**Logistic Regression to Assess Customer Churn**

Kelseyann Wright

Student ID: 00565814

College of Information Technology, Western Governors University

Predictive Modeling - D208

Dr. Eric Straw

April 28, 2023

**Logistic Regression to Assess Customer Churn**

**A1. Research Question**

The question that will be addressed is, "What variables are associated with customer churn?". This research question is important because understanding which factors determine if a customer “churns”, or cancels services, will allow the business to implement targeted changes to prevent customer churn.

**A2. Goals**

The goal of this analysis is to use a logistic regression to identify features that are related to customer churn to make predictions about how the related features will affect customer churn.

**B1. Summary of Assumptions**

The assumptions of a logistic regression are: 1) the outcome variable is binary, such as yes/no, 2) there is a linear relationship between each continuous variable and the log-odds of the outcome, 3) there are no strongly influential outliers, and 4) there is no multicollinearity.

**B2. Tool Benefits**

Python programming language was used to model the data. Python was chosen for its versatility, readability, and diverse functions. In addition, the syntax is consistent and simple to interpret, which is ideal when sharing your analysis with others who may not have coding experience. R does not have consistent syntax across libraries, making it more difficult to understand learn and understand.

Several python libraries were used, including pandas, numpy, matplotlib, scipy, sci-kit learn, and seaborn. Pandas is a common library used for working with data frames in python. It includes many tools for computations and analysis on data frames (McKinney, 2010). Numpy is used for many summary statistics throughout the analysis process (Harris et al., 2020; Larose & Larose, 2019). Scipy and sci-kit learn libraries work together to offer various statistical analyses (Pedregosa et al., 2011; Virtanen et al., 2020). Matplotlib and seaborn offer data visualization tools (Hunter, 2007; Waskom, 2021).

**B3. Appropriate Technique**

Logistic regression allows us to assess the relationship between a binary categorical variable, like customer churn, and multiple predictor variables of different types. This analysis will allow us to identify which variables have a significant relationship with customer churn and make predictions about customer churn based on other predictor variables.

**C1. Data Cleaning Goals**

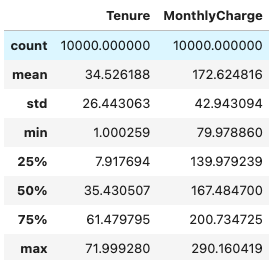
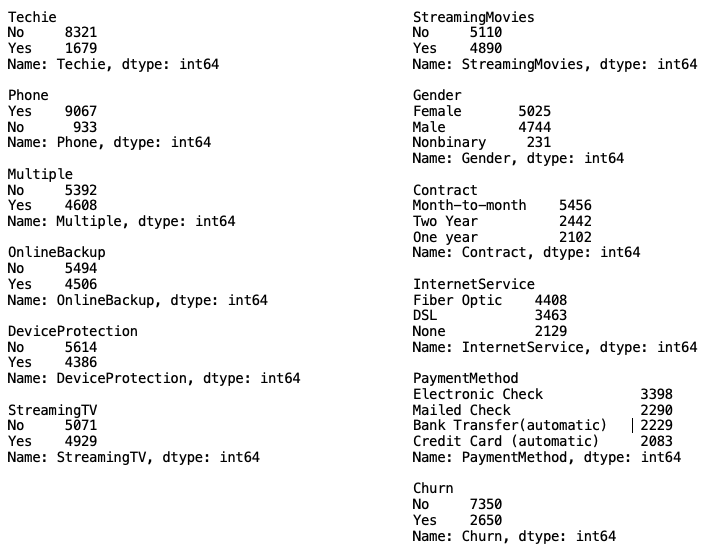
First, data was checked for missing values and duplicates. No missing values or duplicates were found in the provided dataset. Second, the categorical variables with high cardinality, specifically more than five groups, were dropped from the dataset. Variables with high cardinality can result in the “curse of dimensionality”, which results in sparse data in each unique group. This can reduce overall model performance because there are not enough data points in each unique group. In addition, high cardinality features can take up a lot of memory during model fit and cause computation issues. Therefore, those variables with high cardinality were removed to avoid issues with dimensionality and computer memory. The annotated code is attached.

Table : Summary Statistics of Relevant Quantitative Variables

**C2. Summary Statistics**

The dependent variable for the analysis is ‘Churn’. The independent variables chosen for the initial model are 'Techie', 'Phone', 'Multiple', 'OnlineBackup', 'DeviceProtection', 'StreamingTV', 'StreamingMovies', 'Tenure', 'MonthlyCharge', 'Gender’, 'Contract’ 'InternetService’, and 'PaymentMethod. These variables were identified using t-test statistics to identify potentially significant relationships between the variables and customer churn. In addition, a correlation heatmap was used to eliminate variables with multicollinearity.

The describe() function in python was used to calculate summary statistics (Table 1). The relevant categorical variables are not shown in the output as summary statistics are not calculated for categorical variables, only for quantitative variables. The mean customer tenure is 34.5 months, which means that customers stay enrolled in services for 34.5 months on average. The standard deviation is 26.4 months, with a minimum of 1 month and a maximum of almost 72 months. This means that there is a wide range in customer tenures. The mean falls close to the middle of the range. The mean monthly charge is 172.6 GB per year, with a standard deviation of 42.9 and a range from 79.98 to 290.16. As with the tenure variable, the mean falls close the middle of the range.

To summarize categorical variables, I used the value\_counts() function in python to count the number of customers in each group of the categorical variables. The output is provided below.

Diagram

Description automatically generated with low confidenceThe variables “Multiple”, “OnlineBackup”, “DeviceProtection”, “StreamingTV”, and “StreamingMovies” have relatively even distributions between groups. Most customers do not identify as “Techie”, and most customers do have phones. For gender, there is a relatively even distribution between males and females, with only about 2% of customers identifying as nonbinary. For the “Contract” variable, about half of the customers are month-to-month customers, while the other half is split fairly evenly between one-year and two-year contracts. Most customers have fiberoptic internet service, but 21% have no internet service at all. Around 34% of customers pay by electronic check, while the remaining customers are relatively evenly Background pattern

Description automatically generatedsplit between the other payment methods. For the “Churn” variable, there are almost three times as many “no” customers are there are “yes”.

Figure : Univariate graphs

**C3. Visualizations**

Univariate graphs for all variables are provided in figure 1. Bivariate graphs for all variables are provided in figure 2.

**C4. Data Transformation**

Categorical variables were transformed into dummy variables for the analysis. Dummy variables are binary values of 0 or 1, where 0 represents "no" or not being a member of that group and 1 represents "yes" or being a member of the group. Using dummy variables prevents us from having to make multiple models to represent different groups (Garavaglia and Sharma, 1998).

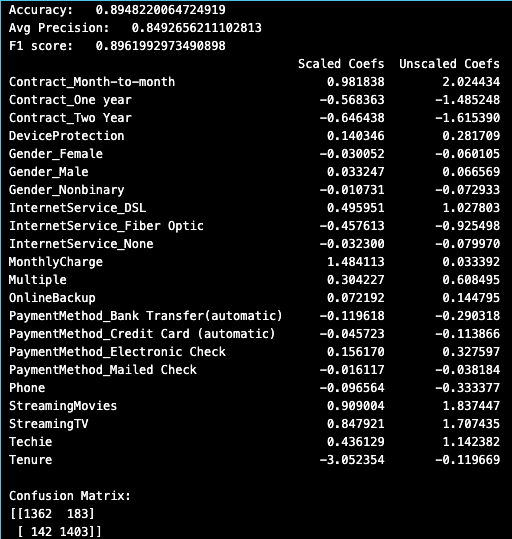
Figure 2: Bivariate graphs

After creation of dummy variables, the data were scaled using the StandardScaler() function from the sci-kit learn library in python (Pedregosa et. al 2011). This function scales features so that they will have a Gaussian normal distribution with a mean of zero and a standard deviation of 1. Scaling data is important so that all features carry the same weight in the model and so that regression coefficients can be directly compared between features. This is also important for the recursive feature elimination used during model reduction.

Finally. The initial dataset was unbalanced, with there being about one third as many “churn” customers as “no churn” customers. The data was balanced using the synthetic minority oversampling technique (SMOTE), as balanced data is important for logistic regression (Brownlee, 2020).

**C5. Prepared Dataset**

The final scaled and balanced dataset is attached.

**D1. Initial Model**

The initial model was run using the LogisticRegression() function in the sci-kit learn library. The regression output is provided to the right.

The assumption that there is a linear relationship between each continuous variable and the log-odds of the outcome was also assessed, and the assumption was met (Figure 3).

|  |  |
| --- | --- |
| **Tenure** | **Monthly Charge** |
| Correlation = 0.73 | Correlation = -0.57 |
|  |  |

Figure 3: Log-odds scatterplots for initial model

**D2. Justification of Model Reduction**

Recursive feature elimination (RFE) was used to identify which features should be dropped from the model. RFE eliminates features by fitting the model multiple times while recursively eliminating the weakest features based on their importance. RFE is preferred over stepwise feature selection because stepwise feature selection can sometimes include features that do not directly impact the dependent variable but are correlated with another feature that does (Smith 2018). The RFE function from the sci-kit learn python library was used for this analysis (Pedregosa et al. 2011).



**D3. Reduced Logistic Regression Model**

The reduced logistic regression model output is provided to the right. The assumption that there is a linear relationship between each continuous variable and the log-odds of the outcome was also assessed, and the assumption was met (Figure 4).

|  |  |
| --- | --- |
| **Tenure** | **Monthly Charge** |
| Correlation = 0.74 | Correlation = -0.57 |
| Figure 4: Log-odds scatterplots for reduced model |  |

**E1. Model Comparison**

RFE removed eleven of the features from the initial model, resulting in a logistic regression model with eleven independent variables: Contract\_Month-to-month', 'Contract\_One year', 'Contract\_Two Year', 'InternetService\_DSL', 'InternetService\_Fiber Optic', 'MonthlyCharge', 'Multiple', 'StreamingMovies', 'StreamingTV', 'Techie', and 'Tenure'. The accuracy, average precision, and F1 score all increased slightly in the reduced model. The accuracy increased from 0.8948 to 0.8958. The average precision increased from 0.849 to 0.850. Finally, the F1 score increased from 0.896 to 0.897. The increases in these metrics are small but do show improvement in the reduced model.

**E2. Output and Calculations**

The confusion matrix and accuracy for the initial and final reduced models are provided below (Table 2). The accuracy of the final model improved slightly, from 0.8948 to 0.8958. The confusion matrix shows that the reduced model had fewer false negatives than the initial model, which resulted in the slight increase in model accuracy.

*Table 2: Calculation Summary*

|  |  |  |
| --- | --- | --- |
|  | **Initial Model** | **Reduced Model** |
| **Accuracy** | 0.8948 | 0.8958 |
| **Confusion Matrix** | 1362 183  142 1403 | 1362 183  139 1406 |

**E3. Code**

The python code written in Jupyter lab notebook is attached.

**F1. Results**

The reduced regression model equation with unscaled coefficients is:

*Logit(p) = 1.98\*Contract\_Month-to-month – 1.46\*Contract\_One year – 1.58\*Contract\_Two Year + 1.00\*InternetService\_DSL – 1.04\*InternetService\_Fiber Optic + 0.04\*MonthlyCharge + 0.40\*Multiple   
+ 1.50\*StreamingMovies + 1.43\*StreamingTV + 1.13\*Techie – 0.12\*Tenure*

Where p is the probability that ‘Churn’ is equal to 1, or “yes”.

Text

Description automatically generatedFor logistic regressions, the interpretation of coefficients can be made more clear by calculating the odds and odds ratios from the coefficients. Odds for continuous variables and odds ratios for categorical variables are calculated as *O = eC*, where O is the odds or odds ratio and C is the coefficient. The odds and odds ratios for each independent variable of the reduced model are provided to the right. These values are used to interpret the coefficients of the reduced model.

When interpreting the coefficients, it is important to separate categorical and continuous variables. The only continuous independent variables in this regression are ‘MonthlyCharge’ and ‘Tenure’. A one unit change in ‘MonthlyCharge’ would increase the natural log of the probability of customer churn occurring (‘Churn’=1) by 0.04, or multiply the odds of customer churn by 1.04. A one unit change in ‘Tenure’ would decrease the natural log of the probability of customer churn occurring (‘Churn’=1) by 0.12, or multiply the odds of customer churn by 0.88.

For the remaining independent variables, which are all categorical, the natural log of the probability of customer churn will not change if the value of the categorical variable is zero. When the value of ‘Contract\_Month-to-month’ is 1, the natural log of the probability of churn would increase by 1.98, and costumers with monthly contracts have 7.25 times the odds of customer churn than the other contract types. If the value of ‘Contract\_One year is 1, the natural log of the probability of churn would decrease by 1.46, and this group of customers has 0.23 times the odds of churn. When the value of ‘Contract\_Two year is 1, the natural log of the probability of churn would decrease by 1.58 and the customers have 0.20 times the odds of churn. When the value of ‘InternetService\_DSL’ is 1, the natural log of the probability of churn increases by 1.00 and the customers have 2.7 times the odds of churn. When the value of ‘InternetService\_Fiber Optic’ is 1, the natural log of the probability of churn decreases by 1.04 and the customers have 0.35 times the odds of churn. When the value of ‘Multiple’ is 1, the natural log of the probability of churn increases by 0.40 and the customers have 1.5 times the odds of churn. When the value of ‘StreamingMovies’ is 1, the natural log of the probability of churn increases by 1.50 and the customers have 4.5 times the odds of churn. When the value of ‘StreamingTV’ is 1, the natural log of the probability of churn increases by 1.43 and the customers have 4.2 times the odds of churn. Finally, when the value of ‘Techie’ is 1, meaning the customer identifies as technically inclined, the natural log of the probability of churn increases by 1.13 and the customers have 3.1 times the odds of churn.

Based on the high accuracy, precision, and F1 score all being close to 1, the model is statistically significant. The model is also practically significant, because it tells us which variables increase the odds of customer churn and which variables reduce the odds of customer churn, and by how much the odds are increased or reduced. The model can be used to predict the odds of customer churn based on their contract type, type of internet service, monthly charge, if they have multiple services, if they are technically inclined, if they stream TV and/or movies, and their tenure. The direct comparison of the scaled coefficients implies that tenure, monthly charge, and whether a customer is on a month-to-month contract have the most significant impact on the probability of customer churn. This information will allow the company to make changes to sales, marketing, and services in an attempt to retain customers and reduce churn.

One of the limitations of this analysis is we lack information to identify the causality of these relationships, as correlation does not imply causation. Further statistical testing would need to be done in order to determine causality. In addition, the initial dataset was unbalanced, with there being about one third as many “churn” customers as “no churn” customers. The data was balanced using an oversampling technique in order to complete the analysis, which may impact the results.

**F2. Recommendations**

My recommendation for next steps would be for the company to offer promotions for customers that switch from monthly contracts to longer contracts. I suggest this because month-to-month contract customers have much higher odds of customer churn, as do customers with high monthly bills. Offering a promotion to reduce monthly bills but increase contract length would address both of these variables. In addition, I would suggest that we research why customers who stream movies and TV may be more likely to churn – is it because the internet service is unreliable and their streams are interrupted? Further research, including additional customer surveys and data exploration, to identify why some of these independent variables may be correlated with customer churn is a reasonable next step.

**G. Panopto Video**

A Panopto video was attached and submitted with the performance task. The link can also be accessed here: <https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=f90b5a04-6681-486c-b7e8-aff3016ac841>

1. **Sources of Third-Party Code**

Brownlee, J. (2020). *SMOTE for Imbalanced Classification with Python*. Machine Learning Mastery. <https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/>

1. **Sources**

Garavaglia, S., & Sharma, A. (1998). A smart guide to dummy variables: Four applications and a macro. In Proceedings of the northeast SAS users group conference (Vol. 43).

Harris, C.R., Millman, J.K., van der Walt, S.J, Gommers, R., Virtanen, P., Cournapeau, V., Wieser, E., Taylor, J., Berg, S., Smith, N.J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M.H., Brett, M., Haldane, A., Fernandez del Rio, F., Wiebe, M., Peterson, P., Gerard-Marchant, P., Sheppard, K., … Oliphant, T.E. (2020). Array programming with NumPy. Nature, 585(7825), 357-362.

Hunter, J.D., "Matplotlib: A 2D Graphics Environment", Computing in Science & Engineering, vol. 9, no. 3, pp. 90-95, 2007.

Larose, C. D., & Larose, D. T. (2019). Data science using Python and R. Wiley.

McKinney, W. (2010). Data structures for statistical computing in python. In Proceedings of the 9th Python in Science Conference (Vol. 445, pp. 51–56).

Pedregosa, F., Varoquaux, Ga"el, Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., and Dubourg, V. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12(Oct), 2825–2830.

Smith, G. (2018). Step away from stepwise. J Big Data 5, 32.

Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S.J., Brett, M., Wilson, J., Millman, K.J., Mayorov, N., Nelson, A.R.J., Jones, E., Kern, R., Larson, E., … van Mulbregt,P. (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. Nature Methods, 17(3), 261-272.

Waskom, M. L. (2021). Seaborn: statistical data visualization. Journal of Open Source Software, 6(60), 3021.