Vegetation Document :

Data frame : (DATA JAN 2022 - MARCH 2023)

```
df = pd.read_csv('/content/DATA JAN 2022 - MARCH 2023.csv')
```

	Month	Days	YEAR	Date	Temperature	Min_Temperature	Max_Temperature	Humidity	Precipitation	Surface_Pressure	Wind_speed
0	Jan	1	2022	01-01-2022	21.95	17.12	27.66	13.24	0.0	94.21	2.05
1	jan	2	2022	02-01-2022	22.23	16.81	28.35	12.76	0.0	94.07	2.03
2	Jan	3	2022	03-01-2022	21.78	16.42	28.05	12.15	0.0	94.03	1.85
3	Jan	4	2022	04-01-2022	21.82	16.48	28.43	11.35	0.0	94.02	1.97
4	Jan	5	2022	05-01-2022	20.19	13.77	27.81	10.56	0.0	94.02	1.30

Vegetation Density was calculated with respect to temperature , humidity , precipitation and wind speed .

```
vegetation_density = 0.3 * df['Temperature'] + 0.2 * df['Humidity'] + 0.1 * df['Precipitation'] + 0.05 * df['Wind_speed']

df['Vegetation Density'] = vegetation_density
```

Vegetation Density was rescaled for the range between 0 to 1 .

```
max_value = vegetation_density.max()
min_value = vegetation_density.min()
rescaled_vegetation_density = (vegetation_density - min_value) / (max_value - min_value)

# Step 4: Add Rescaled Vegetation Density to the DataFrame
df['Rescaled Vegetation Density'] = rescaled_vegetation_density
```

Training and testing of data .

```
X = df[['Temperature', 'Humidity', 'Precipitation', 'Wind_speed']]
y = df['Rescaled Vegetation Density']
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
rf_model = RandomForestRegressor(n_estimators=100, random_state=42)
rf_model.fit(X_train, y_train)
```

```
RandomForestRegressor
RandomForestRegressor(random_state=42)
```

```
# Step 5: Model Training and Evaluation
y_pred = rf_model.predict(X_test)

```

Checking performance of the model :

```
mse = mean_squared_error(y_test, y_pred)
print(f"Mean Squared Error: {mse}")
```

```
Mean Squared Error: 0.00026715498697174595
```

THRESHOLD Value used to differentiate the vegetation

```
# Defining the Threshold Values
dense_threshold = 0.36
sparse_threshold = 0.2

# Classifing Dense and Sparse Vegetation
df['Vegetation Class'] = 'Moderate'
df.loc[df['Rescaled Vegetation Density'] >= dense_threshold, 'Vegetation Class'] = 'Dense'
df.loc[df['Rescaled Vegetation Density'] <= sparse_threshold, 'Vegetation Class'] = 'Sparse'

```

Calculated for Jan 2022 to march 2023 .

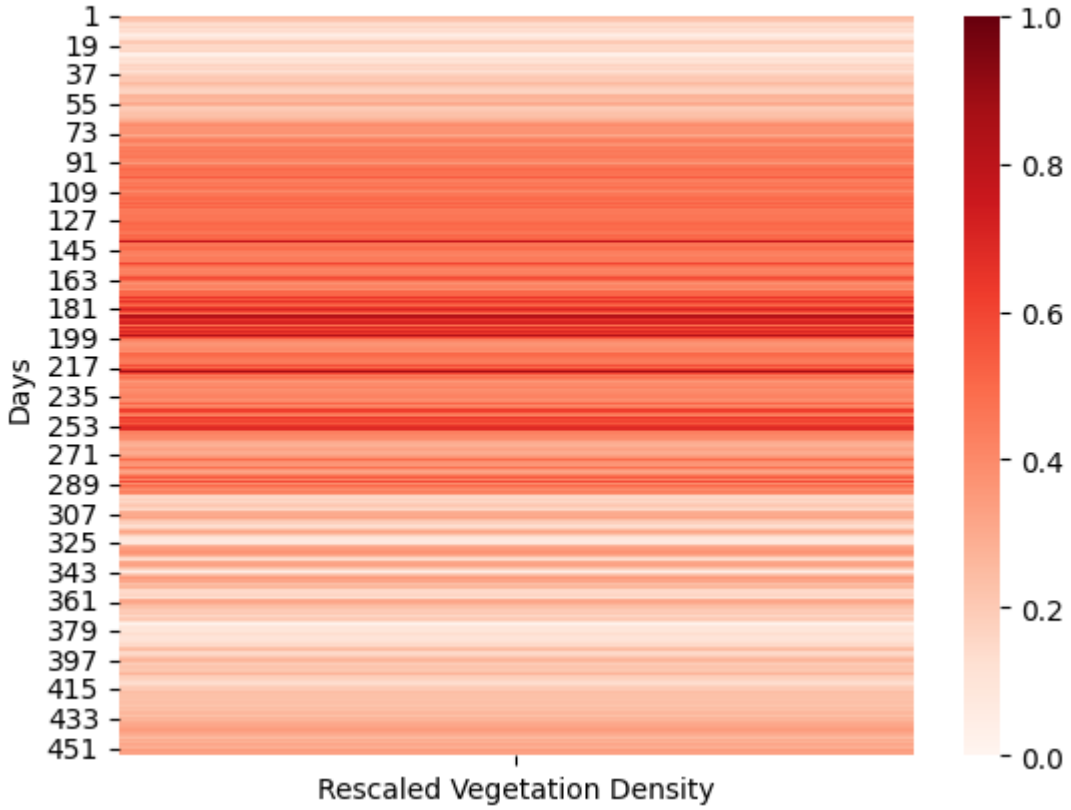
```
[157]  1 print(df)
```

	Month	Days	YEAR	Date	Temperature	Min_Temperature	Max_Temperature	Humidity	Precipitation	Surface_Pressure	Wind_speed	Vegetation Density	Rescaled Vegetation Density	Vegetation Class
0	Jan	1	2022	2022-01-01	21.95	17.12	27.66	13.24	0.00	94.21	2.05	9.3355	0.250748	Moderate
1	jan	2	2022	2022-01-02	22.23	16.81	28.35	12.76	0.00	94.07	2.03	9.3225	0.249452	Moderate
2	Jan	3	2022	2022-01-03	21.78	16.42	28.05	12.15	0.00	94.03	1.85	9.0565	0.222931	Moderate
3	Jan	4	2022	2022-01-04	21.82	16.48	28.43	11.35	0.00	94.02	1.97	8.9145	0.208774	Moderate
4	Jan	5	2022	2022-01-05	20.19	13.77	27.81	10.56	0.00	94.02	1.30	8.2340	0.140927	Sparse
..	...	...	...	...	...	...	...	...	...	...	...	...	...	...
450	Mar	451	2023	2023-03-27	25.69	16.69	37.17	9.28	0.00	93.70	2.38	9.6820	0.285294	Moderate
451	Mar	452	2023	2023-03-28	26.85	17.60	37.91	10.68	0.08	93.68	2.02	10.3000	0.346909	Moderate
452	Mar	453	2023	2023-03-29	26.76	18.22	37.31	9.83	0.00	93.74	2.29	10.1085	0.327817	Moderate
453	Mar	454	2023	2023-03-30	26.17	17.93	37.24	10.80	0.09	93.68	2.27	10.1335	0.330309	Moderate
454	Mar	455	2023	2023-03-31	25.82	19.02	35.37	11.29	0.21	93.66	2.42	10.1460	0.331555	Moderate

[455 rows x 14 columns]

Heat Map is Used to find the vegetation

```
NOte :
The Range between 0.2 - 0.36 is Sparse Vegetaion
The Range between 0.36 - 0.8 is Dense Vegetaion
```



For whole year we calculated the mean Vegetation Density and its average for vegetation class .

Month	Mean Vegetation Density	Average Vegetation Class
1	0.15	Sparse
2	0.21	sparse
3	0.33	sparse
4	0.46	Dense
5	0.49	Dense
6	0.51	Dense
7	0.55	Dense
8	0.49	Dense
9	0.44	Dense
10	0.36	Dense
11	0.23	sparse
12	0.23	sparse