Giulia Mannaioli

IMA205 - Apprentissage pour l'image et la reconnaissance d'objets

Introduction

The purpose of this report is to present the results of a project focused on the automatic diagnosis of cardiac pathologies using cardiac magnetic resonance imaging (CMRI). Cardiac pathologies can lead to life-threatening complications, such as heart failure and sudden cardiac arrest. Early detection of these pathologies is essential for effective treatment and prevention of complications.

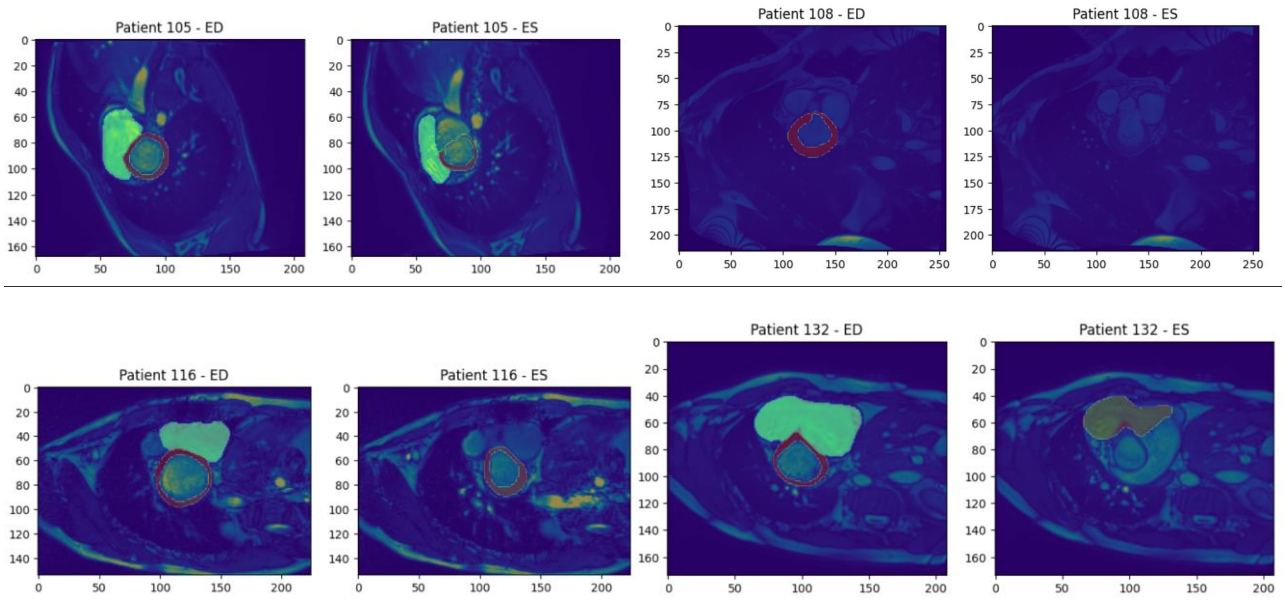
To address this challenge, we used a dataset of 150 subjects and extracted features to classify MRI images into one of five diagnostic classes, including healthy controls and four cardiac pathologies.

The dataset was split into a training-validation set and then the algorithm is used with the given test set.

I tried three algorithms to classify the subjects: Random Forest, Linear SVM and non Linear SVM. I also tried to do some preprocessing such as PCA and I tried to segment the images that presented some issues. Of course I segmented the left ventricle but only using the floodfill function to take the region inside the myocardium.

Exploring Data

The data were balanced, we had 20 patient for each class in the training set. Observing the data I could notice some anomalies in the segmentation of some of the images in the test set.



For this reason, I lost many many time to try to segment these images and you can see the code in the section “segmentation”. These images are only the last ones of each folder, so they are not significant. Anyway, when I tried to undestand better I look at the data frame and these same images had the volume of the missing segmentation were missing. At the end I understood it was not useful because I was considering only the last images in the for and not all the folder. Actually, when I modified the segmented image I did not create a new Nifti1Image objects. For this reason I had these anomalies and I had the values highlighted in the image below.

	Height	Weight	ED_RV	ED_My	ED_LV	ES_RV	ES_My	ES_LV	EF_LV	EF_RV
0	167.0	89.0	28.692818	20.779414	36.067895	21.613821	21.909901	26.566427	0.263433	0.246717
1	137.0	35.0	15.587109	3.949867	5.592105	11.339940	2.760660	4.176382	0.253165	0.272480
2	167.0	116.0	28.417969	15.917969	17.700195	19.116211	21.704102	7.324219	0.586207	0.327320
3	160.0	98.0	26.617529	19.178079	17.084566	18.953774	24.206250	5.009479	0.706783	0.287921
4	174.0	64.0	15.856096	5.790306	7.744004	9.980846	3.411892	5.139074	0.336380	0.370536
5	175.0	107.0	14.184570	25.488281	21.679688	6.323242	28.173828	10.620117	0.510135	0.554217
6	175.0	75.0	15.077448	6.866256	11.991173	10.971852	5.521319	5.110759	0.573790	0.272300
7	185.0	104.0	0.000000	17.943893	14.497162	0.000000	0.000000	0.000000	1.000000	0.000000
8	172.0	104.0	30.487398	9.569697	10.971244	19.290099	12.011101	4.084076	0.627747	0.367276
13	170.0	74.0	28.159248	16.349109	25.713491	21.306723	26.374507	10.003363	0.610968	0.243349
14	157.0	88.0	21.495850	13.921875	19.716064	13.209961	17.659424	7.949707	0.596790	0.385465
15	183.0	75.0	18.907884	7.639549	15.661075	0.000000	6.793742	7.639549	0.512195	1.000000
16	163.0	80.0	32.019542	21.252901	39.552452	28.318509	22.150121	35.963572	0.090737	0.115587
29	170.0	75.0	55.253804	15.832196	18.692085	32.337308	14.224677	8.486207	0.546000	0.414750
30	165.0	92.0	27.907283	19.215464	27.234368	22.112737	24.486632	13.103152	0.518874	0.207636
31	165.0	68.0	34.260493	5.917722	10.009161	19.239673	0.000000	0.000000	1.000000	0.438430
32	172.0	80.0	21.159668	20.210449	19.834717	10.105225	26.419922	5.754639	0.709870	0.522430

After, I added a for loop to save the images as another Nifti1Image objects and solve the problem and the accuracy grew immediately.I think the most challenging part was to treat the images in the correct way, as we never handled this before.

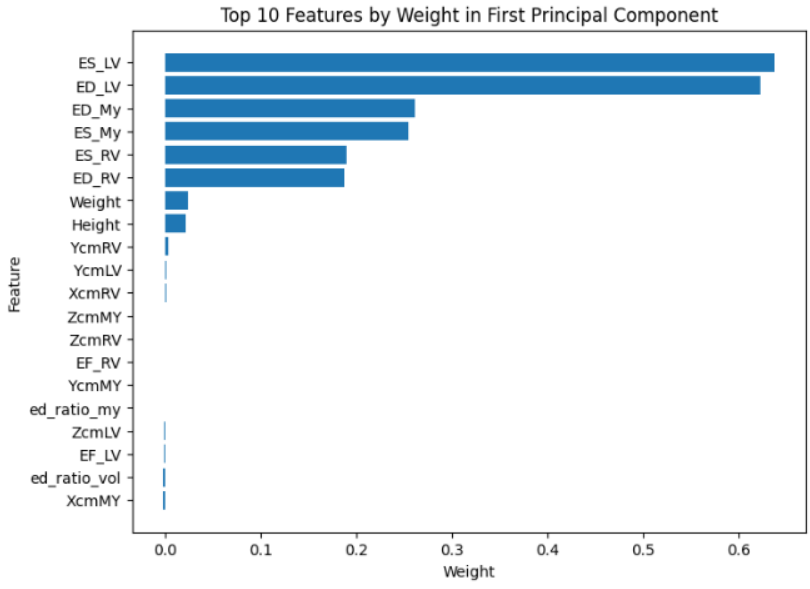
Feature extraction

In total, 21 features were extracted, 2 patient-based and 19 image-based. The patient-based features are height and weight and the image-based features are:

- myocardial, left ventricle and right ventricle volumes for both the end of the cardiac contraction (ES) and end of the dilation (ED) (6 features)
-the ejection fractions (EF) of right and left ventricle (2)
-ratio between right and left ventricle volume at ED and ES (1)
-ratio between myocardial and left ventricle volume at ED and ES (1)
-the difference between the centre of mass at ED and ES of the myocardium, right and left ventricle (we have x, y and z so we have 9 features in total but representing just 3 points in the image)
I decided to select these features considering the article [2] and I added the centre of mass to improve the result.

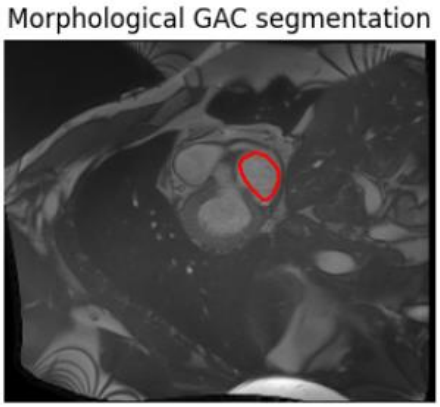
In the image below we can observe the importance of each feature and we can see that the first 12 are the most important form the code. The number of features using PCA with 99% of data explained is 12. Observing the graph we can notice that the volumes are the most important features, the others are just combiantion of them. Anyway, some of them are essential to improve the results. For example, to compute the mass center improved enough the result.

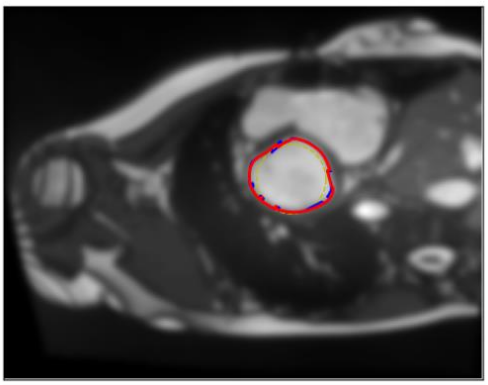
	feature	weight
7	ES_LV	0.638254
4	ED_LV	0.622368
3	ED_My	0.261269
6	ES_My	0.254214
5	ES_RV	0.189955
2	ED_RV	0.187649
1	Weight	0.023595
0	Height	0.021174
18	YcmRV	0.002445
15	YcmLV	0.000104
17	XcmRV	0.000095
22	ZcmMY	-0.000040
19	ZcmRV	-0.000231
9	EF_RV	-0.000434
21	YcmMY	-0.000640
12	ed_ratio_my	-0.000971
16	ZcmLV	-0.001136
8	EF_LV	-0.001265
10	ed_ratio_vol	-0.002582
20	XcmMY	-0.002945



Segmentation

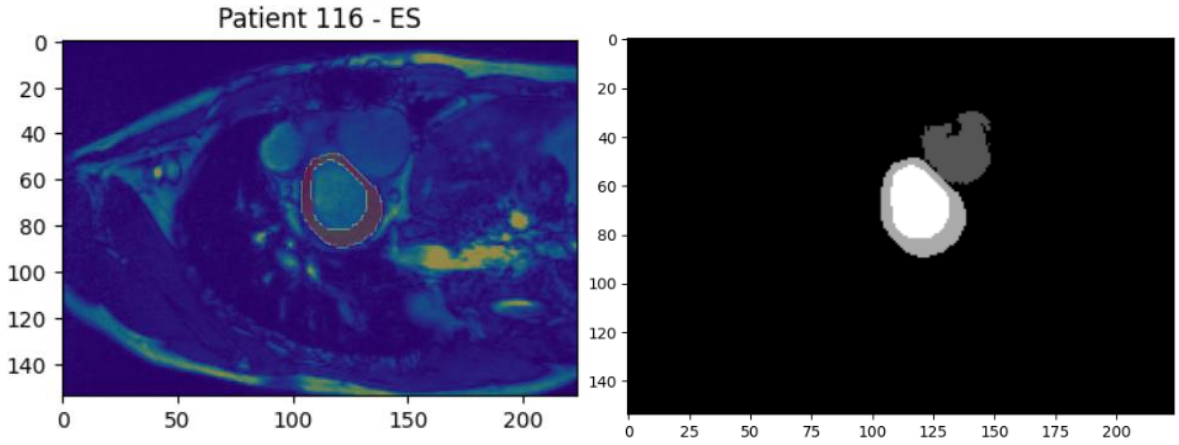
As I said before, for segmentation, I did not use a real segmentation technique but for the images that looked with anomalies I tried using active contours, as suggested in the paper [1]. Anyway, snakes and active contours face difficulties working on images with low contrast and may not be able to flag important features such as wall thinning. Since there was not much contrast in the images and we had to center manually every time, it was not easy given that we had many images. These are some results:

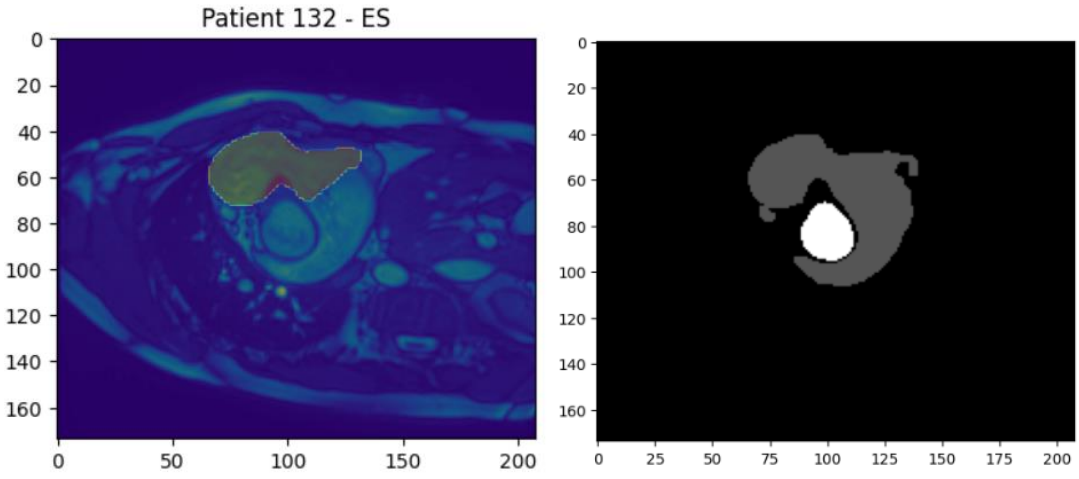


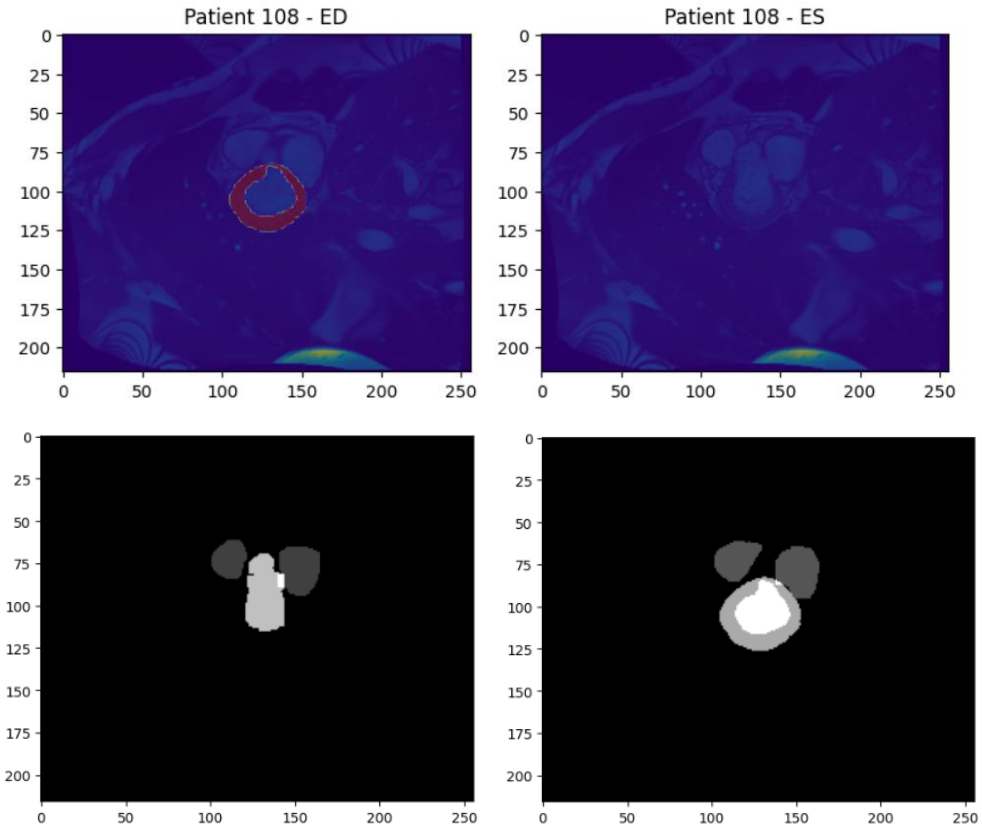


Then I tried to use morphological operation such as opening and closing to process the images mixed with thresholds and different functions to pick the region and to select a region as mask to not consider the other parts. All the segmentation I obtained where pretty good, but at the end it was not important.

Below some attempts:







In the final code, I did not use a real segmentation technique because I observed that the myocardium was already segmented in most cases, so to segment the left ventricle, we just had to take the zone inside the myocardium. Therefore, I used the Floodfill function of cv2, and it worked pretty well.

Classification

To classify, of course, I performed the same operation of the train set for the test set. I used the segmented images as discussed earlier. I standardized the features because some values (for the center of mass) were less than 0 and also because standardizing the features is particularly important when using distance-based algorithms (such as SVM), as variables on different scales can have different effects on the algorithm's performance. Standardization ensures that all variables have a mean of 0 and a standard deviation of 1, making them comparable.

Then, I shuffled the data to ensure that the model doesn't learn the order of the examples.

After, I used the Random Forest algorithm because it was the most useful according to the research articles.

I trained a Random Forest model using GridSearchCV to search for the best hyperparameters using the training set.

The trained model was then used to perform 10-fold cross-validation on the training set.

At last I tried to use the Support vector machine and I tried the linear and the non-linear, and I also tried to perform the PCA before them. PCA can be useful for reducing the complexity of the model and avoiding overfitting. In this case, I tried using PCA before applying SVM to see if it could improve the model's performance but it was not the case.

Results

Taking into account the best results (because sometimes they changed even if I did not change the code) we obtained:

- SVM non linear gave me the best result such as 0.85714
- SVM linear 0.82857
- PCA + SVM linear 0.85714
- Random forest 0.8

SVM non-linear achieved the highest accuracy score of 0.85714, which suggests that the decision boundary between the two classes might not be linearly separable, and the non-linear SVM was able to capture this non-linear relationship between the features and the target variable.

On the other hand, PCA+SVM linear also achieved an accuracy score of 0.85714, which suggests that the principal components extracted by PCA were able to capture most of the variability in the data, and the linear SVM was able to find a linear decision boundary in the reduced feature space.

In contrast, the random forest algorithm did not perform as well, with an accuracy score of 0.8. One possible reason for this could be that the random forest was not able to capture the non-linear relationships between the features and the target variable or maybe I did not extract perfectly the features, given that it was the most useful according to the articles gave me the worst result.

[1] *Automatic segmentation of the left ventricle cavity and myocardium in MRI data, M. Lynch, O. Ghita, P.F. Whelan. Vision Systems Group, School of Electronic Engineering, Dublin City University, Dublin 9, Ireland. Received 20 October 2004, Accepted 31 January 2005, Available online 31 May 2005.*

[2] *Wolterink, J.M., Leiner, T., Viergever, M.A., Išgum, I. (2018). Automatic Segmentation and Disease Classification Using Cardiac Cine MR Images. In: , et al. Statistical Atlases and Computational Models of the Heart. ACDC and MMWHS Challenges. STACOM 2017. Lecture Notes in Computer Science(), vol 10663. Springer, Cham.*