Vashrith Vinodh

Problem Statement:

Do the top 5 manufacturers (by the number of vehicles) have distinct patterns in their models, prices, and electric ranges? Are there common characteristics among the models within each manufacturer that contribute to their popularity?

Import Libraries

In [124…

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

Data Dictionary

1. Make

Description: The manufacturer or brand of the vehicle.

Type: String

Example: `Tesla`, `BMW`, `Nissan`

2. EV_Type

Description: The type of electric vehicle. This can be either Battery Electric Vehicle (BEV) or Plug-in Hybrid Electric Vehicle (PHEV).

Type: String

Example: `Battery Electric Vehicle (BEV)`, `Plug-in Hybrid Electric Vehicle (PHEV)`

3. Electric Range

Description: The electric driving range of the vehicle, measured in miles, before it needs recharging.

Type: Integer

Example: `215`

4. Base MSRP

Description: The base manufacturer's suggested retail price for the vehicle in USD, without additional options or features.

Type: Integer

Example: `35000`

Load Data

In [125…

```
df = pd.read_csv("Electric_Vehicle_Population_Data.csv")
```

Data Transformation

In [126…

```
df.rename(columns={
    'VIN (1-10)': 'VIN',
    'Model Year': 'Model_Year',
    'Postal Code': 'Postal_Code',
    'Electric Vehicle Type': 'EV_Type',
    'Clean Alternative Fuel Vehicle (CAFV) Eligibility': 'CAFV_Eligibility',
    'Electric Range': 'Electric_Range',
    'Base MSRP': 'Base_MSRP',
    'Legislative District': 'Legislative_District',
    'DOL Vehicle ID': 'DOL_Vehicle_ID',
    'Vehicle Location': 'Vehicle_Location',
    'Electric Utility': 'Electric_Utility',
    '2020 Census Tract': 'Census_Tract'
}, inplace=True)
```

In [127…

```
brand_counts = df['Make'].value_counts()
df['Brand_Count'] = df['Make'].map(brand_counts)
top_5_brands = brand_counts.nlargest(5).index
df_top_brands = df[df['Make'].isin(top_5_brands)]
df['Latitude'] = df['Vehicle_Location'].str.extract(r'POINT \(-?\d+\.\d+ (-?\d+\.\d
df['Longitude'] = df['Vehicle_Location'].str.extract(r'POINT \((-?\d+\.\d+) -?\d+\.
df['Make'] = df['Make'].str.upper()
df['Model'] = df['Model'].str.upper()
df['EV_Category'] = df['EV_Type'].apply(lambda x: 'BEV' if 'Battery' in x else 'PHE
```

Data Cleaning

In [128…

```
df.drop_duplicates(inplace=True)
df = df[df['Electric_Range'] > 0]
df = df[df['Base_MSRP'] > 0]
```

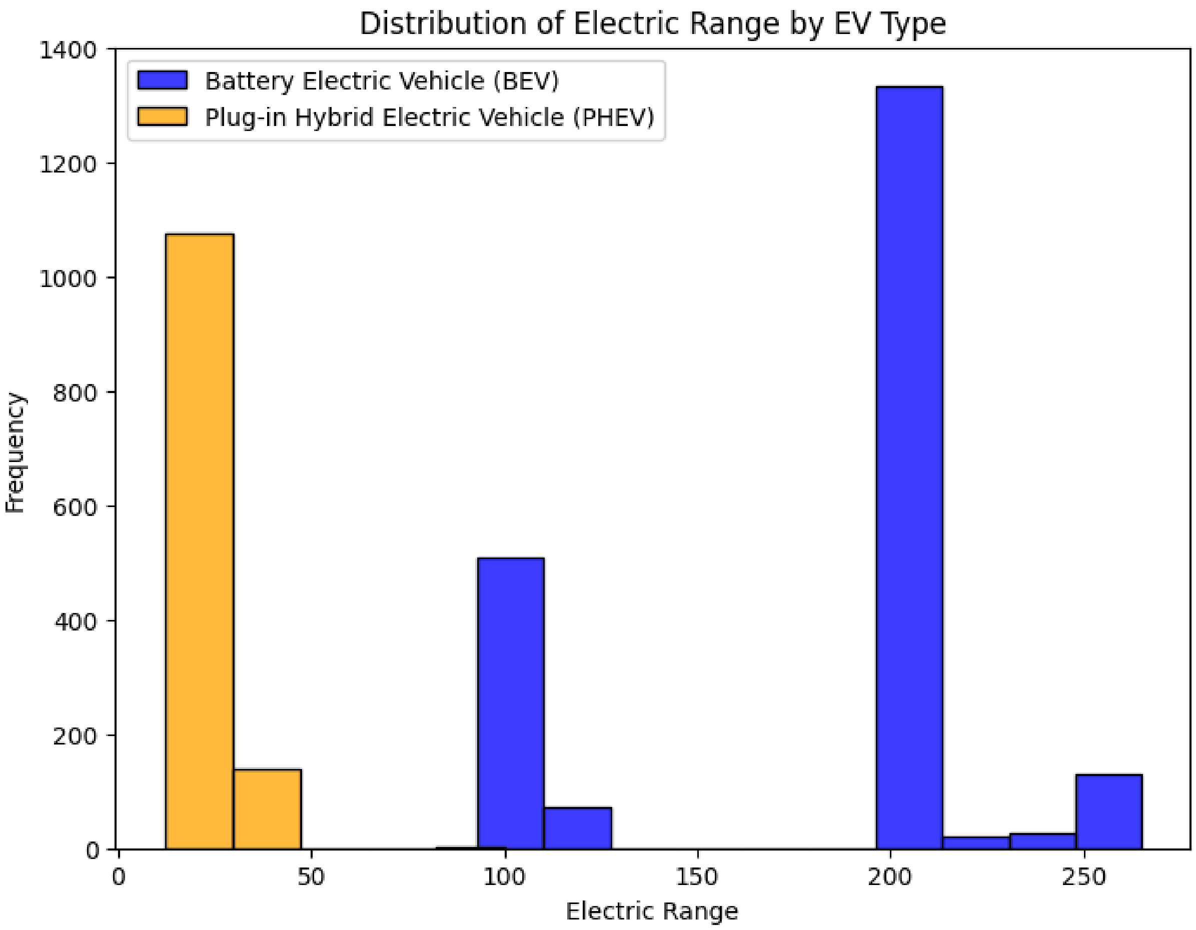
Data Exploration

In [129…

```
summary_stats = df.describe()
make_counts = df['Make'].value_counts()
ev_type_counts = df['EV_Type'].value_counts()
bev_df = df[df['EV_Type'] == 'Battery Electric Vehicle (BEV)']
phev_df = df[df['EV_Type'] == 'Plug-in Hybrid Electric Vehicle (PHEV)']
```

In [141…

```
plt.figure(figsize=(8, 6))
sns.histplot(bev_df['Electric_Range'], bins=10, color='blue', label='Battery Electr
sns.histplot(phev_df['Electric_Range'], bins=5, color='orange', label='Plug-in Hybr
plt.title('Distribution of Electric Range by EV Type')
plt.xlabel('Electric Range')
plt.ylabel('Frequency')
plt.legend()
plt.grid(False)
plt.show()
```

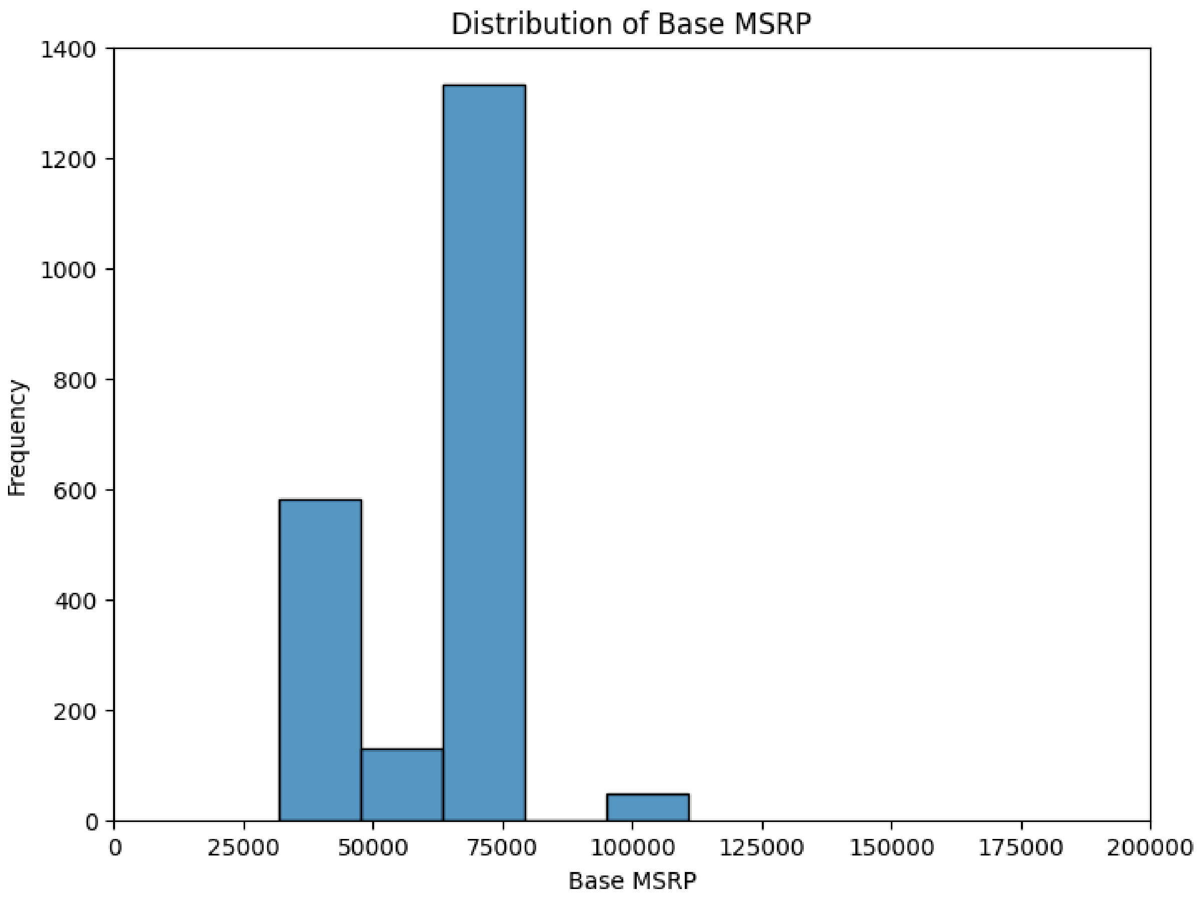


In [131…

```
df_cleaned = df[df['EV_Type'] != 'Plug-in Hybrid Electric Vehicle (PHEV)']
```

In [132…

```
plt.figure(figsize=(8, 6))
sns.histplot(df_cleaned['Base_MSRP'], bins=5)
plt.title('Distribution of Base MSRP')
plt.xlabel('Base MSRP')
plt.ylabel('Frequency')
plt.xlim(0, 200000)
plt.show()
```

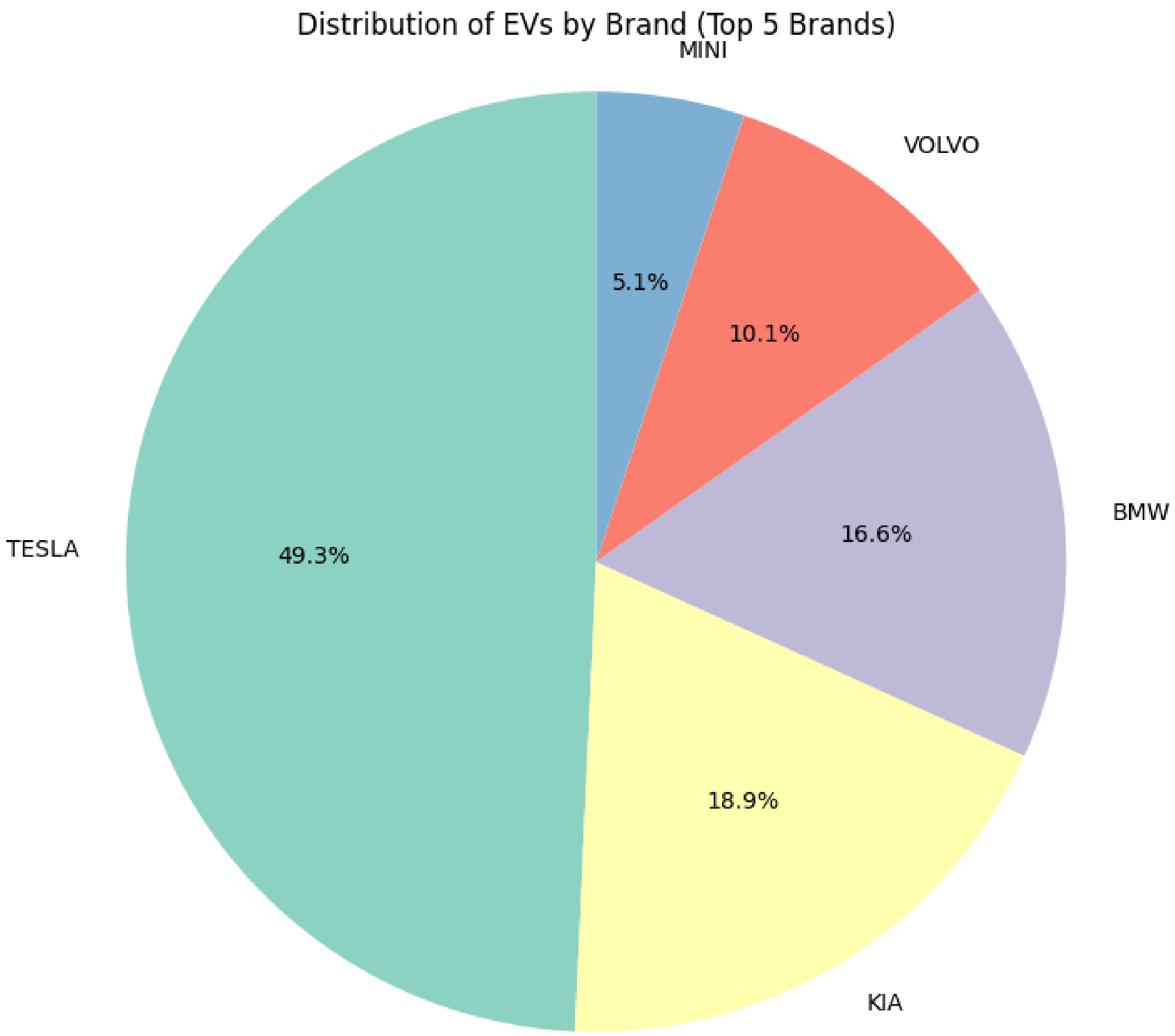


In [133…

```
df_cleaned = df[df['EV_Type'].isin(['Battery Electric Vehicle (BEV)', 'Plug-in Hybr
top_5_brands = df_cleaned['Make'].value_counts().nlargest(5)
```

In [134…

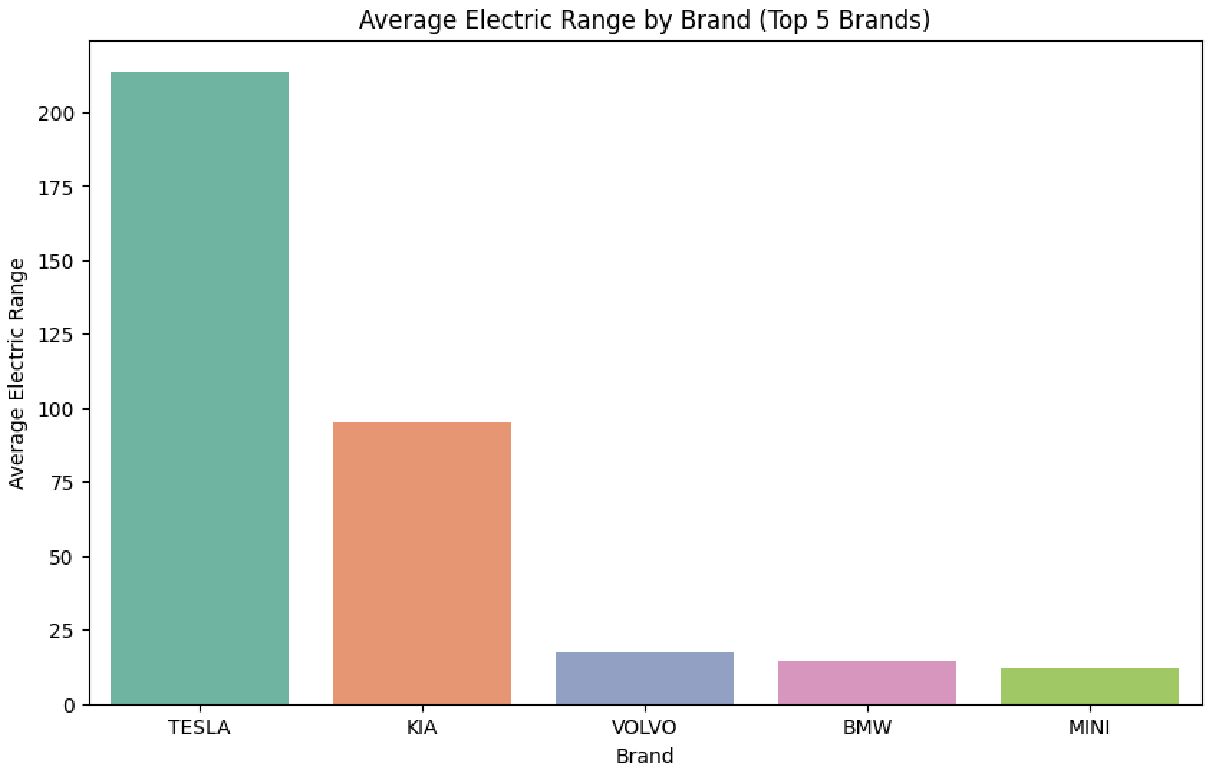
```
top_5_brands = df['Make'].value_counts().nlargest(5).index
df_top_brands = df[df['Make'].isin(top_5_brands)]
brand_counts = df_top_brands['Make'].value_counts()
plt.figure(figsize=(8, 8))
plt.pie(brand_counts, labels=brand_counts.index, autopct='%1.1f%%', startangle=90,
plt.title('Distribution of EVs by Brand (Top 5 Brands)')
plt.axis('equal')
plt.show()
```



In [135…

```
avg_range_by_brand = df_top_brands.groupby('Make')['Electric_Range'].mean().sort_va
plt.figure(figsize=(10, 6))
sns.barplot(x=avg_range_by_brand.index, y=avg_range_by_brand.values, palette='Set2'
plt.title('Average Electric Range by Brand (Top 5 Brands)')
plt.xlabel('Brand')
plt.ylabel('Average Electric Range')
plt.grid(False)
plt.show()
```

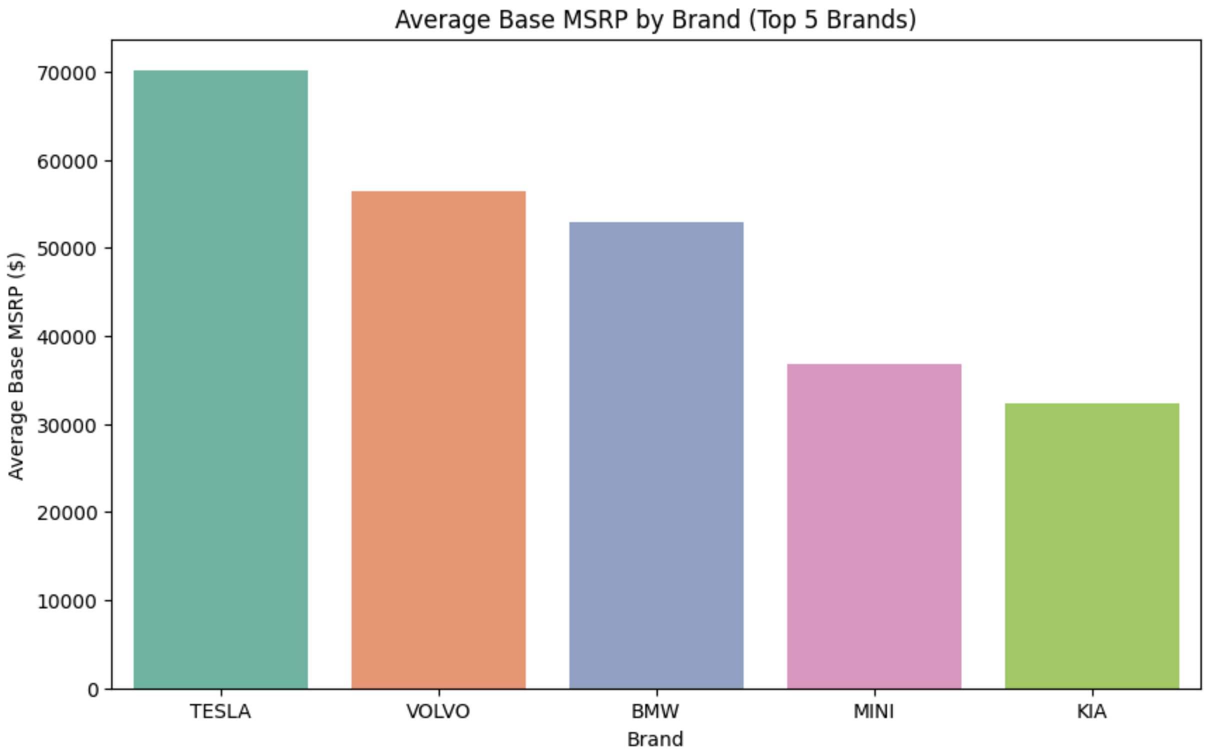
```
C:\Users\DSU\AppData\Local\Temp\ipykernel_23164\1850438583.py:3: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.1
4.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

  sns.barplot(x=avg_range_by_brand.index, y=avg_range_by_brand.values, palette='Set
2')
```



In [136…

```
# Calculate the average Base MSRP for each brand
avg_msrp_by_brand = df_top_brands.groupby('Make')['Base_MSRP'].mean().sort_values(a

# Plot the result
plt.figure(figsize=(10, 6))
sns.barplot(x=avg_msrp_by_brand.index, y=avg_msrp_by_brand.values, palette='Set2')
plt.title('Average Base MSRP by Brand (Top 5 Brands)')
plt.xlabel('Brand')
plt.ylabel('Average Base MSRP ($)')
plt.grid(False)
plt.show()
```

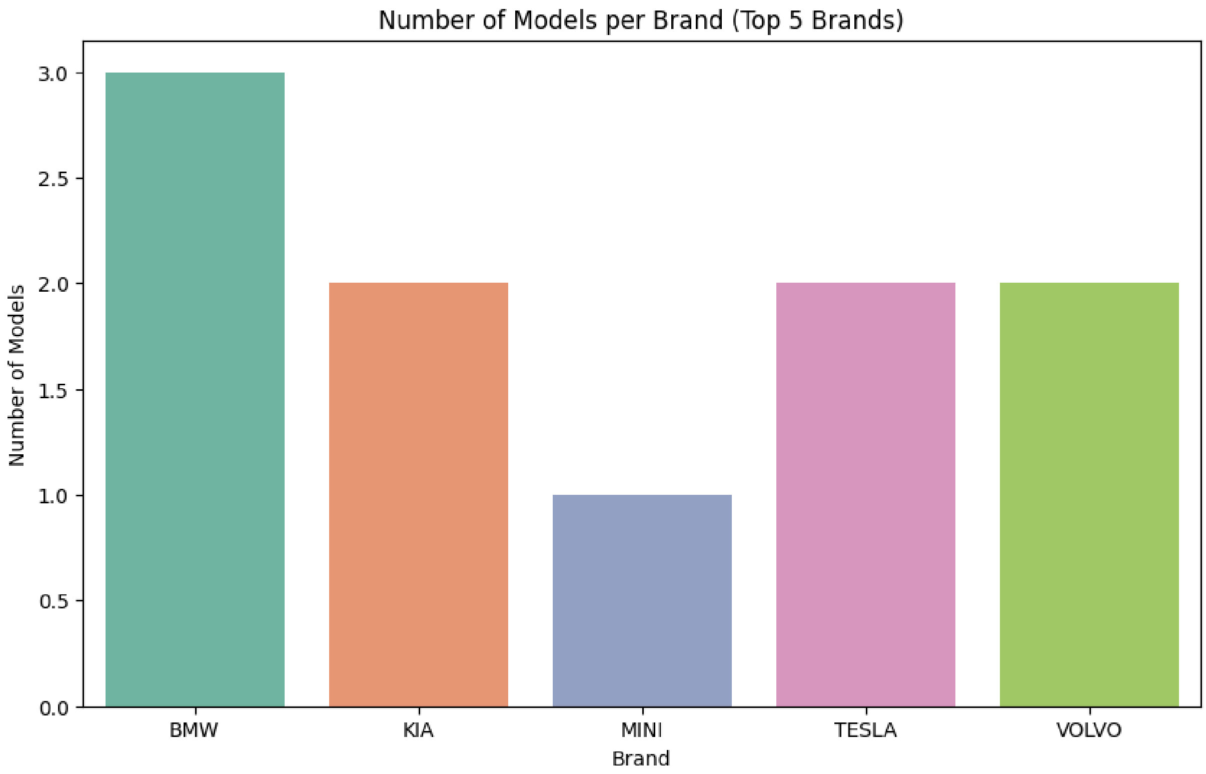
```
C:\Users\DSU\AppData\Local\Temp\ipykernel_23164\131424543.py:6: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.1
4.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

  sns.barplot(x=avg_msrp_by_brand.index, y=avg_msrp_by_brand.values, palette='Set2')
```



In [ ]:

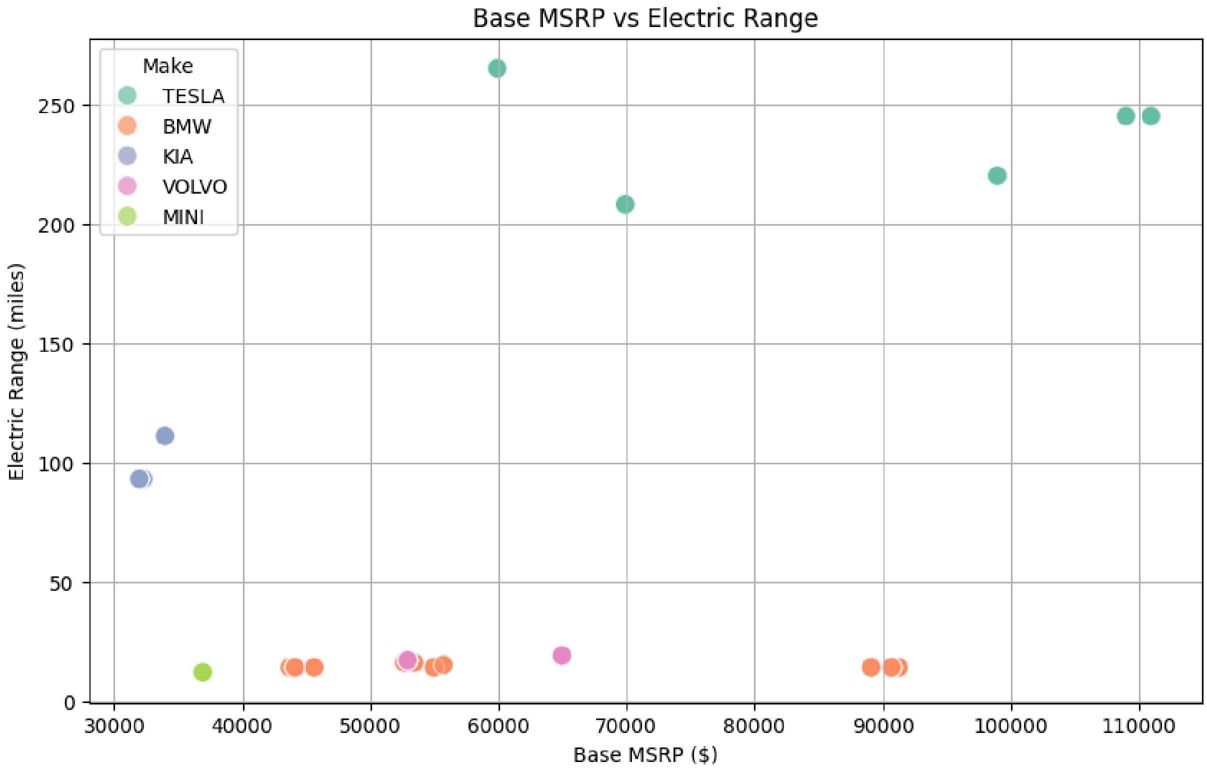
```
model_counts_by_brand = df_top_brands.groupby('Make')['Model'].nunique()

# Plot the result
plt.figure(figsize=(10, 6))
sns.barplot(x=model_counts_by_brand.index, y=model_counts_by_brand.values, palette=
plt.title('Number of Models per Brand')
plt.xlabel('Brand')
plt.ylabel('Number of Models')
plt.grid(False)
plt.show()
```

```
C:\Users\DSU\AppData\Local\Temp\ipykernel_23164\1264672327.py:5: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.1
4.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

  sns.barplot(x=model_counts_by_brand.index, y=model_counts_by_brand.values, palette
='Set2')
```



In [138…

```
plt.figure(figsize=(10, 6))
sns.scatterplot(data=df_top_brands, x='Base_MSRP', y='Electric_Range', hue='Make',
plt.title('Base MSRP vs Electric Range')
plt.xlabel('Base MSRP ($)')
plt.ylabel('Electric Range (miles)')
plt.grid(True)
plt.show()
```



Discussion of Findings

There seems to be a relatively noticeable correlation between the base price and the range on the vehicles, which should theoretically be the case. Looking back at our problem statement, the popularity stems from the range of a vehicle. The farther it can go, the more sales it gets. However, the face of Electric Vehicles 'Tesla' statistically outperforms in the range of its vehicles for their prices. This may have contributed to its strong value in the electrical vehicle market.