

INF264 - Homework 1

Pierre Gillot Natacha Galmiche*

Week 34 - 2021

Instructions

Since it is the first exercise, instructions will be as detailed as possible. An **optional** jupyter notebook is available too with a template and hints about which functions to use.

- **Format:** Jupyter notebook
- **Deadline:** Friday 03rd September, 23:59
- **Submission place:** https://mitt.uib.no/courses/30808/assignments/48247?module_item_id=283728

1 k -NN for a classification problem on the Iris dataset

Iris is a small dataset consisting of 150 vectors describing iris flowers, split into three different classes representing three species of the iris family. Each vector comes with a label (the name of the species) and a set of four features which are measurements of different parts of the flower.



Left: The three species in the Iris dataset

Right: The four features in the Iris dataset (petal and sepal width and length)

Those measurements tend to differ between the different species, thus it is possible to train and evaluate a classifier from this dataset whose task is to predict the species of an iris flower represented by aforementioned set of features. In this exercise we will use k -NN classifier.

1. Iris Dataset:

- (a) Load the Iris dataset directly from sklearn. You can alternatively download the dataset here: <https://archive.ics.uci.edu/ml/datasets/iris>
- (b) Store the first 2 features (sepal length and sepal width) in a matrix X and labels in a vector Y .
- (c) Split the dataset into 3 datasets: training set, validation set and a testing set, i.e. split X and Y into X_{train} , X_{val} , X_{test} and Y_{train} , Y_{val} , Y_{test} respectively. You can for instance use a train/validation/test ratio of 0.7/0.15/0.15.

*Not a TA for this course this year.

2. Perform a k -NN classification of your dataset for each k in 1, 5, 10, 20, 30:
 - (a) Plot both training and validation Iris datapoints with respect to the two selected features. Since there are three classes, you will need three different colors.
 - (b) Create an instance of the KNeighborsClassifier class
 - (c) Train your instance of k -NN on your training data set
 - (d) Plot the decision boundaries as decided by the trained k -NN.
 - (e) Compute model accuracy on training dataset and validation dataset
 - (f) Which model (i.e which k) would you select? Compute model accuracy on testing dataset
3. Interpretation:
 - (a) Plot a curve representing the training accuracy as a function of k and same for the validation accuracy.
 - (b) From your observations, for which values of k does k -NN overfit ?
 - (c) For $k = 1$, k -NN train accuracy should be equal to 1 (100% correct predictions). Explain why this is not the case here.