

Churn analysis

anantara vacation club

Presented by: mojtaba peyrovi(moji)

Date: 8 may 2018

## cleaning & making the data file ready for analysis

The following procedure has been done on the data file to prepare it for churn analysis. Here is the list of all adjustments separated by sheet name:

**DEMO**

1. “Upgrade Flag” column has been removed because of missing so much data and definition unclarity.
2. “Date of Birth” has been replaced with “Age” in order to be analyzed by numerical techniques.
3. “Anniversary Date” has been replaced with “days\_ago\_joined” to be capable of being analyzed by numerical techniques.
4. On “Tier” column we have “Voyager” and “Club Discovery” values which belong to new comers and also these records are only 332 rows, so they have been removed.
5. Missing “Age” values have been removed with the average age (44). If there were too many empty rows and the age variance was too much we could fill the missing values with ages by specific groups for instance by Tier.
6. “accum\_points” added that shows how many accumulative points each customer has spent so far.
7. “points\_per\_day” indicates daily points usage for each customer. Although it may sound correlated with “accum\_points”, but since the value is from “points\_per\_day” divided by “days\_ago\_joined” and “days\_ago\_joined” isn’t constant but identical per customer, so I added it as another churn predictor.
8. More than 90% of customers are coming from 12 countries. So, I removed the other countries for this project just because it’s for testing purpose not the real-life project.
9. “days\_ago\_last\_arrived” comes from a pivot table called “arrivals” which gathers the arrival date info from “Reservation” table. It shows how many days ago the customer last visited a property or in better words “how many days ago a customer has used one of Anantara Club’s services.” Negative values in this column show pre-bookings that aren’t due yet.
10. “arrival\_count” comes from the “arrivals” pivot from “Reservation” which indicates how many times each customer has used Anantara Club’s services.
11. Since our target field is “Status” and it has two values: “Closed” and “CxlBadDebt” we replace them with 1 for “Closed” and 0 for “CxlBadDebt”.

**RESERVATION**

1. “Arrival” replaced with “days\_ago\_last\_arrived” in order to be analyzed by numerical techniques.
2. Departure” replaced with “days\_stayed” (days stayed - departure-arrival)
3. “days\_ago\_last\_arrived” left joined to “Demo” table.

**PAYMENT**

One of the most important predictors for churn rate is how much the customer owes and how long ago their last payment was. Also “Sale Type” can affect the customer status.

The following changes were implemented on “payment” sheet.

1. “% installment” datatype has changed from percentage to float in order to be capable of being analyzed by numerical techniques.
2. “Last Payment Date” has been replaced with “days\_ago\_last\_paid”.
3. “%Installment” has been replaced with “owes” which is the indicator of what percent of their payment still remains. Here is the logic behind it:

IF “Sale Type” = “Cash” THEN “owes”= 0

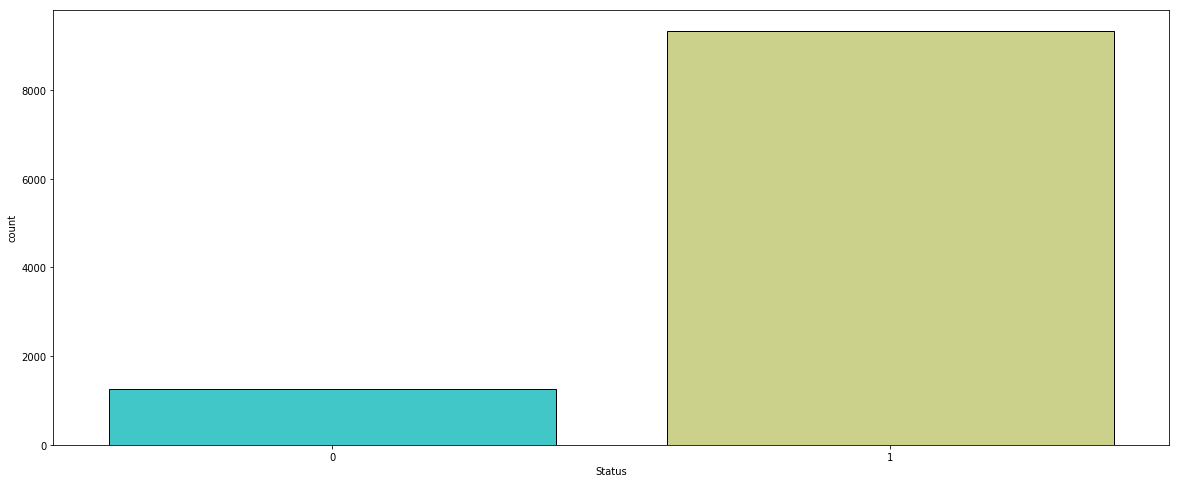
IF “Sale Type” = “Installment Plan” THEN “owes” = 1 – “%Installment”

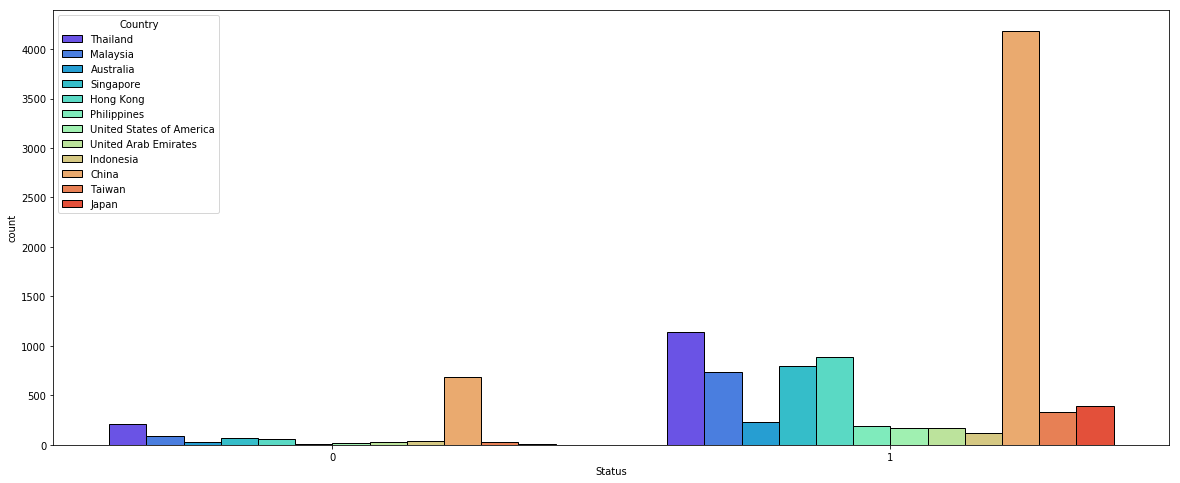
IF “Sale Type” = “Null” THEN remove the row.

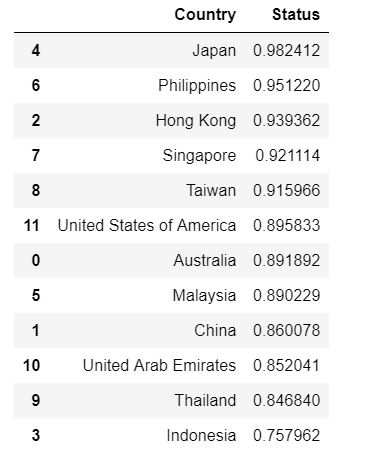
Since the remaining payments are only around 2,000 out of 10,000 customers listed in “Demo” we can’t use predictors such as “days\_ago\_last\_paid” or “owes” to predict the “Status” unless we make two different analyses, first on a dataset of customers with pay info, and another dataset with no pay info, but for this test I just used the whole dataset of customers without considering their payment status.

## SOME INITIAL INSIGHTS

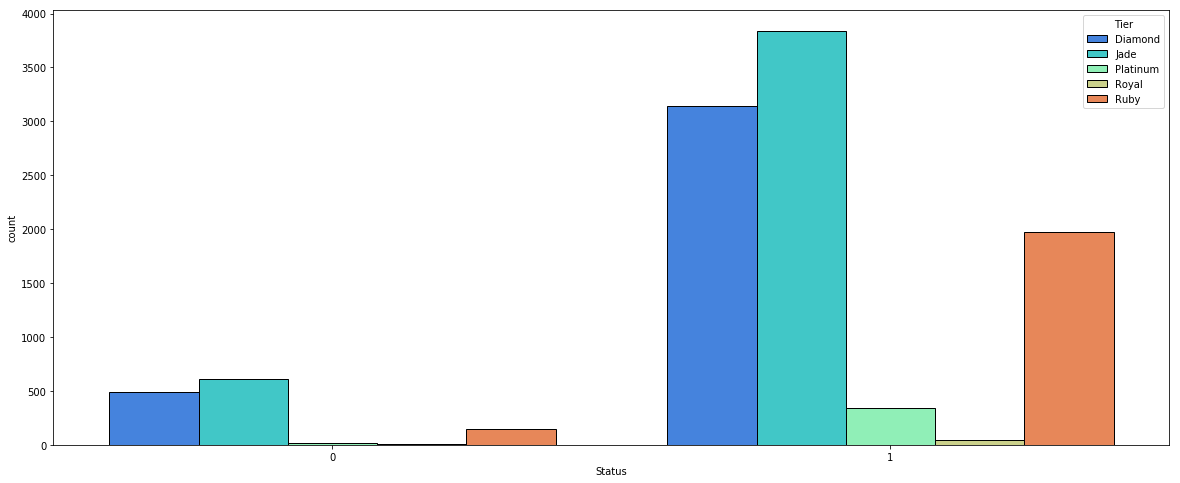
1. Count of “closed” vs “CxlBadDebt.



1. **Churn per Country**: Count of “Closed” and “CxlBadDebt” per country. As the plot and the table show, countries like Japan, Singapore, Hong Kong, and the Philippines are the most loyal countries.



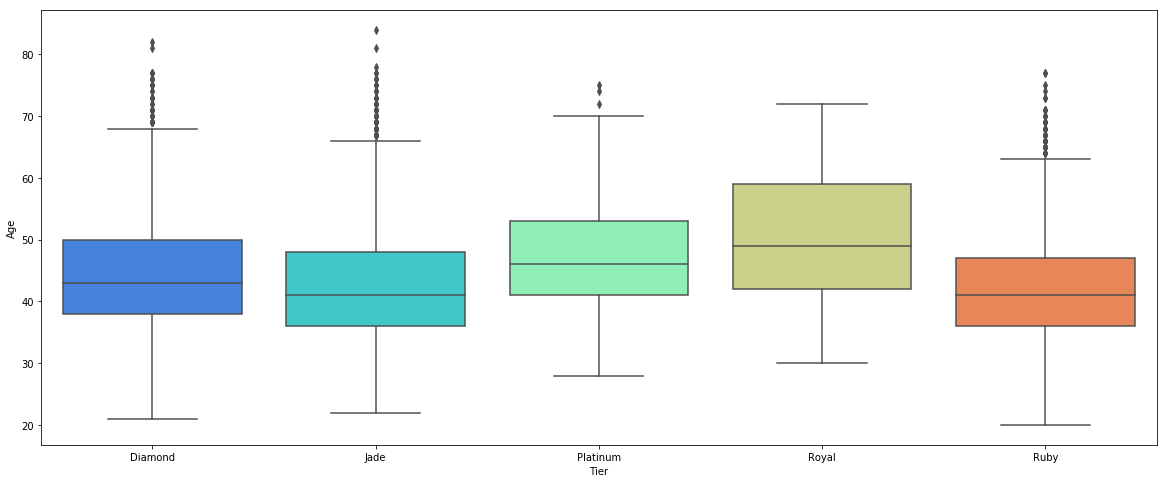
*Countries sorted by the percentage of their customers being “Closed”*

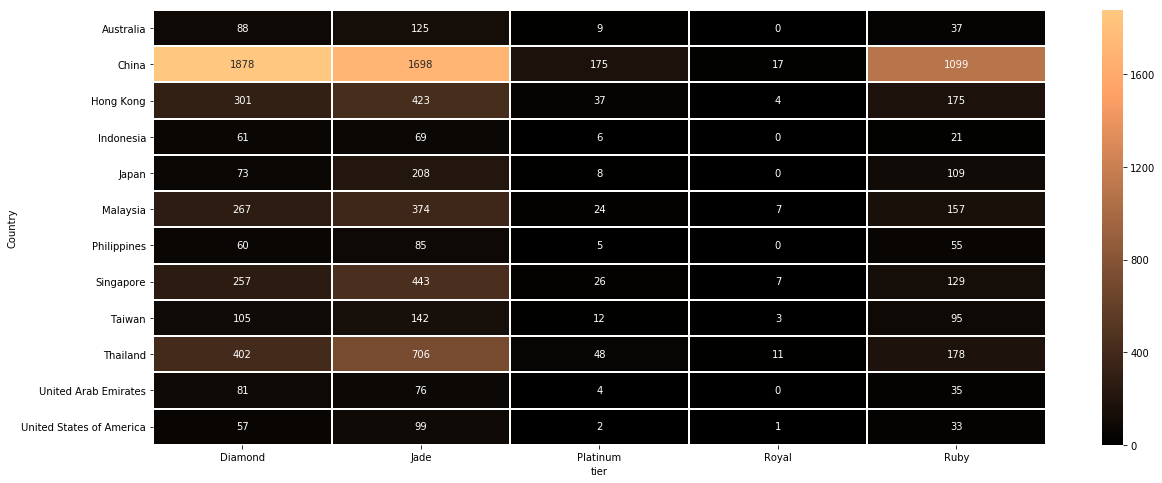
1. **Churn per Tier:** Count of “Closed” and “CxlBadDebt” per “Tier”



*Tiers sorted by the percentage of their customers being “Closed”*

1. **Tier vs Age:** As we see, the age range of Royal customers are higher than other tiers. It means that older people tend to get more senior services such as “Royal.”



5– Count of customers in each country per “Tier.”

There is more data visualization done with Excel, which will be emailed to the examiner.

## making the model based on classification methods

Since the churn analysis has a non-continuous nature, we must use classification techniques to predict the churn probabilities.

In this experiment we want to predict how many people will cut their ties from the company or simply convert from “Closed” to “CxlBadDeb.” So, our target field would be “Status.” In our code we call it “y.”

The rest of the columns, except for a few removed columns, will become our “predictors” or “X.”

In order to do this analysis, we need to have all categorical values converted to numerical values; even the Status or “Country.” I use a technique called dummy variables to convert each category to a column with 0-1 values for yes/no identifiers.

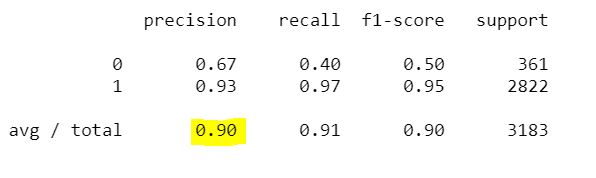
As mentioned before, the Status field values changed to 0-1 as well. 1 for “Closed” and 0 for “CxlBadDebt.”

The cleaned table will be sent to examiner by email.

LOGISTIC REGRESSION MODEL

This method is popular for classification cases with binary returns. I used Python Scikit-learn library to do it.

After splitting data to train and test values, and fitting them to the model, we evaluated to see how accurate the model was. By getting the confusion matrix we could see how many of the values were correctly predicted.



As we see, this model has 90% accuracy. Below is the confusion matrix:

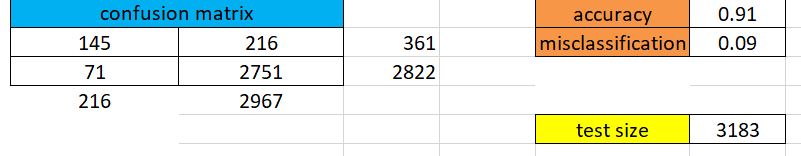


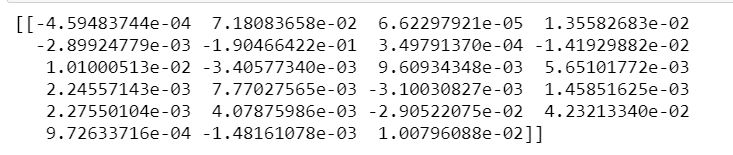
**It means:**

Out of 3,183 records used by the model, 2,967 of them were “Closed” or 1 and our model has projected 2,751 of them correctly and only 216 of them were incorrect.

Also, out of 216 “CxlBadDebt” the model predicted 145 correctly and only 71 incorrectly.

So here is the brief:



Now using this model, we can calculate coefficients of each predictor.

Also, we need to know the intercept, which is calculated by Python as:

n = [0.02283976]

In order to have the probability for each record we can use the following formulas:

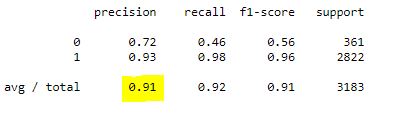
1. Z = n + coeff1X1 + coeff2X2+ …..coeffiXi

Then:

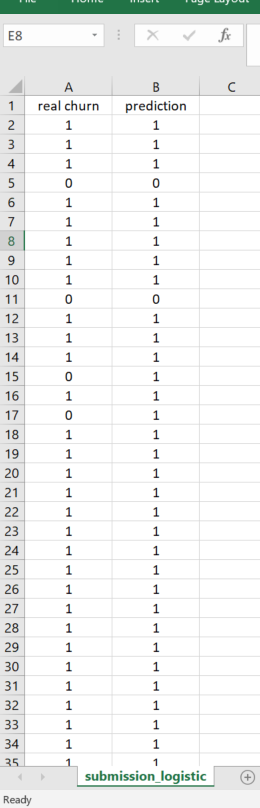
1. Y = exp(Z)/(1+exp(Z))

The outcome would be a percentage. Normally in logistic regression we can say any probability more than 50% is categorized as 1 and less than 50% would be 0, or in our example, if the calculated probability is more than 50% the customer is “Closed” otherwise the risk of being “CxlBadDebt” would be high.

NOTE: Because an important part of the machine learning process is to continuously improve the model, I tried to standardize the dataframe because the column “days\_ago\_last\_arrived” has negative values. After doing it and running the algorithm again, it increased the precision from 90% to 91%.



Again, we can calculate coefficients, the intercept, and the predictions. Since the process is the same except for a few numerical changes, I didn’t do the calculations here again, but we can see the results as Python has done it.

(the full results will be sent to the examiner by email)

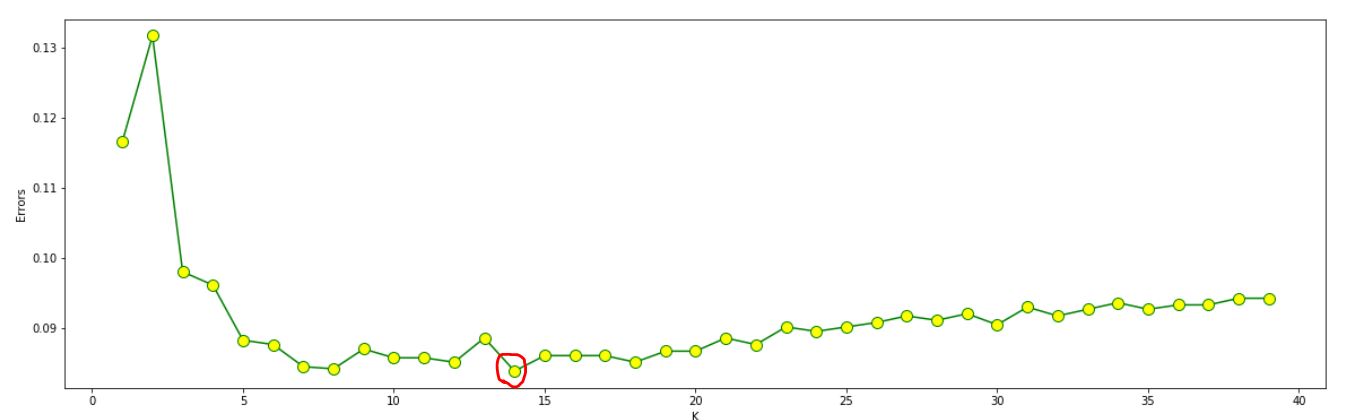
*Column “real churn” is the reality and “prediction” is how the model has predicted it 🡺*

K-NEAREST NEIGHBORS MODEL

I tried to use this algorithm for the same dataframe to see if we can make any improvement. For this method we will have to use the same cleaned standardized dataset and the script is slightly different from Logistic Regression.

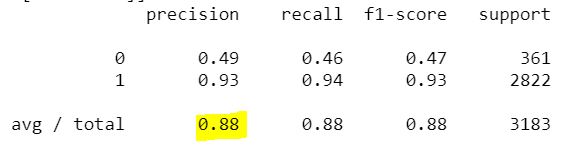
Since we need to have K neighbors, we need to find the best K that minimizes the errors. In order to do this, we start from a specific K for example 1, and then we loop through different K values to see which one will have less error (real values – predictions).

Here is the outcome:

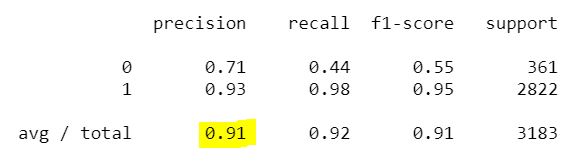


As we can see, the minimum error happens at K=14.

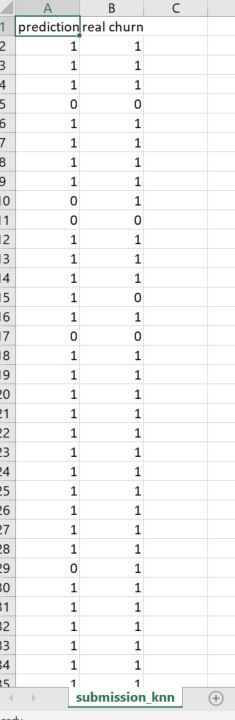
K=14 can improve the model’s precision from 0.88 for K=1 to 91% for K=14



*Precision when K=1*



*Precision when K=14*

**

Here is the screenshot of the final results calculated with KNN algorithm.

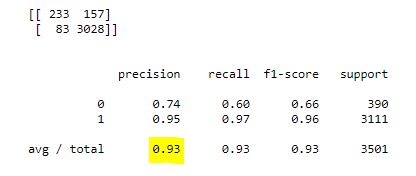
The full results will be submitted by email.

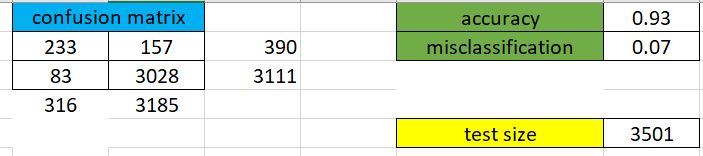
RANDOM FOREST MODEL

Random Forest model is a great tool and normally works very well with large datasets.

I used Random forest algorithm and used the cleaned dataset to evaluate.

Here is the result:



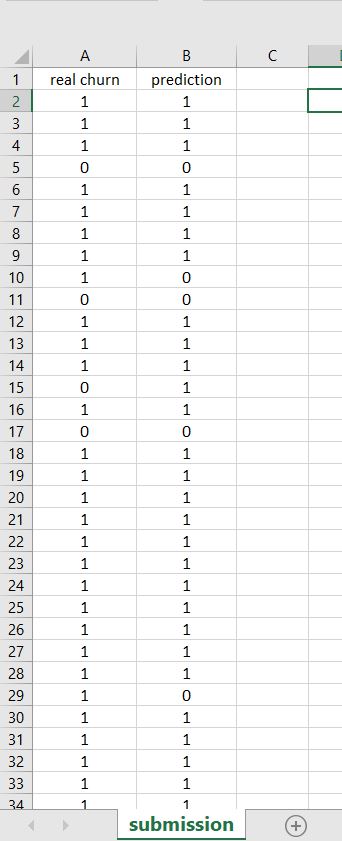


As we can see this model is able to:

* Predict 3,028 “Closed” cases out of 3,111 cases correctly. (95%)
* Predict 233 “CxlBadDebt” cases out of 390 cases correctly. (74%)

The overall accuracy is 93% and the total error is 7%.

Here is a screenshot of the prediction vs reality file,

which will be sent to the examiner as well.

*Column “real churn” is the reality and “prediction” is how the model has predicted it 🡺*

There are other methods to do this project such as Neural Networks, Decision trees, etc., but for now I think 93% of accuracy with Random Forest is a fair judgement.