Assignment 1 — Electric Vehicle Population Data

Jaipreet Kaur

2025-09-07

Overview

This report completes the required tasks: 1. Import a web-sourced dataset with both quantitative and categorical variables
2. Print descriptive statistics (numeric & categorical)
3. Transform at least one variable
4. Create one univariate plot and one scatterplot
5. Include a link to the dataset source

> **Dataset**: *Electric Vehicle Population Data (Washington State)*
> **Source**: _https://catalog.data.gov/dataset/electric-vehicle-population-data._ (https://catalog.data.gov/dataset/electric-vehicle-population-data._)

```
# Packages
library(readr)
library(dplyr)
library(ggplot2)
library(scales)
```

1) Import the dataset

Below we read the CSV file named `Electric_Vehicle_Population_Data.csv` .
If you keep the file in the same folder as this Rmd, the following line will work as-is.

```
# If your file is in a different folder, update the path below.
csv_path <- "Electric_Vehicle_Population_Data.csv"

# Read the data
df <- read_csv(csv_path, show_col_types = FALSE)

# Quick look
dim(df)
```

```
## [1] 257635     17
```

```
colnames(df)
```

```
##  [1] "VIN (1-10)"
##  [2] "County"
##  [3] "City"
##  [4] "State"
##  [5] "Postal Code"
##  [6] "Model Year"
##  [7] "Make"
##  [8] "Model"
##  [9] "Electric Vehicle Type"
## [10] "Clean Alternative Fuel Vehicle (CAFV) Eligibility"
## [11] "Electric Range"
## [12] "Base MSRP"
## [13] "Legislative District"
## [14] "DOL Vehicle ID"
## [15] "Vehicle Location"
## [16] "Electric Utility"
## [17] "2020 Census Tract"
```

```
head(df, 5)
```

```
## # A tibble: 5 × 17
##   `VIN (1-10)` County    City       State `Postal Code` `Model Year` Make  Model
##   <chr>        <chr>     <chr>      <chr> <chr>                <dbl> <chr> <chr>
## 1 5YJ3E1EB5K   Yakima    Yakima     WA    98901                 2019 TESLA MODE…
## 2 1C4RJXU67R   Kitsap    Port Orch… WA    98367                 2024 JEEP  WRAN…
## 3 KNDCD3LD0N   Snohomish Lynnwood   WA    98036                 2022 KIA   NIRO
## 4 5UXKT0C37H   King      Auburn     WA    98001                 2017 BMW   X5
## 5 1N4AZ0CP1D   Skagit    Mount Ver… WA    98273                 2013 NISS… LEAF
## # ℹ 9 more variables: `Electric Vehicle Type` <chr>,
## #   `Clean Alternative Fuel Vehicle (CAFV) Eligibility` <chr>,
## #   `Electric Range` <dbl>, `Base MSRP` <dbl>, `Legislative District` <dbl>,
## #   `DOL Vehicle ID` <dbl>, `Vehicle Location` <chr>, `Electric Utility` <chr>,
## #   `2020 Census Tract` <chr>
```

Columns (abbrev.) - Quantitative: `Model Year`, `Electric Range`, `Base MSRP`
- Categorical: `Make`, `Model`, `Electric Vehicle Type`, `Clean Alternative Fuel Vehicle (CAFV) Eligibility`, `County`, `City`

2) Descriptive statistics

Numeric summaries

```
# Choose numeric variables present in the dataset
numeric_vars <- c("Model Year", "Electric Range", "Base MSRP")

# Defensive: keep only columns that exist & are numeric
num_cols <- intersect(numeric_vars, names(df))

df %>%
  select(all_of(num_cols)) %>%
  summary()
```

```
##    Model Year   Electric Range     Base MSRP
##  Min.   :2000   Min.   :  0.00   Min.   :     0.0
##  1st Qu.:2020   1st Qu.:  0.00   1st Qu.:     0.0
##  Median :2023   Median :  0.00   Median :     0.0
##  Mean   :2022   Mean   : 43.13   Mean   :   705.3
##  3rd Qu.:2024   3rd Qu.: 35.00   3rd Qu.:     0.0
##  Max.   :2026   Max.   :337.00   Max.   :845000.0
##                 NA's   :3        NA's   :3
```

> *Note:* In this dataset, `Base MSRP` can contain zeros that represent missing/unknown list prices.
> We can treat 0 as missing for MSRP-related summaries/plots if needed.

```
df <- df %>% mutate(Base_MSRP_clean = ifelse(`Base MSRP` <= 0 | is.na(`Base MSRP`), NA_real_, `Base MSRP`))

summary(df$Base_MSRP_clean)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   31950   39995   57800   57174   69900  845000  254457
```

Categorical summaries

```r
cat_vars <- c("Make", "Model", "Electric Vehicle Type", "Clean Alternative Fuel Vehicle (CAFV) Eligibility")

# Show top categories for large-cardinality columns
lapply(cat_vars, function(v) {
  if (v %in% names(df)) {
    tab <- table(df[[v]], useNA = "ifany")
    sort(tab, decreasing = TRUE)[1:10]  # top 10 levels
  } else {
    paste("Column not found:", v)
  }
})
```

```
## [[1]]
## 
##      TESLA CHEVROLET    NISSAN      FORD       KIA       BMW    TOYOTA   HYUNDAI 
##     107535     18602     16274     13750     12586     10656     10622      8638 
##     RIVIAN     VOLVO 
##       7816      6673 
## 
## [[2]]
## 
##        MODEL Y        MODEL 3           LEAF        MODEL S        BOLT EV 
##          53560          37807          13971           7911           7812 
##        MODEL X MUSTANG MACH-E            ID.4        IONIQ 5       WRANGLER 
##           6713           5597           5338           4833           4831 
## 
## [[3]]
## 
##          Battery Electric Vehicle (BEV) Plug-in Hybrid Electric Vehicle (PHEV) 
##                                  205095                                  52540 
##                                    <NA>                                   <NA> 
## 
##                                    <NA>                                   <NA> 
## 
##                                    <NA>                                   <NA> 
## 
##                                    <NA>                                   <NA> 
## 
## 
## [[4]]
## 
## Eligibility unknown as battery range has not been researched 
##                                                       157670 
##                      Clean Alternative Fuel Vehicle Eligible 
##                                                        76157 
##                        Not eligible due to low battery range 
##                                                        23808 
##                                                         <NA> 
## 
##                                                         <NA> 
## 
##                                                         <NA> 
## 
##                                                         <NA> 
## 
##                                                         <NA> 
## 
##                                                         <NA> 
## 
##                                                         <NA> 
## 
```

3) Transform at least one variable

Here we add two example transformations (either one satisfies the requirement): - `is_Tesla` : converts `Make` to a binary indicator (1 if Tesla, else 0).
- `log1p_range` : log-transform of `Electric Range` to reduce right skew.

```
df <- df %>%
  mutate(
    is_Tesla = ifelse(Make == "TESLA" | Make == "Tesla", 1L, 0L),
    log1p_range = ifelse(is.na(`Electric Range`), NA_real_, log1p(`Electric Range`))
  )

# Quick check
df %>% select(Make, `Electric Range`, is_Tesla, log1p_range) %>% head(10)
```

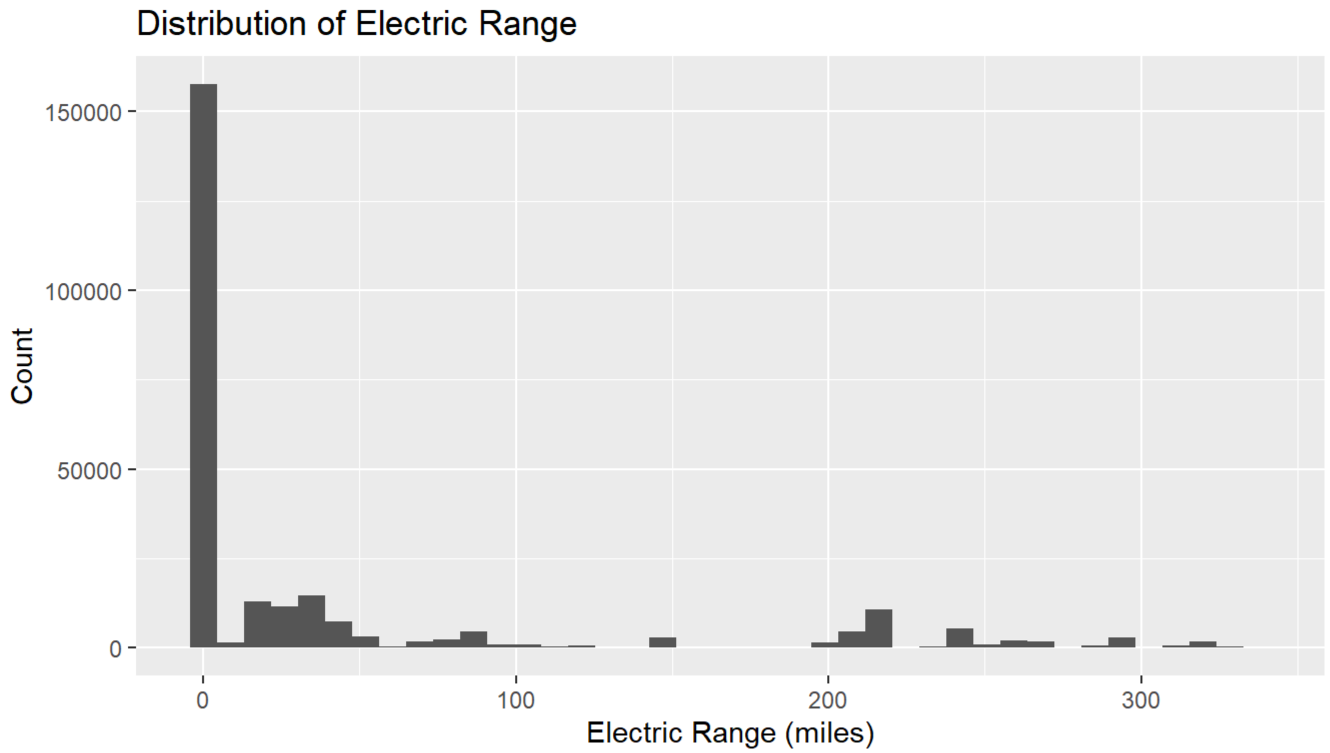
```
## # A tibble: 10 × 4
##    Make    `Electric Range` is_Tesla log1p_range
##    <chr>              <dbl>    <int>       <dbl>
##  1 TESLA                220        1        5.40
##  2 JEEP                  21        0        3.09
##  3 KIA                   26        0        3.30
##  4 BMW                   14        0        2.71
##  5 NISSAN                75        0        4.33
##  6 NISSAN                84        0        4.44
##  7 TESLA                210        1        5.35
##  8 TESLA                  0        1        0
##  9 NISSAN                84        0        4.44
## 10 PORSCHE               14        0        2.71
```

4) Plots

A) Univariate plot (Histogram of Electric Range)

```
ggplot(df, aes(x = `Electric Range`)) +
  geom_histogram(bins = 40) +
  labs(
    title = "Distribution of Electric Range",
    x = "Electric Range (miles)",
    y = "Count"
  )
```

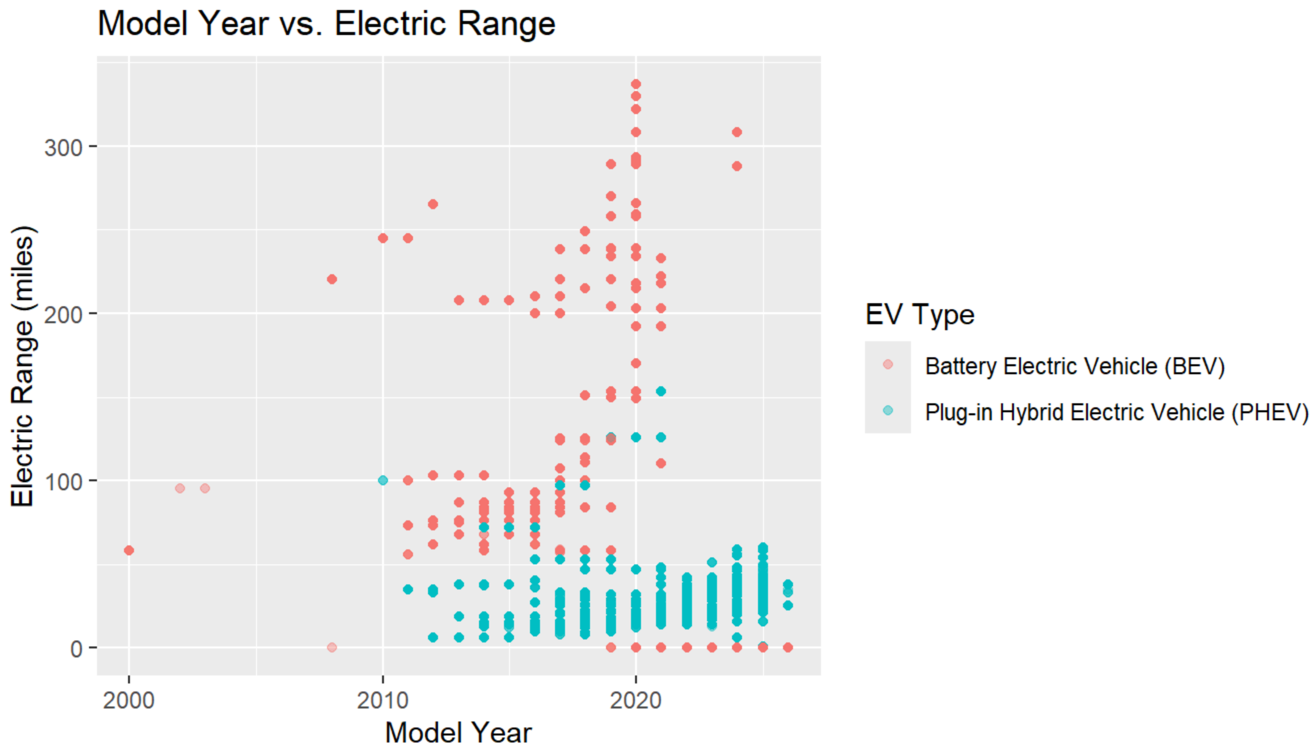
```
## Warning: Removed 3 rows containing non-finite outside the scale range
## (`stat_bin()`).
```



B) Scatterplot: Model Year vs Electric Range, colored by Electric Vehicle Type

```r
# Keep rows with both Model Year and Electric Range available
plot_df <- df %>%
  filter(!is.na(`Model Year`), !is.na(`Electric Range`))

ggplot(plot_df, aes(x = `Model Year`, y = `Electric Range`, color = `Electric Vehicle Type`)) +
  geom_point(alpha = 0.4) +
  labs(
    title = "Model Year vs. Electric Range",
    x = "Model Year",
    y = "Electric Range (miles)",
    color = "EV Type"
  )
```



5) Notes & Interpretation (brief)

- **Electric Range** histogram shows the typical driving range distribution for vehicles in the dataset.
- **Model Year vs Range**: newer model years generally trend toward higher ranges; BEVs typically offer higher range than PHEVs.
- **Transformations**: `log1p_range` can help normalize the distribution for modeling; `is_Tesla` can be used to compare Tesla vs non-Tesla vehicles on summary stats or plots.

Appendix

```
sessionInfo()
```

```
## R version 4.3.2 (2023-10-31 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 11 x64 (build 26100)
## 
## Matrix products: default
## 
## 
## locale:
## [1] LC_COLLATE=English_United States.utf8 
## [2] LC_CTYPE=English_United States.utf8   
## [3] LC_MONETARY=English_United States.utf8
## [4] LC_NUMERIC=C                          
## [5] LC_TIME=English_United States.utf8    
## 
## time zone: America/New_York
## tzcode source: internal
## 
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base     
## 
## other attached packages:
## [1] scales_1.3.0  ggplot2_3.5.0 dplyr_1.1.4   readr_2.1.5  
## 
## loaded via a namespace (and not attached):
##  [1] bit_4.0.5         gtable_0.3.4      jsonlite_1.8.8    highr_0.10       
##  [5] crayon_1.5.2      compiler_4.3.2    tidyselect_1.2.0  parallel_4.3.2   
##  [9] jquerylib_0.1.4   yaml_2.3.8        fastmap_1.1.1     R6_2.5.1         
## [13] labeling_0.4.3    generics_0.1.3    knitr_1.45        tibble_3.2.1     
## [17] munsell_0.5.0     bslib_0.6.1       pillar_1.9.0      tzdb_0.4.0       
## [21] rlang_1.1.3       utf8_1.2.4        cachem_1.0.8      xfun_0.41        
## [25] sass_0.4.8        bit64_4.0.5       cli_3.6.2         withr_3.0.0      
## [29] magrittr_2.0.3    digest_0.6.34     grid_4.3.2        vroom_1.6.5      
## [33] rstudioapi_0.15.0 hms_1.1.3         lifecycle_1.0.4   vctrs_0.6.5      
## [37] evaluate_0.23     glue_1.7.0        farver_2.1.1      fansi_1.0.6      
## [41] colorspace_2.1-0  rmarkdown_2.25    tools_4.3.2       pkgconfig_2.0.3  
## [45] htmltools_0.5.7
```