Romand Lansangan

Predicting Customer Churn with Regression-Based and Tree-Based Methods

DSC 1107 CAPSTONE PROJECT

May 9, 2025

3rd Year Applied Mathematics (Data Science)

Table of Contents

1. Introduction & Problem Motivation

As a private company, it is important for *Orange Telecom* to study the behavior of their clients, even more so to those who chose to depart their services, or the `*churners*`. To gain the knowledge as to why churners decide to leave is to gain key information to develop a better business model for the company. A better business model that not only retain its clients but also a business model that could be used to entice the market.

In this data modelling report, the data modeler utilized ***Orange Telecom Churn Dataset*** to build and evaluate machine learning models that predict whether a customer will churn. The modeler used both regression-based (particularly *Logistic Regression*) and tree-based (particularly *Random Forest*) to attain the objectives. The modeler also provided a comparison in the performance of both models and some nuance consideration.

2. Exploratory Data Analysis

2.1 Dataset

The ***Orange Telecom Churn Dataset*** is a dataset consisting of cleaned customer data together with their churn label that specify whether a customer cancelled their subscription on the company. The dataset has 3,333 rows and 20 columns. This dataset comprises the following variables:

	dtypes	nunique	unique
State	object	51	[KS, OH, NJ, OK, AL, MA, MO, WV, RI, IA, MT, I...
Account length	int64	205	[128, 107, 137, 84, 75, 118, 121, 147, 141, 74...
Area code	int64	3	[415, 408, 510]
International plan	object	2	[No, Yes]
Voice mail plan	object	2	[Yes, No]
Number vmail messages	int64	42	[25, 26, 0, 24, 37, 27, 33, 39, 41, 28, 30, 34...
Total day minutes	float64	1489	[265.1, 161.6, 243.4, 299.4, 166.7, 223.4, 218...
Total day calls	int64	115	[110, 123, 114, 71, 113, 98, 88, 79, 84, 127, ...
Total day charge	float64	1489	[45.07, 27.47, 41.38, 50.9, 28.34, 37.98, 37.0...
Total eve minutes	float64	1442	[197.4, 195.5, 121.2, 61.9, 148.3, 220.6, 348....
Total eve calls	int64	120	[99, 103, 110, 88, 122, 101, 108, 94, 111, 148...
Total eve charge	float64	1301	[16.78, 16.62, 10.3, 5.26, 12.61, 18.75, 29.62...
Total night minutes	float64	1444	[244.7, 254.4, 162.6, 196.9, 186.9, 203.9, 212...
Total night calls	int64	118	[91, 103, 104, 89, 121, 118, 96, 97, 94, 128, ...
Total night charge	float64	885	[11.01, 11.45, 7.32, 8.86, 8.41, 9.18, 9.57, 9...
Total intl minutes	float64	158	[10.0, 13.7, 12.2, 6.6, 10.1, 6.3, 7.5, 7.1, 1...
Total intl calls	int64	21	[3, 5, 7, 6, 2, 4, 19, 10, 9, 15, 8, 1, 11, 0....
Total intl charge	float64	158	[2.7, 3.7, 3.29, 1.78, 2.73, 1.7, 2.03, 1.92, ...
Customer service calls	int64	10	[1, 0, 2, 3, 4, 5, 7, 9, 6, 8]
Churn	bool	2	[False, True]

Table 1: Telecommunications account metrics showing data types, unique value counts, and sample values across call patterns, billing information, and customer attributes.

2.2 Split

The data was split into two sets, training set and test set, with a proportion of 80:20. This was done before any type of processing to prevent data leakage.

2.3 Missing Data

There was no missing data in the dataset.

2.4 Distributions

2.4.1 Categorical Variables

The modeler defined the categorical variable as a column that has a cardinality of 21 or less.

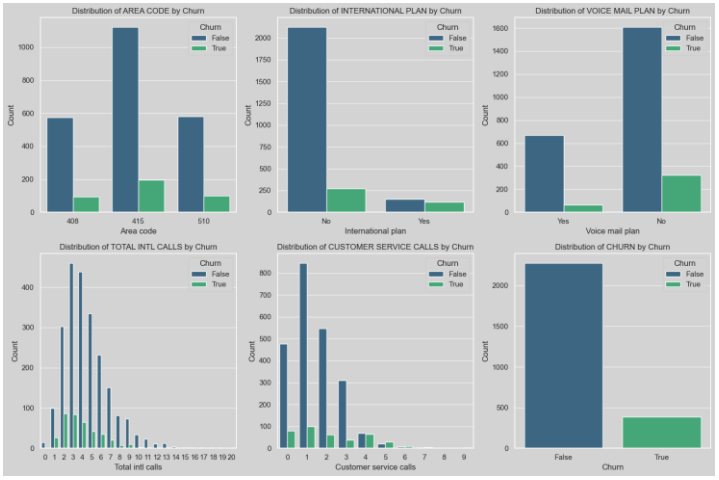

Figure 1: Distribution plots segmented by churn status for the categorical variables, showing differences between customers who churned (True) versus those who stayed (False).

Both Figure 1 and Figure 2 show the imbalance between churners and non-churners. It becomes clear that a strict resampling technique was imperative. Also, the distribution of "Total int calls" and "Customer service calls" showed that some categories are rarer compared to their peers. To prevent the rows under such rare categories from being seen as *pseudo-outliers*, the modeler decided to group them together. For "Total int calls," those who are 11 and above were grouped together. For the "Customer service calls," those who are 1 and below are grouped together as well as those who are 4 and above.

For the processing of categorical variables (with cardinality of greater than 2), the modeler decided to use one-hot encoding.

2.4.2 Continuous Variables

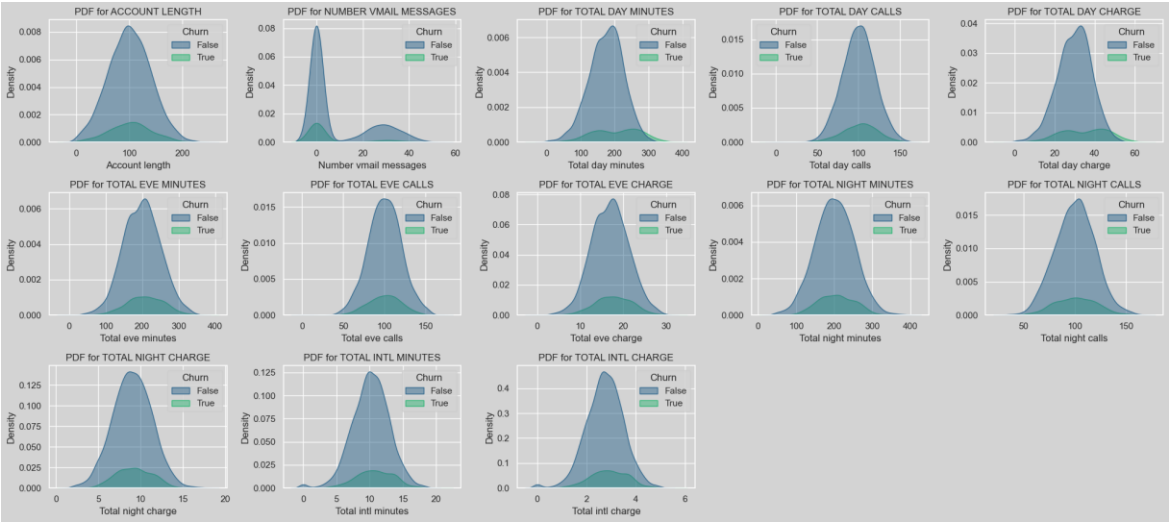

Figure 4: Probability density function (PDF) of continuous variables segmented by churn status, highlighting differences in usage patterns between customers who churned versus those who remained.

All are normally distributed except “Number vmail messages” which is skewed to the right. However, this would not be an issue whatsoever since both random forest and logistic regression doesn’t assume normality of predictors.

For the processing of the continuous variables, the modeler decided to use *Standard Scaler* which calculates each entries z-scores relative to their own column.

2.5 Outliers

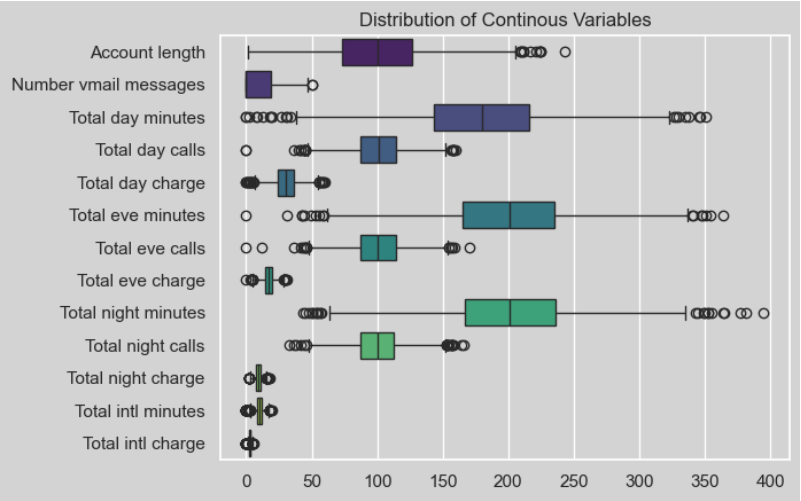

Figure 5: Box plot showing the distribution of continuous telecommunications variables, with metrics like account length and calling patterns (minutes, calls, charges) across day, evening, night, and international categories.

Account length	0.004501
Number vmail messages	0.000750
Total day minutes	0.007877
Total day calls	0.006752
Total day charge	0.007877
Total eve minutes	0.006377
Total eve calls	0.005626
Total eve charge	0.006377
Total night minutes	0.008252
Total night calls	0.007127
Total night charge	0.008252
Total intl minutes	0.013878
Total intl charge	0.015004

Table 2: Proportion of outliers in each continuous variable calculated using IQR method (1.5× multiplier). International charge (1.5%) and minutes (1.4%) show the highest outlier percentages.

The outliers in “Total International charge” and “Total International Minutes” were removed (40 rows). The rest were retained due to the insignificance of the frequency and to the fact that they might contain some key information for the model to be robust to outliers.

2.6 Correlation

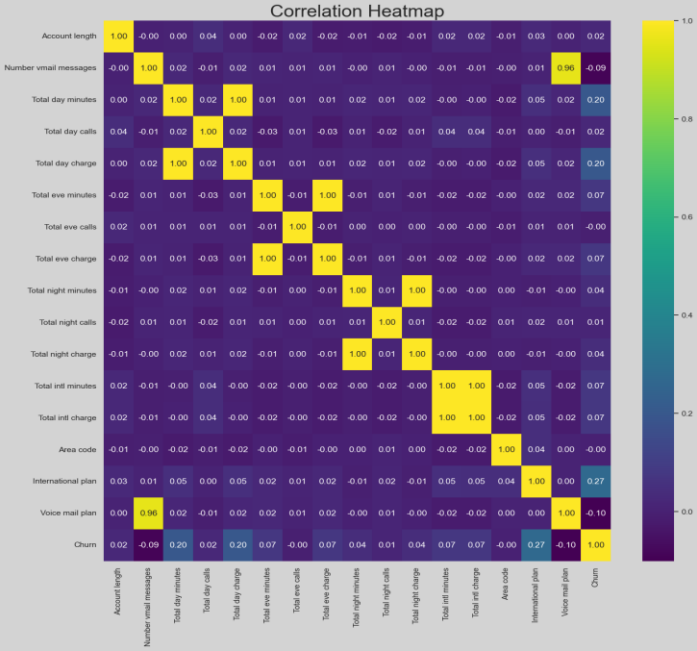

Figure 6: Correlation heatmap of telecommunications variables

Some “total minutes” and their corresponding “total charges” are perfectly correlated. It means that it is safe to drop any of the pair. The modeler decided to drop the “total charges” columns. It is also the case that “Number vmail messages” is highly correlated to “Voice mail plan” with 0.96 Pearson correlation. The modeler decided to drop the “Voice mail plan” since its correlation to the response variable “Churn” is lower (-0.09) than the that of “Number vmail messages” (-0.10).

2.6 Wrangle Function

The wrangle function can be found on the notebook file. But the function does the following:

1. Ask for data and transformer object (*StandardScaler*)
2. If the *StandardScaler* object is present, the function is in test mode. Which means that outliers won’t be dropped and *StandardScaler* wouldn’t be instantiated but rather the give transformer will be used to standardized continuous variables.
3. If there’s no *StandardScaler* object however, the function is in train mode. This means that a new *StandardScaler* will be trained based on the *mean* and *std* of the training data and outliers mentioned in Section 2.5 will be removed.
4. Process the categorical as was indicated in Section 2.4.1.
5. Process the continuous variable as was indicated in Section 2.4.2.
6. Map the binary variables into 0s and 1s.
7. Dropped the following variables: 'Customer service calls', 'Total intl calls', 'Voice mail plan', 'Total day charge', 'Total eve charge', 'Total night charge'
8. Return both the cleaned data and the *StandardScaler* object.

Rows decreased to 2666 in the training set (no removal for test set) and number of columns increased to 27.

3. Modeling

3.1 Baseline

For the baseline, the modeler calculated for the normalized frequency of the majority class in test set, which is labeled as "0"or the *non-churners* with about 0.857571. Meaning, the baseline model will predict everything as 0 and will be right by about 85.76% percent of the time.

3.2 Resampling

To increase the representation of the minority class, resampling is in order. The modeler decided to use a hybrid resampling technique that combines Synthetic Minority Over-sampling Technique (SMOTE) which is an oversampling with awareness of its neighbors and Edited Nearest Neighbors (ENN) which is an under sampling technique that remove some noise and clean the boundaries between classes.

Training set went from 2666 rows to 3473 total rows.

3.3 Model Fitting

For the model, the modeler decided to resort to Logistic Regression (for regression-based) and Random Forest (for tree-based). The modeler used grid search with 5 folds to find the best combination of parameters for each model.

3.4 Best Parameters

Logistic Regression	Random Forest
{'C': 1, 'penalty': 'l2', 'solver': 'newton-cg'}	{'bootstrap': True, 'criterion': 'entropy', 'max_depth': None, 'max_features': 'log2', 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 200}

Table 3: Best parameters for each model after hyperparameter tuning.

4. Model Evaluation

4.1 Accuracy and Log Loss

	model	resampled_train	unsampled_train	test_set	test_set_log_loss
0	baseline	0.367982	0.853770	0.857571	5.133654
1	logistic regression	0.894328	0.790175	0.755622	0.599926
2	random forest	1.000000	0.905560	0.820090	0.419354

Table 4: Model performance comparison showing that while baseline performed the best at test set (85.76%; which is to be expected given the imbalance). The random forest achieves the highest accuracy with 100% in resampled train, 90.56% on unsampled train, 82.01% on test and the lowest log loss (0.419), outperforming logistic regression and baseline models.

4.2 Classification Report

Logitic Regression Model:

	precision	recall	f1-score	support
0	0.94	0.76	0.84	572
1	0.33	0.71	0.45	95
accuracy			0.76	667
macro avg	0.64	0.73	0.65	667
weighted avg	0.85	0.76	0.79	667

Table 4: Logistic Regression classification metrics

Random Forest Model:

	precision	recall	f1-score	support
0	0.97	0.82	0.89	572
1	0.43	0.82	0.57	95
accuracy			0.82	667
macro avg	0.70	0.82	0.73	667
weighted avg	0.89	0.82	0.84	667

Table 3: Random Forest classification metrics

Random Forest model upped the logistic regression by a wider margin for all metrics: accuracy, precision, recall, and f1-score.

4.3 Confusion Matrix

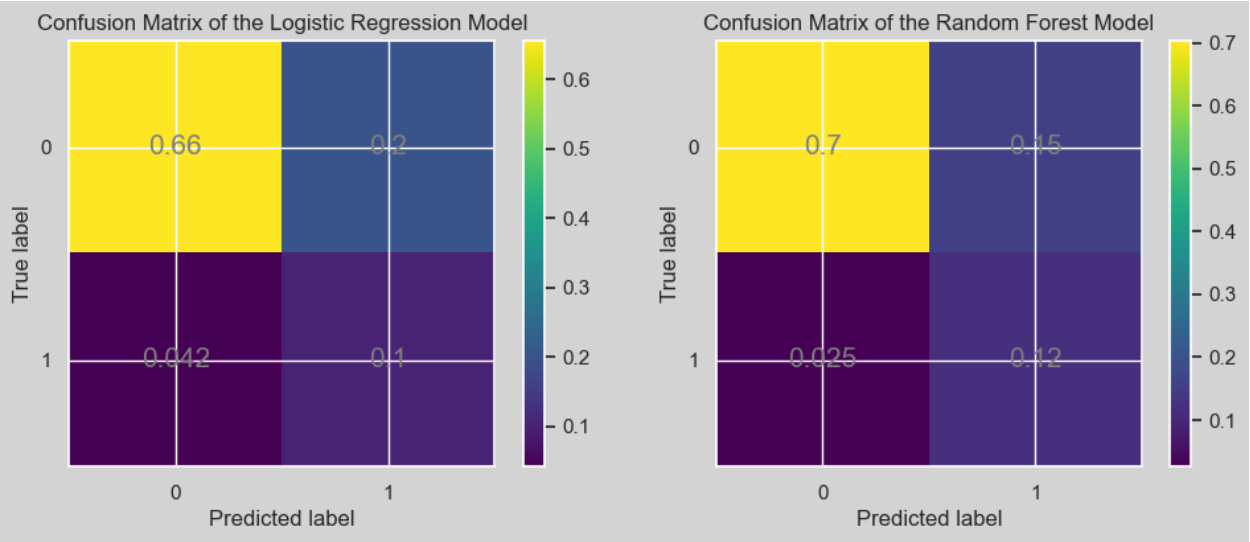

Figure 7: Confusion matrices comparing model performance. Random Forest (right) shows better overall accuracy with 70% true negatives and 12% true positives, while Logistic Regression (left) shows 66% true negatives and 10% true positives.

4.4 Feature Importance

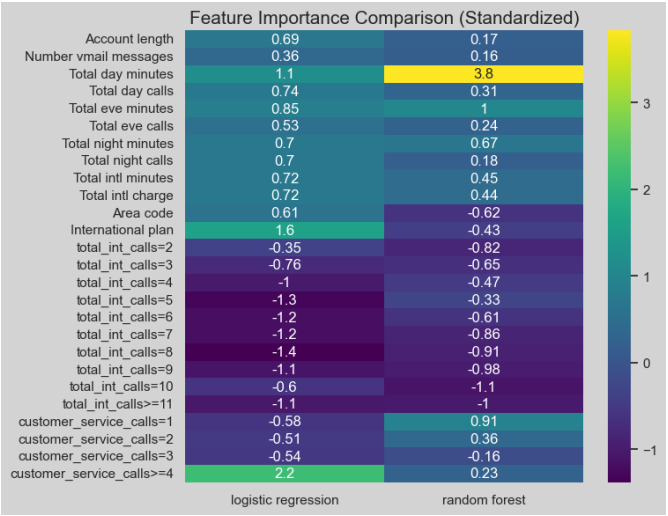

Figure 8: Feature importance comparison between logistic regression and random forest models. Importances for each model were standardized according to the mean and standard deviation of features under the same model.

A clear difference between the Logistic Regression and Random Forest model can be asserted by looking at Figure 8. Figure 8 showed that as opposed to a more distinct coefficient distribution of Logistic Regression, Random Forest showed a clear bias to a feature then a vague distinction to everything else. “Total day minutes” has highest coefficient for Random Forest with 3.8 standard deviations from the mean. This is significantly farther than the highest coefficient in Logistic Regression that is “custumer_service_call>=4.” “Total day minutes” is just about 1.1 standard deviation away from the mean

in Logistic Regression. This is the most glaring difference between the two results. Perhaps Random Forest has found a non-linear relationship between this predictor and the response.

5. Results and Recommendations

It is clear from the results that **Random Forest** performed better than the Logistic Regression model. Quantitative metrics favored the Random Forest by a significant margin; accuracy, log loss, precision, recall, and even f1-score. However, before making any immense business decision, a discussion of nuances is in order.

5.1 Predictive Power vs Interpretability

One of the most surprising results in this report is all (except one) of the coefficients of Random Forest lie around 1 or -1 standard deviation away from the mean (or closer). While the Logistic Regression showed more features farther than the said interval. This sheds light on how linear regression tends to find the distinction between each feature and their importance (recall the 'l2' penalty from section 3.4). This ought to be helpful for feature selection. Random Forest seemed to have a less distinct boundaries between feature importance. This is an important consideration when picking between the two models because there is a trade-off between accuracy metrics and interpretability.

Interpretability is particularly important when the company *Orange Telecom* will be looking for the specific aspects/behaviors of customers who were churners. **If the budget is limited and aims to address only few behaviors of customers to make key business decisions, Logistic Regression might just be the better choice since the distinction between feature importance is clearer in Logistic Regression.** However, if the company is looking for a specific model to predict or flag a probable churner, the following sections ought to be considered.

5.2 Bias-Variance trade-off

The glaring difference between the two models lies within their assumptions. The Random Forest does not assume any form of relationship between predictor and response. This means that it could detect more complex relationship (recall the "Total day minutes" coefficient). This is not the case for Logistic Regression which does best if the relationships of features are approximately linear. That is the same characteristic that makes the Logistic Regression much simpler to implement and safer to generalize.

Assuming any form means that the model leans on bias while assuming nothing ensues high variance. Random Forest, being the more variant of the two, tends to overfit and may not generalize well when not implemented properly. Random Forest might be prone to overly familiarizing itself with the training data by considering even the noise. This was shown by the perfect accuracy of Random Forest in training set while being less accurate by around 18% in the test set. However, Logistic Regression, being the more biased of the two, may tend to underfit and may also not generalize well if the relationship is not linear in the first place.

The caveat? If the difference is not much, stick with the simpler one, Logistic Regression, because it is the safer choice of the two. **However, in the case of the *Orange Telecom Churn Dataset,* the difference is significant enough to favor Random Forest.**

6. Conclusion

Both Random Forest and Logistic Regression provided distinctive results and either could be use (depending on the use case) in predicting churners using the *Orange Telecom Churn Dataset*. The modeler started with the observation of distributions for each variable and found a significant imbalance between churners and non-churners (20:80). After cleaning and resampling, the data was fitted to both Random Forest and Logistic Regression. The two showed significant improvements from the baseline in terms of accuracy and log loss. However, among Logistic Regression and Random Forest, the latter was flagged as the more robust of the two by all predictive metrics—accuracy, log loss, recall, precision, and f1-score. Although Logistic Regressiion has the advantage in terms of interpretability (which is beneficial for making key business decisions), a nuanced consideration to variance-bias tradeoff declared the Random Forest as the superior model in terms of predicting churners and non-churners. It thus becomes imperative to ponder upon which aspects of modelling is prioritized, interpretability or maximum predictive performance.

9. Github Link