

### Data Collection and Preprocessing Phase

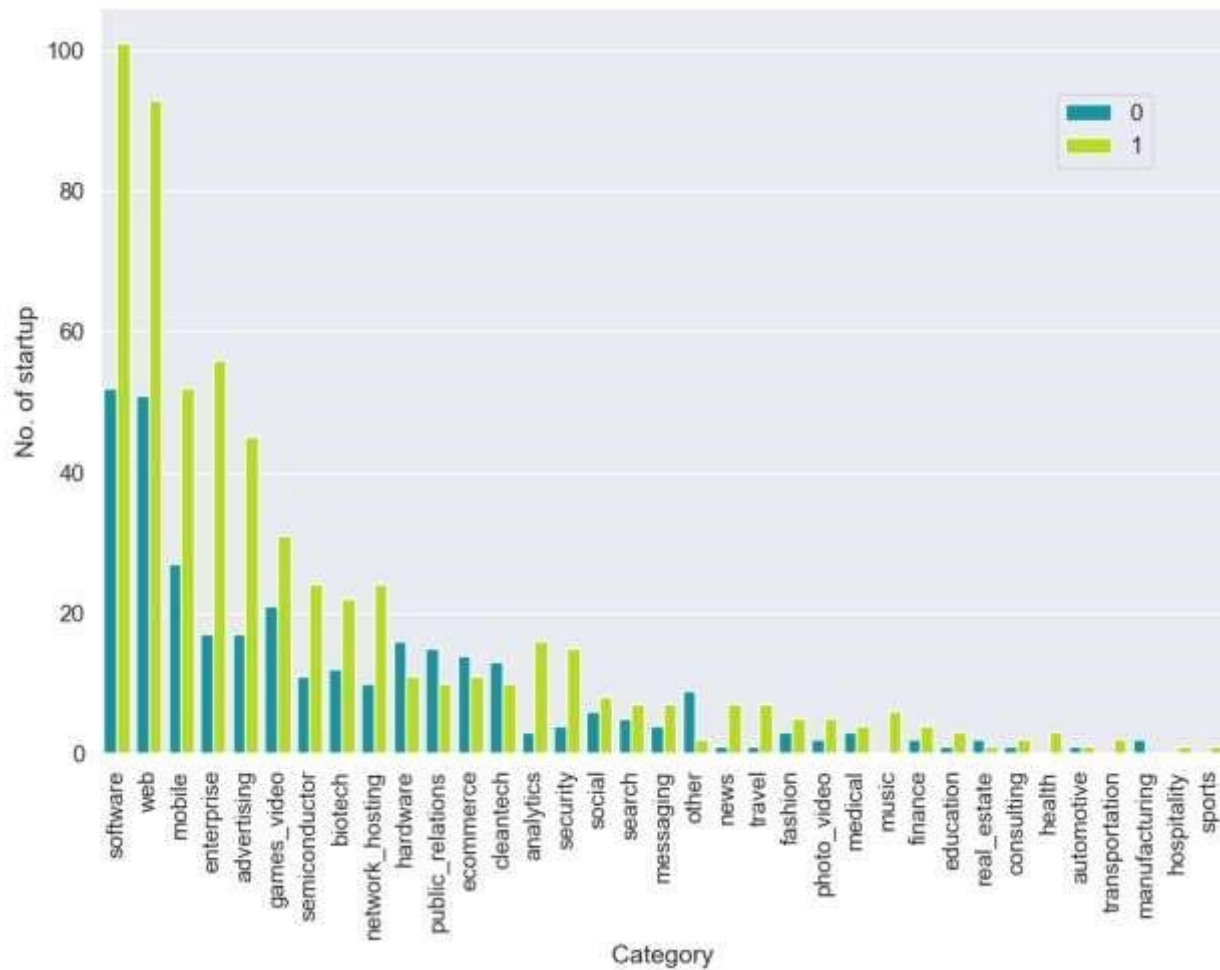
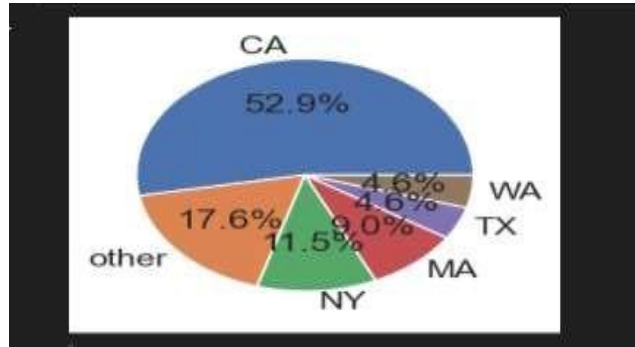
Date	7 July 2024
Team ID	740010
Project Title	prosperity Prognosticator : Machine Learning for Startup Success Prediction
Maximum Marks	6 Marks

### Data Exploration and Preprocessing Report

Dataset variables will be statistically analyzed to identify patterns and outliers, with Python employed for preprocessing tasks like normalization and feature engineering. Data cleaning will address missing values and outliers, ensuring quality for subsequent analysis and modeling, and forming a strong foundation for insights and predictions.

Section	Description
---------	-------------

	<p><u>Dimension:</u></p> <p>923 rows × 49 columns</p> <p><u>Descriptive statistics:</u></p>																																																																																																
Data Overview	<table><tr><th>Unnamed: 0</th><th>state_code</th><th>latitude</th><th>longitude</th><th>zip_code</th><th>id</th><th>city</th><th>Unnamed: 6</th><th>name</th><th>labels</th><th>...</th><th>object_id</th><th>has_VC</th><th>has_angel</th><th>has_roundA</th><th>has_roundB</th></tr><tr><td>0</td><td>1005</td><td>CA</td><td>42.358880</td><td>-71.056820</td><td>92101</td><td>c6669</td><td>San Diego</td><td>NaN</td><td>Bandaintown</td><td>1</td><td>c6669</td><td>0</td><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>204</td><td>CA</td><td>37.238916</td><td>-121.973718</td><td>95032</td><td>c16283</td><td>Los Gatos</td><td>NaN</td><td>TriCipher</td><td>1</td><td>c16283</td><td>1</td><td>0</td><td>0</td><td>1</td></tr><tr><td>2</td><td>1001</td><td>CA</td><td>32.910949</td><td>-117.192656</td><td>92121</td><td>c65620</td><td>San Diego</td><td>San Diego CA 92121</td><td>Phi</td><td>1</td><td>c65620</td><td>0</td><td>0</td><td>1</td><td>0</td></tr><tr><td>3</td><td>738</td><td>CA</td><td>37.320309</td><td>-122.050040</td><td>95014</td><td>c42660</td><td>Cupertino</td><td>Cupertino CA 95014</td><td>Solidcore Systems</td><td>1</td><td>c42660</td><td>0</td><td>0</td><td>0</td><td>1</td></tr><tr><td>4</td><td>1002</td><td>CA</td><td>37.779281</td><td>-122.419236</td><td>94105</td><td>c65806</td><td>San Francisco</td><td>San Francisco CA 94105</td><td>Whole Digital</td><td>0</td><td>c65806</td><td>1</td><td>1</td><td>0</td><td>0</td></tr></table>	Unnamed: 0	state_code	latitude	longitude	zip_code	id	city	Unnamed: 6	name	labels	...	object_id	has_VC	has_angel	has_roundA	has_roundB	0	1005	CA	42.358880	-71.056820	92101	c6669	San Diego	NaN	Bandaintown	1	c6669	0	1	0	0	1	204	CA	37.238916	-121.973718	95032	c16283	Los Gatos	NaN	TriCipher	1	c16283	1	0	0	1	2	1001	CA	32.910949	-117.192656	92121	c65620	San Diego	San Diego CA 92121	Phi	1	c65620	0	0	1	0	3	738	CA	37.320309	-122.050040	95014	c42660	Cupertino	Cupertino CA 95014	Solidcore Systems	1	c42660	0	0	0	1	4	1002	CA	37.779281	-122.419236	94105	c65806	San Francisco	San Francisco CA 94105	Whole Digital	0	c65806	1	1	0	0
Unnamed: 0	state_code	latitude	longitude	zip_code	id	city	Unnamed: 6	name	labels	...	object_id	has_VC	has_angel	has_roundA	has_roundB																																																																																		
0	1005	CA	42.358880	-71.056820	92101	c6669	San Diego	NaN	Bandaintown	1	c6669	0	1	0	0																																																																																		
1	204	CA	37.238916	-121.973718	95032	c16283	Los Gatos	NaN	TriCipher	1	c16283	1	0	0	1																																																																																		
2	1001	CA	32.910949	-117.192656	92121	c65620	San Diego	San Diego CA 92121	Phi	1	c65620	0	0	1	0																																																																																		
3	738	CA	37.320309	-122.050040	95014	c42660	Cupertino	Cupertino CA 95014	Solidcore Systems	1	c42660	0	0	0	1																																																																																		
4	1002	CA	37.779281	-122.419236	94105	c65806	San Francisco	San Francisco CA 94105	Whole Digital	0	c65806	1	1	0	0																																																																																		
Univariate Analysis																																																																																																	



Bivariate  
Analysis

Outliers and  
Anomalies

-

## Data Preprocessing Code Screenshots

Loading Data

```
data = pd.read_csv('startup.csv')
data
```

	Unnamed: 0	state_code	latitude	longitude	zip_code	id	city	Unnamed: 6	name	labels	...	object_id	has_VC	has_angel	has_roundA	has_roundB
0	1005	CA	42.350681	-71.056620	02101	c5668	San Diego	NaN	Sandtown	1	...	c5668	0	1	0	1
1	294	CA	32.28994	-121.972758	95032	c16263	Los Gatos	NaN	HiGphr	1	...	c16263	1	0	0	1
2	1001	CA	32.901049	-117.393336	92121	c15620	San Diego	San Diego CA 92121	Plai	1	...	c15620	0	0	1	1
3	738	CA	37.320381	-122.030940	95014	c42668	Capetino	Capetino CA 95014	Soldcore Systems	1	...	c42668	0	0	0	1
4	1002	CA	37.779281	-122.419236	94105	c83906	San Francisco	San Francisco CA 94105	Whole Digital	0	...	c83906	1	1	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
918	352	CA	37.462594	-122.376471	94102	c21343	San Francisco	NaN	Colacort	1	...	c21343	0	0	1	1
919	721	MA	42.594817	-71.295811	1003	c41747	Burlington	Burlington MA 1003	Red Point Systems	0	...	c41747	1	0	0	1
920	557	CA	37.493281	-122.073303	94089	c11549	Sunnyvale	NaN	Paraco Medical	0	...	c11549	0	0	0	1
921	584	CA	37.56732	-122.260729	94034	c33198	San Francisco	NaN	Carista	1	...	c33198	0	0	1	1
922	462	CA	37.386778	-122.946277	93554	c25702	Santa Clara	Santa Clara CA 95054	Aurepra Technologies	1	...	c25702	0	0	0	1

Handling  
Missing Data


```
#filling missing value column(unnamed:6)
data['unnamed: 6'] = data.apply(lambda row: (row.city) + " " + (row.state_code) + " " + (row.zip_code) , axis = 1)

# Total Missing Values column "unnamed: 6"
totalNull = data['unnamed: 6'].isnull().sum()

print('Total Missing Values kolom "unnamed: 6": ', totalNull)

#filling missing values of column(closed_at)
data['closed_at'] = data['closed_at'].fillna(value="31/12/2013")
totalNull = data['closed_at'].isnull().sum()

print('Total Missing Values kolom "closed_at": ', totalNull)
```

Data Transformation	 <pre>data["status"] = data.status.map({'acquired':1, 'closed':0}) data["status"].astype(int)</pre> <p>0 1 1 1 2 1 3 1 4 0 918 1 919 0 920 0 921 1 922 1 Name: status, Length: 923, dtype: int64</p>
Feature Engineering	Attached the codes in final submission.
Save Processed Data	-