In [9]:
```
import os
%pip install pandas
import pandas as pd
```

```
Requirement already satisfied: pandas in /opt/anaconda3/envs/new_env/lib/python3.12/site-packages (2.2.3)
Requirement already satisfied: numpy>=1.26.0 in /opt/anaconda3/envs/new_env/lib/python3.12/site-packages (from pandas) (2.2.2)
Requirement already satisfied: python-dateutil>=2.8.2 in /opt/anaconda3/envs/new_env/lib/python3.12/site-packages (from pandas) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in /opt/anaconda3/envs/new_env/lib/python3.12/site-packages (from pandas) (2025.1)
Requirement already satisfied: tzdata>=2022.7 in /opt/anaconda3/envs/new_env/lib/python3.12/site-packages (from pandas) (2025.1)
Requirement already satisfied: six>=1.5 in /opt/anaconda3/envs/new_env/lib/python3.12/site-packages (from python-dateutil>=2.8.2->pandas) (1.16.0)
Note: you may need to restart the kernel to use updated packages.
```

In [10]:
```
df = pd.read_csv('merged_data.csv')

df.head()
```

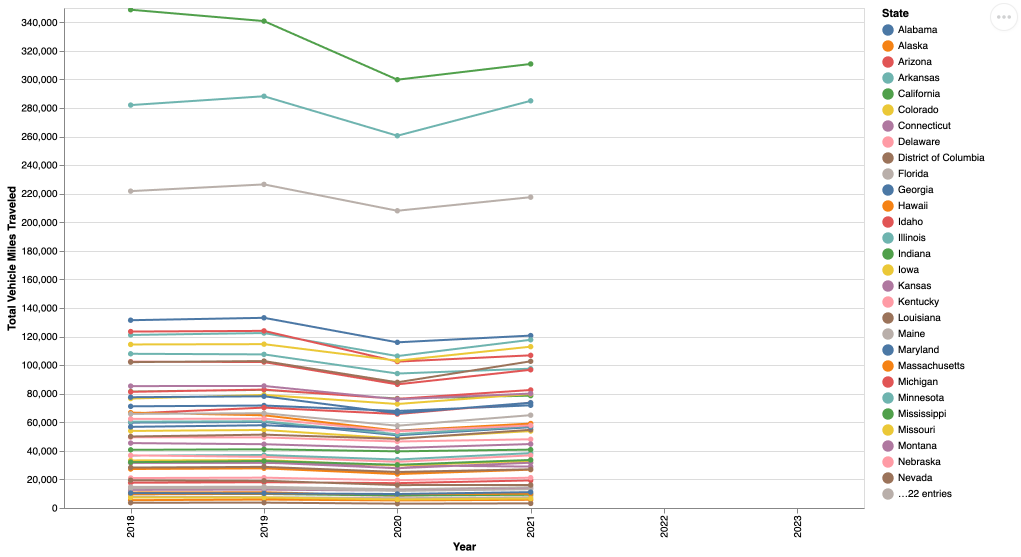
Out[10]:

	STATE	Total_Population	Avg_AGI	Electric (EV)	Plug-In Hybrid Electric (PHEV)	Hybrid Electric (HEV)	Biodiesel	Ethanol/Flex (E85)	Compressed Natural Gas (CNG)	Propane	Hydrogen	Methanol	Gasoline	Diesel	Unknown Fuel	Year	Rural VMT	Urban VMT	Total VMT
0	Alaska	7788980.0	4.5	500	300	5400	0	62000	3700	0	0	0	484900	41000	16100	2018	2276.520690035	3210.799304395	5487.3199944299995
1	Alaska	7788980.0	4.5	700	400	5800	0	64000	3200	0	0	0	473700	40100	14600	2019	2589.85610256	3291.447025075	5881.303127635
2	Alaska	7788980.0	4.5	900	400	6200	0	64000	2700	0	0	0	460900	39600	11400	2020	2482.95744	2822.93111	5305.88855
3	Alaska	7788980.0	4.5	1300	500	7300	7600	50100	100	0	0	0	464200	31700	6300	2021	2682.07237	3070.11528	5752.18765
4	Alaska	7788980.0	4.5	2700	900	10600	9600	46800	0	0	0	0	454300	29800	5100	2023	2,467	3,150	5,617

In [11]:
```
df.info()
```

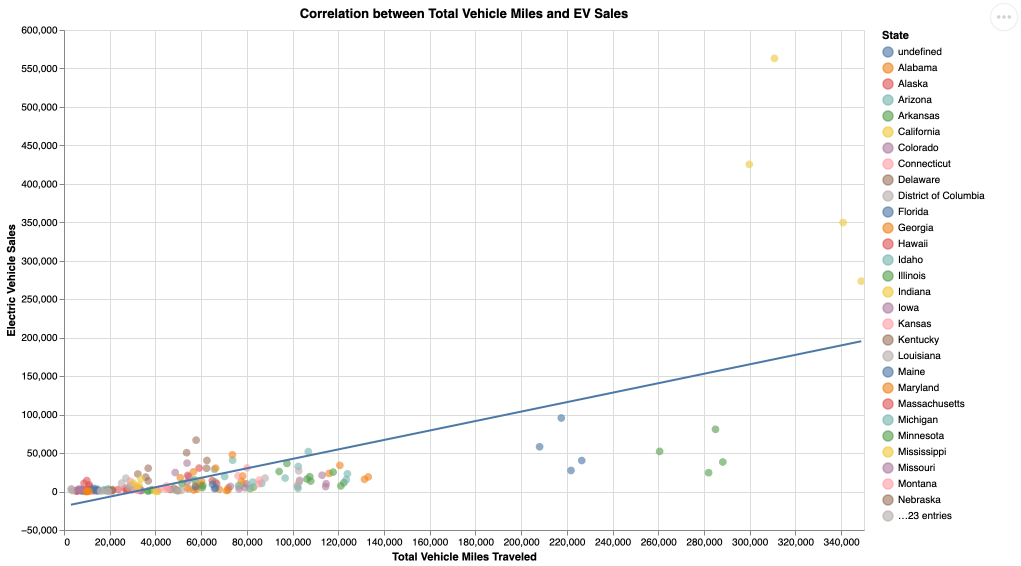
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 306 entries, 0 to 305
Data columns (total 19 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   STATE                           306 non-null    object
 1   Total_Population                306 non-null    float64
 2   Avg_AGI                         306 non-null    float64
 3   Electric (EV)                   306 non-null    int64
 4   Plug-In Hybrid Electric (PHEV)  306 non-null    int64
 5   Hybrid Electric (HEV)           306 non-null    int64
 6   Biodiesel                       306 non-null    int64
 7   Ethanol/Flex (E85)              306 non-null    int64
 8   Compressed Natural Gas (CNG)    306 non-null    int64
 9   Propane                         306 non-null    int64
 10  Hydrogen                        306 non-null    int64
 11  Methanol                        306 non-null    int64
 12  Gasoline                        306 non-null    int64
 13  Diesel                          306 non-null    int64
 14  Unknown Fuel                    306 non-null    int64
 15  Year                            306 non-null    int64
 16  Rural VMT                       306 non-null    object
 17  Urban VMT                       306 non-null    object
 18  Total VMT                       306 non-null    object
dtypes: float64(2), int64(13), object(4)
memory usage: 45.6+ KB
```

In [12]:
```
import pandas as pd
import altair as alt
import numpy as np

# Clean the VMT columns
for col in ["Rural VMT", "Urban VMT", "Total VMT"]:
    df[col] = df[col].astype(str).str.replace(r'\xa0', '', regex=True)  # Remove non-breaking spaces
    df[col] = df[col].str.replace('-', '0')  # Replace '-' with '0' (or use np.nan if missing)
    df[col] = pd.to_numeric(df[col], errors='coerce')  # Convert to float, setting errors to NaN

chart = alt.Chart(df).mark_line(point=True).encode(
    x=alt.X("Year:O", title="Year", sort="ascending"),
    y=alt.Y("Total VMT:Q", title="Total Vehicle Miles Traveled"),
    color=alt.Color("STATE:N", legend=alt.Legend(title="State")),
    tooltip=["STATE", "Year", "Total VMT"]
).properties(
    width=800,  # Adjust width (default is 600)
    height=500  # Adjust height (default is 400)
).interactive()

chart.show()
```



In [13]:

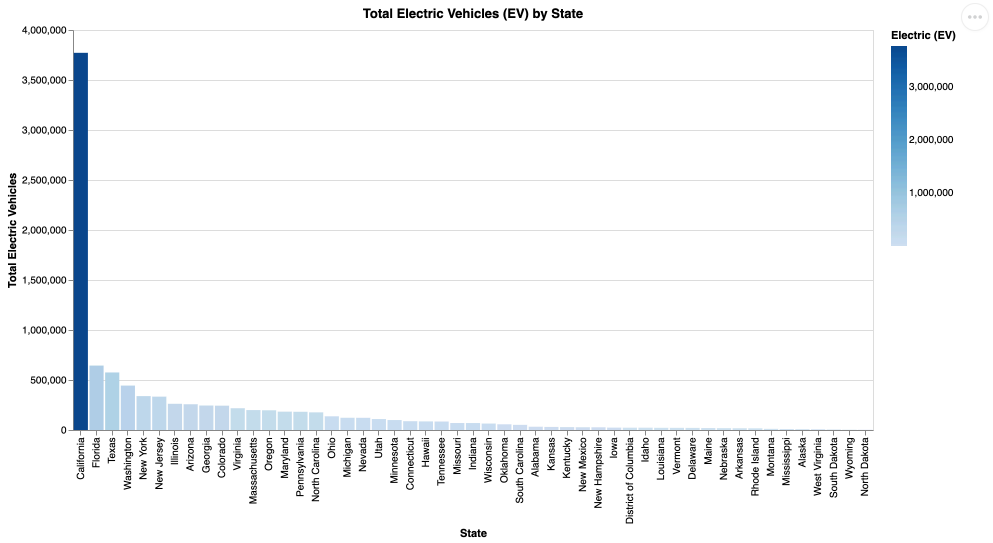
```
# Convert 'Total VMT' to numeric (remove commas if needed)
df["Total VMT"] = df["Total VMT"].astype(str).str.replace(',', '').astype(float)

# Create scatter plot with regression line
scatter_plot = alt.Chart(df).mark_circle(size=60, opacity=0.6).encode(
    x=alt.X("Total VMT:Q", title="Total Vehicle Miles Traveled"),
    y=alt.Y("Electric (EV):Q", title="Electric Vehicle Sales"),
    color=alt.Color("STATE:N", legend=alt.Legend(title="State")),
    tooltip=["STATE", "Year", "Total VMT", "Electric (EV)"]
)

# Regression line (trendline)
regression_line = scatter_plot.transform_regression(
    "Total VMT", "Electric (EV)", method="linear"
).mark_line(color="red")

# Combine scatter plot and regression line
chart = (scatter_plot + regression_line).properties(
    width=800,
    height=500,
    title="Correlation between Total Vehicle Miles and EV Sales"
).interactive()

chart.show()
```



In [14]:

```
import altair as alt
import pandas as pd

# Ensure numeric columns are correctly formatted
df['Electric (EV)'] = pd.to_numeric(df['Electric (EV)'], errors='coerce')

# Aggregate the total number of EVs per state
ev_per_state = df.groupby("STATE", as_index=False)["Electric (EV)"].sum()

# Create an interactive bar chart
chart = alt.Chart(ev_per_state).mark_bar().encode(
    x=alt.X("STATE:N", sort="-y", title="State"),
    y=alt.Y("Electric (EV):Q", title="Total Electric Vehicles"),
    tooltip=["STATE", "Electric (EV)"],
    color=alt.Color("Electric (EV):Q", scale=alt.Scale(scheme="blues"))
).properties(
    width=800,
    height=400,
    title="Total Electric Vehicles (EV) by State"
).interactive()

chart
```

Out[14]:


Total Electric Vehicles (EV) by State
Total Electric Vehicles
4,000,000
3,500,000
3,000,000
2,500,000
2,000,000
1,500,000
1,000,000
500,000
0
California
Florida
Texas
Washington
New York
New Jersey
Illinois
Arizona
Georgia
Colorado
Virginia
Massachusetts
Oregon
Maryland
Pennsylvania
North Carolina
Ohio
Michigan
Nevada
Utah
Minnesota
Connecticut
Hawaii
Tennessee
Missouri
Indiana
Wisconsin
Oklahoma
South Carolina
Alabama
Kansas
Kentucky
New Mexico
New Hampshire
Iowa
District of Columbia
Idaho
Louisiana
Vermont
Delaware
Maine
Nebraska
Arkansas
Rhode Island
Montana
Mississippi
Alaska
West Virginia
South Dakota
Wyoming
North Dakota
State
Electric (EV)
3,000,000
2,000,000
1,000,000

In [ ]: