

DEVELOPMENT PHASE PART 1

Customer Churn Prediction Project

Date	26-09-2023
Team ID	1288
Project Name	Customer Churn Prediction

DATA PRE-PROCESSING

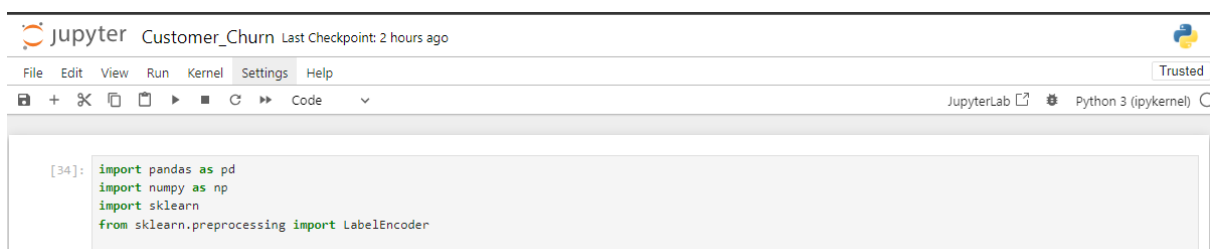
Data pre-processing is a component of data preparation, describes any type of processing performed on raw data to prepare it for another data processing procedure. It has traditionally been an important preliminary steps for the data mining process. More recently, data pre-processing techniques have been adapted for training machine learning models and AI models and for running inference against them.

We used the jupyter platform for the data pre-processing phase. In that we initially imported the necessary python library files. The library files were

Pandas - used for working with data sets

Numpy - used for working with arrays

Sklearn - Used machine learning models and statistical modelling



```
[34]: import pandas as pd
import numpy as np
import sklearn
from sklearn.preprocessing import LabelEncoder
```

Then we imported the given data set “WA_Fn-UseC_-Telco-Customer-Churn.csv” and viewed the first 5 rows of the dataset

```
[35]: df = pd.read_csv('WA_Fn-UseC_-Telco-Customer-Churn.csv')
df.head()
```

```
[35]:
```

	Customer ID	Gender	Senior Citizen	Partner	Dependents	Tenure	Phone Service	Multiple Lines	InternetService	Online Security	...	Device Protection	TechSupport	Streaming TV	StreamingMovies	Churn
0	7590-VHVEG	Female	0	Yes	No	1	No	No phone service	DSL	No	...	No	No	No	No	No
1	5575-GNVDE	Male	0	No	No	34	Yes	No	DSL	Yes	...	Yes	No	No	No	No
2	3668-QPYBK	Male	0	No	No	2	Yes	No	DSL	Yes	...	No	No	No	No	No
3	7795-CFOCW	Male	0	No	No	45	No	No phone service	DSL	Yes	...	Yes	Yes	No	No	No
4	9237-HQITU	Female	0	No	No	2	Yes	No	Fiber optic	No	...	No	No	No	No	No

5 rows × 21 columns

We started the data pre-processing steps by copying the given dataset.

```
[36]: #data preprocessing
df_transformed = df.copy()
```

The necessary dependent variables were selected and variable as columns.

```
[37]: columns1 = ['Gender', 'Partner', 'Dependents', 'Paperless Billing', 'Churn', 'Phone Service']
```

Afterwards in the dataset the datum we modified to binary values (0's & 1's)

```
[38]: for i in columns1:
    if i == 'Gender':
        df_transformed[i] = df_transformed[i].map({'Female': 0, 'Male': 1})
    else:
        df_transformed[i] = df_transformed[i].map({'yes': 1, 'No': 0})
```

All the columns we contained in a variable named as `df_transformed`

```
[39]: e Lines', 'InternetService', 'Online Security', 'OnlineBackup', 'Device Protection', 'TechSupport', 'Streaming TV', 'StreamingMovies', 'Contract', 'PaymentMethod']
```

```
[40]:
```

We created a dummies of the given dataset and stored in the variable named as `df_transformed`

```
[40]:
```

```
df_transformed = pd.get_dummies(df_transformed, columns = columns2)
```

Again we made a copy of the dataset as `df1`

```
[42]:
```

```
df1 = df_transformed.copy()
```

In this step, a for loop is used and used a `fit_transform` library from the sklearn library

```
[43]: for i in df_transformed.columns:
```

```
df1[i] = lenc.fit_transform(df_transformed[i])
```

Then we again printed the first 5 rows from the dataset with head function

```
[44]: df1.head()
```

	Customer ID	Gender	Senior Citizen	Partner	Dependents	Tenure	Phone Service	Paperless Billing	Monthly Charges	TotalCharges	...	StreamingMovies_No	StreamingMovies_No internet service	StreamingMovies_No
0	5375	0	0	1	0	1	0	1	142	2505	...	1	0	0
1	3962	1	0	0	0	34	1	0	498	1466	...	1	0	0
2	2564	1	0	0	0	2	1	1	436	157	...	1	0	0
3	5535	1	0	0	0	45	0	0	266	1400	...	1	0	0
4	6511	0	0	0	0	2	1	1	729	925	...	1	0	0

5 rows × 14 columns

We used a dropna function to remove the rows which contains null values

```
[45]: df_cleaned = df.dropna()
```

Then the drop_duplicate function is used to remove the duplicate rows.

```
[48]: # Remove duplicate rows
df_cleaned = df.drop_duplicates()
```

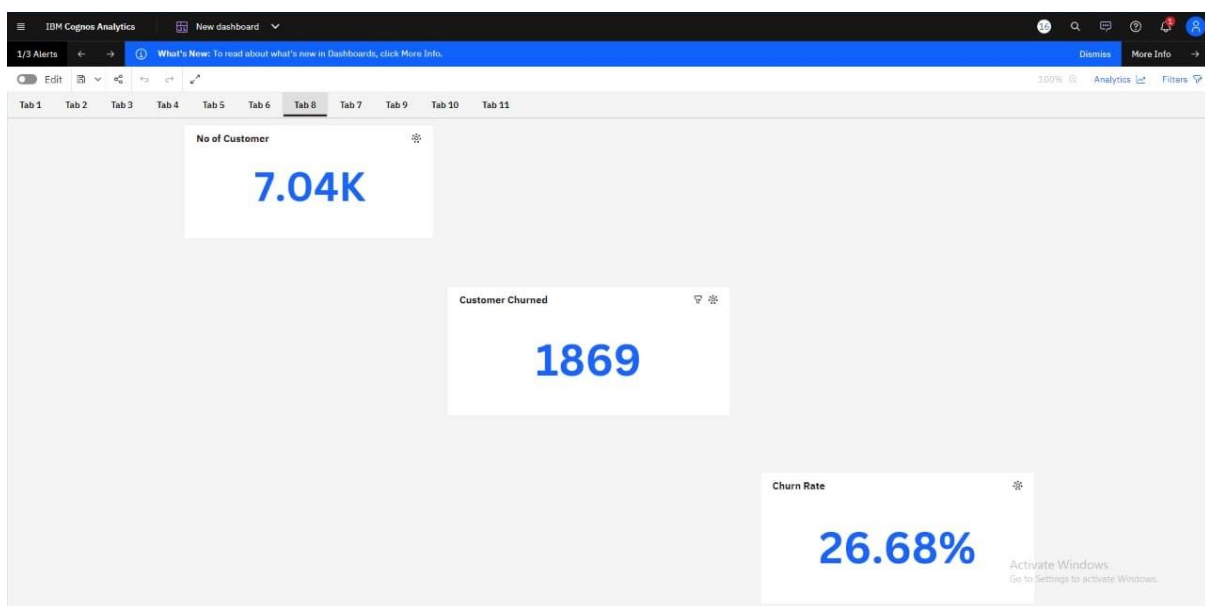
Here we replaced the path where we want to save the cleaned data..

```
[49]: output_csv = 'cleaned_file.csv' # Replace with the path where you want to save the cleaned data
df_cleaned.to_csv(output_csv, index=False)
```

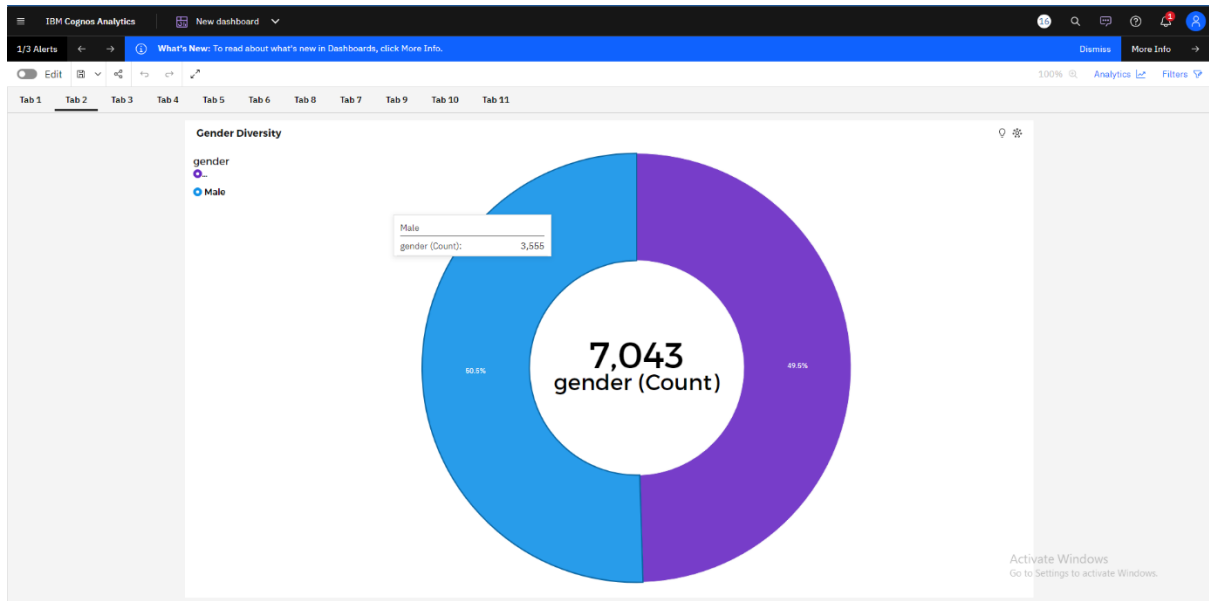
VISUALIZATION

The visualization were implemented by the IBM Cognos platform. Where we made a many visualization with the given telco dataset by merging lots of columns in that plstform.

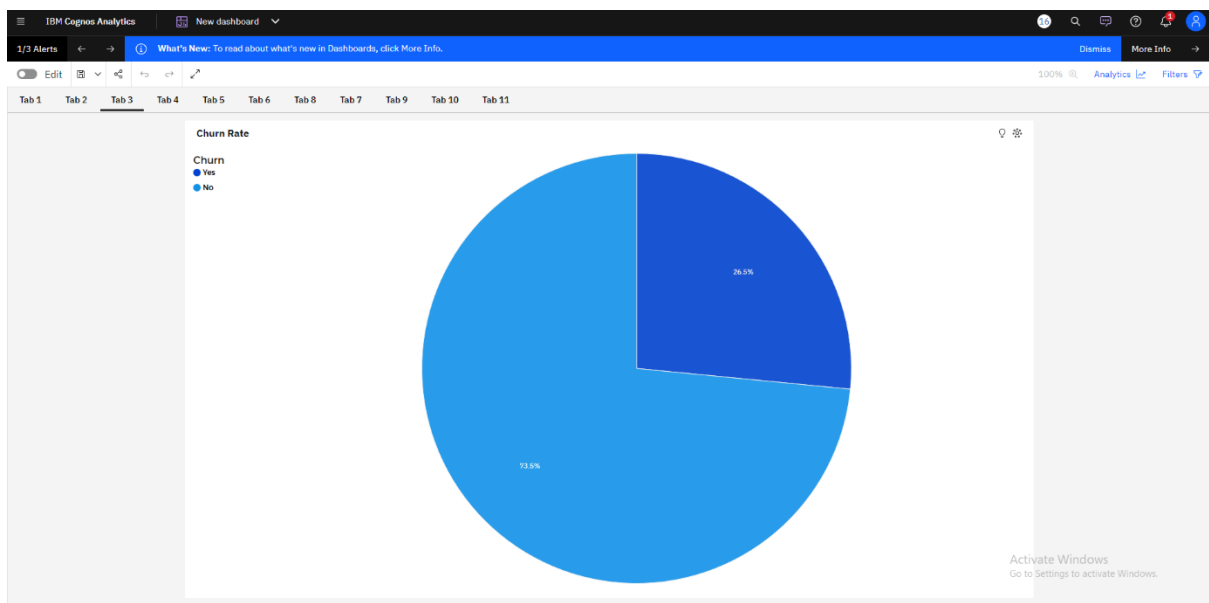
1.No.of Cusomer , Churned Customer , Churn Percentage



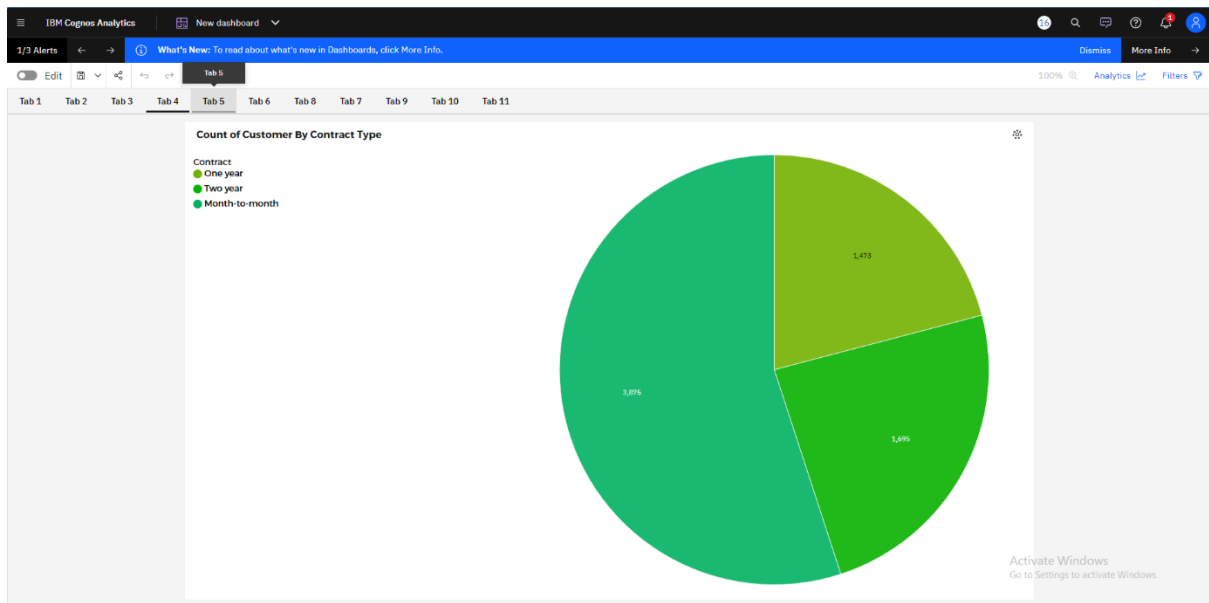
2. Gender Diversity



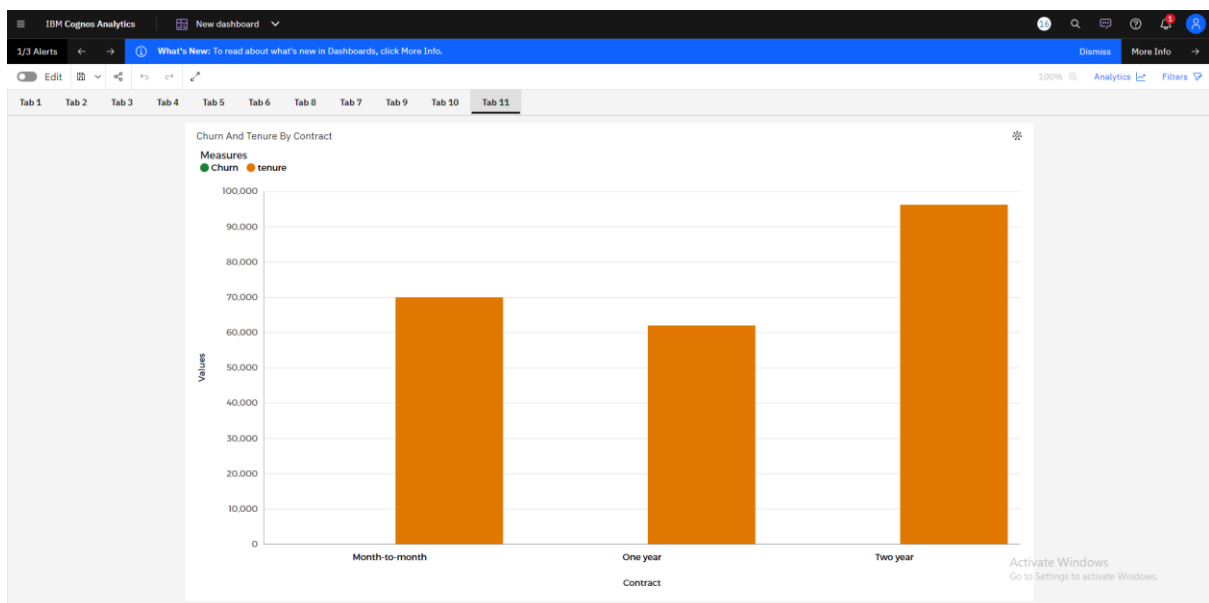
3. Churn Rate



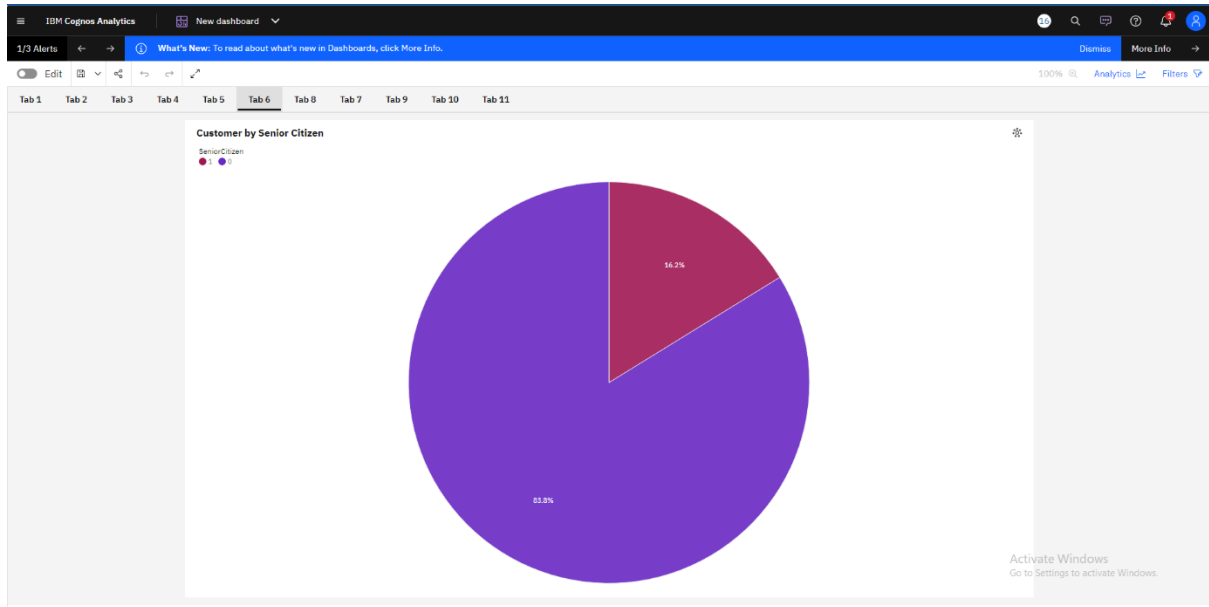
4.Count of Customer By Contract Type



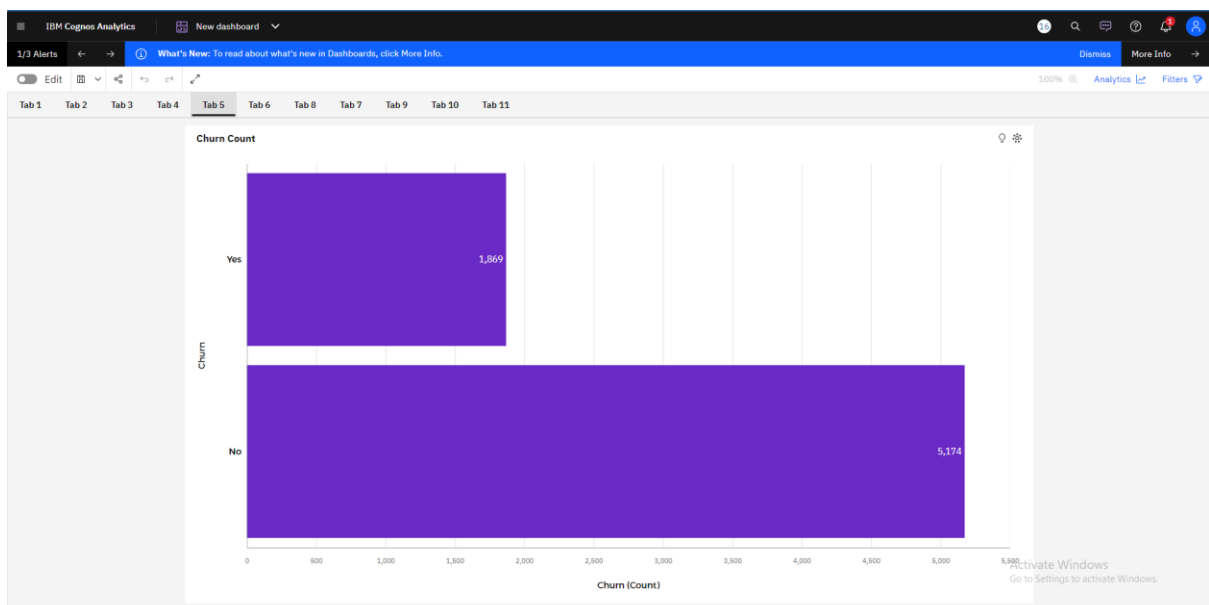
5.Churn And Tenure By Contract



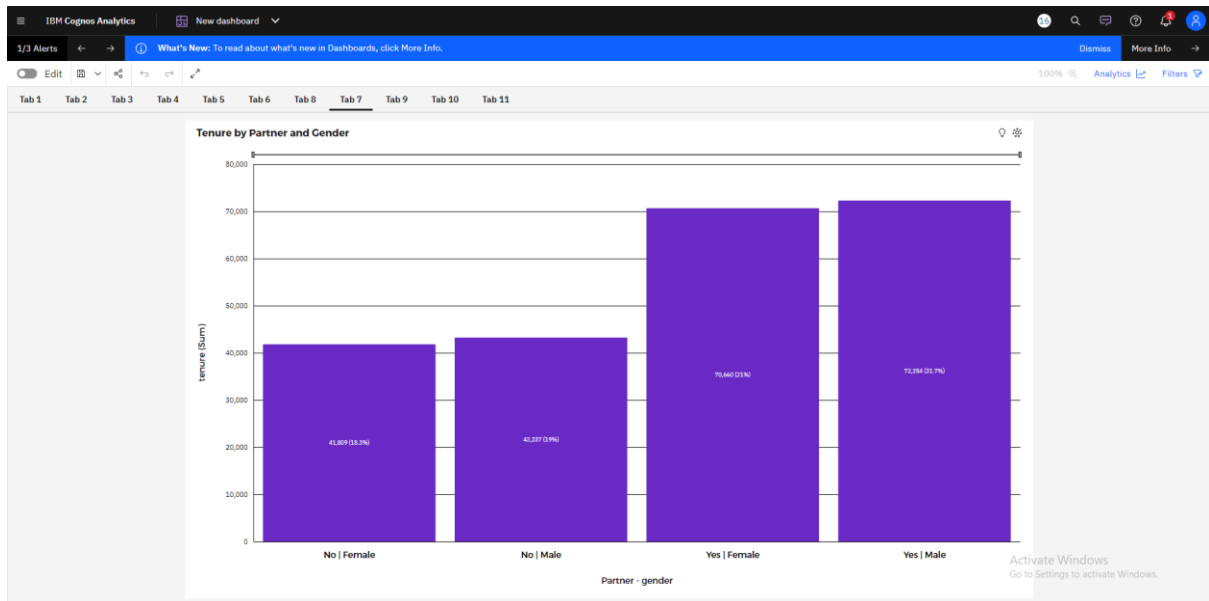
6.Customer By Senior Citizen



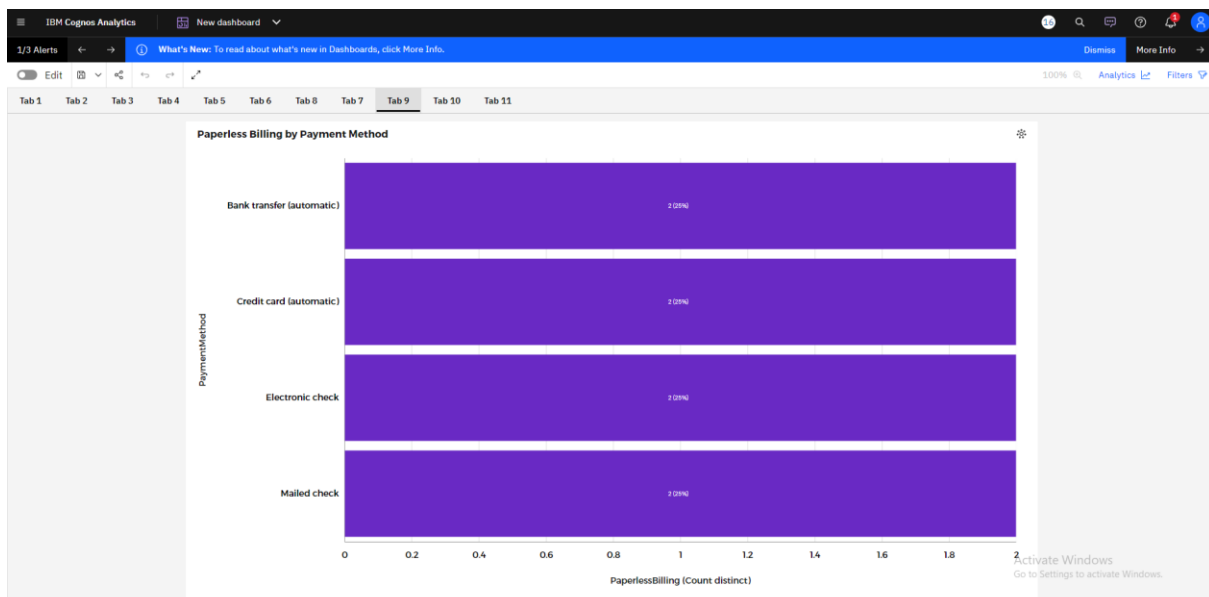
7.Churn Count



8.Tenure By Partner And Gender



9.Paperless Billing By Payment Method



10. Internet Services

