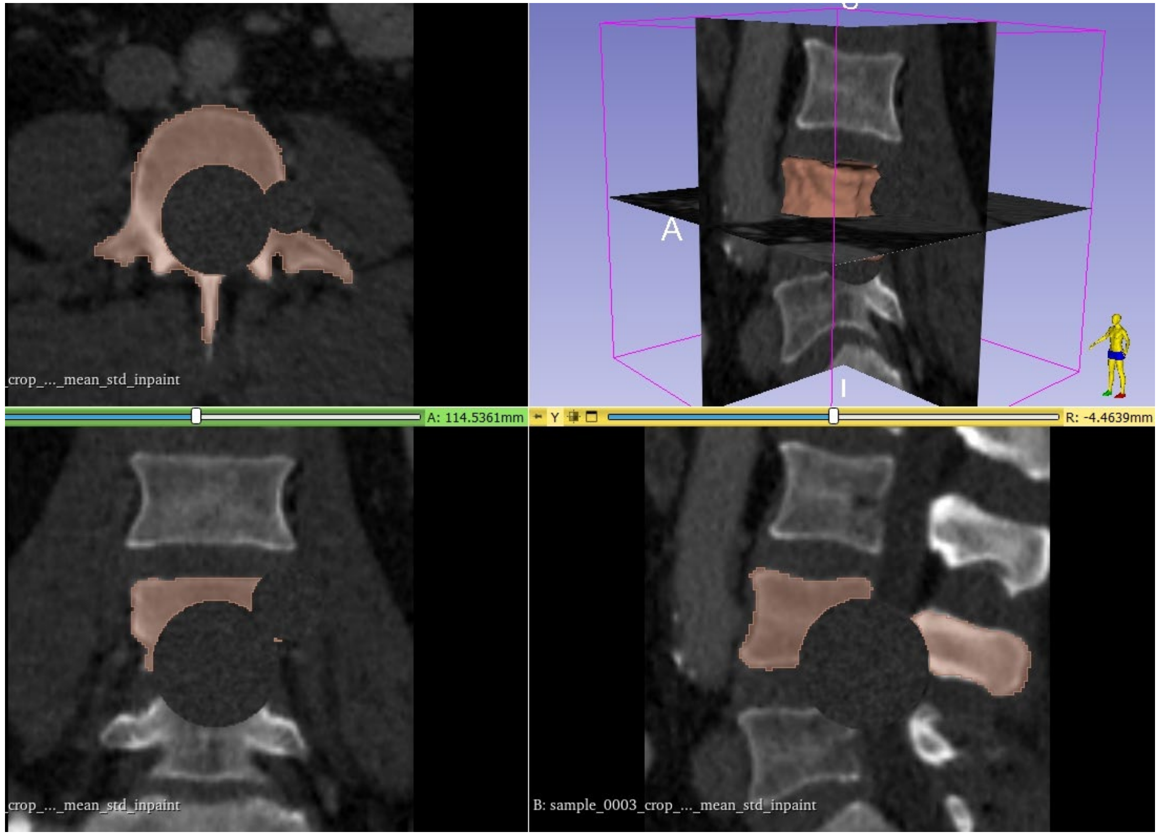
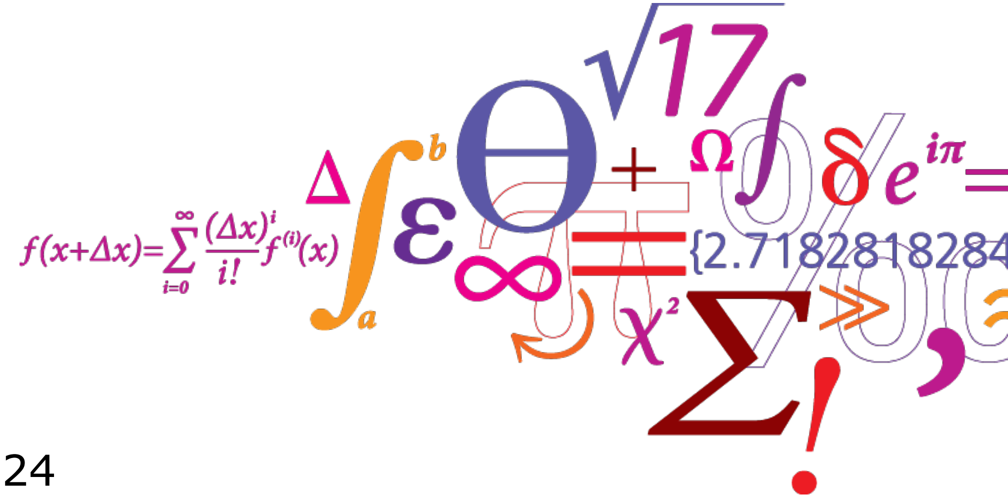
Outlier detection challenge 2024


crop_..._mean_std_inpaint
A
I
A: 114.5361mm
Y
R: -4.4639mm
crop_..._mean_std_inpaint
B: sample_0003_crop_..._mean_std_inpaint

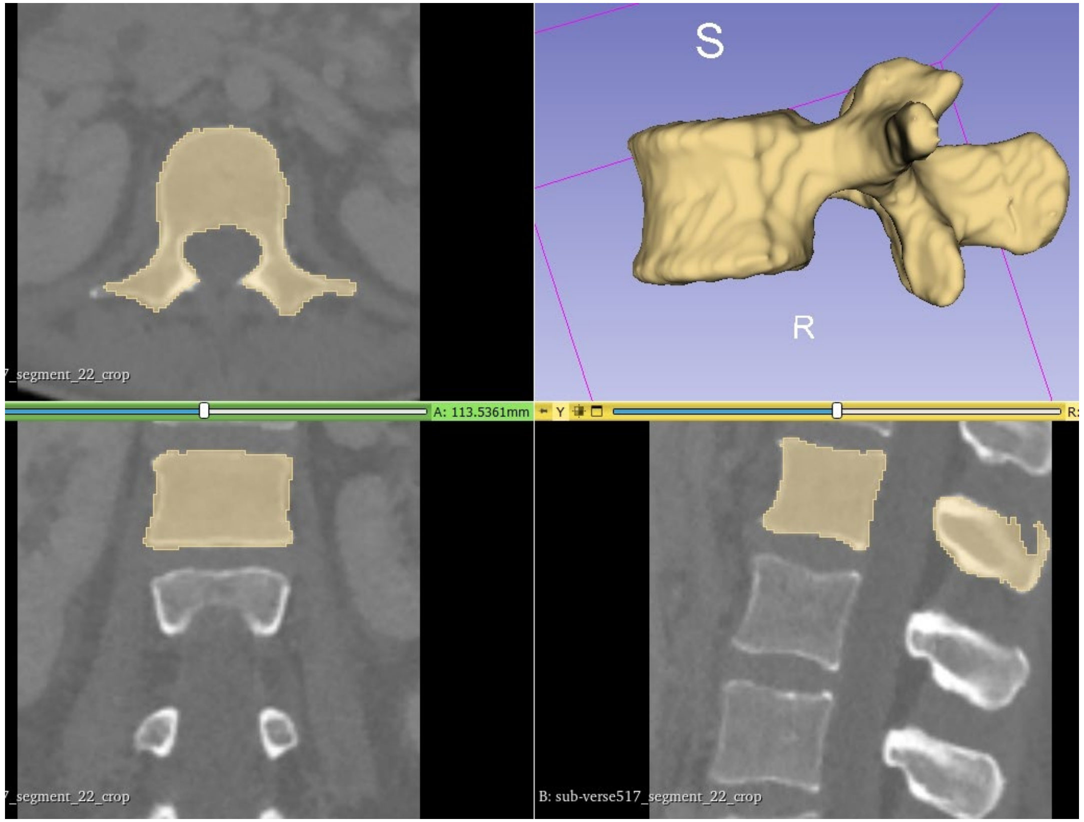


https://github.com/RasmusRPaulsen/OutlierDetectionChallenge2024

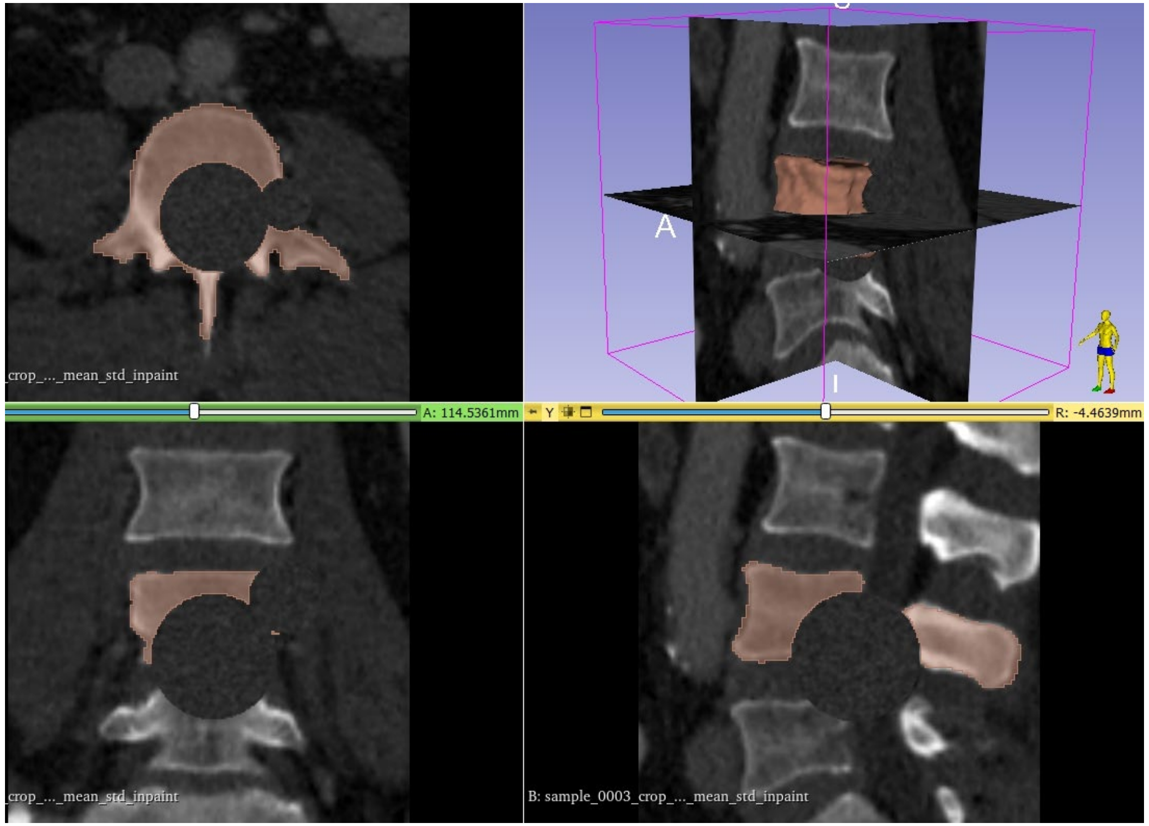
DTU Compute
Department of Applied Mathematics and Computer Science

Outlier detection challenge 2024

- The goal is to create an algorithm that can predict outliers in 3D medical images



Normal


A
I
crop_..._mean_std_inpaint
A: 114.5361mm
Y
R: -4.4639mm
crop_..._mean_std_inpaint
B: sample_0003_crop_..._mean_std_inpaint

Outlier

Something about vocabulary

- Outlier detection:
  - Set of samples with both normal samples and anomalies

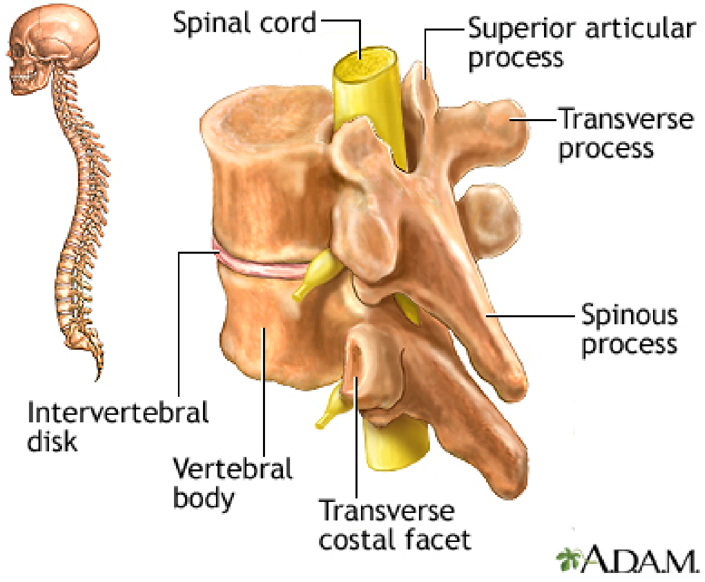
- Novelty detection:
  - Known training set with normal samples
  - Decide if new samples belong to the normal set or can be considered anomalies

- Our challenge:
  - Training set with normal samples + known artificial anomalies
  - Test set with normal samples + known anomalies

- Our challenge is *probably* a **novelty detection** challenge

https://scikit-learn.org/stable/modules/outlier_detection.html

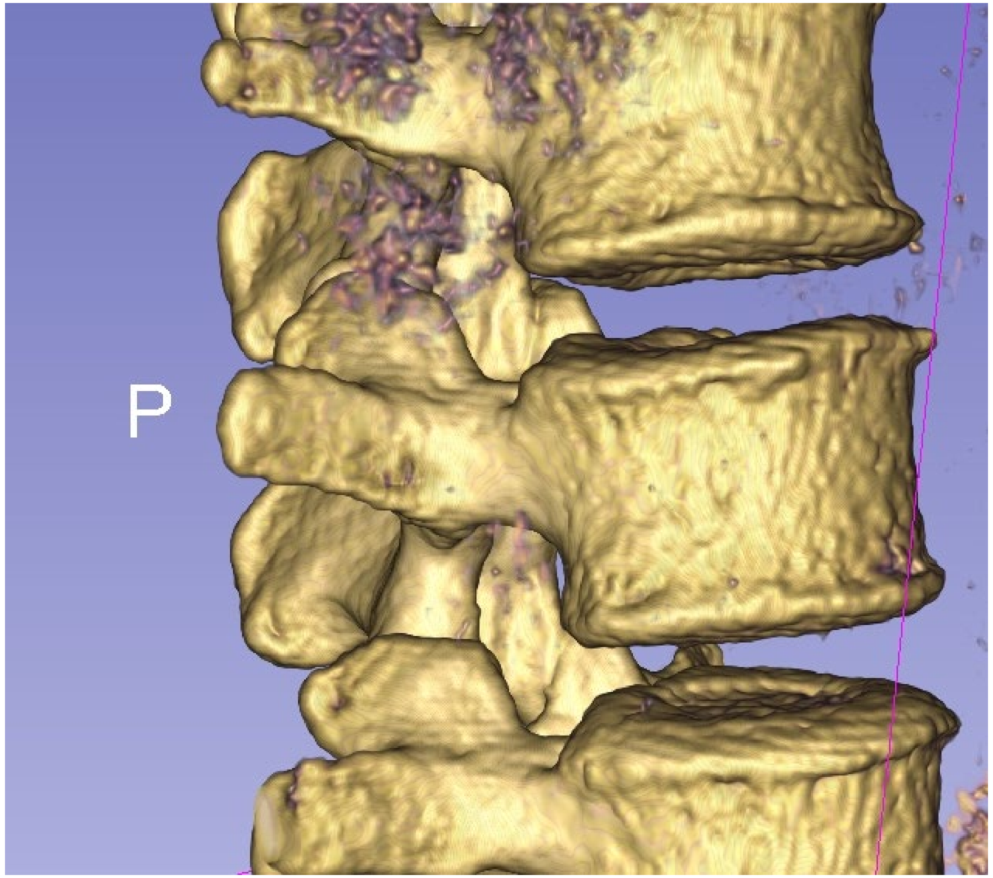
Why is this relevant?



- Rare diseases or diseases that manifests in many different ways
  - You can get a good set of normal samples – but **lot** of variation in normal samples
  - Very difficult to get samples representing all pathologies
  - If you can just tell that this sample is *probably not normal* it is a large help in the further analysis

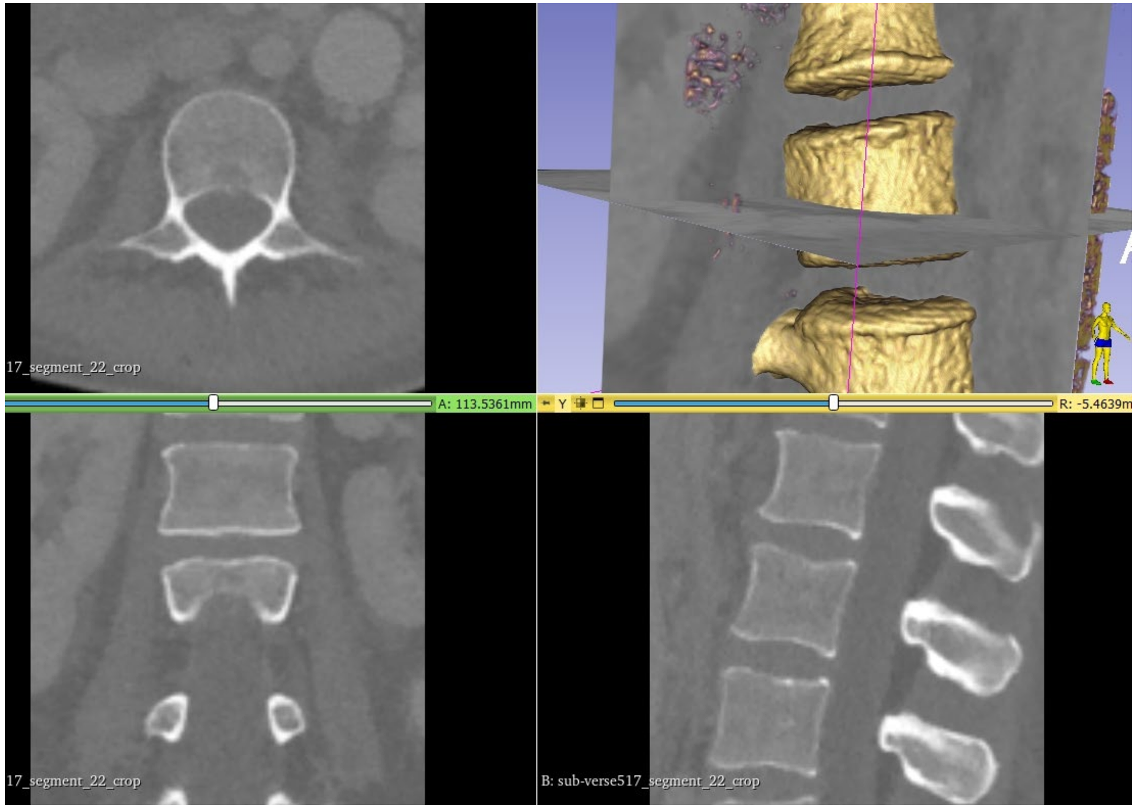
- Lots of different ways to approach the problem:
  - Feature based using simpler statistics
  - Statistical approaches:
    - Principal component analysis, discriminant analysis, support vector machines and so on…
  - Registration based methods:
    - Fit template and examine residuals
  - Deep learning methods
    - Autoencoders

Data representations



- Complex anatomy
  - Lots of natural variation in a normal population
- Several ways to represent the 3D anatomy. We ideally want:
  - Information preserving
  - Compact
  - Possible to use with modern deep learning frameworks

Raw CT scans



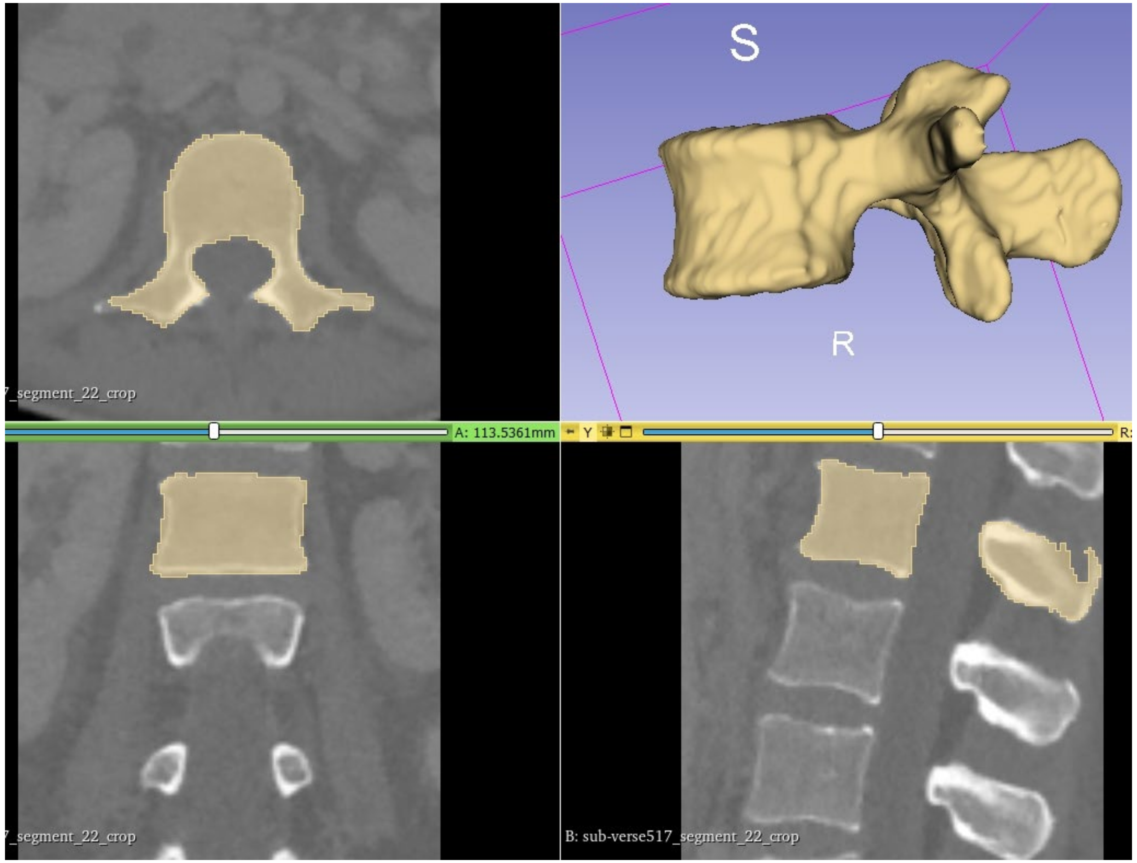
- 3D Voxel volumes

- The voxel values are in calibrated Hounsfield Units (HU)

- HU = 0 : Equals X-ray absorption of water
- HU = -1000: Equals X-ray absorption of air

- We have pre-aligned all 3D volumes using a rigid-body registration using the L1 as the anatomy of interest.

Segmentations / labels

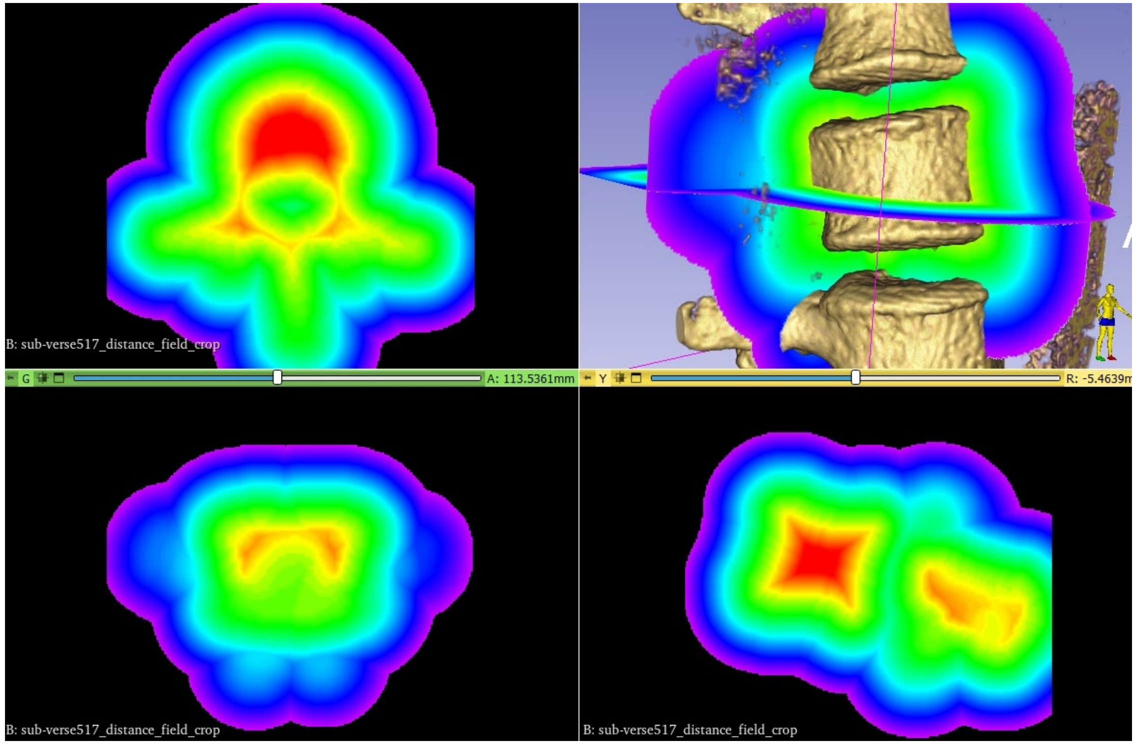


- 3D voxel volume with same dimensions as the CT image
- The voxel value corresponds to the underlying anatomy:

- Value 0: Background
- Value 20: L1

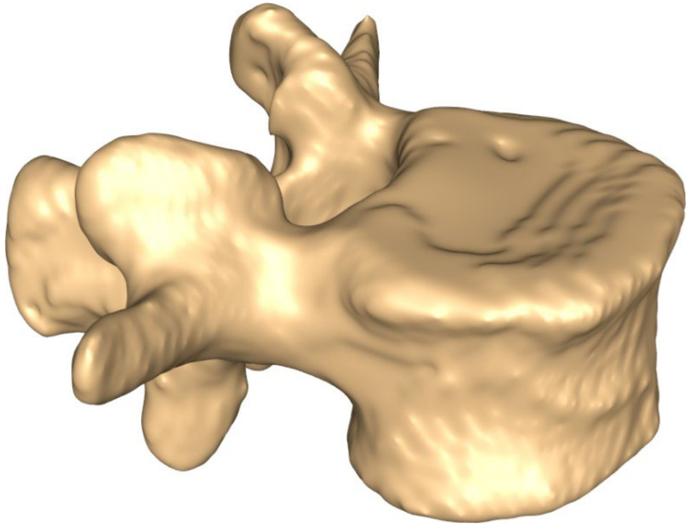
- We have pre-aligned all 3D volumes using a rigid-body registration using the L1 as the anatomy of interest.

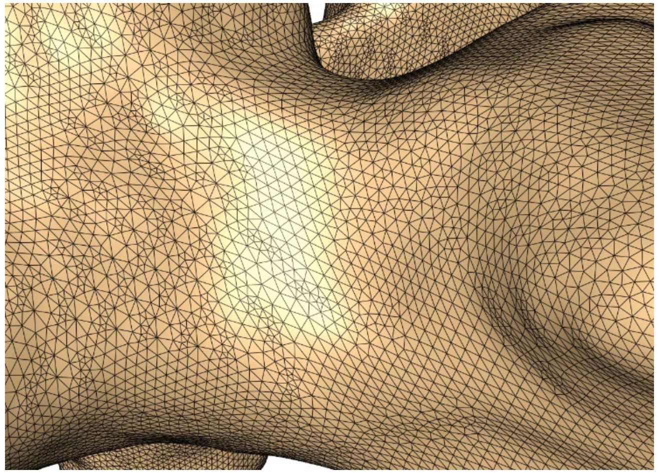
Signed distance field


B: sub-verse517_distance_field_crop
G
A: 113.5361mm
Y
R: -5.4639m
B: sub-verse517_distance_field_crop
B: sub-verse517_distance_field_crop

- 3D voxel volume with same dimensions as the CT image
- The voxel value are the signed distance to the outer surface of the L1
- Clamped to [-50, 50]

Surface / Mesh





- Triangle mesh representing the surface of the L1
- Consists of vertices and edges
- Stored in VTK format (can be converted to other formats)

- We have pre-processed the data
  - Point correspondence over the entire dataset
  - All L1 are represented by the same number of vertices and the same triangles

Data

- Raw CT, Segmentations, distance fields and surfaces
- Artificial outliers

- **Training:** A set of 500+ samples with both normal samples and for each sample there are also artificially generated outliers.

- **Test:** A smaller set of 200 samples and a full set with more samples. Used for keeping track of your prediction scores during the challenge.

- You should split the training data into your own custom training / validation set – there is a script for that.

Teams

- We have pre-made 15 teams
- You should add your name and email to a team on the paper in the conference room
- Try to distribute yourself evenly
- You are responsible for gathering and organizing your team

Getting starting with the Python code

- Clone the GitHub repository
  - https://github.com/RasmusRPaulsen/OutlierDetectionChallenge2024
- Download the data (hopefully already done)
- Change your team name in the JSON configuration file
- Follow the instruction on the GitHub repository – see the getting started part

Challenge results

- The team results are computed and put on the homepage several times a day
- http://fungi.compute.dtu.dk:8080/
- Submit results to the challenge server, at least, two times daily
- You should submit the final set scores latest Thursday at 16h!

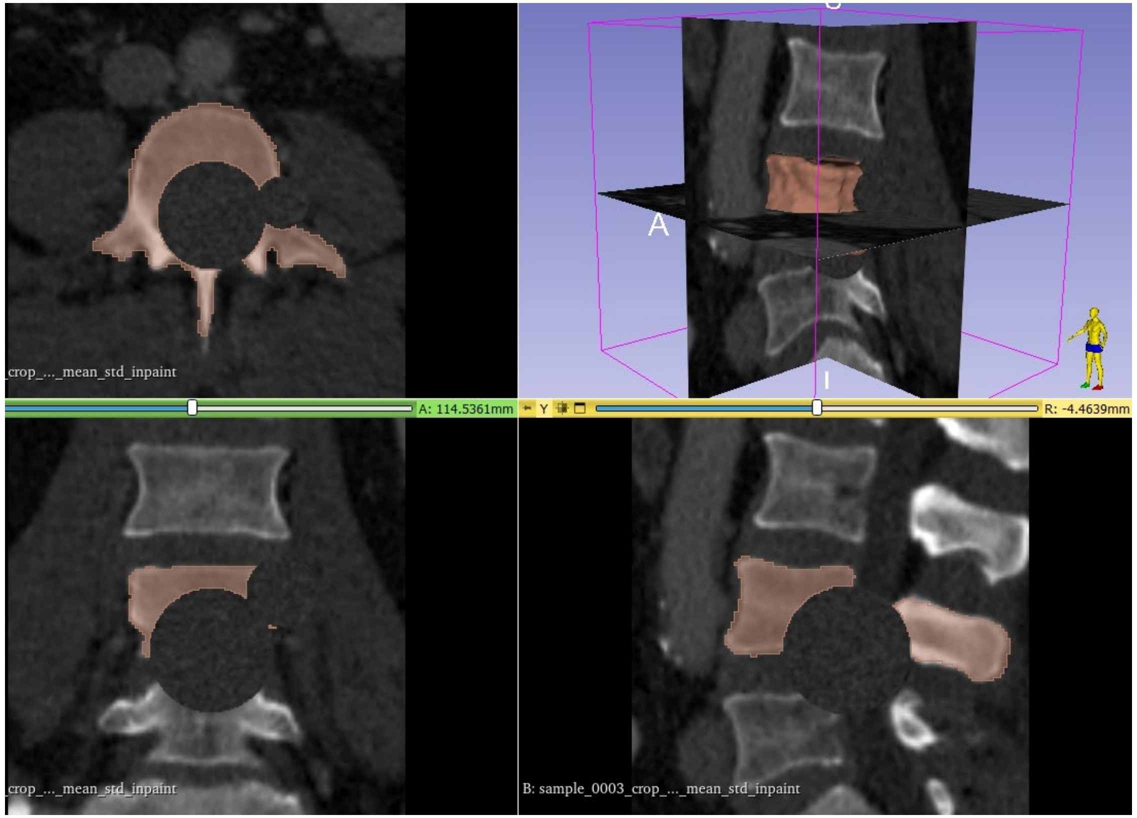
Metrics

- We use the same metrics as used in binary / two-class classification

- They are derived from the confusion matrix:
  - The number of true positives (TP)
  - The number of false positives (FP)
  - The number of true negatives (TN)
  - The number of false negatives (FN)

- The metrics we compute are:
  - Accuracy
  - Precision
  - Recall
  - F1
  - Cohens Kappa

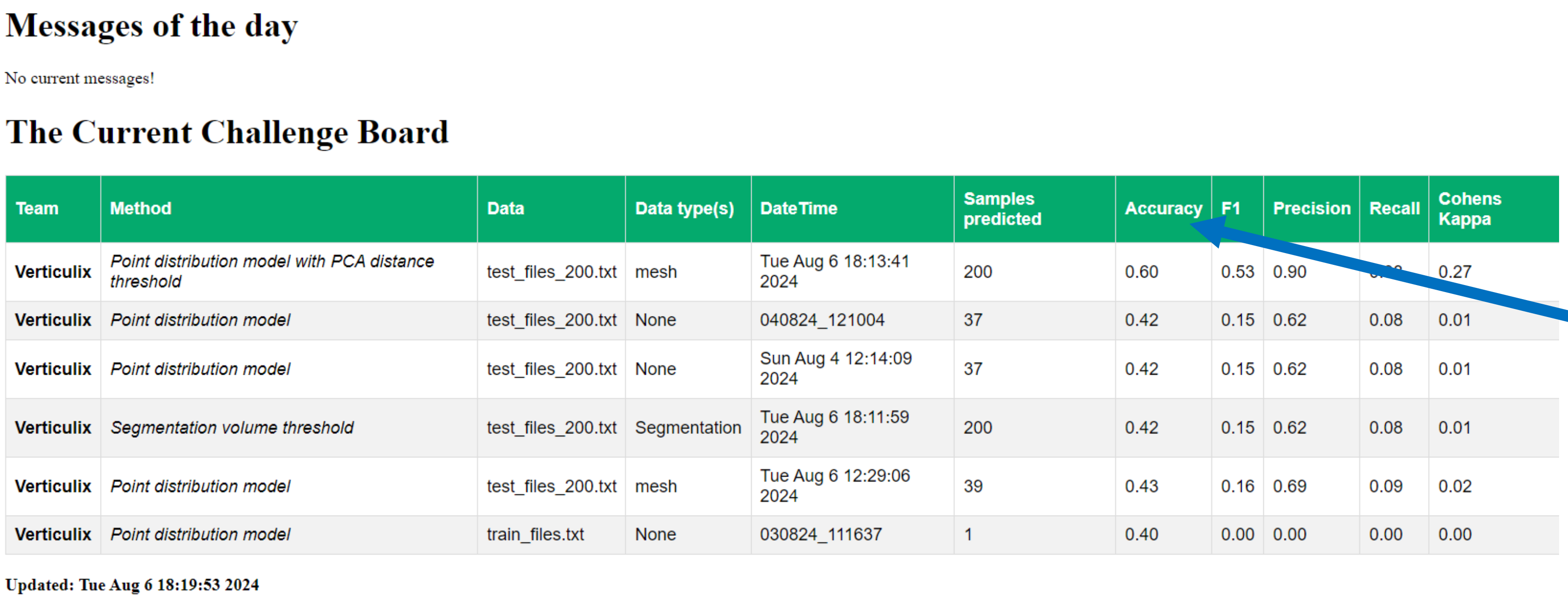
Data visualization


crop_..._mean_std_inpaint
A
I
A: 114.5361mm
Y
R: -4.4639mm
crop_..._mean_std_inpaint
B: sample_0003_crop_..._mean_std_inpaint

- It is important to be able to visualise the samples
- In particular if you are using generative models
  - To check the output

- 3D Slicer is a very powerful tool. It can (among tons of other things)
  - Visualise 3D volumes
  - Volume rendering
  - Segmentation rendering
  - Surface rendering

The board – some baseline results



Messages of the day

No current messages!

The Current Challenge Board

Team	Method	Data	Data type(s)	DateTime	Samples predicted	Accuracy	F1	Precision	Recall	Cohens Kappa
Verticulix	*Point distribution model with PCA distance threshold*	test_files_200.txt	mesh	Tue Aug 6 18:13:41 2024	200	0.60	0.53	0.90	[illegible]	0.27
Verticulix	*Point distribution model*	test_files_200.txt	None	040824_121004	37	0.42	0.15	0.62	0.08	0.01
Verticulix	*Point distribution model*	test_files_200.txt	None	Sun Aug 4 12:14:09 2024	37	0.42	0.15	0.62	0.08	0.01
Verticulix	*Segmentation volume threshold*	test_files_200.txt	Segmentation	Tue Aug 6 18:11:59 2024	200	0.42	0.15	0.62	0.08	0.01
Verticulix	*Point distribution model*	test_files_200.txt	mesh	Tue Aug 6 12:29:06 2024	39	0.43	0.16	0.69	0.09	0.02
Verticulix	*Point distribution model*	train_files.txt	None	030824_111637	1	0.40	0.00	0.00	0.00	0.00

Updated: Tue Aug 6 18:19:53 2024

Press here to sort by that column

http://fungi.compute.dtu.dk:8080/

Presentations and results

- The final project presentations and results are on Thursday from 17:10-18:45

- Each team has 4 minutes to present their project with maximum 3 slides
  - Brief review of your approach
  - Did it work as expected?

- Finally, the **final test** results are presented by the organizers

Adobe Stock | #220770130

Guide and tips

- Use, at least, one non-deep learning method first

- Divide the data with ground truth into your own train / validation splits

Rules

- We are not checking for cheating and believe in fair play and that you are here to learn

- We do not recommend you to:
  - Use other teams names
  - Get images and segmentations from external sites
  - Hack or modify the submission script

Have fun!