Introduction

Machine Learning

uczeniemaszynowe@cs.uni.wroc.pl

14 Oct 2025, 10:00

lab

General Notes

- I suggest completing the tasks in `python` using `Jupyter` notebooks, with appropriate packages such as `matplotlib` and/or `seaborn` and/or `plotly` (for plotting), and `pandas` and/or `numpy` (for loading and processing data).
- Please ensure the readability and interpretability of your plots: proper axis labels, clear data descriptions, sensible color choices, appropriate font sizes, etc.
- Be prepared to answer the questions included in the tasks for the instructor, based on the data and plots you have prepared.

Task 1 [2.5 pts]

0. Find and download the IRIS dataset from the UCI Machine Learning Repository. Learn what the downloaded data describe (hint: read the `iris.names` file; Wikipedia illustrations can clarify biological terms).
1. What values does *sepal length* take? Make a scatter plot: X–axis is the sample index, Y–axis is *sepal length*. Does it make sense to connect the dots in this plot?
2. Check how many samples of class *Iris setosa* have *sepal width* less than 2.5. And between 2.5 and 3.0? And between 3.0 and 3.5? etc. Make an appropriate (bar) plot. Did you actually make a histogram?
3. Make a scatter plot: X–axis *sepal length*, Y–axis *sepal width*. Does it make sense to connect the dots in this plot? Color the points with three different colors depending on the iris species. Set point sizes proportional to *petal length* (choose the size scale so the figure is legible). What interesting patterns do you see? Can you use the plot to propose a rule that distinguishes *Iris setosa* from the other two species?
4. Create the plot from the previous point for *every pair of distinct attributes*. Consider arranging the plots side by side rather than one under another.
5. Find out what a *violin plot* is and make one for the attributes of the dataset.

Task 2 [2.5 pts]

0. Find and download the BANK MARKETING dataset from the UCI Machine Learning Repository. Learn what the downloaded data describe.
1. Plot a histogram of the `duration` attribute for the entire dataset. Then make two analogous histograms, separately for positive and negative samples (attribute `y` equal to `yes` or `no`, respectively). What interesting patterns do you see in these plots?

2. Plot a histogram of the `balance` attribute for people over 25 years old (`age` > 25). Get familiar with the `ipywidgets` package, then turn your plot into an interactive one with a slider to choose the age threshold and observe how the histogram changes.

3. For a fixed threshold $t = 360$ of call duration (`duration`), select samples with `duration` above t. Compute what percentage of them are positive samples. Repeat for various values of t and plot: X–axis is threshold t, Y–axis is the percentage of positive samples. Try to explain the phenomenon visible in the plot.

4. Repeat the previous point for the `balance` attribute. Imagine you are running a marketing campaign for bank term deposits. You can contact clients selected uniformly at random from the whole dataset, or only clients with `balance` above a chosen threshold (you may pick the threshold). Which strategy is better? What threshold would you choose?

5. Randomly split the dataset into two parts of comparable size. Using the first part, make the plot described in point 4 and choose a threshold (according to your idea). Assume that clients with `balance` above the chosen threshold should be interested in a term deposit (should be positive samples). Compute what percentage of them were *not*. Also compute what percentage of interested clients had `balance` not greater than the chosen threshold. Then compute these errors on the second part of the data. Repeat for different threshold values. Create a plot (your design) presenting the results. Ultimately, which threshold should be chosen?

Task 3 [2 pts]

0. Find and download the CAR EVALUATION dataset from the UCI Machine Learning Repository. Learn what the downloaded data describe.

1. In a pie chart, show how many of all cars are: unacceptable, acceptable, good, and very good.

2. In a bar chart, show how many cars with 2, 3, and 4 doors there are across different safety classes.

3. Find out what a *radar plot* (also called a *star* or *spider* plot) is, and make one for 5 selected attributes of the data.

4. Try to determine which values of which attributes decide that a car is good or very good. Do this intuitively, according to your own idea—later in the course you will learn how to do this with machine learning methods. Formulate a rule that, based on the values of several chosen attributes, says **YES** (good or very good) or **NO**. Does your rule work correctly for all the data or only for a subset? For how large a subset? Create a figure (your design) showing where the rule makes mistakes.

Task 4 [1.5 pts]

0. Recall (or listen to the lecture recap) what the normal distribution is.

1. Generate one million numbers from the normal distribution with mean 0 and variance 1. Draw a histogram of these data.

2. Overlay the probability density function on the histogram of the generated data. Compare the histogram with the density function.

3. Recall point 3 from Task 1. Add random noise with a normal distribution of mean 0 and standard deviation $s = 0.25$ to the *sepal length* and *sepal width* attributes. See how the plot changes. Does your proposed rule for distinguishing *Iris setosa* still work? Try other values of s as well.

Task 5 [1.5 pts]

0. Recall (or listen to the lecture recap) what the multivariate normal distribution is.

1. Generate one million points from the bivariate normal distribution with mean $(0, 0)$ and the identity covariance matrix. Draw a histogram of these data (hint: the histogram will be a 3D plot).

2. Overlay the probability density function on the histogram of the generated data (hint: this will be a 3D plot). Compare the histogram with the density function (hint: an interactive plot that allows rotation and zooming will be convenient; `plotly`, for example, supports this).