

```
colab.research.google.com/drive/1F5QVNEFxIMQO9ncypEzy6ifsKlbaEttgq
```

Customer churn prediction.ipynb

File Edit View Insert Runtime Tools Help

Commands + Code + Text Run all

```
#customer churn rate prediction based on be demographic data
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
df=pd.read_csv("Tele_customer_dataset.csv")
print(df.head(10))
```

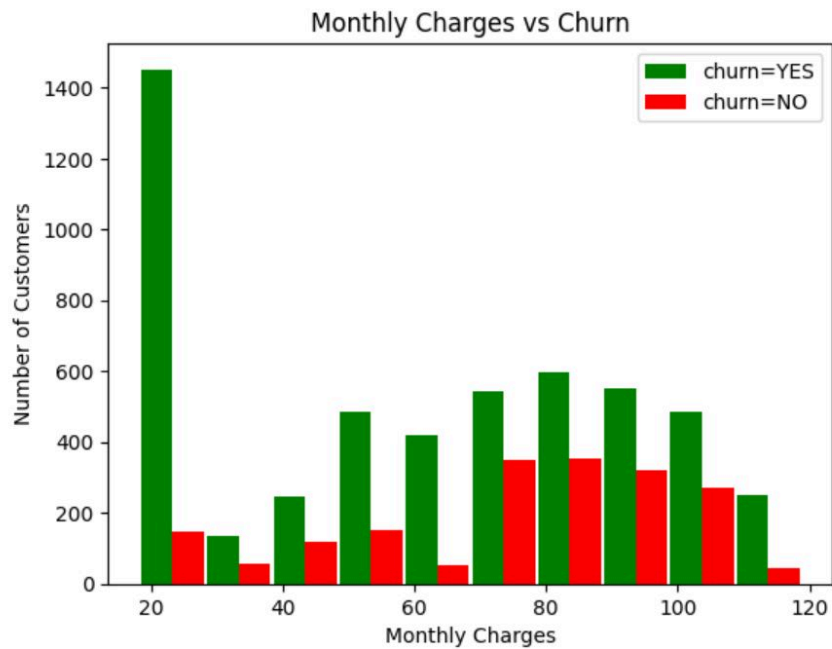
	customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	\
0	7598-VHVEG	Female	0	Yes	No	1	No	
1	5575-GNVDE	Male	0	No	No	34	Yes	
2	3668-QPYBK	Male	0	No	No	2	Yes	
3	7795-CFOCM	Male	0	No	No	45	No	
4	9237-HQITU	Female	0	No	No	2	Yes	
5	9305-CDSKC	Female	0	No	No	8	Yes	
6	1452-KIOVK	Male	0	No	Yes	22	Yes	
7	6713-OKOMC	Female	0	No	No	10	No	
8	7892-POOKP	Female	0	Yes	No	28	Yes	
9	6388-TABGU	Male	0	No	Yes	62	Yes	

	MultipleLines	InternetService	OnlineSecurity	... DeviceProtection	\
0	No	phone service	DSL	No	...
1	No	No	DSL	Yes	...
2	No	No	DSL	Yes	...

## MONTHLY CHARGES V/S CHURN RATE BARPLOT

```
plt.show()
```



In [28]: `#data cleaning as ml dont understand variables in data set so we gonna change in 0s and 1s`

## CLASSIFICATION REPORT

```
else:  
    y_pred.append(0)  
  
y_pred[:10]
```

44/44 ————— 0s 935us/step

Out[49]: [0, 1, 0, 1, 0, 1, 0, 0, 0, 0]

```
In [50]: from sklearn.metrics import confusion_matrix,classification_report  
print(classification_report(y_test,y_pred))
```

	precision	recall	f1-score	support
0	0.82	0.86	0.84	999
1	0.61	0.52	0.56	408
accuracy			0.77	1407
macro avg	0.71	0.69	0.70	1407
weighted avg	0.76	0.77	0.76	1407

```
In [51]: #confusion matrix  
import seaborn as sn  
cm=tf.math.confusion_matrix(labels=y_test,predictions=y_pred)  
plt.figure(figsize=(10,7))  
sn.heatmap(cm,annot=True,fmt="d")  
plt.xlabel("Predicted")  
plt.ylabel("Truth")
```

## CONFUSION MATRIX

