

DECISION TREE ANALYSIS & LOGISTIC REGRESSION

-Roshan Tiwari

PROBLEM STATEMENT

Qualtrics is a platform for companies to create surveys and monitor the responses made to those surveys. Qualtrics offers deep insight to companies by summarizing the data and applying concepts like text mining on the responses received. Companies can utilize these features to improve their employee experience, customer experience and brand experience. Qualtrics understands the importance of data and has collected all the information on their clients including various metrics that quantify the usage of platform, metrics that measure client's response to Qualtrics marketing campaigns and most importantly the sales data. The client names in these datasets are replaced by anonymous AccountID. The problem at hand is to utilize these data sets to derive insights that can help Qualtrics to boost their customer retention, customer acquisition and profitability. I have trained decision tree and logistic regression models to derive these insights.

EXECUTIVE SUMMARY

Qualtrics has categorized their clients into five main revenue brackets based on the revenue they generate. An initial time series analysis of Qualtrics revenues shows that 70% of its revenue is generated by the top bracket companies. I tried to implement models that will predict whether a company belongs to this bracket or not by using their responses to Qualtrics marketing campaign and their behavior on Qualtrics platform. Initial exploratory analysis shows that high revenue generated for Qualtrics is no way co-related to the high number of projects created by companies on the platform. Hence, this task was much more complicated for the models and after multiple iterations two models presented promising results.

TIME SERIES ANALYSIS OF REVENUE

As per the Revenue Bracket categories provided by Qualtrics, the data clearly shows that 70% of total Quarterly Reported Annual Recurring Revenue belongs to companies in top most revenue bracket. As per this insight, we have termed the companies in the top bracket as Platinum Clients, they make up for 15% of all clients.

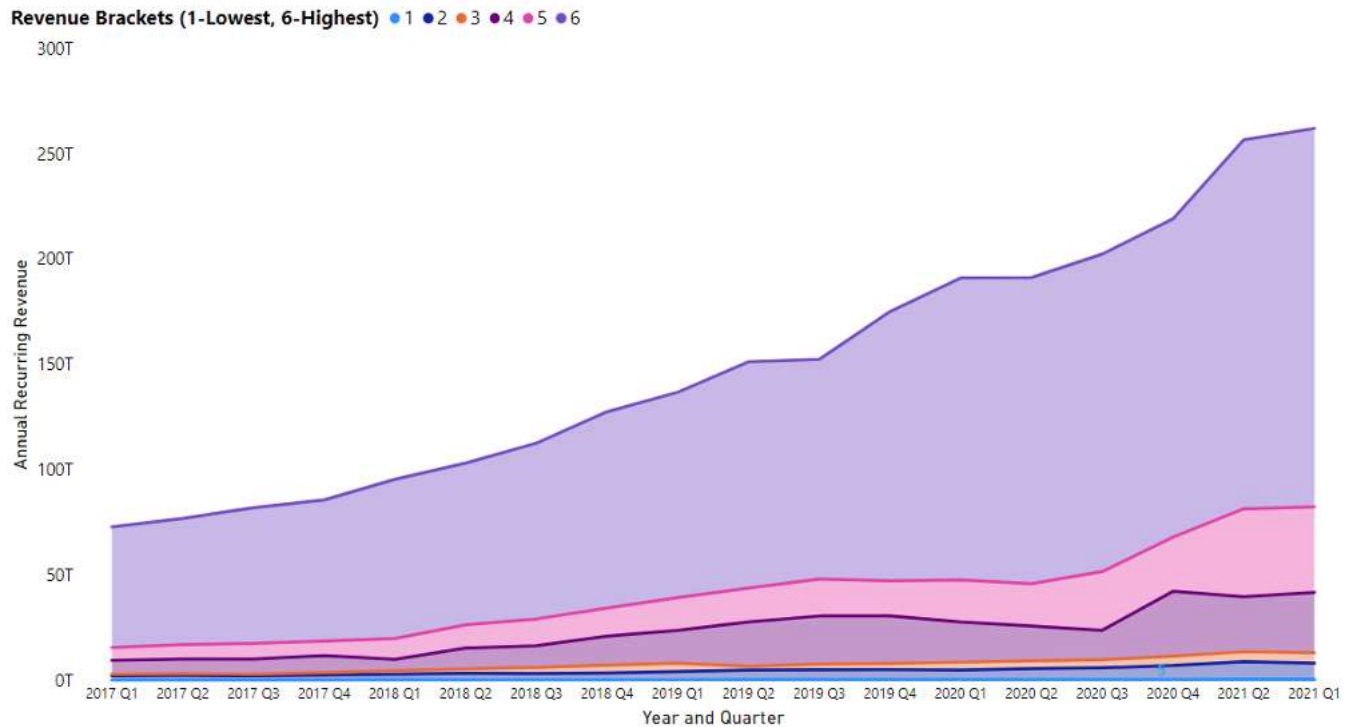


Figure 1: Quarterly Reported Annual Recurring Revenue Segmented by Revenue Brackets

DECISION TREE ANALYSIS

This decision tree model utilizes marketing data to predict whether a client belongs to the top most revenue bracket or not.

Data Description – Marketing data consists of metrics that record client’s response to Qualtrics marketing campaigns. The below table provides a detailed description of all the attributes and metrics in this dataset.

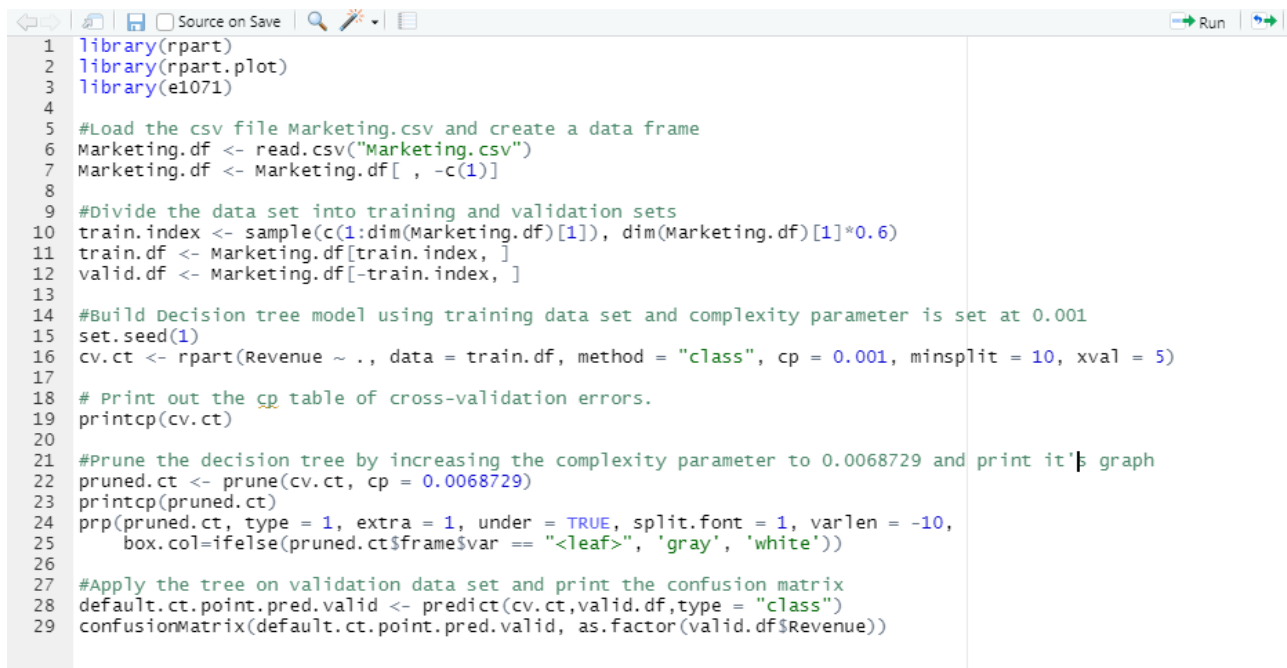
MARKETING_Account_PersonLevelBehavioralAgg.csv

This table contains information at a per person level based on marketing interactions. These leads can be tied to the AccountID shown to join the person to their associated account.

LeadID	ID for activity shown
CountWPVisits	Number of webpages visited in the last 90 days
CountFormFills	Number of form fills in the last 90 days
CountLinkClicks	Number of link clicks in the last 90 days
CountEmailOpens	Number of emails opened in the last 90 days
AccountID	Id for account lead is associated with
MLAccountID	Same as above
CountWPVisits_A	Number of webpages visited in the last 90 days by all contacts from the associated account
CountFormFills_A	Number of form fills in the last 90 days by all contacts from the associated account
CountLinkClicks_A	Number of link clicks in the last 90 days by all contacts from the associated account
CountEmailOpens_A	Number of emails opened in the last 90 days by all contacts from the associated account
CountPeopleWPVisits_A	Number of unique people visiting webpages in the last 90 days
CountPeopleFormFills_A	Number of unique people filling out forms in the last 90 days

R CODE – Executed in R studio

The following code divides the data set in training and validation dataset in 60:40 ratios. The minimum number of observations for a single node (minsplit) is set at 10. The tree is further pruned and summary is printed. Then it prints out a map for the decision tree and displays the confusion matrix after implementing the tree on validation dataset.



```
1 library(rpart)
2 library(rpart.plot)
3 library(e1071)
4
5 #Load the csv file Marketing.csv and create a data frame
6 Marketing.df <- read.csv("Marketing.csv")
7 Marketing.df <- Marketing.df[, -c(1)]
8
9 #Divide the data set into training and validation sets
10 train.index <- sample(c(1:dim(Marketing.df)[1]), dim(Marketing.df)[1]*0.6)
11 train.df <- Marketing.df[train.index, ]
12 valid.df <- Marketing.df[-train.index, ]
13
14 #Build Decision tree model using training data set and complexity parameter is set at 0.001
15 set.seed(1)
16 cv.ct <- rpart(Revenue ~ ., data = train.df, method = "class", cp = 0.001, minsplit = 10, xval = 5)
17
18 # Print out the cp table of cross-validation errors.
19 printcp(cv.ct)
20
21 #Prune the decision tree by increasing the complexity parameter to 0.0068729 and print its graph
22 pruned.ct <- prune(cv.ct, cp = 0.0068729)
23 printcp(pruned.ct)
24 prp(pruned.ct, type = 1, extra = 1, under = TRUE, split.font = 1, varlen = -10,
25     box.col=ifelse(pruned.ct$frame$var == "<leaf>", 'gray', 'white'))
26
27 #Apply the tree on validation data set and print the confusion matrix
28 default.ct.point.pred.valid <- predict(cv.ct,valid.df,type = "class")
29 confusionMatrix(default.ct.point.pred.valid, as.factor(valid.df$Revenue))
```

Figure 2 Decision Tree Code

OUTPUT- Tree Map & Summary

The following graph shows the mechanism of decision tree and the summary shows there are 5 attributes that are used to train the tree – Account Score, Number of marketing emails opened, Number of people who open marketing emails, Number of Webpage visits and Number of people visiting the webpage.

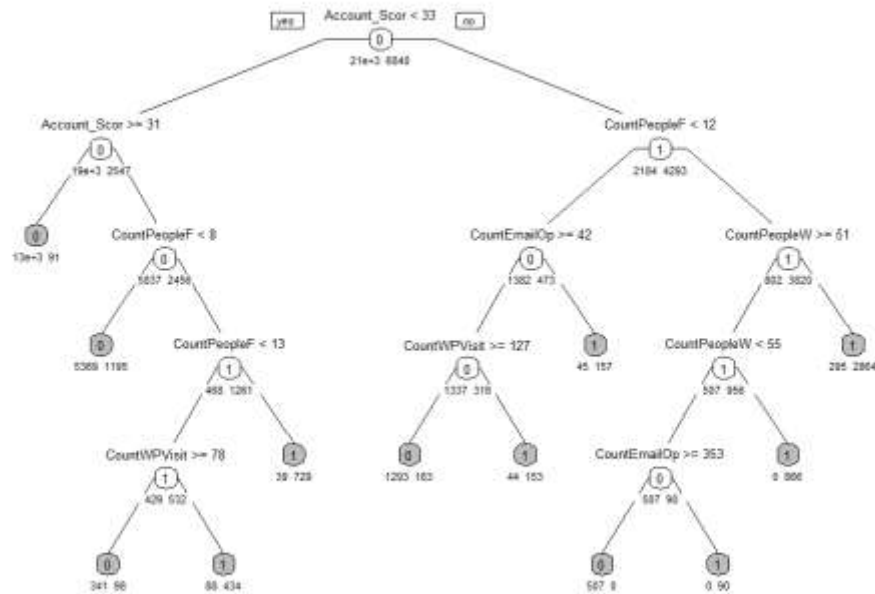


Figure 3 Decision Tree Plot

```

Classification tree:
rpart(formula = Revenue ~ ., data = train.df, method = "class",
      cp = 0.001, minsplit = 10, xval = 5)

Variables actually used in tree construction:
[1] Account_Score      CountEmailopens_A    CountPeopleFormFills_A CountPeopleWPvisits_A CountWPvisits_A

Root node error: 6840/28153 = 0.24296

n= 28153

   CP nsplit rel error  xerror   xstd
1 0.3083333    0  1.00000 1.00000 0.0105204
2 0.1328947    1  0.69167 0.69167 0.0091721
3 0.0579678    2  0.55877 0.55877 0.0084025
4 0.0304825    4  0.44284 0.44722 0.0076341
5 0.0177632    6  0.38187 0.39985 0.0072649
6 0.0163743    8  0.34635 0.36257 0.0069526
7 0.0159357    9  0.32997 0.35439 0.0068811
8 0.0131579   10  0.31404 0.32120 0.0065798
9 0.0068729   11  0.30088 0.29547 0.0063321
  
```

Figure 4 Decision Tree Summary

OUTPUT- Confusion Matrix

The confusion matrix for the decision tree analysis presented a great accuracy of 98.5. The model correctly classifies 99.8 % of non-platinum clients and 94.7 of platinum clients.

```
Confusion Matrix and Statistics

      Reference
Prediction  0      1
0 14184  243
1    33 4310

      Accuracy : 0.9853
      95% CI : (0.9835, 0.987)
No Information Rate : 0.7574
P-Value [Acc > NIR] : < 2.2e-16

      Kappa : 0.9593

McNemar's Test P-Value : < 2.2e-16

      Sensitivity : 0.9977
      Specificity : 0.9466
      Pos Pred Value : 0.9832
      Neg Pred Value : 0.9924
      Prevalence : 0.7574
      Detection Rate : 0.7557
      Detection Prevalence : 0.7686
      Balanced Accuracy : 0.9722

      'Positive' class : 0

> |
```

Figure 5 Confusion Matrix Decision Tree

Logistic Regression

Data Description: Platform usage dataset provides various metrics that record the behavior of a client on the Qualtrics platform. The below table provides a detailed description of all the attributes in the dataset.

USAGE_PlatformUsage.csv

UltimateParentAccountID	A company who pays for Qualtrics. aka "Account".
Account ID	Salesforce AccountID associated with the Brand and ultimate parent
BrandIDHash	An ID representing a unique instance of Qualtrics, aka a "brand". More Accounts will have 1 QualtricsID but it is possible for 1 account to have multiple brands and for brands to have multiple associated accounts.
Date	Month when the Activity took place
SessionCount_Amp	Number of sessions across all users in the brand. A session is defined as usage separate by 30+ minutes of inactivity.
ResponseCount	Number of responses received
ActionCount_ProjectsCreated	Number of projects created.
MUserCount_ProjectsCreated	Number of users creating projects.
Pageland_Count	Number of pagelands across all users in the brand. A pageland event occurs whenever a unique url is loaded.
AMRActions_UniqueUsers	Number of unique "AMR users" active in the brand. AMR users are users who use more advanced features in the product, such as TextIQ or block randomization.
TextIqConsumption_Events	Number of events related to TextIQ. Events can be everything from pagelands to clicks on certain buttons.
TextIqConsumption_Uniques	Number of unique users performing TextIQ events.

R CODE – Executed in R studio

The following code divides the dataset into training and validation sets. Then it trains a logistic regression model and prints a confusion matrix after validating the model. It first builds a full logistic model and then trains a stepwise model by removing one attribute that did not add any significant value in every step.

```
1 library(caret)
2 library(e1071)
3
4 #Load the csv file Spend21_usage.csv and create a data frame
5 Platform_Usage.df <- read.csv("spend21_usage.csv")
6 Platform_Usage.df <- Platform_Usage.df[, -c(1)]
7
8 #Divide the data set into training and validation sets
9 train.index <- sample(c(1:dim(Platform_Usage.df)[1]), dim(Platform_Usage.df)[1]*0.6)
10 train.df <- Platform_Usage.df[train.index, ]
11 valid.df <- Platform_Usage.df[-train.index, ]
12
13 #Train a full logistic regression model
14 full.logit.reg <- glm(ARR ~ ., data = train.df, family = "binomial")
15
16 #Train a stepwise model by removing the attributes from the full model one by one
17 backwards = stepAIC(full.logit.reg)
18 summary(backwards)
19
20 #Predict the probability of an account being in the top revenue bracket in validation data set
21 logit.reg.pred <- predict(backwards, valid.df[, -1], type = "response")
22
23 #Confusion Matrix with probability >0.2 considered as a top revenue bracket account
24 confusionMatrix(as.factor(ifelse(logit.reg.pred>0.2, 1, 0)), as.factor(valid.df$ARR))
```

OUTPUT - SUMMARY

```
call:
glm(formula = ARR ~ ., family = "binomial", data = train.df)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-7.5239  -0.2989  -0.2691  -0.2635   2.5128

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   -3.34343486761  0.16523142601 -20.235 < 0.0000000000000002 ***
UserCount_All    0.00002951879  0.00001620918   1.821   0.06859 .
SessionCount_Amp  0.00000364676  0.00000231364   1.576   0.11498 .
ResponseCount    0.00000006089  0.00000002293   2.656   0.00791 **
ActionCount_ProjectsCreated -0.00047105582  0.00027112076  -1.737   0.08231 .
MuserCount_ProjectsCreated  0.00078219848  0.00045476163   1.720   0.08543 .
Pageland_Count    0.00000135143  0.00000067068   2.015   0.04390 *
AMRActions_UniqueUsers -0.00066494281  0.00022768506  -2.920   0.00350 **
TextIqConsumption_Events  0.03748528968  0.00637071174   5.884  0.000000004 ***
TextIqConsumption_Uniques -0.00000676976  0.00002939403  -0.230   0.81785

---
Signif. codes:  0. '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 798.81  on 1166  degrees of freedom
Residual deviance: 474.41  on 1157  degrees of freedom
AIC: 494.41

Number of Fisher Scoring iterations: 25
```

Figure 6 Logistic Regression Summary

OUTPUT – Confusion Matrix

Evaluating the logistic regression model using confusion matrix we see that model has an 94.48% accuracy. The probability limit that shows the best results is 0.2 and it correctly classifies 96.5 % of non-platinum clients and 77.8% of platinum clients. The lower prediction rate of platinum clients might be due to the fact that some platinum clients are not utilizing the platform as much as they should.

Confusion Matrix and Statistics

	Reference	
Prediction	0	1
0	671	19
1	24	65

Accuracy : 0.9448
95% CI : (0.9264, 0.9598)
No Information Rate : 0.8922
P-Value [Acc > NIR] : 0.0000001807

Kappa : 0.7204

McNemar's Test P-Value : 0.5419

Sensitivity : 0.9655
Specificity : 0.7738
Pos Pred Value : 0.9725
Neg Pred Value : 0.7303
Prevalence : 0.8922
Detection Rate : 0.8614
Detection Prevalence : 0.8858
Balanced Accuracy : 0.8696

'Positive' Class : 0

Figure 7 Logistic Regression Confusion Matrix

BUSINESS INSIGHTS

1. Qualtrics is making 70% of its annual revenue from top 15% of its clients. Hence, identifying these clients beforehand can help the sales, marketing and support teams to target their efforts at these clients. This will help in their acquisition as well as retention.
2. Decision tree model can use the marketing data to identify the top 15% of clients with 98% accuracy. This prediction can help marketing teams to channel their campaign efforts at these clients.
3. Logistic regression model uses the platform usage data to predict top 15% of clients with 94.5% accuracy. This prediction can help support teams understand the behavior of the top clients and segregate their support requests from all other requests.
4. Logistic regression model also identifies the platinum clients who are not utilizing the platform as much as they should. This could be used as an insight to target these clients with marketing campaigns that show how to utilize the Qualtrics platform in a much more efficient way.
5. Both these models can be used in combination to predict the platinum clients with maximum accuracy.