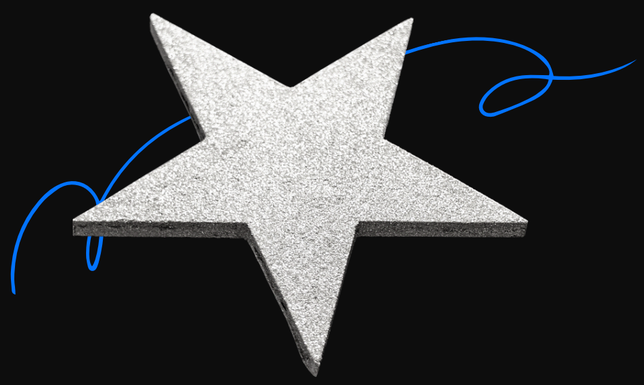
VEHICLE FUEL CONSUMPTION AND CO2 EMISSIONS

A DEEPER DIVE INTO CLUSTERING AND DATA VISUALIZATION: SOFIA GUTTMANN

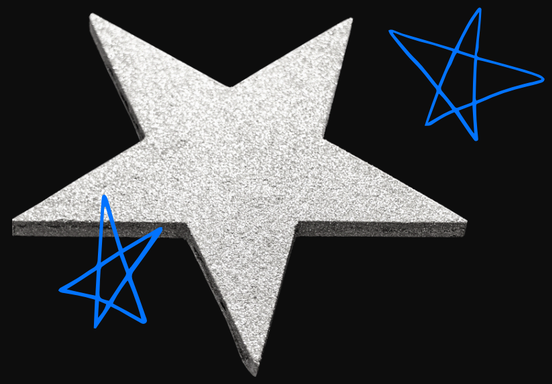
AGENDA

TOPICS COVERED



INTRODUCTION TO TOPIC

DATA WRANGLING

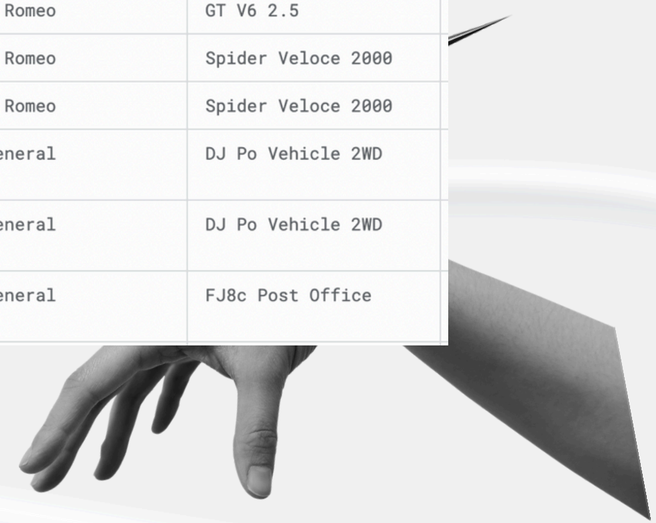


CLUSTERING

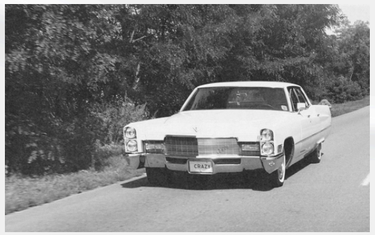
IMPLICATIONS

Detail | Compact | Column | 10 of 81 columns

vehicle_id id of vehcile	# year year	make make	model model name
 1 38.5k	 1984 2017	Chevrolet 10% Ford 8% Other (31148) 82%	**3748** unique values
26587	1984	Alfa Romeo	GT V6 2.5
27705	1984	Alfa Romeo	GT V6 2.5
26561	1984	Alfa Romeo	Spider Veloce 2000
27681	1984	Alfa Romeo	Spider Veloce 2000
27550	1984	AM General	DJ Po Vehicle 2WD
28426	1984	AM General	DJ Po Vehicle 2WD
27549	1984	AM General	FJ8c Post Office



This dataset offers a detailed compilation of engine specifications, fuel variations, and transmission setups across a range of vehicles. Additionally, it encompasses information on fuel efficiency and CO2 emissions, providing valuable insights for automotive enthusiasts and researchers keen on understanding both vehicle engineering and its environmental implications.



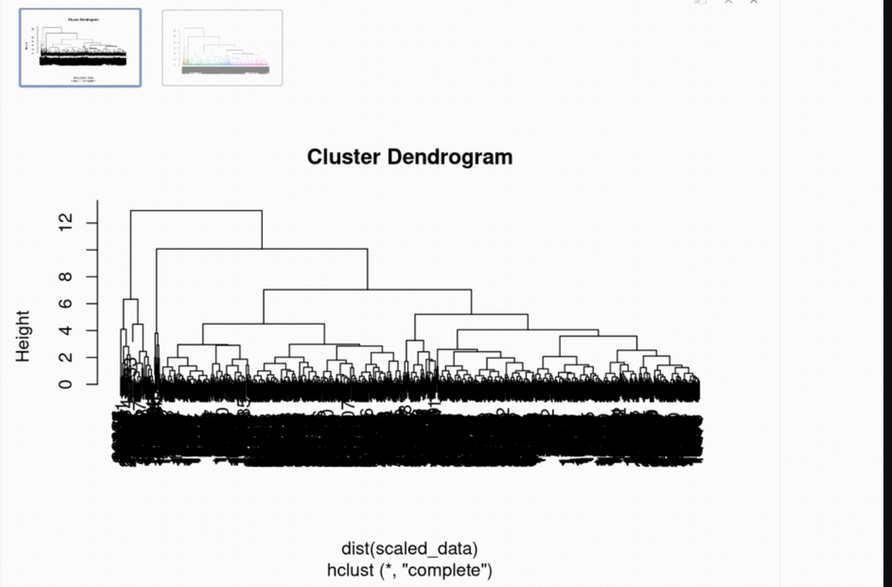
HCLUST: HIERARCHICAL CLUSTERING

```r
# Sample a subset of rows from your dataset
sampled_data <- Fuel_hclust[sample(nrow(Fuel_hclust), 1000), ]

# Scale the sampled data
scaled_data <- scale(sampled_data)

# Perform hierarchical clustering on the scaled data
hclust_result <- hclust(dist(scaled_data), method = "complete")

# Cut the dendrogram to obtain clusters
num_clusters <- 3  # Adjust the number of clusters as needed
cluster_assignment <- cutree(hclust_result, k = num_clusters)

# View cluster assignments
print(cluster_assignment)

# Perform hierarchical clustering with complete linkage
hclust_result <- hclust(dist(scaled_data), method = "complete")

# View cluster assignments
cluster_assignments <- cutree(hclust_result, k = 3)  # Adjust k as needed
print(cluster_assignments)
```

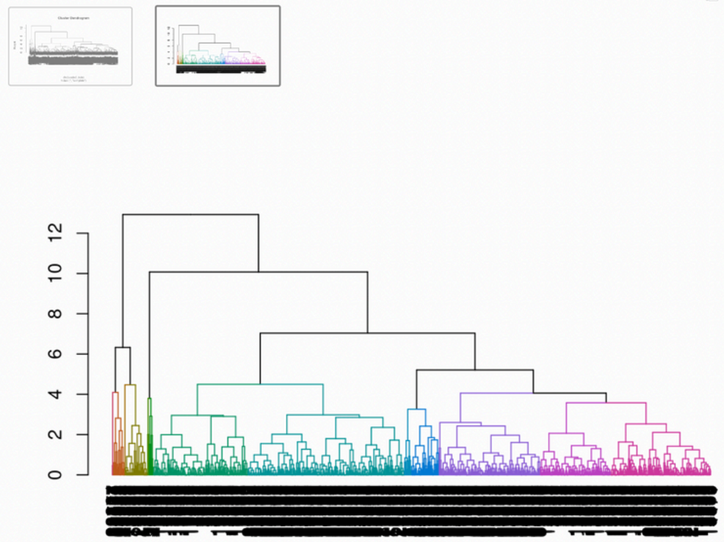
```
21816 19923 23721 33039 19875 26004 26635 19348 28667  3096 37541 11173 18170 14456 19793
33334 37266
    1     1     1     1     1     1     1     1     1     1     1     1     1     1     1
2     1
31527 37547  1667  5775 23678 24042 12780 24725  7397   309  2478 17861 12934 27673   848
16476  6646
    1     1     1     1     1     1     1     1     1     1     1     1     1     1     1
1     1
12868 32495 18234 23366 36887 36422 29507  2338 10419 10799  2401 17071 36040 27140 26611
3024 17896
    1     1     1     1     1     1     1     2     1     1     1     1     1     1     1
1     1
17890   562  2309 27095 17929  2641 19866 27433 35350 31085 147                 20434 18064
26623 24369
    1     1     1     1     3     1     1     1     1     1                       1     1
1     2
24659 38088 24467 15611 20323 31438 10588 32580  4147  6188 13532  747,          16929
22245 36212
    1     1     1     1     1     1     1     1     1     1     1     1     1     1     1
1     1
25312  2259 21307 25618 29350  3865 28681 15549 30885 18240 13018 37003 17126 11928 11007
35570 17303
    1     2     1     1     1     2     1     1     1     1     1     1     1     1     1
1     1
32765  9460  5076 30094  2224 19617  9647  1872  5748 18230 35784 32813   581 25490  9717
34314 32433
    1     1     1     1     1     1     1     1     1     2     1     2     1     1     2
1     1
 7896 37340 29868  8288 37810 16089 31934 18062 11354 15378  9343 14995 31473 11658 23785
11362 31136
    1     1     1     1     2     1     1     1     1     1     1     2     1     1     1
1     1
25161  7344 27860 32404 24985 36728 10486 29469  4808 20980  4266 33778  6719 36141   435
30454 36000
    1     1     1     1     1     1     1     1     1     1     1     1     1     1     1
1     1
19423 29688 12745  5587 36249 15875 16401 24361 17417 15110 21699 33466  9644 26606 25153
19266 28070
    1     1     1     1     1     1     1     1     1     2     1     1     2     1     1
1     1
14140 18620 19525  4234  1590 35041 34627 17770 17132 25525 16497 34429  5235 37680 23609
24983 35071
    2     1     1     1     1     1     1     1     1     1     1     1     1     1     1
1     1
20216 27434 12701  3252   259  9313 13329 36880  5576 28115  4810 27441 17540 13188 18097
17259 11106
    1     1     1     1     1     1     1     1     1     1     1     1     1     1     1
1     1
20205 26524 37740 31481 27685 10667 30953 26465  9310 11606 16255 37904 36983 30139 19253
25418 33829
    1     1     1     1     1     1     1     1     1     1     1     1     1     1     1
1     1
   60 28357 17664 28752 18639   217 14673 22238  1686 17685  5154 31452  5819 19188   496
34445  5117
    1     1     1     2     1     1     1     1     1     1     1     1     1     1     1
1     1
 5958 25626 13282 36341 11469  3929  3528 11797  5168 23583 27023   312 26985 25376 27150
34050 31564
    1     1     2     1     1     1     1     1     1     1     1     1     1     1     1
1     1
27130  3986  3413  9046    23 25092 26722 23924 33605 26947 21206 24557 25581 29236 17340
13215  4894
```

BACK TO **AGENDA PAGE**

```r
{r}
# Plot the dendrogram
plot(hclust_result)
suppressPackageStartupMessages(library(dendextend))
avg_dend_obj <- as.dendrogram(hclust_result)
avg_col_dend <- color_branches(avg_dend_obj, h = 3)
plot(avg_col_dend)
```



```r
{r}
# Plot the dendrogram
plot(hclust_result)
suppressPackageStartupMessages(library(dendextend))
avg_dend_obj <- as.dendrogram(hclust_result)
avg_col_dend <- color_branches(avg_dend_obj, h = 3)
plot(avg_col_dend)
```

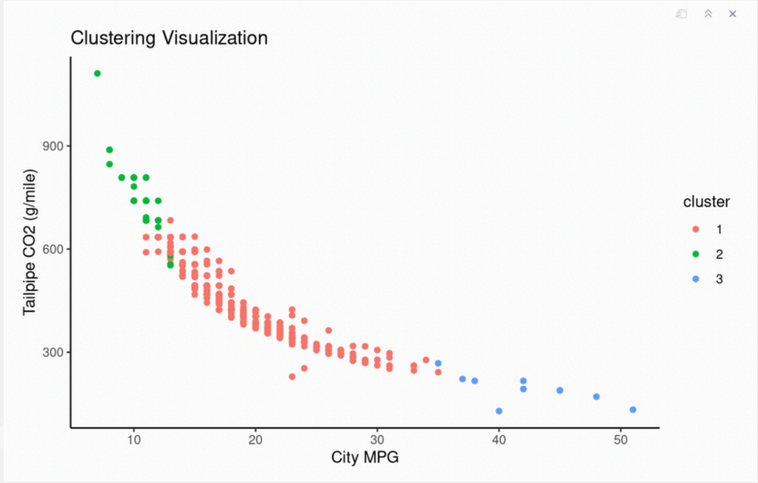


PLOT(HCLUST_RESULT):

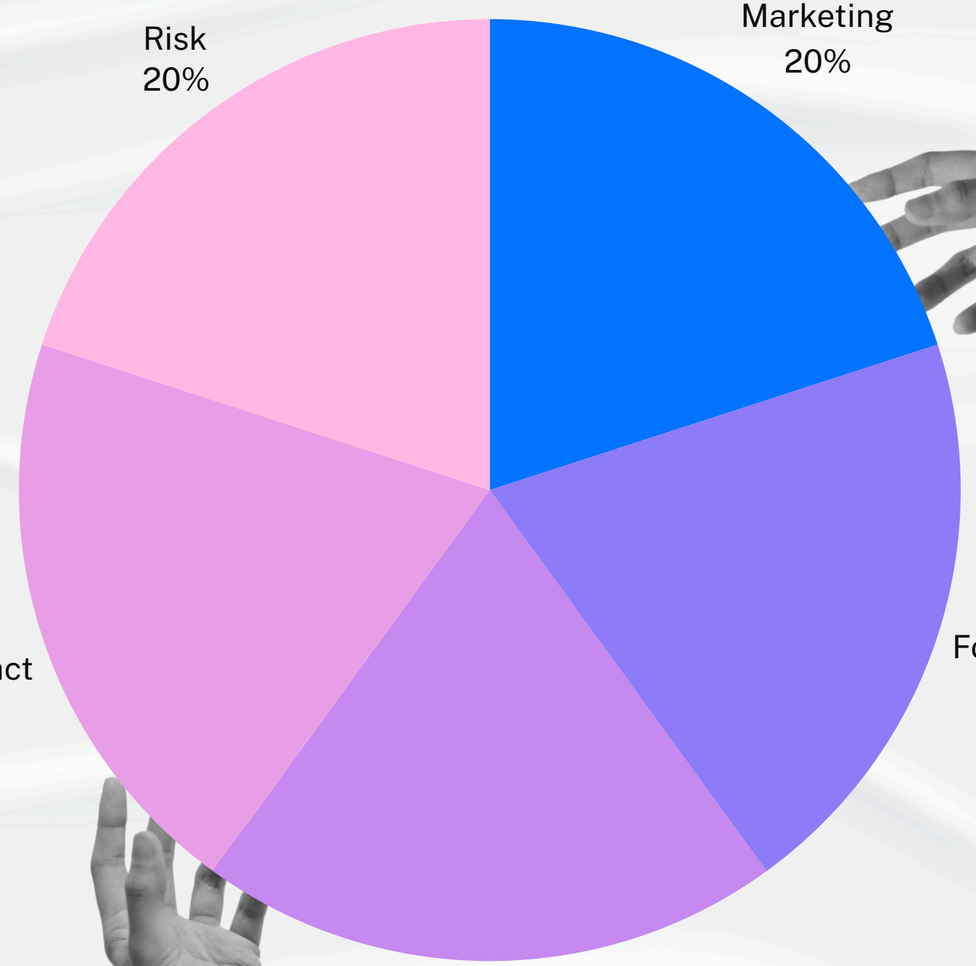
DENDEXTEND

IMPLICATIONS

```r
{r}
library(ggplot2)

# Add cluster assignments to the sampled data
sampled_data$cluster <- factor(cluster_assignments)

# Plot the data points with cluster assignments
ggplot(sampled_data, aes(x = city_mpg_ft1, y = tailpipe_co2_in_grams_mile_ft1, color = cluster
)) +
  geom_point() +
  labs(title = "Clustering Visualization", x = "City MPG", y = "Tailpipe CO2 (g/mile)")
```



Environmental Impact
20%



THANK YOU!