**Customer Churn Analysis**

**Problem Statement:-** Customer churn is when a company’s customers stop doing business with that company. Businesses are very keen on measuring churn because keeping an existing customer is far less expensive than acquiring a new customer. New business involves working leads through a sales funnel, using marketing and sales budgets to gain additional customers. Existing customers will often have a higher volume of service consumption and can generate additional customer referrals.

Customer retention can be achieved with good customer service and products. But the most effective way for a company to prevent attrition of customers is to truly know them. The vast volumes of data collected about customers can be used to build churn prediction models. Knowing who is most likely to defect means that a company can prioritise focused marketing efforts on that subset of their customer base.

Preventing customer churn is critically important to the telecommunications sector, as the barriers to entry for switching services are so low.

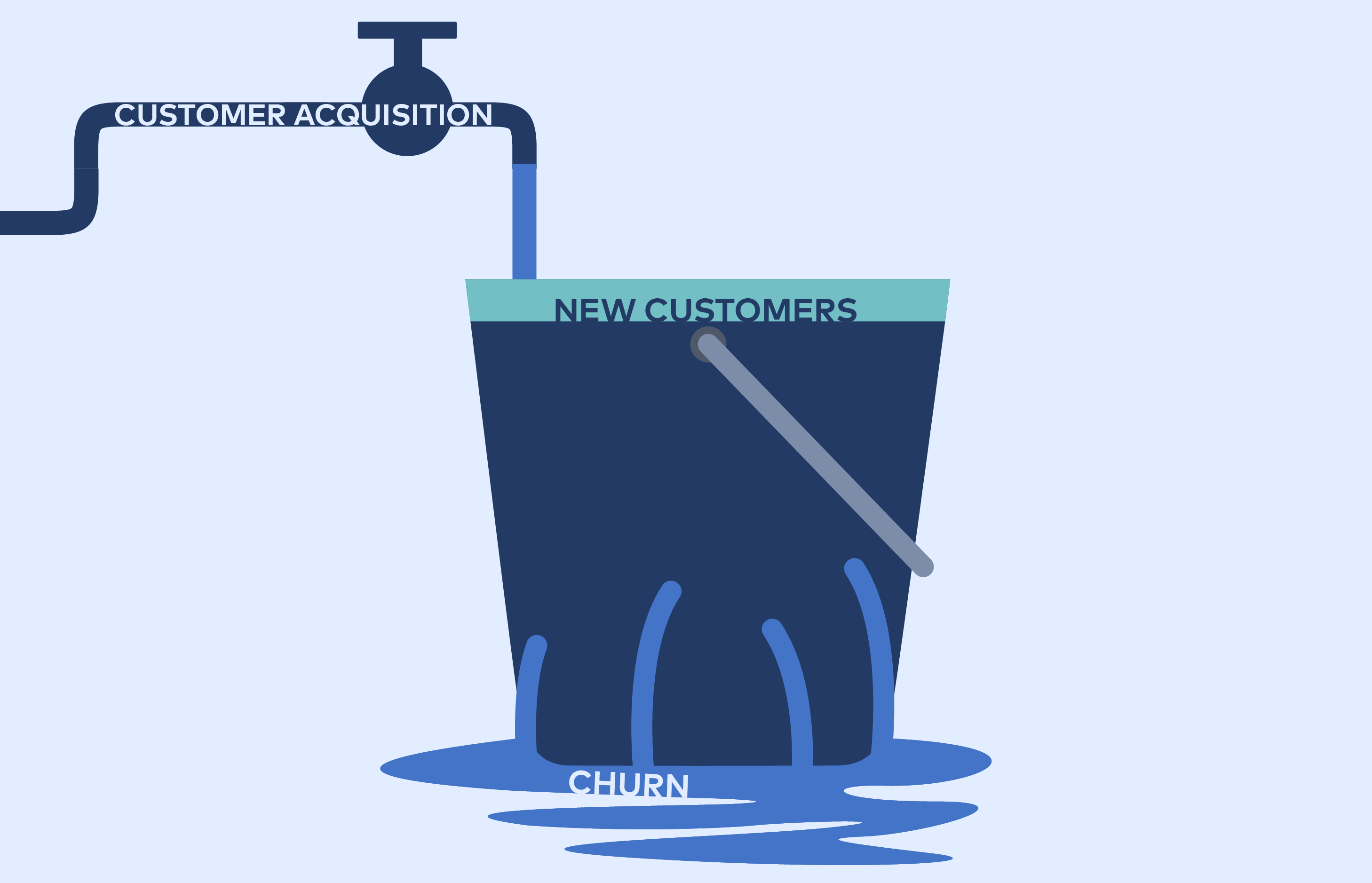
**---------WHAT WE UNDERSTAND FROM PROBLEM STATEMENT--------**

From Problem statement we can understand that Customer Churn is when a Customer Stop doing Business with that company. Churn is Important because keeping and existing Customer is less expensive than acquiring the new Customer. Existing Customer will often have a higher volume of service consumption and can generate additional customer referrals.

The most effective way for a company to prevent churn is to truly know them. The vast volume of data collected about the customers can be used to build Churn prediction models.

Preventing Customer churn is important to telecommunication industry as the customer switching the. Service is very low.

**So from above we can understand that it is important to lower the churn rate of the telecommunication industry because as the customer changes his/her telecommunication company there is very less chances that he/she will be coming back to the First Company**



**From above diagram we can easily get the definition of Churn in an Industry.**

**Now we will be doing some steps to find out the Users who are going to Churn.**

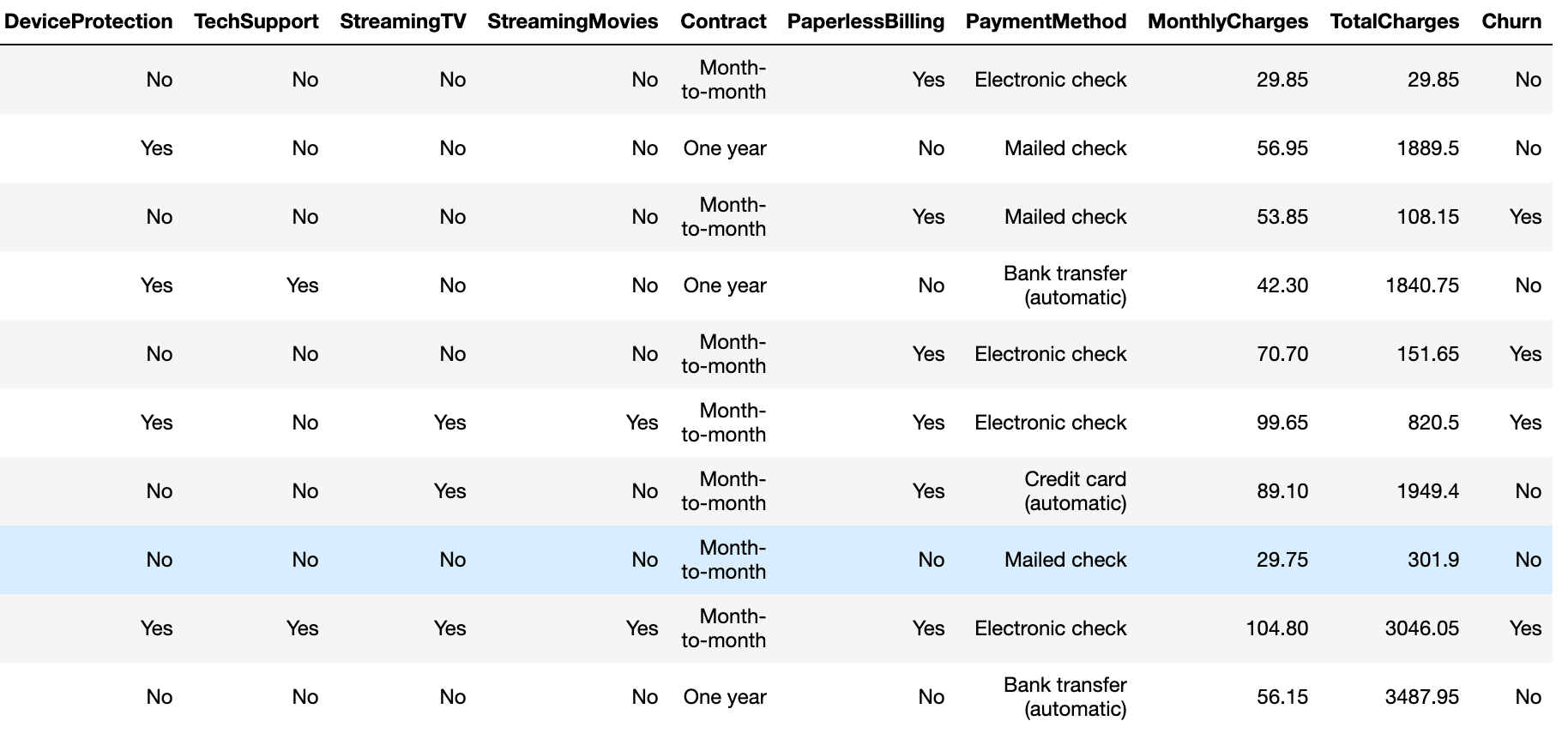
**Methodology :-**

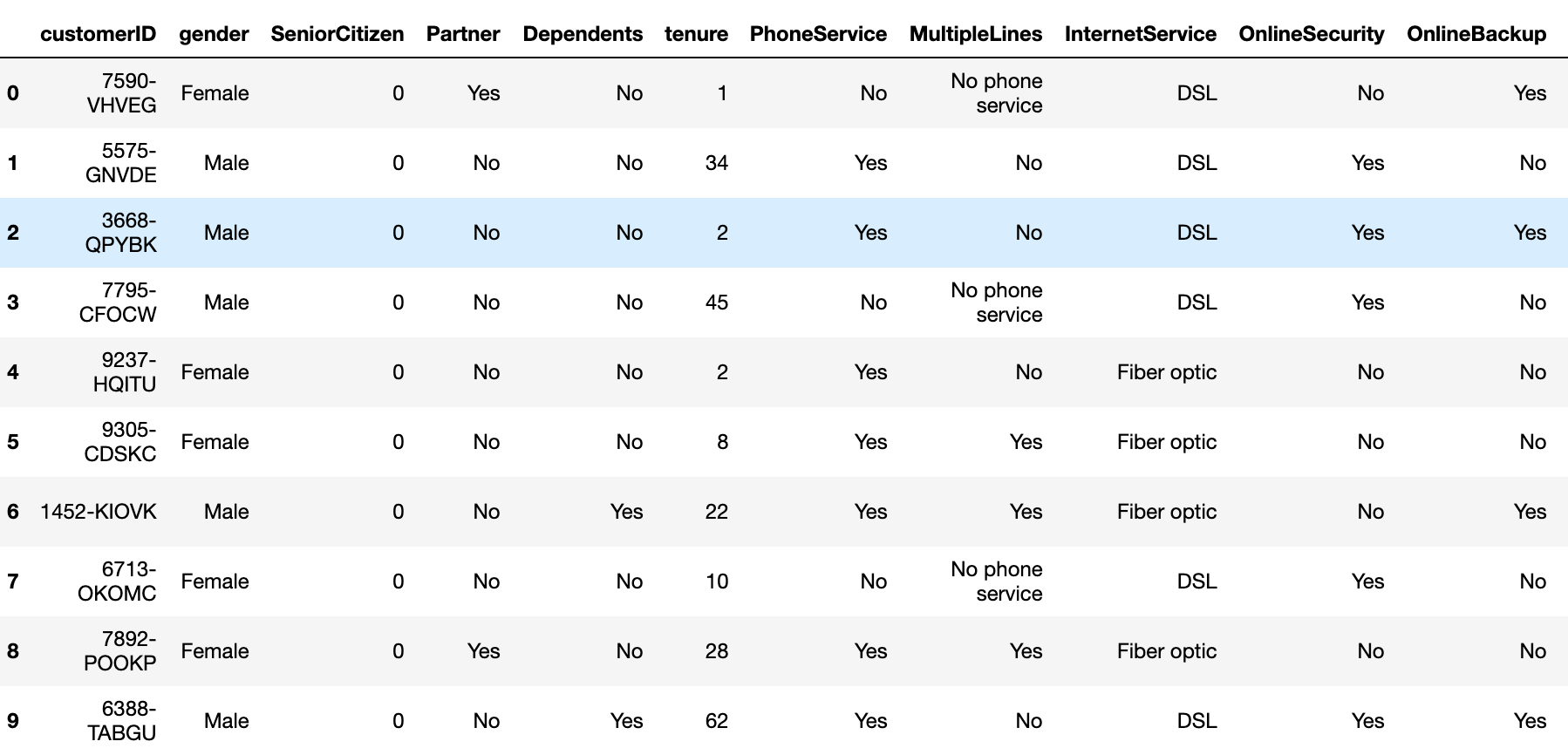
1. **DATA COLLECTION**
2. **DATA PRE-PROCESSING**
3. **DIVIDING THE DATA into TWO PARTS ‘TRANING’ AND ‘TESTING’.**
4. **BUILD UP THE MODEL USING TRAINING DATASET**
5. **DO THE ACCURACY TEST USING TESTING ACCURACY**
6. **FINDING THE CROSS-VALIDATION SCORE**
7. **COMPARING ACCURACY SCORE AND CROSS VALIDATION SCORE AND FIND OUT BEST ALGORITHM.**
8. **SAVING THE BEST MODEL FOR FURTHER USE.**

**DATA EXPLORATION: -**

This data is from IBM Sample Data Sets with the aim of building and comparing several customer churn prediction models. Here our target variable is Churn and it has 2 classes that is ‘Yes’ and ‘No’ by looking at the Churn column we can say that it is a classification problem as we have to classify the customer on the basis of they will churn or not.

**Lets have a quick look at the dataset: -**





**From above images we can see how the dataset looks like the last column named churn is our output variable i.e. we have to predict churn.**

**Data Pre-Processing :-**

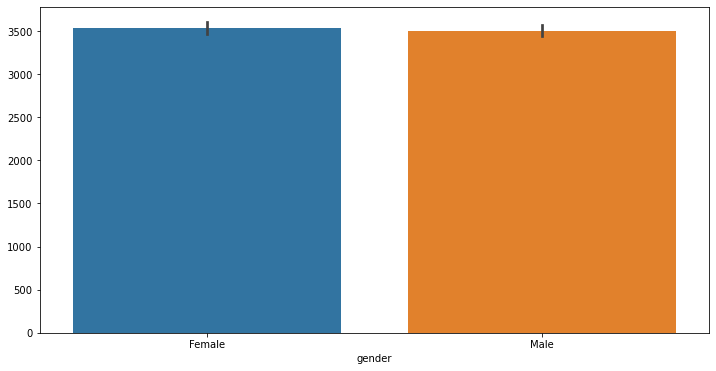
Lets now look at the missing values in the data using ds.isnull().sum() and if present we will fix them using mean, median or mode operations. But lets see if there are any missing values and its diagram using heatmap. From using above function ds.isnull().sum() we are getting zero missing values lets look at the heatmap of this function.



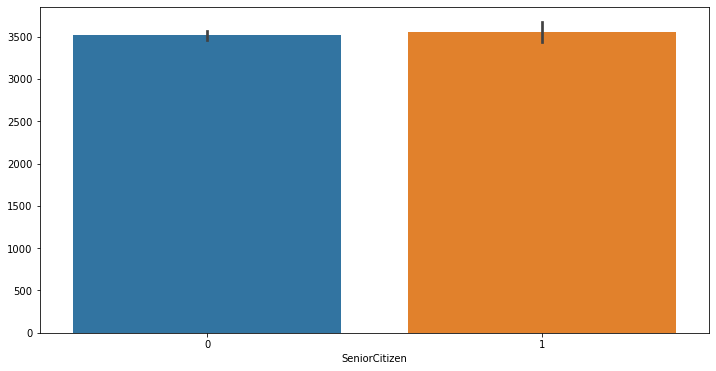
From above we can see that there is only 1 color in the heatmap so there is no missing values in the dataset

By using the above method we can easily convert categorical to numeric variable.

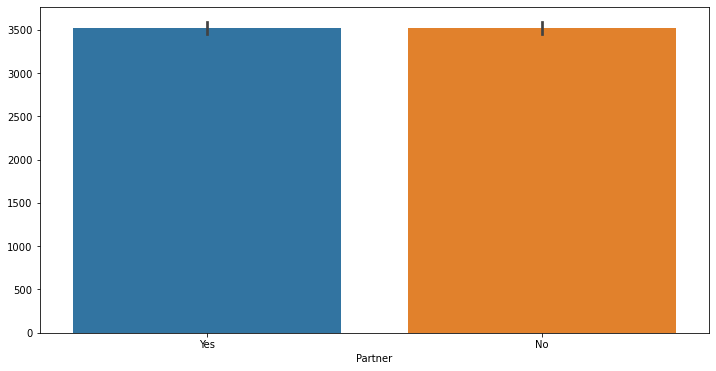
**Lets now look at the Churn Analysis Exploratory Data Analysis:-**



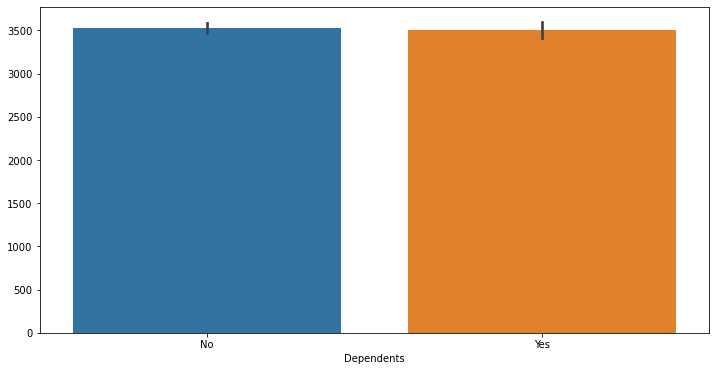
from above we can see that there are 3488 Female and 3555 male in dataset

****

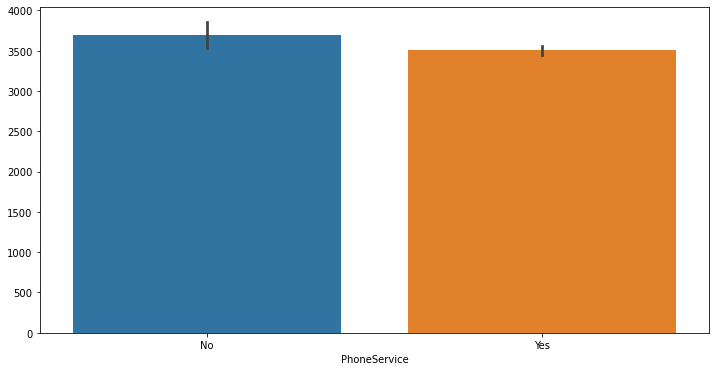
from above we can see that there are 1142 Senior Citizen in the dataset



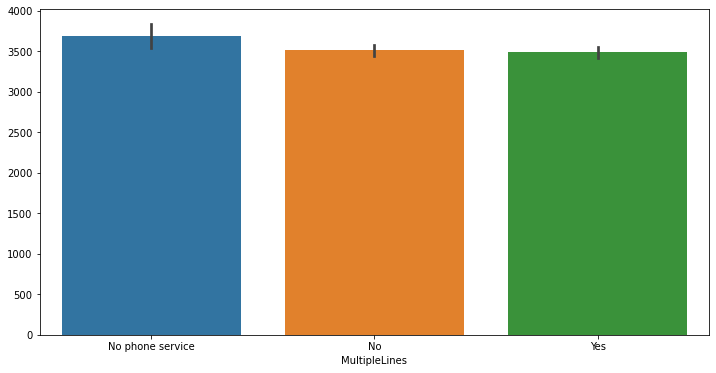
from above we can see that there are 3402 customer with Partner in the same Telecom company



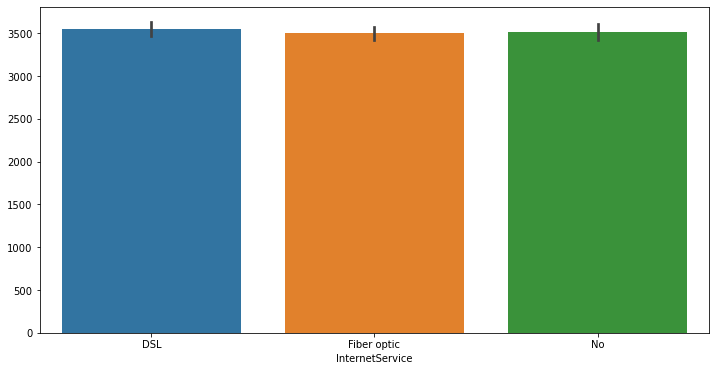
from above we can see that there are 2110 customer who has dependents



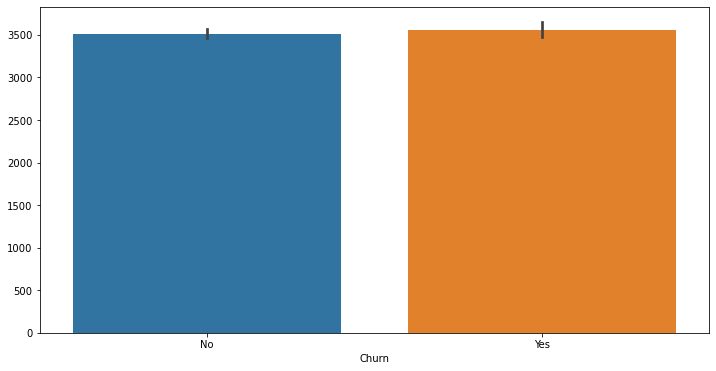
we can see that from all the customers 6361 of them has taken phone service



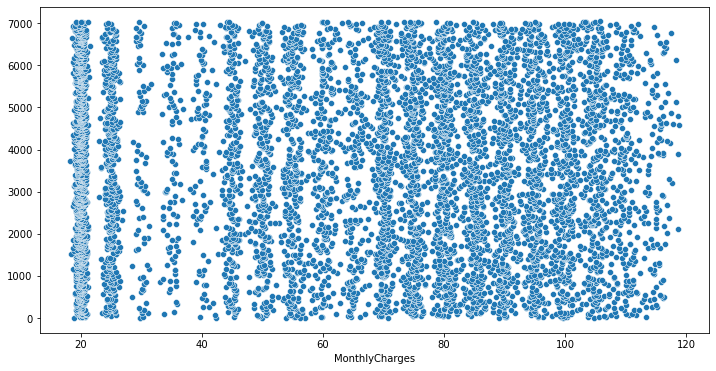
from above we can see that there are 2971 people with multiple lines 3390 without multiple line and 682 of them don't have phone service



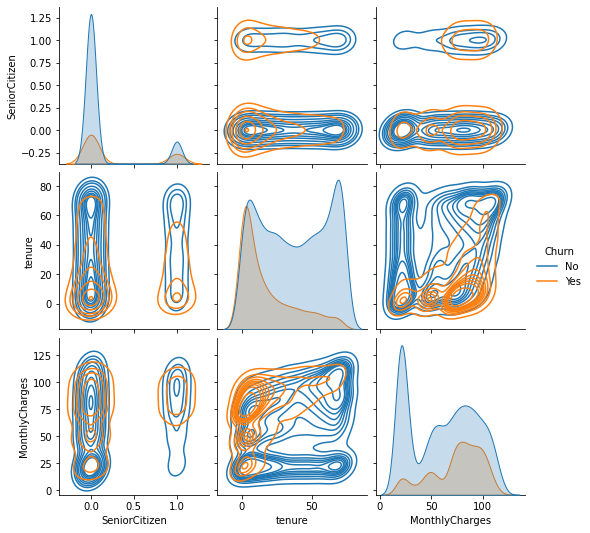
from above we can see the type of Service the person is using most of them are usinf Fiber Optics after that 2421 people are using digital subscriber line and 1526 people haven't taken the internet service



from above we can see there are 1869 people who are leaving the service and 5174 who are staying with us.



From here we can see the scatter plot of the monthly charges and the number of counts of the dataset it is evenly distributed.



From above we can see all the numeric column with the churn variable using seaborn.

sns.pairplot(ds, hue='Churn', data = ds, kind = 'kde')

Here we need to do LabelEncoding as there are many object type columns in the dataset by using label encoding we can easily convert Categorical values into Numerical values.

By using the below code we will convert all the object column into numeric

le = LabelEncoder()

ds.select\_dtypes('object')

ds[ds.select\_dtypes(include=['object']).columns] = ds[ds.select\_dtypes(include=['object']).columns].apply(le.fit\_transform)

from above code all the object type variable will be converted into numeric type without loosing their impact on the dataset

Now lets see the correlation between our data using ds.corr()



**From above we can see the heatmap of Correlation of the dataset and we can understand the correlation by looking at the graph using the color combination it is very easy to understand if the color is dark then the correlation is Positive and when the color is light the correlation between the column decreases and if the column is white then the column has negative correlation.**

Lets now separate our independent and dependent variable and saved them in X and Y

From above graph we can see the normal distributed column.

**X = Independent Variable**

**Y = Dependent / Output Variable**

X = ds.drop(‘Churn’, axis = 1)

Y = ds[‘Churn’]

Here we have specified to drop the dependent variable from axis = column i.e. axis = 1

As we now have Dependent variable and independent variable lets now do Standard Scaling to our Independent variable to convert all the values in same scale.

X\_new = sc.fit\_transform(X)

X\_new = pd.DataFrame(X\_new, columns = X.columns)

X\_new.head()

By using above code we can Scale all out independent column values.

Lets now split our data using train\_test\_split

X\_train, X\_test, Y\_train, Y\_test = train\_test\_split(X, Y, test\_size = 0.22, random\_state = 42)

From above we have successfully separated our training and testing dataset with testing size as 22% and training size as 78% we can also do as 20 – 80 or 30 – 70 and here random state we are taking as 42 later we will find out the best random state. Using python loop.

lr = LogisticRegression()

lr.fit(X\_train, Y\_train)

lr.score(X\_train, Y\_train)

predlr = lr.predict(X\_test)

print("Accuracy Score:", accuracy\_score(Y\_test, predlr))

print("Classification Report", classification\_report(Y\_test, predlr))

print("Confusion Matrix", confusion\_matrix(Y\_test, predlr))

by writing the above code we can see the below output and we can see lots of maths by looking at it

**Accuracy score: 81.29032258064515**

**Classification Report: precision recall f1-score support**

**0 0.85 0.91 0.88 1133**

**1 0.69 0.55 0.61 417**

**accuracy 0.81 1550**

**macro avg 0.77 0.73 0.74 1550**

**weighted avg 0.80 0.81 0.81 1550**

**Confusion Matrix: [[1031 102]**

**[ 188 229]]**

from above we can see that testing accuracy us 81% now lets find out the best random state

From above we can see that Accuracy score is 81% for Logistic regression and by looking at this we can say that model has performed very well but why we used other metrics? Because we want to see how the model works with different kind of data from classification report we can see f1 score from there we can see 0 is predicted 91% of time and 1 is predicted 55% of time that means f1 score of the model is 55% as Churn is Important for us we have to look at F1 score to as it is important to us so lets see which model performs well in all the metrics. Confusion Matrix is solved as 1031+102 / 1031+102+188+229 it will give 81% same as the Accuracy score so why this by using this we can find our correct predictions and the false predictions.

Lets now find out the best random state and the best accuracy score of the dataset using python loop and the classification models

Step 1: Create a List of model

models = [LogisticRegression(), DecisionTreeClassifier(), RandomForestClassifier(), AdaBoostClassifier(), SVC(), KNeighborsClassifier(), GaussianNB()]

Step 2: Write the Logic:

maxacc = 0

maxrs = 0

n = 0

for i in range(1,1000):

X\_train, X\_test, Y\_train, Y\_test = train\_test\_split(X, Y, test\_size = 0.22, random\_state = i)

for m in models:

m.fit(X\_train, Y\_train)

pred = m.predict(X\_test)

acc = accuracy\_score(Y\_test, pred)

if acc > maxacc:

maxacc = acc

maxrs = i

n = m

print("Maximum Accuracy is {} at random\_state {} for model {}". format(maxacc, maxrs, n))

from above we can see that maxacc gives the maximum accuracy for the model maxrs gives the maximum Accuracy random state and n gives the name of the model which has the highest Accuracy at random state = i

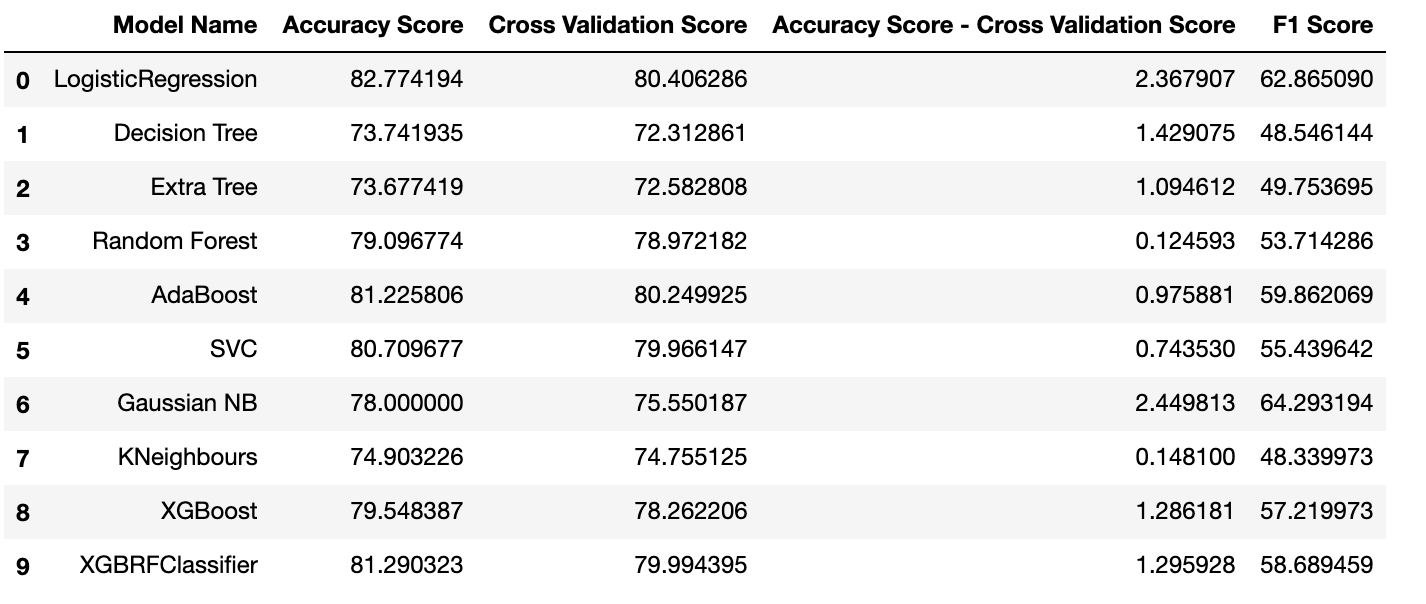
By running above code I got:

**Maximum Accuracy is 0.827741935483871 at Random State 158 for model LogisticRegression()**

I got 82.77% accuracy score at random state 158 for LogisticRegression()

Now we fill fit all the model with random state 158 for getting the better result

After finding all the model accuracy score we will see the Cross\_validation\_Score to find out if the model is Overfitting the data or Underfitting the data after that we will compare the accuracy score after cross validation and before cross validation and see which model has performed better.



From above we can see that all the models has performed very well after cross validation also so now lets look at the f1 score and choose our best model.

from above we can see that:

Logistic Regression

Gaussian NB has the heighest f1 score so we will choose this models as our best models lets now do GridSearchCV for our best models

**We Got the Accuracy Metrics as: -**

**Logistic Regression**

**Logistic Regression**

**Accuracy Score after GridSearchCV: 82.51612903225806**

**Classification Report: precision recall f1-score support**

**0 0.84 0.93 0.89 1130**

**1 0.75 0.53 0.62 420**

**accuracy 0.83 1550**

**macro avg 0.80 0.73 0.75 1550**

**weighted avg 0.82 0.83 0.81 1550**

**Confusion Matrix: [[1055 75]**

**[ 196 224]]**

**F1 Score: 62.30876216968011**

**Gaussian NB**

**Gaussian NB**

**Accuracy Score: 78.0**

**Classification Report: precision recall f1-score support**

**0 0.89 0.80 0.84 1130**

**1 0.57 0.73 0.64 420**

**accuracy 0.78 1550**

**macro avg 0.73 0.76 0.74 1550**

**weighted avg 0.80 0.78 0.79 1550**

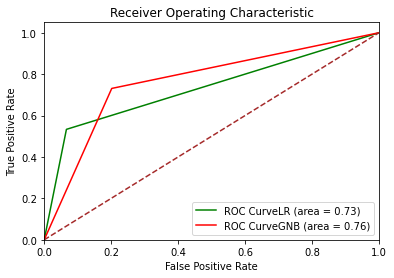
**Confusion Matrix: [[902 228]**

**[113 307]]**

**F1 Score: 64.2931937172775**

**from here we see that Gaussian NB is performing better as it is important for us to find out the customer who us going to Churn.**

**Lets now perform AUC ROC CURVE and then we will select our best model.**

****

**From above we can see the AUC ROC CURVE and we can see that GaussianNB is measuring the performance across all possible classification thresholds with an Area under Curve of 76% so we are choosing this model as our final model and save it as an Object file using joblib**

**joblib.dump(gscvgnb.best\_estimator\_,'CustomerChurnAnalysis.pkl')**

**Conclusion:-**

As Telecom Industries run on Customer so it is important for the company to save the customer from churn as if the person Churns and go to other telecom Industry it is difficult for the previous company to retain that customer from other company and because of Churn the company suffer losses and even they might loose many other customer as there can be dependents, Siblings those will be following the same as the other family member and even the Internet service can get cancelled by them after changing to other industry.

So to prevent the churn of the customer telecom company analysis the data and find out who has the higher probability to churn and after predicting that they can give those customer some more services to change there mind from churning and make them more profit by adding new people from their circle thinking about they will get the same services as he is getting hence increasing the revenew of the Telecom Company.