AI-Driven Exploration and Prediction of Company Registration Trends with Registrar of Companies (RoC)

Phase-1: Project Submission

Abstract

Companies are the backbone of any economy, and their growth is essential for sustainable development. The Registrar of Companies (ROC) in India is the government body responsible for registering and regulating companies. The ROC maintains a database of all registered companies, including information such as the company name, address, directors, and shareholders Artificial intelligence (AI) can be used to automate the analysis of company registration data and identify trends more efficiently. For example, AI can be used to classify companies into different industries and to identify geographical clusters of companies. AI can also be used to predict future trends in company registration.

Modules

Data Acquisition and Preprocessing Module:

This initial module focuses on gathering and preprocessing vast amounts of data from the RoC, including historical registration records, financial reports, and associated metadata. Data cleansing and normalization techniques are applied to ensure data quality and consistency.

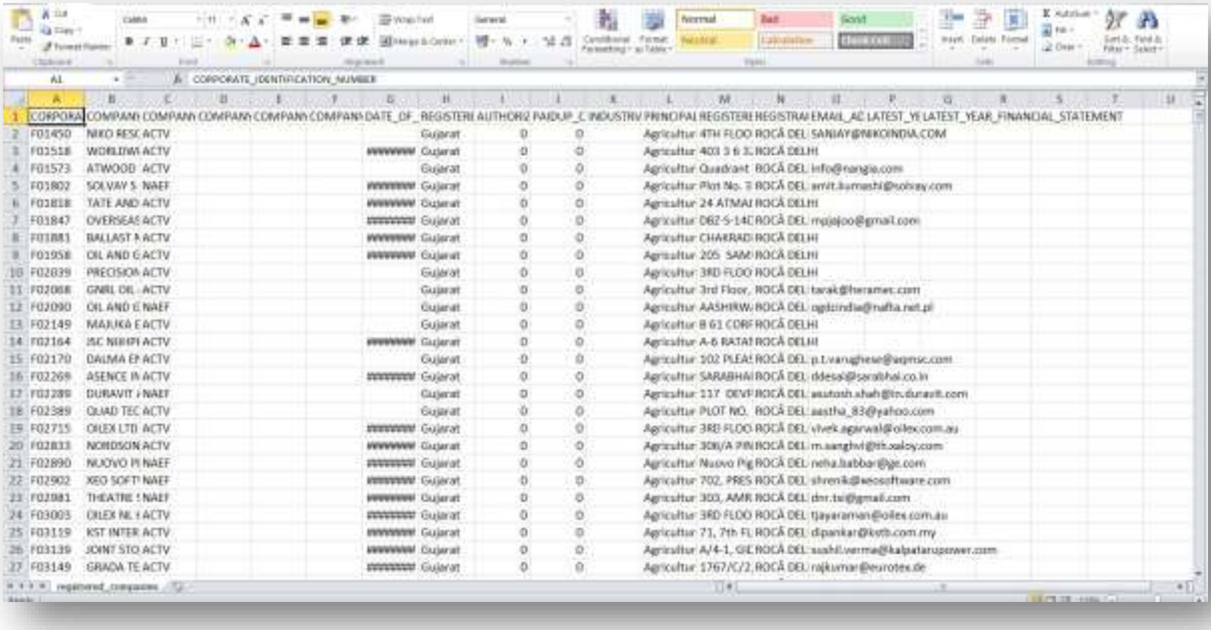
Natural Language Processing (NLP) Module:

To extract valuable insights from unstructured textual data such as company descriptions and business activities, a dedicated NLP module employs techniques like sentiment analysis, topic modeling, and named entity recognition. This module helps in understanding the diversity and nature of businesses being registered.

Time Series Analysis Module:

Leveraging time series analysis techniques, this module identifies patterns, seasonality, and cyclicality within the historical registration data. It also integrates external factors, such as economic indicators and regulatory changes, to provide a holistic view of the registration trends.

Machine Learning Predictive Module:



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	CORPORA	COMPANY	COMPANY	COMPANY	COMPANY	COMPANY	DATE_OF_	REGISTERE	AUTHORIZ	PAIDUP_C	INDUSTRY	PRINCIPAL	REGISTERE	REGISTRAI	EMAIL_AD	LATEST_YE	LATEST_YEAR_FINANCIAL_STATEMENT
2	F01450	NIKO RESC	ACTV					Gujarat	0	0		Agricultur	4TH FLOO	ROCĂ DEL	SANJAY@NIKOINDIA.COM		
3	F01518	WORLDWI	ACTV				########	Gujarat	0	0		Agricultur	403 3 6 3.	ROCĂ DELHI			
4	F01573	ATWOOD	ACTV					Gujarat	0	0		Agricultur	Quadrant	ROCĂ DEL	info@nangia.com		
5	F01802	SOLVAY S	NAEF				########	Gujarat	0	0		Agricultur	Plot No. 3	ROCĂ DEL	amit.kumashi@solvay.com		
6	F01818	TATE AND	ACTV				########	Gujarat	0	0		Agricultur	24 ATMAI	ROCĂ DELHI			
7	F01847	OVERSEAS	ACTV				########	Gujarat	0	0		Agricultur	DBZ-S-14C	ROCĂ DEL	mpjajoo@gmail.com		
8	F01881	BALLAST N	ACTV				########	Gujarat	0	0		Agricultur	CHAKRADI	ROCĂ DELHI			
9	F01958	OIL AND G	ACTV				########	Gujarat	0	0		Agricultur	205 SAM	ROCĂ DELHI			
10	F02039	PRECISION	ACTV					Gujarat	0	0		Agricultur	3RD FLOO	ROCĂ DELHI			
11	F02068	GNRL OIL	ACTV					Gujarat	0	0		Agricultur	3rd Floor,	ROCĂ DEL	tarak@heramec.com		
12	F02090	OIL AND G	NAEF					Gujarat	0	0		Agricultur	AASHIRW	ROCĂ DEL	ogdcindia@nafta.net.pl		
13	F02149	MAJUKA E	ACTV					Gujarat	0	0		Agricultur	B 61 CORF	ROCĂ DELHI			
14	F02164	JSC NIIHPI	ACTV				########	Gujarat	0	0		Agricultur	A-6 RATAI	ROCĂ DELHI			
15	F02170	DALMA EP	ACTV					Gujarat	0	0		Agricultur	102 PLEA!	ROCĂ DEL	p.t.varughese@aqmsc.com		
16	F02269	ASENCE IN	ACTV				########	Gujarat	0	0		Agricultur	SARABHAI	ROCĂ DEL	ddesai@sarabhai.co.in		
17	F02289	DURAVIT I	NAEF					Gujarat	0	0		Agricultur	117 DEVF	ROCĂ DEL	asutosh.shah@in.duravit.com		
18	F02389	QUAD TEC	ACTV					Gujarat	0	0		Agricultur	PLOT NO.	ROCĂ DEL	aastha_83@yahoo.com		
19	F02715	OILEX LTD	ACTV				########	Gujarat	0	0		Agricultur	3RD FLOO	ROCĂ DEL	vivek.agarwal@oilex.com.au		
20	F02833	NORDSON	ACTV				########	Gujarat	0	0		Agricultur	306/A PIN	ROCĂ DEL	m.sanghvi@th.xalloy.com		
21	F02890	NUOVO PI	NAEF				########	Gujarat	0	0		Agricultur	Nuovo Pig	ROCĂ DEL	neha.babbar@ge.com		
22	F02902	XEO SOFT	NAEF				########	Gujarat	0	0		Agricultur	702, PRES	ROCĂ DEL	shrenik@xeosoftware.com		
23	F02981	THEATRE !	NAEF				########	Gujarat	0	0		Agricultur	303, AMR	ROCĂ DEL	dnr.tsi@gmail.com		
24	F03003	OILEX NL I	ACTV				########	Gujarat	0	0		Agricultur	3RD FLOO	ROCĂ DEL	tjayaraman@oilex.com.au		
25	F03119	KST INTER	ACTV				########	Gujarat	0	0		Agricultur	71, 7th FL	ROCĂ DEL	dipankar@kstb.com.my		
26	F03139	JOINT STO	ACTV				########	Gujarat	0	0		Agricultur	A/4-1, GE	ROCĂ DEL	sushil.verma@kalpatarupower.com		
27	F03149	GRADA TE	ACTV				########	Gujarat	0	0		Agricultur	1767/C/2,	ROCĂ DEL	rajkumar@eurotex.de		

Employing various machine learning algorithms, this module develops predictive models that forecast future company registration trends. These models take into account the insights generated from the NLP module and time series analysis, enabling accurate predictions of registration volumes and industry shifts.

Visualization and Interpretation Module:

To facilitate decision-making, this module offers interactive data visualizations and intuitive dashboards. Stakeholders can explore registration trends, drill down into specific industries, and gain actionable insights from the AI-driven predictions.

Continuous Learning and Updating Module:

Given the dynamic nature of the business environment, this module ensures that the AI-driven system continually adapts to changing trends and regulations. It employs online learning techniques and periodic model updates to maintain prediction accuracy.

PYTHON PROGRAMMING:

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

```
In [2]:
#Importing the dataset
data = pd.read_csv('/kaggle/input/all-indian-companies-registration-data-1900-2019
/registered_companies.csv')
```

```
/opt/conda/lib/python3.7/site-packages/IPython/core/interactiveshell.py:3166:
DtypeWarning: Columns (10) have mixed types.Specify dtype option on import or
set low_memory=False.
  interactivity=interactivity, compiler=compiler, result=result)
```

```
In [3]:
df = data
```

Working the data

```
In [4]:
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1992170 entries, 0 to 1992169
Data columns (total 17 columns):
 #   Column                                 Dtype
---                                         -------
 0   CORPORATE_IDENTIFICATION_NUMBER        object
 1   COMPANY_NAME                           object
 2   COMPANY_STATUS                         object
 3   COMPANY_CLASS                          object
 4   COMPANY_CATEGORY                       object
 5   COMPANY_SUB_CATEGORY                   object
 6   DATE_OF_REGISTRATION                   object
 7   REGISTERED_STATE                       object
 8   AUTHORIZED_CAP                         float64
 9   PAIDUP_CAPITAL                         float64
 10  INDUSTRIAL_CLASS                       object
 11  PRINCIPAL_BUSINESS_ACTIVITY_AS_PER_CIN object
 12  REGISTERED_OFFICE_ADDRESS              object
 13  REGISTRAR_OF_COMPANIES                 object
 14  EMAIL_ADDR                             object
 15  LATEST_YEAR_ANNUAL_RETURN              object
 16  LATEST_YEAR_FINANCIAL_STATEMENT        object
dtypes: float64(2), object(15)
memory usage: 258.4+ MB
```

-There are a lot of categoricals to work with \ -Many columns seem broadly empty \

```
In [5]:
```

```
print(f"Total Values : {len(df)}\n")
for x in df.columns:
    print(f'{len(df)-df[x].count()} values missing in {x}')
```

```
Total Values : 1992170

0 values missing in CORPORATE_IDENTIFICATION_NUMBER
0 values missing in COMPANY_NAME
0 values missing  in  COMPANY_STATUS
5078 values missing in COMPANY_CLASS
5085 values missing in COMPANY_CATEGORY
5090 values missing in COMPANY_SUB_CATEGORY
2525 values missing in DATE_OF_REGISTRATION
0 values missing in REGISTERED_STATE
0 values missing in AUTHORIZED_CAP
0 values missing in PAIDUP_CAPITAL
4811 values missing in INDUSTRIAL_CLASS
12 values missing in PRINCIPAL_BUSINESS_ACTIVITY_AS_PER_CIN
15259 values missing in REGISTERED_OFFICE_ADDRESS
42198 values missing in REGISTRAR_OF_COMPANIES
370208 values missing in EMAIL_ADDR
831317 values missing in LATEST_YEAR_ANNUAL_RETURN
828829 values missing in LATEST_YEAR_FINANCIAL_STATEMENT
```

In [6]:

```
print(len(df))
df.head(3)
```

```
1992170
```

In [7]:

```
df = df.dropna()
print(len(df))
df.head(3)
```

```
1124485
```

In [8]:

```
#Interesting data, will keep it narrow and efficient for now
dropCols = ["LATEST_YEAR_FINANCIAL_STATEMENT",
            "EMAIL_ADDR", "COMPANY_NAME",
            "LATEST_YEAR_ANNUAL_RETURN",
            "CORPORATE_IDENTIFICATION_NUMBER",
            "REGISTERED_OFFICE_ADDRESS"]

df = df.drop(dropCols, axis=1)
```

In [9]:

```
df["DATE_OF_REGISTRATION"] = df["DATE_OF_REGISTRATION"].apply(pd.to_datetime)
#df["INDUSTRIAL_CLASS"] = df["INDUSTRIAL_CLASS"].astype(int)
```

In [10]:

```
df['REG_YEAR'] = df['DATE_OF_REGISTRATION'].dt.year
df['REG_MONTH'] = df['DATE_OF_REGISTRATION'].dt.month
```

In [11]:

```
#columns along the number of unique items in them along a list of it
for x in df.columns:
    print(f'{x} : {len(df[x].unique())}\n{df[x].unique()[:20]}\n')
```

```
COMPANY_STATUS : 12
['ACTV' 'ULQD' 'AMAL' 'DISD' 'CLLD' 'UPSO' 'STOF' 'CLLP' 'D455' 'NAEF'
 'LIQD' 'DRMT']
```

```
COMPANY_CLASS : 3
['Public' 'Private' 'Private(One Person Company)']

COMPANY_CATEGORY : 3
['Company limited by Shares' 'Company Limited by Guarantee'
 'Unlimited Company']

COMPANY_SUB_CATEGORY : 5
['Non-govt company' 'State Govt company' 'Subsidiary of Foreign Company'
 'Guarantee and Association comp' 'Union Govt company']

DATE_OF_REGISTRATION : 21228
['1991-06-21T00:00:00.000000000'   '1994-01-17T00:00:00.000000000'
 '1994-02-23T00:00:00.000000000'   '1996-04-15T00:00:00.000000000'
 '2011-12-27T00:00:00.000000000'   '2011-09-14T00:00:00.000000000'
 '1994-10-19T00:00:00.000000000'   '1994-01-24T00:00:00.000000000'
 '2004-04-02T00:00:00.000000000'   '1990-09-26T00:00:00.000000000'
 '2014-07-28T00:00:00.000000000'   '1995-07-03T00:00:00.000000000'
 '1980-06-26T00:00:00.000000000'   '1996-08-27T00:00:00.000000000'
 '2008-08-29T00:00:00.000000000'   '2009-06-03T00:00:00.000000000'
 '1994-12-16T00:00:00.000000000'   '1994-09-11T00:00:00.000000000'
 '1996-11-06T00:00:00.000000000' '2005-10-24T00:00:00.000000000']

REGISTERED_STATE : 36
['Gujarat' 'Karnataka' 'Rajasthan' 'Madhya Pradesh' 'Uttaranchal' 'Assam'
 'Jharkhand' 'Tamil Nadu' 'Delhi' 'Maharashtra' 'Haryana' 'Chattisgarh'
 'Daman and Diu' 'West Bengal' 'Lakshadweep' 'Himachal Pradesh'
 'Dadra and Nagra Haveli' 'Kerala' 'Pondicherry' 'Jammu and Kashmir']

AUTHORIZED_CAP : 8463
[1.3e+08 2.2e+08 5.5e+07 6.0e+07 3.2e+08 1.7e+08 8.0e+07 7.7e+08 6.1e+07
 5.0e+07 1.0e+08 7.5e+07 2.5e+08 7.0e+07 5.0e+09 1.5e+11 1.5e+08 1.5e+09
 2.0e+07 2.6e+08]

PAIDUP_CAPITAL : 134551
 [1.20300000e+08 2.11200000e+08 2.73162000e+07 4.93628000e+07
  3.00732620e+08 1.09801580e+08 6.10207000e+07 2.49275094e+08
  6.06799000e+07 4.83000000e+07 7.87029600e+07 5.33120000e+07
  4.77845600e+07 2.35266220e+08 2.14200000e+08 1.26066000e+08
 1.00000000e+08 3.93610000e+07 1.32259300e+09 4.49998688e+10]

INDUSTRIAL_CLASS : 6828
 [1110.0 1111.0 1112.0 1119.0 1122.0 1130.0 1132.0 1135.0 1200.0 1403.0
  1405.0 5004.0 10102.0 10300.0 11100.0 11711.0 14100.0 15122.0 15140.0
 15142.0]

PRINCIPAL_BUSINESS_ACTIVITY_AS_PER_CIN : 17
['Agriculture & allied' 'Mining and quarrying' 'Manufacturing'
 'Electricity gas and water supply' 'Construction'
 'Wholesale and retail trade repair of motor vehicles motorcycles and personal
and household goods'
 'Unclassified' 'Hotels and restaurants'
 'Transport storage and communications' 'Financial intermediation'
 'Real estate renting and business activities' 'Education'
 'Health and social work'
```

```
 'Other community social and personal service activities'
 'Extraterritorial organizations and bodies'
 'Activities of private households as employers and undifferentiated productio
n activities of private households'
 'Public administration and defence compulsory social security']

REGISTRAR_OF_COMPANIES : 25
['ROC\xa0AHMEDABAD' 'ROC\xa0GOA' 'ROC\xa0BANGALORE' 'ROC\xa0JAIPUR'
 'ROC\xa0KOLKATA' 'ROC\xa0GWALIOR' 'ROC\xa0UTTARAKHAND' 'ROC\xa0KANPUR'
 'ROC\xa0SHILLONG' 'ROC\xa0JHARKHAND' 'ROC\xa0PATNA' 'ROC\xa0COIMBATORE'
 'ROC\xa0CHENNAI' 'ROC\xa0HYDERABAD' 'ROC\xa0DELHI' 'ROC\xa0MUMBAI'
 'ROC\xa0PUNE' 'ROC\xa0ERNAKULAM' 'ROC\xa0CHHATTISGARH' 'ROC\xa0HP']

REG_YEAR : 148
[1991 1994 1996 2011 2004 1990 2014 1995 1980 2008 2009 2005 1984 1963
 1979 1992 1993 1983 1971 2013]

REG_MONTH : 12
[ 6  1  2  4 12  9 10  7  8 11  5  3]
```

In [12]:

```
#Naming conveniences

df["COMPANY_CLASS"] = df["COMPANY_CLASS"].apply(lambda x:"Solo" if x == 'Private(O
ne Person Company)' else x)
print(df["COMPANY_CLASS"].unique())

df["REGISTRAR"] = df["REGISTRAR_OF_COMPANIES"].apply(lambda x:x.split("ROC\xa0")[-
1])
print(df["REGISTRAR"].unique())

df["PRINCIPAL_BUSINESS"] = df["PRINCIPAL_BUSINESS_ACTIVITY_AS_PER_CIN"].apply(lamb
da x:x.split(" ")[0])
print(df["PRINCIPAL_BUSINESS"].unique())

#Creating bins for REGISTRATION_YEAR

df["REG_YEAR_5BIN"] = df["REG_YEAR"].apply(lambda x:(round(x/5))*5)
print(df["REG_YEAR_5BIN"].unique())

df["REG_YEAR_10BIN"] = df["REG_YEAR"].apply(lambda x:(round(x/10))*10)
print(df["REG_YEAR_10BIN"].unique())

df["REG_YEAR_20BIN"] = df["REG_YEAR"].apply(lambda x:(round(x/20))*20)
print(df["REG_YEAR_20BIN"].unique())
```

```
['Public' 'Private' 'Solo']
['AHMEDABAD' 'GOA' 'BANGALORE' 'JAIPUR' 'KOLKATA' 'GWALIOR' 'UTTARAKHAND'
 'KANPUR' 'SHILLONG' 'JHARKHAND' 'PATNA' 'COIMBATORE' 'CHENNAI'
 'HYDERABAD' 'DELHI' 'MUMBAI' 'PUNE' 'ERNAKULAM' 'CHHATTISGARH' 'HP'
 'PONDICHERRY' 'JAMMU' 'CUTTAK' 'CHANDIGARH' 'ANDAMAN']
['Agriculture' 'Mining' 'Manufacturing' 'Electricity' 'Construction'
 'Wholesale' 'Unclassified' 'Hotels' 'Transport' 'Financial' 'Real'
 'Education' 'Health' 'Other' 'Extraterritorial' 'Activities' 'Public']
[1990 1995 2010 2005 2015 1980 1985 1965 1970 1955 1915 1920 1930 1935
 1950 1960 1975 2000 1945 1940 2020 1905 1910 1890 1895 1900 1925 1880
 1875 1885 1870 1865]
[1990 2000 2010 1980 1960 1970 1910 1920 1930 1940 1950 2020 1900 1890
```

```
 1880 1870 1860]
[2000 2020 1980 1960 1920 1940 1900 1880 1860]
```

Visualization

In [13]:

```python
#Working with a smaller randomly picked sample space for efficiency and overall po
pulation testing.
df2 = df
df = df.sample(n=10000)
```

In [14]:

```python
#For readability since I use dark mode
sns.set_theme(context='notebook',
              style='darkgrid',
              palette='magma',
              font='sans-serif',
              font_scale=0.6,
              color_codes=True,
              rc=None)
```

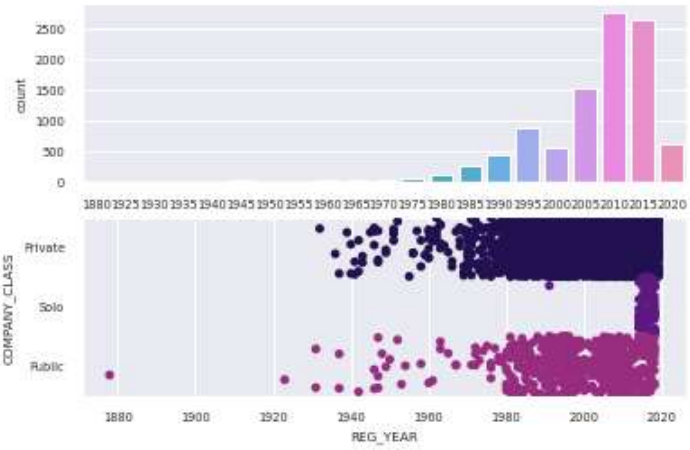
In [15]:

linkcode

```python
f, ax = plt.subplots(2)

#Counting all the number fo companies by REG_YEAR

 sns.countplot(x="REG_YEAR_5BIN",
              data=df, ax = ax[0])


#Year of registration by COMPANY_CLASS

sns.stripplot(x="REG_YEAR",
               y="COMPANY_CLASS",
              data=df, jitter=0.5,
              ax = ax[1])
```

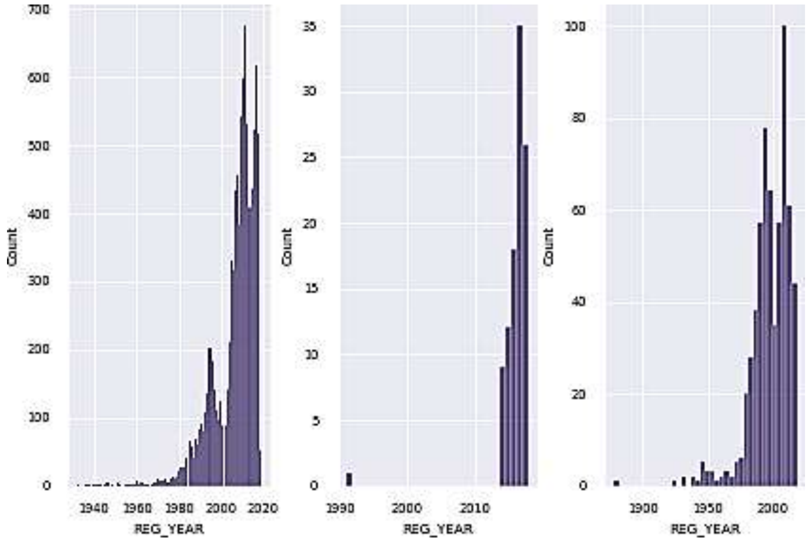
Out[15]:

```
<AxesSubplot:xlabel='REG_YEAR', ylabel='COMPANY_CLASS'>
```



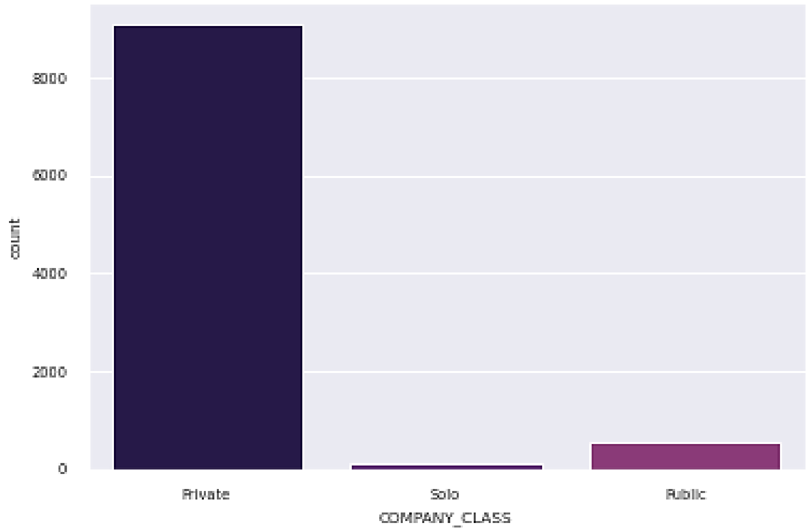

```python
f, ax = plt.subplots(1, len(df["COMPANY_CLASS"].unique()))
f.tight_layout()
y=0
print(df["COMPANY_CLASS"].unique())
for x in df["COMPANY_CLASS"].unique():
    sns.histplot(x="REG_YEAR",
```

```
                 data=df[df["COMPANY_CLASS"]==x],
                 ax=ax[y])
    y+=1
['Private' 'Solo' 'Public']
```

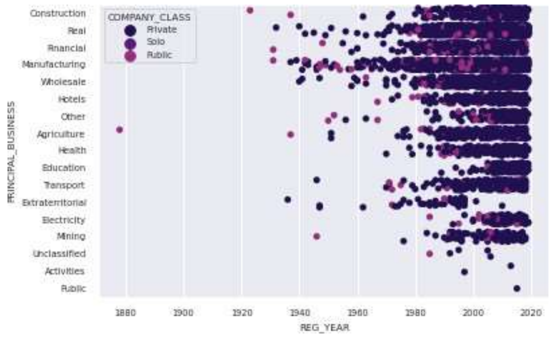


India is seeing a increasing number of new companies being registered which a vast proportion of them being in the 2000s+ \ Solo COMPANY_CLASS catches traction post 2010+ from the first lower plot and can see appearing from 2014 in this sample space on the second rightmost plot.\ The majority of companies are classified as Private (density of the first lower plot & count in second plot).

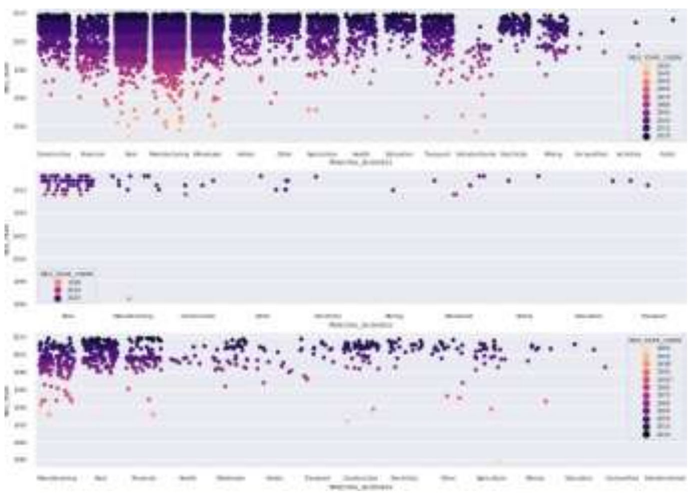
```
In [17]:
sns.countplot(x="COMPANY_CLASS",
              data=df[df["REG_YEAR"] >= 1982]) #Change year
```




```
#Industry of companies

sns.stripplot(x="REG_YEAR",
              y="PRINCIPAL_BUSINESS",
              hue="COMPANY_CLASS",
              data=df, jitter=0.3)
<AxesSubplot:xlabel='REG_YEAR', ylabel='PRINCIPAL_BUSINESS'>
```




```
#Companies in respect to COMPANY_CLASS over time.
y=0
f, ax = plt.subplots(len(df["COMPANY_CLASS"].unique()),1)
f.subplots_adjust(top=1, bottom=-0.9, left=-0.9, hspace=0.2)
print(df["COMPANY_CLASS"].unique())
for x in df["COMPANY_CLASS"].unique():
    sns.stripplot(y="REG_YEAR",
                  x="PRINCIPAL_BUSINESS",
                  data=df[df["COMPANY_CLASS"]==x],
                  hue="REG_YEAR_10BIN",
                  palette="magma_r",
                  jitter=0.4,
                  ax=ax[y])
    y+=1
['Private' 'Solo' 'Public']
```



```
#State wise looking at the industries through time

y=0; f, ax = plt.subplots(len(df["REGISTERED_STATE"].unique()),1)
f.subplots_adjust(top=10, bottom=-0.9,  left=-0.5,  hspace=0.2)
for x in df["REGISTERED_STATE"].unique():
     sns.histplot(y='REG_YEAR',
                  x='PRINCIPAL_BUSINESS',
                  #jitter=0.3,
                  hue="REGISTERED_STATE",
                  palette="magma_r",
                  data= df[df["REGISTERED_STATE"]==x],
                  ax=ax[y])
    y+=1
```

```
len(df)
```

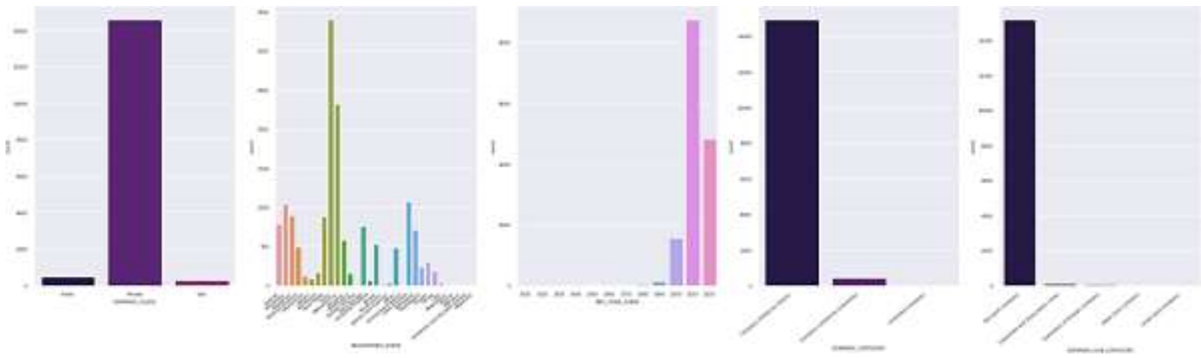
Out[21]:
10000

In [22]:
linkcode

```
df[df["PRINCIPAL_BUSINESS"] == "Education"].head(3)


f, axes = plt.subplots(1,5)
f.subplots_adjust(top=0.5, bottom=-0.9, left=-3, hspace=0.2)
ax1 = sns.countplot(x="COMPANY_CLASS", data=df2[df2["PRINCIPAL_BUSINESS"] == "Education"], ax=axes[0])
ax2 = sns.countplot(x="REGISTERED_STATE", data=df2[df2["PRINCIPAL_BUSINESS"] == "Education"], ax=axes[1])
ax3 = sns.countplot(x="REG_YEAR_10BIN", data=df2[df2["PRINCIPAL_BUSINESS"] == "Education"], ax=axes[2])
ax4 = sns.countplot(x="COMPANY_CATEGORY", data=df2[df2["PRINCIPAL_BUSINESS"] == "Education"], ax=axes[3])
ax5 = sns.countplot(x="COMPANY_SUB_CATEGORY", data=df2[df2["PRINCIPAL_BUSINESS"] == "Education"], ax=axes[4])
ax2.set_xticklabels(ax2.get_xticklabels(), rotation=40, ha="right")
ax4.set_xticklabels(ax4.get_xticklabels(), rotation=40, ha="right")
ax5.set_xticklabels(ax5.get_xticklabels(), rotation=40, ha="right")
```

Out[23]:

```
[Text(0, 0, 'Non-govt company'),
 Text(1, 0, 'Guarantee and Association comp'),
 Text(2, 0, 'Subsidiary of Foreign Company'),
 Text(3, 0, 'State Govt company'),
 Text(4, 0, 'Union Govt company')]
```



Using the df2 full set DataFrame the result and distribution changes quite a bit

In [24]:

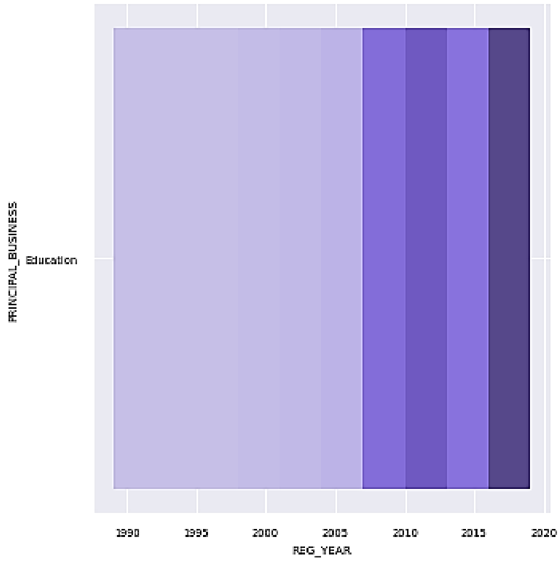
```
print(len(df), len(df2))
```

10000 1124485

In [25]:

```
sns.displot(x="REG_YEAR", y="PRINCIPAL_BUSINESS", data=df[df["PRINCIPAL_BUSINESS"] == "Education"])
```

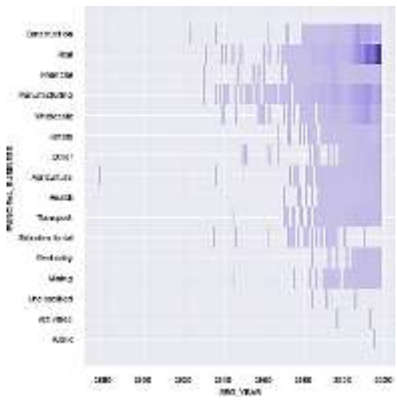
Out[25]:
<seaborn.axisgrid.FacetGrid at 0x7f4abaaa1890>




```
sns.displot(x="REG_YEAR", y="PRINCIPAL_BUSINESS", data=df[df["PRINCIPAL_BUSINESS"]
!= "Education"])
```

Out[26]:

```
<seaborn.axisgrid.FacetGrid at 0x7f4abab1a710>
```



```
df["AUTHORIZED_CAP"].describe()
```

Out[27]:

```
count    1.000000e+04
mean     7.809848e+07
std      2.047122e+09
min      0.000000e+00
25%      1.000000e+05
50%      1.000000e+06
75%      4.000000e+06
max      1.500000e+11
Name: AUTHORIZED_CAP, dtype: float64
```

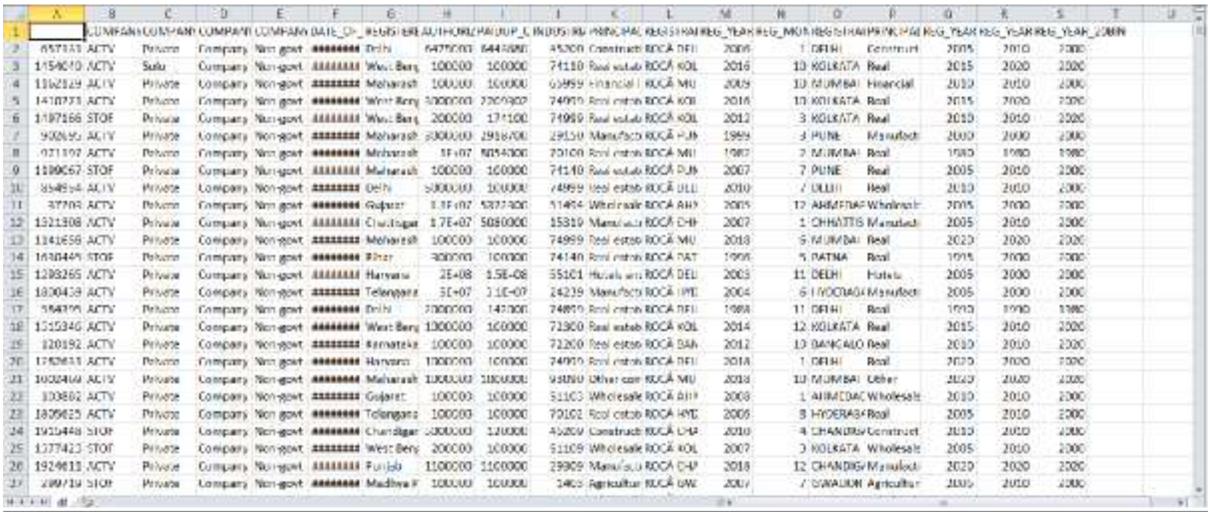
In [28]:

```
df.to_csv('/kaggle/working/df.csv')
```

In [29]:

```
df2.head()
```

OUTPUT:



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1		COMPANY	COMPANY	COMPANY	COMPANY	DATE_OF_	REGISTERE	AUTHORIZ	PAIDUP_C	INDUSTRI	PRINCIPAL	REGISTRAR	REG_YEAR	REG_MON	REGISTRAR	PRINCIPAL	REG_YEAR	REG_YEAR	REG_YEAR_20BIN
2	657113	ACTV	Private	Company	Non-govt	########	Delhi	6475000	6443880	45200	Construct	ROCÂ DEL	2006	1	DELHI	Construct	2005	2010	2000
3	1154010	ACTV	Sulu	Company	Non-govt	########	West Beng	100000	100000	74110	Real estab	ROCÂ KOL	2016	10	KOLKATA	Real	2015	2020	2020
4	1162129	ACTV	Private	Company	Non-govt	########	Maharash	100000	100000	65999	Financial	ROCÂ MU	2009	10	MUMBAI	Financial	2010	2010	2000
5	1410773	ACTV	Private	Company	Non-govt	########	West Beng	3000000	2200907	74010	Real estab	ROCÂ KOL	2016	10	KOLKATA	Real	2015	2020	2020
6	1197166	STOF	Private	Company	Non-govt	########	West Beng	200000	171100	74989	Real estab	ROCÂ KOL	2012	3	KOLKATA	Real	2010	2010	2020
7	900695	ACTV	Private	Company	Non-govt	########	Maharash	3000000	2918700	29150	Manufact	ROCÂ PUN	1999	3	PUNE	Manufact	2000	2000	2000
8	921197	ACTV	Private	Company	Non-govt	########	Maharash	1E+07	8054000	70100	Real estab	ROCÂ MU	1987	2	MUMBAI	Real	1985	1980	1980
9	1100067	STOF	Private	Company	Non-govt	########	Maharash	100000	100000	74140	Real estab	ROCÂ PUN	2007	7	PUNE	Real	2005	2010	2000
10	854894	ACTV	Private	Company	Non-govt	########	Delhi	5000000	100000	74999	Real estab	ROCÂ DEL	2010	7	DELHI	Real	2010	2010	2000
11	37703	ACTV	Private	Company	Non-govt	########	Gujarat	1.3E+07	5877900	51494	Wholesale	ROCÂ AHM	2005	12	AHMEDAB	Wholesale	2005	2000	2000
12	1321308	ACTV	Private	Company	Non-govt	########	Chattisgar	1.7E+07	5080000	15319	Manufact	ROCÂ CHH	2007	1	CHHATTIS	Manufact	2005	2010	2000
13	1141658	ACTV	Private	Company	Non-govt	########	Maharash	100000	100000	74999	Real estab	ROCÂ MU	2018	6	MUMBAI	Real	2020	2020	2020
14	1630445	STOF	Private	Company	Non-govt	########	Bihar	300000	100000	74140	Real estab	ROCÂ PAT	1996	5	PATNA	Real	1995	2000	2000
15	1208265	ACTV	Private	Company	Non-govt	########	Haryana	2E+08	1.5E+08	55101	Hotels and	ROCÂ DEL	2003	11	DELHI	Hotels	2005	2000	2000
16	1800439	ACTV	Private	Company	Non-govt	########	Telangana	3E+07	3.1E+07	24239	Manufact	ROCÂ HYD	2004	6	HYDERAB	Manufact	2005	2000	2000
17	554305	ACTV	Private	Company	Non-govt	########	Delhi	2000000	147000	74899	Real estab	ROCÂ DEL	1988	11	DELHI	Real	1990	1990	1980
18	1515346	ACTV	Private	Company	Non-govt	########	West Beng	1000000	100000	72300	Real estab	ROCÂ KOL	2014	12	KOLKATA	Real	2015	2010	2020
19	120192	ACTV	Private	Company	Non-govt	########	Karnataka	100000	100000	72200	Real estab	ROCÂ BAN	2012	10	BANGALO	Real	2010	2010	2020
20	1752611	ACTV	Private	Company	Non-govt	########	Haryana	1000000	100000	74910	Real estab	ROCÂ DEL	2018	1	DELHI	Real	2020	2020	2020
21	1002469	ACTV	Private	Company	Non-govt	########	Maharash	1000000	1000000	93090	Other com	ROCÂ MU	2018	10	MUMBAI	Other	2020	2020	2020
22	103862	ACTV	Private	Company	Non-govt	########	Gujarat	100000	100000	51103	Wholesale	ROCÂ AHM	2008	1	AHMEDAB	Wholesale	2010	2010	2000
23	1809625	ACTV	Private	Company	Non-govt	########	Telangana	100000	100000	70102	Real estab	ROCÂ HYD	2006	8	HYDERAB	Real	2005	2010	2000
24	1915448	STOF	Private	Company	Non-govt	########	Chandigar	5000000	120000	45200	Construct	ROCÂ CHA	2010	4	CHANDIG	Construct	2010	2010	2000
25	1377423	STOF	Private	Company	Non-govt	########	West Beng	200000	100000	51109	Wholesale	ROCÂ KOL	2007	3	KOLKATA	Wholesale	2005	2010	2000
26	1924611	ACTV	Private	Company	Non-govt	########	Punjab	1100000	1100000	29309	Manufact	ROCÂ CHA	2018	12	CHANDIG	Manufact	2020	2020	2020
27	289719	STOF	Private	Company	Non-govt	########	Madhya P	100000	100000	1405	Agricultur	ROCÂ GW	2007	7	GWALIOR	Agricultur	2005	2010	2000

Ethical and Regulatory Compliance Module:

An essential aspect of this framework is its commitment to ethical data usage and compliance with data privacy regulations. This module ensures that the AI system respects data privacy and operates within legal boundaries.

Through the integration of these modules, our proposed framework empowers governments, regulatory bodies, and businesses with the capability to anticipate company registration trends, identify emerging sectors, and make data-driven policy decisions. This research contributes to the evolving field of AI-driven analytics, offering a practical and scalable solution for understanding and predicting company registration dynamics with Registrar of Companies (RoC).

Conclusion

The AI-Driven Exploration and Prediction of Company Registration Trends module is a powerful tool that can be used to analyze and predict company registration trends. The module can be used by businesses, government agencies, and researchers to improve their efficiency, gain better insights, and make better decisions