PHASE 3: DEVELOPMENT PART 1

PROJECT TITLE: ASSESSMENT OF MARGINAL WORKERS IN TAMILNADU-A SOCIO ECONOMIC ANALYSIS

TEAM MEMBERS:

SIVA GURU.K

NITHISH.S

MANIBARATHI

KARTHIKEYAN

DEVELOPMENT PART 1 – LOADING AND PREPROCESSING IN THE WORKERS DATASET USING IBM COGNOS

ANALYSIS OBJECTIVES:

1. Identify Marginal Workers: In the initial phase, our primary focus will be on establishing a comprehensive methodology to accurately identify and classify marginal workers in the Tamil Nadu region. This will involve data collection and categorization techniques.

2. Understand Socio-Economic Factors: We will begin by gathering and analyzing demographic and socio-economic data to gain a foundational understanding of the factors that contribute to the prevalence of marginal work in Tamil Nadu. This analysis will help us identify the key variables to consider.

3. Assess Livelihood Challenges: During this phase, we will conduct preliminary interviews and surveys with a sample of marginal workers to gain insights into the daily challenges they face. These insights will be used to shape our later, more extensive research.

4. Regional Disparities Exploration: Our initial geographical focus will be on one or more districts or regions within Tamil Nadu to understand the regional variations in the prevalence and challenges faced by marginal workers.

DATA COLLECTION:

We will collect customer data from the provided dataset available on Kaggle(link: https://tn.data.gov.in/resource/marginal-workers-classified-age-industrial-category-and-sex-scheduled-caste-2011-tamil)

DATA PREPROCESSING:

Data preprocessing is essential to ensure data quality and prepare it for analysis in IBM Cognos Analytics.

The following steps will be taken:

1.DATA LOADING:

Import the dataset into IBM COGNOS Analytics, ensuring it’s in a format compatible with the tool.

2.DATA EXPLORATION:

Perform exploratory data analysis (EDA) to gain an initial understanding of the dataset's structure and contents. This includes identifying missing values, outliers, and data types.

3.HANDLING MISSING VALUES:

Identify and address missing values in the dataset. Options may include imputation or removal, depending on the impact of missing data on the analysis.

4.DATA CLEANING:

Detect and address data anomalies, such as outliers and incorrect entries, which could negatively affect the analysis.

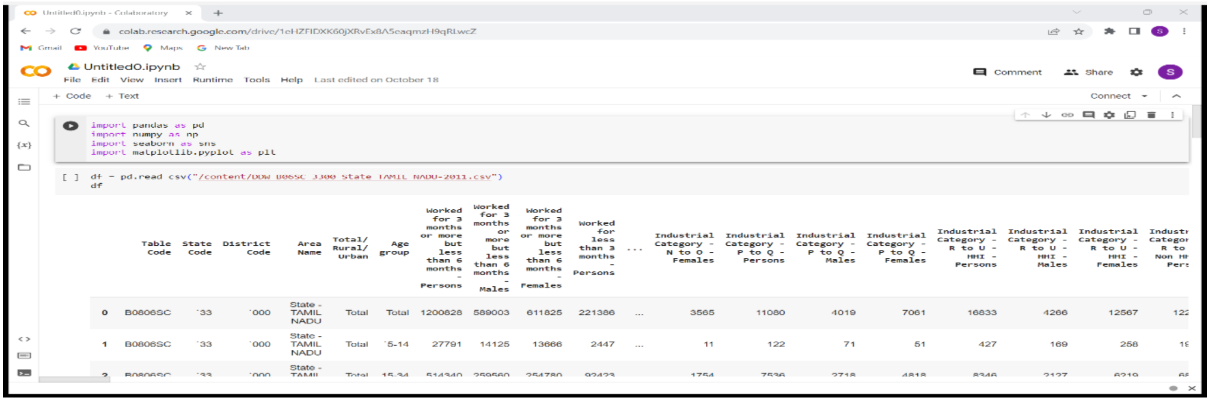
5.DATA TRANSFORMATION:

Transform data into a format suitable for modeling. Ensure that the target variable (churn status) is correctly defined.

6)DATA SPLITTING:

Divide the dataset into training and testing sets for model development and evaluation.

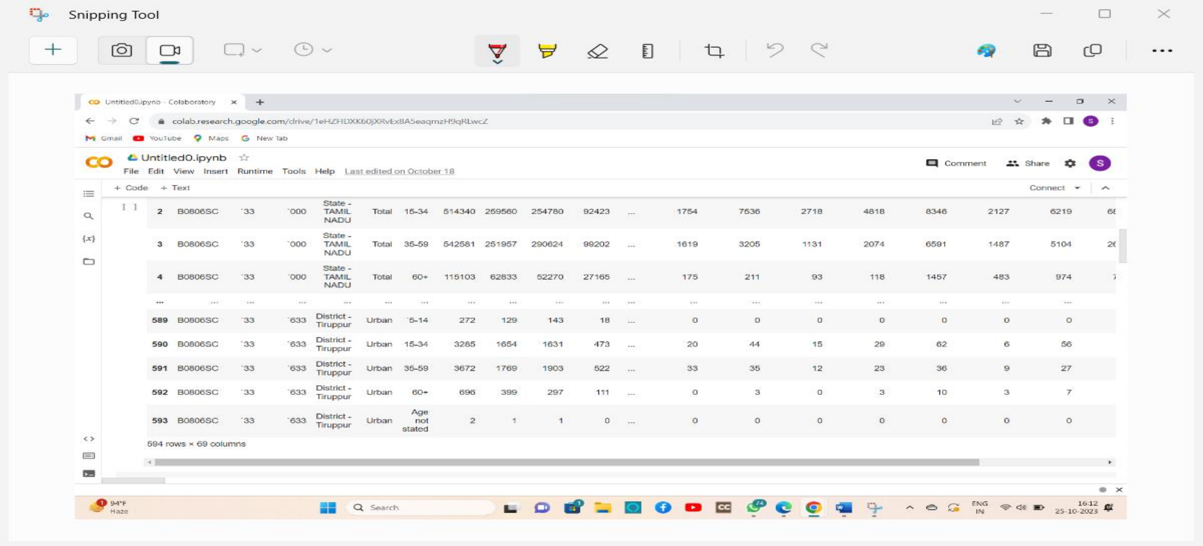
DATA EXPLORATION:


Untitled0.ipynb - Colaboratory
colab.research.google.com/drive/1eHZFIDXK60jXRvEx8A5eaqmzH9qRLwcZ
Gmail
YouTube
Maps
New Tab
Untitled0.ipynb
File
Edit
View
Insert
Runtime
Tools
Help
Last edited on October 18
Comment
Share
+ Code
+ Text
Connect

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

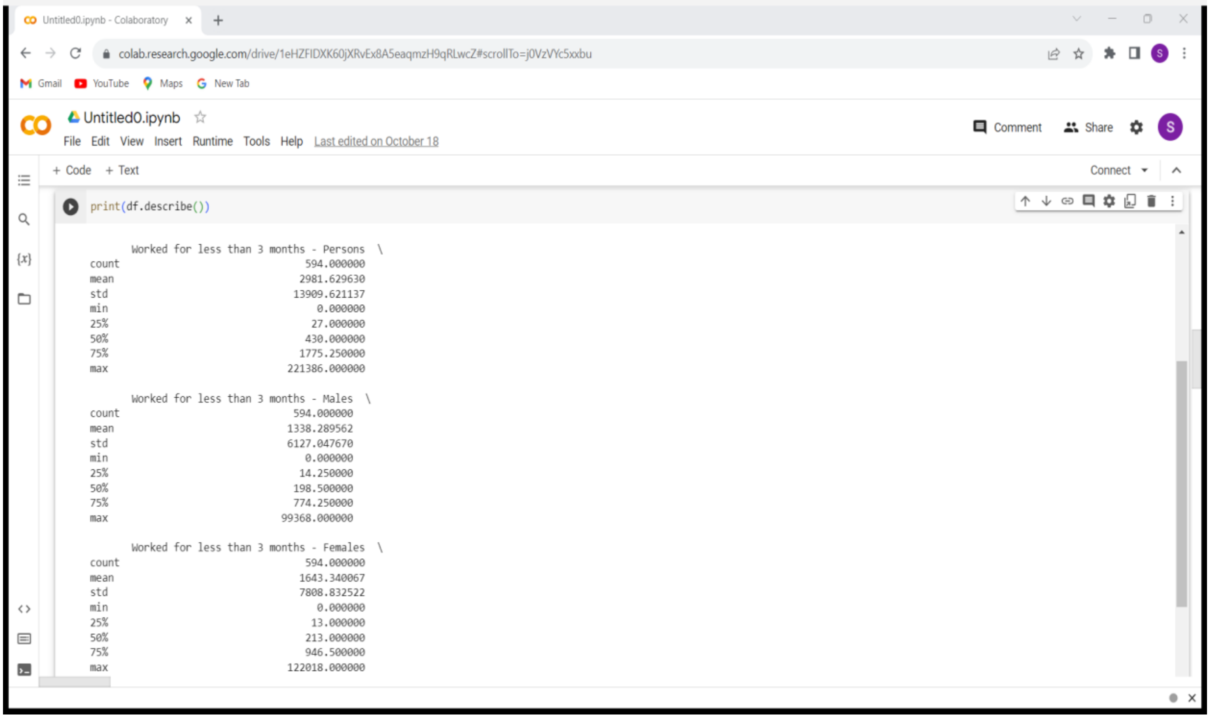
```
df = pd.read_csv("/content/DDW_B06SC_3300_State_TAMIL_NADU-2011.csv")
df
```

	Table Code	State Code	District Code	Area Name	Total/ Rural/ Urban	Age group	Worked for 3 months or more but less than 6 months - Persons	Worked for 3 months or more but less than 6 months - Males	Worked for 3 months or more but less than 6 months - Females	Worked for less than 3 months - Persons	...	Industrial Category - N to O - Females	Industrial Category - P to Q - Persons	Industrial Category - P to Q - Males	Industrial Category - P to Q - Females	Industrial Category - R to U - HHI - Persons	Industrial Category - R to U - HHI - Males	Industrial Category - R to U - HHI - Females	Industri Categor R to Non HH Pers
0	B0806SC	`33	`000	State - TAMIL NADU	Total	Total	1200828	589003	611825	221386	...	3565	11080	4019	7061	16833	4266	12567	122
1	B0806SC	`33	`000	State - TAMIL NADU	Total	`5-14	27791	14125	13666	2447	...	11	122	71	51	427	169	258	19
2	B0806SC	`33	`000	State - TAMIL	Total	15-34	514340	259560	254780	92423		1754	7536	2718	4818	8346	2127	6219	68



2	B0806SC	`33	`000	State - TAMIL NADU	Total	15-34	514340	259560	254780	92423	...	1754	7536	2718	4818	8346	2127	6219	68
3	B0806SC	`33	`000	State - TAMIL NADU	Total	35-59	542581	251957	290624	99202	...	1619	3205	1131	2074	6591	1487	5104	26
4	B0806SC	`33	`000	State - TAMIL NADU	Total	60+	115103	62833	52270	27165	...	175	211	93	118	1457	483	974	7
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
589	B0806SC	`33	`633	District - Tiruppur	Urban	`5-14	272	129	143	18	...	0	0	0	0	0	0	0	
590	B0806SC	`33	`633	District - Tiruppur	Urban	15-34	3285	1654	1631	473	...	20	44	15	29	62	6	56	
591	B0806SC	`33	`633	District - Tiruppur	Urban	35-59	3672	1769	1903	522	...	33	35	12	23	36	9	27	
592	B0806SC	`33	`633	District - Tiruppur	Urban	60+	696	399	297	111	...	0	3	0	3	10	3	7	
593	B0806SC	`33	`633	District - Tiruppur	Urban	Age not stated	2	1	1	0	...	0	0	0	0	0	0	0	

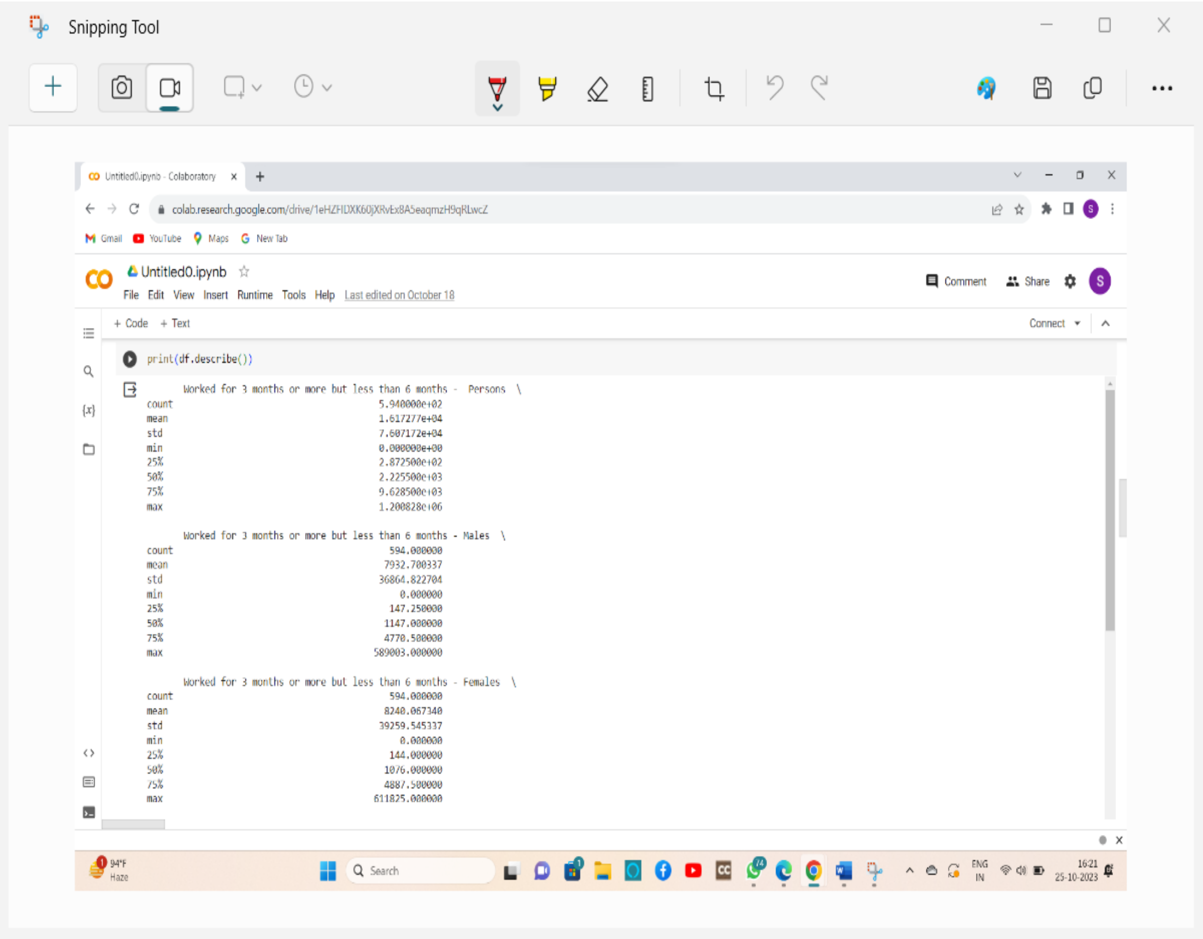
594 rows × 69 columns



Untitled0.ipynb - Colaboratory
colab.research.google.com/drive/1eHZFIDXK60jXRvEx8A5eaqmzH9qRLwcZ#scrollTo=j0VzVYc5xxbu
Gmail
YouTube
Maps
New Tab
Untitled0.ipynb
File
Edit
View
Insert
Runtime
Tools
Help
Last edited on October 18
Comment
Share
+ Code
+ Text
Connect

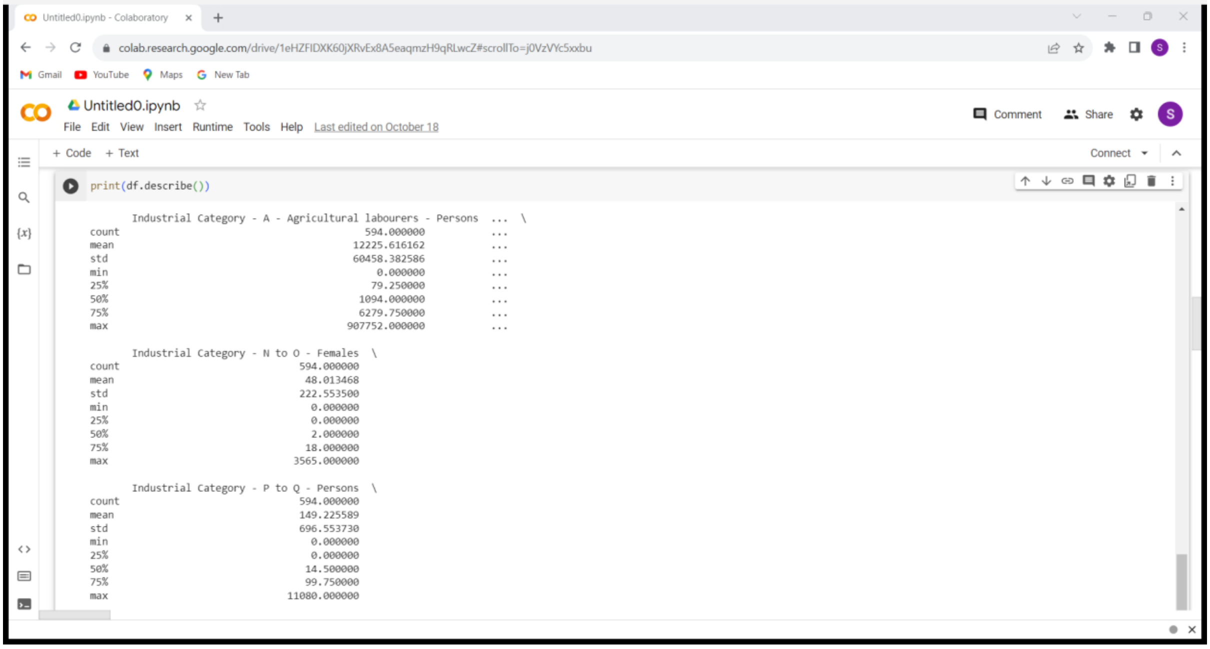
```
print(df.describe())
```

```
       Worked for less than 3 months - Persons  \
count                               594.000000
mean                               2981.629630
std                               13909.621137
min                                   0.000000
25%                                  27.000000
50%                                 430.000000
75%                                1775.250000
max                              221386.000000

       Worked for less than 3 months - Males  \
count                             594.000000
mean                             1338.289562
std                              6127.047670
min                                 0.000000
25%                                14.250000
50%                               198.500000
75%                               774.250000
max                             99368.000000

       Worked for less than 3 months - Females  \
count                               594.000000
mean                               1643.340067
std                                7808.832522
min                                   0.000000
25%                                  13.000000
50%                                 213.000000
75%                                 946.500000
max                              122018.000000
```



```
print(df.describe())
```

```
       Worked for 3 months or more but less than 6 months -  Persons  \
count                                       5.940000e+02
mean                                        1.617277e+04
std                                         7.607172e+04
min                                         0.000000e+00
25%                                         2.872500e+02
50%                                         2.225500e+03
75%                                         9.628500e+03
max                                         1.200828e+06

       Worked for 3 months or more but less than 6 months - Males  \
count                                             594.000000
mean                                             7932.700337
std                                             36864.822704
min                                                 0.000000
25%                                               147.250000
50%                                              1147.000000
75%                                              4770.500000
max                                            589003.000000

       Worked for 3 months or more but less than 6 months - Females  \
count                                               594.000000
mean                                               8240.067340
std                                               39259.545337
min                                                   0.000000
25%                                                 144.000000
50%                                                1076.000000
75%                                                4887.500000
max                                              611825.000000
```

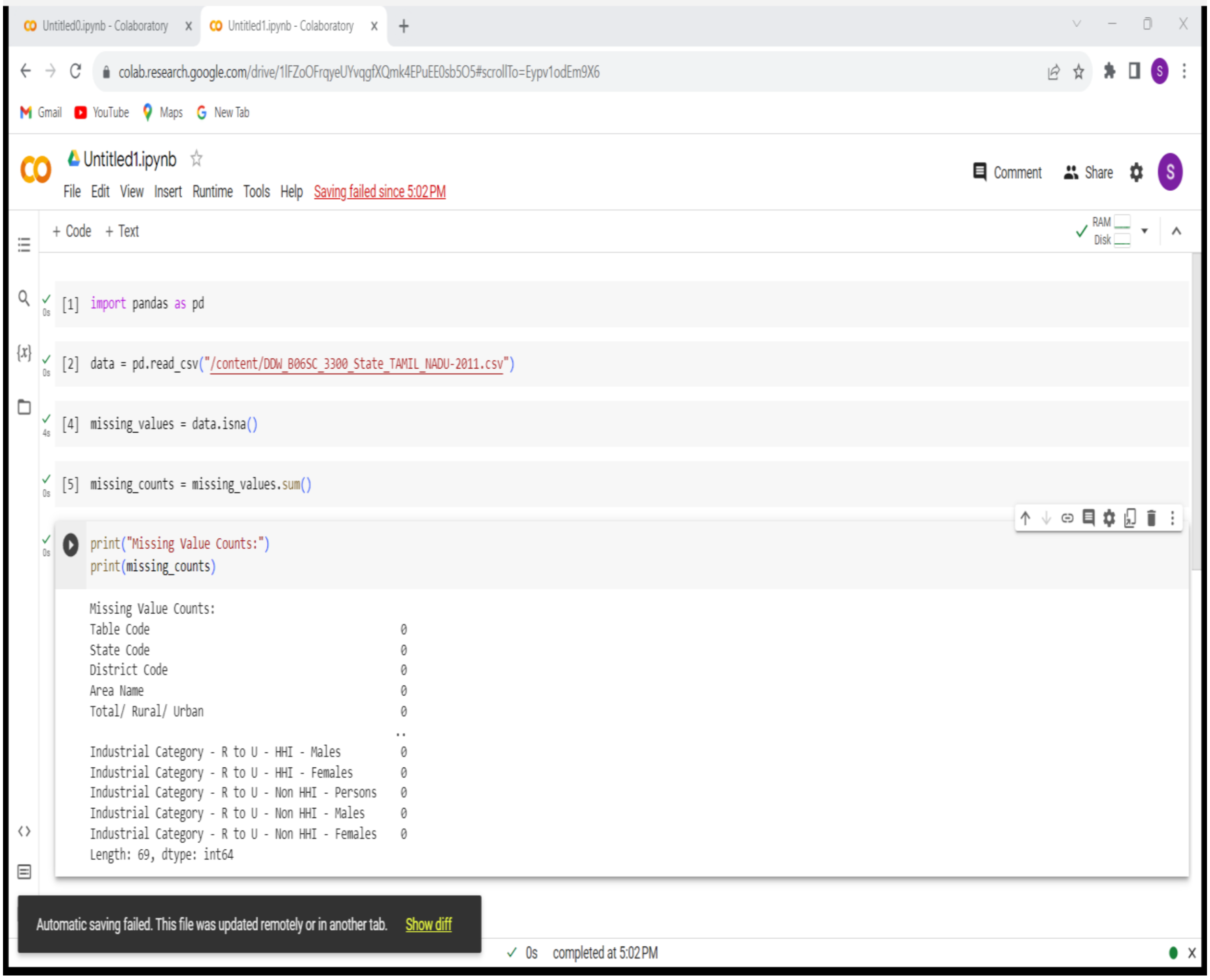


Untitled0.ipynb - Colaboratory
colab.research.google.com/drive/1eHZFIDXK60jXRvEx8A5eaqmzH9qRLwcZ#scrollTo=j0VzVYc5xxbu
Gmail
YouTube
Maps
New Tab
Untitled0.ipynb
File Edit View Insert Runtime Tools Help
Last edited on October 18
Comment
Share
+ Code
+ Text
Connect

```
print(df.describe())
```

```
          Industrial Category - A - Agricultural labourers - Persons  ...  \
count                                           594.000000            ...
mean                                          12225.616162            ...
std                                           60458.382586            ...
min                                               0.000000            ...
25%                                              79.250000            ...
50%                                            1094.000000            ...
75%                                            6279.750000            ...
max                                          907752.000000            ...

          Industrial Category - N to O - Females  \
count                                 594.000000
mean                                   48.013468
std                                   222.553500
min                                     0.000000
25%                                     0.000000
50%                                     2.000000
75%                                    18.000000
max                                  3565.000000

          Industrial Category - P to Q - Persons  \
count                                 594.000000
mean                                  149.225589
std                                   696.553730
min                                     0.000000
25%                                     0.000000
50%                                    14.500000
75%                                    99.750000
max                                 11080.000000
```

HANDLING MISSING VALUES:



Untitled0.ipynb - Colaboratory
Untitled1.ipynb - Colaboratory
colab.research.google.com/drive/1lFZoOFrqyeUYvqgfXQmk4EPuEE0sb5O5#scrollTo=Eypv1odEm9X6
Gmail
YouTube
Maps
New Tab
Untitled1.ipynb
File Edit View Insert Runtime Tools Help
Saving failed since 5:02 PM
Comment
Share
+ Code
+ Text
RAM
Disk

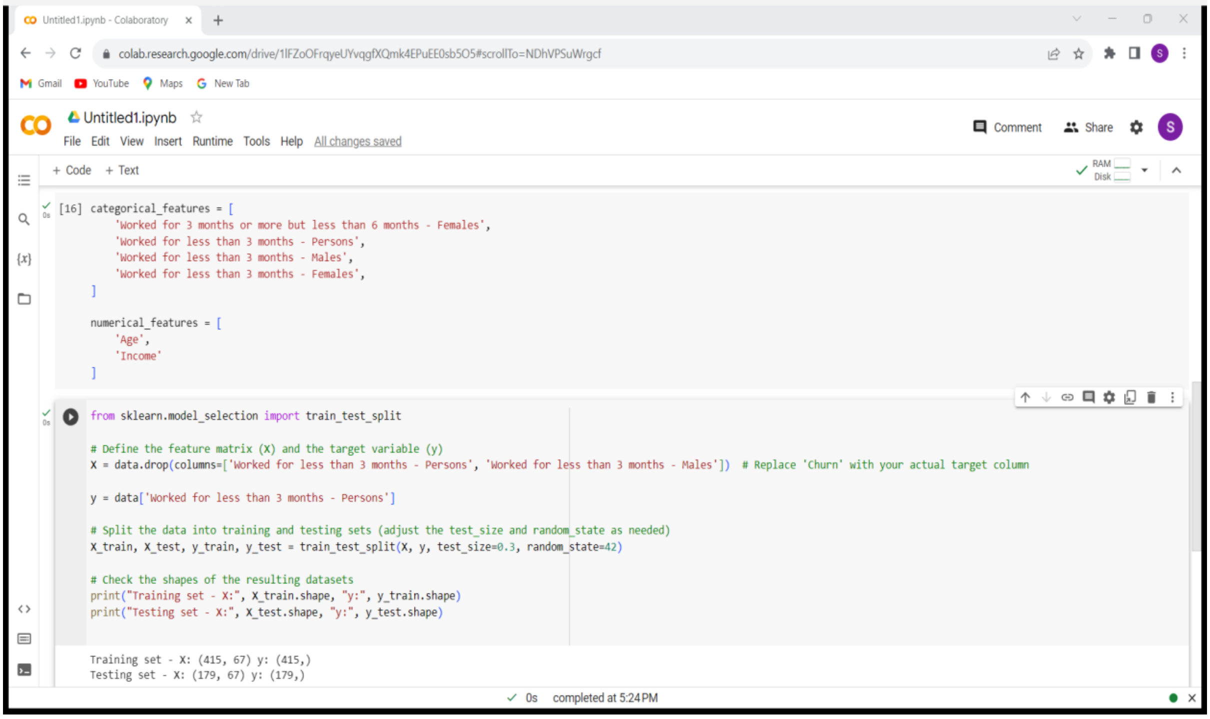
```
[1] import pandas as pd

[2] data = pd.read_csv("/content/DDW_B06SC_3300_State_TAMIL_NADU-2011.csv")

[4] missing_values = data.isna()

[5] missing_counts = missing_values.sum()

print("Missing Value Counts:")
print(missing_counts)
```

```
Missing Value Counts:
Table Code                                          0
State Code                                          0
District Code                                       0
Area Name                                           0
Total/ Rural/ Urban                                 0
                                                   ..
Industrial Category - R to U - HHI - Males          0
Industrial Category - R to U - HHI - Females        0
Industrial Category - R to U - Non HHI - Persons    0
Industrial Category - R to U - Non HHI - Males      0
Industrial Category - R to U - Non HHI - Females    0
Length: 69, dtype: int64
```

Automatic saving failed. This file was updated remotely or in another tab. Show diff

✓ 0s completed at 5:02 PM

DATA SPLITTING:



Untitled1.ipynb - Colaboratory
colab.research.google.com/drive/1lFZoOFrqyeUYvqgfXQmk4EPuEE0sb5O5#scrollTo=NDhVPSuWrgcf
Gmail
YouTube
Maps
New Tab
Untitled1.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk

```
[16] categorical_features = [
         'Worked for 3 months or more but less than 6 months - Females',
         'Worked for less than 3 months - Persons',
         'Worked for less than 3 months - Males',
         'Worked for less than 3 months - Females',
     ]

     numerical_features = [
         'Age',
         'Income'
     ]
```

```
from sklearn.model_selection import train_test_split

# Define the feature matrix (X) and the target variable (y)
X = data.drop(columns=['Worked for less than 3 months - Persons', 'Worked for less than 3 months - Males'])  # Replace 'Churn' with your actual target column

y = data['Worked for less than 3 months - Persons']

# Split the data into training and testing sets (adjust the test_size and random_state as needed)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

# Check the shapes of the resulting datasets
print("Training set - X:", X_train.shape, "y:", y_train.shape)
print("Testing set - X:", X_test.shape, "y:", y_test.shape)
```

```
Training set - X: (415, 67) y: (415,)
Testing set - X: (179, 67) y: (179,)
```