Lab #1

Vittorio Zampinetti* Mauro Gasparini†

2022-10-11

Lab 1

Iris Dataset

Download the dataset at this link (https://archive.ics.uci.edu/ml/datsets/iris) (look for the `iris.data` file) and place the file together with your R script (for simplicity). At that web page, you can also get some information regarding the origin and nature of the data.

The dataset is available as Comma-Separated Values (CSV) file, which is nothing but a plain text file where each row is a row of a table and every column is separated by a comma. To make it look more like a CSV file, rename it from `iris.data` to `iris.csv`.

Import data

Now we're ready to import the data in R and view it as a table.

> Hint: use `read_csv()` function from the `readr` library. Check the function arguments. Also, you might need to manually add the column names. To check which column names should be added and in which order, check under **"attribute information"** on the dataset reference page linked above.

Answer these questions:

- How many observations does the dataset consist of?
- Which are the different classes?

Exploratory Data Analysis (EDA)

First compute the mean and standard deviation of each measure for each class separately.

- What can you infer? Is there any measure which is more indicative of a certain class?

You can also plot the empirical distribution of the four measures separately, in order to better visualize how far (or close) they are from each other.

- Plot the distributions of the four measures in a 2x2 grid, differentiating the types with color encoding (optional).

  Hint: You can use `melt()` from the `reshape2` library to transform the dataset, and then plot the various densities with color encoding on the four measures.

We can also visualize the four measures on a set of plots that shows the correlation between variables, two-by-two. This is easily done with a pair plot.

- Draw a pair plot (optional).

*Politecnico di Torino, vittorio.zampinetti@polito.it
†Politecnico di Torino, mauro.gasparini@polito.it

Hint: install the package with `install.packages("GGally")` and load it with `library(GGally)`, then read the help document of the `ggpairs()` function. You can also use R base `pairs()` function.

Confidence intervals

Let's make our decision more statistically relevant. Select only one class, the Setosa type, and one measure, petal length.

Now build a 95% CI around the mean value of the Setosa petal length. Assume σ unknown.

Formally, we want to find x_l, x_u s.t.

$$P(x_l \leq \bar{X} \leq x_u) = 95\%\,.$$

- Find such CI, using only R base operators (`mean`, `sd`).

To do this, remember that

- $\frac{(\bar{X}-\mu_0)\sqrt{n}}{s} \sim T(n-1)$ where s is the sample standard deviation
- `qt()` is the R function for the t-distribution quantile
- Validate your result by using `t.test()` to get the CI.

Compare this confidence interval to the mean and std-dev values you got for each class for the petal length measure.

- Do you think the petal length is a good measure to differentiate between Setosa and the other two types? Why?

P-value

Imagine you get measurements in terms of all 4 indicators of 5 iris flowers belonging to the same class, but you don't know which. These are the observations.

```
x_sample <- tibble(
  sepal_length = c(4.738759, 5.545983, 5.389729, 4.549803, 5.896723),
  sepal_width = c(3.132478, 3.537232, 3.217107, 3.190097, 3.636949),
  petal_length = c(1.220472, 1.321923, 1.573662, 1.289875, 1.705737),
  petal_width = c(0.23, 0.09, 0.33, 0.22, 0.18)
)
```

- Just looking at the values of the petal length, can you take advantage of the CI you just computed in order to make a guess about the class of these flowers? (Setosa or not Setosa)

To make this guess more statistically relevant, we have to quantify our confidence.

- What is the *p-value* of this sample, only looking at the petal length, against the null hypothesis that the samples are coming from the Setosa type?

  Remember that the *p-value* is defined as

$$P\left(T \geq \frac{(\bar{\mathbf{x}} - \mu_0)\sqrt{n}}{s} | \mu = \mu_0\right)$$

- Based on the value you found, what can you say about the class of this sample?

Now take this other sample:

```r
y_sample <- tibble(
  sepal_length = c(6.303990, 6.705969, 7.795044, 7.015665, 7.056670),
  sepal_width = c(3.152411, 2.499612, 3.293934, 3.275724, 2.843923),
  petal_length = c(6.356729, 5.975576, 4.998114, 5.423811, 5.382871),
  petal_width = c(1.589342, 2.305014, 2.260900, 2.185519, 1.589370)
)
```

- What is the *p-value*?
- Is it higher or lower than 0.025?
- Could we guess the answer to this last question without computing it?