Phase 3 – Development Part 1 Document submission

Project Title :

AI – Driven Exploration and prediction

Phase 3 Topic :

Read (loading) and Preprocessing given dataset

By

H.KISHORE KANNAN

922121106034

au922121106034

SSMIET

Introduction

AI-Driven Exploration and Prediction of Company Registration Trends with the Registrar of Companies (RoC) involves leveraging artificial intelligence (AI) methodologies to analyze data related to company registrations maintained by the Registrar of Companies. The Registrar of Companies is an authoritative entity responsible for overseeing and maintaining the registry of companies within a specific jurisdiction.

By employing AI algorithms, this approach aims to extract valuable insights and forecast patterns from the data compiled by the RoC. These insights can aid in understanding trends, emerging patterns, and other significant aspects of company registrations, empowering stakeholders to make informed decisions in the business landscape.

The utilization of AI in this domain encompasses data collection, processing, exploratory data analysis, machine learning modeling, and predictive analytics to anticipate future trends in company registrations. Ultimately, this AI-driven approach enables proactive decision-making and strategic planning based on comprehensive analyses of registration trends and associated data.

Overview

For Phase 3

1.Data collecting

2.Data Preprocessing

Analyzing Data

Statistics Summary

3.Exploratory Data Analysis (EDA)

Univariate Analysis

Bivariate Analysis

Multivariate Analysis

Data Collecting

AI-Driven Exploration and Prediction of Company Registration Trends with the Registrar of Companies (RoC), the process of collecting data involves gathering relevant information from given sources to create a comprehensive dataset for analysis and modeling

Given Data

Import Python library

The first step involved in ML using python is understanding and playing around with our data using libraries

Import all libraries which are required for our analysis, such as Data Loading, Statistical analysis, Visualizations, Data Transformations, Merge and Joins, etc.

Pandas and Numpy have been used for Data Manipulation and numerical Calculations

Matplotlib and Seaborn have been used for Data visualizations.

Program :

```
import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

(Optional)

# to ignore warnings

import warnings

warnings.filterwarnings('ignore')
```

Reading Dataset

The Pandas library offers a wide range of possibilities for loading data into the pandas DataFrame from files like JSON, .csv, .xlsx, .sql, .pickle, .html, .txt, images etc.

Given data are available in a tabular format of CSV files. It is trendy and easy to access. Using the read_csv() function, data can be converted to a pandas DataFrame.

We have stored the data in the DataFrame data.

Program

```
data=pd.read_csv("Data_Gov_Tamil_Nadu.csv",encoding='latin-1')

df
```

	CORPORATE_IDENTIFICATION_NUMBER	COMPANY_NAME	COMPANY_STATUS	COMPANY_CLASS	COMPANY_CATEGORY	COMPANY_SUB_CATEGORY
0	F00643	HOCHTIEFF AG.	NAEF	NaN	NaN	NaN
1	F00721	SUMITOMO CORPORATION (SUMITOMO SHOJI KAISHA LI...	ACTV	NaN	NaN	NaN
2	F00892	SRILANKAN AIRLINES LIMITED	ACTV	NaN	NaN	NaN
3	F01208	CALTEX INDIA LIMITED	NAEF	NaN	NaN	NaN
4	F01218	GE HEALTHCARE BIO-SCIENCES LIMITED	ACTV	NaN	NaN	NaN
...	...	...	...	...	...	...
150866	U74997TN2016PTC112556	QUAD42 MEDIA PRIVATE LIMITED	ACTV	Private	Company limited by Shares	Non-govt company
150867	U74997TN2018PTC121491	IYERAATHU FOODS PRIVATE LIMITED	ACTV	Private	Company limited by Shares	Non-govt company
150868	U74997TZ2016PTC027802	POLYGAR FARM SOLUTIONS PRIVATE LIMITED	STOF	Private	Company limited by Shares	Non-govt company
150869	U74997TZ2018PTC030177	PANDIYA AGRI SOLUTIONS PRIVATE LIMITED	ACTV	Private	Company limited by Shares	Non-govt company
150870	U74997TZ2019PTC032491	NROOT TECHNOLOGIES PRIVATE LIMITED	ACTV	Private	Company limited by Shares	Non-govt company

150871 rows × 17 columns

Analyzing the Data

`# head()` will display the top 5 observations of the dataset

```
df.head()
```

	CORPORATE_IDENTIFICATION_NUMBER	COMPANY_NAME	COMPANY_STATUS	COMPANY_CLASS	COMPANY_CATEGORY	COMPANY_SUB_CATEGORY	DATE
0	F00643	HOCHTIEFF AG.	NAEF	NaN	NaN	NaN	NaN
1	F00721	SUMITOMO CORPORATION (SUMITOMO SHOJI KAISHA LI...	ACTV	NaN	NaN	NaN	NaN
2	F00892	SRILANKAN AIRLINES LIMITED	ACTV	NaN	NaN	NaN	NaN
3	F01208	CALTEX INDIA LIMITED	NAEF	NaN	NaN	NaN	NaN
4	F01218	GE HEALTHCARE BIO-SCIENCES LIMITED	ACTV	NaN	NaN	NaN	NaN

tail() will display the last 5 observations of the dataset

df.tail()

In [18]: df.tail()

Out[18]:

	CORPORATE_IDENTIFICATION_NUMBER	COMPANY_NAME	COMPANY_STATUS	COMPANY_CLASS	COMPANY_CATEGORY	COMPANY_SUB_CATEGORY
150866	U74997TN2016PTC112556	QUAD42 MEDIA PRIVATE LIMITED	ACTV	Private	Company limited by Shares	Non-govt company
150867	U74997TN2018PTC121491	IYERAATHU FOODS PRIVATE LIMITED	ACTV	Private	Company limited by Shares	Non-govt company
150868	U74997TZ2016PTC027802	POLYGAR FARM SOLUTIONS PRIVATE LIMITED	STOF	Private	Company limited by Shares	Non-govt company
150869	U74997TZ2018PTC030177	PANDIYA AGRI SOLUTIONS PRIVATE LIMITED	ACTV	Private	Company limited by Shares	Non-govt company
150870	U74997TZ2019PTC032491	NROOT TECHNOLOGIES PRIVATE LIMITED	ACTV	Private	Company limited by Shares	Non-govt company

info() helps to understand the data type and information about data, including the number of records in each column, data having null or not null, Data type, the memory usage of the dataset

df.info()

In [17]: df.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150871 entries, 0 to 150870
Data columns (total 17 columns):
 #   Column                                   Non-Null Count    Dtype
---  ------                                   --------------    -----
 0   CORPORATE_IDENTIFICATION_NUMBER          150871 non-null   object
 1   COMPANY_NAME                             150871 non-null   object
 2   COMPANY_STATUS                           150871 non-null   object
 3   COMPANY_CLASS                            150537 non-null   object
 4   COMPANY_CATEGORY                         150537 non-null   object
 5   COMPANY_SUB_CATEGORY                     150537 non-null   object
 6   DATE_OF_REGISTRATION                     150832 non-null   object
 7   REGISTERED_STATE                         150871 non-null   object
 8   AUTHORIZED_CAP                           150871 non-null   float64
 9   PAIDUP_CAPITAL                           150871 non-null   float64
 10  INDUSTRIAL_CLASS                         150561 non-null   object
 11  PRINCIPAL_BUSINESS_ACTIVITY_AS_PER_CIN   150871 non-null   object
 12  REGISTERED_OFFICE_ADDRESS                150781 non-null   object
 13  REGISTRAR_OF_COMPANIES                   150697 non-null   object
 14  EMAIL_ADDR                               112742 non-null   object
 15  LATEST_YEAR_ANNUAL_RETURN                74982 non-null    object
 16  LATEST_YEAR_FINANCIAL_STATEMENT          75089 non-null    object
dtypes: float64(2), object(15)
memory usage: 19.6+ MB
```

Check for Duplication

\# nunique() based on several unique values in each column and the data description, we can identify the continuous and categorical columns in the data. Duplicated data can be handled or removed based on further analysis

df.nunique()

```
In [19]: df.nunique()

Out[19]: CORPORATE_IDENTIFICATION_NUMBER          150871
         COMPANY_NAME                             150560
         COMPANY_STATUS                               11
         COMPANY_CLASS                                 3
         COMPANY_CATEGORY                              3
         COMPANY_SUB_CATEGORY                          5
         DATE_OF_REGISTRATION                      13540
         REGISTERED_STATE                              1
         AUTHORIZED_CAP                             1623
         PAIDUP_CAPITAL                            16294
         INDUSTRIAL_CLASS                           1562
         PRINCIPAL_BUSINESS_ACTIVITY_AS_PER_CIN       17
         REGISTERED_OFFICE_ADDRESS                142910
         REGISTRAR_OF_COMPANIES                        4
         EMAIL_ADDR                                79940
         LATEST_YEAR_ANNUAL_RETURN                   169
         LATEST_YEAR_FINANCIAL_STATEMENT             138
         dtype: int64
```

Missing Values Calculation

\# **isnull()** is widely been in all pre-processing steps to identify null values in the data

\# **data.isnull().sum()** is used to get the number of missing records in each column

df.isnull().sum()

```
In [20]: df.isnull().sum()

Out[20]: CORPORATE_IDENTIFICATION_NUMBER              0
         COMPANY_NAME                                 0
         COMPANY_STATUS                               0
         COMPANY_CLASS                              334
         COMPANY_CATEGORY                           334
         COMPANY_SUB_CATEGORY                       334
         DATE_OF_REGISTRATION                        39
         REGISTERED_STATE                             0
         AUTHORIZED_CAP                               0
         PAIDUP_CAPITAL                               0
         INDUSTRIAL_CLASS                           310
         PRINCIPAL_BUSINESS_ACTIVITY_AS_PER_CIN       0
         REGISTERED_OFFICE_ADDRESS                   90
         REGISTRAR_OF_COMPANIES                     174
         EMAIL_ADDR                               38129
         LATEST_YEAR_ANNUAL_RETURN                75889
         LATEST_YEAR_FINANCIAL_STATEMENT          75782
         dtype: int64
```

Statistics Summary

```
describe() function gives all statistics summary of data
```

```
In [4]: df.describe().T
Out[4]:
```

	count	mean	std	min	25%	50%	75%	max
AUTHORIZED_CAP	150871.0	3.522781e+07	1.408554e+09	0.0	100000.0	800000.0	2000000.0	3.000000e+11
PAIDUP_CAPITAL	150871.0	2.328824e+07	1.072458e+09	0.0	100000.0	100000.0	685745.0	2.461235e+11

#describe()- Provide a statistics summary of data belonging to
numerical datatype such as int, float Can include Count, Mean,
Standard Deviation, median, mode, minimum value, maximum value, range,
standard deviation, etc.

Exploratory Data Analysis

Exploratory Data Analysis refers to the crucial process of
performing initial investigations on data to discover
patterns to check assumptions with the help of summary
statistics and graphical representations.

EDA can be leveraged to check for outliers, patterns, and
trends in the given data.

EDA helps to find meaningful patterns in data.

EDA provides in-depth insights into the data sets to solve
our business problems.

EDA gives a clue to impute missing values in the dataset

EDA Univariate Analysis

Analyzing the dataset by taking one variable at a time

Program :

```
# Select the specified columns for analysis

columns_for_analysis = ['CORPORATE_IDENTIFICATION_NUMBER',
'COMPANY_NAME', 'COMPANY_STATUS','COMPANY_CLASS',
'COMPANY_CATEGORY','COMPANY_SUB_CATEGORY','DATE_OF_REGISTRATION','REGI
STERED_STATE','AUTHORIZED_CAP','PAIDUP_CAPITAL','INDUSTRIAL_CLASS','PR
INCIPAL_BUSINESS_ACTIVITY_AS_PER_CIN','REGISTERED_OFFICE_ADDRESS','REG
ISTRAR_OF_COMPANIES','EMAIL_ADDR','LATEST_YEAR_ANNUAL_RETURN','LATEST_
YEAR_FINANCIAL_STATEMENT']


# Subset the DataFrame with the selected columns

selected_df = df[columns_for_analysis]


# Display basic statistical summaries for numerical columns

print(selected_df.describe())


# Univariate analysis for categorical columns

for col in selected_df.select_dtypes(include='object'):

    print(f'\n{col} Value
Counts:\n{selected_df[col].value_counts()}\n')
```

OUTPUT :

	AUTHORIZED_CAP	PAIDUP_CAPITAL
count	1.508710e+05	1.508710e+05
mean	3.522781e+07	2.328824e+07
std	1.408554e+09	1.072458e+09
min	0.000000e+00	0.000000e+00
25%	1.000000e+05	1.000000e+05
50%	8.000000e+05	1.000000e+05
75%	2.000000e+06	6.857450e+05
max	3.000000e+11	2.461235e+11

CORPORATE_IDENTIFICATION_NUMBER Value Counts:

CORPORATE_IDENTIFICATION_NUMBER

F00643   1

U72900TN2008PTC067545   1

U72900TN2008PTC067391   1

U72900TN2008PTC067393   1

U72900TN2008PTC067405   1

                   ..

U93090TZ2010PTC016187   1

U93090TZ2011PTC017199   1

U93090TZ2014PTC020864   1

U93090TZ2016NPL027599   1

U74997TZ2019PTC032491   1

Name: count, Length: 150871, dtype: int64

COMPANY_NAME Value Counts:

COMPANY_NAME

PATSEN BIOTEC PRIVATE LIMITED   3

PEARL PLANTATIONS PRIVATE LIMITED   3

SUPER ANALYSERS PRIVATE LIMITED   3

SRI VISHNU MARKETING PRIVATE LIMITED   3

TITAN WIRES PRIVATE LIMITED   3

                       ..

YARYA SEKUR MARK PRIVATE LIMITED   1

ASSORT ENTERPRISES PRIVATE LIMITED   1

JUVAGO PRIVATE LIMITED   1

VGROW FACILITY SERVICES PRIVATE LIMITED   1

NROOT TECHNOLOGIES PRIVATE LIMITED   1

Name: count, Length: 150560, dtype: int64

COMPANY_STATUS Value Counts:

COMPANY_STATUS

ACTV   78689

STOF   64058

UPSO   3531

AMAL   1635

DISD   851

NAEF   732

ULQD   408

LIQD   389

CLLP   291

D455   164

CLLD   123

Name: count, dtype: int64

COMPANY_CLASS Value Counts:

COMPANY_CLASS

Private   137173

Public   11237

Private(One Person Company)   2127

Name: count, dtype: int64

COMPANY_CATEGORY Value Counts:

COMPANY_CATEGORY

Company limited by Shares   149924

Company Limited by Guarantee   598

Unlimited Company   15

Name: count, dtype: int64

COMPANY_SUB_CATEGORY Value Counts:

COMPANY_SUB_CATEGORY

Non-govt company   149181

Subsidiary of Foreign Company   1083

Guarantee and Association comp   140

State Govt company   109

Union Govt company   24

Name: count, dtype: int64

DATE_OF_REGISTRATION Value Counts:

DATE_OF_REGISTRATION

01-04-1956   190

20-09-2018   144

26-03-2019   91

26-02-2016   73

24-03-2016   71

      ...

23-09-1967   1

27-05-1968   1

07-02-1968   1

15-04-1968   1

06-05-2006   1

Name: count, Length: 13540, dtype: int64

REGISTERED_STATE Value Counts:

REGISTERED_STATE

Tamil Nadu   150871

Name: count, dtype: int64

INDUSTRIAL_CLASS Value Counts:

INDUSTRIAL_CLASS

74999   14809

72900   8121

72200   6093

74900	5232
65991	3934
...	
17254	1
15315	1
31504	1
34209	1
24130	1

Name: count, Length: 1562, dtype: int64

PRINCIPAL_BUSINESS_ACTIVITY_AS_PER_CIN Value Counts:

PRINCIPAL_BUSINESS_ACTIVITY_AS_PER_CIN

Real estate renting and business activities	48697
Manufacturing	35757
Financial intermediation	13772
Wholesale and retail trade repair of motor vehicles motorcycles and personal and household goods	13681
Construction	9079
Agriculture & allied	7496
Transport storage and communications	6231
Other community social and personal service activities	4725
Hotels and restaurants	2673
Electricity gas and water supply	2459
Health and social work	2270
Education	1822
Mining and quarrying	1377
Extraterritorial organizations and bodies	781
Public administration and defence compulsory social security	27
Activities of private households as employers and undifferentiated production activities of private households	19
Unclassified	5

Name: count, dtype: int64

REGISTERED_OFFICE_ADDRESS Value Counts:

REGISTERED_OFFICE_ADDRESS

MADRAS   211

Sri sai subhodhaya ApartmentsNo.57/2B, East Coast Road, Thiruvanmiyur   58

Flat No 6J, Century Plaza, 560-562, Anna Salai,Teynampet   54

Times Partner No: 58Perambur Barracks Road   45

"R R LANDMARK"NO.1E-1 NAVA INDIA ROAD   44

                                    ...

NO.47, SOUTH REDDY STREET,ATHIPET, AMBATTUR   1

FLAT NO.10, SRI NARAYANA FLATS25, TILAK STREET, T.NAGAR   1

Plot No.52Sidco Industrial Estate,Alathur   1

22/160-AThengapattanam Road   1

139/1BPUDHUKOTTAI ROAD, MAPILLAI NAYAKKANPATTI   1

Name: count, Length: 142910, dtype: int64

REGISTRAR_OF_COMPANIES Value Counts:

REGISTRAR_OF_COMPANIES

ROC CHENNAI   122233

ROC COIMBATORE   28153

ROC DELHI   310

ROC HYDERABAD   1

Name: count, dtype: int64

EMAIL_ADDR Value Counts:

EMAIL_ADDR

ganravi@gmail.com   182

compliance@kanakkupillai.com   176

secretarial@stjohntrack.com   161

smrajunaidu@gmail.com   144

pcschn1@gmail.com   133

            ...

info@skymaxlogistics.com   1

vishnu2444@yahoo.com   1

rashahuljob@gmail.com   1

baskar.mrl@gmail.com   1

nroottechnologies@gmail.com   1

Name: count, Length: 79940, dtype: int64

LATEST_YEAR_ANNUAL_RETURN Value Counts:

LATEST_YEAR_ANNUAL_RETURN

31-03-2019   44168

31-03-2018   8816

31-03-2017   3149

31-03-2013   2514

31-03-2014   2329

          ...

24-03-2008   1

15-06-2009   1

30-03-2011   1

30-06-2016   1

31-01-2015   1

Name: count, Length: 169, dtype: int64

LATEST_YEAR_FINANCIAL_STATEMENT Value Counts:

LATEST_YEAR_FINANCIAL_STATEMENT

31-03-2019   44171

31-03-2018   9008

31-03-2017   3122

31-03-2013   2585

31-03-2014   2175

          ...

10-04-2009   1

24-05-2006	1
31-07-2006	1
24-03-2008	1
31-01-2015	1

Name: count, Length: 138, dtype: int64

EDA Bivariate Analysis

Bivariate Analysis helps to understand how variables are related to each other and the relationship between dependent and independent variables present in the dataset.

For Numerical variables, Pair plots and Scatter plots are widely been used to do Bivariate Analysis.

A Stacked bar chart can be used for categorical variables if the output variable is a classifier. Bar plots can be used if the output variable is continuous

In our example, a pair plot has been used to show the relationship between two Categorical variables.

Program :

```
# Subset the DataFrame with the selected columns

selected_df = df[columns_for_analysis]


# Bivariate analysis: Numerical vs. Numerical (AUTHORIZED_CAP vs.
PAIDUP_CAPITAL)


plt.figure(figsize=(8, 6))

sns.scatterplot(x='AUTHORIZED_CAP', y='PAIDUP_CAPITAL',
data=selected_df)
```
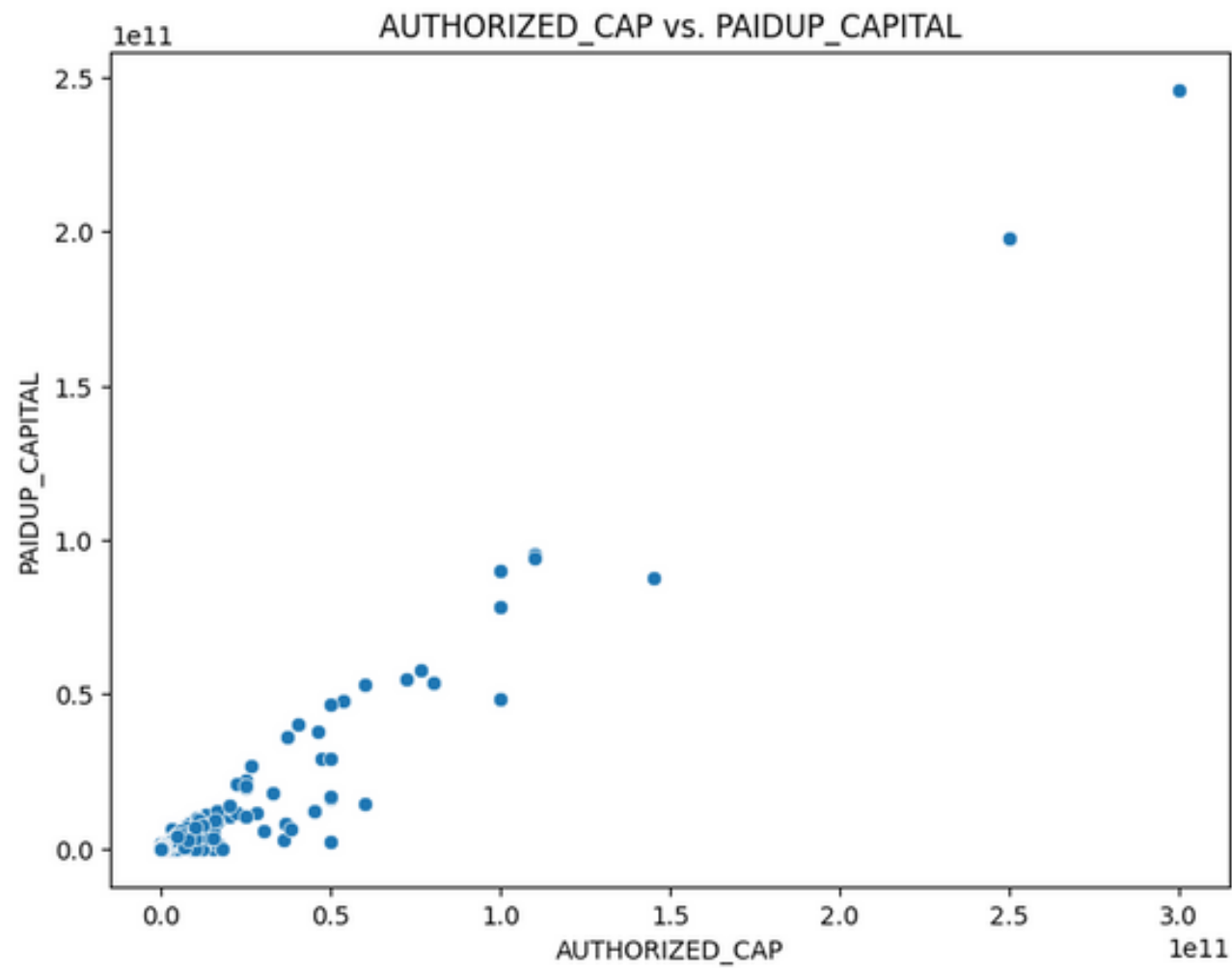
```
plt.title('AUTHORIZED_CAP vs. PAIDUP_CAPITAL')
plt.xlabel('AUTHORIZED_CAP')
plt.ylabel('PAIDUP_CAPITAL')
plt.show()
```


Bivariate analysis: Categorical vs. Categorical (COMPANY_STATUS vs.

REGISTERED_STATE)

```python
crosstab = pd.crosstab(selected_df['COMPANY_STATUS'],
selected_df['REGISTERED_STATE'])

crosstab.plot(kind='bar', stacked=True, figsize=(10, 6))

plt.title('COMPANY_STATUS vs. REGISTERED_STATE')

plt.xlabel('COMPANY_STATUS')

plt.ylabel('Count')

plt.xticks(rotation=45)

plt.show()
```
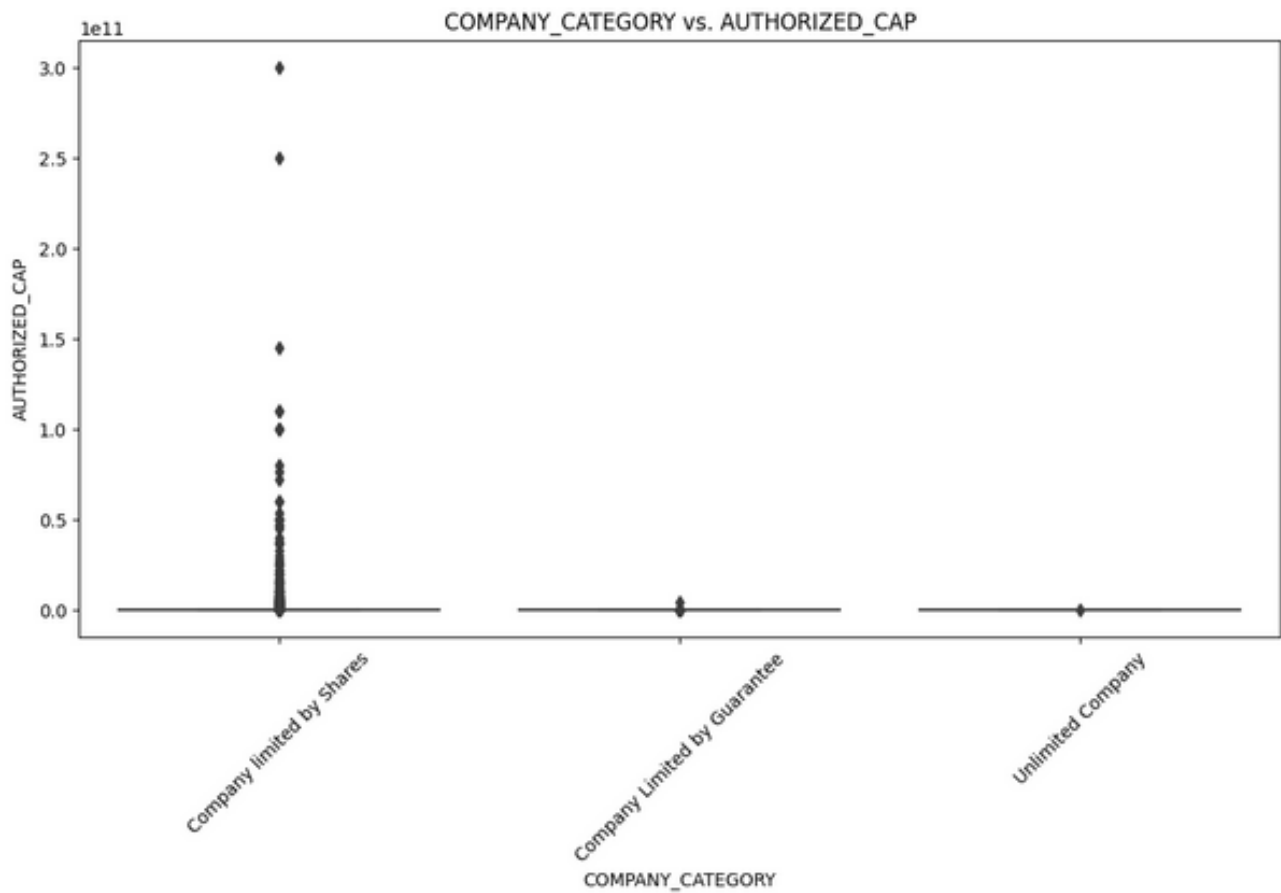


```python
# Bivariate analysis: Categorical vs. Numerical (COMPANY_CATEGORY vs.
AUTHORIZED_CAP)

plt.figure(figsize=(12, 6))
```
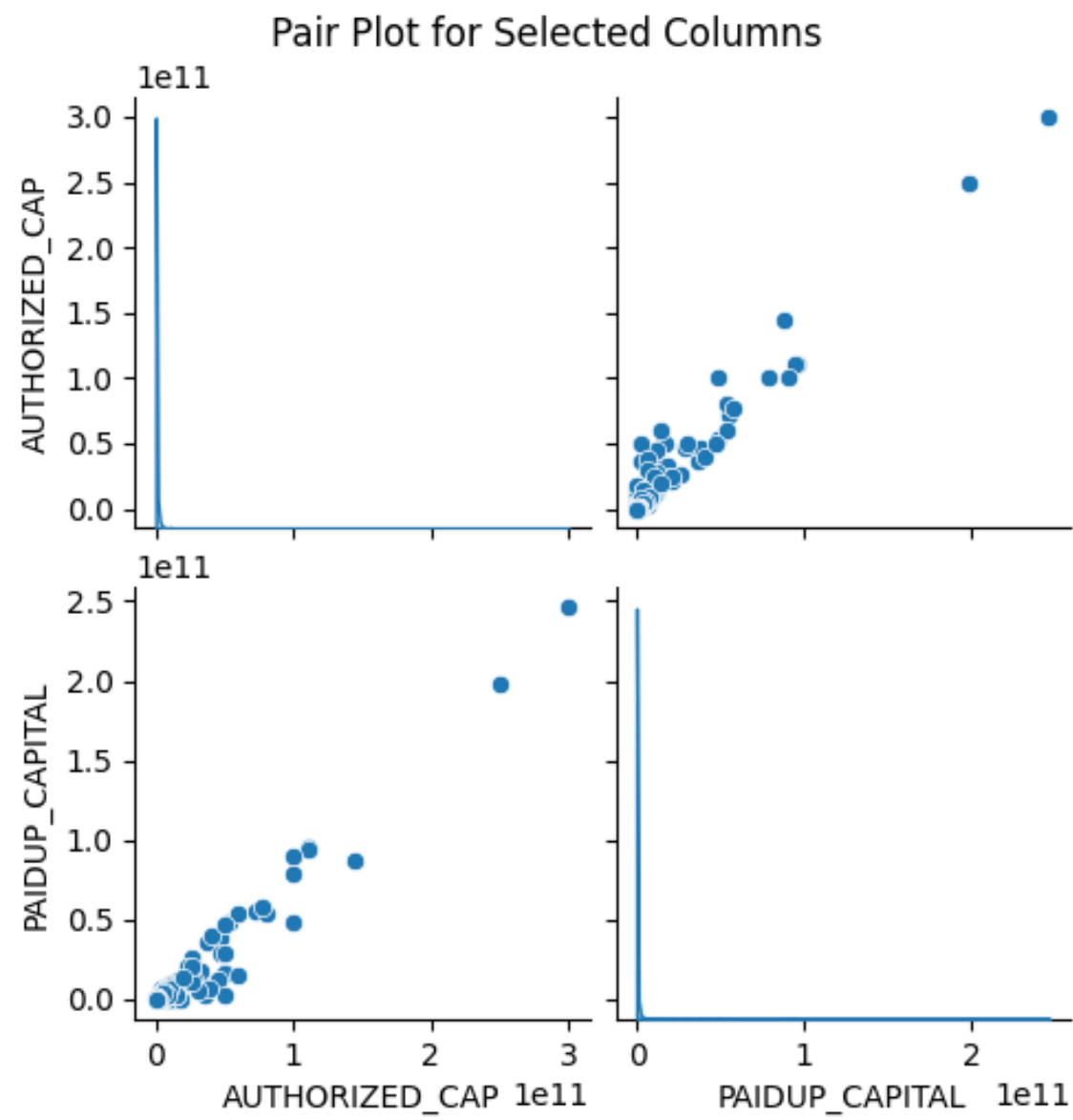
```
sns.boxplot(x='COMPANY_CATEGORY', y='AUTHORIZED_CAP',
data=selected_df)

plt.title('COMPANY_CATEGORY vs. AUTHORIZED_CAP')

plt.xlabel('COMPANY_CATEGORY')

plt.ylabel('AUTHORIZED_CAP')

plt.xticks(rotation=45)

plt.show()
```



```
# Plot the pair plot

sns.pairplot(selected_df, diag_kind='kde', height=2.5)

plt.suptitle('Pair Plot for Selected Columns', y=1.02)
```

```
plt.show()
```

Pair Plot for Selected Columns

EDA Multivariate Analysis

Multivariate analysis is one of the most useful methods to determine relationships and analyze patterns for any dataset.

A heat map is widely been used for Multivariate Analysis

Heat Map gives the correlation between the variables, whether it has a positive or negative correlation.

In our example heat map shows the correlation between the variables.

Program :

```python
# Select the specified columns for analysis
columns_for_analysis = ['AUTHORIZED_CAP', 'PAIDUP_CAPITAL']


# Subset the DataFrame with the selected columns
selected_df = df[columns_for_analysis]


# Convert columns to numeric (if they're not already)
selected_df = selected_df.apply(pd.to_numeric, errors='coerce')


# Calculate the correlation matrix
correlation_matrix = selected_df.corr()


# Plot the heatmap
plt.figure(figsize=(6, 4))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt='.2f', linewidths=0.5)
plt.title('Correlation Matrix Heatmap')
plt.show()
```
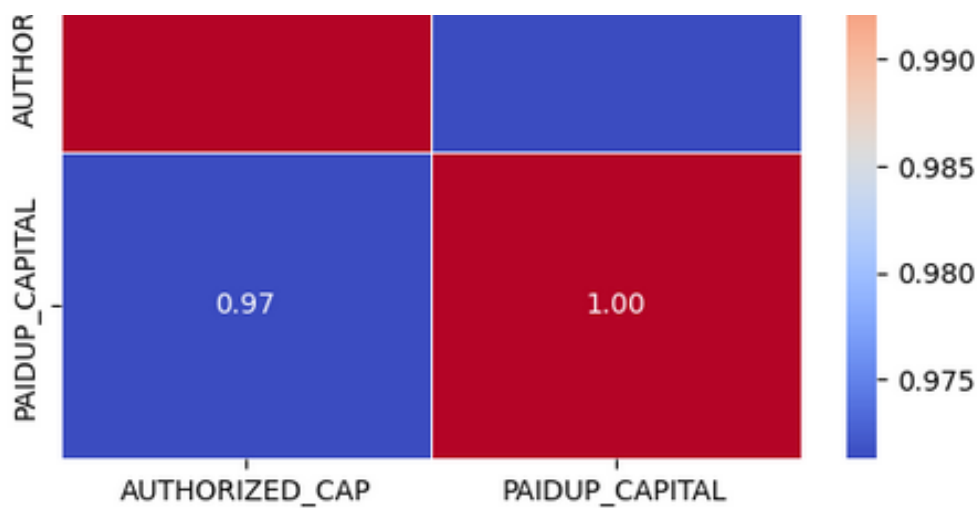
Conclusion

In the task of exploring and predicting company registration trends using data obtained from the Registrar of Companies (RoC), a comprehensive approach was adopted involving data collection, data preprocessing, and exploratory data analysis (EDA). Initially, data was collected from various sources including RoC records, financial data, industry information, and market indicators. This data encompassed essential attributes such as corporate identification numbers, company names, registration dates, financial figures, business activities, and more.

The subsequent step involved data preprocessing, where meticulous attention was given to handling missing values and converting data types to ensure consistency and

accuracy in the dataset. Techniques such as imputation were utilized to manage missing data, and categorical variables were appropriately encoded to facilitate subsequent analysis. Furthermore, numerical features were scaled to a consistent range, while relevant date columns were transformed for improved insights.

Following data preprocessing, univariate analysis was conducted, scrutinizing individual features to grasp their distributions, frequencies, and unique values. This analysis shed light on the status and characteristics of companies, their registration dates, authorized and paid-up capital, and other crucial aspects. Moving to bivariate analysis, relationships between pairs of variables were explored, offering insights into potential correlations and patterns. Specifically, correlations between authorized capital and paid-up capital were studied, revealing interesting trends.

Finally, multivariate analysis was employed, focusing on understanding the interrelationships between multiple variables. A correlation matrix and heatmap were constructed, illuminating the associations between selected numeric features such as authorized capital and paid-up capital. The heatmap visually represented the strength and direction of these relationships, providing valuable insights for further analysis and predictions.

In conclusion, the seamless integration of data collection, preprocessing, and exploratory analysis enabled a holistic understanding of the company registration landscape. These foundational steps are pivotal in laying the groundwork for subsequent predictive modeling and informed decision-making in the realm of business and finance.