Analyzing Electric Vehicle Populations: Range and Total Cars by Make and Model

By: Luis Mireles

In this analysis, we explore data of electric vehicles by make in Washington State. This code analyzes data on electric vehicles to determine the top electric car models and manufacturers based on their electric range and total number of cars produced. The code loads the data from a CSV file and filters out electric vehicles with a zero electric range. It then calculates the total number of electric cars by model year, make, and model and ranks the top 10 car models by total number of electric cars produced. The code also generates statistics on the mean, max, min, and standard deviation of electric range by make and model and ranks the top 10 electric car models by mean electric range. A box plot is then generated for the top 10 electric car manufacturers, showing the distribution of mean electric range for each manufacturer. Lastly, a scatter plot is created to display the relationship between mean range and total number of cars produced, with each car make represented by a different color.

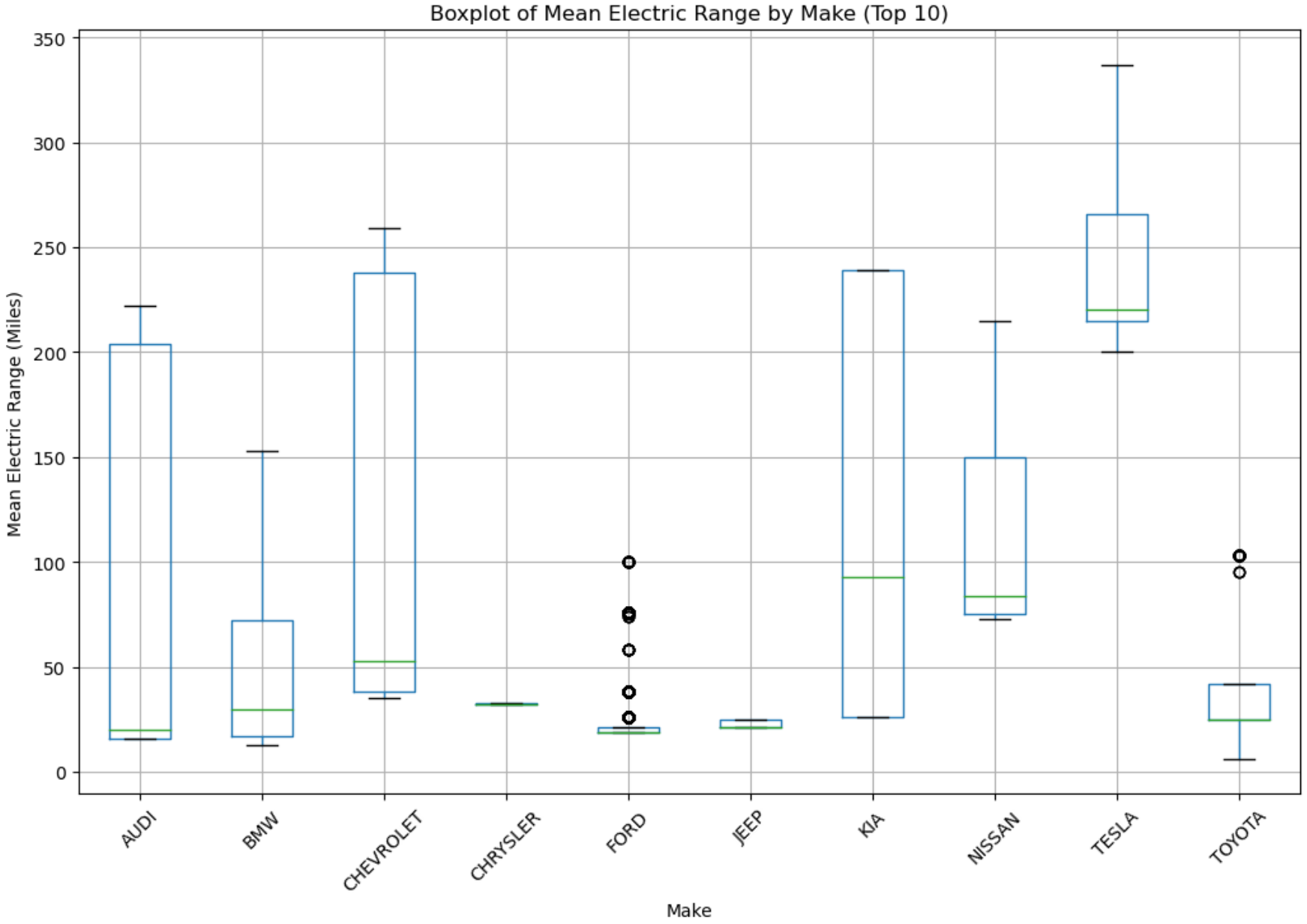
source data: https://catalog.data.gov/dataset/electric-vehicle-population-data

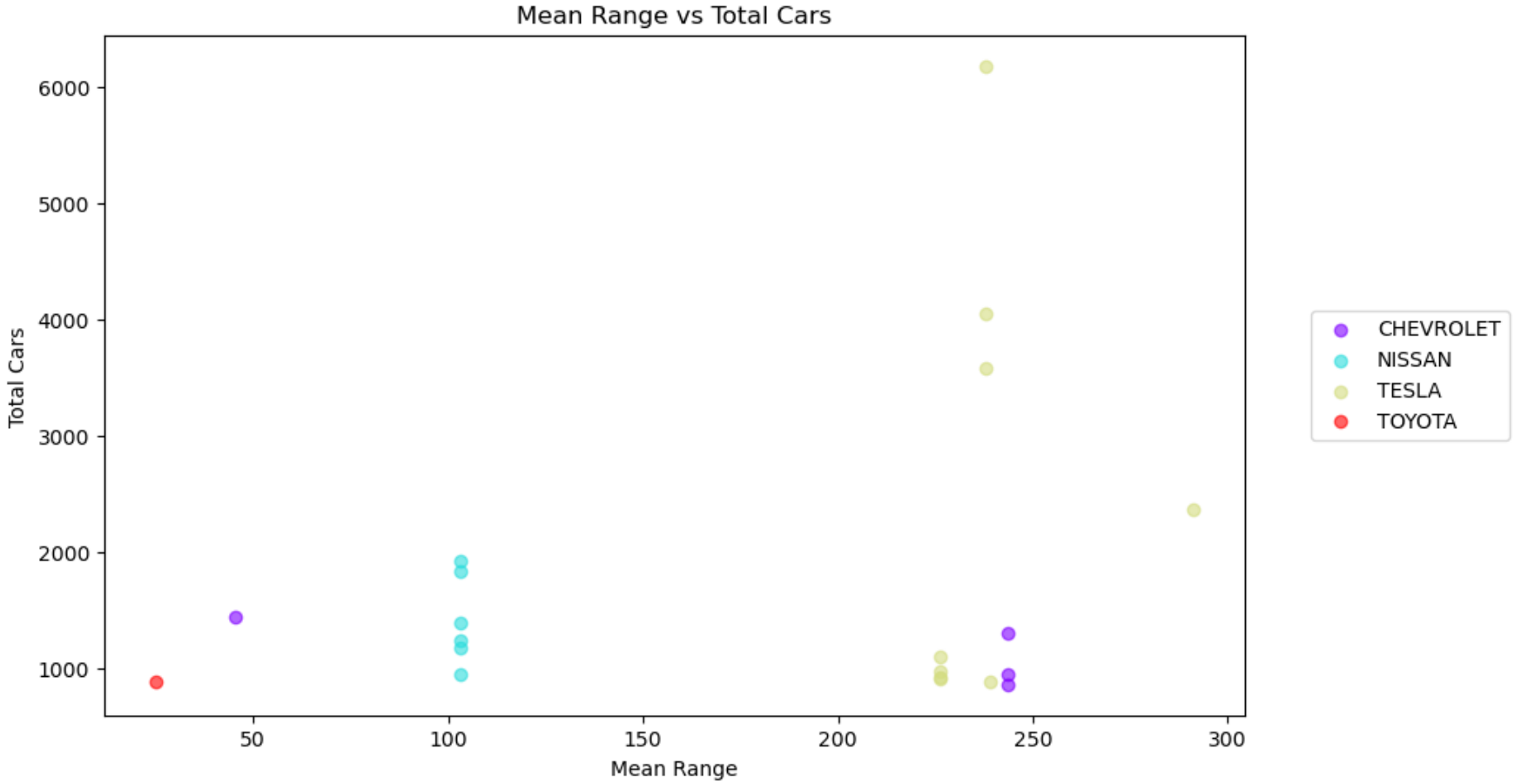
In [4]:
```
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.cm as cm
import numpy as np

file_path = '/Users/fernando/Desktop/Code/Sources/Electric_Vehicle_Population_Data.csv'
data = pd.read_csv(file_path)

data_filtered = data[data['Electric Range'] != 0]

counted_data = (
    data_filtered
    .groupby(['Model Year','Make', 'Model'])
    .size()
```

```
    .reset_index(name='TotalCars')
    .sort_values('TotalCars', ascending=False)
    .reset_index(drop=True)
)

# I assign the rank column and start it from 1 up to length of data set
counted_data['Rank'] = range(1, len(counted_data) + 1)
rank_col = counted_data.pop('Rank')
counted_data.insert(0, 'Rank', rank_col)

# Creating the range with stats view
range_stats = (
    data_filtered.groupby(['Make','Model'])
    ['Electric Range']
    .describe()
    .loc[:, ['mean', 'max', 'min', 'std']]
    .sort_values('mean', ascending=False)
)

formatted_range_stats = range_stats.head(10).reset_index()

formatted_range_stats['mean'] = formatted_range_stats['mean'].apply(lambda x: f"{x:.2f}")
formatted_range_stats['max'] = formatted_range_stats['max'].apply(lambda x: f"{x:.2f}")
formatted_range_stats['min'] = formatted_range_stats['min'].apply(lambda x: f"{x:.2f}")
formatted_range_stats['std'] = formatted_range_stats['std'].apply(lambda x: f"{x:.2f}")

formatted_range_stats = formatted_range_stats.rename(columns={'mean': 'MeanRange',
                        'max': 'MaxRange',
                        'min': 'MinRange',
                        'std': 'StandardDeviation'})

formatted_range_stats['Rank'] = range(1, len(formatted_range_stats) + 1)
rank_col = formatted_range_stats.pop('Rank')
formatted_range_stats.insert(0, 'Rank', rank_col)


print("Top 10 Total Cars Data:")
print(counted_data.head(10))
print("Range Stats by mean range rank:\n", formatted_range_stats)
```

```
# Box Plot
make_counts = data_filtered['Make'].value_counts().sort_values(ascending=False)
top_10_makes = make_counts.head(10).index
data_filtered_top_10_makes = data_filtered[data_filtered['Make'].isin(top_10_makes)]

data_filtered_top_10_makes.boxplot(column='Electric Range', by='Make', figsize=(12,8))
plt.title('Boxplot of Mean Electric Range by Make (Top 10)')
plt.suptitle('')
plt.ylabel('Mean Electric Range (Miles)')
plt.xlabel('Make')
plt.xticks(rotation=45)
plt.show()

merged_data = pd.merge(range_stats, counted_data,  on=['Make', 'Model'])

merged_data = merged_data.sort_values('TotalCars', ascending=False).head(20)

fig, ax = plt.subplots(figsize=(10, 6))

colors = cm.rainbow(np.linspace(0, 1, len(merged_data['Make'].unique())))

for i, (make, group) in enumerate(merged_data.groupby('Make')):
    ax.scatter(group['mean'], group['TotalCars'], color=colors[i], alpha=0.6, label=make)

ax.set_title('Mean Range vs Total Cars')
ax.set_xlabel('Mean Range')
ax.set_ylabel('Total Cars')

ax.legend(loc='center left', bbox_to_anchor=(1.05, 0.5))

plt.show()
```

```
Top 10 Total Cars Data:
   Rank  Model Year       Make    Model  TotalCars
0     1         2018      TESLA  MODEL 3       6179
1     2         2019      TESLA  MODEL 3       4052
2     3         2020      TESLA  MODEL 3       3583
3     4         2020      TESLA  MODEL Y       2369
4     5         2013     NISSAN     LEAF       1925
5     6         2015     NISSAN     LEAF       1831
6     7         2017  CHEVROLET     VOLT       1433
7     8         2019     NISSAN     LEAF       1384
8     9         2017  CHEVROLET  BOLT EV       1296
9    10         2018     NISSAN     LEAF       1233
Range Stats by mean range rank:
    Rank       Make             Model MeanRange MaxRange MinRange  \
0     1      TESLA           MODEL Y    291.00   291.00   291.00
1     2    HYUNDAI              KONA    258.00   258.00   258.00
2     3  CHEVROLET           BOLT EV    243.61   259.00   238.00
3     4      TESLA           MODEL X    239.25   293.00   200.00
4     5      TESLA           MODEL 3    237.86   322.00   215.00
5     6     JAGUAR            I-PACE    234.00   234.00   234.00
6     7      TESLA          ROADSTER    234.00   245.00   220.00
7     8   POLESTAR               PS2    233.00   233.00   233.00
8     9      TESLA           MODEL S    226.33   337.00   208.00
9    10       AUDI  E-TRON SPORTBACK    218.00   218.00   218.00

  StandardDeviation
0              0.00
1              0.00
2              9.30
3             38.33
4             37.85
5              0.00
6             12.54
7              0.00
8             33.24
9              0.00
```


Boxplot of Mean Electric Range by Make (Top 10)
Mean Electric Range (Miles)
0
50
100
150
200
250
300
350
AUDI
BMW
CHEVROLET
CHRYSLER
FORD
JEEP
KIA
NISSAN
TESLA
TOYOTA
Make


Mean Range vs Total Cars
Total Cars
Mean Range
6000
5000
4000
3000
2000
1000
50
100
150
200
250
300
CHEVROLET
NISSAN
TESLA
TOYOTA

In [ ]: