

Agenda

What is Customer churn?

Types of Churn and Causes for churn?

Factors that affect customer churn

Predict Customer Churn using R & Python

Action Plans to Prevent Customer Churn

What is a churn?

- Churn is also known as attrition or turnover or defection
 For ex, Customer attrition, Employee turnover, Customer defection
- •When your customers stop subscribing to a service or spending less with you or completely stop buying products.
- * Churn rate is an impossible substitution of the number of participants who discontinue their use of a service to the average number of total participants during a period

Churn Rate No. of user turned inactive / Avg. no. of user in a period

Types of Churn

- Involuntary Customers are forced to discontinue service and/or payments Example: Credit card expiration
- Voluntary Customers make the choice to leave or discontinue the service due to various reasons



Why customers churn?

- Due to competitors best offers
- Inadequate services
- Bad Customer Service
- * Product Issues

Problem Definition

Background

Telecom Churn Analysis

Churn (loss of customers to competition) is a problem for telecom companies because it is expensive to acquire a new customer and companies want to retain their existing customers. Most telecom companies suffer from voluntary churn. Churn rate has strong impact on the life time value of the customer because it affects the length of service and the future revenue of the company. For example if a company has 25% churn rate then, the average customer lifetime is 4 years; similarly a company with a churn rate of 50%, has an average customer lifetime of 2 years.

In the targeted represent the company tries to identify in advance customers who are likely to churn. The company then targets those customers with special programs or incentives. This approach can bring in huge loss for a company, if churn predictions are inaccurate, because then firms are wasting incentive money on customers who would have stayed anyway.

Problem Definition

Churn Analysis

Business Problem

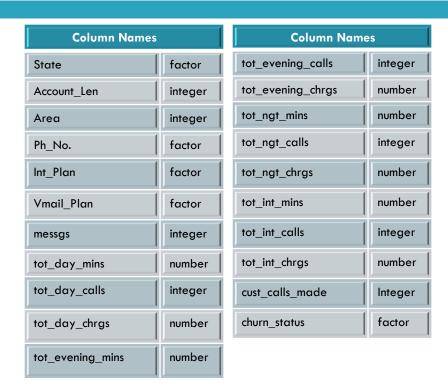
- Strengthen churn detection strategy.
- Telecom service provider needed to overhaul the customer experience by moving away from an ineffective and manual "one size fits all" approach, to creating 1-to-1 personalized experiences for its customers.
- Use historical data on telecom customer data to identify customers those are at risk of churn

Project Objective

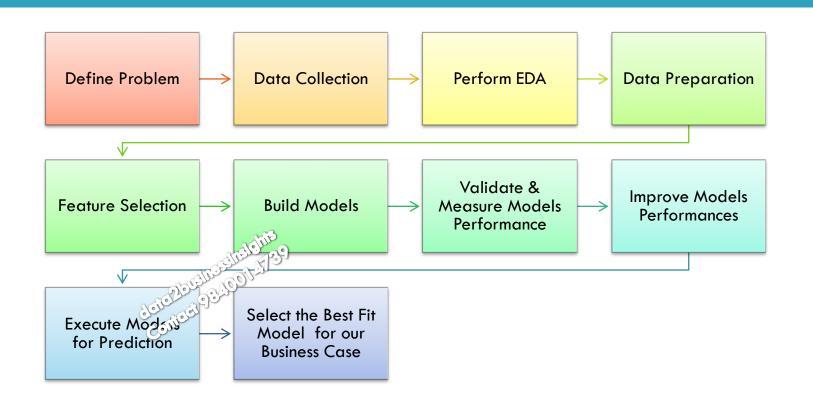
- To predict Customer Churn.
- Highlighting the main variables/factors influencing Customer Churn.
- * Use various ML algorithms to build prediction models, evaluate the accuracy and performance of these models.
- * Finding out the best model for our business case & providing executive Summary.

Dataset Description

- Source dataset is in txt format with csv.
- Dataset contains 4617 rows and 21 columns
- There is no missing values for the provided input dataset.
- Churn_status is the variable which notifies whether a particular customer is churned or not. And we will be deseloping our models to predict this variable andy.
- The predictor Ichel would be
 - 1 : Custo Ser will churn
 - 0 : Customer would not churn



Model Building Steps





In this project, we will be addressing Customer churn problem for a fictitious US Telecom company. Telco customer has provided us the usage pattern for 4617 customers for over a period of time along with the info whether particular customer is churned or not. Utilizing the input data, we have to build a model which can predict the customer going to be churned well in advance.

The input data has been revided by Telco Customer in txt format with comma separated values. Read in the revision R workbook by appropriate methods ensuring without loosing the data. Here for the loading the data into R we have used read table command with sep = ',' . As columns names were not part of input dataset and we have provided them separately, hence we have provided the colonames as well to read table command.



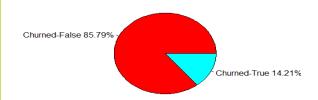
Perform general housekeeping activities such as checking for the completeness of the content, look at the dimensions & review the structure of input dataset, peek into the data, summarize the data, and get a snapshot of all the features. Check if there is any missing data for particular customers, luckily for our case there is no missing data.

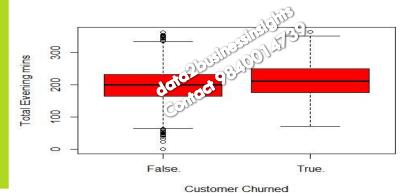
As a final step, transform the data into a format ready to apply with the algorithms and modeling.

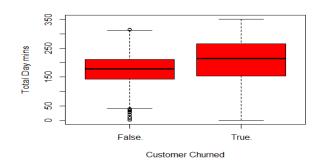
Perform EDA (exploratory sata analysis) to understand the data and its applicability on the problem. Understand how the data and its features are interrelated & correlated, evaluate presence of satiliers and its effects. Here we will use various box plots and bar plots to understand what are the features which majorly impact on our outcome variable 'churn_status'. We will see in our next slides the various outcomes from our EDA.

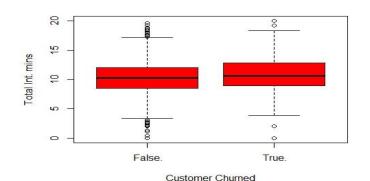


Percentage of Customer Churned

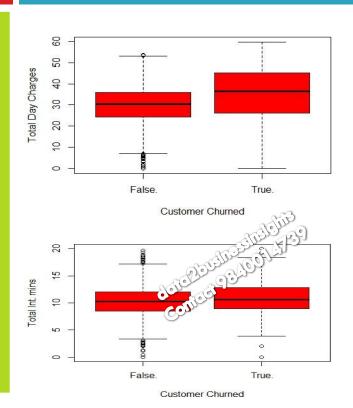


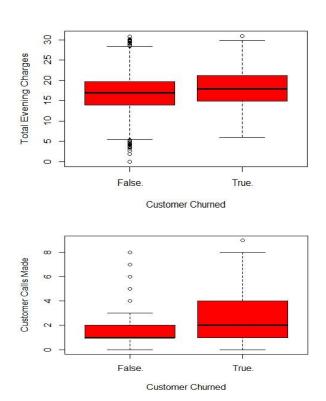




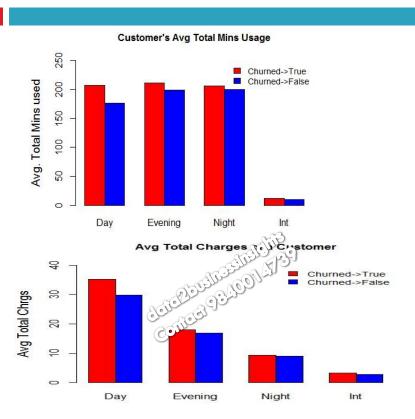


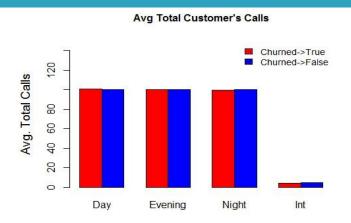


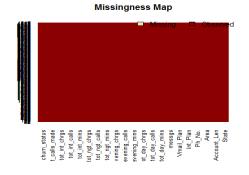














Main Observations from our EDA

- Usage pattern for Churned Customers which mainly includes minutes (Day/Evening/Night/Int) and their charges incurred is higher than the Not Churned Customers. So, Churned Customers seems to be relatively high value customers & more business relevant.
- Both Box and Bar plots are complementing each other with their findings.
- * Box plot for Customer calls made gives us enough proofs that customers going to be Churned Call Customer care many times before getting churned.
- Lucky enough to have no missing values in our input dataset.
- From Survival Analysis it is confirmed that Customer Calls Made & International Plan are most influential features. Total Day Mins Used is the also one suggested by our traditional EDA methods.



Feature Selection After EDA, its now possible for us to identify the major features/variables which influence significantly Customer Churn, we will now use these features for our model preparation. Further we can Drop the features that do/may not influence our outcome variable.

Before starting with Building Model, we will segregate our input data in 3 datasets with proportions 60(train):20(test):20(pred for final prediction).

As our business case is reconsisting assification model, we will be building models for all possible ML algorithms which can be applied for Classification problem. Our approach will be to proceed with Malgorithm one by one and at the same time will be validating/improving the each ML algorithm outcomes.



Logistic Regression

Now lets Test the above build model on test dataset, as the model is a logistic regression we assume for any observation if the probability of customer to be churned is > 0.5 we will assume its not churned. Lets validate and measure our assumption .

Logit Pred	False.	True.
False.	741	115
True.	13	13

Accuracy	Specificity	Sensitivity
85.48	98.27	10.15



Variating the Threshold Value

In previous step we used our threshold value as 0.5 to decide whether customer is churned or not depending upon whether the value is above 0.5 or not. Lets variate the threshold value a bit and check if the performance of the model is increasing or not. Upon testing for various values (0.4 & 0.3), it was found that the performance of the overall model increases when threshold value is selected as 0.3. We request you please refer to the attached R code for more details on this.

Final Prediction with the shall value as 0.3 on pred dataset, and calculate the model performance for one has time for logit model.

Logit Pred	False.	True.
False.	756	86
True.	67	41

Accuracy	Specificity	Sensitivity
84.21	91.85	32.28



K-nearest neighbour

knn_model <- knn(train = knn_train, test = knn_test,cl = train\$churn_status, k=53) k= 53 has been calculated as the sqrt of number of observations 2785

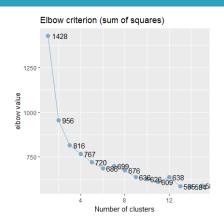
Validating the above model against our test data set, lets see what our confusion metrix and other performance paramovers convey:-

Knn.Pred	False.	True.
False.	752	111
True.	2	1 <i>7</i>

Accuracy	Specificity	Sensitivity
87.55	99.73	13.28



- Using Elbow Method to identify Optimum value of k.
- -> Normalize the data to supress the scale irregularities as knn works on distance between the nearest neighbors.



Final Prediction after polying above improvement techniques on pred dataset, lets calculate the Final performance for knn algorithm

Knn.Pred	False.	True.
False.	805	84
True.	18	43

Accuracy	Specificity	Sensitivity
89.26	92.04	33.85



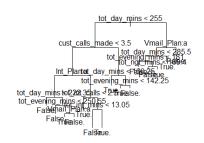
Decision Trees

tree_model <- tree(churn_status \sim .-State-Area-Ph_No., data = train)

Classification tree: tree

Variables actually used in tree construction:

"tot_day_mins" "cust_calls_made" "Int_Plan" "tot_evening_mins" "Vmail_Plan" "tot_int_calls" "tot_int_mins" "tot_ngt_mins", Number of terminal nodes: 15



Validating the above moving gainst our test data set, lets see what our confusion matrix and other performance where convey:-

Tree.Preu	False.	True.
False.	739	38
True.	15	90

Accuracy	Specificity	Sensitivity
93.99	98.01	70.31

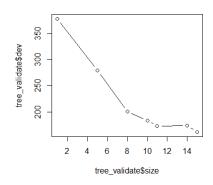


-> Pruning, so that the model doesn't overfit the dataset.

tree_validate <- cv.tree(object = tree_model,FUN = prune.misclass)

plot(x=tree_validate\$size,y=tree_validate\$dev, type="b")

From the plot one can see that tree_validate\$dev diff is same from for tree levels 11-14, so can we assume the best tree level size to be 12 (rather than original number 15) at the cost of some bias.



Final Prediction after and its above improvement techniques on pred dataset, lets calculate the Final model performance for decision trees algorithm

Tree Pre	False.	True.
False.	801	44
True.	22	83

Accuracy	Specificity	Sensitivity
93.05	97.32	65.35

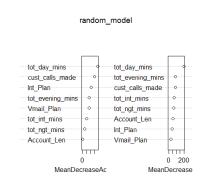


Random Forest

from EDA.

random_model <- randomForest(as.factor(churn_status)~ Account_Len + Int_Plan + Vmail_Plan + tot_day_mins + tot_evening_mins + tot_ngt_mins + tot_int_mins + cust_calls_made,data = train,ntree = 2000, importance = TRUE)

Please note the variable importance plot also compliments our earlier findings



Validating the above modular pinst our test data set, lets see what our confusion matrix and other performance parameters onvey:-

Ran.Pred	False.	True.
False.	746	44
True.	8	84

Accuracy	Specificity	Sensitivity
94.10	98.93	64.62



-> Performance Boosting by Cross Validation with folds k=10. train_control <- trainControl(method="cv", number=10)

```
random_model_train <- train(as.factor(churn_status)~ Account_Len + Int_Plan + Vmail_Plan
+ tot_day_mins + tot_evening_mins + tot_ngt_mins
+ tot_int_mins + cust_calls_made, data=train, method="rf",
metric="Accuracy", trControl=train_control)
```

Final Prediction after and its above improvement techniques on pred dataset, lets calculate the Final model performance for Random Forest algorithm

Ran Prea	False.	True.
False.	808	47
True.	15	80

Accuracy	Specificity	Sensitivity
93.05	98.1 <i>7</i>	62.99



Support Vector Machine

svm_model<-svm(churn_status,data=train,kernel='radial',gamma=1,cost=100)</pre>

SVM-Type: C-classification SVM-Kernel: radial

cost: 100

gamma: 1 Number of Support Vectors: 2785

Validating the above modularism performance parameters on vey:-

SVM.Pred	False.	True.
False.	754	128
True.	0	0

Accuracy	Specificity	Sensitivity	
85.49	100	0	



```
-> Using Cross Validation technique to optimise the best values for gamma and cost.

svm.tune <- tune(svm,churn_status ~ Account_Len + Int_Plan + Vmail_Plan
+ tot_day_mins + tot_evening_mins + tot_ngt_mins
+ tot_int_mins + cust_calls_made,data=train,kernel='radial',ranges = list(cost = c(0.1,1,10,100,1000), gamma = c(0.5,1,2,3,4)))
```

Final Prediction after applying above improvement techniques on pred dataset, lets calculate the Final model performace for SVM algorithm

Ran Prea	False.	True.
False.	813	59
True.	10	68

Accuracy	Specificity	Sensitivity
92.74	98.78	55.54



In the past few slides for our Classification Business Problem, we have created many models, measured their performances, applied various improvement techniques and at the end of each step we created best model for each corresponding ML algorithms. As we have the performance results with us for the models, lets select the best one out of them which suits best for our business case.

	Logit	kNN	Decision T	Random F	SVM
Accuracy	84.21	89.26	93.05	93.05	92.74
Specificity	53.8500	92.04	97.32	98.17	98.78
Sensitivity	৩2.28	33.85	65.35	62.99	55.54

Decision Trees (CART) model / Random Forest suits best for our business scenario.

Summary



- Customers Churned were high business Value customers, their usage pattern was high as compared to not churned.
- Analysis from Survival Model suggests that we are able to retain all our customers in our first 50-70 days, then in next 50-70 days we are able to retain 80% and further this % reduces to 40% by the end of 200 days.
- Customers who are having International Plan OR those who call Customer Care for their service related queric charn more. These two factors are the most influential ones for a customer to be charned. So, one possible measure can be like company can always keep a check for all the customers who call Customer Care more frequently and can pool them in "Possible to be Churned Customers in future", so that company can take their issues on priority or address them rapidly. Same measure/check can be implemented for International Plan Customers as well.

Summary



❖ Out of various models developed for Prediction of Customers going to be churned, Decision Trees Model (CART) and Random Forest suits best for our business case. Both the models are almost similar w.r.t each other. The accuracy for both the models is almost same i.e. 93.05 %. The only difference between their performance is, RF is able to correctly predict 62.99% of customers to be churned as true but also with the false out ratio of 1.83%. That means apart from predicting 62.99% correctly it is also predicting incorrectly for some 1.83% customers as churned(which is not true).

Where as Decision Tree(CART) model is able to predict correctly 65.35% customers to be churned as true but a predicts incorrectly for some 2.68% customers as churned(which also is not true). So it see that upon Business which model it would like to use if it would like to go for higher predictive power at cost of some Fall Out ratio/error deviation Decision Tree is best. ELSE if company wants to stress more on minimizing error rate than RF can be the best bet.

