Classification of Cardiac Pathologies from MRI Data – IMA205 Challenge Report

Anissa El Idrissi–Feissel
Nickname : Anissa El Idrissi
Supervised by Loic Le Folgoc and Pietro Gori

May 2025

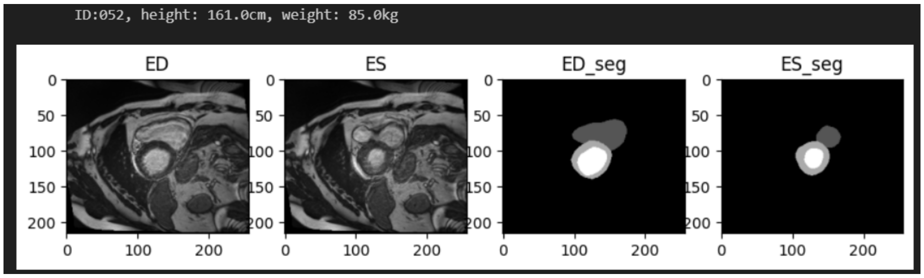
Contents

1 Introduction

Cardiovascular diseases are among the leading causes of mortality worldwide. Accurate, automated classification of cardiac pathologies from MRI data can support clinical diagnosis and improve patient outcomes. This challenge aims to develop such an automated classification system using the provided datasets.

The data given in this challenge are made up of two datasets :

- **Training dataset:** consisting of 100 folders, each representing a patient, composed of:
  - the patient ID (from 1 to 100),
  - two MRIs of the patient, one taken during diastole and the other during systole,
  - the segmented images of the MRIs, with the segmentation of:
    - the right ventricle,
    - the myocardium,
    - and the left ventricle
  - the patient's height and weight, respectively in cm and kg, in a CSV file,
  - the patient's true label (from 0 to 4), also in the CSV file.
- **Test dataset :** composed of 50 folders, each representing a patient, composed of:
  - the patient ID (from 101 to 150),
  - two MRIs of the patient, one taken during diastole and the other during systole,
  - the segmented images of the MRIs, with the segmentation of:
    - the right ventricle,
    - and the myocardium (but not the left ventricle),
  - the patient's height and weight, respectively in cm and kg, in a CSV file.



```
ID:052, height: 161.0cm, weight: 85.0kg
```

Figure 1: Example of what contains the folder of the patient 052

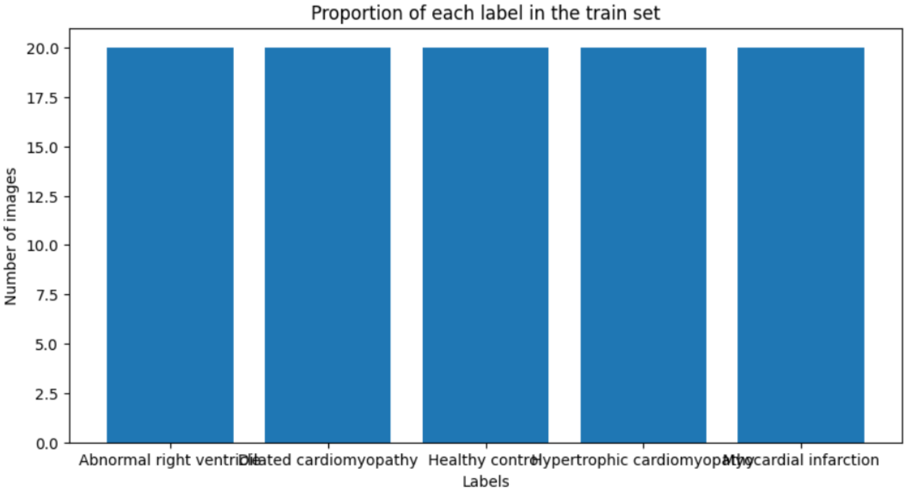


Figure 2: Distribution of each label in the training data

Here is, above, an example of what the datasets contain (Figure 1).

We begin by examining the dataset a little further by plotting the distribution of each label in the train set. We observe that our train set is evenly distributed, thus we will not need any oversampling to avoid a potential bias (Figure 2).

2 Preprocessing

First, I checked that no part of the data is missing. Thus, I will not implement any function to check any lack in the dataset and to raise an error when it is the case.

To prepare the test data for analysis, I implemented a structured preprocessing routine that combines image loading and metadata extraction. Initially I wanted to use a DataFrame but I abandoned that idea because I found it too difficult. Then, I first loaded the metadata from the metadataTest.csv file using NumPy's loadtxt function, which retrieves the patient IDs, heights, and weights. Each patient is identified by a unique ID corresponding to a folder containing their cardiac MRI data. For every subject in the dataset, I iterated through both cardiac phases—end-diastole (ED) and end-systole (ES)—and loaded the corresponding raw images and segmentation masks using nibabel. Specifically, I accessed four files per subject: ED.nii, ES.nii, ED-seg.nii, and ES-seg.nii. These were stored in a dictionary structure under the patient's ID, along with their height and weight. This design allows for efficient access to both the imaging data and associated metadata during feature extraction and classification, while ensuring that all relevant files are consistently and correctly linked to each subject.

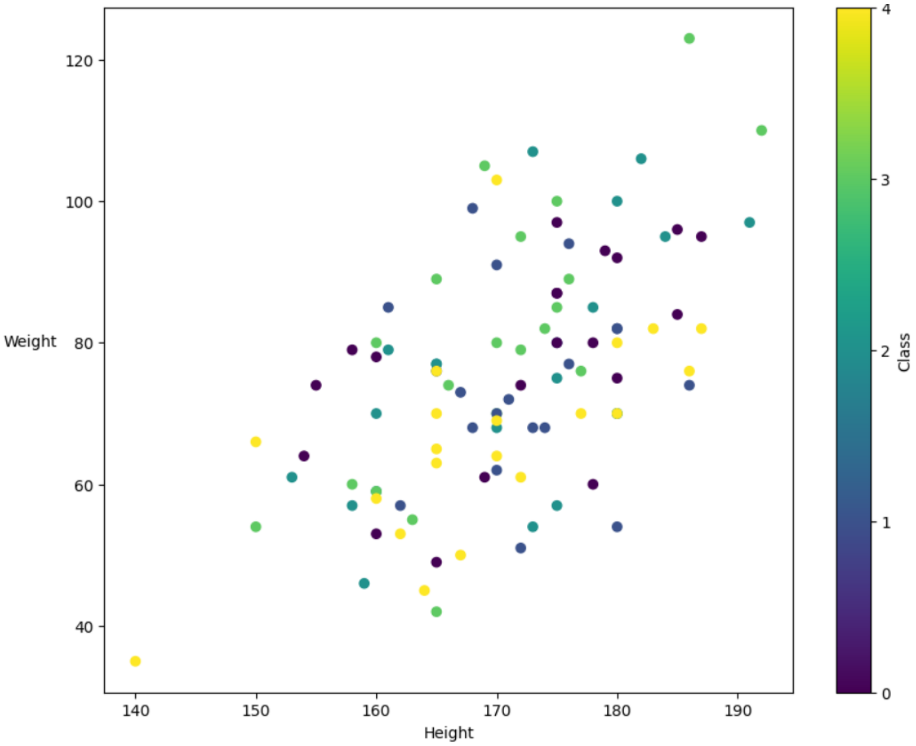


Figure 3: Variability of the classes only using height and weight

3 Extract features

3.1 Height and weight

In all my methods, my features are stocked in a dictionary where the key is the name of the feature and the value is the value of the feature.

As explained before, two features are already given in both the data sets : the patient's height and weight. Thus, I first tried a very simple model only using both those features and a random forest classifier. As expected, the model achieved low accuracy, indicating that height and weight alone are insufficient to discriminate between pathology classes. That is indeed observed in the Figure 3.

3.2 New features extraction

As explained in the previous part, we need to improve our model by extracting new features in order to classify more efficiently. Thus, I tried several features described in the article provided in this challenge [1] and made the graph of importance in order to delete the least important ones. Let us describe those features.

First, I decided to segment the left ventricle in my test data set. In order to achieve that, I employed a morphological approach based on flood fill. Specifically, I first isolated the myocardium (label 2) and applied flood fill from a corner pixel, assuming that the background surrounds the heart. This allowed me to identify the region enclosed by the myocardium but not part of it, which corresponds to the left ventricle cavity. This method was chosen over more complex machine learning or atlas-based techniques due to its simplicity, speed, and reliance on the already accurate myocardium segmentation provided. Moreover, it is robust in cases where the left ventricle is fully enclosed by the myocardium, making it effective in well-segmented clinical images without requiring additional annotations or training data.

Then, from the segmentation obtained, a comprehensive set of quantitative features was extracted for each patient. These features include:

- Left and right ventricular volumes measured during diastole and systole
- Myocardial volume at both end-diastolic and end-systolic phases
- Global cardiac function metrics:
  - Stroke Volume (SV)
  - End-Diastolic Volume (EDV)
  - Ejection Fraction (EF)
- Volume-based ratios:
  - Myocardium-to-left-ventricle volume ratio
  - Right-to-left ventricular volume ratio
  - Indexed volumes normalized by Body Surface Area (BSA)
- Derived anatomical metrics such as overall heart size

To derive clinically relevant features from the segmented MRI data, I computed anatomical volumes using voxel counting. Specifically, each 3D segmented image is composed of labeled voxels, where each label corresponds to a different cardiac structure. The volume of a given structure is calculated by counting the number of voxels assigned to its label and multiplying by the volume of a single voxel (derived from image metadata).

For the left ventricle, whose label is 3, I computed:
The end-diastolic volume (EDV) as:

$$volume_{ed} = \sum(img_{ed} == 3) * voxel_{volume}$$

The end-systolic volume (ESV) as:

$$volume_{es} = \sum(img_{es} == 3) * voxel_{volume}$$

Similarly, for the myocardium (label 2), I calculated myocardial volume at end-diastole and end-systole using the same voxel-counting approach. I computed voxel volume as the product of the spatial resolution in each dimension, extracted from the image header.

This method offers a straightforward and robust way to extract volumetric measurements from labeled images without the need for geometric approximations. It ensures that the volumes reflect the actual segmented regions in the images, making them suitable for computing derived metrics such as ejection fraction or myocardial mass.

The other features were computed using NumPy operations applied to the labeled 3D MRI volumes.

3.3 Relevance of Extracted Features

The features extracted from the segmented cardiac MR images are designed to capture physiologically meaningful characteristics of the heart that are directly linked to cardiac function and pathology.

These features are known biomarkers in cardiology and are commonly used to evaluate the presence and severity of cardiovascular diseases. For instance, the ejection fraction is a standard metric for assessing heart failure, while abnormal myocardium-to-ventricle ratios may reflect hypertrophic or dilated cardiomyopathies. Furthermore, by indexing volume measurements to body surface area (BSA), we ensure normalization across patients of different sizes, improving the robustness of the features.

The inclusion of additional features, such as the ratio between the RV and LV volumes or between the myocardium in different cardiac phases, enables the model to capture subtle anatomical and functional differences that may be difficult to observe directly. These domain-informed features provide a richer representation of cardiac function and improve the discriminative power of the classifier. In order to observe which of those features are indeed useful for the discrimination I plotted the most important features among all (Figure 4).

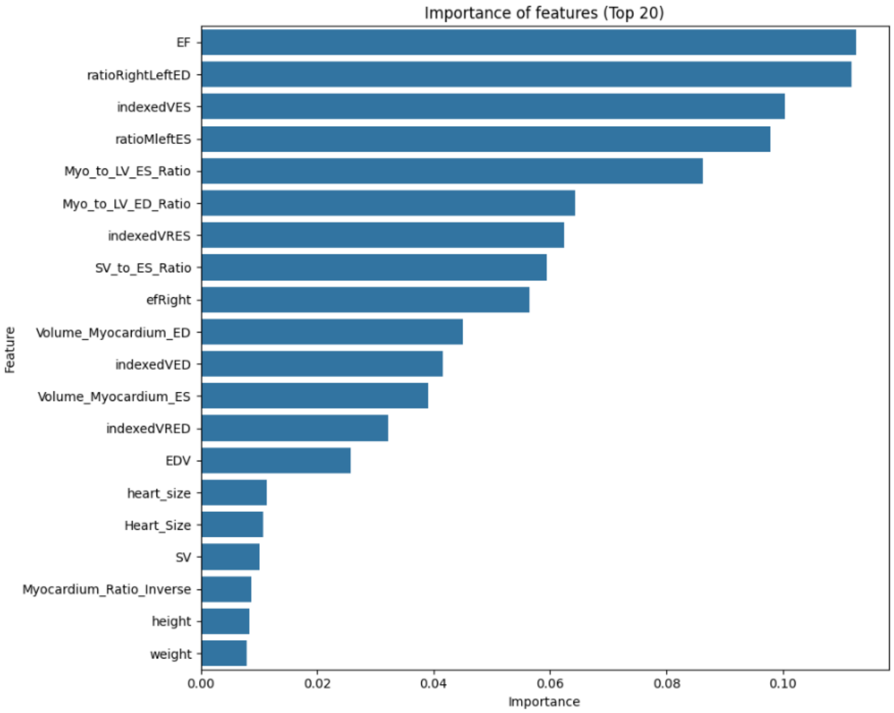


Figure 4: Twenty most important features used in our Random Forest

4 Training

While convolutional neural networks have been proved to be highly effective in medical imaging tasks, I chose not to use them in this challenge for several practical and methodological reasons. Firstly, CNNs typically require large datasets to generalize well and avoid overfitting. With only 100 patients in the train set, training a deep network would probably result in poor performance. Moreover, CNN-based pipelines tend to lack interpretability, which is a crucial factor in medical contexts where understanding model decisions is essential for clinical adoption. I thus aimed for a more interpretable and robust solution that aligns better with the size of the dataset.

4.1 Random Forest

I decided to use a Random Forest classifier as the main model. Feature vectors, as described in the previous part, were constructed for each training patient using the extracted features and the corresponding labels were taken from the provided metadata.

The classifier was trained using scikit-learn with 1000 trees (found thanks to some GridSearch) and 5-fold cross-validation to estimate performance.

The mean cross-validation accuracy and variance were computed to evaluate model stability obtaining very satisfying results around 0.9199 +/- 0.00059 (it depends on our random forest of course). Then the confusion matrix was plotted (Figure 5), showing that our classifier is very efficient in the training phase.

For the classification task, a Random Forest classifier is chosen due to its strong performance on high-dimensional, tabular data and its robustness to noisy or irrelevant features. In our challenge, Random Forest was chosen because of :

- **Non-linearity**: They can model complex, non-linear relationships between the features and the target labels without requiring prior transformations or feature scaling.
- **Feature Importance**: Random Forests provide intrinsic measures of feature importance, which is crucial in a clinical setting to identify which anatomical or functional markers most influence classification.
- **Robustness to Overfitting**: By aggregating over many trees and using techniques like bootstrap aggregation and random feature selection at each split, Random Forests reduce the risk of overfitting, especially when the number of features is large relative to the dataset size, which is the case here.

Once the training is satisfying, I predict the labels for the test data and export it as a .csv file by creating a dataframe.

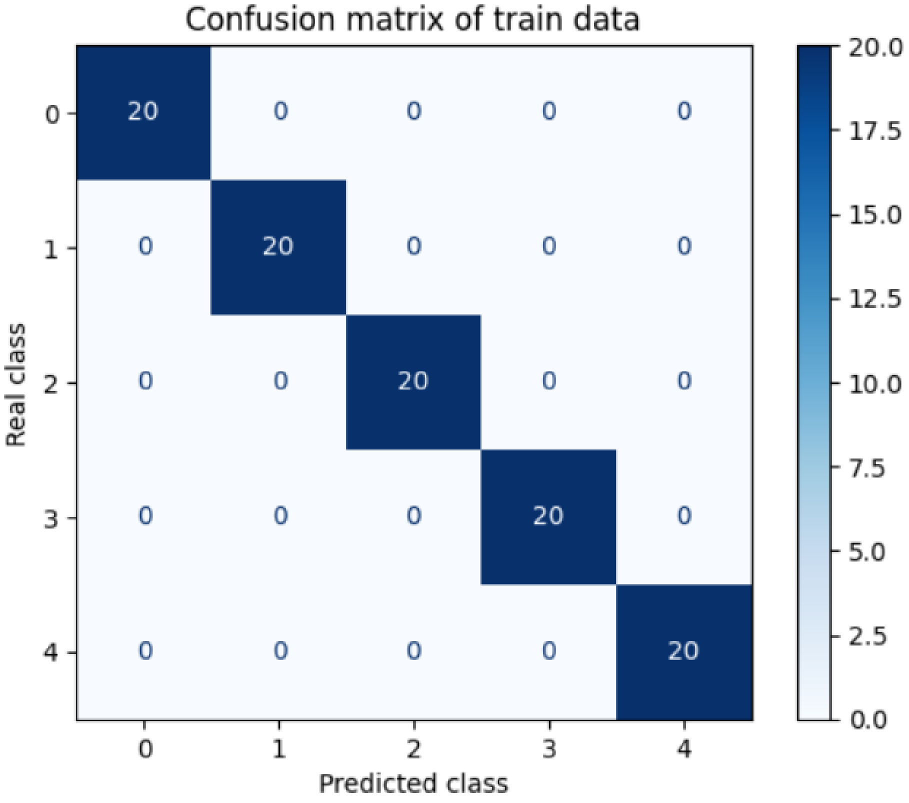


Figure 5: Confusion matrix of our Random Forest Classifier

4.2 AdaBoost

I also decided to try an AdaBoost classifier but it turned out to be inefficient in our problem. Indeed, AdaBoost is a powerful ensemble method, particularly effective when combining many weak learners but it tends to perform poorly in this challenge due to several factors. That can be observed on the confusion matrix plotted as Figure 6.

First, AdaBoost is sensitive to noisy data and class overlap, which is often the case in medical datasets where patients with different cardiac pathologies may exhibit similar feature profiles. Unlike Random Forests, which average predictions across many diverse trees to reduce variance, AdaBoost emphasizes samples that are misclassified in each iteration, potentially overfitting to noise or outliers in a small dataset.

Moreover, with only 100 patients for training, AdaBoost does not benefit from enough data to meaningfully improve performance through iterative reweighting. Finally, since the model relies heavily on decision stumps or shallow trees by default, it may lack the capacity to model the complex, non-linear interactions between anatomical features and cardiac labels. As a result, its classification accuracy and generalization ability are significantly lower than that of Random Forests in this setting.

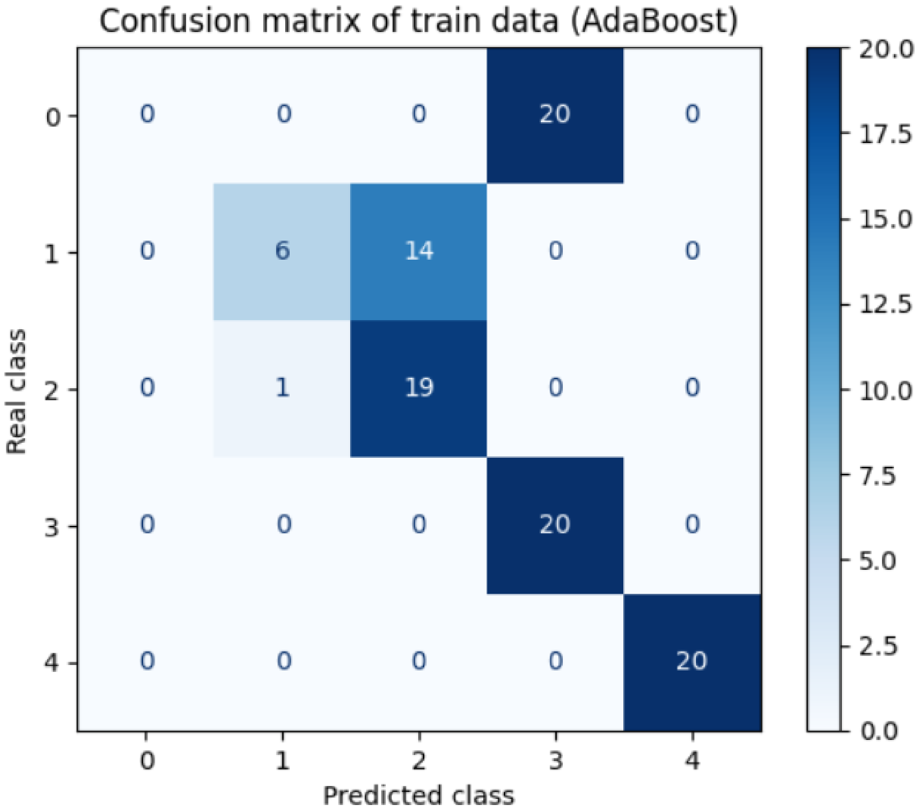


Figure 6: Confusion matrix of our AdaBoost Classifier

5 Limitations

Although the developed pipeline yielded promising results using classical image processing and ensemble learning, several limitations should be acknowledged:

- **Dataset size :** With only 100 labeled patients in the training set, the model may struggle to generalize to new samples. The limited dataset size increases the risk of overfitting, especially when using high-dimensional feature spaces or complex models.

- **Segmentation assumptions :** The segmentation of the left ventricle in the test set was based on a morphological flood-fill method. While this approach is efficient and interpretable, it relies on strong assumptions (e.g., the myocardium fully enclosing the cavity, no leakage through image borders), which may not hold for all cases.

- **No temporal modeling** : Cardiac MRI inherently contains temporal information, capturing the full cardiac cycle. Our model only uses two snapshots (diastole and systole), potentially discarding valuable dynamic information about wall motion or filling patterns that could enhance classification performance.

Addressing these limitations in future work could lead to substantial performance improvements and broader applicability in real-world clinical settings.

6 Conclusion

This project successfully demonstrated a pipeline for segmenting cardiac MRIs, extracting meaningful features, and training a robust classifier without deep learning. The combination of classical image processing and ensemble learning offered a transparent and interpretable alternative suitable for medical applications.

To further improve the robustness and accuracy of the classification model, several avenues could be explored:

- **Deep learning for feature learning :** Instead of manually engineering features, convolutional neural networks (CNNs) or 3D architectures could be trained end-to-end on the MRI volumes or their segmentations, potentially capturing more subtle and complex anatomical patterns.
- **Integration of temporal dynamics :** Incorporating cine MRI data, i.e., multiple frames throughout the cardiac cycle, would allow the model to analyze dynamic cardiac motion. Recurrent or attention-based models could be employed to leverage this temporal information.
- **Advanced preprocessing and harmonization:** MRI images are subject to variability in acquisition parameters and quality. Techniques such as histogram normalization, resampling, or bias field correction may improve consistency across patients and enhance downstream analysis.
- **Explainability and clinical integration:** Employing methods such as SHAP or feature attribution could help identify which features contribute most to each prediction, facilitating clinical validation and acceptance of the model in medical practice.

These directions would require larger datasets and more computational resources but offer a promising path toward more accurate and clinically relevant cardiac pathology classification.

7 Sources

[1] Fabian Isensee1, Paul F. Jaeger, Peter M. Full, Ivo Wolf, Sandy Engelhardt, and Klaus H. Maier-Hein.

Automatic Cardiac Disease Assessment on cine-MRI via Time-Series Segmentation and Domain Specific Features

https://link.springer.com/chapter/10.1007/978-3-319-75541-0$_1$5