

a4-churn

November 14, 2024

```
[16]: import numpy as np
import pandas as pd
churn_data = pd.read_csv('datasets/churn.csv', index_col=0)
```

```
[4]: print(churn_data.dtypes)
```

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

```
[5]: #Count total number of duplicate records in the dataframe
print(len(churn_data) - len(churn_data.drop_duplicates()))
```

7

```
[6]: #count duplicate records based on attribute (customerID)
df = churn_data['customerID'].duplicated()
print(sum(df)) #add all true values
```

7

```
[7]: #missing values in each column
print(churn_data.isnull().sum())
```

```
customerID      0
tenure          0
Contract        0
PaperlessBilling 0
PaymentMethod    0
MonthlyCharges  10
TotalCharges     15
gender          0
SeniorCitizen   5
Partner         0
Dependents      0
PhoneService    0
MultipleLines   0
InternetService 0
OnlineSecurity  0
OnlineBackup    0
DeviceProtection 0
TechSupport     0
StreamingTV     0
StreamingMovies 0
Churn           0
dtype: int64
```

```
[8]: #missing values in column Totalcharges
print(churn_data['TotalCharges'].isnull().sum())
```

15

```
[9]: #Average monthly charge paid by a customer for the services he/she has signed
      ↪up for
print(churn_data['MonthlyCharges'].mean())
```

62.473481781376535

```
[13]: #Display Records having "10#" under the variable Dependents
print(churn_data.loc[churn_data['Dependents']=='10#'])
#print(churn_data.loc[churn_data['Dependents']=='10#'].count())
```

	customerID	tenure	Contract	PaperlessBilling	\
90	1754-GKYPY	22	Month-to-month	Yes	
126	9108-EQPNQ	10	Two year	No	
175	2640-PMGFL	27	Month-to-month	Yes	
221	8854-CCVSQ	18	Month-to-month	Yes	

235	6876-ADESB	One	Month-to-month	No
239	1972-XMUWV	65	Two year	Yes

	PaymentMethod	MonthlyCharges	TotalCharges	gender	\
90	Bank transfer (automatic)	89.75	1938.90	Male	
126	Credit card (automatic)	26.10	225.55	Female	
175	Electronic check	79.50	2180.55	Male	
221	Electronic check	80.65	1451.90	Male	
235	Electronic check	48.95	48.95	Male	
239	Credit card (automatic)	59.80	3808.20	Female	

	SeniorCitizen	Partner	...	PhoneService	MultipleLines	InternetService	\
90	1.0	Yes	...	Yes	No	Fiber optic	
126	0.0	Yes	...	Yes	Yes	No	
175	0.0	No	...	Yes	Yes	Fiber optic	
221	0.0	No	...	Yes	Yes	Fiber optic	
235	0.0	No	...	Yes	No	DSL	
239	0.0	Yes	...	Yes	No	DSL	

	OnlineSecurity	OnlineBackup	DeviceProtection	\
90	No	No	No	
126	No internet service	No internet service	No internet service	
175	No	No	No	
221	No	Yes	No	
235	No	No	Yes	
239	No	No	No	

	TechSupport	StreamingTV	StreamingMovies	Churn
90	No	Yes	Yes	No
126	No internet service	No internet service	No internet service	No
175	Yes	No	No	Yes
221	No	No	No	Yes
235	No	No	No	Yes
239	Yes	Yes	No	No

[6 rows x 21 columns]

```
[17]: #Replace null values by median value or by max count class category
print(churn_data.isnull().any())

churn_data = churn_data.apply(lambda x:x.fillna(x.median()) if x.dtype=='float'
    ↳else x.fillna(x.value_counts().index[0]))

#other way of replacing null values
#churn_data['MonthlyCharges'].fillna(churn_data['MonthlyCharges'].median(),
    ↳inplace = True)
```

```

#churn_data['TotalCharges'].fillna(churn_data['TotalCharges'].median(), inplace=True)
#churn_data['SeniorCitizen'].fillna(churn_data['SeniorCitizen'].median(), inplace = True)
print(".....")
print(churn_data.isnull().any())

print()

```

```

customerID      False
tenure          False
Contract        False
PaperlessBilling False
PaymentMethod   False
MonthlyCharges  True
TotalCharges    True
gender          False
SeniorCitizen   True
Partner         False
Dependents      False
PhoneService    False
MultipleLines   False
InternetService False
OnlineSecurity  False
OnlineBackup    False
DeviceProtection False
TechSupport     False
StreamingTV     False
StreamingMovies False
Churn           False
dtype: bool

```

```

...
customerID      False
tenure          False
Contract        False
PaperlessBilling False
PaymentMethod   False
MonthlyCharges  False
TotalCharges    False
gender          False
SeniorCitizen   False
Partner         False
Dependents      False
PhoneService    False
MultipleLines   False
InternetService False
OnlineSecurity  False

```

```
OnlineBackup      False
DeviceProtection  False
TechSupport       False
StreamingTV       False
StreamingMovies    False
Churn             False
dtype: bool
```

```
[ ]:
```