Automotive Data Analysis

# 1. Data Dimensionality

Below is a summary of the data dimensionality of the automotive dataset.

|  |  |
| --- | --- |
| Metric | Value |
| Total number of vehicles | 428 |
| Number of attributes | 12 |
| Missing Values | Make 0 Model 0 Type 0 Origin 0 DriveTrain 0 MSRP 0 EngineSize 0 Cylinders 2 Horsepower 0 MPG\_Highway 0 Weight 0 Length 0 |

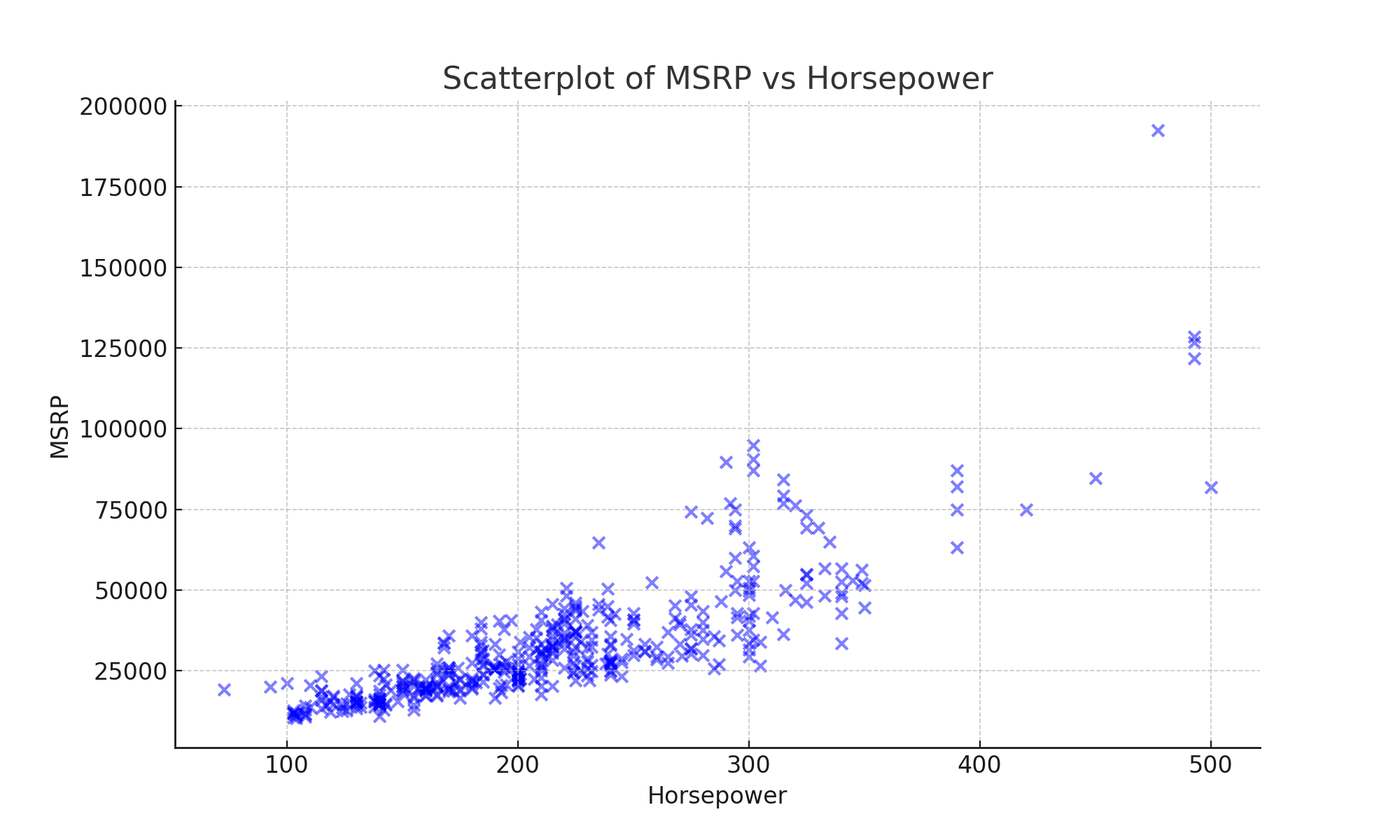
Data Types:

Make object  
Model object  
Type object  
Origin object  
DriveTrain object  
MSRP int64  
EngineSize float64  
Cylinders float64  
Horsepower int64  
MPG\_Highway int64  
Weight int64  
Length int64

# 2. Data Visualization

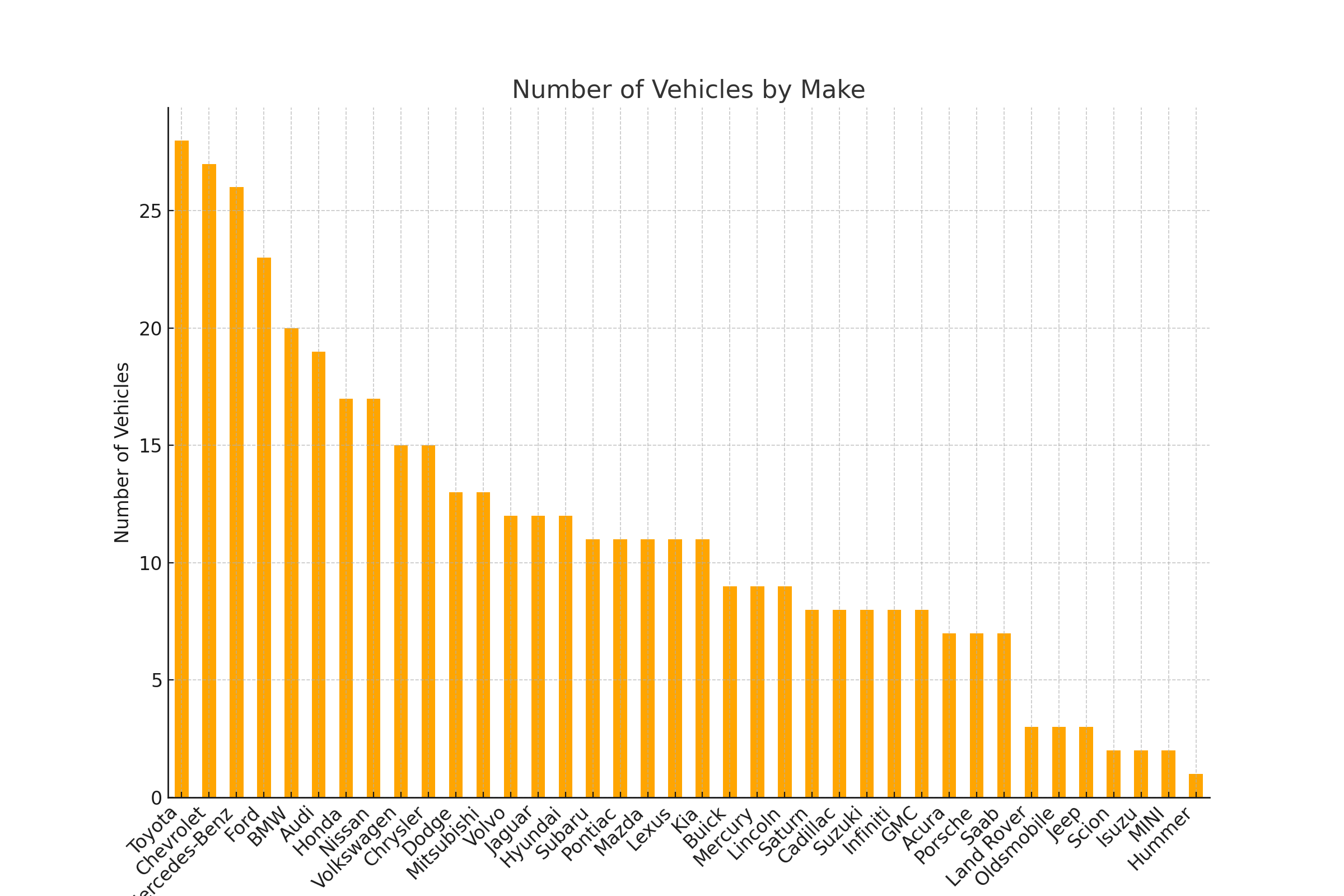
## Scatterplot of MSRP vs Horsepower

The scatterplot below shows the relationship between MSRP and Horsepower.



## Bar plot of Number of Vehicles by Make

The bar plot below shows the number of vehicles by each make.



## Make with the Greatest Number of Vehicles

The make with the greatest number of vehicles is Toyota, with a total of 28 vehicles.

# 3. Normalization and Standardization of Horsepower

The table below shows the original, normalized, and standardized values of the Horsepower variable:

|  |  |  |  |
| --- | --- | --- | --- |
| Index | Original Horsepower | Normalized Horsepower | Standardized Horsepower |
| 0 | 265 | 0.4496 | 0.6845 |
| 1 | 200 | 0.2974 | -0.2214 |
| 2 | 200 | 0.2974 | -0.2214 |
| 3 | 270 | 0.4614 | 0.7542 |
| 4 | 225 | 0.3560 | 0.1270 |

## Statistical Summary

Here is the statistical summary of the transformed data:

Original\_Horsepower Normalized\_Horsepower Standardized\_Horsepower  
count 428.000000 428.000000 4.280000e+02  
mean 215.885514 0.334626 -7.470660e-17  
std 71.836032 0.168234 1.001170e+00  
min 73.000000 0.000000 -1.991379e+00  
25% 165.000000 0.215457 -7.091854e-01  
50% 210.000000 0.320843 -8.202571e-02  
75% 255.000000 0.426230 5.451340e-01  
max 500.000000 1.000000 3.959670e+00

# 4. Discussion

## Differences between Normalization and Standardization

Normalization rescales the data to a fixed range, typically between 0 and 1. This is useful when features have different scales and you want to ensure that they contribute equally to a model, particularly in algorithms like k-nearest neighbors (KNN) or neural networks, where the magnitude of data values can influence the result. Standardization, however, centers the data by subtracting the mean and scales it by the standard deviation, leading to a mean of 0 and a standard deviation of 1. This transformation is more suitable for algorithms like SVM or PCA, which assume that the data is normally distributed or when features have different variances. In this dataset, normalization compressed the horsepower values into a small range, whereas standardization adjusted the spread of the data around the mean, making it easier to identify outliers and compare across features.

## Preferred Transformation Method

In this context, I prefer standardization because it maintains the original distribution of the data while ensuring that features are on the same scale. This is particularly important when analyzing datasets where the relationships between features are more significant than their individual scales. Standardization allows for better comparison between different attributes, especially when the data follows a normal distribution, and it is essential in statistical modeling techniques where the variance of data influences the results.