Churn Prediction

Machine Learning

Customer Churn

- Depending on the business model, the "moment of churn" occurs when someone cancels a subscription, fails to renew a service agreement, or closes their account.
- Customer churn is a critical metric. It impedes companies growth.
- Customer retention, on the other hand, is generally more cost-effective as you've already earned the trust and loyalty of existing customers.

Sectors

- Banks
- Telephone service companies
- Internet service providers,
- Pay TV companies,
- Insurance firms,
- Alarm monitoring services,

Churn Rate

- Churn rate is the rate at which you lose customers or subscribers
- Since attracting new customers costs 6-7 times more than retaining existing ones, a lower rate of churn is generally more profitable

Key Factors

- Experience
- Expectations
- Investment
- Value provided

Objective

- Increase customer retention by building predictive model to push churn rate down closer to 0%.
- Explore the data and find correlations
- Identify the features impacting Churn
- Build an machine learning model to obtain highest level of accuracy

Dataset Overview

- This data set consists of Customer left within last month.
- Service that each customer has signed up for.
- Customer account information like
  - Payment method
  - Billing type
  - Demographic info
  - Gender
  - Age & Dependents

[Source: https://www.kaggle.com/blastchar/telco-customer-churn](https://www.kaggle.com/blastchar/telco-customer-churn)

Tools

- Python -3
- Jupyter Notebook
- ML Algorithm

Technical Approach

Data Wrangling

Data Visualization

Inferential Statistics

Predictive Models (ML)

Import Required Libraries

- Numpy
- Pandas
- Matplotlib
- Scipy
- Sklearn
- Seaborn

Data Wrangling

- Load & Read the Data
- Shape of the Dataset
- Summary for Data frame
- Identify the missing values
- Clean up data

Dataset Analysis

Fortunately this is not a 'messy' data set

The dataset has no missing values

We have more than 7000 rows and 21 attributes (columns)

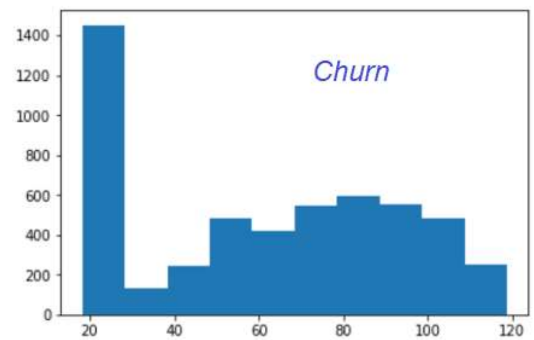
Some data that should be categorical are saved as number. Let's fix this

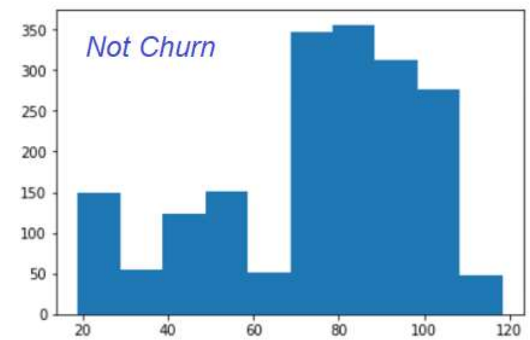
If there are any missing values then fill the missing data

Convert Total Charge to numeric

There are some features that contains ambiguous information's, for example, Online Security contains 3,differents labels, but the correct labels are Yes or No

Visualization





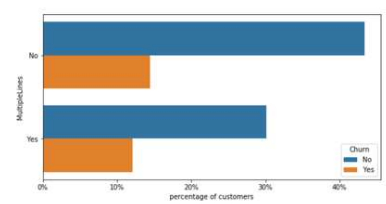
Histogram for total charges

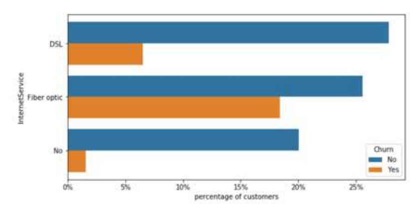
Fig:1 → Churn Customers

Fig:1 → Not Churn Customers

Bar Plot

- Customers with multiple lines have a slightly higher churn rate
- Customers without internet have a very low churn rate
- Customers with fiber are more probable to churn than those with DSL connection

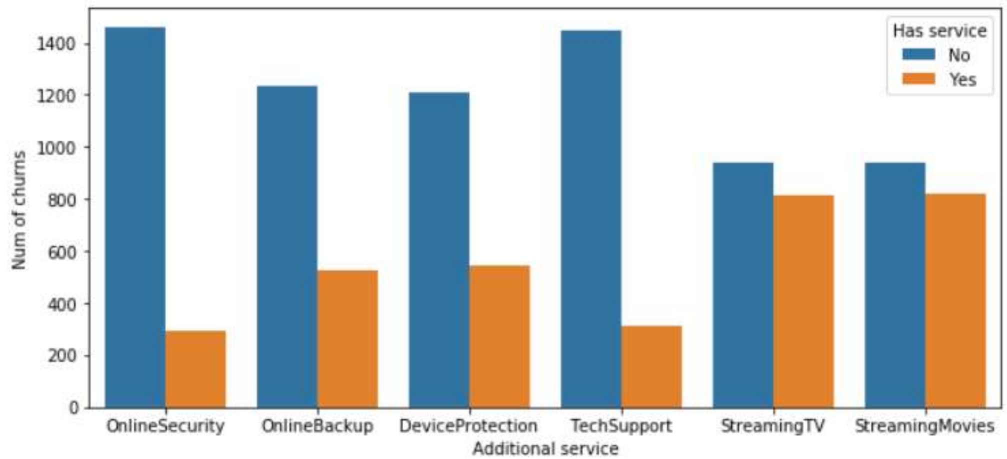




Online Services

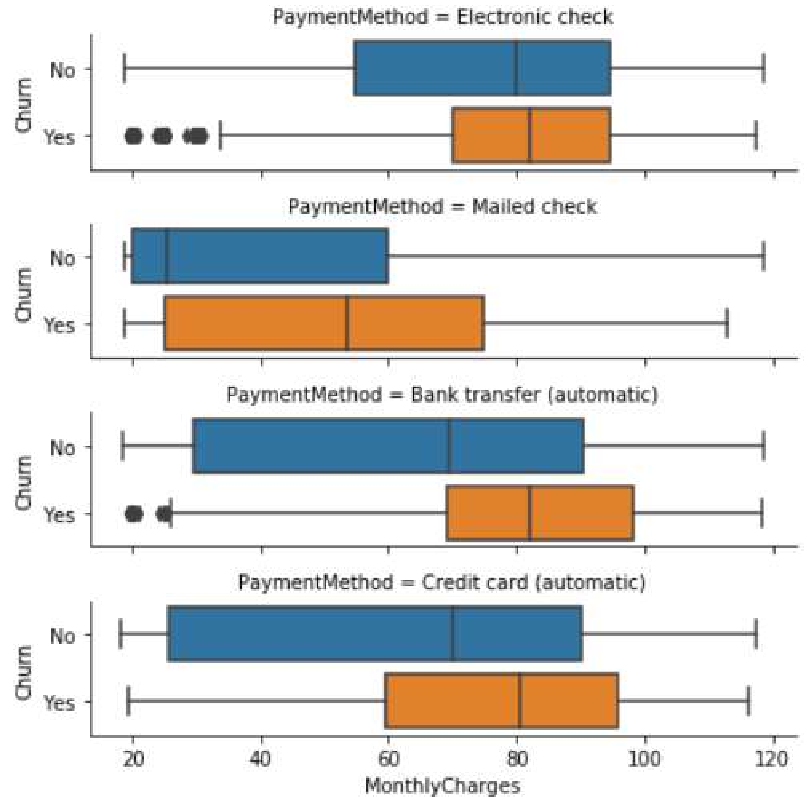
- Customers with online Security and tech support has a very low churn rate

- Customers with Streaming services are more likely to churn



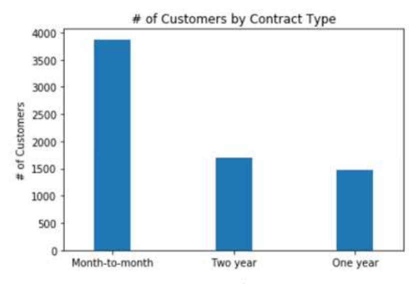
Payment Method

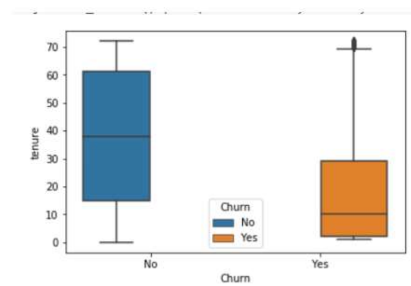
- Customers with automatic payments has high churn rate



Contract Type

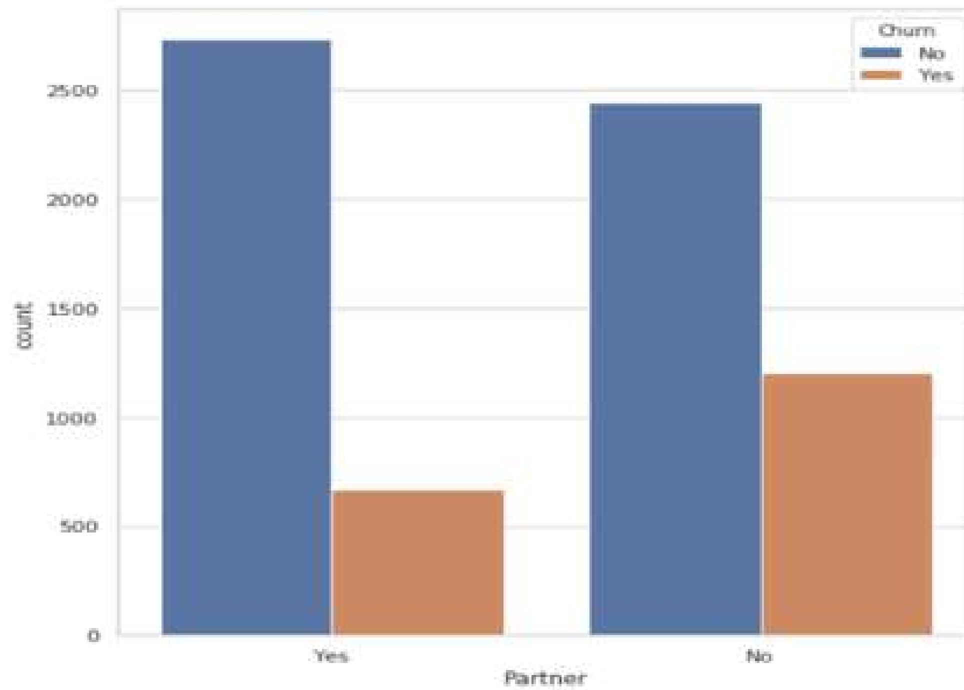
- As we can see from this graph most of the customers are in the month to month contract.
- While there are equal number of customers in the 1 year and 2-year contracts.
- Churn rate is high for the customers who has less tenure/contract with Telco company.





Statistics

- Customer with partners has a very low churn rate when compared with customer without partners.
- There is no behavior difference between women and men.



Statistical summary

	count	mean	std	min	25%	50%	75%	max
gender	7043.0	0.504756	0.500013	0.00	0.00	1.000	1.0000	1.00
SeniorCitizen	7043.0	0.162147	0.368612	0.00	0.00	0.000	0.0000	1.00
Partner	7043.0	0.483033	0.499748	0.00	0.00	0.000	1.0000	1.00
Dependents	7043.0	0.299588	0.458110	0.00	0.00	0.000	1.0000	1.00
tenure	7043.0	32.371149	24.559481	0.00	9.00	29.000	55.0000	72.00
PhoneService	7043.0	0.903166	0.295752	0.00	1.00	1.000	1.0000	1.00
PaperlessBilling	7043.0	0.592219	0.491457	0.00	0.00	1.000	1.0000	1.00
MonthlyCharges	7043.0	64.761692	30.090047	18.25	35.50	70.350	89.8500	118.75
TotalCharges	7032.0	2283.300441	2266.771362	18.80	401.45	1397.475	3794.7375	8684.80
Churn	7043.0	0.265370	0.441561	0.00	0.00	0.000	1.0000	1.00
customerID_0002-ORFBO	7043.0	0.000142	0.011916	0.00	0.00	0.000	0.0000	1.00
customerID_0003-MKNFE	7043.0	0.000142	0.011916	0.00	0.00	0.000	0.0000	1.00
customerID_0004-TLHLJ	7043.0	0.000142	0.011916	0.00	0.00	0.000	0.0000	1.00

Machine Learning Algorithms

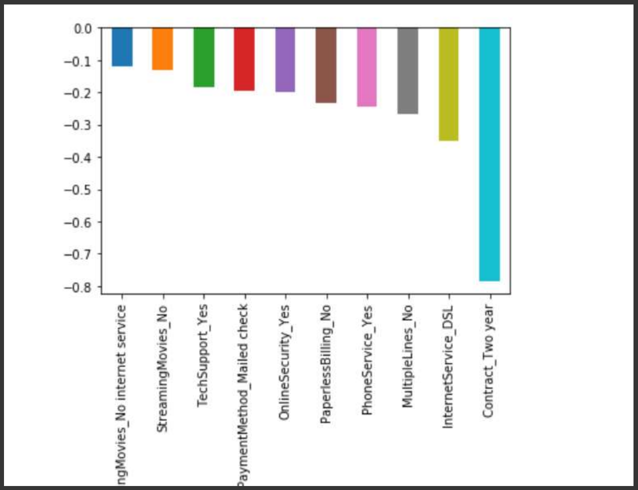
- Logistic Regression
- Random Forest
- SVM
- XG Boost
- Catboost

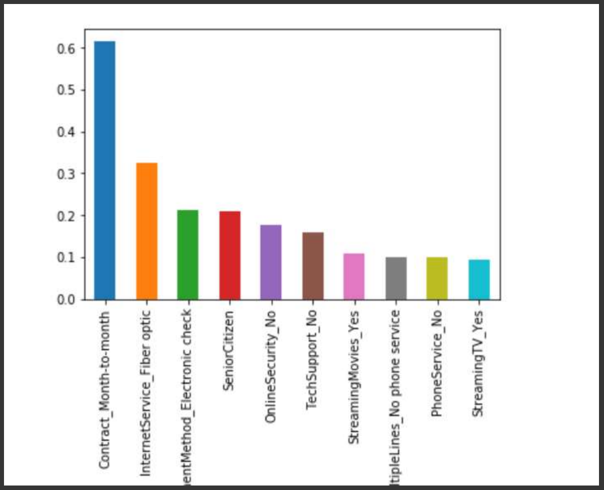
Logistic Regression

```
gorithm : LogisticRegression(C=1.0, class_w
          intercept_scaling=1, max_iter=100,
          penalty='l2', random_state=None, sc
          verbose=0, warm_start=False)

lassification report :
             precision    recall  f1-score

         0       0.84      0.90      0.87
         1       0.65      0.54      0.59

 micro avg       0.80      0.80      0.80
 macro avg       0.75      0.72      0.73
ighted avg       0.79      0.80      0.80

curacy   Score :  0.8020477815699659
ea under curve :  0.7183103048553003
```

Weights Of Features

- From the plots we can observer that some variable have a negative correlation and some has positive correlation
- Customers with Month-Month contract, no online security services and tech support seems to be positively correlated with the churn.
- Some services like streaming services, multiple lines, two-year contract seems to be negatively corelated with churn.
- We can see that some variables have a negative relation to our predicted variable (Churn), while some have positive relation. Negative relation means that likeliness of churn decreases with that variable. Let us summarize some of the interesting features below:

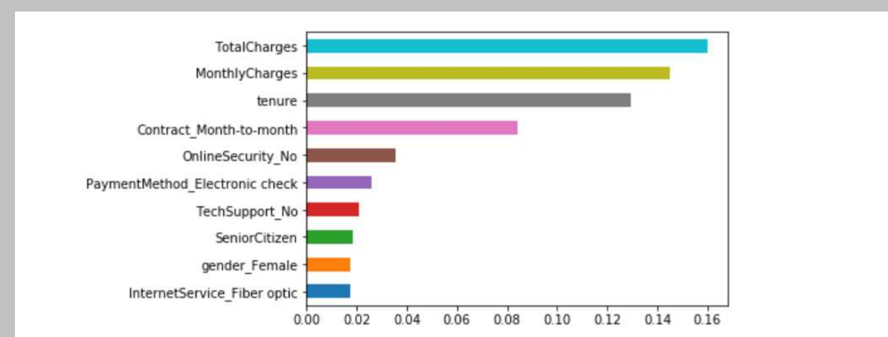




Random Forest

- From the below plot we can observe that monthly contract, tenure and tenure are the most import features to predict churn.
- Algorithm, classification report shown and the accuracy is slightly less than the logistic regression

```
Algorithm : RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
            max_depth=None, max_features='auto', max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=1, min_samples_split=2,
            min_weight_fraction_leaf=0.0, n_estimators=10, n_jobs=None,
            oob_score=False, random_state=0, verbose=0, warm_start=False)

 Classification report :
               precision    recall  f1-score   support

           0       0.82      0.90      0.85      1294
           1       0.60      0.44      0.51       464

   micro avg       0.78      0.78      0.78      1758
   macro avg       0.71      0.67      0.68      1758
weighted avg       0.76      0.78      0.76      1758

Accuracy   Score :  0.7758816837315131
Area under curve :  0.6673589644513137

Accuracy: 0.7758816837315131
```



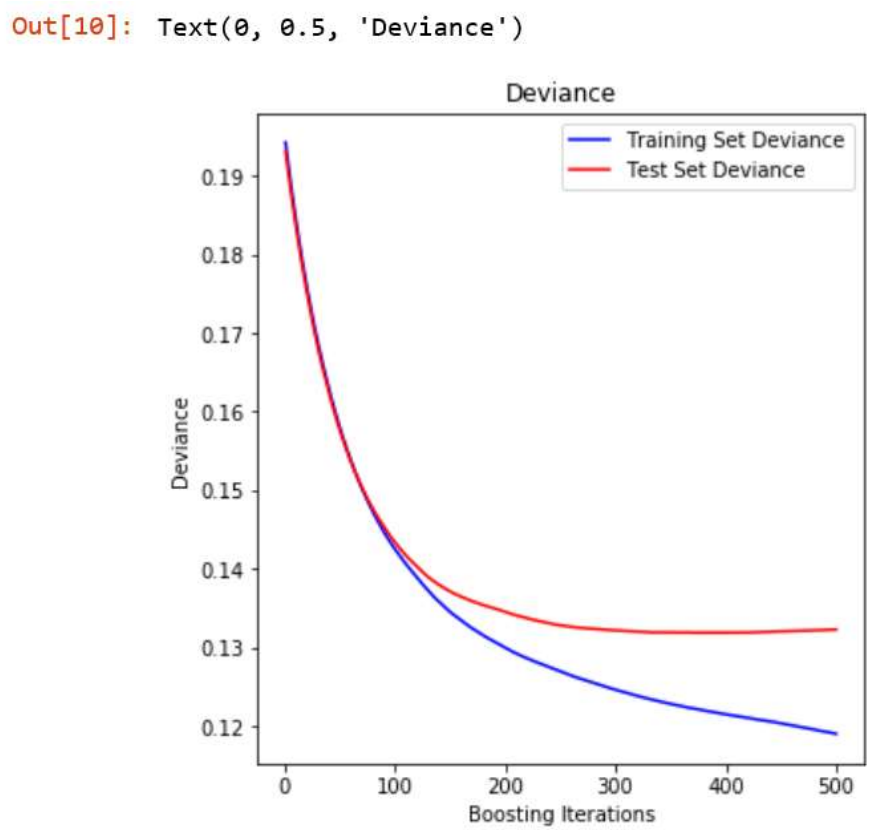
SVM

- With SVM the accuracy is 0.79 and AUC is 0.66
- Algorithm, classification report shown in fig, and the accuracy is in line with Random forest

```
Algorithm : SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto_deprecated',
  kernel='linear', max_iter=-1, probability=False, random_state=None,
  shrinking=True, tol=0.001, verbose=False)
accuracy score 0.7901023890784983

 Classification report :
               precision    recall  f1-score   support

           0       0.81      0.93      0.87      1294
           1       0.67      0.41      0.51       464

   micro avg       0.79      0.79      0.79      1758
   macro avg       0.74      0.67      0.69      1758
weighted avg       0.78      0.79      0.77      1758

Area under curve :  0.6687246842189415
```

Gradient Boosting Regressor

- Gradient Boosting model with least squares loss and 500 regression trees of depth 4.
- MSE: 0.1323



XG Boost

- XG Boost the accuracy on test data to almost 80%, which is in line with logistic regression and SVM.

```
Algorithm : XGBClassifier(base_score=0.5, booster='gbtree', colsample_bylevel=1,
       colsample_bytree=1, gamma=0, learning_rate=0.1, max_delta_step=0,
       max_depth=3, min_child_weight=1, missing=None, n_estimators=100,
       n_jobs=1, nthread=None, objective='binary:logistic', random_state=0,
       reg_alpha=0, reg_lambda=1, scale_pos_weight=1, seed=None,
       silent=True, subsample=1)
accuracy score 0.8060295790671217

 Classification report :
               precision    recall  f1-score   support

           0       0.84      0.91      0.87      1294
           1       0.68      0.51      0.58       464

   micro avg       0.81      0.81      0.81      1758
   macro avg       0.76      0.71      0.73      1758
weighted avg       0.80      0.81      0.80      1758

Area under curve :  0.7113384719927517
```

Cat Boost Regressor

- Cat Boost the accuracy on test data to almost 80%, which is in line with logistic regression, XG boost and SVM.

```
0:      learn: 0.3982526        total: 12.5ms   remaining: 12.5ms
1:      learn: 0.3901868        total: 23.7ms   remaining: 0us
Algorithm : <catboost.core.CatBoostRegressor object at 0x000002221777C240>
accuracy score 0.8060295790671217

 Classification report :
               precision    recall  f1-score   support

           0       0.84      0.91      0.87      1294
           1       0.68      0.51      0.58       464

   micro avg       0.81      0.81      0.81      1758
   macro avg       0.76      0.71      0.73      1758
weighted avg       0.80      0.81      0.80      1758

Area under curve :  0.7113384719927517
```

Comparison matrix

MODEL	ACCURACY	ROC
Logistic Regression	0.802	0.71
Random Forest	0.77	0.66
SVM	0.79	
XG Boost	0.8	0.71
Cat boost	0.8	0.71

Thank You