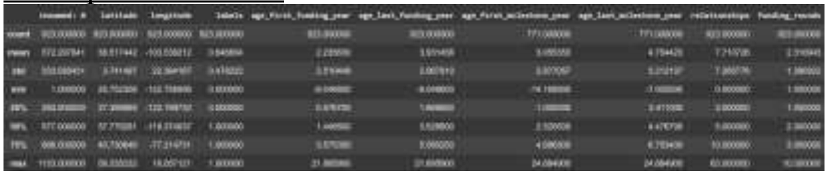
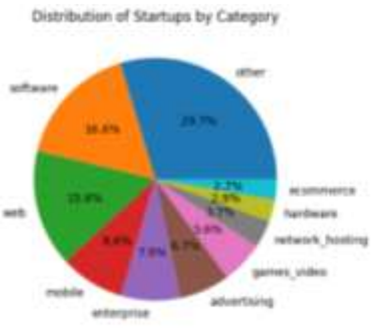
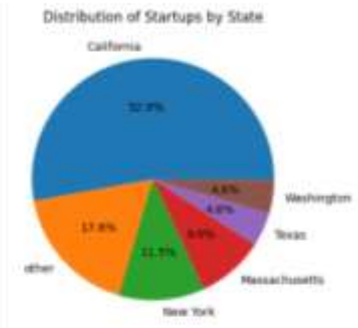
Data Collection and Preprocessing Phase

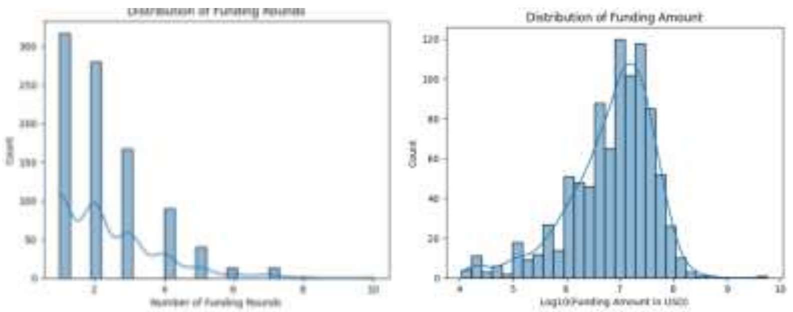
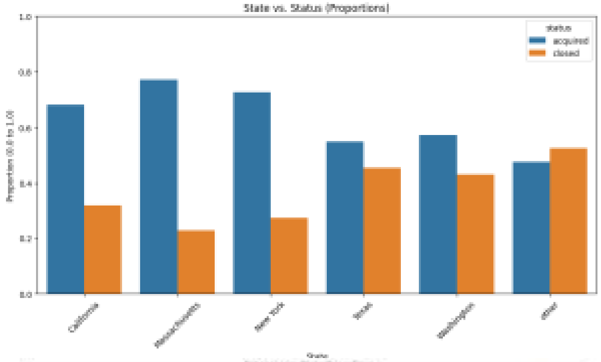
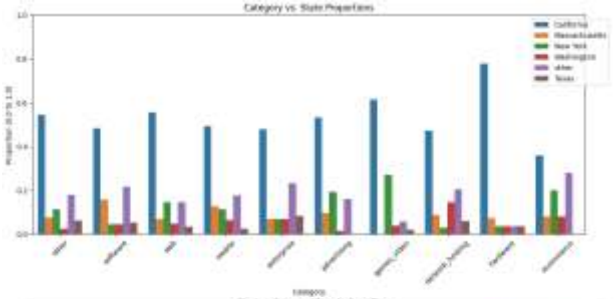
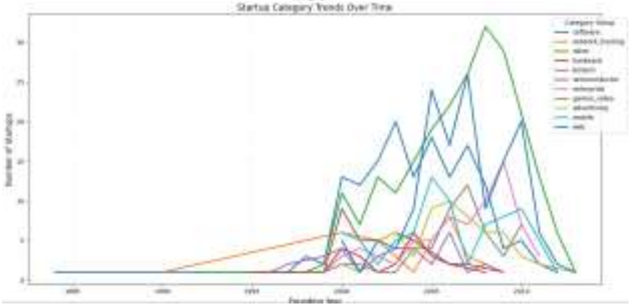
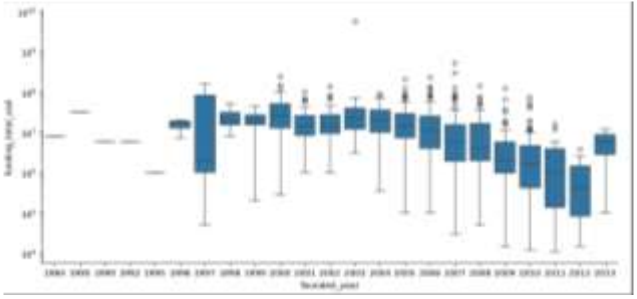
Date	18 June 2025
Team ID	SWTID1749880888
Project Title	Prosperity Prognosticator: Machine Learning for Startup Success Prediction
Maximum Marks	6 Marks

Data Exploration and Preprocessing Template

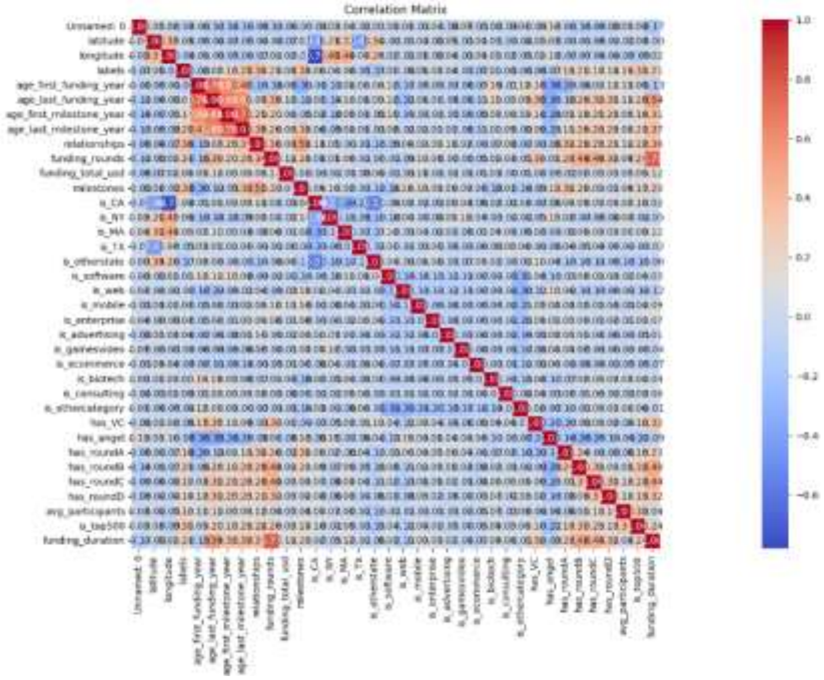
Identifies data sources, assesses quality issues like missing values and duplicates, and implements resolution plans to ensure accurate and reliable analysis.

Section	Description
Data Overview	Dimension: 923 rows x 49 columns Descriptive Statistics: 
Univariate Analysis	 

	Unnamed: 0	latitude	longitude	labels	age_first_funding_year	age_last_funding_year	age_first_milestone_year	age_last_milestone_year	relationships	funding_rounds
count	923.000000	923.000000	923.000000	923.000000	923.000000	923.000000	771.000000	771.000000	923.000000	923.000000
mean	572.297941	38.517442	-103.539212	0.646804	2.235630	3.931456	3.055353	4.754423	7.710726	2.310943
std	333.585431	3.741497	22.394167	0.478222	2.510449	2.967910	2.977057	3.212107	7.265776	1.390922
min	1.000000	25.752358	-122.756956	0.000000	-9.046600	-9.046600	-14.169900	-7.005500	0.000000	1.000000
25%	283.500000	37.388869	-122.198732	0.000000	0.576700	1.669900	1.000000	2.411000	3.000000	1.000000
50%	577.000000	37.779281	-118.374037	1.000000	1.446600	3.528800	2.520500	4.476700	5.000000	2.000000
75%	866.500000	40.730646	-77.214731	1.000000	3.575300	5.560300	4.686300	6.753400	10.000000	3.000000
max	1153.000000	59.335232	18.057121	1.000000	21.895900	21.895900	24.684900	24.684900	63.000000	10.000000

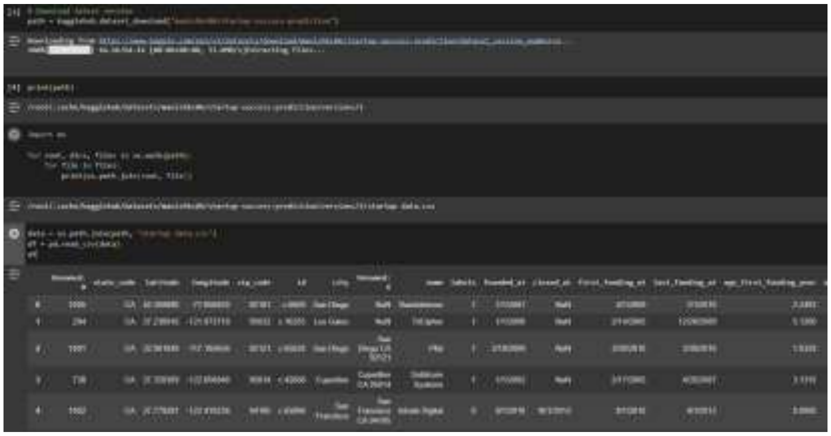
	
Bivariate Analysis	   

Multivariate Analysis



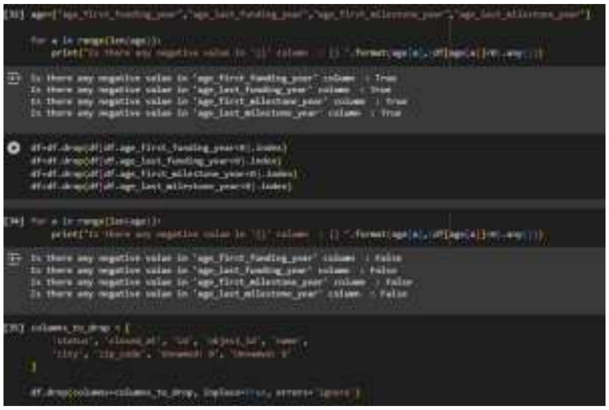
Data Preprocessing Code Screenshots

Loading Data



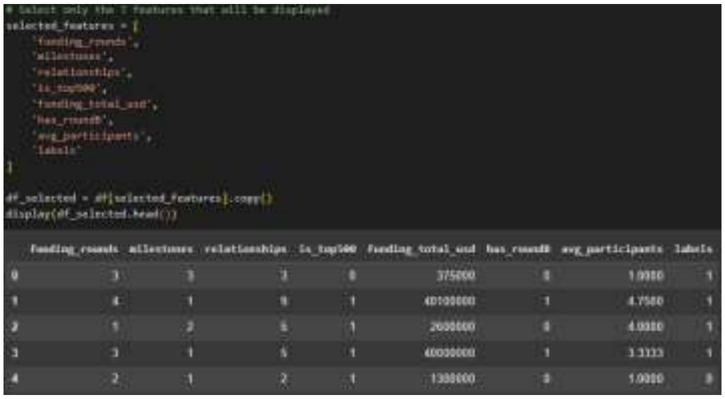
```
[1] # Download latest version
    path = kagglehub.dataset_download("[illegible]/startup-success-prediction")

    Downloading from [illegible]
    100%|[illegible]| [illegible]

[2] print(path)

    [illegible]

    import os

    for root, dirs, files in os.walk(path):
        for file in files:
            print(os.path.join(root, file))

    [illegible]/startup data.csv

    data = os.path.join(path, 'startup data.csv')
    df = pd.read_csv(data)
    df
```

Dropping Irrelevant Columns



```
[33] age=['age_first_funding_year','age_last_funding_year','age_first_milestone_year','age_last_milestone_year']

     for a in range(len(age)):
         print("Is there any negative value in '{}' column  : {} ".format(age[a],(df[age[a]]<0).any()))

     Is there any negative value in 'age_first_funding_year' column  : True
     Is there any negative value in 'age_last_funding_year' column  : True
     Is there any negative value in 'age_first_milestone_year' column  : True
     Is there any negative value in 'age_last_milestone_year' column  : True

     df=df.drop(df[df.age_first_funding_year<0].index)
     df=df.drop(df[df.age_last_funding_year<0].index)
     df=df.drop(df[df.age_first_milestone_year<0].index)
     df=df.drop(df[df.age_last_milestone_year<0].index)

[34] for a in range(len(age)):
         print("Is there any negative value in '{}' column  : {} ".format(age[a],(df[age[a]]<0).any()))

     Is there any negative value in 'age_first_funding_year' column  : False
     Is there any negative value in 'age_last_funding_year' column  : False
     Is there any negative value in 'age_first_milestone_year' column  : False
     Is there any negative value in 'age_last_milestone_year' column  : False

[35] columns_to_drop = [
         'status', 'closed_at', 'id', 'object_id', 'name',
         'city', 'zip_code', 'Unnamed: 0', 'Unnamed: 6'
     ]

     df.drop(columns=columns_to_drop, inplace=True, errors='ignore')
```

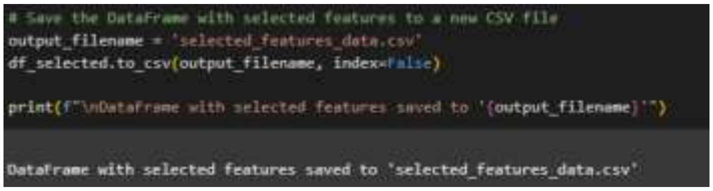
Feature Engineering



```
# Select only the 7 features that will be displayed
selected_features = [
    'funding_rounds',
    'milestones',
    'relationships',
    'is_top500',
    'funding_total_usd',
    'has_roundB',
    'avg_participants',
    'labels'
]

df_selected = df[selected_features].copy()
display(df_selected.head())
```

	funding_rounds	milestones	relationships	is_top500	funding_total_usd	has_roundB	avg_participants	labels
0	3	3	3	0	375000	0	1.0000	1
1	4	1	9	1	40100000	1	4.7500	1
2	1	2	5	1	2600000	0	4.0000	1
3	3	1	5	1	40000000	1	3.3333	1
4	2	1	2	1	1300000	0	1.0000	0

Save Processed Data



```
# Save the DataFrame with selected features to a new CSV file
output_filename = 'selected_features_data.csv'
df_selected.to_csv(output_filename, index=False)

print(f"\nDataFrame with selected features saved to '{output_filename}'")
```

```
DataFrame with selected features saved to 'selected_features_data.csv'
```