Unsupervised Learning and Evolutionary Computation Using R

Winter Term 2024/2025

Exercise Sheet 2 (29th October, 2024)

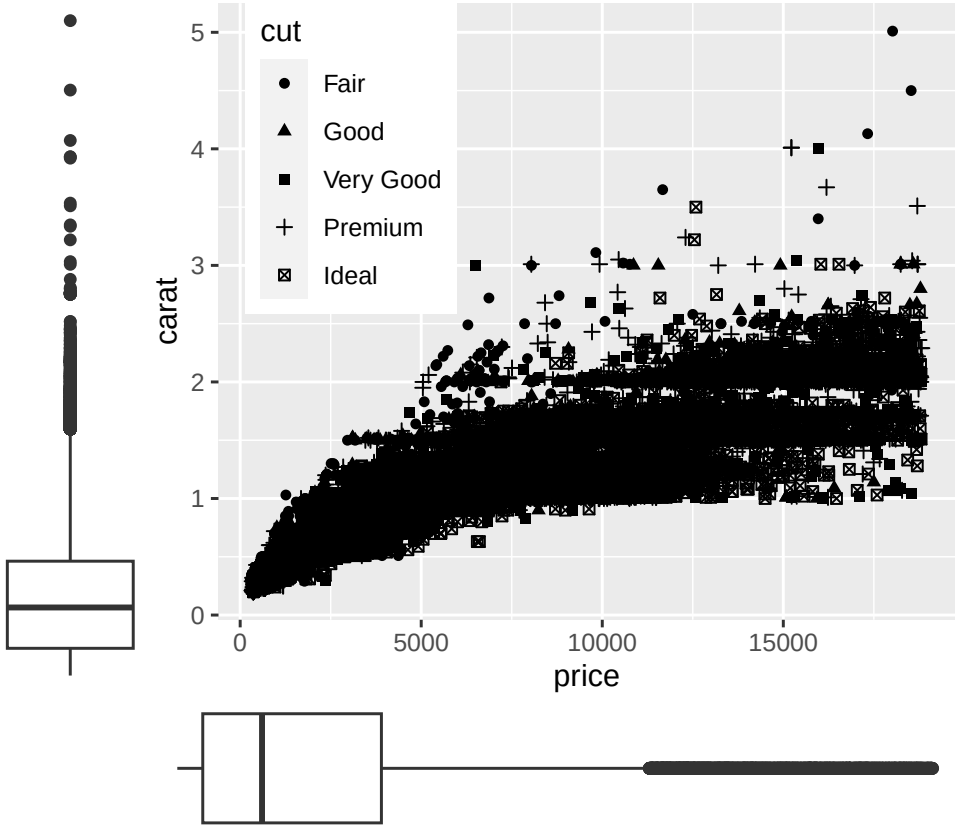
Exercise 1 (Data Cleaning with `tidyverse`)
Using the `nycflights13` dataset, analyse the average flight delays for different carriers and determine which carrier has the most consistent on-time performance.

1. Load the `nycflights13` package (install if needed) and load the flights dataset (`flights`).
2. Filter out flights that were cancelled or diverted, i.e., those with missing departure or arrival delay values (`is.na(dep_delay)` or `is.na(arr_delay)`).
3. Create a summary table including:
   - The carrier (`carrier`).
   - The average departure delay (`dep_delay`) and arrival delay (`arr_delay`) for each carrier.
   - The standard deviation of arrival delays for each carrier to measure consistency.
4. Filter to show only carriers with an average arrival delay of less than 5 minutes.
5. Arrange this table by the standard deviation of arrival delays in ascending order to identify the most consistent carriers.
6. Add an extra column showing the number of flights each carrier operated.
7. Rank carriers based on both average delay and consistency (standard deviation of arrival delay) to determine the best-performing carriers overall.

Exercise 2 (`ggplot2` – Plot Customisation)
In this exercise, you are going to use the `diamonds` data set.

1. Create a normal scatter plot with `carat` on the x-axis and `depth` on the y-axis.
2. Add information about the `colour` to your plot by colouring the points. The colour scale you are going to use is called 'RdYlGn' from the Brewer color palette. *Hint: In case you are not familiar with the Brewer colour palette, you can easily google how to use that in* `ggplot2`.
3. Increase the size of points corresponding to their `price`. Use four different increasing sizes for the values $5\,000, 7\,500, 10\,000, 15\,000$.
4. Add information about the `cut` to the plot by providing different shapes to the points.
5. Lastly, create this plot for each `clarity` level. Make sure, that the scales on each axis are suited to the displayed data! **Hint:** There is an appropriate parameter in `facet_grid` for that.

Exercise 3 (`ggplot2` – Multiple Plots)
In this exercise, you'll have to familiarise yourself with the package `cowplot`. This package allows you to combine multiple plots into a single view. Recreate the following plot for the `diamonds` data set already used in Exercise 3. Take a close look at the documentation of the function `plot_grid` and keep in mind that you can create nested plot grids!


cut
Fair
Good
Very Good
Premium
Ideal
carat
price
0
1
2
3
4
5
0
5000
10000
15000