Waze Churn Prediction

capstone project

João Carvalho


waze

About the Company:

Waze is a free collaborative navigation app, founded in 2009, that's makes it easy for drivers around the world to find their destinations.

Waze's community of map editors, beta testers, translators, partners, and users helps make every trip better and safer.

Project overview ↘

In a fictitious scenario, Waze leadership asked its data team to develop a data model to predict user churn. Churn quantifies the number of users who uninstall the Waze app or stop using it.

This project focuses on monthly user churn. An accurate model will help prevent churn, improve user retention, and grow Waze's business.

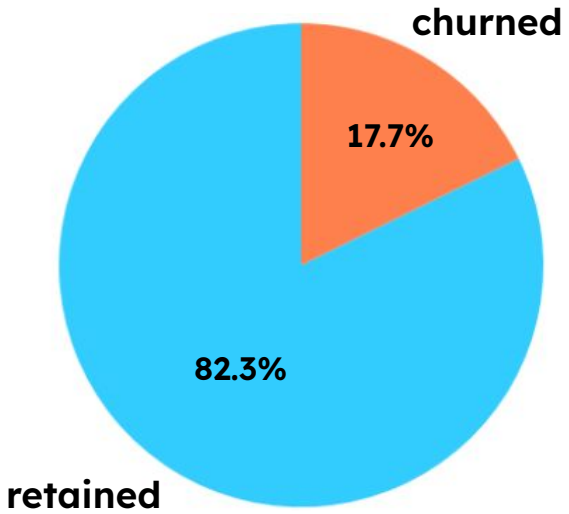
Main objectives

- Discovery the available data
- Make a exploratory data analysis (EDA)
- Identify possible different **habits and motivations** of users who churn
- Perform **feature engineering** to improve data for modeling
- Create and test machine learning models
- **Evaluate** results with score metrics

Data discovering

Retained vs Churned users

The dataset contains **~82%** retained users and **~18%** churned users.



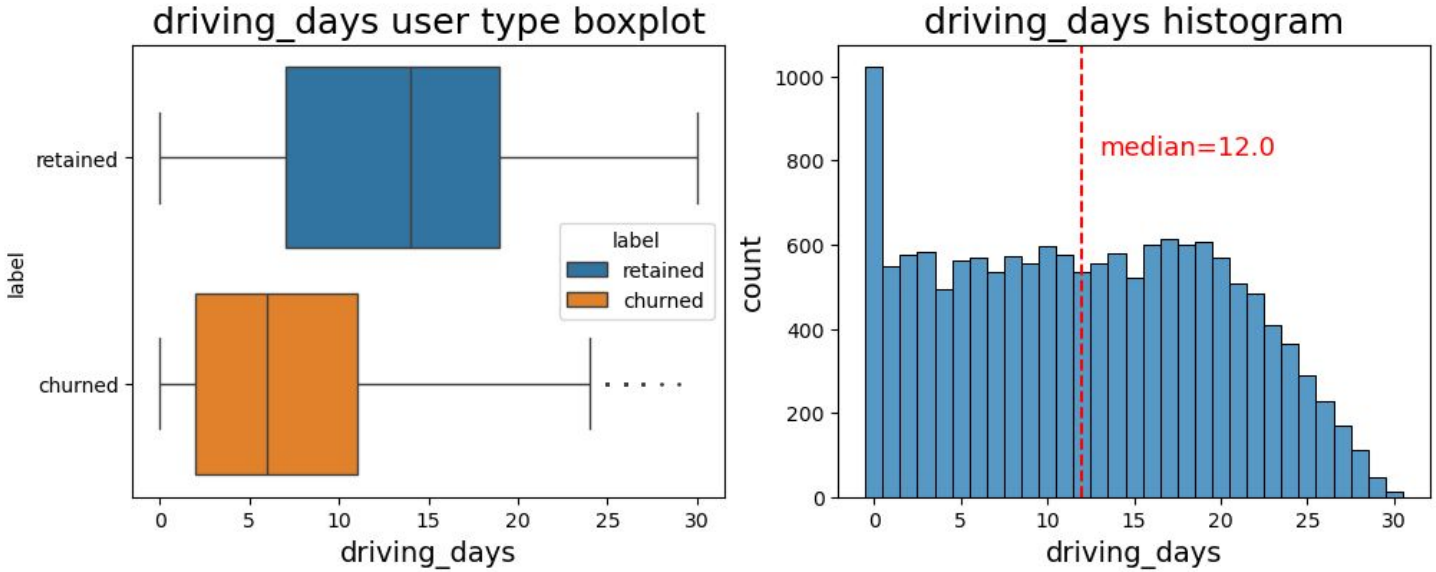
Data discovering

First key findings

- **Churned** users averaged **~3 more drives** in the last month than retained users.
- **Retained** users used the app on over **twice as many days** as churned users in the last month.
- The median **churned** user **drove ~200 more kilometers** and **2.5 more hours** during the last month than the median retained user.
- The median user who **churned drove 698 kilometers** each day they drove last month, which is about **240% the per-drive-day** distance of **retained** users.
- **Churned** users had **more drives in fewer days**, and their trips were **farther** and **longer** in duration.

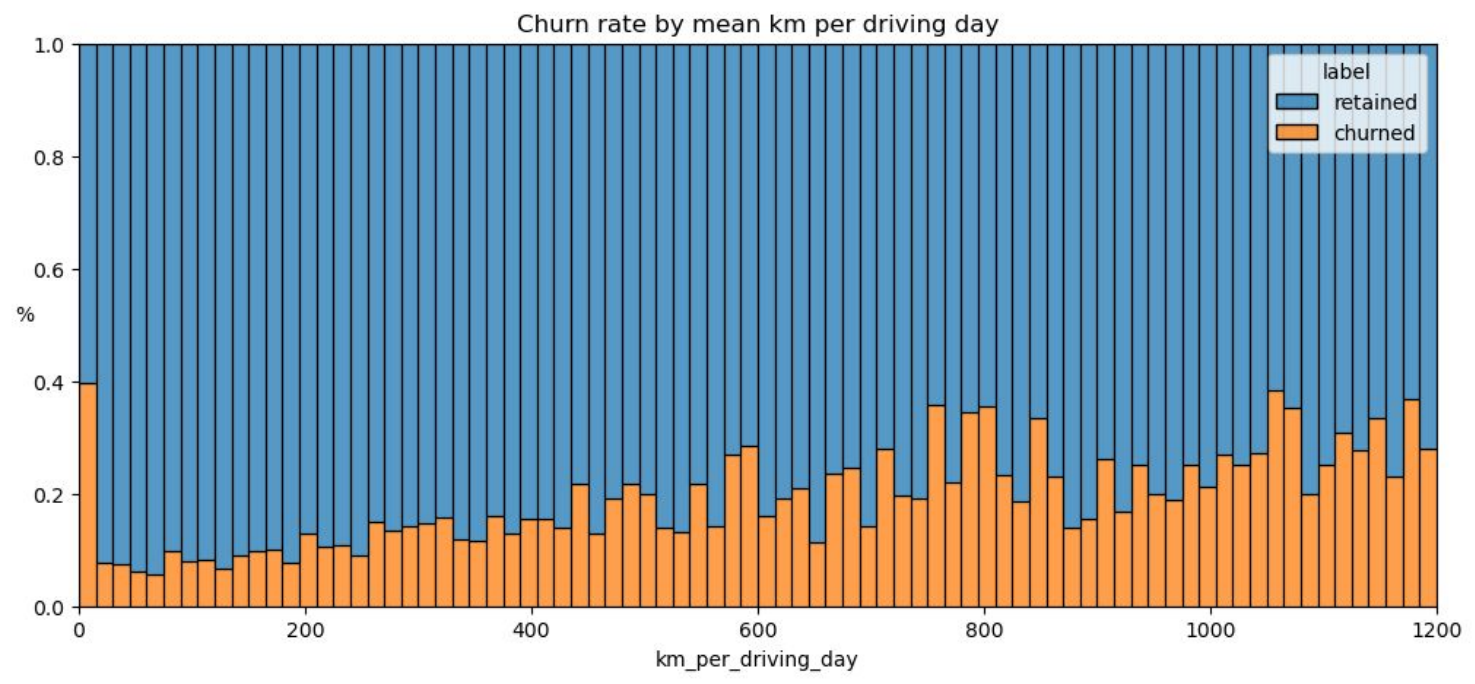
Let's dig deeper into the data to gain more insights

Exploratory data analysis (EDA)

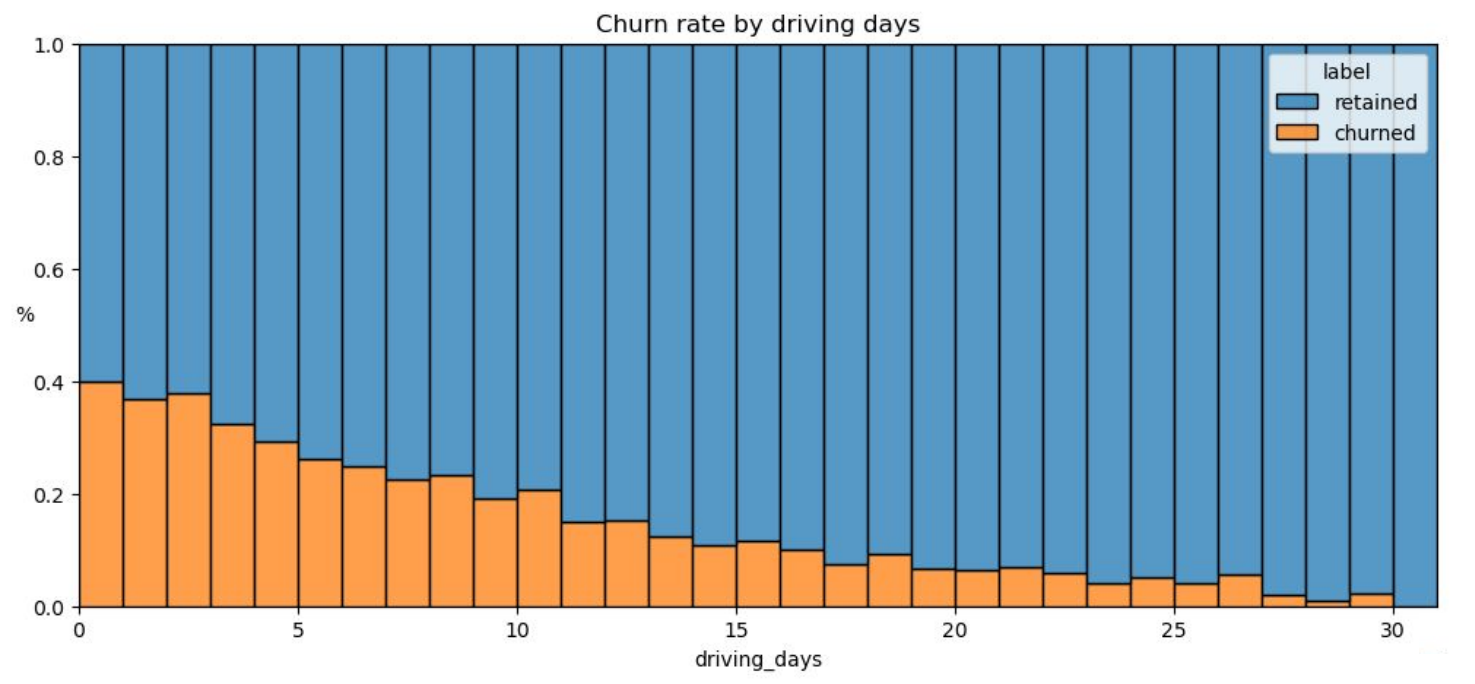


There were almost twice as many users (~1,000 vs ~500) who didn't drive at all during the month. **Churn users drive even fewer days.**

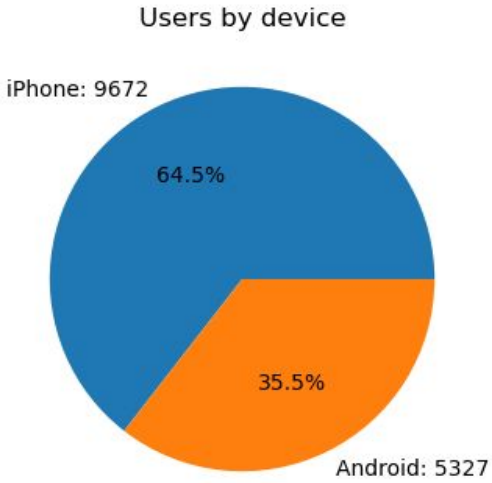
Confirming what was found before, the churn rate tends to increase as the mean daily distance driven increases.



The more times users used the app, the less likely they were to churn. While 40% of the users who didn't use the app at all last month churned, nobody who used the app 30 days churned.

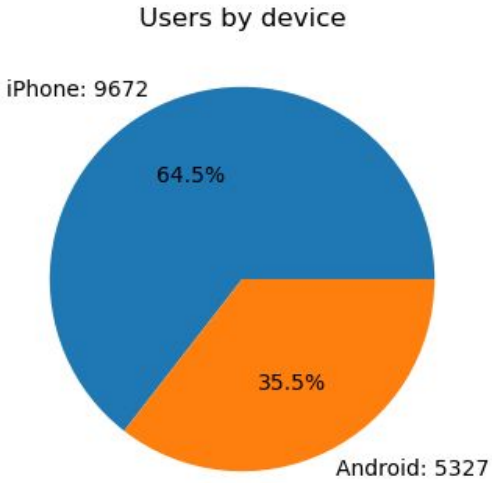


Exploratory data analysis (EDA)



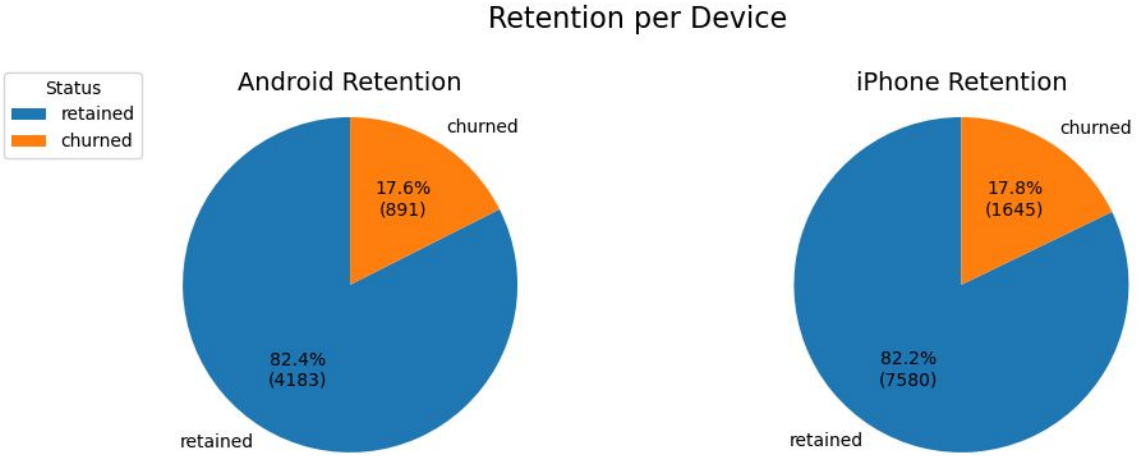
There are nearly **twice as many iPhone** users as **Android** users represented in this data.

Exploratory data analysis (EDA)



There are nearly **twice as many iPhone** users as **Android** users represented in this data.

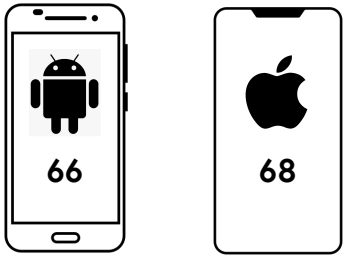
The proportion of churned users to retained users is consistent between device types.



Hypothesis test

During the analysis, it was discovered that the average number of trips for iPhone and Android users were slightly different.

Average Number of Drives



Note: The mean number of drives shown here – 66 for Android and 68 for iPhone – have been rounded up.

A two-sample hypothesis test was made to analyze and determine whether there is a statistically significant difference between mean number of rides and device type.

With a chosen **significance level of 5%.**

The t-test results concluded **there is not a statistically significant difference in mean number of rides between iPhone users and Android users.**

building machine learning models

Machine Learning models

The following models were implemented and compared:

1. Logistic Regression – baseline linear model.
2. Random Forest – ensemble method with decision trees.
3. XGBoost – gradient boosting model for better performance

Machine Learning models

Recall Score was the most important metric in all models, as it measures how many users the model correctly identifies who will churn.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives + False Negatives}}$$

Machine Learning models

Feature engineering

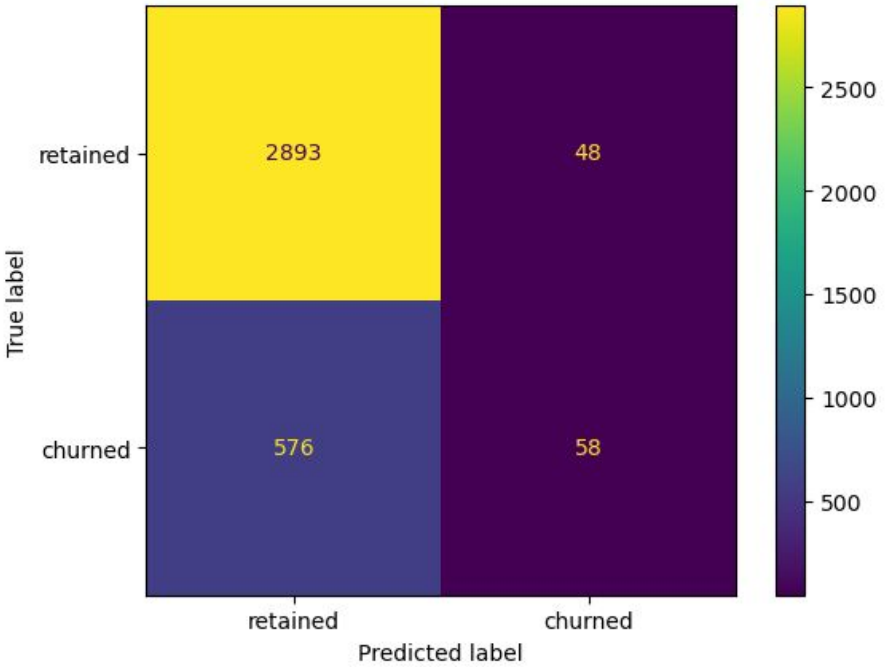
To improve the performance of predictive models, feature engineering is an important process that transforms raw data into more relevant and informative features.

The following columns were added to the dataset:

- km_per_driving_day
- percent_sessions_in_last_month
- professional_driver
- total_sessions_per_day
- km_per_hour, km_per_drive
- percent_of_sessions_to_favorite

Machine Learning models

Logistic Regression | Random Forest | XGBoost



True Retained (TN): 2,893
False Churns (FP): 48
False Retained (FN): 576
True Churns (TP): 58

Machine Learning models

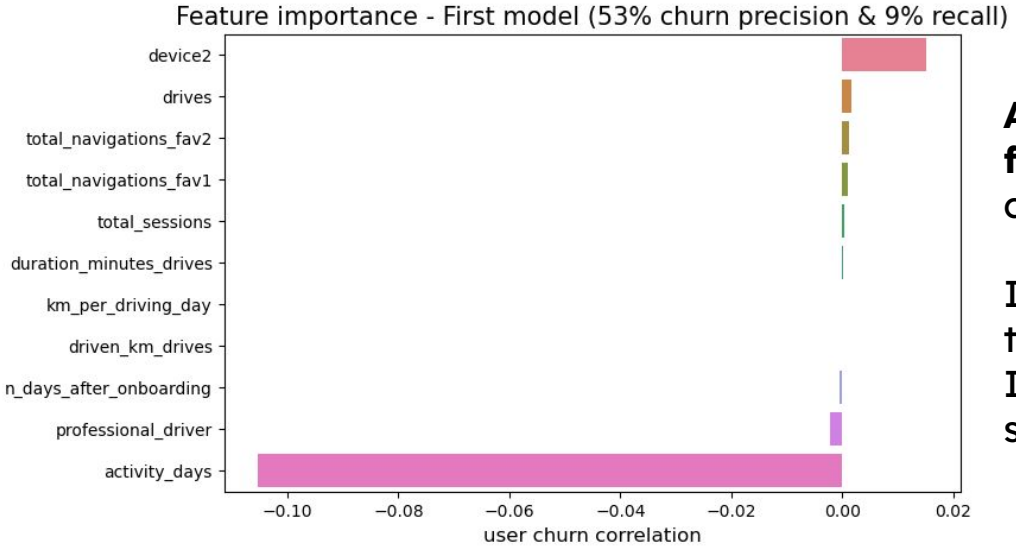
Logistic Regression | Random Forest | XGBoost

model	precision	recall	f1-score	accuracy
Logistic Regression	55%	9%	16%	82%

The model has mediocre precision (55% of its positive predictions are correct) but very low recall, with only 9% of churned users identified. **This means the model makes a lot of false negative predictions and fails to capture users who will churn.**

Machine Learning models

Logistic Regression | **Random Forest** | **XGBoost**



Activity_days was by far the most important feature in the model. It had a negative correlation with user churn.

In previous EDA, user churn rate increased as the values in km_per_driving_day increased. In the model, distance driven per day was the second-least-important variable.

Machine Learning models

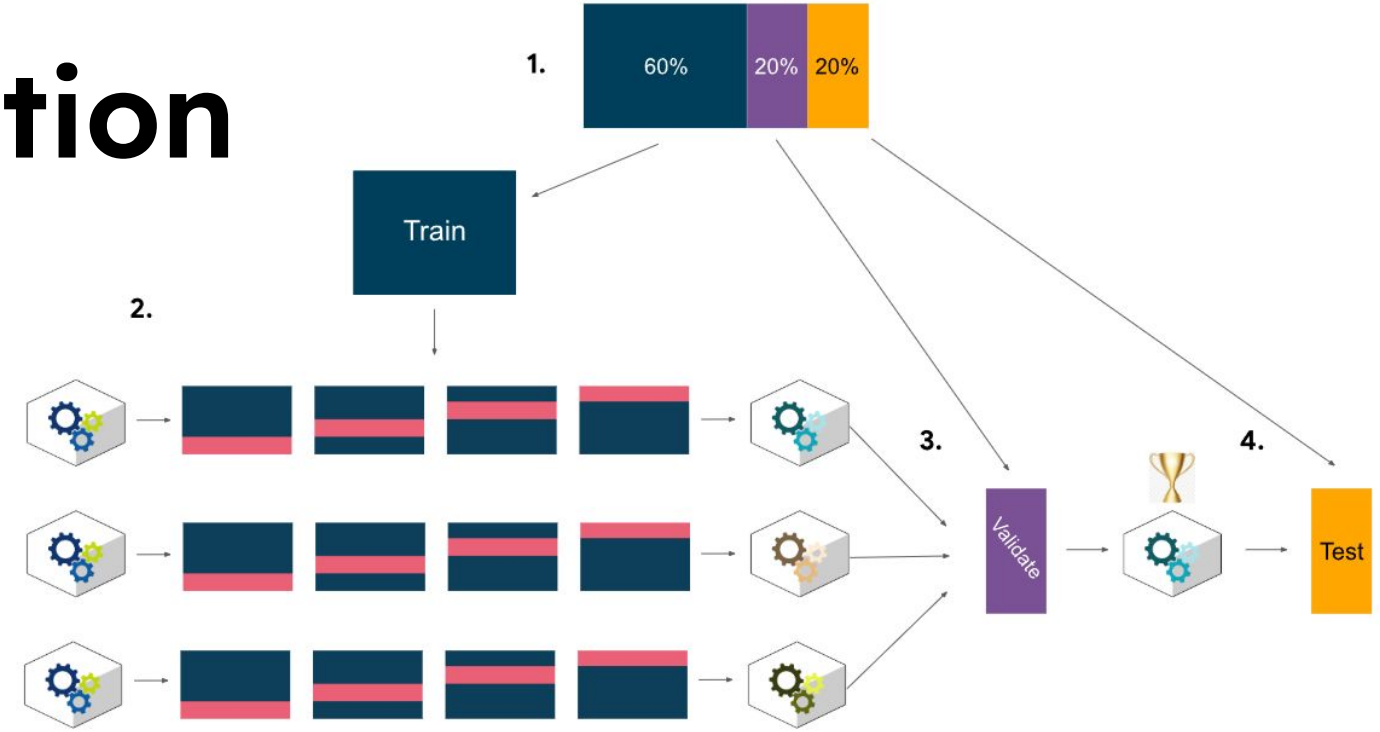
Cross Validation

To evaluate the models more carefully, **Random Forest and XGBoost were trained using cross validation.**

Cross-validation is a strategy used to evaluate the performance of predictive models. The central idea is to divide the data into different parts so that the model is trained on one part and tested on another. This process simulates the model's behavior when encountering new data.

Machine Learning models

Cross Validation



Machine Learning models

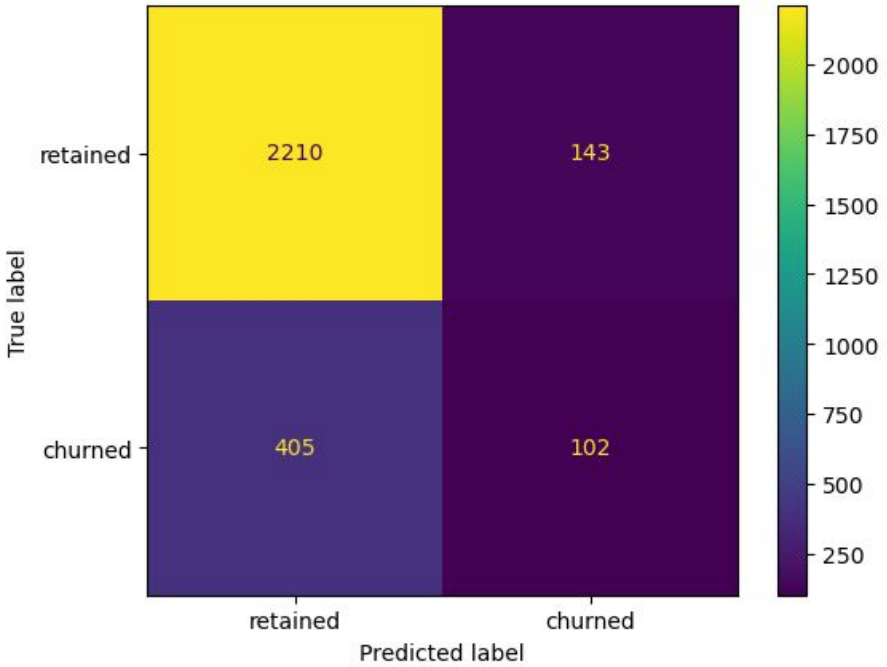
Logistic Regression | **Random Forest** | XGBoost

model	precision	recall	f1-score	accuracy
Logistic Regression	55%	9%	16%	82%
Random Forest	44%	12%	18%	81%

Even training the model with less data, we were able to **increase recall from 9% to 12%**, an increase of approximately 30%.
(tested on validation data)

Machine Learning models

Logistic Regression | Random Forest | **XGBoost**



True Retained (TN): 2,210
False Churns (FP): 143
False Retained (FN): 405
True Churns (TP): 102

Machine Learning models

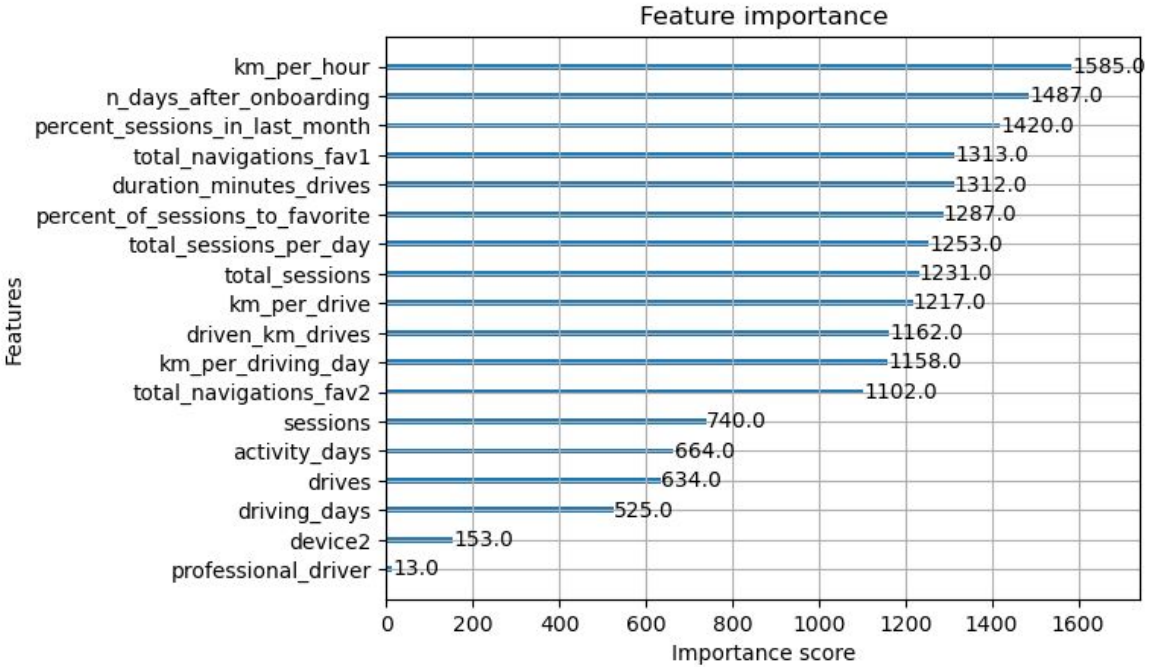
Logistic Regression | Random Forest | **XGBoost**

model	precision	recall	f1-score	accuracy
Logistic Regression	55%	9%	16%	82%
Random Forest	44%	12%	18%	81%
XGBoost	41%	20%	27%	80%

Even with precision dropping, all scores increased, with recall at 20%, a value more than double that of the first model.
(tested on test data)

Machine Learning models

Logistic Regression | Random Forest | **XGBoost**



The XGBoost model made more use of many of the features than did the logistic regression, which weighted a single feature (activity_days) very heavily in its final prediction.

This highlights the importance of feature engineering in creating the new columns. **Engineered features accounted for six of the top 10 features (and three of the top five).**

Machine Learning models

Logistic Regression | Random Forest | **XGBoost**

The ensembles of **tree-based models** in this project **are more valuable** than a singular logistic regression model because they achieve higher scores across all evaluation metrics and require less preprocessing of the data.

However, it is more difficult to understand how they make their predictions.

20% correct predictions seems a bit low.

20% correct predictions seems a bit low.

How can we improve it?

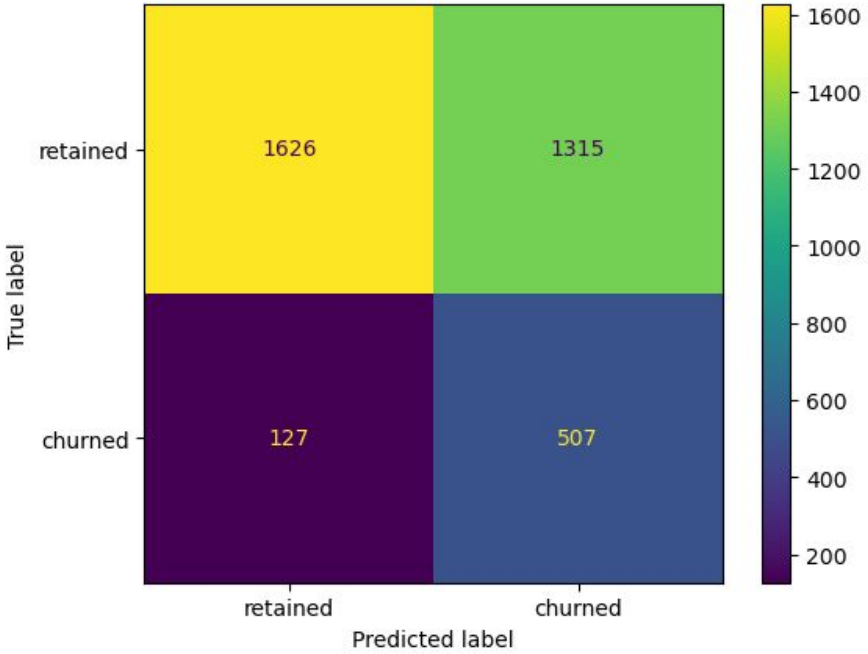
Balance data

The data already has a satisfactory proportion to apply to models, but since the objective is to predict churn members (a minority class), we could balance the data to test the effectiveness of the model.

SMOTE creates synthetic data from churned users, and then undersamples the retained users. **This balances data loss and duplication.**

Balance data

Logistic Regression | Random Forest | XGBoost



True Retained (TN): 1,315
False Churns (FP): 1,315
False Retained (FN): 127
True Churns (TP): 507

Balance data

Logistic Regression | Random Forest | XGBoost

model	precision	recall	f1-score	accuracy
Logistic Regression	55%	9%	16%	82%
Logistic Regression	28%	80%	41%	82%

The new model still remained biased, but now toward the "churned" side. Even with a large increase in false positives, the model reduced false negatives and managed to accurately predict the number of true churns.

Adjust threshold

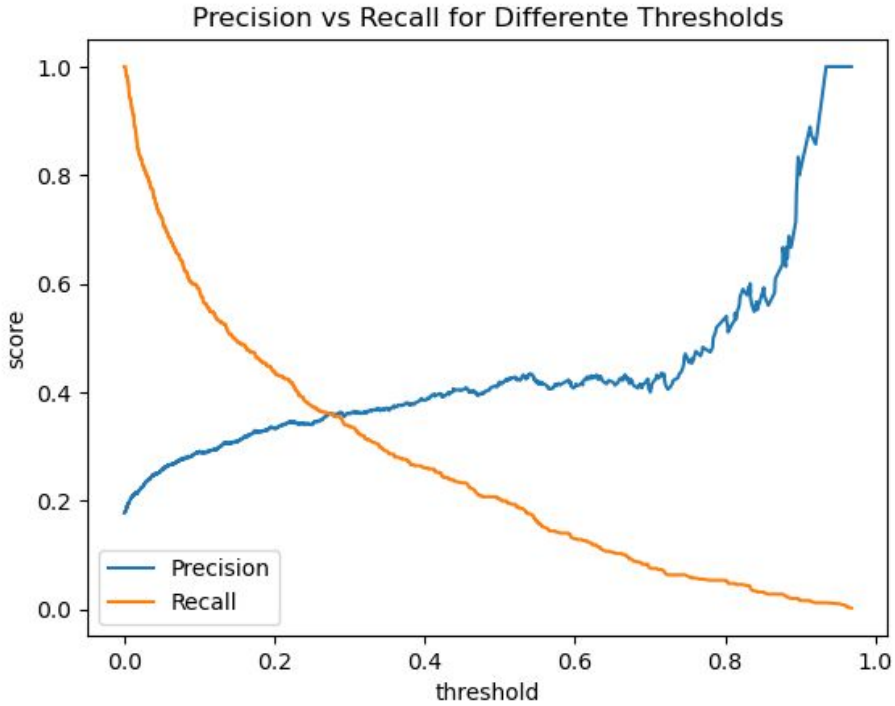
We can adjust the threshold to analyze changes in user churn scores and predictions.

This would increase the number of false positives but decrease the number of false negatives, managing to accurately predict the number of true churns.

For retention strategies that don't require a lot investment, such as an **in-app banner or an email,** it's more interesting to have a more targeted view of users who will churn, even if slightly distorted.

Adjust threshold

As recall increases, precision decreases.



Adjust threshold

Logistic Regression | Random Forest | **XGBoost**

model	precision	recall	f1-score	accuracy
XGBoost default threshold	41%	20%	27%	80%
XGBoost threshold = 0.145	30%	50%	38%	71%

With the threshold at 0.145 we were able to achieve a recall of 50%.

Final considerations

Whether this model is recommended for churn prediction would depend on the company's objective.

- If it were used to guide relevant business decisions, then no, as it is not a strong enough predictor, as evidenced by its low recall score.

- However, if the model is only being used to guide additional exploratory efforts or lightweight retention marketing campaigns such as banners and emails, it may have value.

Final considerations

Dividing the data into three parts (train, validation, and test) has its **advantages and disadvantages**. The biggest advantage is that it provides **another step in model verification**, providing a better estimate of future performance.

Tree-based models are often better predictors, as was the case in this case study.

Despite all the effort, the scores were lower than ideally desired. Even using techniques to increase recall, the other indicators dropped even further.

Next steps

Perform more tests tuning hyperparameters.

It would be extremely important for the development of an effective machine learning model to have more information about users.

- Personal information such as age, job, geographic and location information,
- Or more granular data such as reports of alerts on the route, reading alerts from other users, and how many different destinations they enter into the app.

Cross-referencing this data with current data in different combinations across more feature engineering stages could help the model become more predictive.

Tools and resources

Programming Language: **Python 3.13.5**

Libraries: pandas, numpy, scipy, matplotlib, seaborn, scikit-learn, imblearn, xgboost

Git repository: link

Complete project in jupyter notebook: link

Github: carvalhojm

LinkedIn: joaosuhett

Thank You

João Carvalho