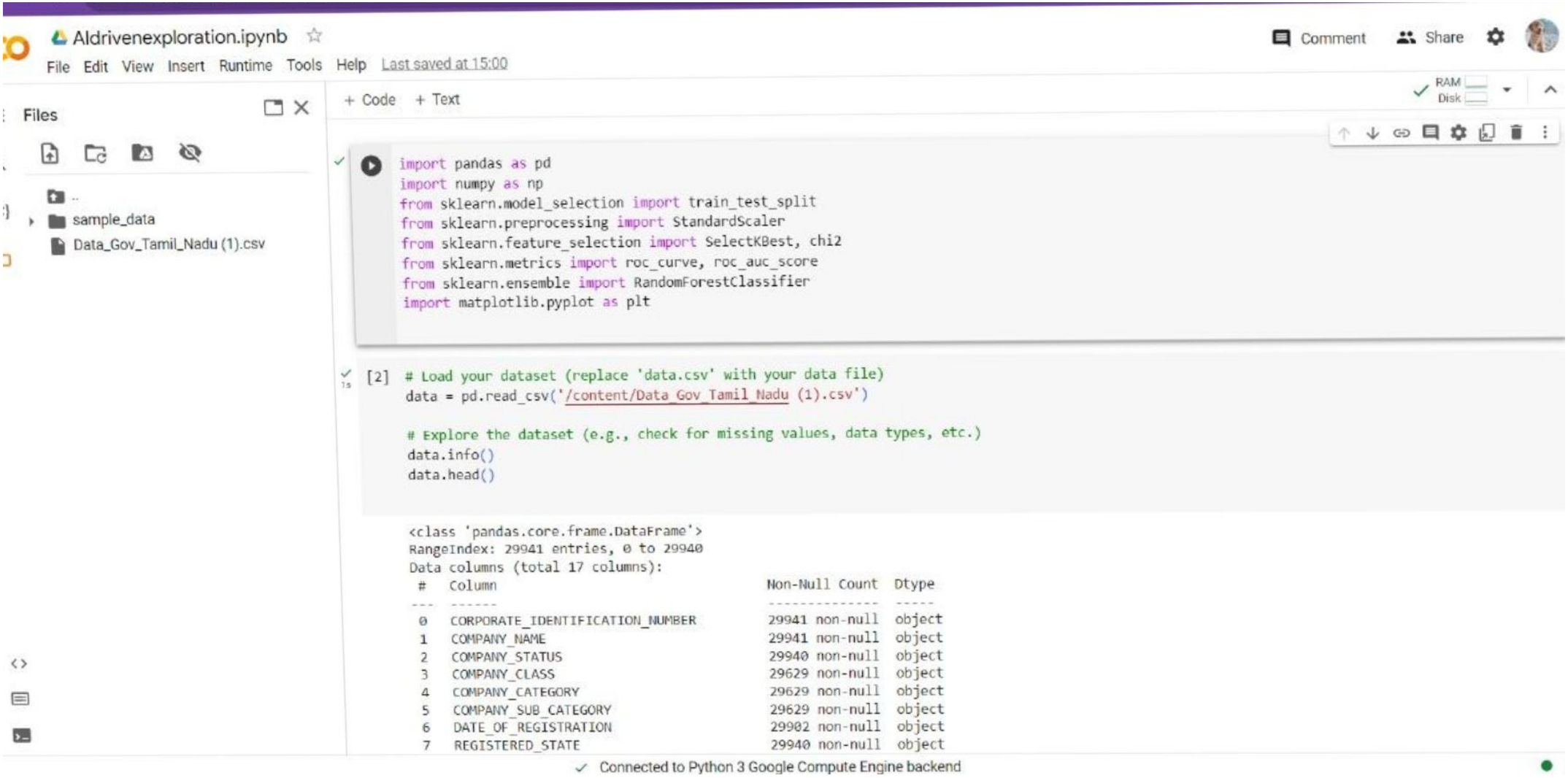
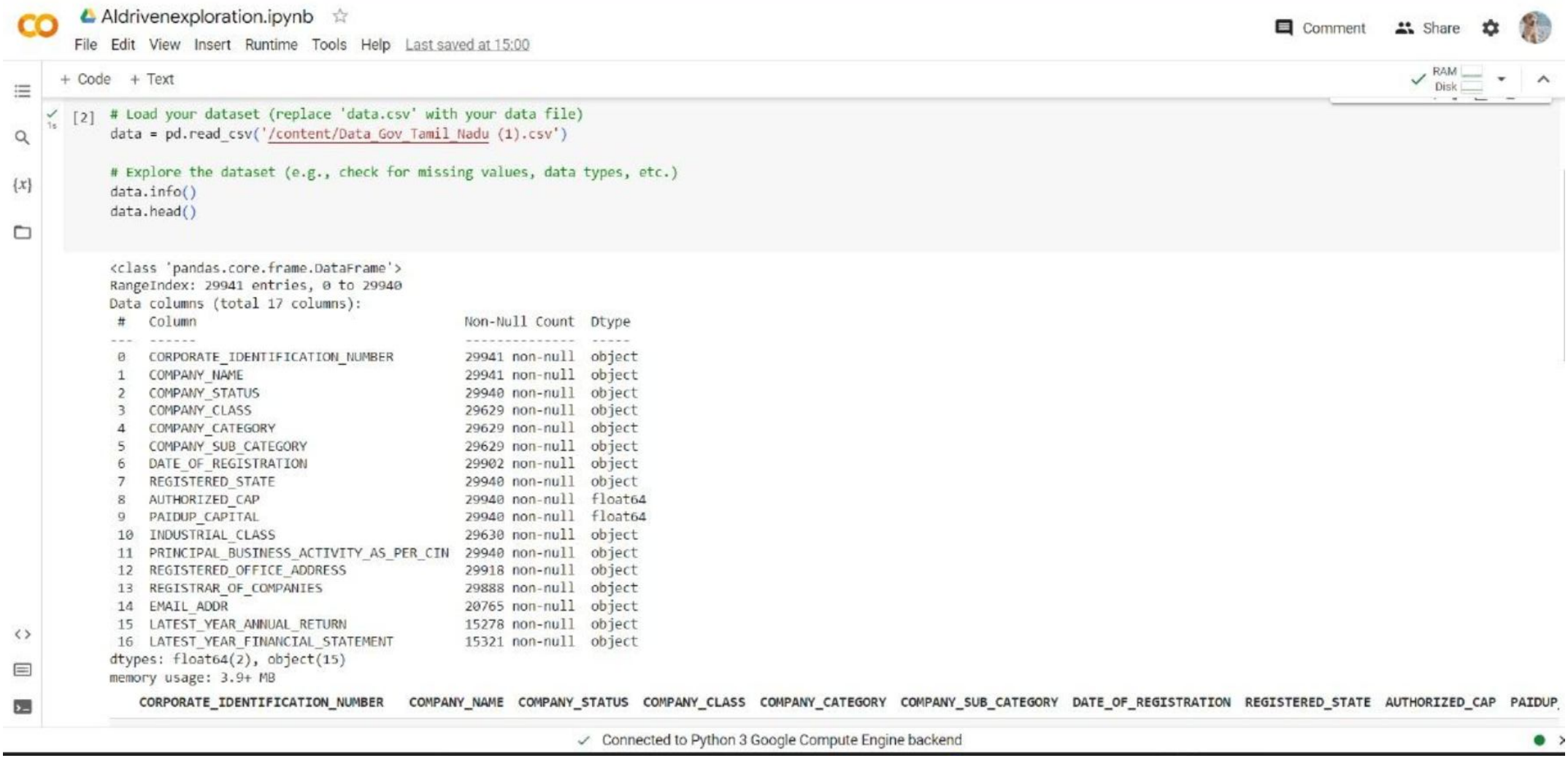
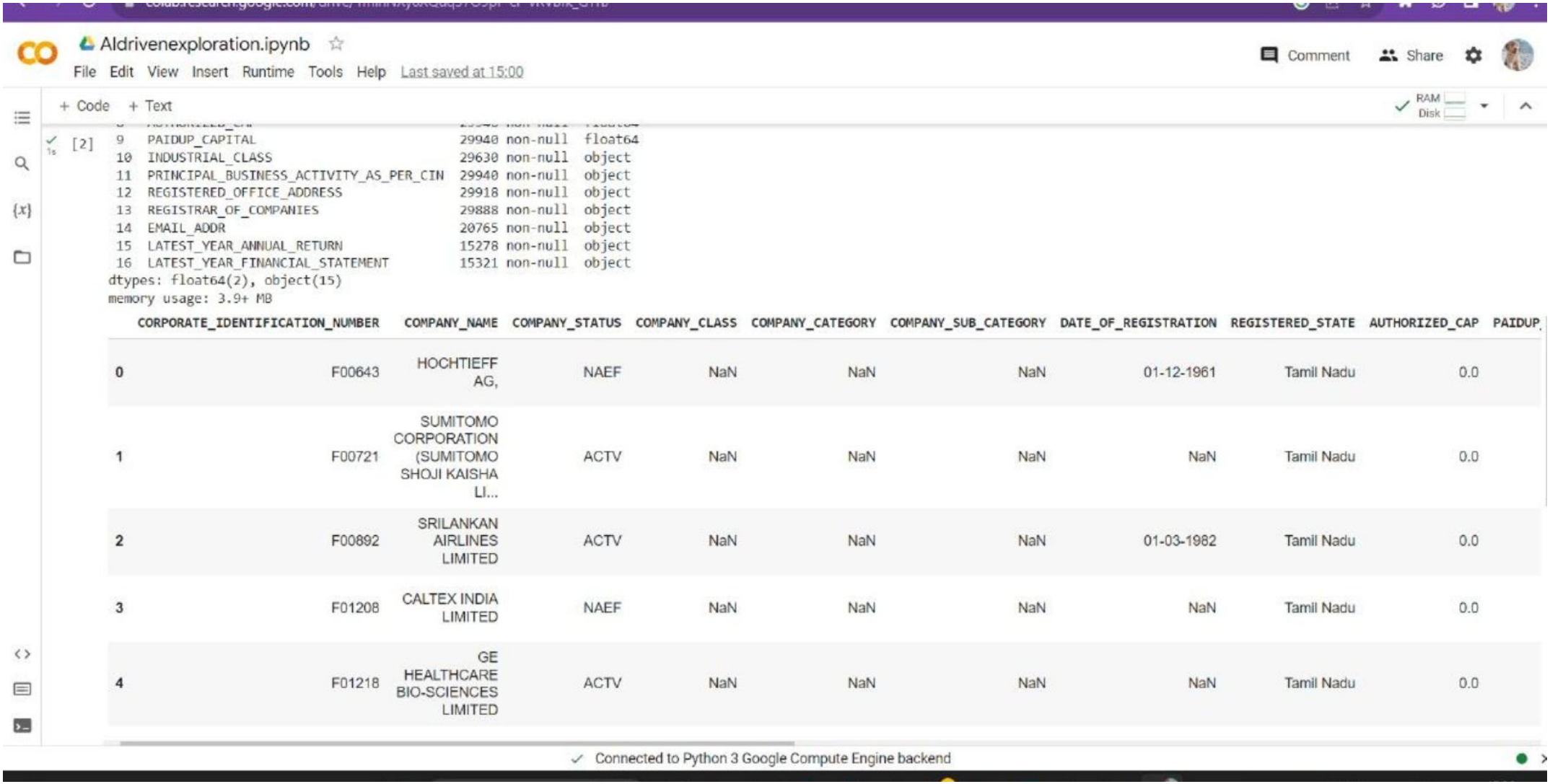
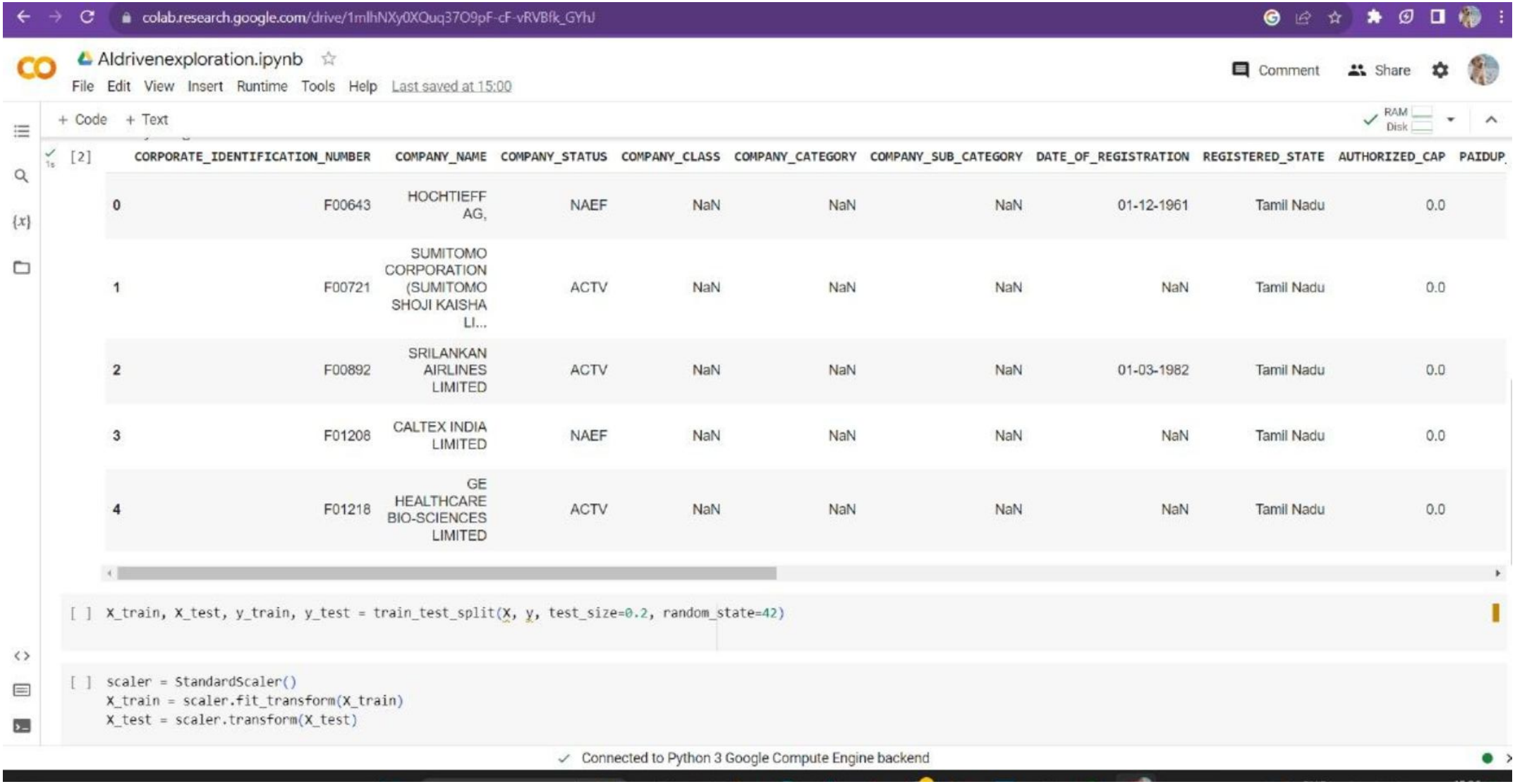

AIdrivenexploration.ipynb
File Edit View Insert Runtime Tools Help Last saved at 15:00
Comment
Share
Files
sample_data
Data_Gov_Tamil_Nadu (1).csv
+ Code + Text
RAM
Disk
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.feature_selection import SelectKBest, chi2
from sklearn.metrics import roc_curve, roc_auc_score
from sklearn.ensemble import RandomForestClassifier
import matplotlib.pyplot as plt
[2]
# Load your dataset (replace 'data.csv' with your data file)
data = pd.read_csv('/content/Data_Gov_Tamil_Nadu (1).csv')
# Explore the dataset (e.g., check for missing values, data types, etc.)
data.info()
data.head()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 29941 entries, 0 to 29940
Data columns (total 17 columns):
Column Non-Null Count Dtype
0 CORPORATE_IDENTIFICATION_NUMBER 29941 non-null object
1 COMPANY_NAME 29941 non-null object
2 COMPANY_STATUS 29940 non-null object
3 COMPANY_CLASS 29629 non-null object
4 COMPANY_CATEGORY 29629 non-null object
5 COMPANY_SUB_CATEGORY 29629 non-null object
6 DATE_OF_REGISTRATION 29902 non-null object
7 REGISTERED_STATE 29940 non-null object
Connected to Python 3 Google Compute Engine backend


AIdrivenexploration.ipynb
File Edit View Insert Runtime Tools Help Last saved at 15:00
Comment
Share
+ Code + Text
RAM
Disk
[2] # Load your dataset (replace 'data.csv' with your data file)
data = pd.read_csv('/content/Data_Gov_Tamil_Nadu (1).csv')
# Explore the dataset (e.g., check for missing values, data types, etc.)
data.info()
data.head()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 29941 entries, 0 to 29940
Data columns (total 17 columns):
Column Non-Null Count Dtype
0 CORPORATE_IDENTIFICATION_NUMBER 29941 non-null object
1 COMPANY_NAME 29941 non-null object
2 COMPANY_STATUS 29940 non-null object
3 COMPANY_CLASS 29629 non-null object
4 COMPANY_CATEGORY 29629 non-null object
5 COMPANY_SUB_CATEGORY 29629 non-null object
6 DATE_OF_REGISTRATION 29902 non-null object
7 REGISTERED_STATE 29940 non-null object
8 AUTHORIZED_CAP 29940 non-null float64
9 PAIDUP_CAPITAL 29940 non-null float64
10 INDUSTRIAL_CLASS 29630 non-null object
11 PRINCIPAL_BUSINESS_ACTIVITY_AS_PER_CIN 29940 non-null object
12 REGISTERED_OFFICE_ADDRESS 29918 non-null object
13 REGISTRAR_OF_COMPANIES 29888 non-null object
14 EMAIL_ADDR 20765 non-null object
15 LATEST_YEAR_ANNUAL_RETURN 15278 non-null object
16 LATEST_YEAR_FINANCIAL_STATEMENT 15321 non-null object
dtypes: float64(2), object(15)
memory usage: 3.9+ MB
CORPORATE_IDENTIFICATION_NUMBER COMPANY_NAME COMPANY_STATUS COMPANY_CLASS COMPANY_CATEGORY COMPANY_SUB_CATEGORY DATE_OF_REGISTRATION REGISTERED_STATE AUTHORIZED_CAP PAIDUP
Connected to Python 3 Google Compute Engine backend



	CORPORATE_IDENTIFICATION_NUMBER	COMPANY_NAME	COMPANY_STATUS	COMPANY_CLASS	COMPANY_CATEGORY	COMPANY_SUB_CATEGORY	DATE_OF_REGISTRATION	REGISTERED_STATE	AUTHORIZED_CAP	PAIDUP
0	F00643	HOCHTIEFF AG,	NAEF	NaN	NaN	NaN	01-12-1961	Tamil Nadu	0.0	
1	F00721	SUMITOMO CORPORATION (SUMITOMO SHOJI KAISHA LI...	ACTV	NaN	NaN	NaN	NaN	Tamil Nadu	0.0	
2	F00892	SRILANKAN AIRLINES LIMITED	ACTV	NaN	NaN	NaN	01-03-1982	Tamil Nadu	0.0	
3	F01208	CALTEX INDIA LIMITED	NAEF	NaN	NaN	NaN	NaN	Tamil Nadu	0.0	
4	F01218	GE HEALTHCARE BIO-SCIENCES LIMITED	ACTV	NaN	NaN	NaN	NaN	Tamil Nadu	0.0	


colab.research.google.com/drive/1mlhNXy0XQuq37O9pF-cF-vRVBfk_GYhJ
Aldrivenexploration.ipynb
File Edit View Insert Runtime Tools Help Last saved at 15:00
Comment
Share
+ Code + Text
RAM
Disk
[2]
CORPORATE_IDENTIFICATION_NUMBER COMPANY_NAME COMPANY_STATUS COMPANY_CLASS COMPANY_CATEGORY COMPANY_SUB_CATEGORY DATE_OF_REGISTRATION REGISTERED_STATE AUTHORIZED_CAP PAIDUP
0 F00643 HOCHTIEFF AG, NAEF NaN NaN NaN 01-12-1961 Tamil Nadu 0.0
1 F00721 SUMITOMO CORPORATION (SUMITOMO SHOJI KAISHA LI... ACTV NaN NaN NaN NaN Tamil Nadu 0.0
2 F00892 SRILANKAN AIRLINES LIMITED ACTV NaN NaN NaN 01-03-1982 Tamil Nadu 0.0
3 F01208 CALTEX INDIA LIMITED NAEF NaN NaN NaN NaN Tamil Nadu 0.0
4 F01218 GE HEALTHCARE BIO-SCIENCES LIMITED ACTV NaN NaN NaN NaN Tamil Nadu 0.0
[ ] X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
[ ] scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
Connected to Python 3 Google Compute Engine backend

File Edit View Insert Runtime Tools Help Last saved at 15:00

Comment Share

+ Code + Text

RAM Disk

..
sample_data
Data_Gov_Tamil_Nadu (1).csv

80.89 GB available

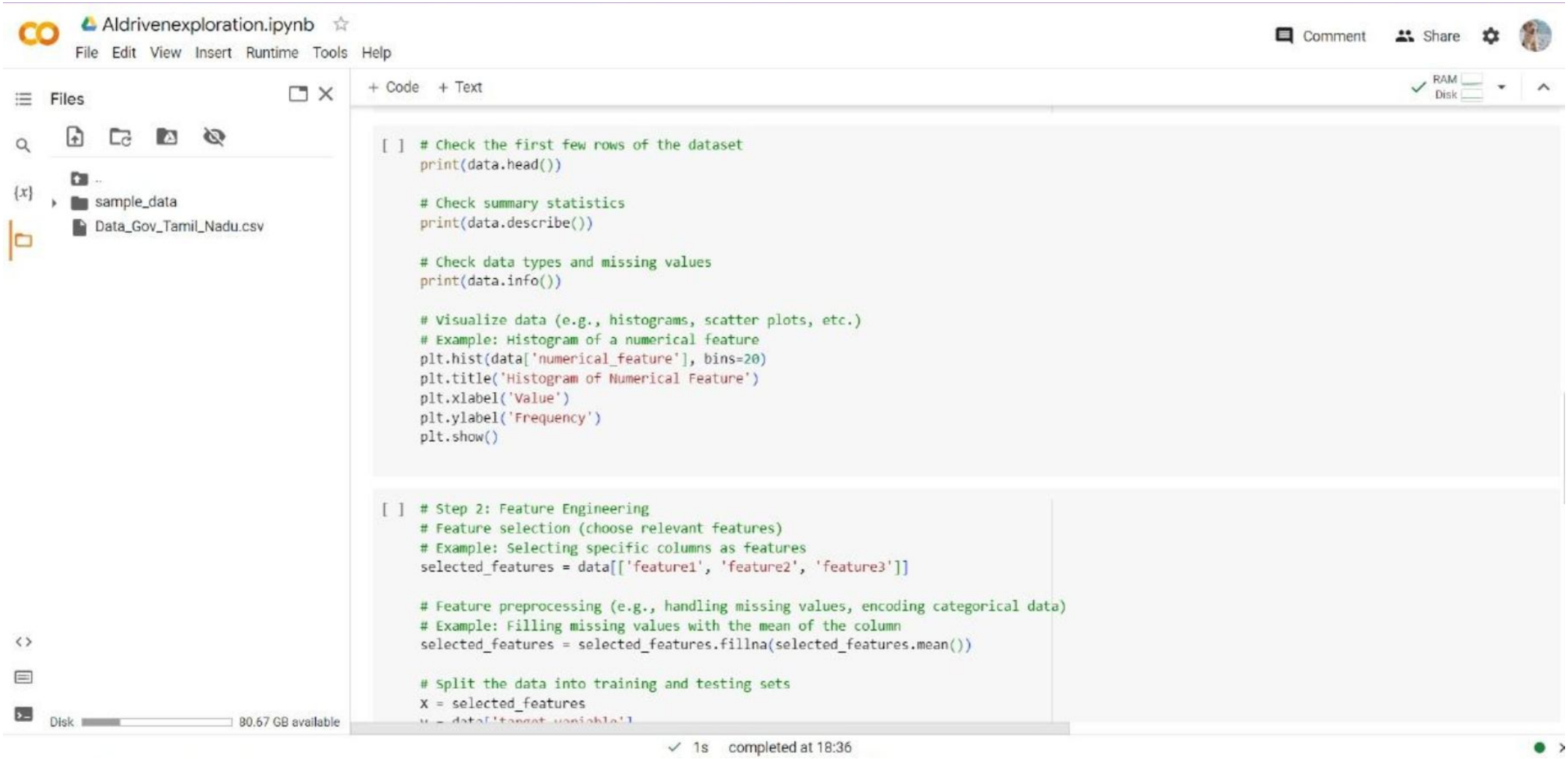
[2]

2	F00892	SRILANKAN AIRLINES LIMITED	ACTV	NaN	NaN	NaN	01-03-1982
3	F01208	CALTEX INDIA LIMITED	NAEF	NaN	NaN	NaN	NaN
4	F01218	GE HEALTHCARE BIO-SCIENCES LIMITED	ACTV	NaN	NaN	NaN	NaN

```
[ ] X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
[ ] scaler = StandardScaler()
    X_train = scaler.fit_transform(X_train)
    X_test = scaler.transform(X_test)
```

```
[ ] # Select the top k features using chi-squared statistic (replace k with your desired number)
    selector = SelectKBest(score_func=chi2, k=10)
    X_train = selector.fit_transform(X_train, y_train)
    X_test = selector.transform(X_test)
```


AIdrivenexploration.ipynb
File Edit View Insert Runtime Tools Help
Comment
Share
Files
+ Code + Text
RAM
Disk
..
sample_data
Data_Gov_Tamil_Nadu.csv
[ ] # Check the first few rows of the dataset
print(data.head())
# Check summary statistics
print(data.describe())
# Check data types and missing values
print(data.info())
# Visualize data (e.g., histograms, scatter plots, etc.)
# Example: Histogram of a numerical feature
plt.hist(data['numerical_feature'], bins=20)
plt.title('Histogram of Numerical Feature')
plt.xlabel('Value')
plt.ylabel('Frequency')
plt.show()
[ ] # Step 2: Feature Engineering
# Feature selection (choose relevant features)
# Example: Selecting specific columns as features
selected_features = data[['feature1', 'feature2', 'feature3']]
# Feature preprocessing (e.g., handling missing values, encoding categorical data)
# Example: Filling missing values with the mean of the column
selected_features = selected_features.fillna(selected_features.mean())
# Split the data into training and testing sets
X = selected_features
Disk 80.67 GB available
1s completed at 18:36

Aldrivenexploration.ipynb

File Edit View Insert Runtime Tools Help

Comment Share

+ Code + Text

RAM Disk

Files

..

sample_data

Data_Gov_Tamil_Nadu.csv

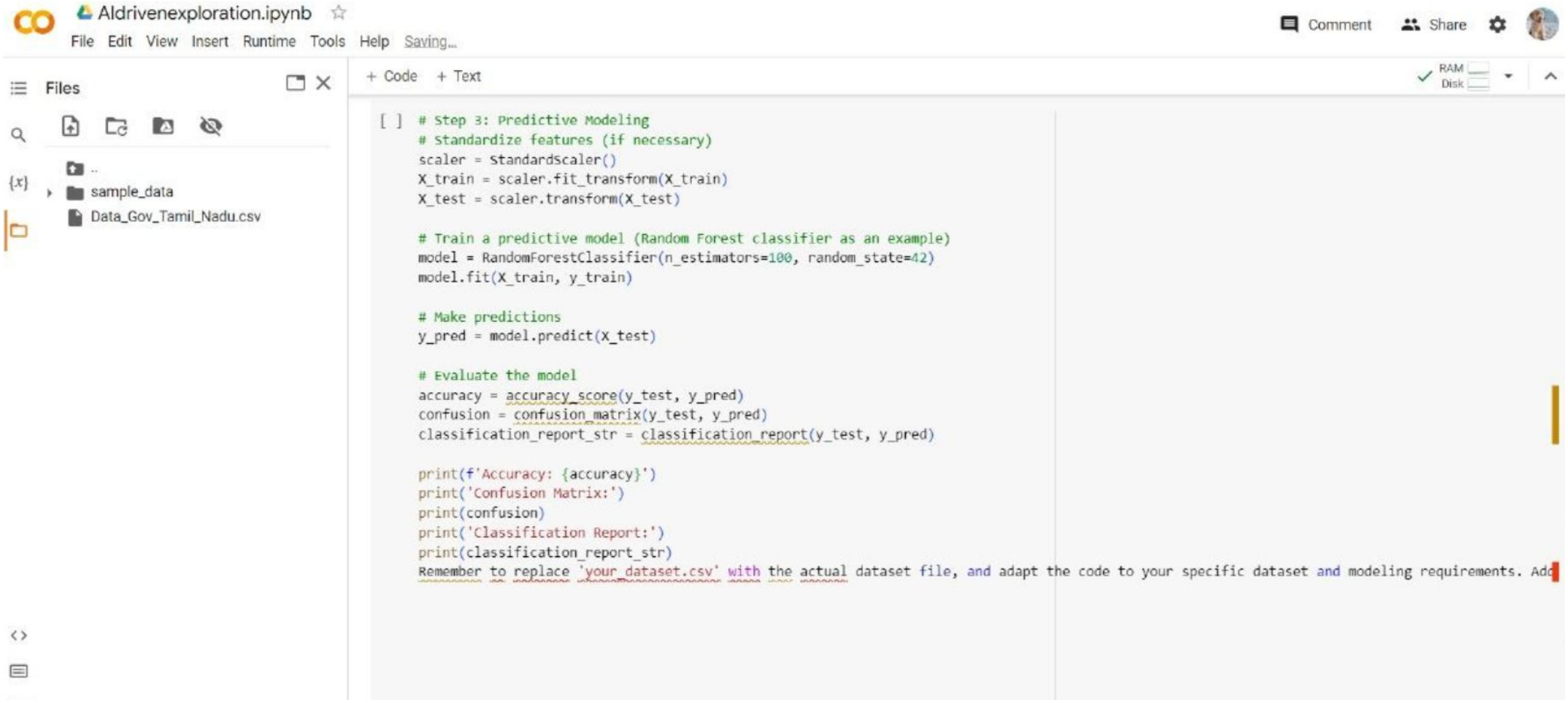
Disk 80.67 GB available

```
plt.title('Histogram of Numerical Feature')
plt.xlabel('Value')
plt.ylabel('Frequency')
plt.show()
```

```
# Step 2: Feature Engineering
# Feature selection (choose relevant features)
# Example: Selecting specific columns as features
selected_features = data[['feature1', 'feature2', 'feature3']]

# Feature preprocessing (e.g., handling missing values, encoding categorical data)
# Example: Filling missing values with the mean of the column
selected_features = selected_features.fillna(selected_features.mean())

# Split the data into training and testing sets
X = selected_features
y = data['target_variable']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
# Step 3: Predictive Modeling
# Standardize features (if necessary)
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

# Train a predictive model (Random Forest classifier as an example)
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)
```

✓ 1s completed at 18:36


Aldrivenexploration.ipynb
File Edit View Insert Runtime Tools Help Saving...
Comment
Share
Files
+ Code + Text
RAM
Disk
..
sample_data
Data_Gov_Tamil_Nadu.csv
[ ]
# Step 3: Predictive Modeling
# Standardize features (if necessary)
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
# Train a predictive model (Random Forest classifier as an example)
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)
# Make predictions
y_pred = model.predict(X_test)
# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
confusion = confusion_matrix(y_test, y_pred)
classification_report_str = classification_report(y_test, y_pred)
print(f'Accuracy: {accuracy}')
print('Confusion Matrix:')
print(confusion)
print('Classification Report:')
print(classification_report_str)
Remember to replace 'your_dataset.csv' with the actual dataset file, and adapt the code to your specific dataset and modeling requirements. Add