Collaborators: Anurag Pathak (anp185)

Define Project/Novelty and Importance:

The dataset that I would like to use is from the US.gov website. The website has a plethora of datasets from all different states, all for public use. That being said, I chose to work with one dataset in particular that interests me for a number of reasons. The dataset that I am looking at holds information regarding all electric vehicles that are currently registered through Washington State Department of Licensing. You can view the website regarding the datasets here (in case hyper link doesn't work, I will include the raw link at the bottom). In recent years, we have seen the popularity of electric/hybrid vehicles sky-rocket. This sudden surge of popularity is likely attributed to the increasing gas costs following the Ukraine and Russia war along with EV companies optimizing their processes of creating their vehicles, which ultimately decreases the prices of their cars. Additionally, even though this isn't a part of my project, this popularity trend is not something that's only happening in the US, but is also happening globally. In fact, in 2023, 37% of the overall automotive sales in China were plug in electric vehicles. Needless to say, in our near future, EV's will become increasingly important. With that in mind, I wanted to do a project that analyzes EV's, more specifically the differences between fully electric cars and plug-in hybrids. This then leads me to finding this dataset off the US.gov datasets that are made publicly available. I think its an important topic for most car buyers since there are pros and cons for choosing either option, and I hope by doing this project, I can help car buyers understand the key differences between each vehicle type. This project seeks to analyze the differences in ownership, costs, and popularity trends between Battery Electric Vehicles (BEVs) and Plug-in Hybrid Electric Vehicles (PHEVs) in Washington. By examining these distinctions, the project will provide insight into factors that may influence consumer decisions.

Strategically, this project aligns with sustainability goals, economic implications, and consumer insights. Given the push towards clean energy and reduced carbon footprints, understanding EV trends can provide insights into future demands, economic incentives, and sustainability progress. This project will apply data management practices discussed in class, such as data cleaning, handling public datasets, and integrating external data sources for contextual insights. Since this is a dataset from the state, there is likely going to be some holes in the data that will need to be cleaned. Additionally, I will be using SQL to create a database using the data from this dataset. By using real-world data, the project will demonstrate the practical applications of structured data management. I am excited about this project because analyzing real-world data that directly impacts consumers and contributes to sustainability aligns with my interests in Economics. As someone who is triple majoring in CS, Economics, and Data Science, I always look for ways for me to integrate the knowledge and interests I have in all of the fields. With that, I firmly believe working on this project will be the perfect intersection of those goals. The interesting part of this project lies in the possible differences I could find between fully electric and hybrid models, specifically within Washington State, potentially uncovering regional trends that could guide other states or policymakers along with potential car buyers. As mentioned before, this project is essential as it tackles current trends and issues in sustainable transportation, providing potential buyers with insights into an evolving market also with direct

environmental and economic implications. As far as public datasets like this, they often have issues like missing entries, inconsistent formats, or incomplete data, which are challenges covered in data management practices. This project will apply techniques to address such issues, ensuring that the dataset is clean, reliable, and actionable for analysis.

Data:

As mentioned before, the data comes from the Washington State Department of Licensing. It is a culmination of Battery Electric Vehicles (BEVs) and Plug-in Hybrid Electric Vehicles (PHEVs) that are currently registered in Washington. The data is in a form of a csv file, with 210,000 rows and 17 columns. I plan on first cleaning the dataset using a plethora of different techniques that we covered in class. Then from there based on the the value's type, I will either just remove the entry or if there is a way where i can impute a value instead of just outright removing it, I will do that instead. Then, from there, I plan to do more EDA to better understand the variables and how they react with each other. To accomplish this, I will start by creating correlation heat maps to determine if there are any variables with higher correlations with each other. The reasoning behind this is to understand the dataset a bit more before maybe trying to model costs for each different EV type. Additionally, I hope to use SQL databases to input the data from the csv that we are given, and to help create queries that can be used to better understand the importance of each of the variables more. I also plan to use several modeling techniques that I think would be beneficial, primarily linear models, but as I continue to understand the dataset more, I am open to learn different kind of modeling techniques that would be more applicable to this data. As for what I would be modeling, I would likely be trying to predict the price of the cars based on the variables listed, and as for how I would do it, I would use the standard method where we train the model using 85% of our data, and then testing on the remaining 15%, and seeing how much the difference is overall. Its important to try and understand what's going on these models since an accurate model is pivotal for helping consumers understand what their average costs might be. We can test our "success" by using a R-squared, and we can have a threshold before hand of what we consider as a successful model. Here is a basic representation of my steps:

1. Load and clean the data, addressing missing entries and verifying key columns.
2. Store cleaned data in a SQL database for ease of access and querying.
3. Conduct exploratory data analysis (EDA) to identify relationships between variables.
4. Train and test predictive models, using 85/15 split validation.
5. Use SQL queries to extract insights related to vehicle types, registration trends, and potential costs.

https://data.wa.gov/Transportation/Electric-Vehicle-Population-Data/f6w7-q2d2/about_data

Cleaning:

So as far as cleaning my data, there several factors that I accounted for, including null values, duplicates, outliers, inconsistent values, one-hot encoding,

imputing values, and correct data types. The first thing I did in my cleaning process was imputing/dealing with null values. Now, there were many considerations I did when doing this process, so I will go step-by-step to explain my reasoning and logic. So the first thing I did was to deal with the entries that had a null value for the County variable. When taking a look at the amount of null values for my data, I noticed that County had 5 null values. In comparison to the tens of thousands of data points in this data set, rather than trying to impute a value for this variable, I decided it would be best to just remove these entries, as those specific 5 entries ended up also having null or 0's for most of their other fields as well. Next up, I decided for more consistency and to make my life easier for later parts of the cleaning process, for all my numerical variables, I imputed the value 0 in place of the null values. Now, for this project, two of the most important variables, Electric Range and Base MSRP, had a lot of entries that had missing values for this field. According to the dataset's website, for a lot of the cars, they simply didn't have the information regarding the MSRP price and the mileage range of the vehicle. This was an issue for roughly 90% of the entries in this dataset, which gave me two options: either I remove those 90% entries and be left with 3k observations or I try imputing values for the rest of those entries. The problem with the first option is that I may fall victim to biases that stem from not including the entire sample/covering all demographics. However, I believe that the second option leads to even more problems. For example, if we tried imputing values for the rest of those entries, it would mean that the analysis that we are doing is based almost entirely on "predicted" values (assuming that the imputed values would be done through some prediction model to best estimate what the value would be) and not actual values. Because of that, I decided to go with the first option, especially since 3 thousand observations is still more than enough to do a proper analysis on the dataset, which meant removing all the entries with 0 as a value (which is the reason why earlier I imputed the NA's with a 0 to make this part easier). From there, I started working on duplicate values. Now luckily for me, each car has their own VIN number, which acts as a unique identifier. Even though the dataset only includes the first 10 digits of each VIN number, the fact remains that no two cars can have the same VIN number, meaning that there are no duplicates in this dataset. Next, for possible usage in the modeling portion of this project, I used one-hot encoding the change the following categorical variables to numerical versions: 'Model Year', 'Make', 'Model' and 'Electric Vehicle Type.' Lastly, I used an interquartile function to remove any outliers in the two categorical variables, making my dataset clean and ready for the next part of the project.

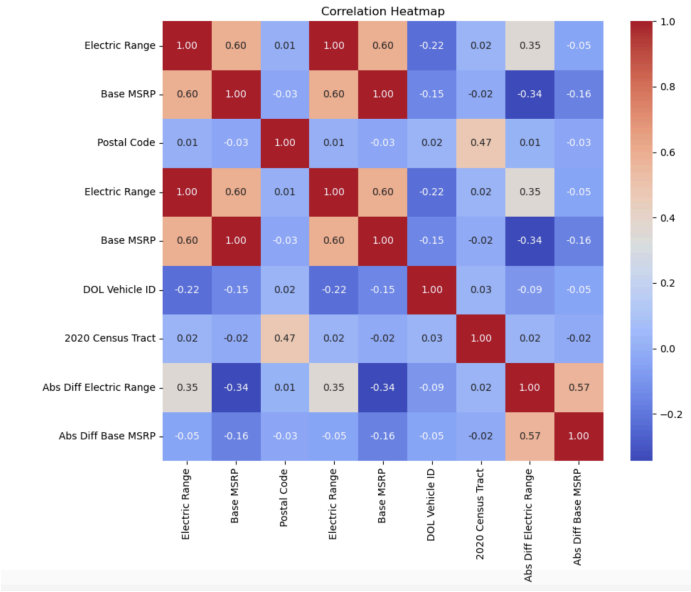
Below I have attached a picture of the code that I have used for my cleaning processes:

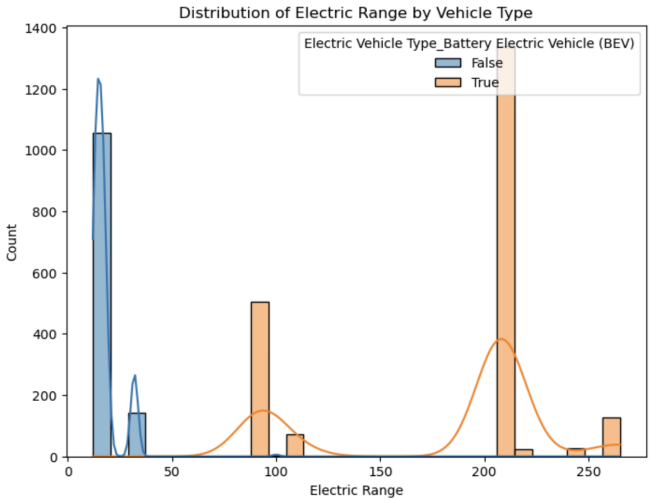
```
##Importing the dataset and cleaning it
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
import pandas as pd
df = pd.read_csv("ElectricVehicle.csv")
null_counts = df.isnull().sum()
#print(null_counts)

#things to consider when cleaning data:
#Understanding data types
#Null values
#Duplicates
#Outliers
#Inconsistent Values
#One-hot Encoding for categorical variables
#Correct Data types
#imputing values

#Cleaning Null Values (Null Values and Inconsistent Values) and imputing values
#When looking at the variables who have null value for County, City, and Postal Code
#we see 5 entries, who seem to missing a lot of values. For that reason, we are removing those
#entries entirely due to their being a lack of information to attempt to try and impute a value
df = df[df["County"].isnull() == False]
#replacing nan with 0's
#for the Base MSRP and Electric Range variables, I plan on replacing the NAN variables with 0's
#I'm doing this so that its easier to replace the 0 values under the Electric Range variable, since
# a vehicle with a 0 electric range isn't actually a EV. There could be a lot of reasons why its 0,
#but the main one is likely lack of proper data collection. For that reason, we are removing any 0 value.
count_zero_range = (df["Electric Range"] == 0).sum()
#print(count_zero_range)
df['Base MSRP'] = df['Base MSRP'].fillna(0)
df['Electric Range'] = df['Electric Range'].fillna(0)
df['Legislative District'] = df['Legislative District'].fillna("Unknown")
df['Vehicle Location'] = df['Vehicle Location'].fillna("")
df = df[df['Electric Range']!= 0]
df = df[df['Base MSRP']!= 0]
#print(len(df))
null_counts = df.isnull().sum()
#print(null_counts)
#Since the only nan values left are for legistlative District and vehicle location, which is a variable i won't be interacting with,
#this part of the cleaning process is complete.

#Duplicates
#Luckily for us, this dataset only uses one entry-per car, meaning that we wouldn't need to check for duplicates. That being said, there is no way for us
#to 100% make sure this is the case, so we just need to trust what they are saying since the vin variable only includes the first 10 characters, which
#isn't enough to identify a duplicate or not.

#One-hot encoding
columns_to_encode = ['Model Year', 'Make', 'Model', 'Electric Vehicle Type']
df = pd.get_dummies(df, columns=columns_to_encode)
#print(df)
'
#unique_makes = df['Make'].unique()
#print(unique_makes)
#unique_models = df['Model'].unique()
#print(unique_models)
#unique_model_year = df['Model Year'].unique()
#print(unique_model_year)
#unique_electric_range = df['Electric Range'].unique()
#print(unique_electric_range)
#unique_EVType = df['Electric Vehicle Type'].unique()
#print(unique_EVType)
# columns_to_check = ['Make', 'Model Year', 'Model', 'Electric Range']
# for col in columns_to_check:
#     null_count = df[col].isnull().sum()
#     print(f"{col} has {null_count} empty values.")

#outliers
#I am using the interquartile method to get rid of outliers.
def remove_outliers_iqr(df, column):
    Q1 = df[column].quantile(0.25)
    Q3 = df[column].quantile(0.75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR
    return df[(df[column] >= lower_bound) & (df[column] <= upper_bound)]
df = remove_outliers_iqr(df, 'Electric Range')
df = remove_outliers_iqr(df, 'Base MSRP')


null_counts = df.isnull().sum()
print(null_counts)
print(len(df))
```

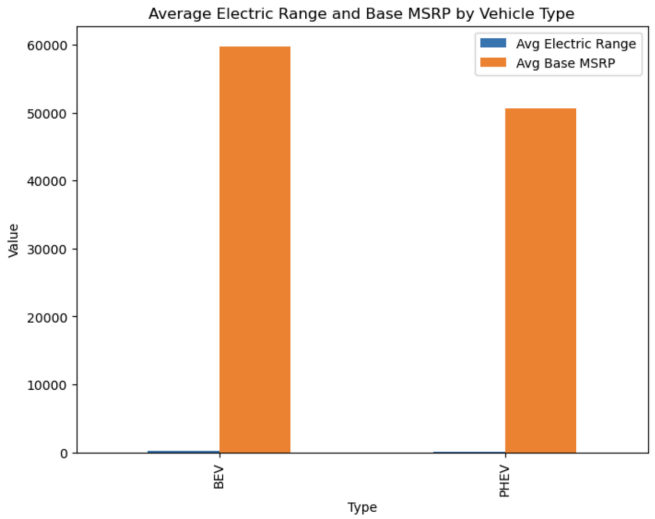
EDA (Exploratory Data Analysis):

So, to begin with my exploratory data analysis, I first started by generating a correlation heat map to better understand what variables have a bigger impact on the price of a car. Below is a picture of that correlation heat map. As you can see from the visual, it seems like electric range and base MSRP have a relatively high correlation with each other, making it a variable of interest in our machine learning portion of this project. That being said, the correlation between the two could be an overestimate that

is derived from OVB, so its important to keep that in mind. In addition to that, I have attached a few graphs showing the relationship between vehicle type and the electric range that the vehicle has. The picture is also shown below. As we can see, the fully electric cars almost always had a higher electric range compared to the plug-in hybrid vehicles. However, that increased range does come with an increase in cost on the vehicle, which is shown by the third image below. More specifically, on average, BEV had 181.08 miles on electric range and $59785.51 on base MSRP, while PHEV had an electric range of 17.39 and a base MSRP of 50670.95. So the trade off seems very obvious here, you are paying more for a fully electric vehicle to have more range whereas a hybrid is cheaper, has electric range, and also has gas costs associated with it. This is something we will look back at in the Machine learning portion, where we will talk about which one would be better in the long term. Now, I will go to the SQL portion of this project, where I will look at more in-depth insights, such as what brands are the most expensive on average.








```
correlation_columns = ['Electric Range', 'Base MSRP'] + [col for col in df.columns if df[col].dtype in ['int64', 'float64']]
correlation_data = df[correlation_columns].corr()
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_data, annot=True, cmap='coolwarm', fmt=".2f")
plt.title("Correlation Heatmap")
plt.show()

# Calculate averages for BEV and PHEV
bev_avg = df[df["Electric Vehicle Type_Battery Electric Vehicle (BEV)"] == 1][["Electric Range", "Base MSRP"]].mean()
phev_avg = df[df["Electric Vehicle Type_Plug-in Hybrid Electric Vehicle (PHEV)"] == 1][["Electric Range", "Base MSRP"]].mean()
print(f"BEV Averages:\n{bev_avg}")
print(f"PHEV Averages:\n{phev_avg}")

#Adding a new column with actual value - average
df['Abs Diff Electric Range'] = df.apply(
    lambda row: abs(row['Electric Range'] - (bev_avg['Electric Range'] if row["Electric Vehicle Type_Battery Electric Vehicle (BEV)"] == 1 else phev_avg['Electric Range'])),
    axis=1
)
df['Abs Diff Base MSRP'] = df.apply(
    lambda row: abs(row['Base MSRP'] - (bev_avg['Base MSRP'] if row["Electric Vehicle Type_Battery Electric Vehicle (BEV)"] == 1 else phev_avg['Base MSRP'])),
    axis=1
)
print(df[['Abs Diff Electric Range', 'Abs Diff Base MSRP']].head())

plt.figure(figsize=(8, 6))
sns.scatterplot(data=df, x='Electric Range', y='Base MSRP', hue='Electric Vehicle Type_Battery Electric Vehicle (BEV)')
plt.title("Electric Range vs. Base MSRP")
plt.xlabel("Electric Range")
plt.ylabel("Base MSRP")
plt.legend(title="Vehicle Type", labels=["PHEV", "BEV"])
plt.show()

plt.figure(figsize=(8, 6))
sns.histplot(data=df, x='Electric Range', hue='Electric Vehicle Type_Battery Electric Vehicle (BEV)', kde=True, bins=30)
plt.title("Distribution of Electric Range by Vehicle Type")
plt.xlabel("Electric Range")
plt.show()

averages = pd.DataFrame({'Type': ['BEV', 'PHEV'],
                         'Avg Electric Range': [bev_avg['Electric Range'], phev_avg['Electric Range']],
                         'Avg Base MSRP': [bev_avg['Base MSRP'], phev_avg['Base MSRP']]})

averages.set_index('Type').plot(kind='bar', figsize=(8, 6), title="Average Electric Range and Base MSRP by Vehicle Type")
plt.ylabel("Value")
plt.show()
```

Would you like to get notified abo
Jupyter news?

Database/SQL:

Now, for the database component, **PLEASE NOTE**, I will not post a screenshot of my code since I am starting to run out of space, instead I will only show an image of the output. This part is relatively straightforward, so I will keep it brief. What I did was add all of the data from our data frame into a SQL database, and then create two tables that separated into the two different vehicle types. By doing this, it allows for me to do calculations based on the vehicle type. From there, I did any analysis where we looked at each year for each vehicle type, and looked at the average MSRP Cost and Electric Range. Below is a picture of that output. Please note that there are some NAN values due to no data present for those years. One thing you might find shocking is that older

models tend to cost more than the newer ones. That's likely because at the time, electric vehicles were a newer invention, and since the processes for making them were likely not optimized, they overall cost more than they do now. So for people looking to purchase a car, a newer one seems to be the better bet. I also included a graphic that held the information of the averages of base MSRP and Electric Range for different car makes, with Kia's being the cheapest for BEV cars and Wheego for PHEV.

```
Separate tables 'bev' and 'phev' created.
   Model_Year  Avg_Base_MSRP  Avg_Electric_Range Vehicle_Type
0        2008   98950.000000          220.000000          BEV
1        2010  110950.000000          245.000000          BEV
2        2011  109000.000000          245.000000          BEV
3        2012   59900.000000          265.000000          BEV
4        2013   69900.000000          208.000000          BEV
5        2014   69900.000000          208.000000          BEV
6        2015            NaN                 NaN          BEV
7        2016   31950.000000           93.000000          BEV
8        2017   32250.000000           93.000000          BEV
9        2018   33950.000000          111.000000          BEV
10       2019            NaN                 NaN          BEV
11       2020            NaN                 NaN          BEV
0        2008            NaN                 NaN         PHEV
1        2010   32995.000000          100.000000         PHEV
2        2011            NaN                 NaN         PHEV
3        2012  102000.000000           33.000000         PHEV
4        2013            NaN                 NaN         PHEV
5        2014            NaN                 NaN         PHEV
6        2015            NaN                 NaN         PHEV
7        2016   43700.000000           14.000000         PHEV
8        2017   48255.825688           14.467890         PHEV
9        2018   54120.220848           15.839223         PHEV
10       2019   44561.356994           19.212944         PHEV
11       2020   81100.000000           14.000000         PHEV

                   Make  Avg_Base_MSRP  Avg_Electric_Range Vehicle_Type
0                   BMW            NaN                 NaN          BEV
1              CADILLAC            NaN                 NaN          BEV
2              CHRYSLER            NaN                 NaN          BEV
3                FISKER            NaN                 NaN          BEV
4                   KIA   32270.486111           95.218750          BEV
5                  MINI            NaN                 NaN          BEV
6               PORSCHE            NaN                 NaN          BEV
7                SUBARU            NaN                 NaN          BEV
8                 TESLA   70225.988142          213.662714          BEV
9                 VOLVO            NaN                 NaN          BEV
10  WHEEGO ELECTRIC CARS            NaN                 NaN          BEV
0                   BMW   52815.060241           14.552209         PHEV
1              CADILLAC   75095.000000           31.000000         PHEV
2              CHRYSLER   39995.000000           32.000000         PHEV
3                FISKER  102000.000000           33.000000         PHEV
4                   KIA            NaN                 NaN         PHEV
5                  MINI   36865.189873           12.000000         PHEV
6               PORSCHE   81100.000000           14.000000         PHEV
7                SUBARU   34995.000000           17.000000         PHEV
8                 TESLA            NaN                 NaN         PHEV
9                 VOLVO   56320.645161           17.567742         PHEV
10  WHEEGO ELECTRIC CARS   32995.000000          100.000000         PHEV
```

Machine Learning and Outcomes:

So in this part, I created two different models to test and see what factors are the largest contributors/how effectively can we predict the price of these cars using our models. The first model is using Make, Year, and Electric Range to predict the price of the car, and the second model is all of those variables plus the type of vehicle it is (electric/hybrid). The reason I am doing it like this is to try and get a better picture as to how much the type of the vehicle's impact is on the price of vehicle. **PLEASE NOTE**, just as last time, I won't upload pictures of my code since it takes up a lot of space, but I

will put pictures of what my results are shown below. You can see the code as a part of my Jupyter Notebook. Model 1 is the model without the electric/hybrid variable, and model 2 does include that variable. There are a few noteworthy things here. In model two, we noticed that the mean squared error has gone down compared to the first model. Along with this, the R2 score went up, implying that adding the variable that dictates what kind of vehicle it is has an impact on the price. This further supports the claim that there is a price difference from buying a fully electric vs hybrid vehicle. Additionally, it seems as if the model's R2 value is close to 1, making it a relatively good model. Now, this brings into question, what does that all mean for the normal person trying to buy a car? Well, first off, from our earlier analysis in the EDA section, it seems that fully electric cars cost more upfront, but have a lot higher electric range compared to hybrid cars. Now on average, EVs have lower maintenance costs than hybrids because they don't need oil changes or spark plugs. Unfortunately, this dataset doesn't have the information about maintenance costs, but this information coupled with what we found can be used to figure out what car is for you. If you plan on buying a car for long-term use, the fully electric car is the better bet since over time, it will be less expensive than a hybrid vehicle. On the other hand, if you are planning on selling your car a couple years after buying it, the hybrid car is the better bet, since you won't need to worry about future maintenance costs.

```
Model 1 Results
Mean Squared Error: 0.10225
R2 Score: 0.93934

Model 2 Results
Mean Squared Error: 0.02174
R2 Score: 0.95326
```

Limitations:

As mentioned before, one of the big limitations of this dataset is that since we want to stick with only the true data and not synthetic data or heavily imputed data, our analysis may be omitting some groups from different social statuses. Additionally, as shown in the database portion of this project, because we had removed a lot of the data points from that dataset, we ended up having some categories with 0 entries in it. Lastly, this analysis was done with data from Washington State, which limits the scope of the analysis to that state only. Keep in mind, I went into more detail about the limitations in other parts of this document, just wanted to highlight on the important parts. One advantage of my approach is that I didn't use any synthetic data, meaning all the results I derived were from actual observations.