Exploring a car sales dataset in R

Exploring a car sales dataset in R

In this laboratory you will work with a larger, real-world style dataset containing information about car sales. You will practice downloading data from the internet, inspecting its structure, computing basic summaries, and creating visualisations – including boxplots of prices grouped by car make and by state.

1. Downloading and loading the dataset

For this exercise we assume that the car sales data is available as a CSV file at `https://raw.githubusercontent.com/ccfd/courses_data/refs/heads/stat1/car` The dataset contains many rows, each corresponding to a single sale transaction, with columns such as:

- `saledate` – date of the sale
- `state` – two-letter state code
- `make` – car manufacturer (e.g. “Ford”, “Toyota”)
- `model` – model name
- `sellingprice` – final sale price
- `odometer` – number of miles driven

Run the following code in R:

```
# URL of the car sales CSV file
url <- "https://raw.githubusercontent.com/ccfd/courses_data/refs/heads/s
file <- "cars.csv"
download.file(url, file)

# Option 1: read directly from the URL
car_sales <- read.csv(
  "cars.csv",
  stringsAsFactor = TRUE,
  colClasses = list(vin="character", saledate="POSIXct")
)

# Quick sanity checks
dim(car_sales)        # number of rows and columns
head(car_sales)       # first few rows
str(car_sales)        # structure and variable types
summary(car_sales)  # basic summaries for each column
```

Task 1.1: Run the code above.
Write down: - how many observations (rows) and variables (columns) the dataset has, - how many different car makes (`make`) and states (`state`) appear in the data.

Task 1.2: Check whether there are any missing values (`NA`) in the key columns:

```
colSums(is.na(car_sales[, c("sellingprice", "make", "state")]))
```

If you find missing values, decide (together with your instructor) whether you will drop them or replace them before further analysis.

2. Basic summaries of the dataset

In this section you will construct a simple numerical summary of the car sales.

2.1 Overall price distribution

Use the following code as a starting point:

```
# Overall summary of prices
summary(car_sales$sellingprice)

# Mean and standard deviation of prices
mean(car_sales$sellingprice, na.rm = TRUE)
sd(car_sales$sellingprice, na.rm = TRUE)

# Minimum, maximum and quantiles
quantile(car_sales$sellingprice, probs = c(0, 0.25, 0.5, 0.75, 1), na.rm = TRU
```

Task 2.1: - Report the minimum, median, and maximum price. - Report the mean and standard deviation of `price`. - Comment briefly: does the distribution of prices look symmetric, or does it have a long tail (e.g. a few very expensive cars)?

2.2 Summaries by make and by state

Compute summaries grouped by car make and by state using base R:

```
# Average price by make
avg_price_by_make <- tapply(car_sales$sellingprice, car_sales$make, mean
avg_price_by_make

# Average price by state
avg_price_by_state <- tapply(car_sales$sellingprice, car_sales$state, me
avg_price_by_state
```

Task 2.2: - Identify the **three makes** with the highest average price. - Identify the **three states** with the lowest average price. - Comment briefly on whether these differences look large or small in practical terms.

3. Visualising the data

Visual inspection is a key part of data analysis. In this section you will create basic plots for the car sales data.

3.1 Histogram and density of prices

```
hist(car_sales$sellingprice,
     breaks = 30,
     main = "Histogram of car prices",
     xlab  = "Price",
     col   = "lightblue")

# Optional: add a kernel density estimate
plot(density(car_sales$sellingprice, na.rm = TRUE),
     main = "Density estimate of car prices",
     xlab = "Price")
```

Task 3.1: - Create a histogram of `sellingprice`. - Based on the histogram (and optional density plot), describe the general shape of the distribution (e.g. unimodal, skewed to the right, etc.).

3.2 Boxplots of prices grouped by make

Boxplots are very useful for comparing distributions between groups.

```
boxplot(sellingprice ~ make,
        data = car_sales,
        outline = TRUE,
        las = 2,   # rotate labels for readability
        main = "Car prices by make",
        ylab  = "Price")
```

Task 3.2: - Generate the boxplot of `sellingprice` grouped by `make` as above. - Identify: - which makes have the highest median price, - which makes have the lowest median price, - whether any makes show many outliers (points far from the box).

If there are many makes and the plot becomes unreadable, you may: - restrict the plot to the most common makes, or - use `par(mar = c(10, 4, 4, 2))` to increase bottom margin before plotting.

3.3 Boxplots of prices grouped by state

Now compare price distributions between states.

```
boxplot(sellingprice ~ state,
        data = car_sales,
        outline = TRUE,
        las = 2,
        main = "Car prices by state",
        ylab  = "Price")
```

Task 3.3: - Create a boxplot of `sellingprice` grouped by `state`. - Identify: - which states have the highest median car prices, - which states have the lowest median car prices. - Comment on whether price variability (the height of the box and whiskers) is similar across states.

3.4 Boxplots of prices by make and state (combined)

Sometimes we are interested in how two categorical variables jointly affect the response. In base R we can use the `interaction()` function to combine factors.

```r
boxplot(sellingprice ~ interaction(make, state),
        data = car_sales,
        outline = TRUE,
        las = 2,
        main = "Car prices by make and state",
        ylab  = "Price")
```

This plot may be dense if there are many combinations of make and state, but it shows how price distributions differ across these groups.

Task 3.4: - Generate the combined boxplot using `interaction(make, state)`. - Choose three interesting combinations (e.g. the most expensive make in the most expensive state) and compare their median prices.

4. Additional exercises

1. **Filtering the data**
   Create a subset of `car_sales` that only contains:

   - cars with `sellingprice` greater than the **overall median** price, and
   - one selected make (choose any make that appears often in the data).
     For this subset:
   - compute the mean and median price,
   - create a histogram and a boxplot of `sellingprice`.

2. **Price vs. year**
   If the dataset contains a numeric column `year`:

   - create a scatter plot of `sellingprice` versus `year`,
   - compute the correlation between `sellingprice` and `year` using `cor()`,
   - comment on whether new cars tend to be sold for more.

3. **Saving plots**
   Use R to save one of your boxplots to a PNG file:

```r
png("car_prices_by_make.png", width = 800, height = 600)
boxplot(sellingprice ~ make,
        data = car_sales,
        outline = TRUE,
        las = 2,
        main = "Car prices by make",
        ylab  = "Price")
dev.off()
```

Check that the file has been created and can be opened with an image viewer.