- rpart(Churn ~ ., teledf.train, method = "class")

#Predict the probablility on the test data frame

dt.predict <- predict(dt, teledf.test, "class")

#Get the probability of churned customer

#Table for confusion Matrix

compTable.dt <- data.frame(Actual = teledf.test$Churn, Predicted = dt.predict)

#Looking at the confusion Matrix

compTable.dt <- table(compTable.dt)

rpart.plot(dt, main = "Decision Tree for All Customers")

dt.accuracy.final <- (compTable.dt[1,1] + compTable.dt[2,2]) / sum(compTable.dt)

#put Overall recall for this model in the recall vector

dt.recall.final <- compTable.dt[2,2] / (compTable.dt[2,1] + compTable.dt[2,2])

#put precision accuracy for this model in the precision vector

dt.precision.final <- compTable.dt[2,2] / sum(compTable.dt[,2])

#put f1 score accuracy for this model in the f1 score vector

dt.f1.score.final <- 2 \* ((dt.precision.final \* dt.recall.final) / (dt.precision.final + dt.recall.final))

cat("Decision Tree Model quality (All Customers):\n\taccuracy: ", dt.accuracy.final, "\n\trecall: ", dt.recall.final,

"\n\tprecision: ", dt.precision.final, "\n\tF1 score: ", dt.f1.score.final)

#Looking at the naive bayes confusion Matrix

compTable.dt

###Decision tree - internet

dt.internet <- rpart(Churn ~ ., internet.train, method = "class")

#Predict the probablility on the test data frame

dt.predict.internet <- predict(dt.internet, internet.test, "class")

rpart.plot(dt.internet, main = "Decision Tree for Internet Customers")

#Table for confusion Matrix

compTable.dt.internet <- data.frame(Actual = internet.test$Churn, Predicted = dt.predict.internet)

#Looking at the confusion Matrix

compTable.dt.internet <- table(compTable.dt.internet)

compTable.dt.internet

dt.accuracy.final.internet <- (compTable.dt.internet[1,1] + compTable.dt.internet[2,2]) / sum(compTable.dt.internet)

#put Overall recall for this model in the recall vector

dt.recall.final.internet <- compTable.dt.internet[2,2] / (compTable.dt.internet[2,1] + compTable.dt.internet[2,2])

#put precision accuracy for this model in the precision vector

dt.precision.final.internet <- compTable.dt.internet[2,2] / sum(compTable.dt.internet[,2])

#put f1 score accuracy for this model in the f1 score vector

dt.f1.score.final.internet <- 2 \* ((dt.precision.final.internet \* dt.recall.final.internet) / (dt.precision.final.internet + dt.recall.final.internet))

cat("Decision Tree Model quality (Internet Customers):\n\taccuracy: ", dt.accuracy.final.internet, "\n\trecall: ", dt.recall.final.internet,

"\n\tprecision: ", dt.precision.final.internet, "\n\tF1 score: ", dt.f1.score.final.internet)

#Table for confusion Matrix for decision model

compTable.dt.internet <- data.frame(Actual = internet.test$Churn, Predicted = dt.predict.internet)

table(compTable.dt.internet)

######################################################################

Appendix B: Additional Visualizations

