1. INTRODUCTION

Objective of this Notebook:

This notebook aims to:

- Easy and ***Begginers guide***.
- Analyse Each and Every ***Attributes*** in the data set.
- Build Various ***ML Models*** with the view of ***increasing accuracy*** of the Model.

2.IMPORTING THE REQUIRED LIBRARIES

```
import matplotlib.pyplot as plt
import seaborn as sns
import scipy
import re
import missingno as mso
from scipy import stats
from scipy.stats import ttest_ind
from scipy.stats import pearsonr
from sklearn.preprocessing import StandardScaler,LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC
from sklearn.ensemble import GradientBoostingClassifier
from xgboost import XGBClassifier
from sklearn.metrics import
accuracy_score,confusion_matrix,classification_report
```

3.ANALYSING THE DATASET

There are **6 Variables** in this Dataset:

- **4 Continuous** Variables.
- **1 Variable** to accommodate the Date.
- **1 Variable** refers the Weather.

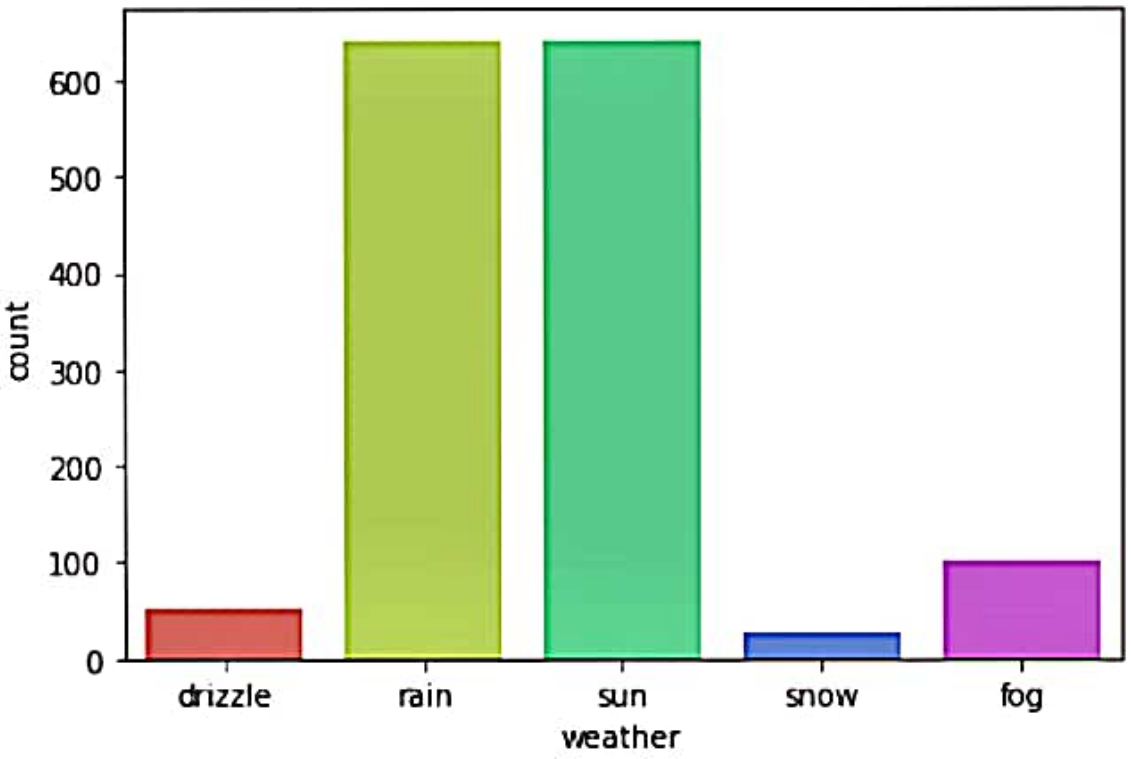
4.DATA EXPLORATION

It is the process of Exploring the data from the **"RAW"** data set tha we have taken or Imported.

First let us Deal with the Categorical variables

```
import warnings
warnings.filterwarnings('ignore')
sns.countplot("weather",data=data,palette="hls")
```

```
<AxesSubplot:xlabel='weather', ylabel='count'>
```

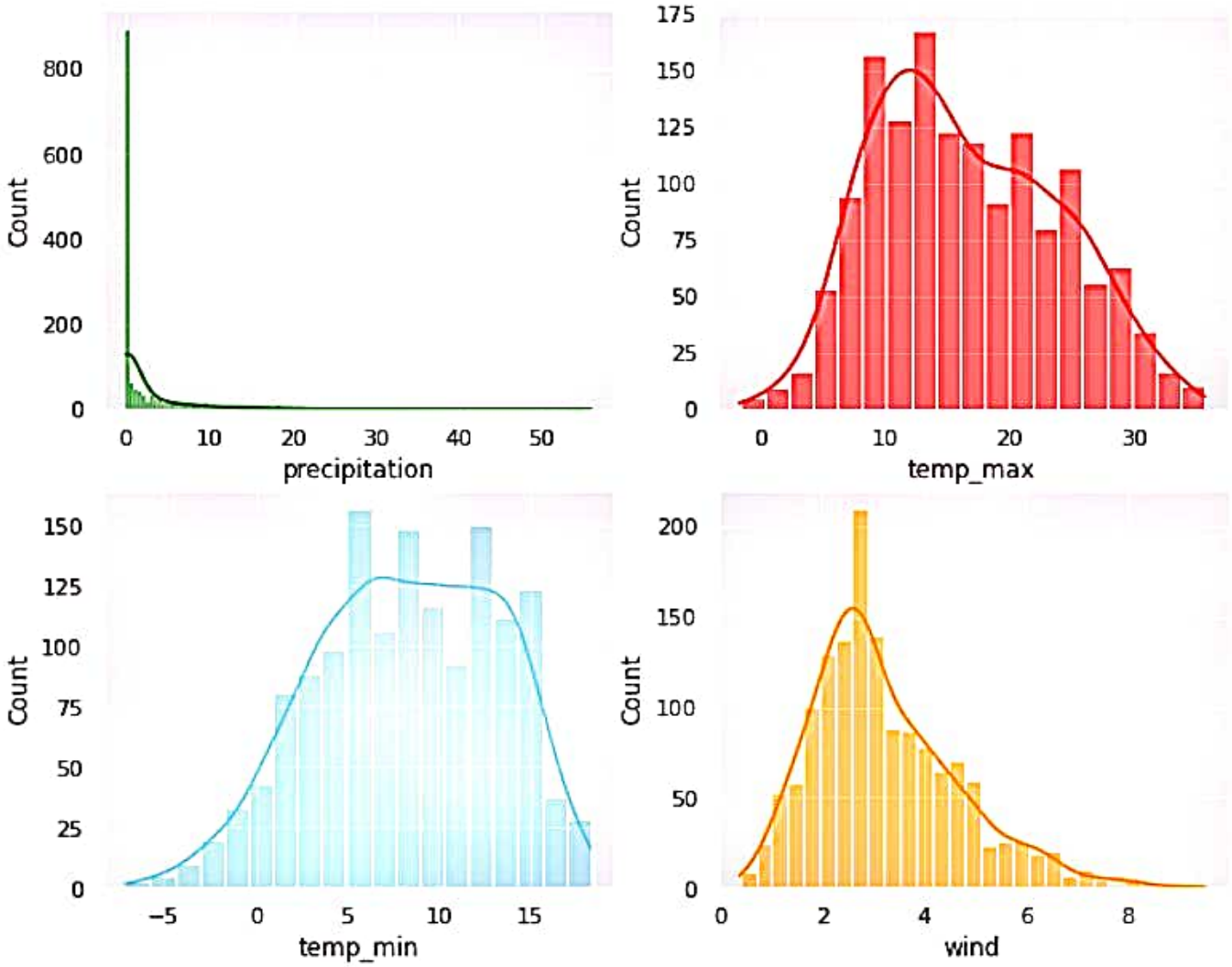

600
500
400
300
200
100
0
count
drizzle
rain
sun
snow
fog
weather

5.NUMERICAL OR CONTINUOUS VARIABLES

Next we will explore the ***Continuous variables***

```
data[["precipitation","temp_max","temp_min","wind"]].describe()
```

	precipitation	temp_max	temp_min	wind
count	1461.000000	1461.000000	1461.000000	1461.000000
mean	3.029432	16.439083	8.234771	3.241136
std	6.680194	7.349758	5.023004	1.437825
min	0.000000	-1.600000	-7.100000	0.400000
25%	0.000000	10.600000	4.400000	2.200000
50%	0.000000	15.600000	8.300000	3.000000
75%	2.800000	22.200000	12.200000	4.000000
max	55.900000	35.600000	18.300000	9.500000



From the above distribution it is clear that **precipitation and wind** are **Positively skewed.**

And **temp_min** is **Negatively skewed** and both has some ***outliers.***

6.HOW TO FIND THE OUTILERS OR SKEW IN DATA SET?

- ***We can find the outliers in the dataset by using following plots:***

  1.Hist plot

  2.Box plot

  3.Violin plot

  4.Dist plot yet both ***box and violin plots*** are easier to handel with.

7.NULL VALUES:

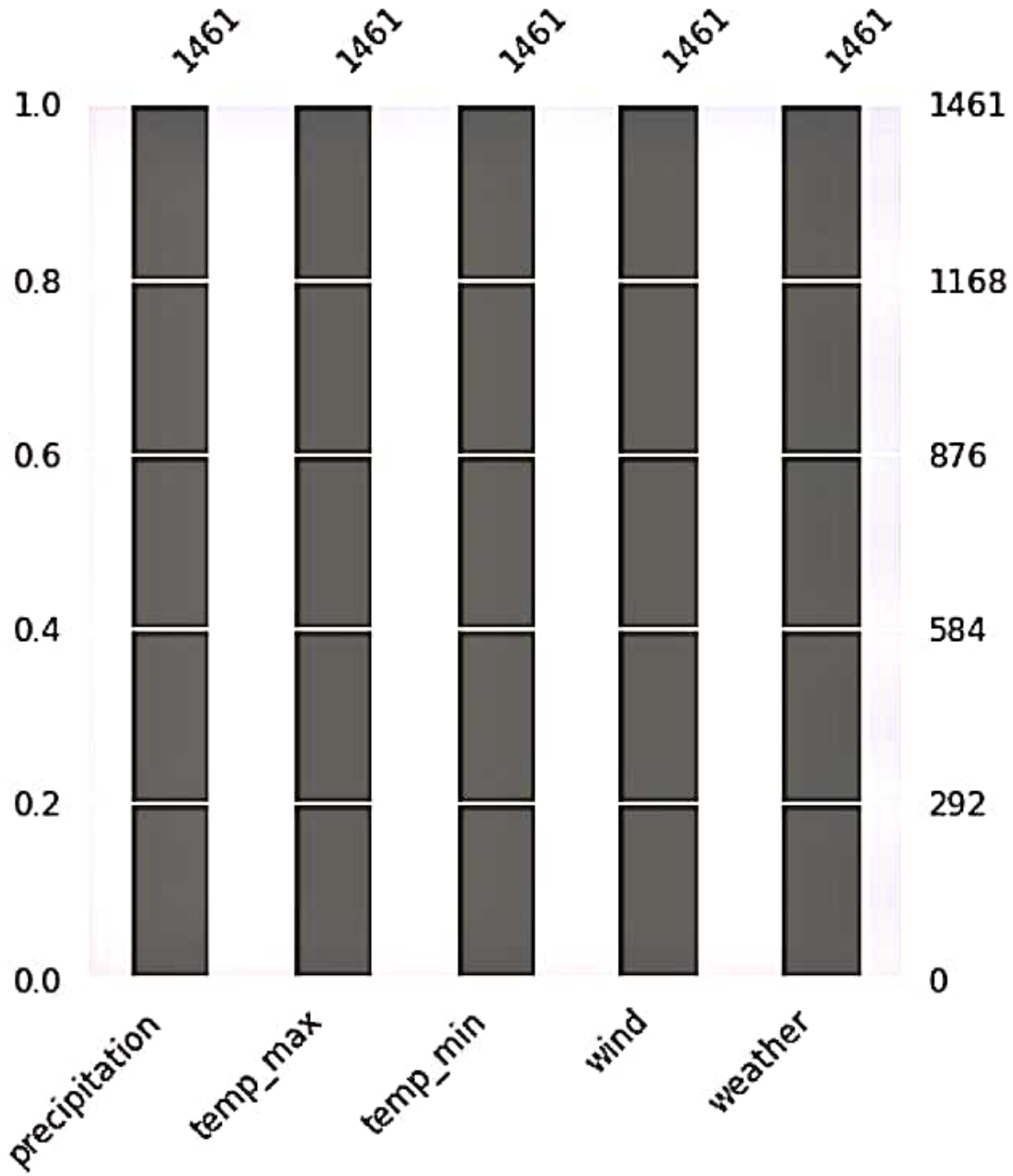
```
data.isna().sum()
```

```
date             0
precipitation    0
temp_max         0
temp_min         0
wind             0
weather          0
dtype: int64
```

Checking for Null values in the data set

The below plot shows that all the columns in the data set ***doesn't contains Null values*** as each columns contains a ***total of 1461*** observations.

```
plt.figure(figsize=(12,6))
axz=plt.subplot(1,2,2)
mso.bar(data.drop(["date"],axis=1),ax=axz,fontsize=12);
```


1461
1461
1461
1461
1461
1.0
0.8
0.6
0.4
0.2
0.0
1461
1168
876
584
292
0
precipitation
temp_max
temp_min
wind
weather

8.DATA PREPROCESSING:

Drop Unnecessary Variables

In this data set Date is a unnecessary variable as it does not affect the data so it can be dropped.

```
df=data.drop(["date"],axis=1)
```

Remove Outliers & Infinite Values

Since this dataset contains ***Outliers ,it will be removed***,to make data set more even.