Monday VIP Consulting

Yonatan Faigenbaum

Problem Description

Monday gets thousands of new customers each day but only has a few dozen consultants to work with them. Monday would like to assign the consultants to VIP clients in order to preserve them. The problem at stake is to design an algorithm that will detect the VIP clients based on the data Monday has on them. The model recommendations will help the consultants to better choose which client to help.

Data

Raw Data

The data we have contains 4 tables:

1. Accounts: contains general information about the account and the lead_score (the target)
2. Users: Each account has multiple users. This table contains information about the users
3. Subscriptions: General information about the subscriptions
4. Events: Log data about all the users activities during their time using Monday

Cleaning

Duplication
All duplicate rows from the accounts and users tables were removed based on the ids(each row should have a unique ids). From subscriptions and events we removed row that were completely identical

Nans
Columns that contain more than 50% Nans where removed for simplicity

Redundant Data
All rows from the users, subscriptions and events that contain account ids that are not in the accounts table were removed (no label)

After the cleaning the new datasets were saved under the intermediate folder.

Target

The data is highly unbalanced. Out of 716,828 accounts there are only 17,753 VIP accounts.

Features Assumptions

Normally the first step after the project kickoff will be to sit with the product manager and understand with them what are the most important features. Why did they label the VIP accounts as VIP, what makes them so special, are the big companies? Pay a lot of money? With this information the feature engineering is much easier.

The first direction is to go on simple and fast to develop models such as Logistic Regression and Random Forest/XGBoost

This means the data should be tabular. The first assumption I made was to aggregate all the users events/data based on the account id so each account has one sample with the aggregated features and the label.

All kinds of personal data such as emails, images, logos, devices, etc. were neglected.
The data set of the subscriptions was neglected completely because there weren't many rows left after the cleaning step

Numeric Features
Number of users per account
Number of active days per account
Number of all kind of event/Board/Columns
Some additional minor features

Categoric Features
Team size
The min/max team seemed important so I did some cleaning. All kinds of answers like. Solo Yo, Moi uniquement, etc. where replaced with team size 1. Accounts with no data were filed with -1 but other strategies like mean/mode/production should be considered.

Countries
All countries that belong to 10 or less accounts got the category: small_country

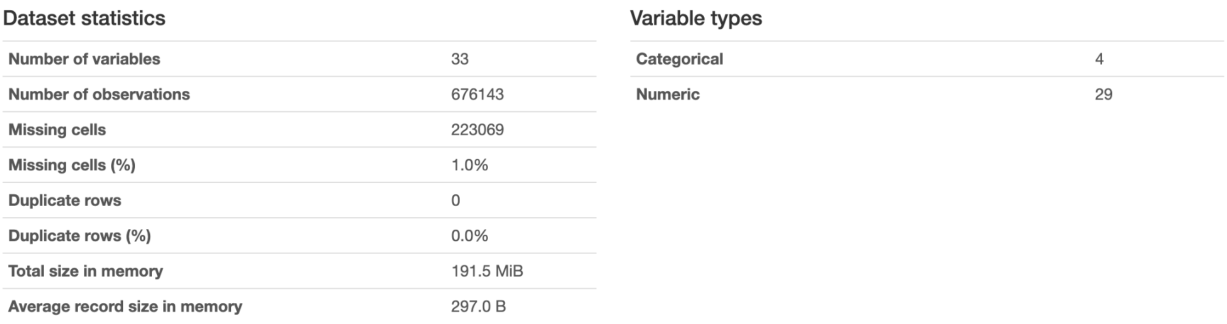
Industry
The industry of the company

Stickiness and Champion features
I hypothesized that some features indicating stickiness could indicate future VIP status. For example, uploading a logo, or adding a photo or phone number. In addition, I hypothesized that the first user from the account could be a potential “champion”, especially if they had the above mentioned stickiness features. However, all the first users uploaded phones,photos and logos so it is uninformative.

Exploratory Data Analysis

During the features engineering and data cleaning I did some visualization and explorations. A summary of the final features can be seen in *‘reports/features_profile.html’.*
The main figure is the correlation matrix. It can be seen that there are no features at all with strong correlation to the target. Some events naturally have some correlation.

Hopefully the model will learn the interactions between the features in order to make good predictions

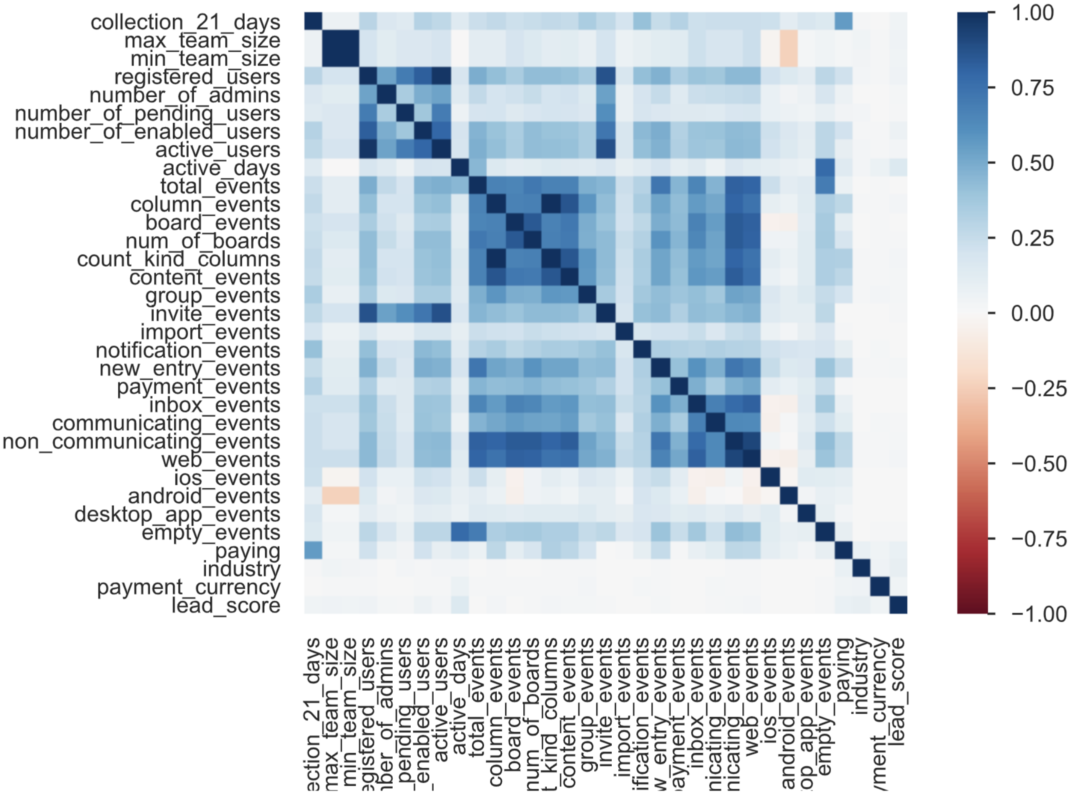


Dataset statistics

Number of variables	33
Number of observations	676143
Missing cells	223069
Missing cells (%)	1.0%
Duplicate rows	0
Duplicate rows (%)	0.0%
Total size in memory	191.5 MiB
Average record size in memory	297.0 B

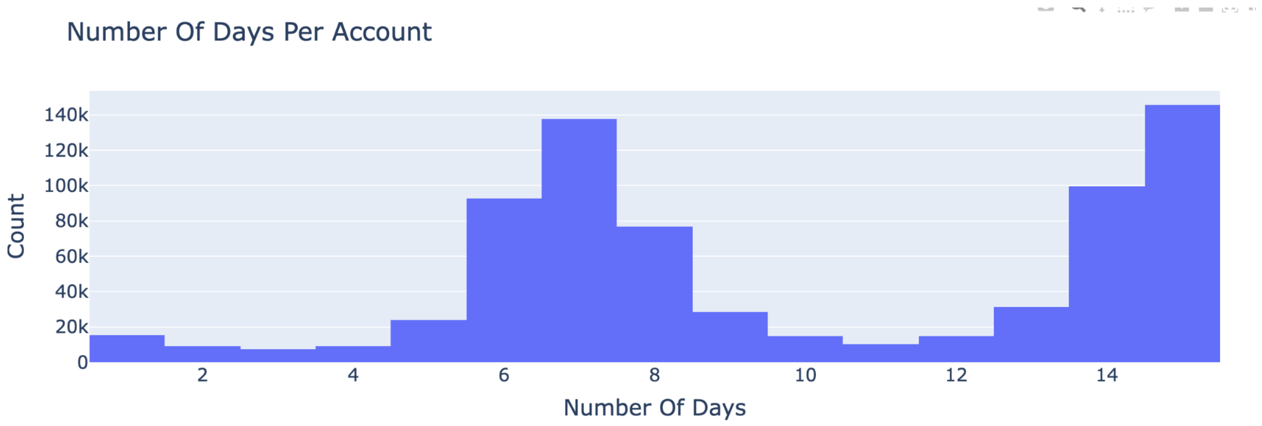
Variable types

Categorical	4
Numeric	29

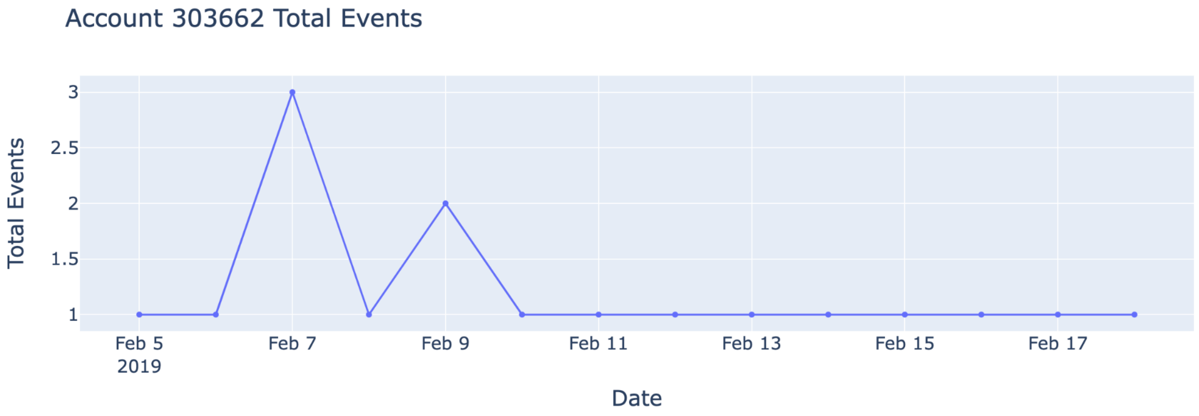


Time series

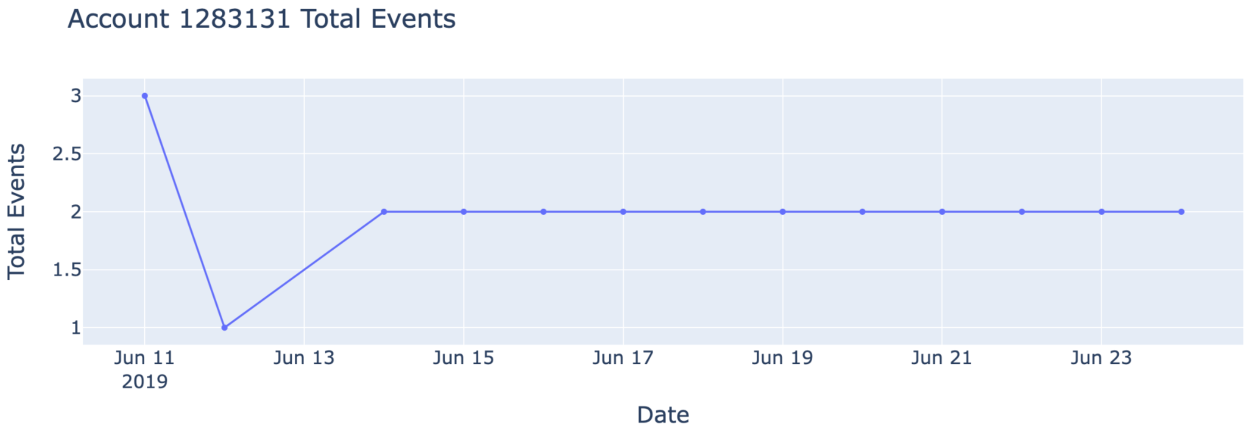
The events dataset contains time series events. From what I saw the events are up to 15 days max. I also looked at some of the time series to see if I can find behaviors that correlate to the VIP account, like positive trend of the number of events, etc. But from a short analysis I didn't see anything. This will be left for more advanced models.



Number Of Days With Events Distribution



VIP Account



Not VIP Account

Models & KPI

KPI

The main goal of the project is to allocate most of the consultants to VIP clients, since there is a limited number of them. This means that a natural KPI will be high precision. High precision means that most of the calls the consultants will do are with real VIP clients.
On the other hand we don't want to ignore too many VIP clients so the KPI for the model will be **F1 Score**.

As a baseline we want to beat a random model. A random model will have an average of 2.4% precision, 50% recall and 0.003 F1

Models

Logistic Regression

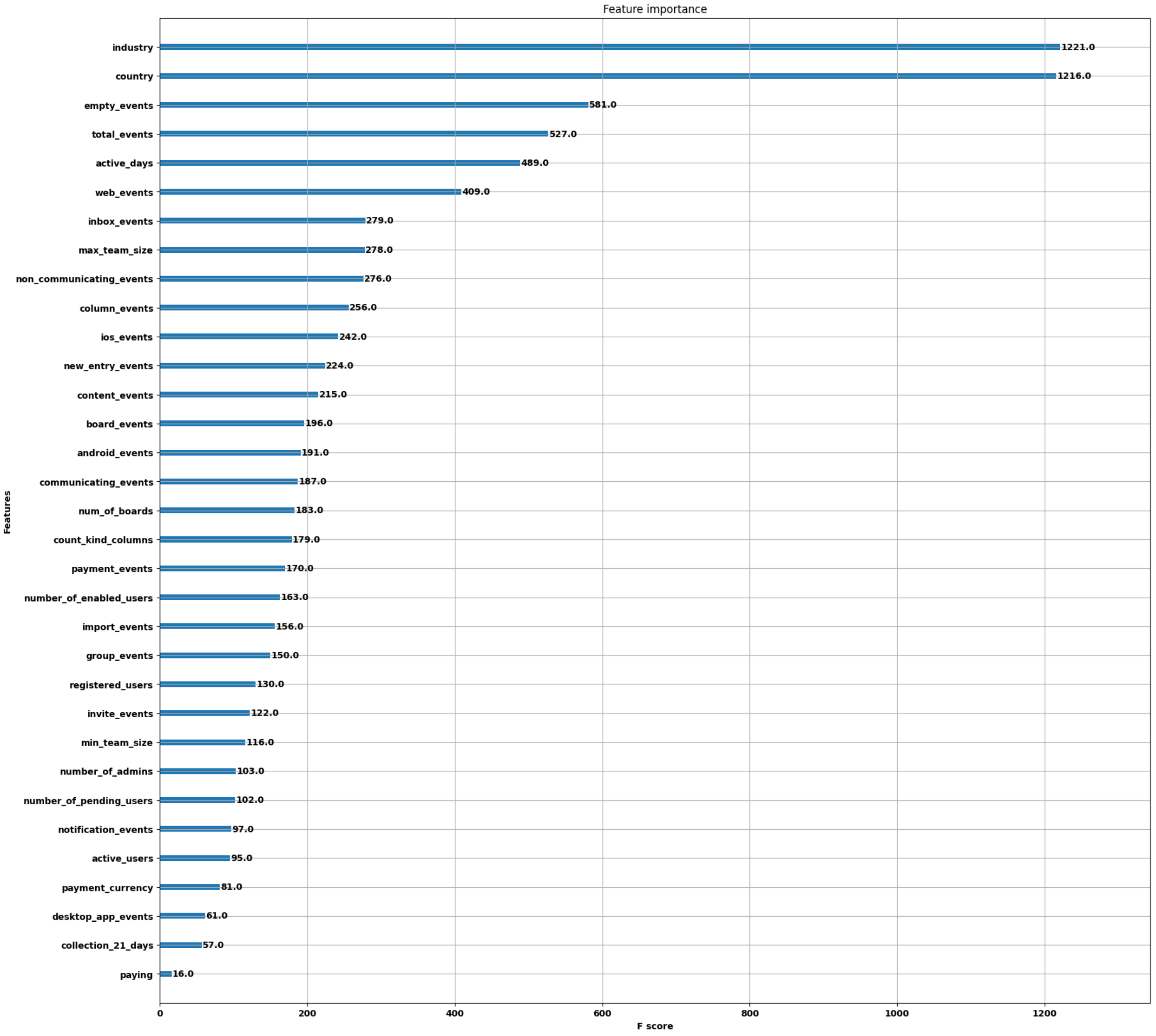
The first model will be a simple logistic regression. For this model we will use only the numeric/boolean features. It's a good starting point because it's easy and fast to implement. If it will get a good score we can learn from it which features are dominant and which are not.

XGBoost

A second good option is a tree based model like XGBoost. Such models are proven to be best for classification of tabular data.

The first results of the XGBoost were not very good. Inorder to improve them a bit I did small a hyperparameter tuning. The parameters I got were the limits of the search space so I heater had a bug or that there is room for increasing the space limit in the direction of the current results.

From the feature importance figure it can be seen that the two dominant features are categorical. It may indicate that more categorical feature should be added sense the activity is not so significant (In the future maybe product will want the model predict VIP client before logging it to the system)


Feature importance
industry 1221.0
country 1216.0
empty_events 581.0
total_events 527.0
active_days 489.0
web_events 409.0
inbox_events 279.0
max_team_size 278.0
non_communicating_events 276.0
column_events 256.0
ios_events 242.0
new_entry_events 224.0
content_events 215.0
board_events 196.0
android_events 191.0
communicating_events 187.0
num_of_boards 183.0
count_kind_columns 179.0
payment_events 170.0
number_of_enabled_users 163.0
import_events 156.0
group_events 150.0
registered_users 130.0
invite_events 122.0
min_team_size 116.0
number_of_admins 103.0
number_of_pending_users 102.0
notification_events 97.0
active_users 95.0
payment_currency 81.0
desktop_app_events 61.0
collection_21_days 57.0
paying 16.0
Features
0
200
400
600
800
1000
1200
F score

Results

Dataset	Score	Logistic Regression	XGBoost
Train	F1	0.122	0.36
	Pression	0.07	0.8
	Recall	0.6	0.23
Test	F1	0.12	0.22
	Pression	0.07	0.56
	Recall	0.6	0.13

In terms of F1 and pression the XGBoost wins. The recall of the logistic regression is higher but we less concern about it

The winning model is **XGBoost**

The meaning of 0.56 pression in the test dataset means that around 56% of the consultants are with VIP clients.

The results is not great but it's a great improvement with little cost comparing to taking random calls

Future Steps

Increase the search space for the hyper parameter algo
Go back to feature engineering
Find more categorical features
Think how to use more users data and not only aggregate them
Maybe try time series features

Code Arrangement

Code arrangement can be found in the READ.md file