Guy's and St Thomas' NHS
NHS Foundation Trust

KING'S College LONDON

How to play with Statistical Shape Models

Pablo Lamata
Eric Kerfoot
Esther Puyol

School of Biomedical Engineering & Imaging Sciences
King's College London

Guy's and St Thomas' NHS
NHS Foundation Trust

KING'S College LONDON

Documentation:

https://www.dropbox.com/s/8f072f2ws2p9utj/Atlas%20Documentation.doc?dl=0

Overview

1. Get the code
2. Get the data
3. Define the features: build the SSM (unsupervised learning)
   Visualize and analyze results
4. Machine learning (supervised learning)
   Visualize and analyze results
5. Export for further analysis

1. Get the code

Step 1: get the code

Github: CompAnat.zip

At home:

1.2 How to get the code

The code is hosted in bitbucket.

1. Create an account if you do not already have one, https://bitbucket.org
2. Ask Pablo Lamata access to https://bitbucket.org/plamata/computationalcardiacanatomy
3. Install a mercurial client in your machine.
    a. Suggested options are:
        i. In windows, Tortoise mercurial, https://tortoisehg.bitbucket.io
        ii. In Mac, Tortoise as well, https://bitbucket.org/tortoisehg/files/downloads
        iii. SourceTree is another software that seems to work quite well too
    b. Once installed, you need to set-up some basic configuration details, your username and e-mail (see full instructions in https://confluence.atlassian.com/bitbucket/set-up-mercurial-726371757.html)
        i. If you have windows:
            1. Open the Mercurial configuration file .hgrc using your favourite editor.
            2. Add a username value to the configuration.
            3. When you are done, the .hgrc file includes the following lines with your own username and email address:

                1 [ui]

                2 # Name data to appear in commits

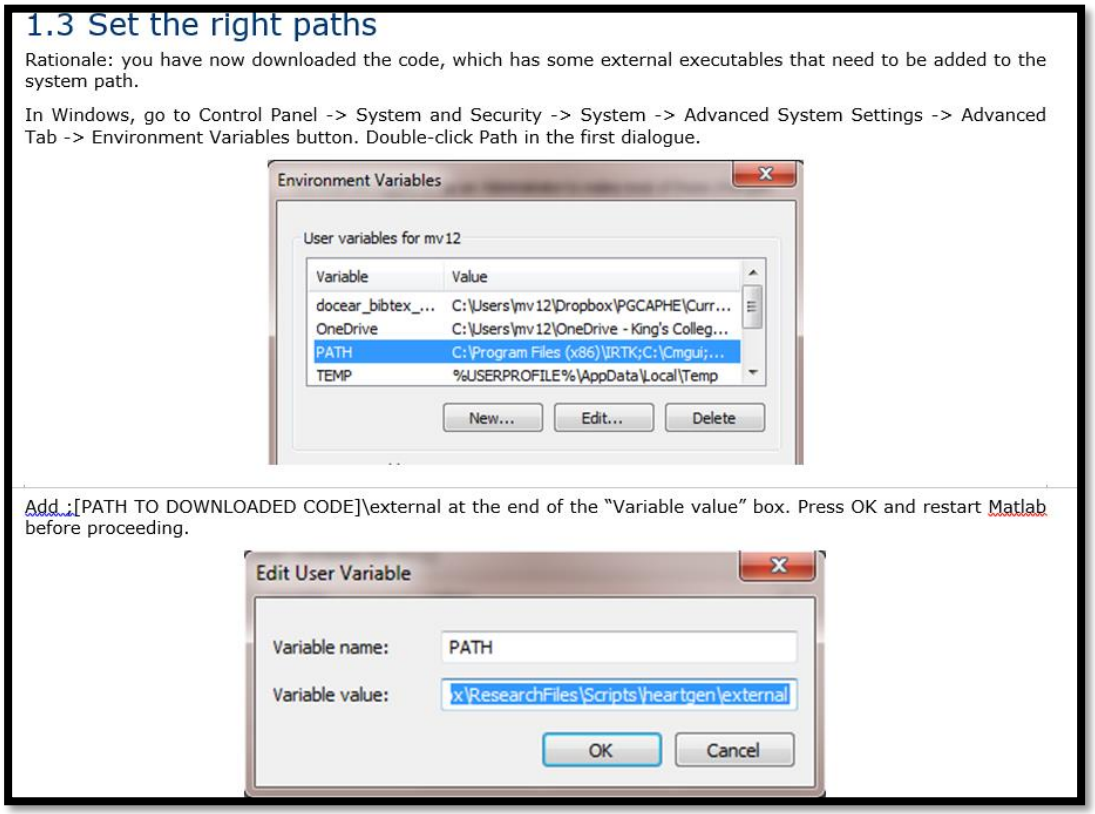
                3 username = Emma Paris <eparis@atlassian.com>

            4. Save and close the file.
        ii. If you have Mac: see steps 9 to 11 in https://confluence.atlassian.com/bitbucket/set-up-git-and-mercurial-mac-os-x-269981802.html
4. Create a folder in your machine where you want to store the local copy of the code.
5. Clone the remote repository into this folder by:

Step 1b: set the paths to executables

For testing, run:

```
>> CallcmGui
```

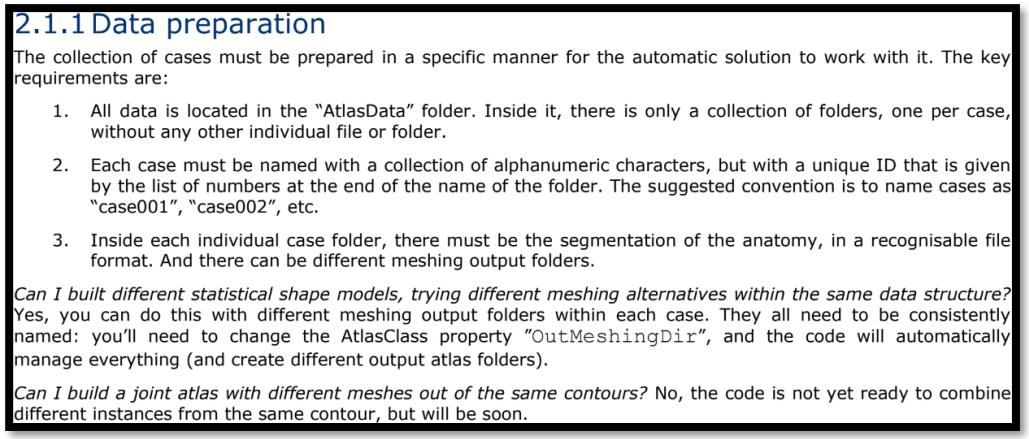


2. Get the data

Step 2: get and prepare the data

Input: segmentations

Githug: cases.zip



2.1.1 Data preparation

The collection of cases must be prepared in a specific manner for the automatic solution to work with it. The key requirements are:

1. All data is located in the "AtlasData" folder. Inside it, there is only a collection of folders, one per case, without any other individual file or folder.
2. Each case must be named with a collection of alphanumeric characters, but with a unique ID that is given by the list of numbers at the end of the name of the folder. The suggested convention is to name cases as "case001", "case002", etc.
3. Inside each individual case folder, there must be the segmentation of the anatomy, in a recognisable file format. And there can be different meshing output folders.

Can I built different statistical shape models, trying different meshing alternatives within the same data structure? Yes, you can do this with different meshing output folders within each case. They all need to be consistently named: you'll need to change the AtlasClass property "`OutMeshingDir`", and the code will automatically manage everything (and create different output atlas folders).

Can I build a joint atlas with different meshes out of the same contours? No, the code is not yet ready to combine different instances from the same contour, but will be soon.

How

Adapt "PrepareData.m"

Check with ITK-Snap,
http://www.itksnap.org/pmwiki/pmwiki.php

3. Define features

Unsupervised learning

Define the features

Dummy Atlas Script

- Set paths - AtlasRoot
- Choose LoD – “level of detail”
- Choose topology
  - nE – “number Elements”

Compt Burden: 1 hour

- 1 minute per mesh
- 5 minutes coefs and visualizat

6 Dummy atlas script

```
% Control flags of the three time consuming tasks:
bBuildMeshes = 1;
bBuildPCA = 0;
bComputeCoefs = 0;

% First instance of the class:
Atlas = AtlasClass(AtlasRoot);

opt.LoD = LoD;
% It is better to directly set this member variable than introducing it as an
option, in case funtions are not called sequentially (i.e. SaveMeshes not
after building the meshes, and then not knowing what the OutMeshingDir is)
Atlas.OutMeshingDir = sprintf('MeshingHG%i',opt.LoD);

if(bBuildMeshes)
    Atlas = Atlas.BuildMeshes(opt);
    % Move to the 'AtlasMeshData' folder:
    Atlas = Atlas.SaveMeshes_sortedbymeshID();
end

if(bBuildPCA)
% Define the template, will be stored in the PCAaxis member variable at the
end of the PCA construction
    TemplateOptions.nE = [6 12 1];
    TemplateOptions.topology = 'LV';
    Atlas = Atlas.SetTemplate(TemplateOptions);
end

PCAoptions.iShapeSpace = ShapeSpace;
        if(bBuildPCA)
            Atlas = Atlas.BuildPCA(PCAoptions);
        else
            Atlas = Atlas.LoadPCAaxis(PCAoptions);
        end
        if(bComputeCoefs)
            Atlas = Atlas.CalculatePCAcoefs();
        else
            Atlas = Atlas.LoadPCAcoefs();
        end
```

Choose topology

nE: number Elements

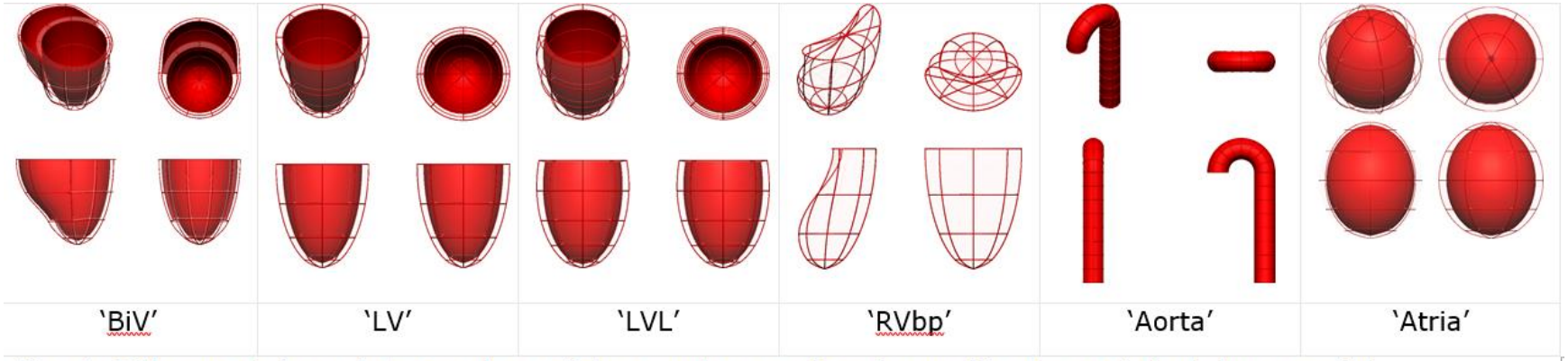


Fig. 4: Different choices of the topology of the template mesh to be used in the statistical shape model.

Choose shape space

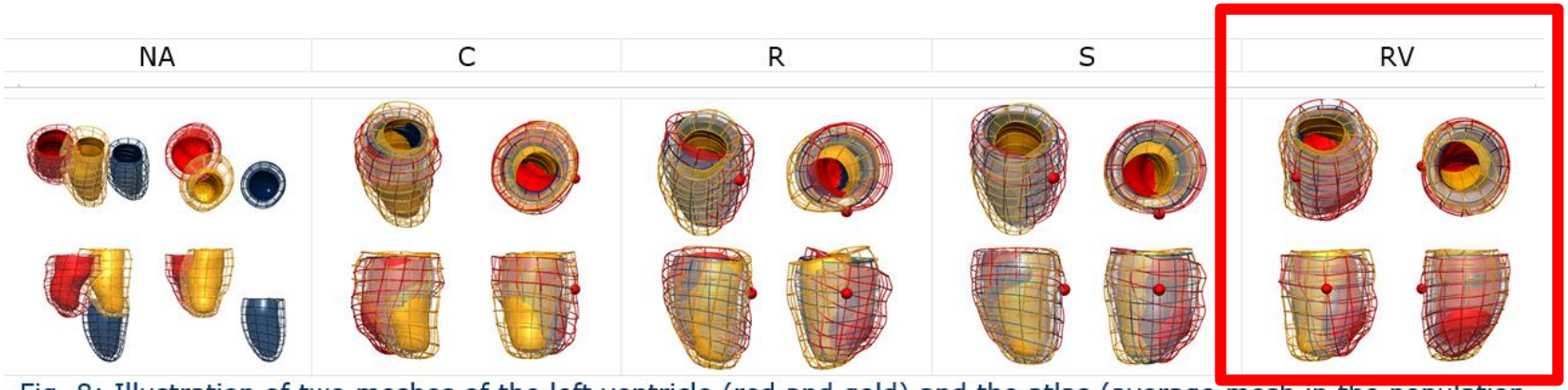
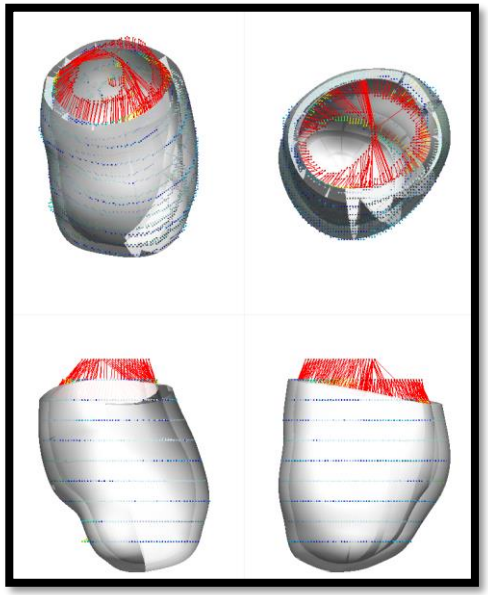


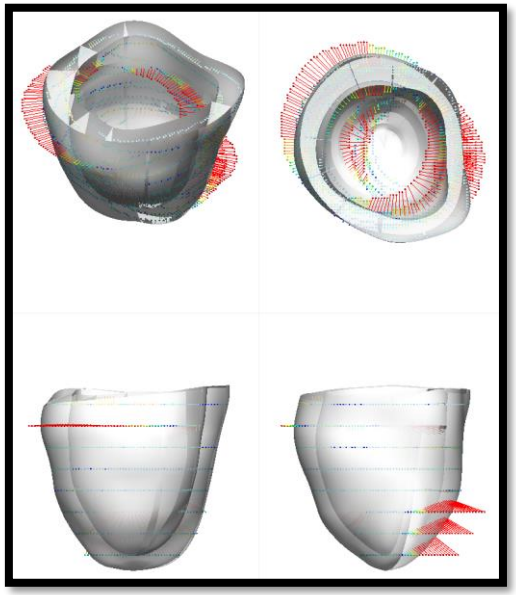
Fig. 8: Illustration of two meshes of the left ventricle (red and gold) and the atlas (average mesh in the population, dark blue) pre-aligned as defined in each of the five shape spaces.

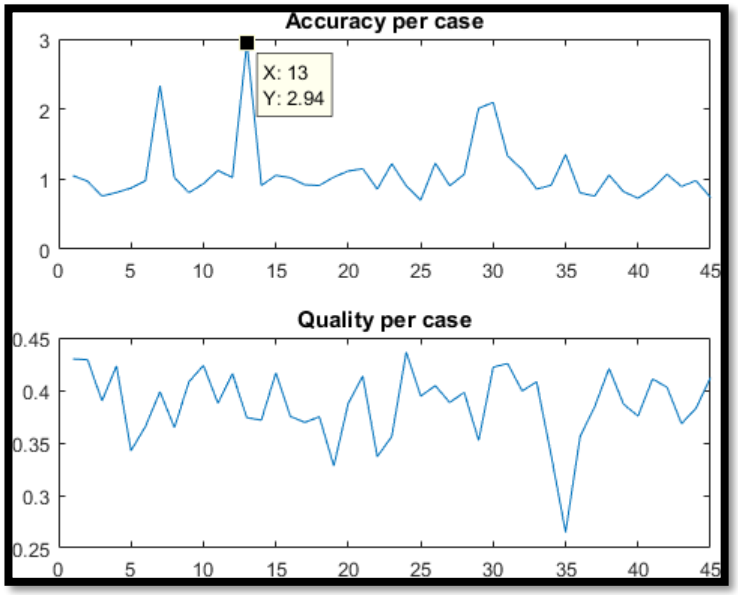
Sanity check 1: fitting accuracy

View mesh:

>> Atlas.ViewMeshIDFittingAccuracy(1320638)





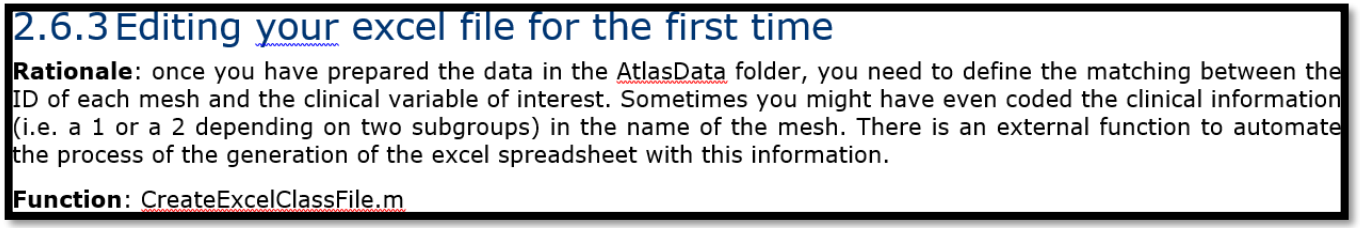


Compare the cathegories

Atlas = Atlas.SetClass(fullfile(Atlas.RootDir,'infocases.xls'),4);

Define them:

- Mesh ID
- Case ID
- Phase ID
- Class: **HYP = 1; HF=2; N=3;**



2.6.3 Editing your excel file for the first time

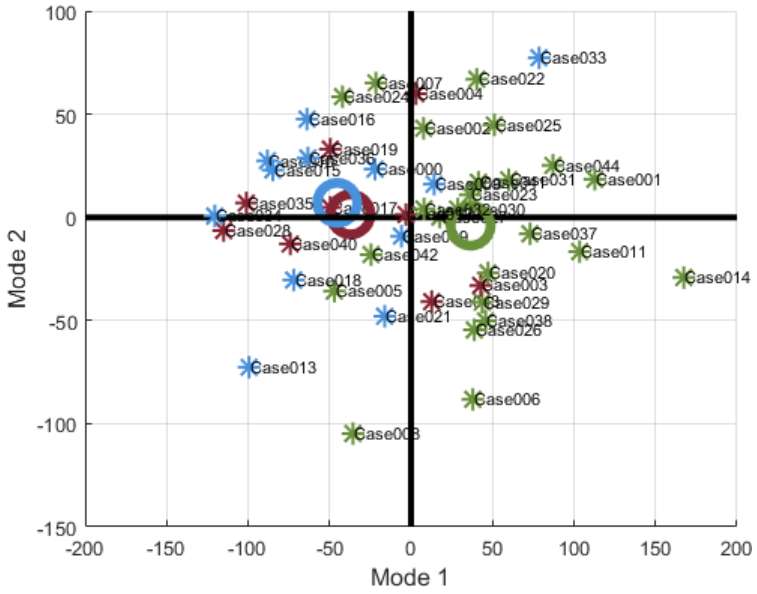
Rationale: once you have prepared the data in the AtlasData folder, you need to define the matching between the ID of each mesh and the clinical variable of interest. Sometimes you might have even coded the clinical information (i.e. a 1 or a 2 depending on two subgroups) in the name of the mesh. There is an external function to automate the process of the generation of the excel spreadsheet with this information.

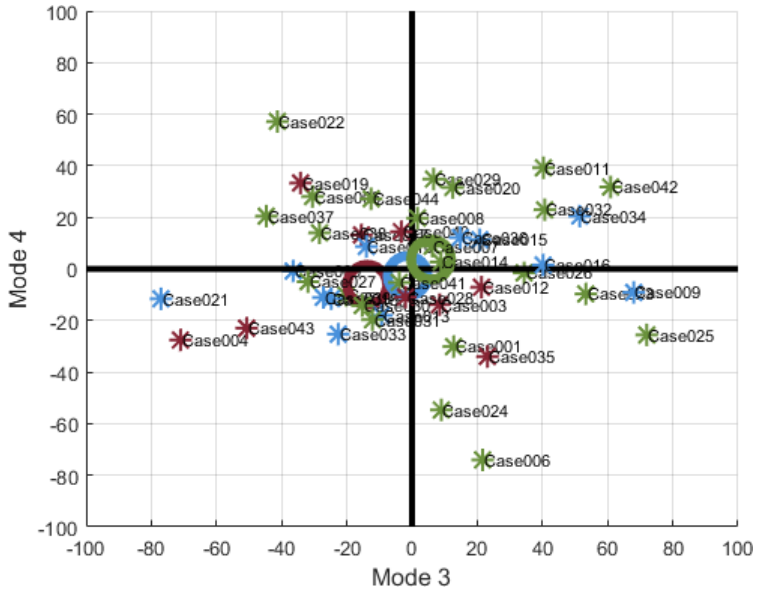
Function: CreateExcelClassFile.m

Sanity check 2: outliers in shape space

HYP = 1; HF=2; N=3;

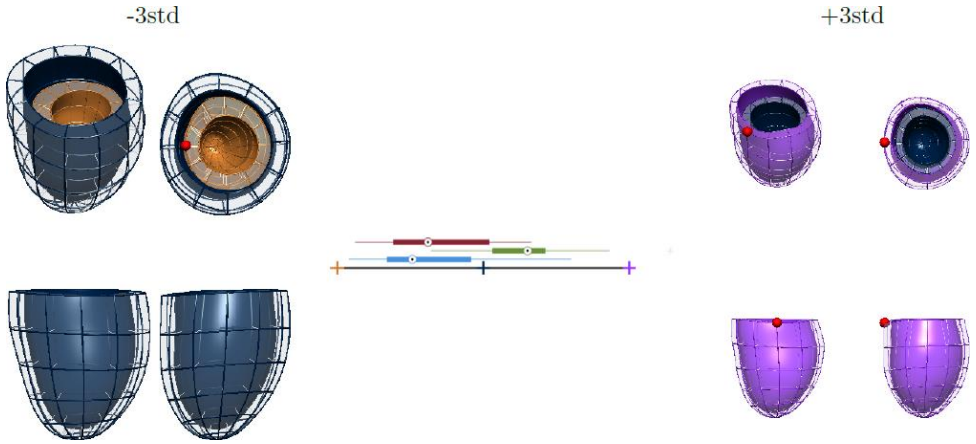
>> Atlas.ViewEigenSpace





Meaning: view the modes

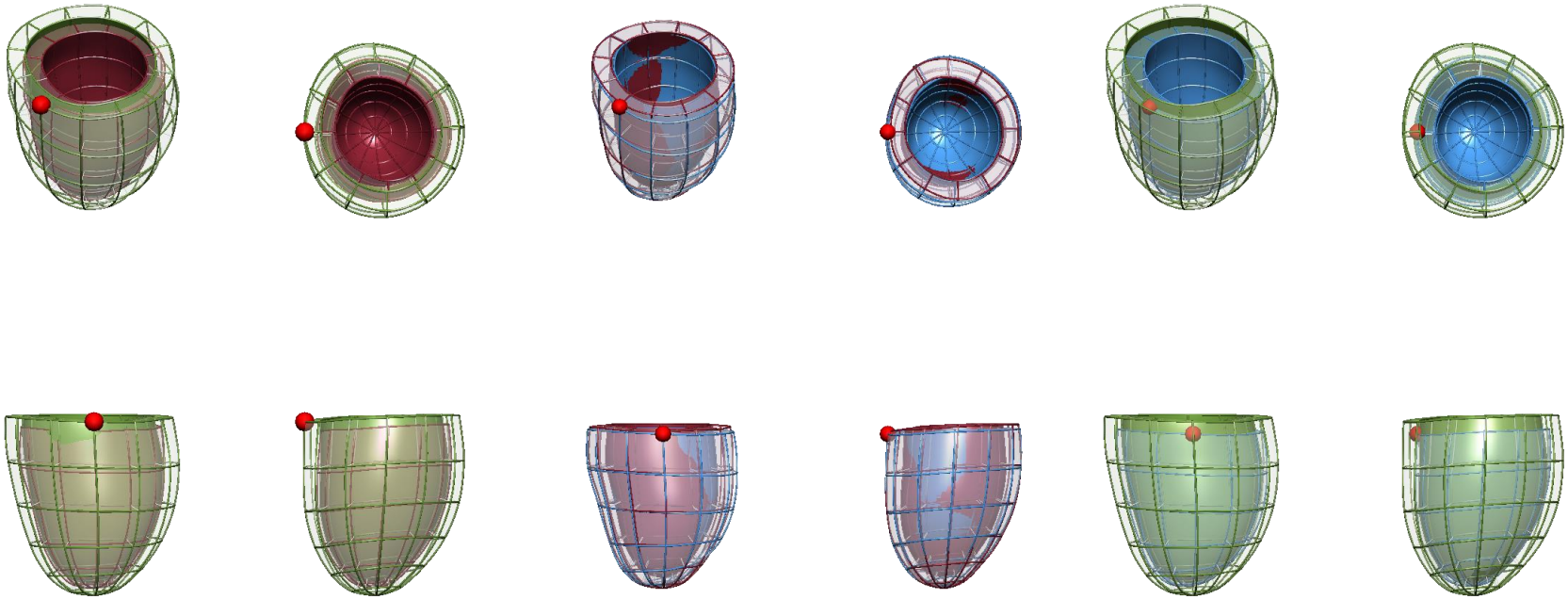
Compile file: Atlas\AtlasOutputrv\ModesAtlasS4LoD2\AtlasModesAtlasS4LoD2_s4.tex



How to use latex: if you are not familiar with latex, there are many tutorials online. If you use windows, my suggestion is that you install texworks (http://www.texworks.org/). If you do not want to bother, you can always get the collection of images in the output folder and compose the image as you want in a powerpoint for example.

Average shapes

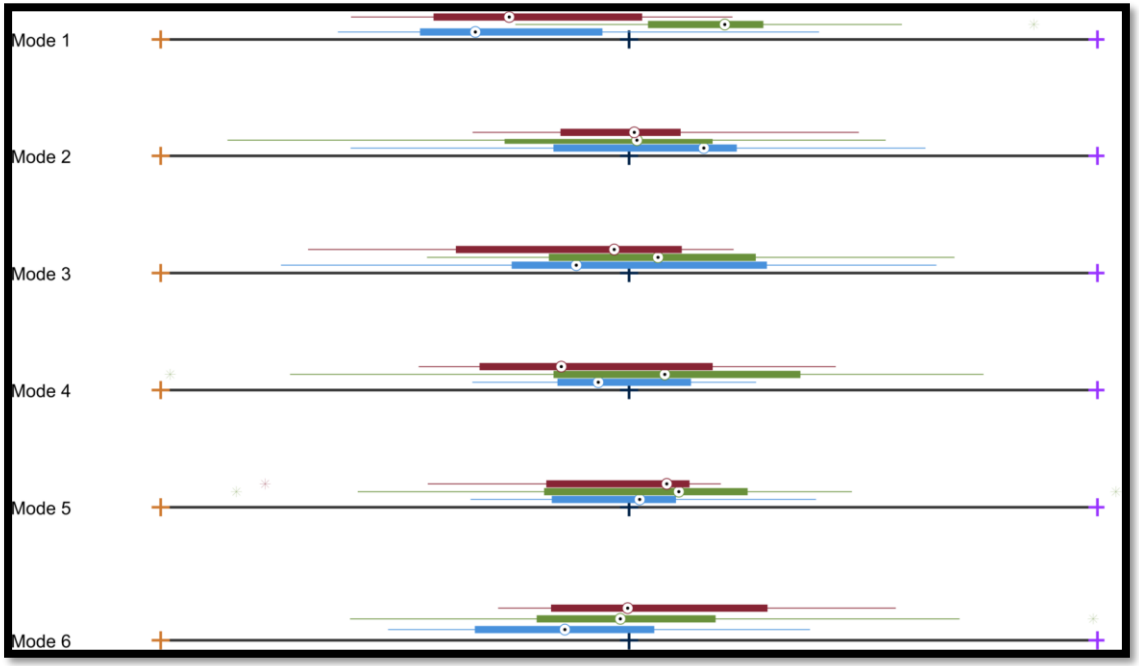
HYP = 1; HF=2; N=3;



View distributions per axis

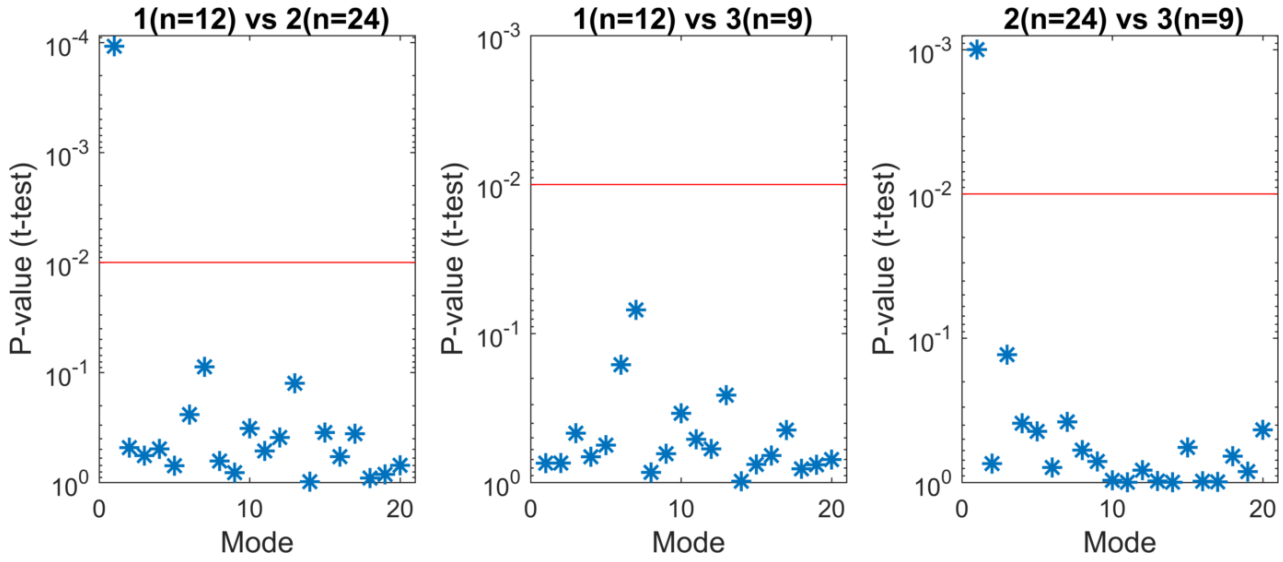
>> Atlas.ViewBoxPlots

HYP = 1; HF=2; N=3;



Basic T-test per coordinate

>> Atlas.CompareClassesByPCAaxis



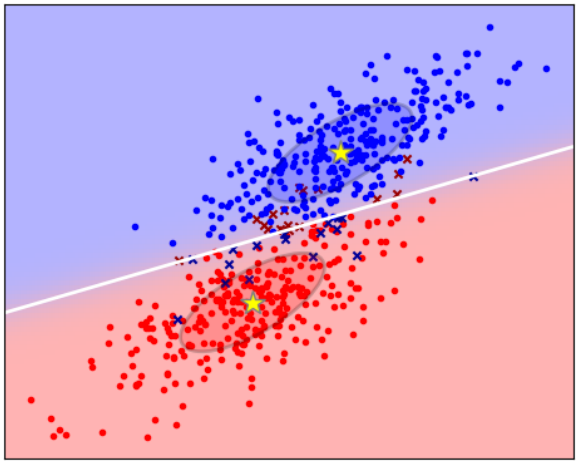
4. Linear Discriminant Analysis

Supervised learning

Linear Discriminant Analysis

Illustration of risks for **overfitting** and lack of **interpretability**, just working with a very simple machine learning tool!!!

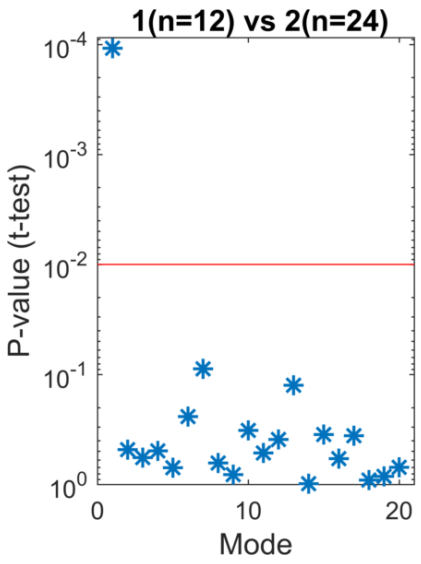
Only by pairs of groups!

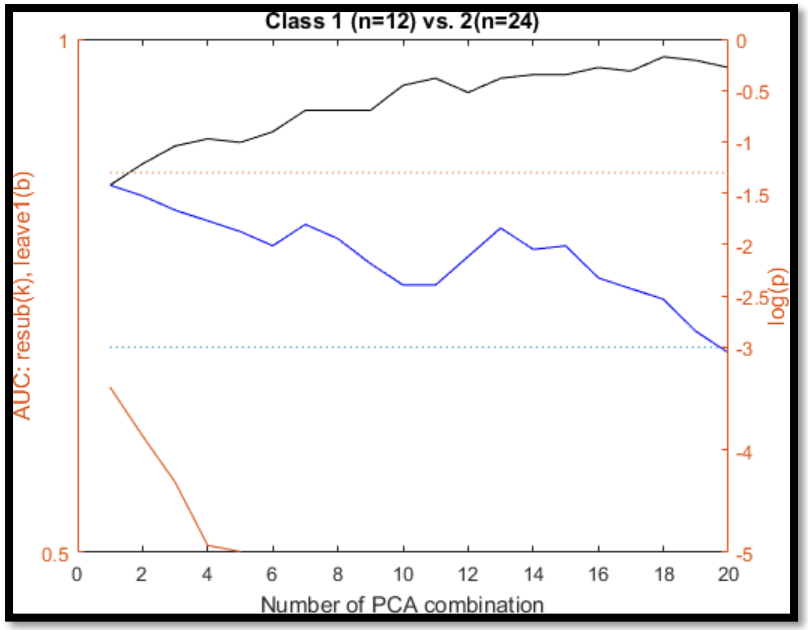


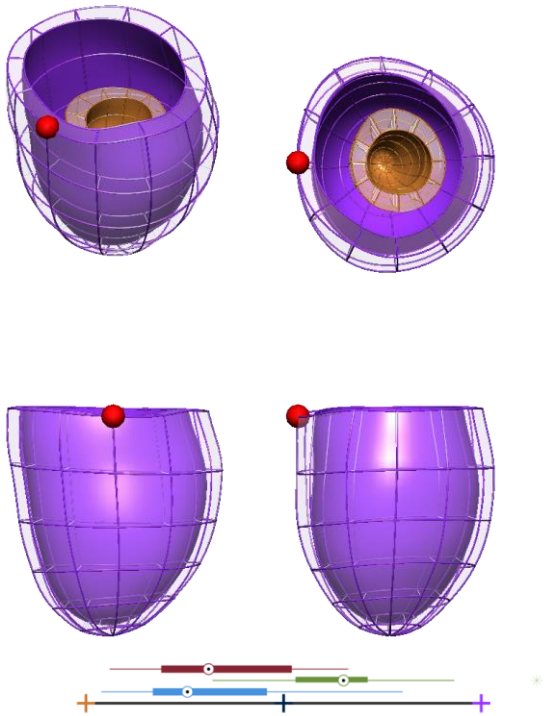
Step 4: HYP to HF (1 vs 2)

>> Atlas.ComprehensiveLDA(20);

HYP = 1; HF=2; N=3;





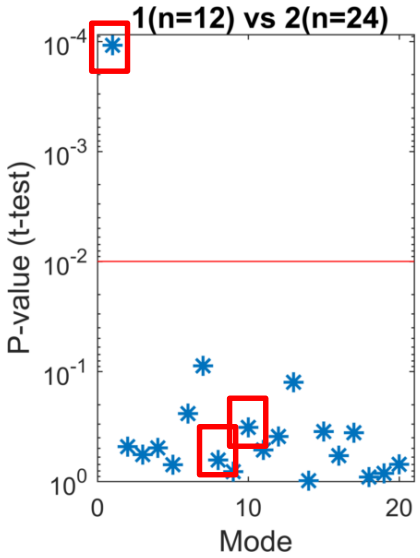


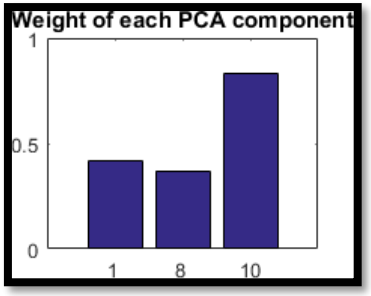
Anything more than that "size"?

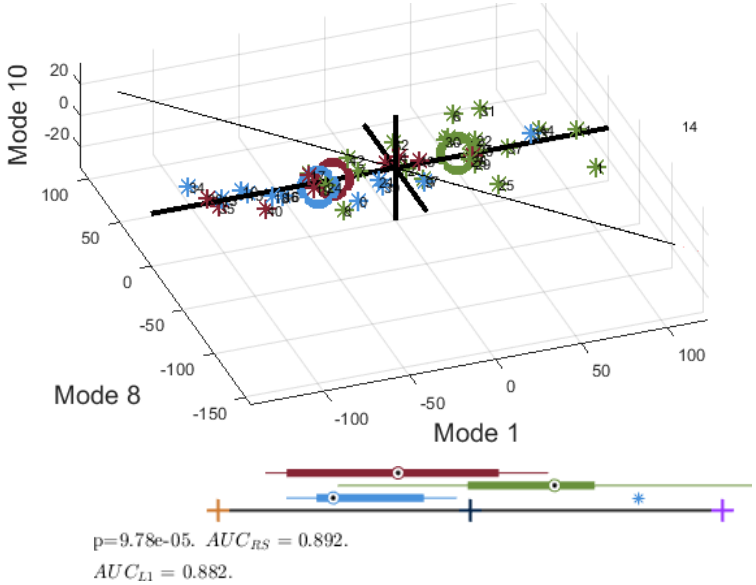
HYP = 1; HF=2; N=3;

>> optLDA.CoreModes = 1;

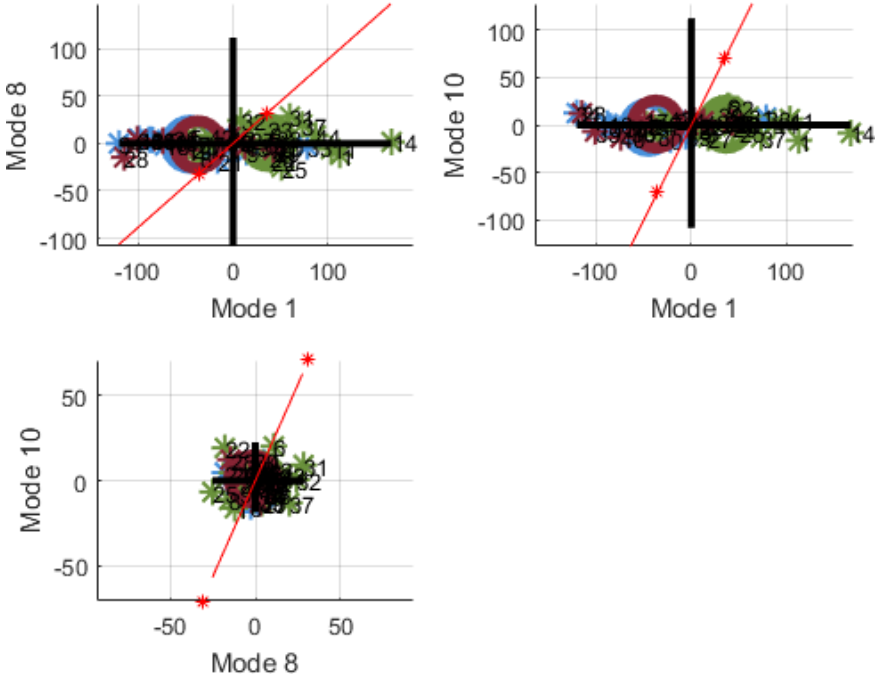
>> Atlas.ComprehensiveLDA([2:10],optLDA);



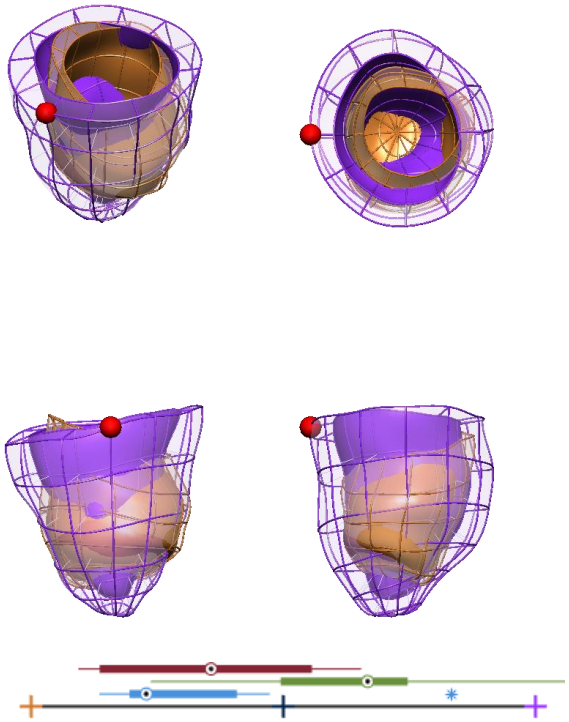




Real extreme shape?



HYP = 1; HF=2; N=3;



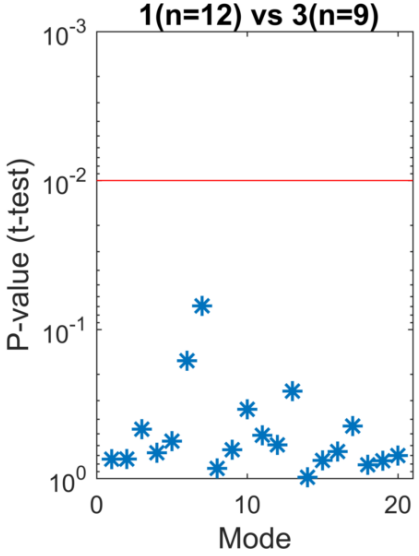
p=9.78e-05. $AUC_{RS} = 0.892$.
$AUC_{L1} = 0.882$.

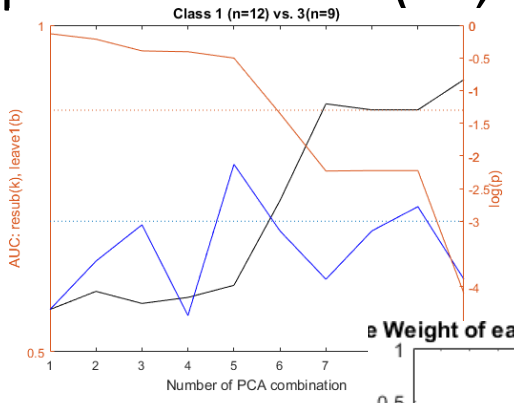
Step 4b: classes 1 to 3 (HYP to N)

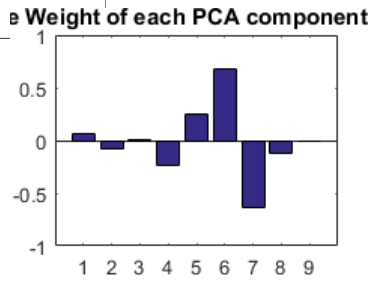
HYP = 1; HF=2; N=3;

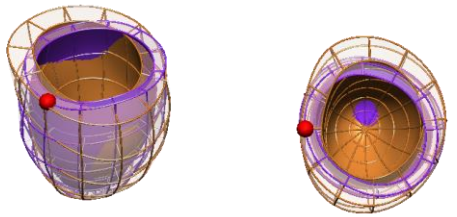
>> optLDA.classes2include = [1 3];

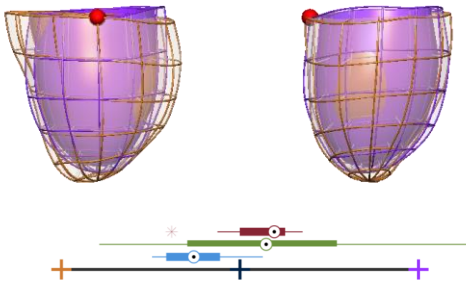
>> Atlas.ComprehensiveLDA(10,optLDA);

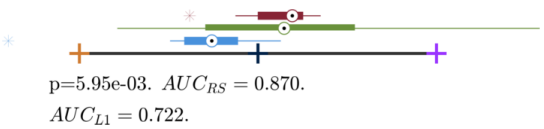








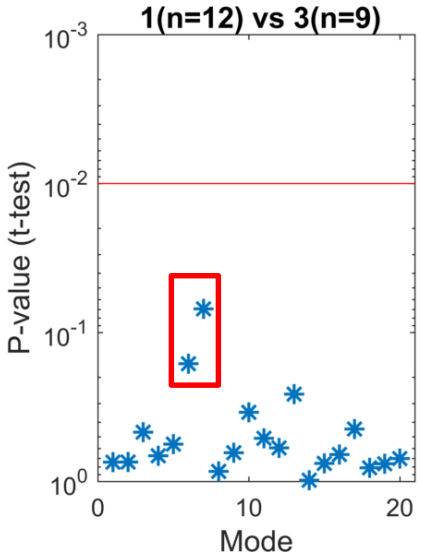


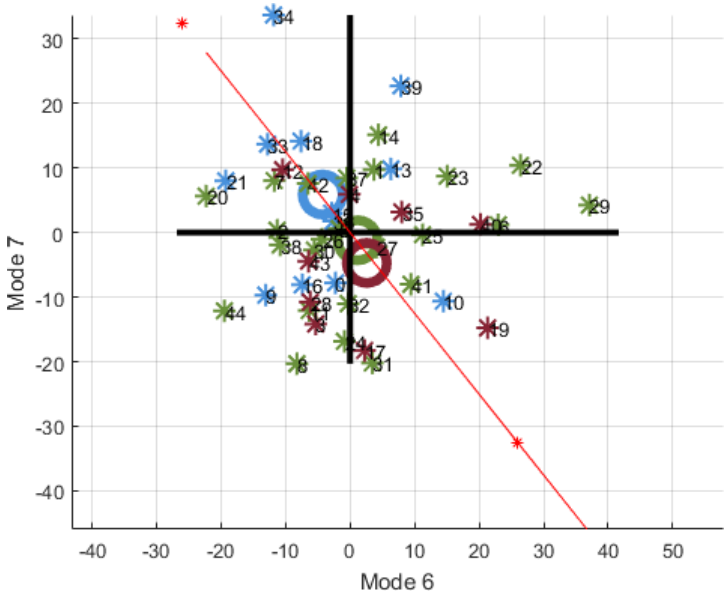


p=5.95e-03. $AUC_{RS} = 0.870$.
$AUC_{L1} = 0.722$.

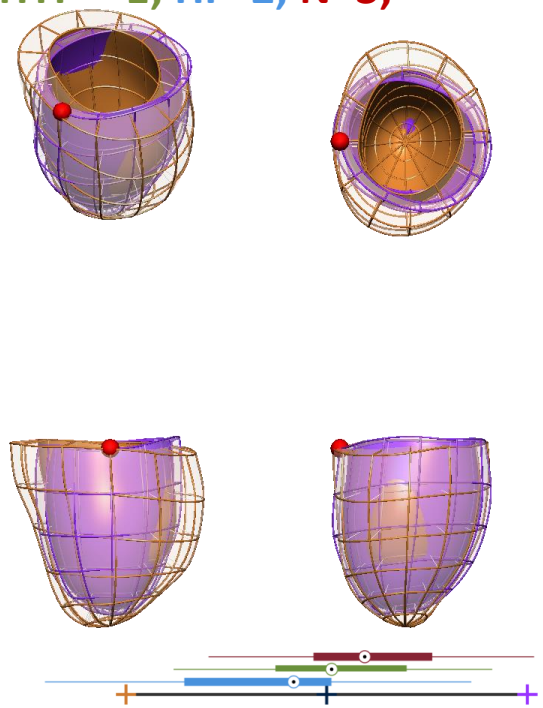
Only the discriminative features

Atlas.ComprehensiveLDA([6:7], optLDA);



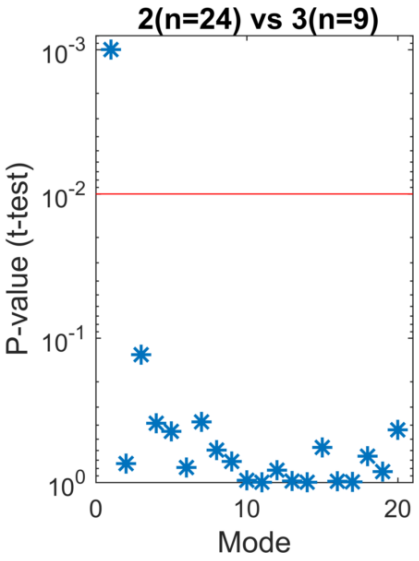


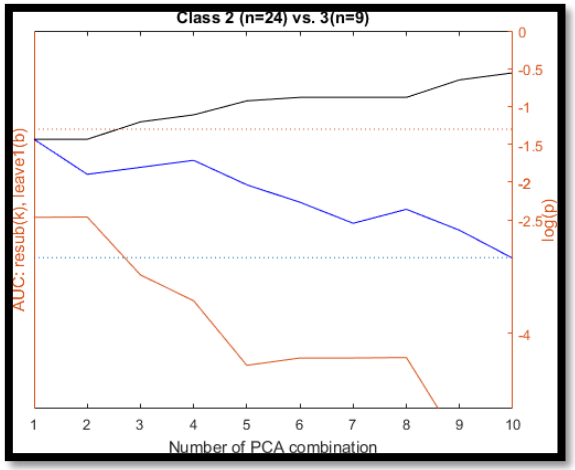
HYP = 1; HF=2; N=3;



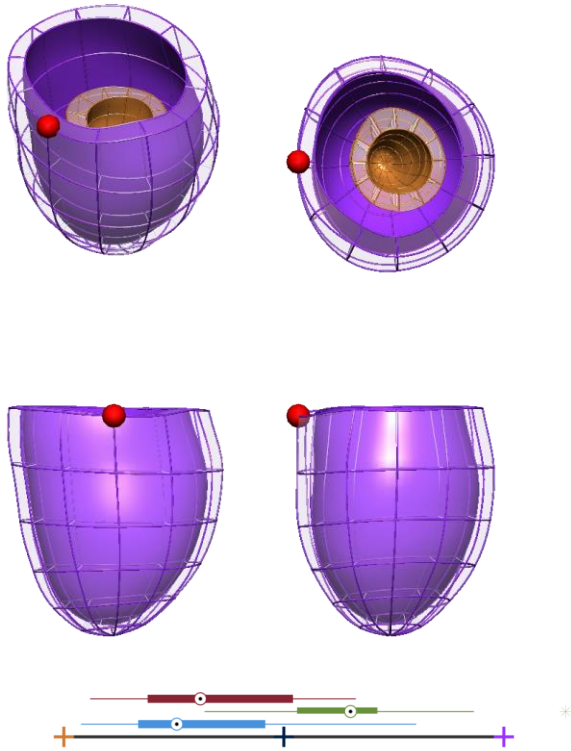
p=3.28e-02. $AUC_{RS} = 0.769$.
$AUC_{L1} = 0.722$.

Step 4c: classes 2 to 3 (HF to N)



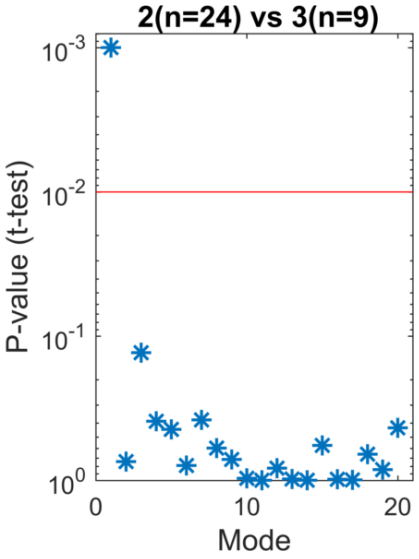


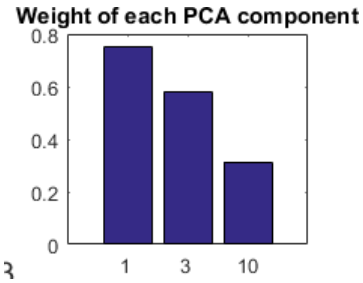
HYP = 1; HF=2; N=3;



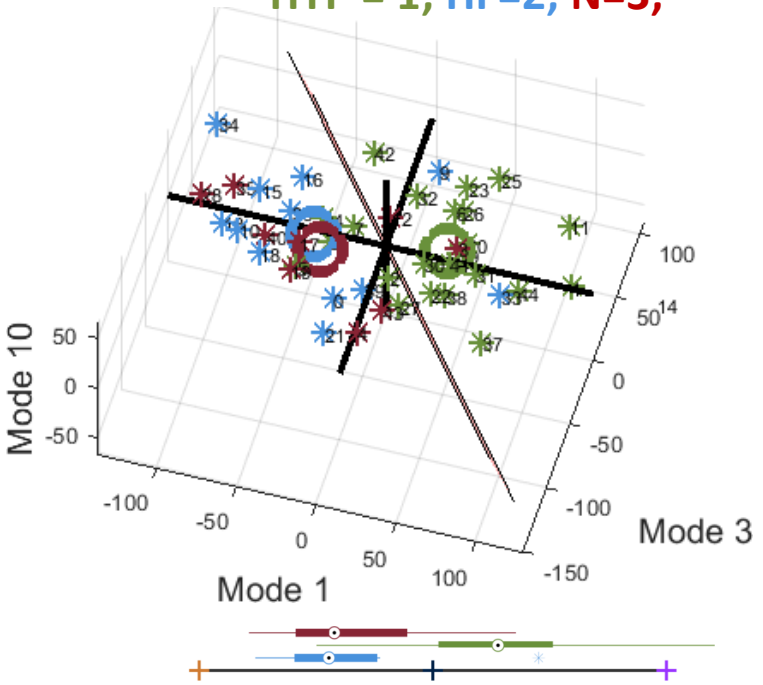
p=3.41e-03. $AUC_{RS} = 0.856$.
$AUC_{L1} = 0.856$.

Anything more than “size”?





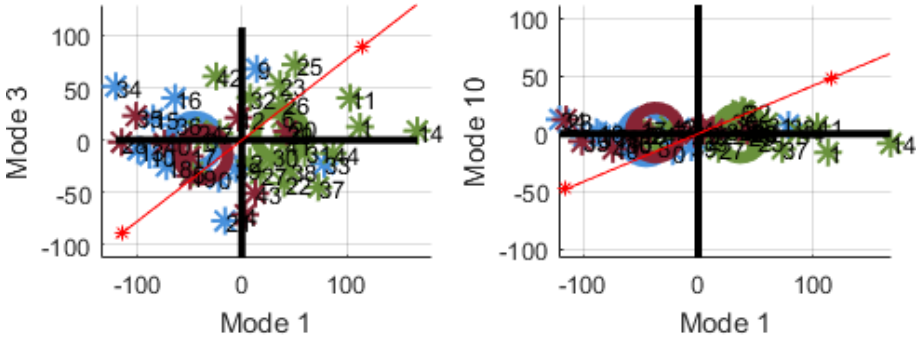
HYP = 1; HF=2; N=3;

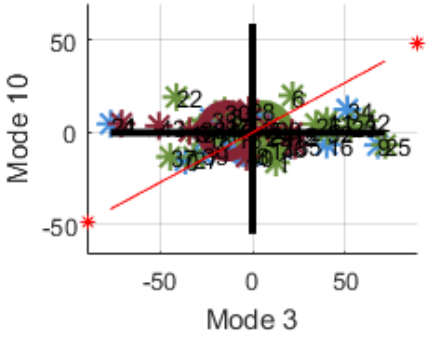


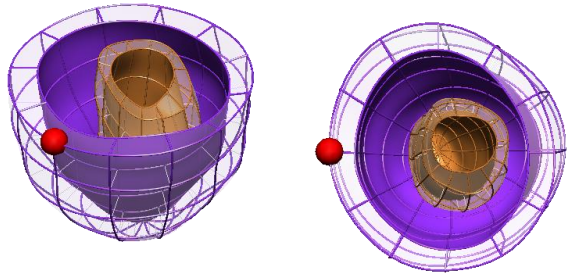
p=6.15e-04. $AUC_{RS} = 0.884$.
$AUC_{L1} = 0.861$.

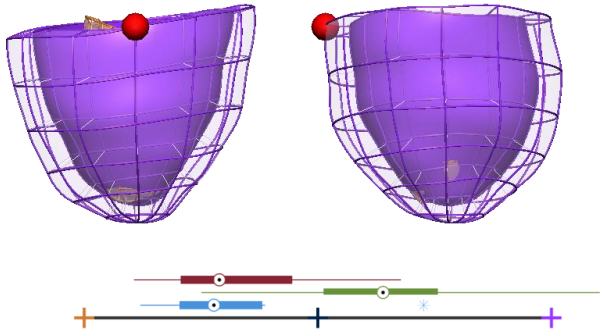
Again, real extreme?

HYP = 1; HF=2; N=3;





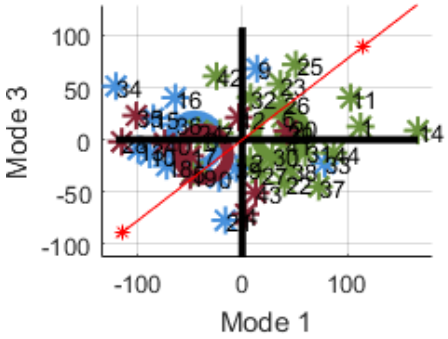




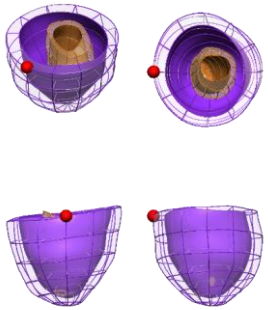
p=6.15e-04. $AUC_{RS} = 0.884$.
$AUC_{L1} = 0.861$.

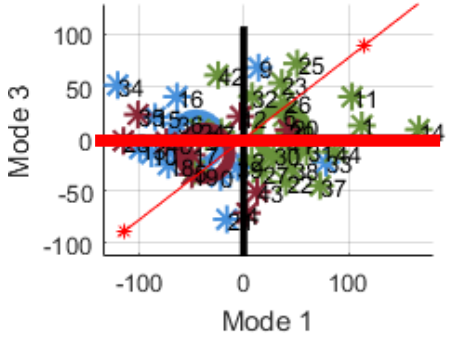
Unique interpretation???

HYP = 1; HF=2; N=3;

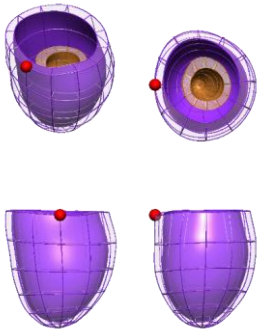


p=6.15e-04. $AUC_{RS} = 0.884$.
$AUC_{L1} = 0.861$.





p=3.41e-03. $AUC_{RS} = 0.856$.
$AUC_{L1} = 0.856$.



5. Export for further analysis

Commands

```
>> Atlas.SaveLDAcoefs();
>> Atlas.SavePCAcoefs();
>> Atlas.SaveVTKmeshes();
```