

IBM Machine Learning Professional Certificate

**- Supervised Learning-
(Classification)**

Choe C.S

October 2021

Main Objective

Objective

- This report aims to analyze and predict Customer Churn in Telcom Industry using Supervised Machine Learning algorithm.

Data Set

- The data set used for this analysis is Telcom Customer Churn from [Kaggle.com](https://www.kaggle.com).

Steps Involved

- 1) Perform Data preprocessing – Data Formatting convert column names for better understanding of each column.
- 2) EDA – To gain better understanding of each feature variables and its affects on the target variable.
- 3) Perform Feature Engineering to remove unwanted columns, transform categorical data into numeric format and perform Feature Scaling.
- 4) Perform Machine Learning on the data set using Classification algorithm
- (Logistic Regression, KNN, Decision Tree and Random Forest classification).

Import Data

- The data set was import from [Kaggle.com](https://www.kaggle.com).
- The table at the bottom shows the top 5 rows for the data set

	customerID	tenure	PhoneService	Contract	PaperlessBilling	PaymentMethod	MonthlyCharges	TotalCharges	Churn	MultipleLines	...	OnlineSecurity	Onlin
0	7590-VHVEG	1	No	Month-to-month	Yes	Electronic check	29.85	29.85	No	No phone service	...	No	
1	5575-GNVDE	34	Yes	One year	No	Mailed check	56.95	1889.5	No	No	...	Yes	
2	3668-QPYBK	2	Yes	Month-to-month	Yes	Mailed check	53.85	108.15	Yes	No	...	Yes	
3	7795-CFOCW	45	No	One year	No	Bank transfer (automatic)	42.30	1840.75	No	No phone service	...	Yes	
4	9237-HQITU	2	Yes	Month-to-month	Yes	Electronic check	70.70	151.65	Yes	No	...	No	

Data Preprocessing

- There are a total of 7042 rows with 21 column of data .
- There is a total of 1 “float64”, 2 “int64” and 18 “object” columns.
- Data formatting will be required on “**TargetChargers**” column for data type conversion.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 7042 entries, 0 to 7041
Data columns (total 21 columns):
#   Column                Non-Null Count  Dtype
---  -
0   customerID            7042 non-null   object
1   tenure                7042 non-null   int64
2   PhoneService          7042 non-null   object
3   Contract              7042 non-null   object
4   PaperlessBilling      7042 non-null   object
5   PaymentMethod         7042 non-null   object
6   MonthlyCharges        7042 non-null   float64
7   TotalCharges          7042 non-null   object
8   Churn                 7042 non-null   object
9   MultipleLines         7042 non-null   object
10  InternetService       7042 non-null   object
11  OnlineSecurity        7042 non-null   object
12  OnlineBackup          7042 non-null   object
13  DeviceProtection      7042 non-null   object
14  TechSupport          7042 non-null   object
15  StreamingTV           7042 non-null   object
16  StreamingMovies       7042 non-null   object
17  gender                7042 non-null   object
18  SeniorCitizen         7042 non-null   int64
19  Partner               7042 non-null   object
20  Dependents            7042 non-null   object
dtypes: float64(1), int64(2), object(18)
```

Data Preprocessing – Data Formatting

- The column “**TotalChargers**” is in “object” data type and not suitable for analysis.
- As chargers must be in numeric type for analysis, data formatting will be performed on this column to convert string to “float64”.

customerID	object
tenure	int64
PhoneService	object
Contract	object
PaperlessBilling	object
PaymentMethod	object
MonthlyCharges	float64
<u>TotalCharges</u>	<u>object</u>
Churn	object
MultipleLines	object
InternetService	object
OnlineSecurity	object
OnlineBackup	object
DeviceProtection	object
TechSupport	object
StreamingTV	object
StreamingMovies	object
gender	object
SeniorCitizen	int64
Partner	object
Dependents	object
dtype:	object



customerID	object
tenure	int64
PhoneService	object
Contract	object
PaperlessBilling	object
PaymentMethod	object
MonthlyCharges	float64
<u>TotalCharges</u>	<u>float64</u>
Churn	object
MultipleLines	object
InternetService	object
OnlineSecurity	object
OnlineBackup	object
DeviceProtection	object
TechSupport	object
StreamingTV	object
StreamingMovies	object
gender	object
SeniorCitizen	int64
Partner	object
Dependents	object
dtype:	object

Data Preprocessing – Handling Missing Data

- The data set has 11 missing data for column **“TotalChargers”**.
- Due to low amount of missing data the decision is to remove the missing rows.

customerID	0
tenure	0
PhoneService	0
Contract	0
PaperlessBilling	0
PaymentMethod	0
MonthlyCharges	0
<u>TotalCharges</u>	<u>11</u>
Churn	0
MultipleLines	0
InternetService	0
OnlineSecurity	0
OnlineBackup	0
DeviceProtection	0
TechSupport	0
StreamingTV	0
StreamingMovies	0
gender	0
SeniorCitizen	0
Partner	0
Dependents	0
dtype:	int64

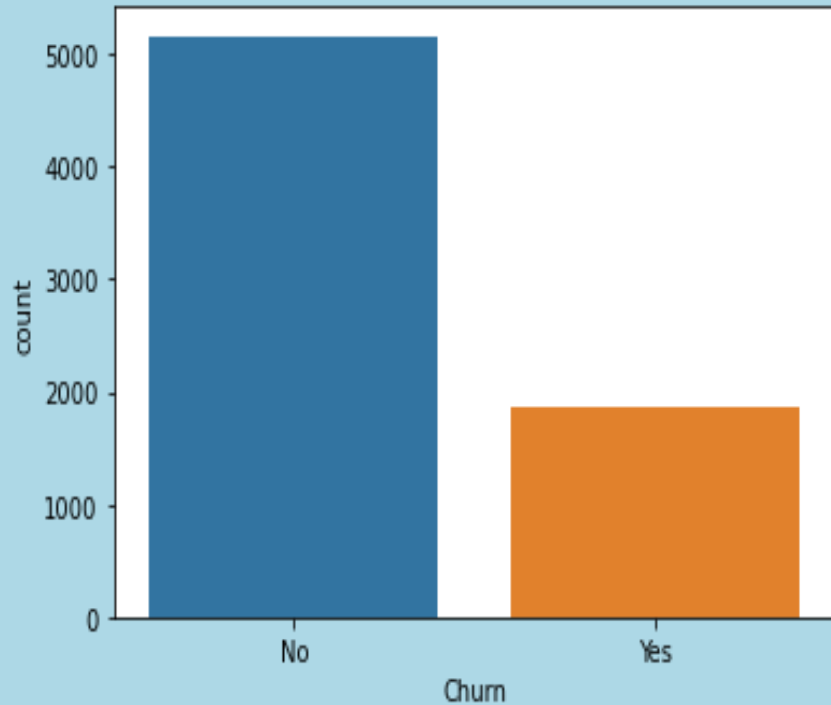
customerID	0
tenure	0
PhoneService	0
Contract	0
PaperlessBilling	0
PaymentMethod	0
MonthlyCharges	0
<u>TotalCharges</u>	<u>0</u>
Churn	0
MultipleLines	0
InternetService	0
OnlineSecurity	0
OnlineBackup	0
DeviceProtection	0
TechSupport	0
StreamingTV	0
StreamingMovies	0
gender	0
SeniorCitizen	0
Partner	0
Dependents	0
dtype:	int64

EDA - Descriptive analysis

- The table on the right shows the descriptive analysis summary for data set.
- The mean value for the tenure is 32.4170, MonthlyChargers is 64.792398, TotalChargers is 2282.651714 and SeniorCitizen is 0.162424.
- The minimum value for the tenure is 1, MonthlyChargers is 18.25, TotalChargers is 18.8 and SeniorCitizen is 0.
- The maximum value for the tenure is 72, MonthlyChargers is 118.75, TotalChargers is 8684.8 and SeniorCitizen is 1.

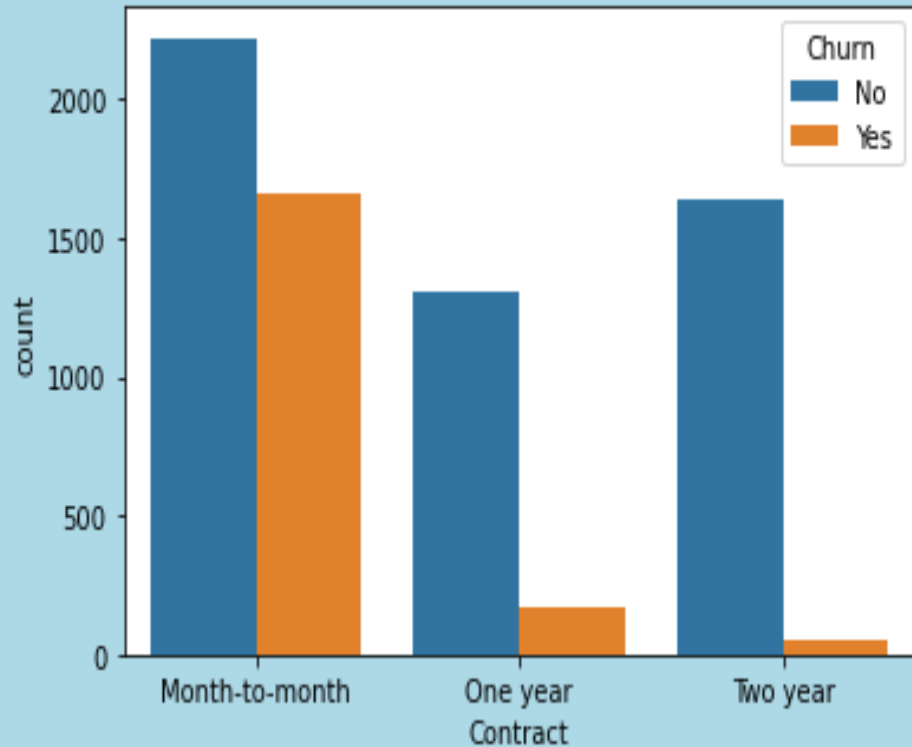
	tenure	MonthlyChargers	TotalChargers	SeniorCitizen
count	7031.000000	7031.000000	7031.000000	7031.000000
mean	32.417010	64.792398	2282.651714	0.162424
std	24.543738	30.084168	2266.279660	0.368865
min	1.000000	18.250000	18.800000	0.000000
25%	9.000000	35.575000	401.400000	0.000000
50%	29.000000	70.350000	1397.300000	0.000000
75%	55.000000	89.850000	3793.050000	0.000000
max	72.000000	118.750000	8684.800000	1.000000

EDA – Visual exploration



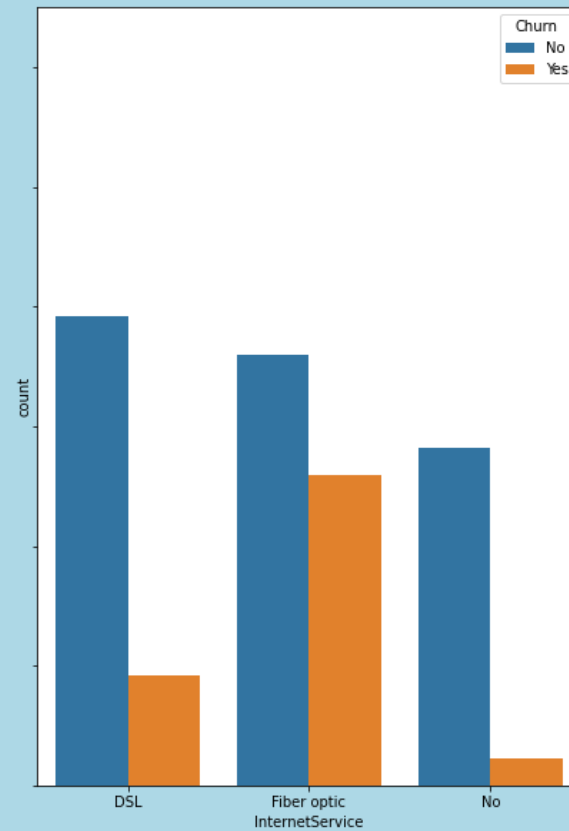
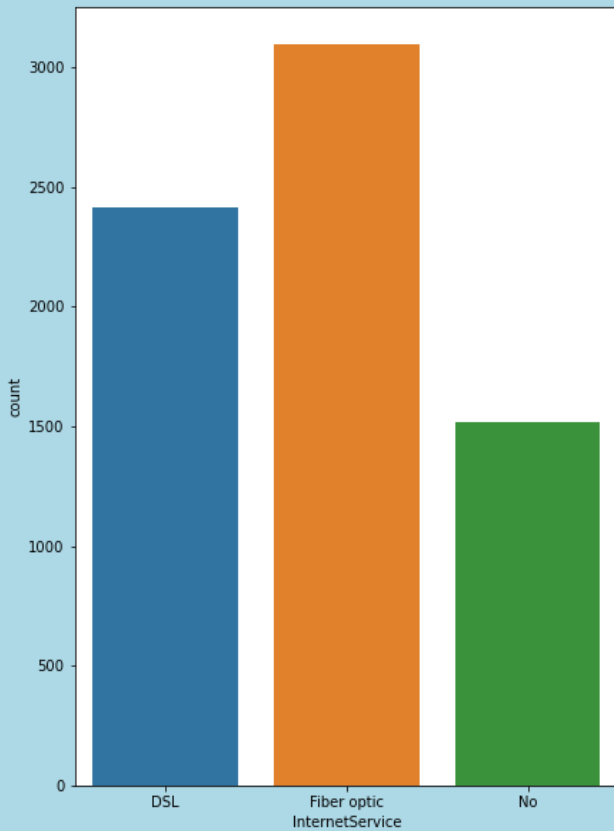
- The bar plot on the left shows total number of customer who churn and did not churn.
- From this plot, there are more customer who did not churn compared to the ones who did.
- An estimated value of 5000 customers did not churn while 2000 customers churn.

EDA – Visual exploration (pairplot with hue)



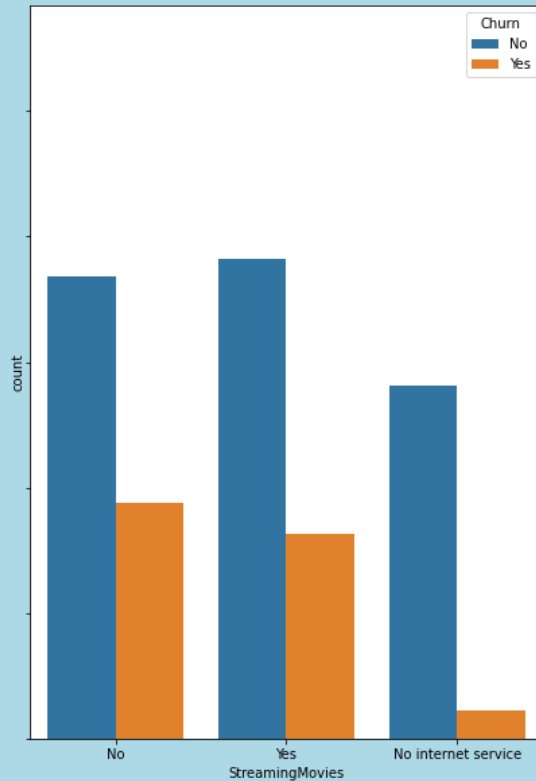
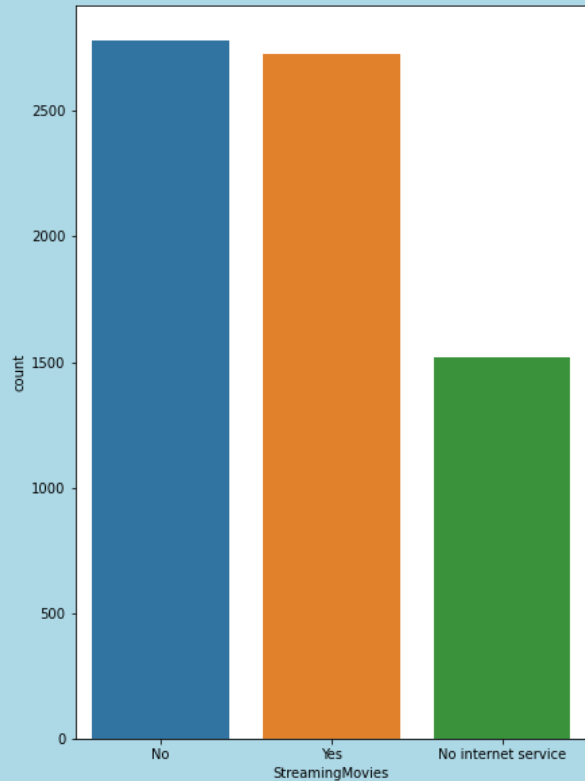
- The bar plot on the left shows total number of customer according to contract duration.
- Form this plot, the highest number of customers falls under **Month-to-month** contract.
- The least amount opt for **One year** Contract.
- Although there are many customers who opt for **Month-to-month** contract, there is a high record of customer churn for this contract at more than 2000.

EDA – Visual exploration (Categorical data)



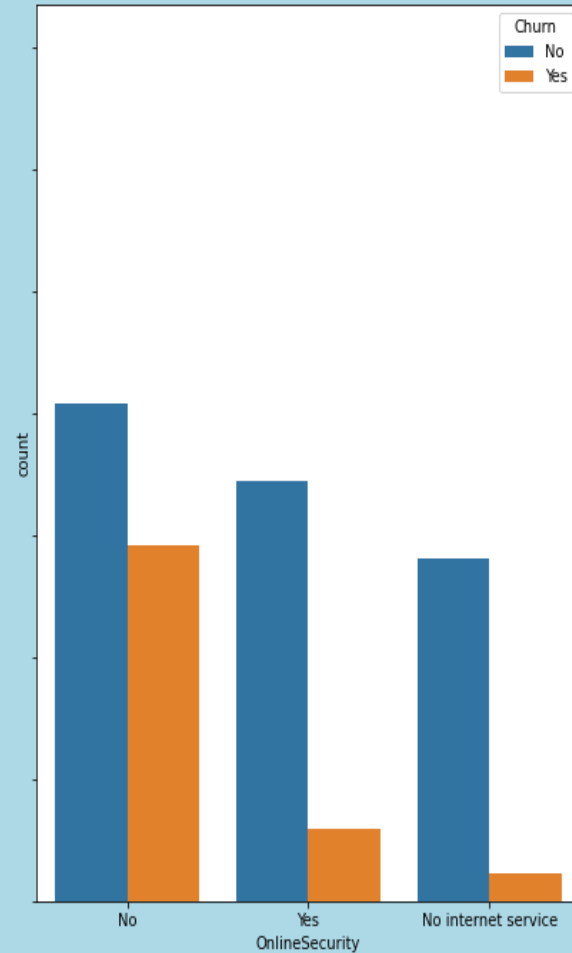
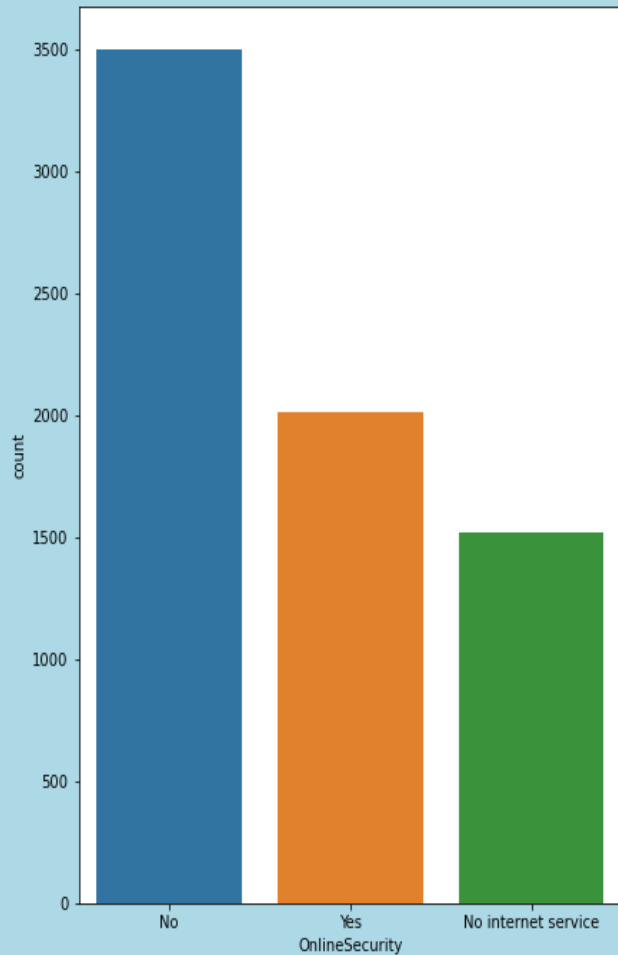
- The bar plot on the right shows the Internet service used by customers.
- The first plot shows **DSL** service at around 2400, **Fiber optic** with the highest amount at roughly 3000, and 1500 for customer with No Internet Service.
- The second graph shows the highest amount of churn coming from customer with **Fiber Optic** Internet Service while the highest amount of customers who did not churn are from **DSL**.
- The lowest amount of churn coming from Customers with **No Internet Service** while the highest are the ones with **Fiber optic**.

EDA – Visual exploration (Categorical data)



- The bar plot on the right records the amount of **movie streaming** activities.
- The first plot shows the highest amount of customers that don't stream movies followed by customers who do at more than 2500 and the lowest for customer without internet service at 1500.
- From the second plot, the highest amount of customer who did not churn are the ones who stream movies at around 2000 while the lowest are the ones without internet service at around 1500.

EDA – Visual exploration (Categorical data)



- The bar plot on the right records the data for Online Security.
- The first plot shows most of the customers not equipped with Online Security with an amount of 3500, while customers with online security at 2000.
- From the second plot, the highest amount of customer who did not churn are the ones without internet service while the lowest are the ones without internet service.

Feature Engineering – Remove ID column

Drop unrequired column

- Customer ID is removed from data set before machine learning process takes place.

Before

	customerID	tenure	PhoneService	Contract	PaperlessBilling	PaymentMethod	MonthlyCharges	TotalCharges	Churn	MultipleLines	...	OnlineSecurity	Onlir
0	7590-VHVEG	1	No	Month-to-month	Yes	Electronic check	29.85	29.85	No	No phone service	...	No	
1	5575-GNVDE	34	Yes	One year	No	Mailed check	56.95	1889.5	No	No	...	Yes	
2	3668-QPYBK	2	Yes	Month-to-month	Yes	Mailed check	53.85	108.15	Yes	No	...	Yes	
3	7795-CFOCW	45	No	One year	No	Bank transfer (automatic)	42.30	1840.75	No	No phone service	...	Yes	
4	9237-HQITU	2	Yes	Month-to-month	Yes	Electronic check	70.70	151.65	Yes	No	...	No	

Removing ID

After

	Private	Apps	Accept	Enroll	Top 10%	Top 25%	Full-time-Undergrad	Part-time-Undergrad	Outstate	Boarding	Books	Personal	PhD	Terminal	S.F.Rat
0	Yes	-0.011583	0.006320	0.427055	0.014037	-0.191827	0.323264	0.361674	-0.746356	-0.964905	-0.601556	1.259401	-0.163028	-0.115729	1.0137
1	Yes	0.245031	0.457067	0.067493	-0.535636	-1.353911	0.252545	0.875509	0.457496	1.909208	1.286817	0.469963	-2.675646	-3.378176	-0.4777
2	Yes	-0.151913	-0.111012	-0.374601	-0.053802	-0.292878	-0.674259	-0.685997	0.201305	-0.554317	-1.036697	-0.050896	-1.204845	-0.931341	-0.3007
3	Yes	-1.298516	-1.267855	-1.313967	1.500947	1.677612	-1.363987	-0.963863	0.626633	0.996791	-0.601556	-0.640670	1.185206	1.175657	-1.6152
4	Yes	-2.014530	-2.145621	-2.262878	-0.535636	-0.596031	-2.060724	0.660925	-0.716508	-0.216723	1.525504	0.469963	0.204672	-0.523535	-0.5535

Feature Engineering – Feature Scaling

Standard Scaler

- Feature Scaling was performed on the numeric feature variables to normalize the ranges of the data.
- Standard Scaler from Scikit Learn preprocessing was used for this process.
- The variables that were scaled are “tenure”, “MonthlyCharges” and “TotalCharges”.

	tenure	MonthlyCharges	TotalCharges
0	1	29.85	29.85
1	34	56.95	1889.5
2	2	53.85	108.15
3	45	42.30	1840.75
4	2	70.70	151.65
...
7037	72	21.15	1419.4
7038	24	84.80	1990.5
7039	72	103.20	7362.9
7040	11	29.60	346.45
7041	4	74.40	306.6

Before Standard Scaler

Standard Scaler



	tenure	MonthlyCharges	TotalCharges
0	-1.280133	-1.161571	-0.994124
1	0.064501	-0.260700	-0.173491
2	-1.239386	-0.363752	-0.959571
3	0.512713	-0.747702	-0.195004
4	-1.239386	0.196383	-0.940375
...
7026	1.612868	-1.450780	-0.380938
7027	-0.342964	0.665101	-0.128922
7028	1.612868	1.276762	2.241828
7029	-0.872668	-1.169881	-0.854413
7030	-1.157893	0.319380	-0.871998

After Standard Scaler

Feature Engineering – Feature Encoding

- As there are many categorical variables present in data set, it is important to perform feature encoder to convert categorical data type to numeric data type for Machine Learning application.
- Each categorical variables were split into Binary variables, Ordinal variables and Numeric variables.
- LabelBinarizer(), LabelEncoder() from Scikit-Learn was used to convert the categorical variables' data type to numeric data type.

Feature Engineering – Feature Encoding

Categorical Variables

- 16 Categorical variables

PhoneService	2
Contract	3
PaperlessBilling	2
PaymentMethod	4
Churn	2
MultipleLines	3
InternetService	3
OnlineSecurity	3
OnlineBackup	3
DeviceProtection	3
TechSupport	3
StreamingTV	3
StreamingMovies	3
gender	2
Partner	2
Dependents	2

Binary Variables

- Variables with only 2 unique values
- 6 Binary variables

Binary_varaibles	
0	PhoneService
1	PaperlessBilling
2	Churn
3	gender
4	Partner
5	Dependents

Numeric Variables

- Categorical Variables that are not ordered according to category
- 9 Numeric Variables

Numeric_values	
0	DeviceProtection
1	StreamingTV
2	OnlineSecurity
3	PaymentMethod
4	MultipleLines
5	InternetService
6	TechSupport
7	OnlineBackup
8	StreamingMovies

Ordinal Variable

- Categorical Variables that are in ordered of category
- 1 Numeric Variables

Ordinal_variables	
0	Contract

Feature Engineering – (Feature Encoding)

	customerID	tenure	PhoneService	Contract	PaperlessBilling	PaymentMethod	MonthlyCharges	TotalCharges	Churn	MultipleLines	...	OnlineSecurity	Onlir
0	7590-VHVEG	1	No	Month-to-month	Yes	Electronic check	29.85	29.85	No	No phone service	...	No	
1	5575-GNVDE	34	Yes	One year	No	Mailed check	56.95	1889.5	No	No	...	Yes	
2	3668-QPYBK	2	Yes	Month-to-month	Yes	Mailed check	53.85	108.15	Yes	No	...	Yes	
3	7795-CFOCW	45	No	One year	No	Bank transfer (automatic)	42.30	1840.75	No	No phone service	...	Yes	
4	9237-HQITU	2	Yes	Month-to-month	Yes	Electronic check	70.70	151.65	Yes	No	...	No	

Before Feature Encoding

↓ Feature Encoding



	tenure	PhoneService	Contract	PaperlessBilling	MonthlyCharges	TotalCharges	Churn	gender	SeniorCitizen	Partner	...	MultipleLines_No phone service	MultipleLine:
0	-1.280133	0	0	1	-1.161571	-0.994124	0	0	0	1	...	1.0	
1	0.064501	1	1	0	-0.260700	-0.173491	0	1	0	0	...	0.0	
2	-1.239386	1	0	1	-0.363752	-0.959571	1	1	0	0	...	0.0	
3	0.512713	0	1	0	-0.747702	-0.195004	0	1	0	0	...	1.0	
4	-1.239386	1	0	1	0.196383	-0.940375	1	0	0	0	...	0.0	

After Feature Encoding

All data types are in numeric form after Feature Encoding was performed on the data set

Machine Learning – Splitting of Data


Total number of Churn

- There is a total of 5162 non-churn and 1869 amount of churn. 
- In other words the percentage of non-churn is 73.418% and 26.58% for churn. 
- The data shows uneven amount of distribution for targeted variable.
- Therefore, data set was split using Stratified Shuffle Split from Scikit Learn's model selection for even split.

```
0    5162
1    1869
Name: Churn, dtype: int64
```

```
0    5162
1    1869
Name: Churn, dtype: int64
```

Data split (StratifiedShuffleSplit)

- Data set was split into training (70%) and testing data (30%). 
- Training set = X_train , Y_train (70%).
- Testing set = X_test , Y_test (30%).

```
Shape for X_train is: (4921, 29)
Shape for y_train is: (4921,)
```

```
Shape for X_test is: (2110, 29)
Shape for y_test is: (2110,)
```

Machine Learning – Logistic Regression

Machine Learning

- Machine learning classification algorithms (Logistic Regression, KNN, Decision Tree and Random Forest Classification) were used for churn prediction.
- Each models' performance were evaluated using : Confusion Matrix and Classification Report

Logistic Regression

Decision Matrix

	0	1
0	1374	175
1	244	317

Classification Report

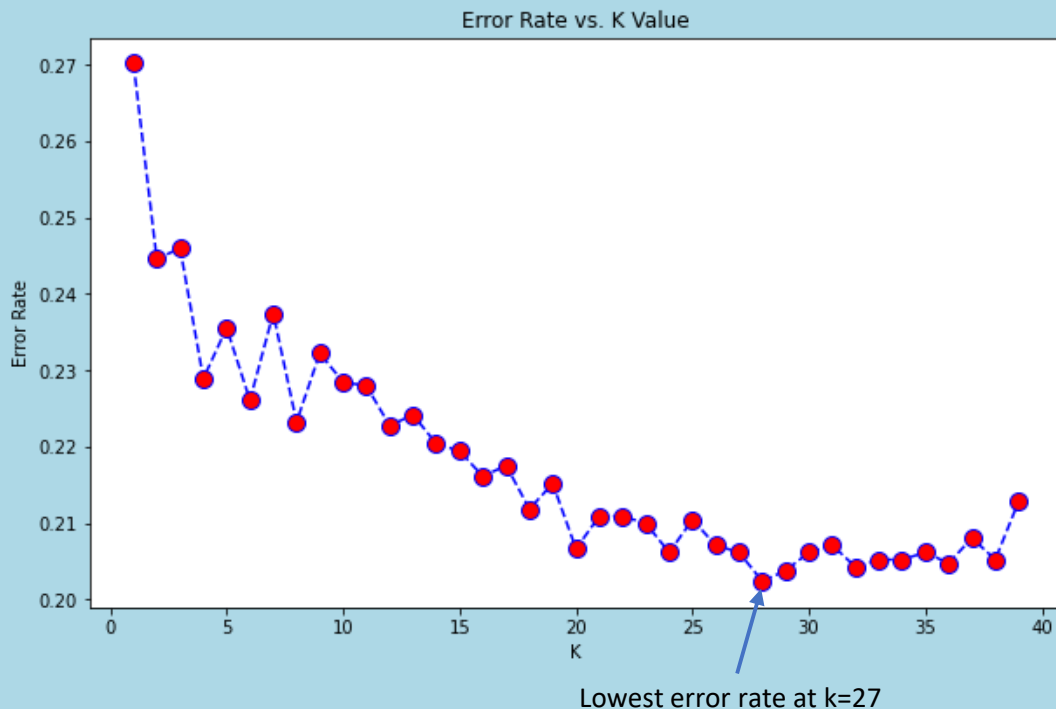
	0	1
precision	0.849197	0.644309
recall	0.887024	0.565062
f1-score	0.867698	0.602089

- The decision matrix for Logistic regression records TN = 1374, FP = 175, FN = 244 and TP = 317
- The Classification Report for Logistic regression records Precision = 0.8495 , Recall = 0.887, f1-score = 0.868 for not churn and Precision = 0.644 , Recall = 0.5651, f1-score = 0.602

Machine Learning - KNN

K-Nearest Neighbor (KNN)

- To select the best number of neighbors an error rate graph was plotted using ranges of neighbor values from $n_neighbors = 1$ to 40.
- From the graph the lowest error rate recorded was $k = 27$



Decision Matrix

	0	1
0	1342	207
1	228	333

Classification Report

	0	1
precision	0.854777	0.616667
recall	0.866365	0.593583
f1-score	0.860532	0.604905

- The decision matrix for Logistic regression records $TN = 1342$, $FP = 207$, $FN = 228$ and $TP = 333$
- The Classification Report for Logistic regression records Precision = 0.855 , Recall = 0.866, f1-score = 0.861 for not churn and Precision = 0.617 , Recall = 0.594, f1-score = 0.605

Machine Learning – Decision Tree

Decision Tree Classification

Decision Matrix

	0	1
0	1263	286
1	303	258

Classification Report

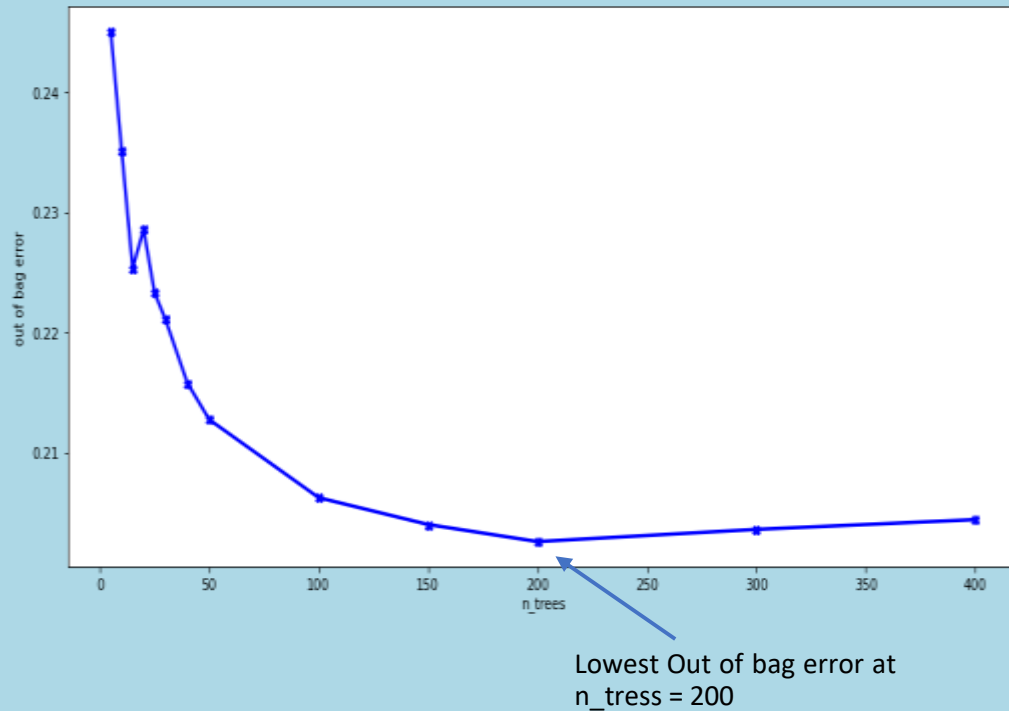
	0	1
precision	0.806513	0.474265
recall	0.815365	0.459893
f1-score	0.810915	0.466968

- The decision matrix for Logistic regression records TN = 1263, FP = 286, FN = 303 and TP = 258
- The Classification Report for Logistic regression records Precision = 0.807 , Recall = 0.815, f1-score = 0.811 for not churn and Precision = 0.474 , Recall = 0.46, f1-score = 0.467

Machine Learning – Random Forest

Random Forest Classification

- To select the best number of trees, out of bag error graph was plotted using number of trees values of `n_estimators` = [5, 10, 15, 20, 25, 30, 40, 50, 100, 150, 200, 300, 400]
- From the graph the Lowest Out of bag error at `n_trees` = 200



Decision Matrix

	0	1
0	1386	163
1	293	268

Classification Report

	0	1
precision	0.825491	0.621810
recall	0.894771	0.477718
f1-score	0.858736	0.540323

- The decision matrix for Logistic regression records TN = 1386, FP = 163, FN = 293 and TP = 268
- The Classification Report for Logistic regression records Precision = 0.855 , Recall = 0.895, f1-score = 0.859 for not churn and Precision = 0.622 , Recall = 0.478, f1-score = 0.543

Overall Results

	accuracy	precision	recall	f1	auc
Logistic Regression	0.801422	0.644309	0.565062	0.602089	0.836583
KNN	0.793839	0.616667	0.593583	0.604905	0.826006
Decision Tree Classification	0.720853	0.474265	0.459893	0.466968	0.638181
Random Forest Classification	0.783886	0.621810	0.477718	0.540323	0.816695

Discussion

- From this analysis, the highest accuracy is Logistic Regression at 0.801 followed by KNN at 0.79, Random Forest Regression at 0.784 and Decision Tree at 0.720.
- The highest precision is Logistic Regression at 0.644 followed by Random Forest at 0.6218, KNN 0.617 and Decision Tree at 0.474.
- The highest recall is KNN at 0.594 followed by Logistic Regression at 0.565, Random Forest 0.478 and Decision Tree at 0.46.
- The highest f-1 score is KNN at 0.605 followed by Logistic Regression at 0.602, Random Forest 0.54 and Decision Tree at 0.467.
- The highest precision is Logistic Regression at 0.837 followed by KNN at 0.826, Random Forest Regression at 0.817 and Decision Tree at 0.638.

Conclusion & Improvements

Conclusion

- Logistic Regression has proven to be the most suitable model for prediction as it has obtained the highest score for accuracy, precision and auc.
- Although KNN may be a good choice as it has higher recall and f-1 score compared to other models, it still has lower accuracy, precision and auc score compared to Logistic Regression .
- The model that performs the poorest is Decision Tree with the lowest value for all scores.

Improvements

- Future method to improve better prediction results can be achieved using Boosting methods such as Ada-Boosting, Gradient Boosting or Xg-Boosting

Appendix

Link to Code

<https://github.com/cs-robot-collab/IBM-ML-DL/blob/master/IBM%20Machine%20Learning%20Classification%20Report.ipynb>