Churn Modeling and Analysis

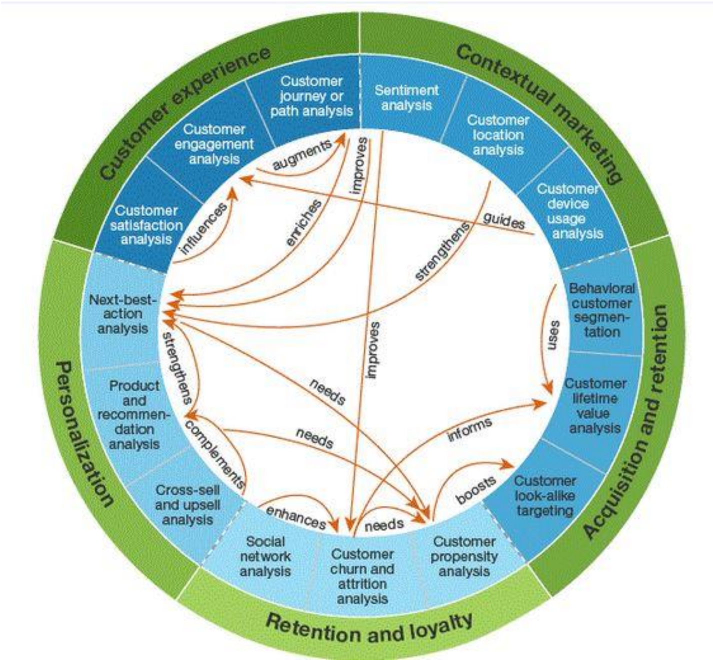
Made By: Mayur Shahasane

Customer churn is a serious problem in the competitive world. Almost every organization face this issue of customer churn. Best way to reduce the churn is by identifying the customers who may churn and then take some action to retain the customer and keep them loyal and happy. Customer churn analysis is important part of customer analysis and we have various frame work available for the same. One of the most popular frame-work for customer analysis is given by Forrester.

What is Forrester customer analytics framework?

The scale and diversity of customer data provide rich new sources of insight, letting firms engage with customers in new ways and enable the digital disruption of entire industries. This will help us to understand customer and provide better understanding and insights. Forrester Research Inc. has provided with reference model called Forrester customer analytics model and Framework, which can be used as a reference for developing any solutions for customer analytics.

Shown below:



Customer experience
Contextual marketing
Acquisition and retention
Retention and loyalty
Personalization
Customer journey or path analysis
Sentiment analysis
Customer location analysis
Customer device usage analysis
Behavioral customer segmen-tation
Customer lifetime value analysis
Customer look-alike targeting
Customer propensity analysis
Customer churn and attrition analysis
Social network analysis
Cross-sell and upsell analysis
Product and recommen-dation analysis
Next-best-action analysis
Customer satisfaction analysis
Customer engagement analysis
augments
improves
guides
influences
enriches
strengthens
uses
strengthens
needs
improves
informs
needs
complements
boosts
enhances
needs

Fig: Forrester customer analytics model

The above model shows major domains in customer analytics, along with the dependencies between each domain. Each domain is further broken down to 15 sub domains and each sub domains places important role in customer analytics.

1) Contextual Marketing:

Customers are everything to any organization but there is very rare chance that customer will approach you first. In the completive world the most important part is to target best clients who will get engage with us in near future.

Contextual marketing refers to online and mobile marketing that provides targeted advertising based upon user information, such as the search terms they're using or recent web-browsing activity (See also Computational Marketing).The goal is to present ads to customers representing products and services they are already interested in. For example, a customer performs an Internet search for cars and fuel efficiency. Afterwards, they check their daily news website and the ads which show up alongside of news includes for hybrid cars. The customer, already thinking about saving fuel on their commute, clicks on the ad to check out the latest hybrids.

2) Acquisition and retention:

Once we get the details of potential customer then the next task is to acquire the customer and get him engage with our products and services.

Best example will be POC done to our clients. Here we know what they require and now it is our task to acquire those clients by providing them confidence with our product and/or service.

3) Retention and loyalty:

Customer retention refers to the activities and actions companies and organizations take to reduce the number of customer defections. The goal of customer retention programs is to help companies retain as many customers as possible, often through customer loyalty and brand loyalty initiatives.

4) Personalization:

This involves generation of recommendation in which customer will be interested in, identifying the best possible action to make business from the customer along with keeping customer happy.

5) Customer Experience:

Customer experience is most important for the business. When customer has good experience they remain loyal and are retained with the business. We can understand customer experience by understanding customer engagement with the organization, satisfaction with the requirement fulfilled also by understanding the journey from the start of service.

Modeling Concepts:

Whenever we need to work on data modeling we follow certain approach given below:

1) Understand the use-case (Problem)

Understanding the use-case is very important as this helps us in taking right step in getting accurate and desired result.

Eg: Identify and predict the customer who may churn in near future.

So we are clear that we need to do some sort of prediction also we need to classify the data based on some condition to churned or non-churned class. We can conclude that the solution will be based on Machine Learning (Classifier Problem).

2) Understand the data

This is one and most important step while working with discovery phase. This involves understanding of meanings and value of data provided. It also includes clarifying the doubts in data if present.

3) Data discovery

We gather, analyze relationship among the data. We can use different approach for the same.

1) Using python libraries such as seaborn, matplotlib, etc.
2) Visualization tools.

The best approach is to use visualization tools as it reduces the coding and also give better picture of data. We can track the changes in data using UI controllers and filters. This helps in understanding the story.

The data which has the highest relationship (dependency) with the problem are used as Features during the modeling phase.

4) Featuring engineering

Once we know the relationship among the data we work for getting the Features for modeling. Feature engineering is the process of using domain knowledge of the data to create features that make machine learning algorithms work. Feature engineering is fundamental to the application of machine learning, and is both difficult and expensive. The need for manual feature engineering can be obviated by automated feature learning.

A feature is an attribute or property shared by all of the independent units on which analysis or prediction is to be done. Any attribute could be a feature, as long as it is useful to the model. The features in the data are important to the predictive models and will have influence on the results. The quality and quantity of the features will have great influence on whether the model is good or not.

General approach for Featuring engineering:

1. Brainstorming or Testing features
2. Deciding what features to create
3. Creating features
4. Checking how the features work with your model
5. Improving your features if needed
6. Go back to brainstorming/creating more features until the work is done.

5) Modeling

There are many algorithms available available for prediction and classification so we don't need to make all of our own. In python we have sklearn library which provide almost all the models which we require for machine learning.
For churn modeling following models are mostly used:
(To know modeling using sklearn in depth click on names)

1) Random Forest Classifier
2) Gradient Boosting Classifier
3) SVC – Linear SVC
4) Logistic Regression
5) Naive Bayes Classifier

5.1) Random Forest Classifier

Random forests or **random decision forests** are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of overfitting to their training set.

A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and use averaging to improve the predictive accuracy and control over-fitting. The sub-sample size is always the same as the original input sample size but the samples are drawn with replacement if *bootstrap=True* (default).

5.2) Gradient Boosting Classifier

Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees as decision trees has a short depth (usually 5-7 layers). It builds the model in a stage-wise fashion like other boosting methods do, and it generalizes them by allowing optimization. This functional gradient view of boosting has led to the development of boosting algorithms in many areas of machine learning and statistics beyond regression and classification.

5.3) Linear SVC

“Support Vector Machine” (SVM) is a supervised machine learning algorithm which can be used for both classification and regression challenges. However, it is mostly used in classification problems. In this algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiate the two classes very well (look at the below snapshot).

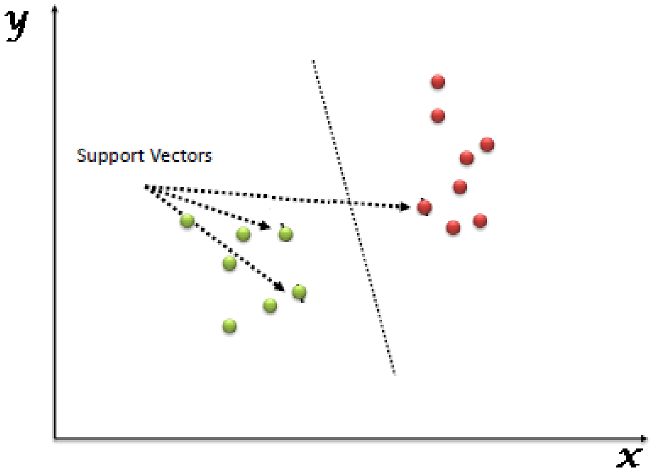


Fig: Hyper Plane differentiating the classes

Support Vectors are simply the co-ordinates of individual observation. Support Vector Machine is a frontier which best segregates the two classes (hyper-plane/ line).

5.4) Logistic Regression

Logistic regression measures the relationship between the categorical dependent variable and one or more independent variables by estimating probabilities using a logistic function.

Logistic Function: Logistic regression is named for the function used at the core of the method, the logistic function. The logistic function, also called the sigmoid function was developed by statisticians to describe properties of population growth in ecology, rising quickly and maxing out at the carrying capacity of the environment. It's an S-shaped curve that can take any real-valued number and map it into a value between 0 and 1, but never exactly at those limits.

sigmoid function = 1 / (1 + e^-value)

e: It is the base of the natural logarithm exponent
value: It is the actual numerical value that you want to transform.

Below is a plot of the numbers between -5 and 5 transformed into the range 0 and 1 using the logitic function.

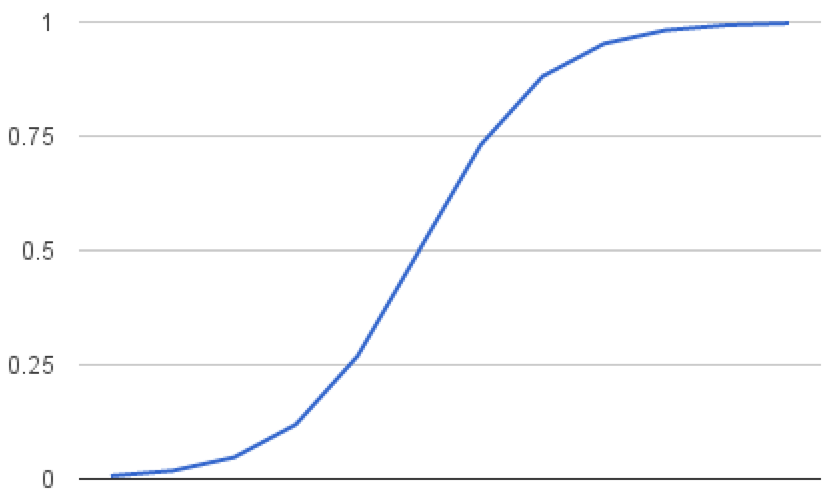


Fig: Sigmoid function

5.5) Naive Bayes Classifier

In machine learning, *naive Bayes classifiers* are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong (naive) independence assumptions between the features. Naive Bayes classifiers are highly scalable, requiring a number of parameters linear in the number of variables (features/predictors) in a learning problem. The time taken by this algorithm to evaluate the results is linear time, not like other classifier which uses expensive iterative approximation that is time consuming.

When dealing with continuous data, a typical assumption is that the continuous values associated with each class are distributed according to a Gaussian distribution. For example, suppose the training data contains a continuous attribute. Another common technique for handling continuous values is to use binning to discretize the feature values, to obtain a new set of Bernoulli-distributed features; some literature in fact suggests that this is necessary to apply naïve. GaussianNB implements the Gaussian Naive Bayes algorithm for classification.

Train & test Split

Another important part of modeling is splitting the data set into test and train dataset. As train and test split helps in avoiding the overfitting of the model. We can perform this using sklearn method or by using manual splitting. In general we use sklearn's train_test_split method. Now the splitting ration depends on the type of data and number of records, in general we split into 70% train and 30% test or 75% train and 25% test when we have good amount of data typically above 50000 records, when we have very less record we my go in splitting the data in 50% each or 60% train and 40% test.

Model Evaluation Basics

We perform model evaluation to identify best model for our purpose. For model evaluation we check value of following parameters.

1) Accuracy: It refers to the closeness of a measured value to a standard or known value.

The formula for **accuracy** is:

$$A(M) = (TN+TP)/(TN+TP+FP+FN)$$

TP is the number of true positives
In simple language, *Predicted as the customer has churned and they really churn.*

TN is the number of true negatives:
In simple language, *Predicted as the customer will remaining with us and they do remain.*

FP is the number of false positives:
In simple language, *Predicted as they will churn, but they stay with us.*

FN is the number of false negatives: In simple language, *Predicted as the customer won't churn but they actually churn.*

We need to really take into consideration of FN in model evaluation along with others.

2) **Sensitivity/Recall** - how well the model *recalls*/identifies those that will leave. Also known as the true positive rate.

   The formula for **Sensitivity** is: TP / (TP + FN)

   In simple, recall shows correctly predicting the churner.

3) **Specificity -** How well the model identifies those that will stay.
   The formula for **Specificity** is: TN / (TN + FP)

4) **Precision-** This help to identify how believable is the model.
   The formula for **Precision** is: TP / (TP + FP)
   A low precision model will tell you; those who are predicted leaving but that are actually staying.

5) **F1 score-** This is the harmonic mean between precision and recall or the *balance*.
   The formula for **F1 score** is: 2 * (precision * recall)/(precision + recall)

   F1 **score** is also called F-**score** or **F**-measure. The F1 **score** can be interpreted as a weighted average of the precision and recall, where an F1 **score** reaches its best value at 1 and worst at 0.

   For better understanding you may refer this site.

   In practical business scenario we actually consider Area under curve (AUC) matric for model evaluation and select the model with highest AUC value. AUC value lies in between 0 to 1.

   Let's have overview of AUC

   The Area under the curve (AUC) is a performance metrics for binary classifiers. The AUC is the Area under the receiver operating characteristic curve (ROC); it captures the extent to which the curve is up in the Northwest corner. Higher the AUC is better is the model.

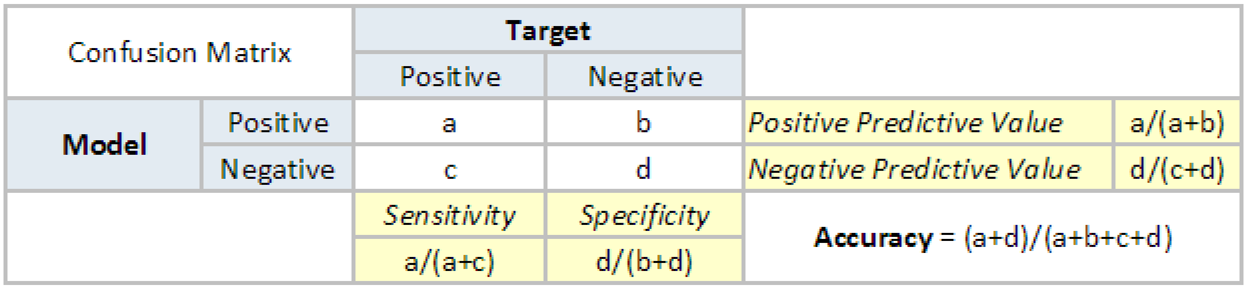
A score of 0.5 is no better than random guessing. 0.85 ~ 0.09 would be a very good model but a score of 0.9999 would be too good to be true and will indicate overfitting.

In python we evaluate the model by using Confusion matrix and also by calculating the AUC value.

Confusion Matrix

A confusion matrix is an N X N matrix, where N is the number of classes being predicted. For the problem in hand, we have N=2, and hence we get a 2 X 2 matrix. Here are a few definitions, you need to remember for a confusion matrix :

- **Accuracy** : the proportion of the total number of predictions that were correct.
- **Positive Predictive Value or Precision** : the proportion of positive cases that were correctly identified.
- **Negative Predictive Value** : the proportion of negative cases that were correctly identified.
- **Sensitivity or Recall** : the proportion of actual positive cases which are correctly identified.
- **Specificity** : the proportion of actual negative cases which are correctly identified.



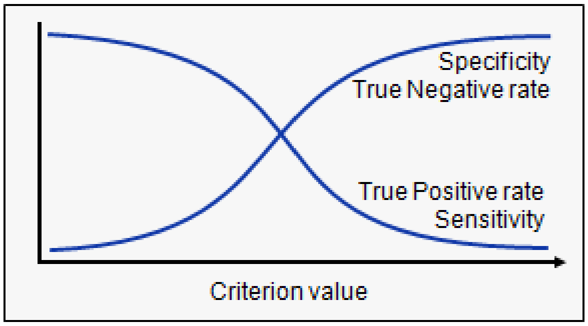
Confusion Matrix		**Target**			
		Positive	Negative		
Model	Positive	a	b	*Positive Predictive Value*	a/(a+b)
	Negative	c	d	*Negative Predictive Value*	d/(c+d)
		Sensitivity	*Specificity*	**Accuracy** = (a+d)/(a+b+c+d)	
		a/(a+c)	d/(b+d)		

Area under the ROC curve (AUC – ROC)

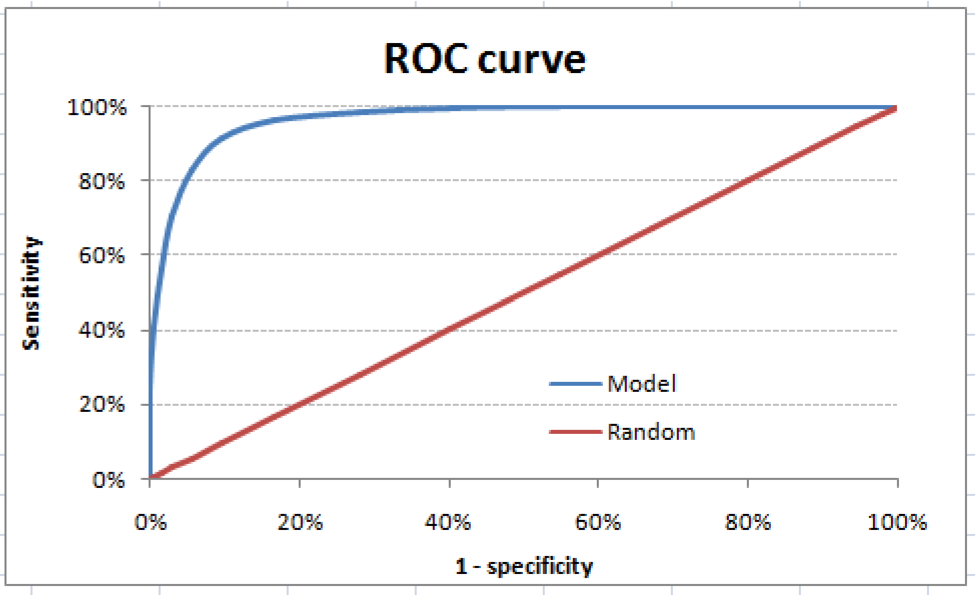
This is one of the popular metrics used in the industry. The biggest advantage of using ROC curve is that it is independent of the change in proportion of responders. This statement will get clearer in the following sections.

Let's first try to understand what is ROC (Receiver operating characteristic) curve? If we look at the confusion matrix in previous section, we observe that for a probabilistic model, we get different value for each metric. Hence, for each sensitivity, we get a different specificity.

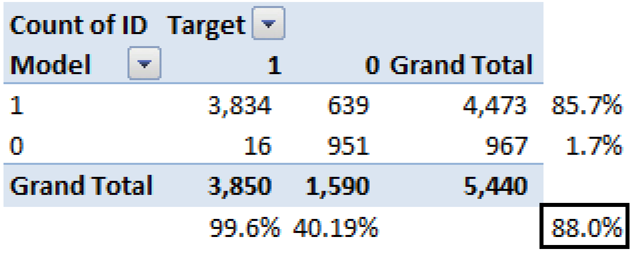
The two vary as follows:



The ROC curve is the plot between sensitivity and (1- specificity). (1- specificity) is also known as false positive rate and sensitivity is also known as True Positive rate. Following is the ROC curve for the case in hand.



Let’s take an example of threshold = 0.5 (refer to confusion matrix). Here is the confusion matrix :



Count of ID	Target ▾			
Model ▾	1	0	Grand Total	
1	3,834	639	4,473	85.7%
0	16	951	967	1.7%
Grand Total	3,850	1,590	5,440	
	99.6%	40.19%		88.0%

As you can see, the sensitivity at this threshold is 99.6% and the (1-specificity) is ~60%. This coordinate becomes on point in our ROC curve. To bring this curve down to a single number, we find the area under this curve (AUC).

Note that the area of entire square is 1*1 = 1. Hence AUC itself is the ratio under the curve and the total area.

Following are a few thumb rules:

- .90-1 = excellent (A)
- .80-.90 = good (B)
- .70-.80 = fair (C)
- .60-.70 = poor (D)
- .50-.60 = fail (F)

6) Predict the result.

Once we have trained model we can use the same model to predict the churn and we have required solution for the given problem.

Let's understand the churn modeling concept by using real life example.

Current Problem Statement:

There is a US based telecom company who are facing the problem with customer churn and that company has approached us (Acrotrend) to help them.

Then are seeking help to identify and predict the customer who will churn in near future and provide the list of the same.

Data Set Provided Overview:

For our modeling they have provided with dataset of existing users for the month of May 2017. And also they have provided us with the description of different columns so that we can understand data better.

Company keeps track of following customer activities listed below.

- State: They are state code (2 character) representing the state from US.
- Account length: This represents customer timeline.
- Area code: It is to specify the exact location of customer, just like pin-code /postcode.
- Phone number: can be used as customer identifier.
- International plan: It states if the customer as opt for international service plan.
- Voice mail plan: It states if the customer as opt for voice mail service plan.
- Number vmail messages: Total voice-mail send.

- Total day minutes: Shows total minute spent in call during the day.
- Total day calls: Total number of calls during the day.
- Total day charge: Total charge the customer has to pay for all the calls during the day.
- Total eve minutes: Shows total minute spent in call during the evening.
- Total eve calls: Total number of calls during the evening.
- Total eve charge: Total charge the customer has to pay for all the calls during the evening.
- Total night minutes: Shows total minute spent in call during the night.
- Total night calls: Total number of calls during the night.
- Total night charge: Total charge the customer has to pay for all the calls during the night.
- Total intl minutes: Shows total minute spent in international calls.
- Total intl calls: Total number of international calls.
- Total intl charge: Total charge the customer has to pay for all the international calls.
- Customer service calls: Calls made to customer service center.
- Churn: Shows whether the customer has churned or not.

Link of dataset: https://www.kaggle.com/c/customer-churn-prediction/data
The dataset has 5000 records and 21 columns.

Data discovery phase

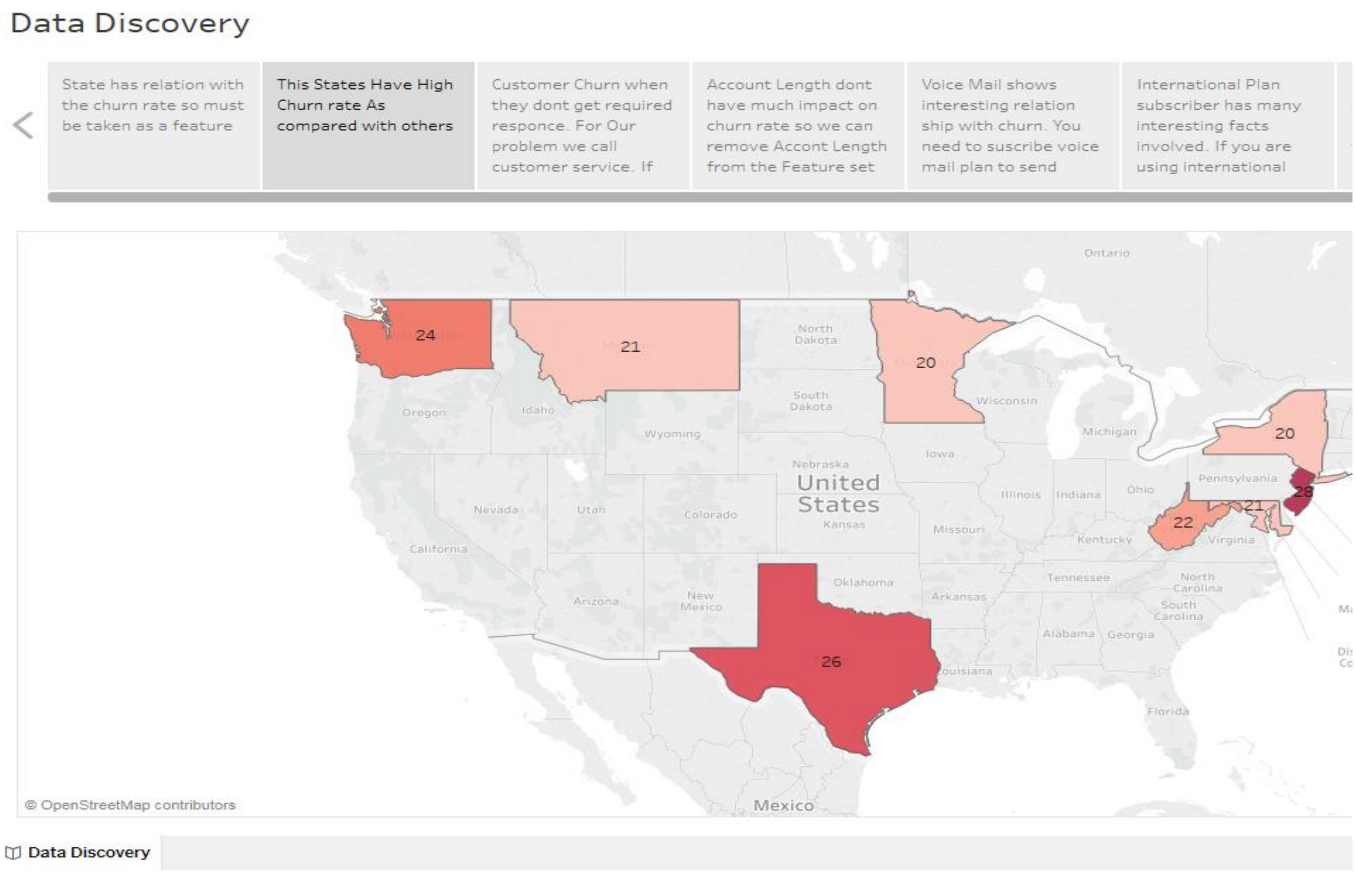
For the given dataset let's find some relationship using visualization tool. In our case we are using Tableau to determine the insights form the data.

By using visualization we should be able to conclude following:

- Whether selected filed is feature or not?
- Do we need to perform some calculation of fields?
- What type of dependencies we have among different columns?
- Do we have some common pattern or policy used in our data?
- Is dataset balance?

Few screenshot :

Let's check if state can be taken as feature or not. We saw that most of the churners are from few states only. As churn rate is not equally divided over states we can consider state in our feature.



Data Discovery

State has relation with the churn rate so must be taken as a feature

This States Have High Churn rate As compared with others

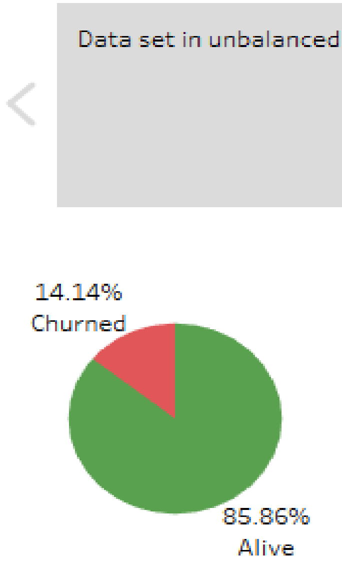
Customer Churn when they dont get required responce. For Our problem we call customer service. If

Account Length dont have much impact on churn rate so we can remove Accont Length from the Feature set

Voice Mail shows interesting relation ship with churn. You need to suscribe voice mail plan to send

International Plan subscriber has many interesting facts involved. If you are using international

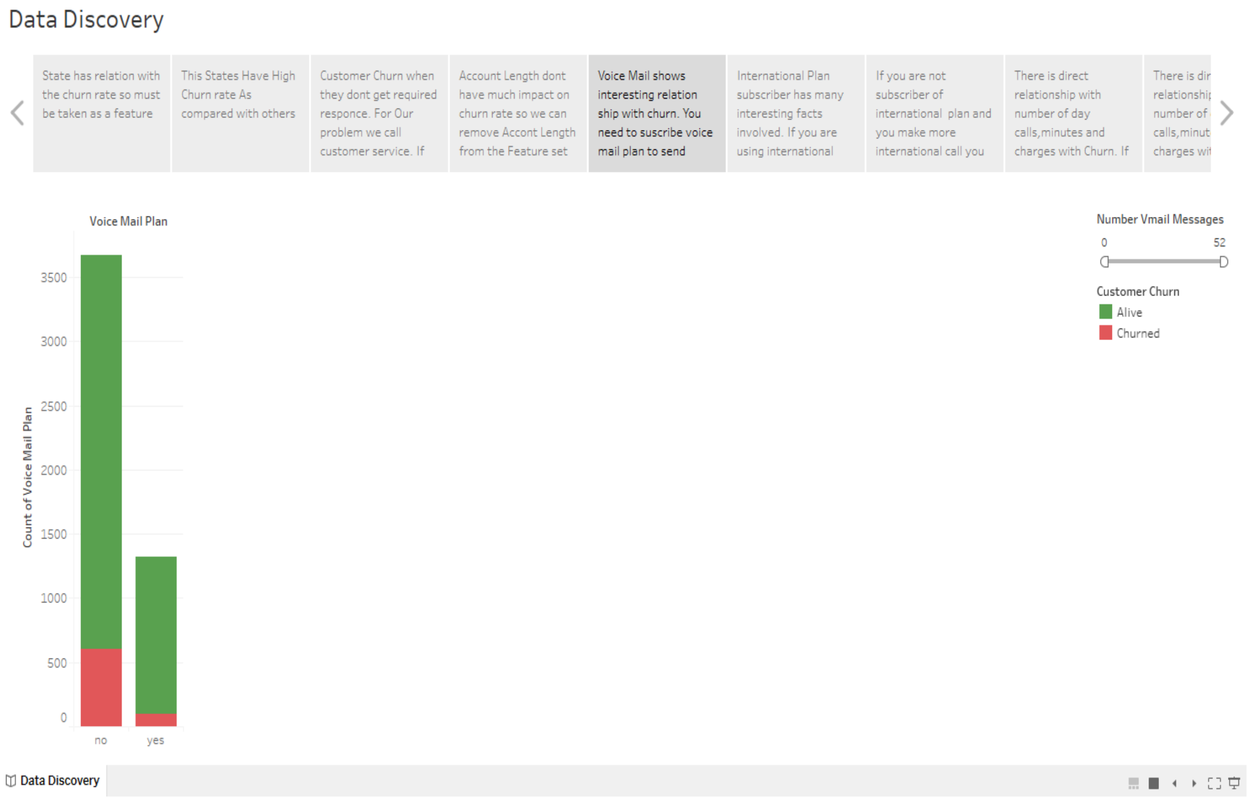
Fig: Above shown states have max number of churns



Data set in unbalanced

Fig: Show balancing of data set.

We can see from the pi chart, dataset is imbalance. As we have only 14.14% Churn data in dataset so it become difficult to model evaluation. ie: If model perform worst then too we will have accuracy around 86%. So we need different approach for model evaluation. Which we have seen in modeling section.



Data Discovery

State has relation with the churn rate so must be taken as a feature

This States Have High Churn rate As compared with others

Customer Churn when they dont get required responce. For Our problem we call customer service. If

Account Length dont have much impact on churn rate so we can remove Accont Length from the Feature set

Voice Mail shows interesting relation ship with churn. You need to suscribe voice mail plan to send

International Plan subscriber has many interesting facts involved. If you are using international

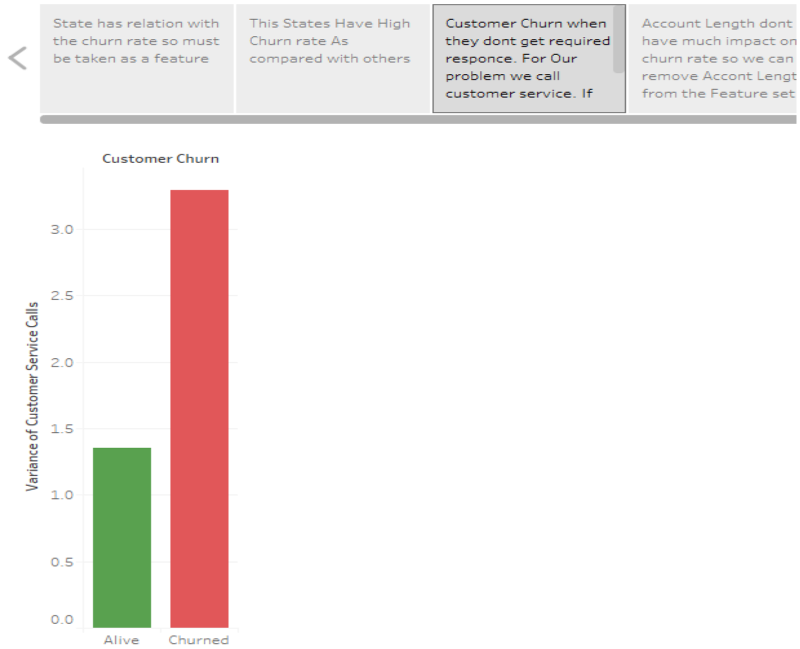
If you are not subscriber of international plan and you make more international call you

There is direct relationship with number of day calls,minutes and charges with Churn. If

There is dir relationshi number of calls,minut charges wi

Fig: Voice mail plan relationship with churn

We found that voice mail plan has policy applied: To send voice mail you shoud subscribe to voice mail plan. Futher we saw that churn rate is low for customer using voice mail plan. So we can consider this as feature.



State has relation with the churn rate so must be taken as a feature

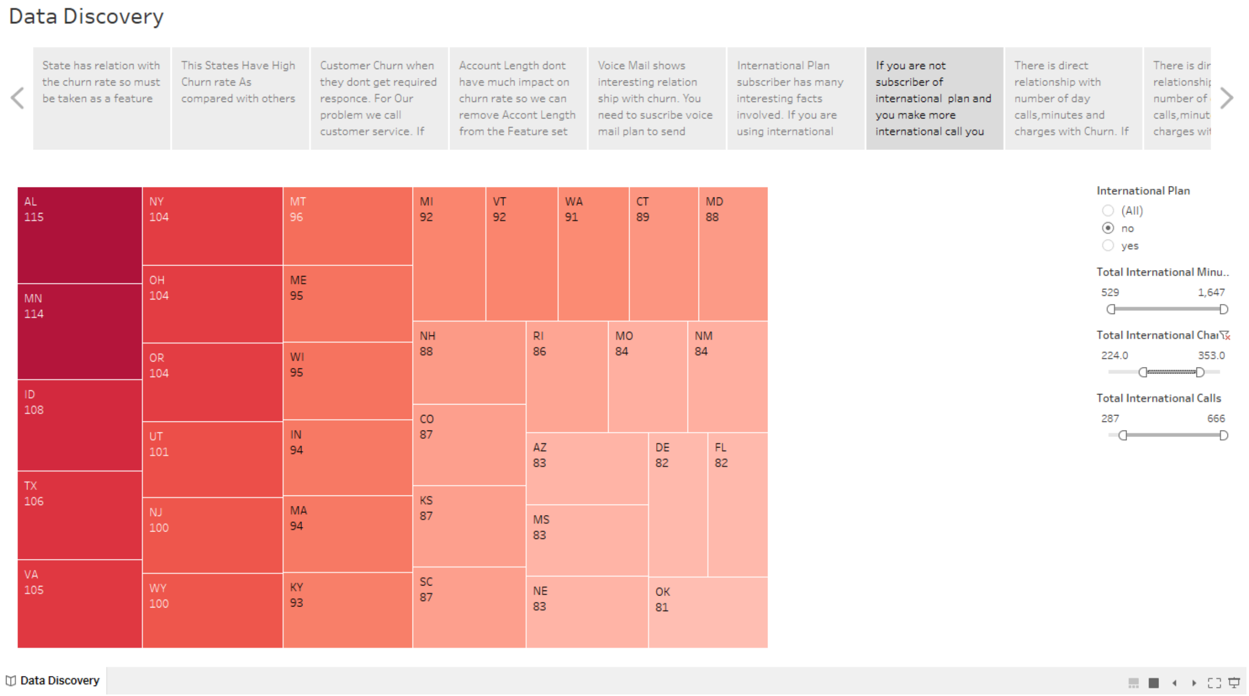
This States Have High Churn rate As compared with others

Customer Churn when they dont get required responce. For Our problem we call customer service. If

Account Length dont have much impact on churn rate so we can remove Accont Lengt from the Feature set

Fig: Customer Service calls role in churn

We can say that if customer is calling more then 3 times then they are more likely to churn



Data Discovery

State has relation with the churn rate so must be taken as a feature

This States Have High Churn rate As compared with others

Customer Churn when they dont get required responce. For Our problem we call customer service. If

Account Length dont have much impact on churn rate so we can remove Accont Length from the Feature set

Voice Mail shows interesting relation ship with churn. You need to suscribe voice mail plan to send

International Plan subscriber has many interesting facts involved. If you are using international

If you are not subscriber of international plan and you make more international call you

There is direct relationship with number of day calls,minutes and charges with Churn. If

There is dir relationshi number of calls,minut charges wi

International Plan
○ (All)
● no
○ yes

Total International Minu..
529 1,647

Total International Char
224.0 353.0

Total International Calls
287 666

Data Discovery

Fig: Relationship of state, international plan, international call (numbers of call,cost,minutes) with churn

Rest relationships and story board can be checked in Tableau file in repository.

Selection of Features

Based on above data discovery we found following relevant features which can be used for modeling:

- State
- International plan
- Voice mail plan
- Number vmail messages
- Total day minutes
- Total day calls
- Total day charge
- Total eve minutes
- Total eve calls
- Total eve charge
- Total night minutes
- Total night calls
- Total night charge
- Total intl minutes

- Total intl calls
- Total intl charge
- Customer service calls

Remaining columns did not showed relationship with churners or are not fit for modeling so we have excluded them from modeling.

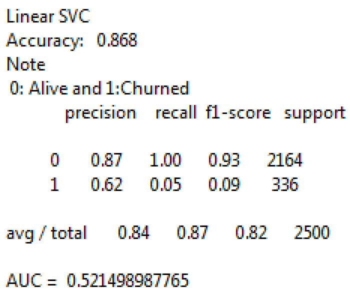
Now we need to decide on **train & test split:** Now in my case I had small dataset with very less churns so in this case I decided to go with 50% train and 50% test

Modeling

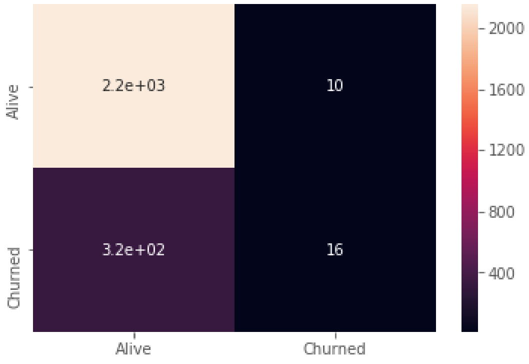
Modeling code can be referred in churn-modeling.ipynb (Jupyter Notebook)

Results of Models

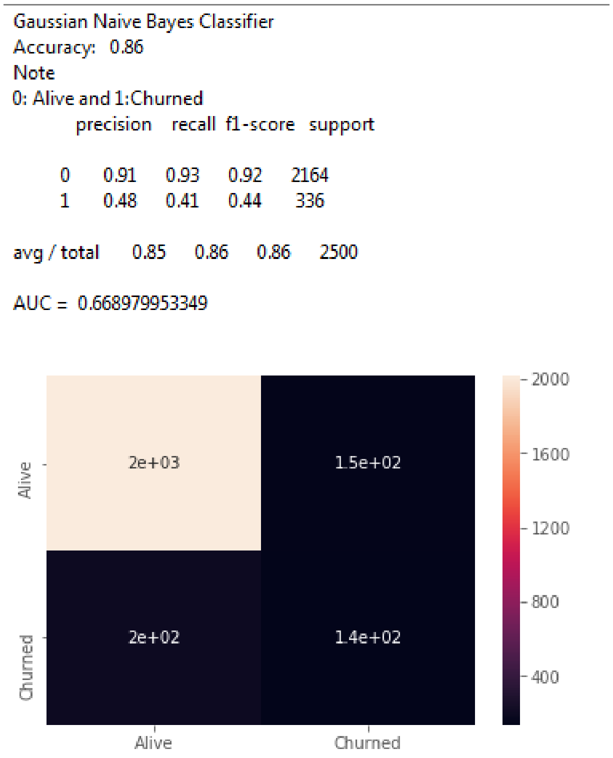
As our data set is imbalanced we will focus on following matrix evaluation: Area under curve (Most Important), accuracy, precision (Important), recall (Important), F1-score (Important).



```
Linear SVC
Accuracy:  0.868
Note
 0: Alive and 1:Churned
            precision    recall  f1-score   support

          0       0.87      1.00      0.93      2164
          1       0.62      0.05      0.09       336

avg / total       0.84      0.87      0.82      2500

AUC =  0.521498987765
```

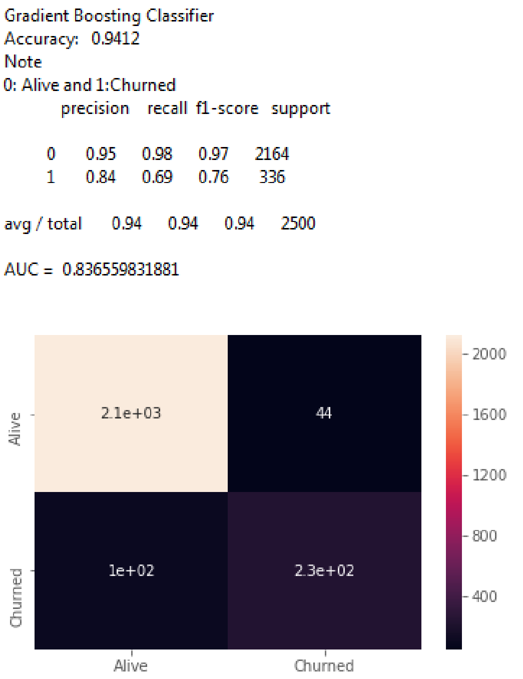


Linear SVC



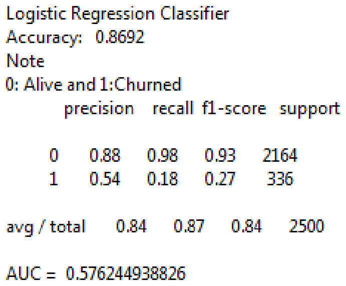
```
Gaussian Naive Bayes Classifier
Accuracy:   0.86
Note
0: Alive and 1:Churned
          precision    recall  f1-score   support

       0       0.91      0.93      0.92      2164
       1       0.48      0.41      0.44       336

avg / total       0.85      0.86      0.86      2500

AUC =  0.668979953349
```

Naive Bayes Classifier

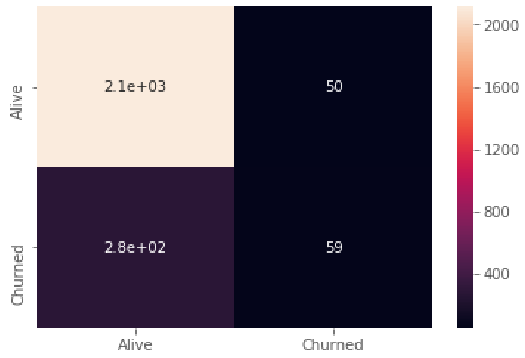


```
Gradient Boosting Classifier
Accuracy:   0.9412
Note
0: Alive and 1:Churned
          precision    recall  f1-score   support

       0       0.95      0.98      0.97      2164
       1       0.84      0.69      0.76       336

avg / total       0.94      0.94      0.94      2500

AUC =  0.836559831881
```

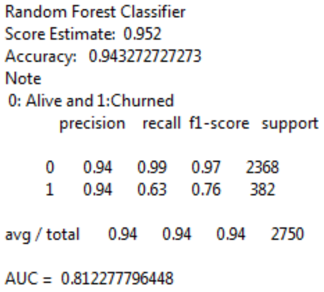
Gradient Boosting Classifier



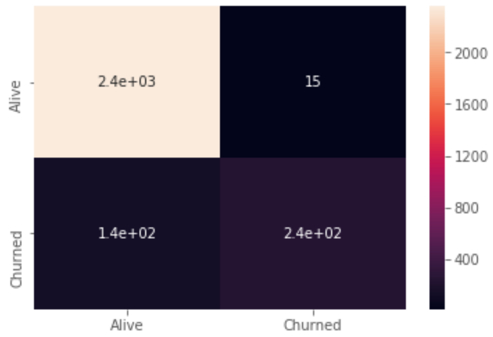
```
Logistic Regression Classifier
Accuracy:   0.8692
Note
0: Alive and 1:Churned
          precision    recall  f1-score   support

       0       0.88      0.98      0.93      2164
       1       0.54      0.18      0.27       336

avg / total       0.84      0.87      0.84      2500

AUC =  0.576244938826
```



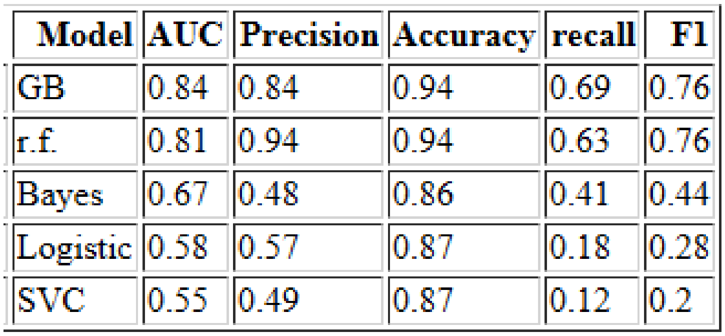
Logistic Regression



```
Random Forest Classifier
Score Estimate:  0.952
Accuracy:   0.943272727273
Note
 0: Alive and 1:Churned
          precision    recall  f1-score   support

       0       0.94      0.99      0.97      2368
       1       0.94      0.63      0.76       382

avg / total       0.94      0.94      0.94      2750

AUC =  0.812277796448
```



Random Forest Classifier



Model	AUC	Precision	Accuracy	recall	F1
GB	0.84	0.84	0.94	0.69	0.76
r.f.	0.81	0.94	0.94	0.63	0.76
Bayes	0.67	0.48	0.86	0.41	0.44
Logistic	0.58	0.57	0.87	0.18	0.28
SVC	0.55	0.49	0.87	0.12	0.2

Models performance matrix

By looking at the above results we can conclude that Gradient Boosting Classifier is best model for our dataset as it has highest AUC and recall also F1 score and precision is good compare to others.

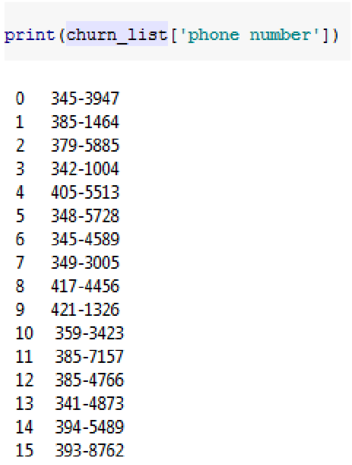
Predict the result

Once we have selected the best model, now we need to take the trained model to predict the churn for unseen data. In our case we have some unseen data in check.csv file.

For prediction of churn we need to read the data and transform the data the way we did during the training phase. But here we don't need split the data.

Then we can use predict function to predict the results. Finally we can export the details of churner into different file, which will be further used by managers to take business decisions and action.

Eg: Following is the example of list which we get after prediction.



```
print(churn_list['phone number'])
```

```
0    345-3947
1    385-1464
2    379-5885
3    342-1004
4    405-5513
5    348-5728
6    345-4589
7    349-3005
8    417-4456
9    421-1326
10    359-3423
11    385-7157
12    385-4766
13    341-4873
14    394-5489
15    393-8762
```

Fig: List of telephone number who may churn.

B) Appendix

When to use which type of models: Tips and Advice.

There is no fixed approach which can be used for churn modeling. As type of model and its parameters all depend on the Features and relationship of the data.

Mostly used models are

Random Forest Classifier:

This model is best in most of the cases as it takes care of model overfitting which make the model bias to trained condition. Also it can easily balance the features by assigning weight to the branches. Random forest runtimes are quite fast, and they are able to deal with imbalanced and missing data.

Generally use when:

1) Have missing data
2) Training data set size is too large.
3) Have imbalance data set
4) Overfitting is major concerned.

Logistic Regression

Logistic regression is used to predict the odds of being a case based on the values of the independent variables (predictors). The odds are defined as the probability that a particular outcome is a case divided by the probability that it is a non-case. This model is best when we can see log graph like pattern in relationship among data.

Gradient Boosting Classifier

This is also quite efficient as random forest and takes care of imbalanced data structure. When the complexity of relationship increased Gradient Boosting Classifier works much better than others.

Naive Bayes Classifier

This is a simple classifier and works when the complexity is low with simple dataset. This is suitable for small sized dataset. This model do have issue with overfitting so not much used for real time solutions but all depends on type and structure of the data.

Conclusion

In this document we have seen many key things. Firstly we understood about Forrester customer analytics model. Then we understood what is churn model and its importance. We also saw approach we should use during modeling and different models available. We touched the concept of model evaluation and seen all the concepts using practical example.

Reference

1. http://scikit-learn.org/stable/auto_examples/model_selection/plot_roc.html#sphx-glr-auto-examples-model-selection-plot-roc-py
2. http://scikit-learn.org/stable/modules/generated/sklearn.metrics.roc_auc_score.html#sklearn.metrics.roc_auc_score
3. https://www.analyticsvidhya.com/blog/2016/02/7-important-model-evaluation-error-metrics/
4. https://en.wikipedia.org/wiki/Random_forest
5. https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm
6. http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html
7. https://www.stat.berkeley.edu/~breiman/randomforest2001.pdf
8. https://en.wikipedia.org/wiki/Gradient_boosting
9. http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html
10. https://statweb.stanford.edu/~jhf/ftp/trebst.pdf
11. https://en.wikipedia.org/wiki/Support_vector_machine
12. https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/
13. http://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html
14. https://en.wikipedia.org/wiki/Naive_Bayes_classifier
15. http://scikit-learn.org/stable/modules/naive_bayes.html
16. https://en.wikipedia.org/wiki/Logistic_regression
17. http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html