

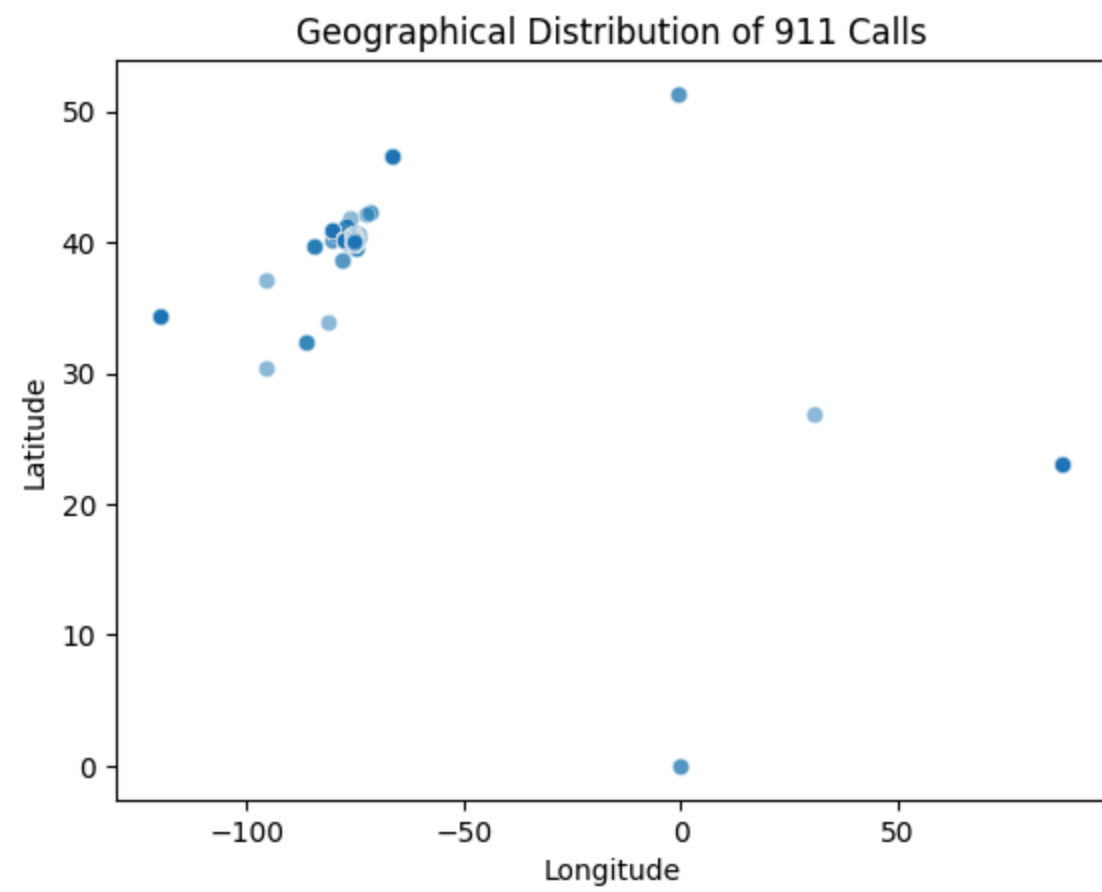
# Lab 8:Advance Chart

1. Analyze and Plot your Geographical information of 911 dataset.

```
In [29]: import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
from pandas.plotting import parallel_coordinates
```

```
In [3]: df = pd.read_csv("911.csv")
df['timeStamp'] = pd.to_datetime(df['timeStamp'])
```

```
In [5]: sns.scatterplot(x=df['lng'], y=df['lat'], alpha=0.5)
plt.title('Geographical Distribution of 911 Calls')
plt.xlabel('Longitude')
plt.ylabel('Latitude')
plt.show()
```



2. Analyze the wine dataset and present your analysis with an advanced chart (parallel coordinate chart, Ternary chart or sunburst chat).

```
In [25]: df = pd.read_csv("Wine-2023.csv")  
df.info()
```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 178 entries, 0 to 177
Data columns (total 14 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Wine                  178 non-null   int64
1   Alcohol               178 non-null   float64
2   Malic.acid            178 non-null   float64
3   Ash                   178 non-null   float64
4   Acl                   178 non-null   float64
5   Mg                    178 non-null   int64
6   Phenols               178 non-null   float64
7   Flavanoids            178 non-null   float64
8   Nonflavanoid.phenols  178 non-null   float64
9   Proanth               178 non-null   float64
10  Color.int              178 non-null   float64
11  Hue                   178 non-null   float64
12  OD                    178 non-null   float64
13  Proline               178 non-null   int64
dtypes: float64(11), int64(3)
memory usage: 19.6 KB

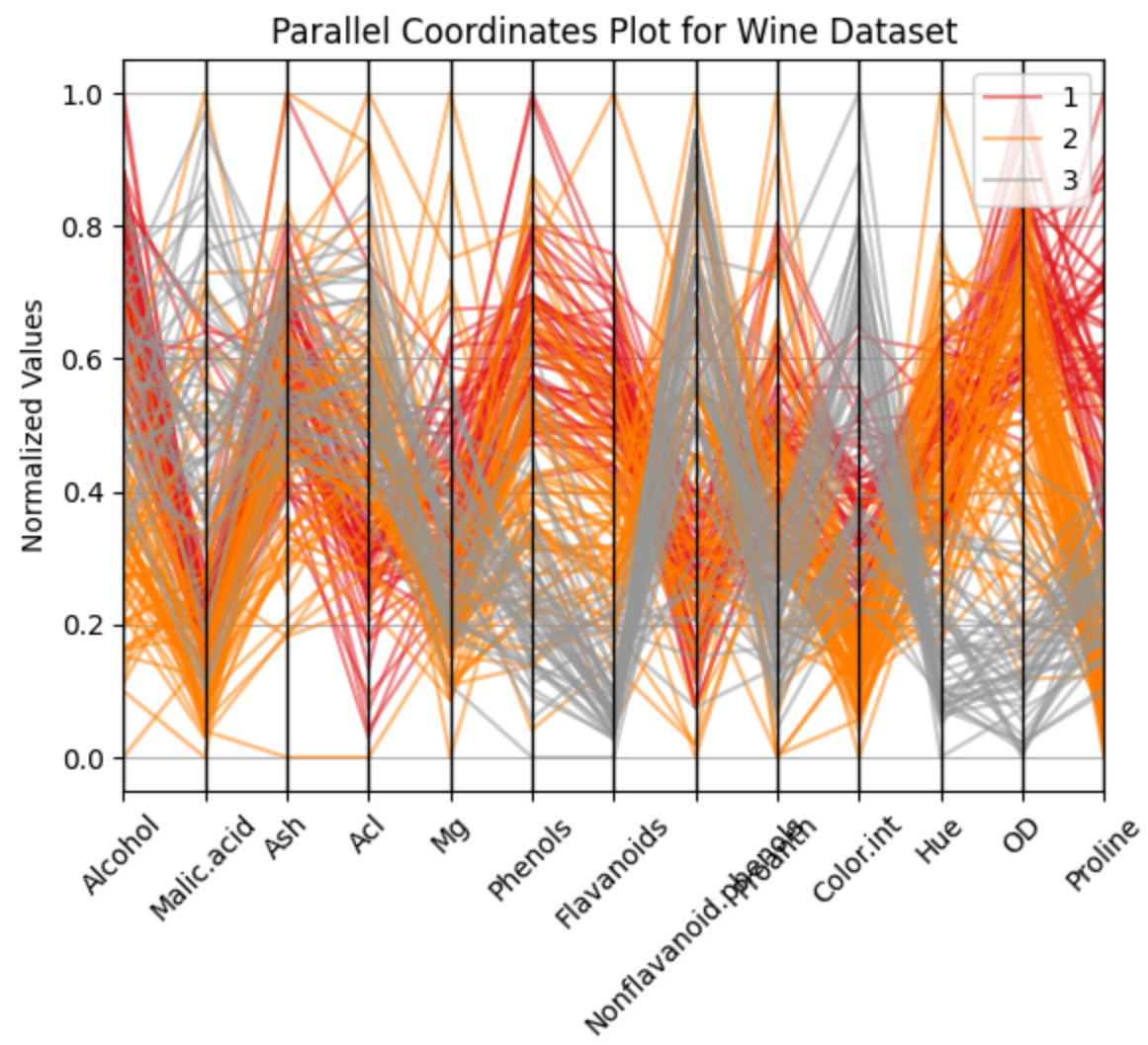
```

```

In [33]: cols_to_normalize = df.columns[1:]
df[cols_to_normalize] = (df[cols_to_normalize] - df[cols_to_normalize].min()) / (df[cols_to_normalize].max() - df[cols_to_normalize].min())

df["Wine"] = df["Wine"].astype(str)
parallel_coordinates(df, class_column="Wine", colormap=plt.get_cmap("Set1"), alpha=0.5)
plt.xticks(rotation=45)
plt.title("Parallel Coordinates Plot for Wine Dataset")
plt.ylabel("Normalized Values")
plt.show()

```



In [ ]: