# LELEC2870 - Machine Learning Project: Heart failure on the rise in the Smurf society

Academic year 2025-2026

## Introduction

All was well in the peaceful world of the Smurfs - until a few decades ago, when they began mingling a bit too much with humans. Their habits soon changed, and with these new lifestyles came unexpected problems, including a rise in cardiovascular diseases.

Doctor Smurf, the village's devoted physician, has been working tirelessly to diagnose and prevent these heart issues. Unfortunately, with so many Smurfs to look after, he's now completely overwhelmed. Training new apprentices would be great, but he needs a solution now. Desperate for help, he turned to his most trusted human friend - you.

Recently, Doctor Smurf overheard rumors about a powerful human technology known as machine learning. Intrigued, he began to wonder whether it could help him assess heart-failure risk more efficiently. As luck would have it, you're currently taking a machine learning course — talk about perfect timing! Even better, Doctor Smurf has already gathered a dataset from his patients, including clinical information, 48×48-pixel heart scans captured with a custom-built Smurf-sized MRI machine, and, of course, is own risk assessments. This seems like the perfect opportunity to put your skills to the test.

Your mission: train a machine learning model capable of accurately predicting the risk of heart failure among Smurfs.This document provides guidance to help you get started, a description of the data, and details about the expected outcomes. Read it carefully — an entire village of little blue creatures is counting on you! (No pressure, of course.)

# Dataset

To complete your task, you will need to work with several data files. These are available on Moodle. In the `labeled_data` folder, the files `X_train.csv` and `X_test.csv` contain medical data stored as a table. Each line/row corresponds to a Smurf and each column to a measured attribute/variable/feature. A description of each variable is given bellow:

| | |
|---|---|
| **age** | Age (can be well over 100 for Smurfs) |
| **blood pressure** | Systolic blood pressure (in mmHg) |
| **calcium** | Level of calcium in blood (in mmol/dL) |
| **cholesterol** | Level of LDL cholesterol ("bad cholesterol") in blood (in mg/dL) |
| **hemoglobin** | Level of hemoglobin in blood (in g/dL) |
| **height** | Height (in cm) |
| **potassium** | Level of potassium in blood (in mmol/L) |
| **profession** | Professional occupation (various) |
| **sarsaparilla** | Consumption of sarsaparilla leaves (very low - low - moderate - high - very high) |
| **smurfberry liquor** | Consumption of smurfberry liquor (very low - low - moderate - high - very high) |
| **smurfin donuts** | Consumption of smurfin donuts (very low - low - moderate - high - very high) |
| **vitamin D** | Level of vitamin B in blood (in ng/ML) |
| **weight** | Body mass of Smurf (in grams) |

The risk of developping a heart failure within the next ten years is the target variable; it is stored in the `y_train.csv` and `y_test.csv` files. The indices match those of `X_train.csv` and `X_test.csv`. The last element of each line in `X_train.csv` and `X_test.csv` is the name of the image file that contains the corresponding heart scan. These images are stored in the folders `Img_train` and `Img_test`. They correspond to transaxial views of the heart (a slice perpendicular to the long axis of the body), looking up from the feets. Refer to Figure 1 for a visual explanation.

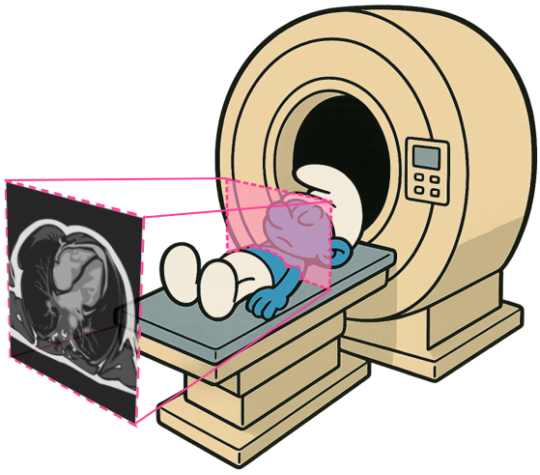

Figure 1: Transaxial Smurf MRI

All these files form the labeled dataset on which you will train your models and estimate their generalization performance. You will find additional data in the folder `unlabeled_data`, for which you do not have the labels (but we do ☺). This second dataset will be used for the evaluation of your best model.

# Instructions

You may complete this project individually or in teams of two. The project consists of **four parts**:

- **Part 1:** Work with tabular data and implement a linear model.
- **Part 2:** Continue using tabular data, but implement a nonlinear model.
- **Part 3:** Integrate image data into your analysis.
- **Part 4:** Take a broader perspective to better understand the growing problem of heart failure within Smurf society.

You must code in Python. You do not need to implement everything from scratch — you may use any appropriate library (e.g. `scikit-learn` for traditional models or `pytorch` for deep learning). In addition to the code, you must write a report summarizing your methodology, results, and conclusions. Below are detailed guidelines for each part.

**Part 1 — Linear Model (Baseline)** Your first task is to prepare the dataset for analysis. This includes: removing features that are clearly irrelevant for prediction; encoding categorical variables so they can be processed by regression models; and transforming numerical variables if necessary (e.g. normalization or standardization). Pay special attention to the preprocessing stage, as it is essential for building robust models and will be reused in later parts. Next, proceed with feature selection and model selection (if applicable). In your report, describe and justify all your choices, clearly present your results, and discuss them critically.

**Part 2 — Nonlinear Models** You will now compare several types of nonlinear models (e.g. non-parametric, tree-based, neural networks, etc). Note that some nonlinear models are sensitive to uninformative features — good feature selection may be useful. Explore various selection methods beyond simple correlation filters. In your report, discuss: which features are most important? Are they different from those in Part 1? Did you modify your preprocessing pipeline?

Also, nonlinear models typically involve many hyperparameters, so model selection and tuning will be more intensive at this stage. For each model: define a relevant set of hyperparameters and fine-tune them within the limits of your computational resources. Be mindful of data splitting for model selection. Clearly explain how you partition the data into training and validation (the test set is provided). Use cross-validation when computationally feasible. Based on your experiments, identify the best-performing model and estimate its generalization performance. In your report, clearly indicate whether reported metrics correspond to the training, validation, or test set.

**Part 3 — Integration of Image Data** In this part, you will integrate heart scan images into your pipeline. Extract features from the images using a deep neural network and combine these image-derived features with the tabular data. Retrain (and, if necessary, adapt) your best nonlinear model from Part 2 on this combined dataset. Compare performance with and without the image features, and discuss your findings. We will cover how to implement a convolutional neural network for image data during the practical session in Week 9 ☺.

**Part 4 — Understanding Heart Failure in Smurf Society** In the final part, you will take a broader analytical perspective. Your goal is to: formulate hypotheses about the causes of heart failure, identify groups of Smurfs most at risk; and support your analysis with clear visualizations and graphs. You are free to use any analytical or visualization tools you find appropriate and may build upon results from the previous parts.

## Deliverable

At the beginning of week 8 (Tuesday, November 4), you will have to deliver:

- **Predictions**. Once your linear model is properly trained, you are asked to produce predictions on the data from `X.csv` (in the `unlabeled_data` folder) for which we have kept secret the corresponding targets. This prediction vector should be uploaded on Moodle in a csv file named `y_pred.csv` that contains one prediction per line and no header (no quotation marks around your numbers either). Check that your format is correct by opening it with a text editor and compare it to `y_test.csv`. We will use **RMSE** as the evaluation criterion.
- **Report**. A report of maximum one page describing what you did for part 1. No worries, you will be able to modify it for the final deadline.

By the end of week 12 (Friday, December 5), you will have to deliver:

- **Predictions.** Choose your best model among part 1, 2 and 3 and use it to produce predictions on the data from `X.csv` (in the `unlabeled_data` folder).This prediction vector should be uploaded on Moodle in a csv file named `y_pred.csv`. We will use **RMSE** as the evaluation criterion. **If you transform the target at some point, do not forget to apply the inverse transform before estimating the generalization performance and/or before making your predictions.**
- **Report**. You will produce a report documenting your technical choices and experimental results. We do not need a course on the methods you use. We are more interested in what you did and why. Be concise and go straight to the point! Follow the structure: Introduction, Part 1, Part 2, Part 3, Part 4, Conclusion. A strict **maximum of 6 pages** will be observed.
- **Code**. Also on Moodle, you should submit a compressed folder containing all your python scripts (notebooks, utils.py, etc). Theses should be runnable and contain at least what you discussed in your report. There is no size limit, but these files should be structured, commented, and clear enough so that information can be easily found without deciphering everything! If you used any packages that were not used during the practical sessions, or a different version of those, don't forget to mention it in the beginning of your file.

## Schedule

Below you will find the schedule for the project.

- As soon as possible: Register your group (maximum two people) on Moodle
- Tuesday 04/11 at 23h55: Intermediate deadline where you submit your work for part 1 as 2 seperate files (a csv file for you first predictions and a pdf for a "pre-report" on part 1)
- Thursday 6/11 at 8h30: Q/A session #1
- Thursday 20/11 at 8h30: Q/A session #2
- Friday 5/12 at 23h55: final deadline where you submit your work as 3 separate files (a csv file for your predictions, a pdf for your report, and a compressed folder for all your scripts)

Do not wait until the last minute to start, and take advantage of the Q/A sessions for asking your questions and receiving feedback. We also encourage you to discuss about the project with other groups. We do not want to see plagiarism, but we certainly value exchange of ideas and experiences. Remember to cite all your sources. Use LLM's wisely, don't let them fool you.

## Evaluation

The project will account for half of the points in this course (10/20). Here is a rough idea of the weighting of each section: Part 1 (2/10), Part 2 (3/10), Part 3 (3/10) and Part 4 (1/10). Performance of your best model on the unlabeled data will account for the last point (1/10). This weighting may be subject to small variations. Finally you will be able to earn one small bonus point if your linear regression submitted on the intermediate deadline performs well and granted that you respect the required file formats. Also, we really insist on the quality of the report; be concise, clear, justify your choices and interpret your results! Embrace this mantra: a good project with a bad report is a bad project!