Data Collection and Preprocessing Phase

Date	26 September 2024
Team ID	LTVIP2024TMID24973
Project Title	Detection of Phishing Websites from URLs Using Machine learning
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.

1. Data Overview

- **Basic Statistics**:
    - Calculate the mean, median, and mode of numeric features (e.g., URL length, number of subdomains).
    - Count the number of phishing vs. legitimate URLs.
    - Check for missing values in important fields like URL and labels.

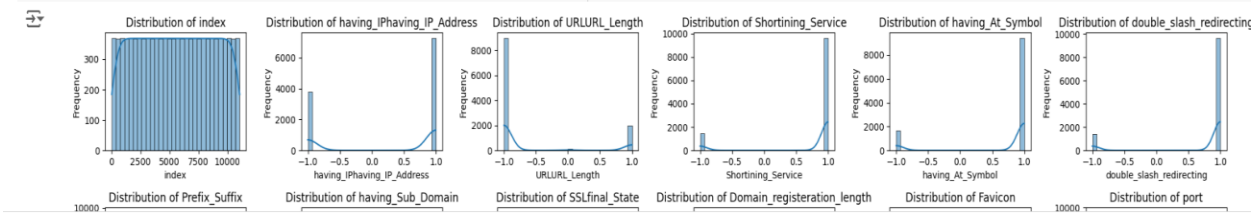
2. Univariate Analysis

Univariate Analysis

```
[11] import seaborn as sns
     import matplotlib.pyplot as plt

     num_cols = ds.select_dtypes(include=['float64', 'int64']).columns
     # Calculate the number of rows and columns for the subplot grid
     num_rows = (len(num_cols) + 5) // 6  # Calculate rows, ensuring enough space
     # +5 to round up for the leftover plots, if any

     plt.figure(figsize=(20, 15))
     for i, col in enumerate(num_cols, 1):
         plt.subplot(num_rows, 6, i)  # Dynamically adjust grid size
         sns.histplot(ds[col], bins=30, kde=True)
         plt.title(f'Distribution of {col}')
         plt.xlabel(col)
         plt.ylabel('Frequency')
     plt.tight_layout()
     plt.show()
```

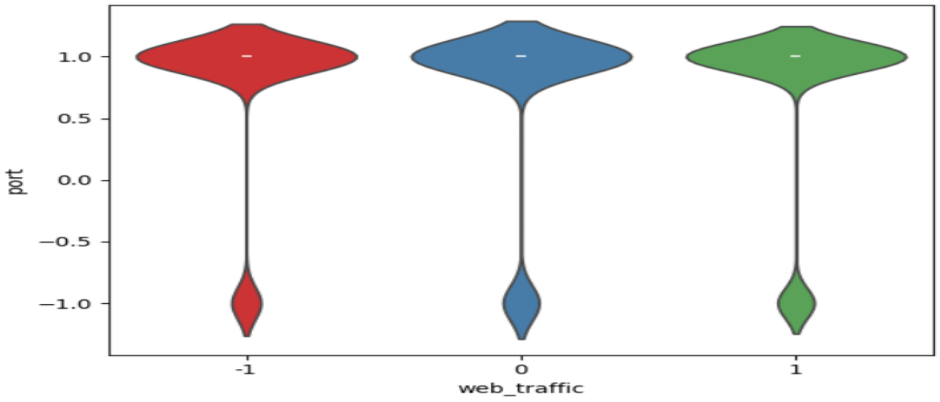


3. Bivariate Analysis

Bivariate Analysis

```
sns.violinplot(x = ds['web_traffic'],y = ds['port'], palette = 'Set1', data =ds)
```

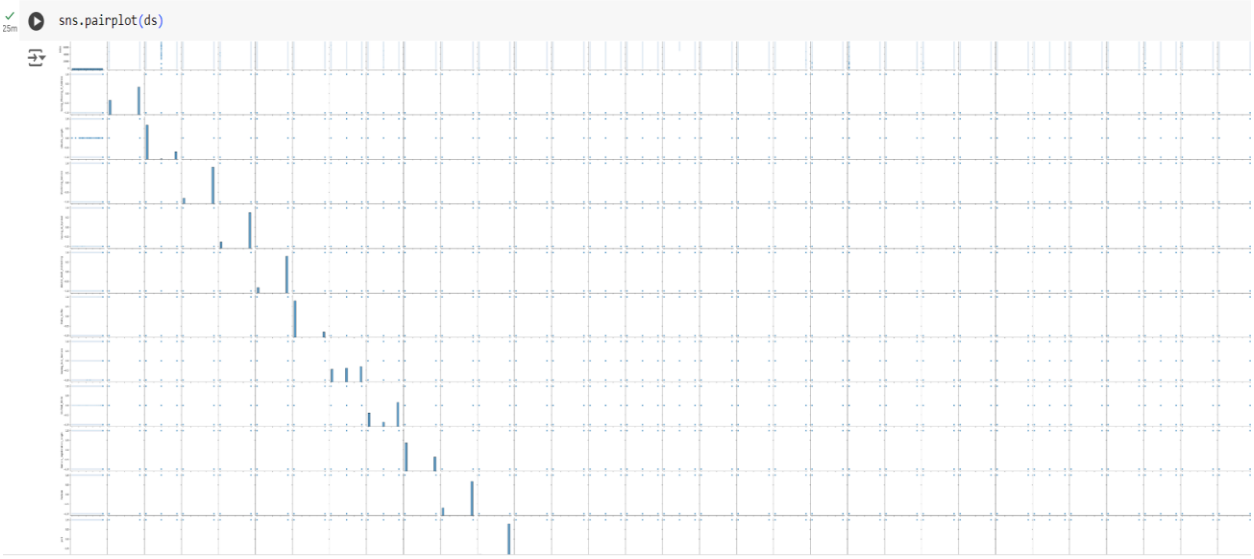
```
<Axes: xlabel='web_traffic', ylabel='port'>
```



4. Multivariate Analysis.

Multi Variate Analysis

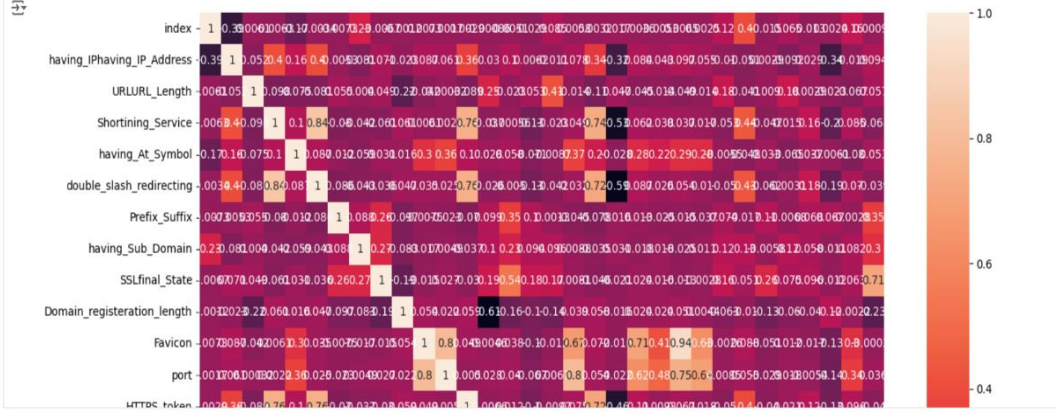
```
sns.pairplot(ds)
```



- **Visualizations**:
  - Generate heatmaps to show correlations among multiple variables.
  - Use 3D scatter plots if applicable to visualize interactions among three features.

- Visualizing the data

```
plt.figure(figsize=(15,15))
sns.heatmap(ds.corr(), annot=True)
plt.show()
```



Data Preprocessing Code Screenshots	
Loading Data	Loading the Dataset `#import dataset` `ds= pd.read_csv("phishing_website dataset.csv")` `ds.head()`

	index	having_IPhaving_IP_Address	URLURL_Length	Shortining_Service	having_At_Symbol	double_slash_redirecting	Prefix_Suffix	having_Sub_Domain	SSLfinal_State	Domain_registerat
0	1	-1	1	1	1	-1	-1	-1	-1	
1	2	1	1	1	1	1	-1	0	1	
2	3	1	0	1	1	1	-1	-1	-1	
3	4	1	0	1	1	1	-1	-1	-1	
4	5	1	0	-1	1	1	-1	1	1	

5 rows × 32 columns

Checking Null Values

```
#Checking null values
ds.info()
ds.isnull().any()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11055 entries, 0 to 11054
Data columns (total 32 columns):
 #   Column                       Non-Null Count  Dtype
---  ------                       --------------  -----
 0   index                        11055 non-null  int64
 1   having_IPhaving_IP_Address   11055 non-null  int64
 2   URLURL_Length                11055 non-null  int64
 3   Shortining_Service           11055 non-null  int64
 4   having_At_Symbol             11055 non-null  int64
 5   double_slash_redirecting     11055 non-null  int64
 6   Prefix_Suffix                11055 non-null  int64
 7   having_Sub_Domain            11055 non-null  int64
 8   SSLfinal_State               11055 non-null  int64
 9   Domain_registeration_length  11055 non-null  int64
 10  Favicon                      11055 non-null  int64
 11  port                         11055 non-null  int64
 12  HTTPS_token                  11055 non-null  int64
 13  Request_URL                  11055 non-null  int64
 14  URL_of_Anchor                11055 non-null  int64
 15  Links_in_tags                11055 non-null  int64
 16  SFH                          11055 non-null  int64
 17  Submitting_to_email          11055 non-null  int64
 18  Abnormal_URL                 11055 non-null  int64
 19  Redirect                     11055 non-null  int64
 20  on_mouseover                 11055 non-null  int64
 21  RightClick                   11055 non-null  int64
 22  popUpWidnow                  11055 non-null  int64
 23  Iframe                       11055 non-null  int64
```

```
[ ]  #Checking null values
     ds.info()
     ds.isnull().any()
```

```
dtypes: int64(32)
memory usage: 2.7 MB
```

	0
index	False
having_IPhaving_IP_Address	False
URLURL_Length	False
Shortining_Service	False
having_At_Symbol	False
double_slash_redirecting	False
Prefix_Suffix	False
having_Sub_Domain	False
SSLfinal_State	False
Domain_registeration_length	False
Favicon	False
port	False
HTTPS_token	False
Request_URL	False

Feature Extraction	**Feature Extration** Double-click (or enter) to edit `[ ]` ```from urllib.parse import urlparse,urlencode import ipaddress import re``` `[ ]` ```def getDomain(url): domain = urlparse(url).netloc if re.match(r"^www.",domain): domain = domain.replace("www.","") return domain``` ```def havingIP(url): try: ipaddress.ip_address(url) ip = 1 except: ip = 0 return ip``` **Visualizing the data** `[ ]` ```plt.figure(figsize=(15,15)) sns.heatmap(ds.corr(), annot=True) plt.show()```