Section	Description
Data Overview	Dimension: 614 rows × 13 columns Descriptive statistics:

6049	2009-01-01	Cobar	17.9	35.2	0.0	12.0	12.3	SSW	48.0	ENE	SW	6.0
6050	2009-01-02	Cobar	18.4	28.9	0.0	14.8	13.0	S	37.0	SSE	SSE	19.0
6052	2009-01-04	Cobar	19.4	37.6	0.0	10.8	10.6	NNE	46.0	NNE	NNW	30.0
6053	2009-01-05	Cobar	21.9	38.4	0.0	11.4	12.2	WNW	31.0	WNW	WSW	6.0
6054	2009-01-06	Cobar	24.2	41.0	0.0	11.2	8.4	WNW	35.0	NW	WNW	17.0
...	...	...	...	...	...	...	...	...	...	...	...	...
142298	2017-06-20	Darwin	19.3	33.4	0.0	6.0	11.0	ENE	35.0	SE	NE	9.0
142299	2017-06-21	Darwin	21.2	32.6	0.0	7.6	8.6	E	37.0	SE	SE	13.0
142300	2017-06-22	Darwin	20.7	32.8	0.0	5.6	11.0	E	33.0	E	W	17.0

	142301 2017-06-23 Darwin 19.5 31.8 0.0 6.2 10.6 ESE 26.0 SE NNW 9.0
Univariate Analysis	

Data Collection and Preprocessing Phase

Date	20 June 2024
Team ID	739723
Project Title	Rain fall prediction using ml
Maximum Marks	6 Marks

Data Exploration and Preprocessing Report

Data exploration involved identifying patterns and outliers. Preprocessing included normalization, handling missing values, and feature engineering. These steps ensured high-quality data for accurate modeling and insightful predictions.

	-
Bivariate Analysis	-

Multivariate Analysis	-

Outliers and Anomalies	-
Data Preprocessing Code Screenshots	
Loading Data	

```
#importing the dataset which is in csv file
data = pd.read_csv('/content/Dataset/loan_prediction.csv')
data
```

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome
0	LP001002	Male	No	0	Graduate	No	5849	0.0
1	LP001003	Male	Yes	1	Graduate	No	4583	1508.0
2	LP001005	Male	Yes	0	Graduate	Yes	3000	0.0
3	LP001006	Male	Yes	0	Not Graduate	No	2583	2358.0
4	LP001008	Male	No	0	Graduate	No	6000	0.0

<table>
<tr><td>Handling Missing Data</td><td>

```
data['Gender'] = data['Gender'].fillna(data['Gender'].mode()[0])
```

```
data['Married'] = data['Married'].fillna(data['Married'].mode()[0])
```

```
#replacing + with space for filling the nan values
data['Dependents']=data['Dependents'].str.replace('+','')
```

```
<ipython-input-71-6ac39c248773>:2: FutureWarning: The default value of regex will change from
  data['Dependents']=data['Dependents'].str.replace('+','')
```

```
data['Dependents'] = data['Dependents'].fillna(data['Dependents'].mode()[0])
```

```
data['Self_Employed'] = data['Self_Employed'].fillna(data['Self_Employed'].mode()[0])
```

```
data['LoanAmount'] = data['LoanAmount'].fillna(data['LoanAmount'].mode()[0])
```

```
data['Loan_Amount_Term'] = data['Loan_Amount_Term'].fillna(data['Loan_Amount_Term'].mode()[0])
```

```
data['Credit_History'] = data['Credit_History'].fillna(data['Credit_History'].mode()[0])
```

</td></tr>
<tr><td>Data Transformation</td><td>

```
data['Gender']=data['Gender'].map({'Female':1,'Male':0})
data['Property_Area']=data['Property_Area'].map({'Urban':2,'Semiurban': 1,'Rural':0})
data['Married']=data['Married'].map({'Yes':1,'No':0})
data['Education']=data['Education'].map({'Graduate':1,'Not Graduate':0})
data['Loan_Status']=data['Loan_Status'].map({'Y':1,'N':0})
```

```
# perfroming feature Scaling op[eration using standard scaller on X part of the dataset because
# there different type of values in the columns
sc=StandardScaler()
x_bal=sc.fit_transform(x_bal)
```

</td></tr>
<tr><td>Feature Engineering</td><td>Attached the codes in final submission.</td></tr>
<tr><td>Save Processed Data</td><td>-</td></tr>
</table>