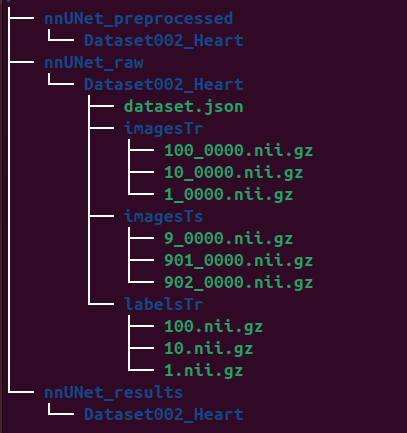
**TRAIN A nnUNet**

**1) Install pytorch with CUDA**

**2) Install nnunetv2**

**3) Organize the data and create/modify “dataset.json” file:**



**4) Set the location of the directories:**

**Linux**

**export nnUNet\_raw="/scratch/awias/data/ImageCas/nnUNet\_dataset/nnUNet\_raw"**

**export nnUNet\_results="/scratch/awias/data/ImageCas/nnUNet\_dataset/nnUNet\_results"**

**export nnUNet\_preprocessed="/scratch/awias/data/ImageCas/nnUNet\_dataset/nnUNet\_preprocessed"**

**Linux WLS**

**export nnUNet\_raw="/home/awias/data/ImageCas/Derivations/nnUNet\_dataset/nnUNet\_raw"**

**export nnUNet\_results="/home/awias/data/ImageCas/Derivations/nnUNet\_dataset/nnUNet\_results"**

**export nnUNet\_preprocessed="/home/awias/data/ImageCas/Derivations/nnUNet\_dataset/nnUNet\_preprocessed"**

Check if they were all set correctly: *printenv*

Or specifically: (because the order seems to be random)

**printenv nnUNet\_raw**

**printenv nnUNet\_results**

**printenv nnUNet\_preprocessed**

**Windows Powershell**

*$env:nnUNet\_raw = “C:\data\ImageCAS\Derivations\nnUNet\_dataset\nnUNet\_raw”*

*$env:nnUNet\_results = "C:\data\ImageCAS\Derivations\nnUNet\_dataset\nnUNet\_results"*

*$env:nnUNet\_preprocessed = "C:\data\ImageCAS\Derivations\nnUNet\_dataset\nnUNet\_preprocessed"*

Check if they were all set correctly: *Get-ChildItem Env:*

**5) Extract Dataset Fingerprint: to analyze the dataset to understand its properties, such as image size, spacing… :**

nnUNetv2\_extract\_fingerprint -d 4 -verify\_dataset\_integrity -verbose -pl nnUNetPlannerResEncL -c 3d\_fullres

Number is dataset ID

**6) Plan the experiment: to determine the configuration (2D, 3D low-resolution or 3D full-resolution):**

nnUNetv2\_plan\_experiment -d 4 -c 3d\_fullres -pl nnUNetPlannerResEncL -np 4

Number is dataset ID

**7) Preprocess the Dataset:**

nnUNetv2\_preprocess -d 4 -c 3d\_fullres -pl nnUNetResEncUNetLPlans -np 8

**8) Train the model for the 5 folds:**

nnUNetv2\_train 4 3d\_fullres 0 -tr nnUNetTrainerNoMirroring -p nnUNetResEncUNetLPlans

nnUNetv2\_train 4 3d\_fullres 1 -tr nnUNetTrainerNoMirroring -p nnUNetResEncUNetLPlans

nnUNetv2\_train 4 3d\_fullres 2 -tr nnUNetTrainerNoMirroring -p nnUNetResEncUNetLPlans

nnUNetv2\_train 4 3d\_fullres 3 -tr nnUNetTrainerNoMirroring -p nnUNetResEncUNetLPlans

nnUNetv2\_train 4 3d\_fullres 4 -tr nnUNetTrainerNoMirroring -p nnUNetResEncUNetLPlans

important paths:

kalder nnUNetDatasetBlosc2 (/home/awias/code/nnUNet/nnunetv2/training/dataloading/nnunet\_dataset.py)

 søg efter self.optimizer.step() i filen nnUNetTrainer.py

Her ændrer du epochs osv:

/home/awias/code/nnUNet\_v2/nnunetv2/training/nnUNetTrainer/nnUNetTrainer.py

Her gemmer jeg logits:

/home/awias/code/nnUNet\_v2/nnunetv2/inference/export\_prediction.py

Ændring af info\_dict

nnUNet\_v2/nnunetv2/utilities/get\_network\_from\_plans.py

**9) Make predictions:**

nnUNetv2\_predict -i ../../nnUNet\_raw/Dataset002\_Heart/imagesTs/ -o predictions/ -d 2 -c 3d\_fullres -f 0 1 2 3 4

**10) Evaluate the results:**

nnUNetv2\_evaluate\_folder labelsTs predictions -jpfile dataset.json -pfile plans.json

