A Cascade of CNNs for Multiple Sclerosis Lesion Segmentation

Ian Jacobsen

Spring 2019

Introduction

- The purpose of this project was to reproduce the winning algorithm in the MICCAI 2016 lesion segmentation challenge.

- The challenge provided a data set of MRI scans of 15 patients with multiple sclerosis (MS).

- The goal of the challenge was to segment lesion areas from nonlesion areas.

Importance

- Manually annotating MRI scans is:
    - Expensive
    - Time consuming
    - Subject to inter-expert variability

- Studying lesion size and abundance is helpful for determining effectiveness of treatment and the stage of MS.

Data Set

- 15 annotated MRI scans.

- Manually annotated by 7 expert radiologists.

- Experts’ annotations were combined to form a consensus image.

- FLAIR modality used.

Data Set

- 3 MRI scanners used:
  - Philips Ingenia 3T (5 patients)
  - Siemes Area 1.5T (5 patients)
  - Siemes Verio 3T (5 patients)

- Largest scan size is 512 x 512 x 144.

- Smallest scan size is 226 x 224 x 128.

Winning Approach

- Cascade of two identical networks.

- Trained in tandem.

- N2’s training data depends on N1.

Winning Approach

- Network 1 (N1) finds *candidate* patches.

- Network 2 (N2) makes final decision of if a candidate patch is centered on a lesion.

Winning Approach

- In practice, every patch is passed through N1.

- All patches that N1 classifies as *positive* are then passed into N2.

- Patch classifications are combined into final segmentation image.

Recreation

- Data handling using nibabel and pandas.

- Learning using Keras with Tensorflow backend.

- Trained on HPC clusters.

Network Architecture

- 7 layer CNNs.

- Conv, MP, Conv, MP, FC, Drop, Softmax.

Training

- Trained 15 models using leave-one-out.
- Each model is trained on 14 patients.
- Patches were fetched for each of the 14 patients, shuffled, then split for training and validation.
- 80% for training.

Validation

- 20% of patches for validation.

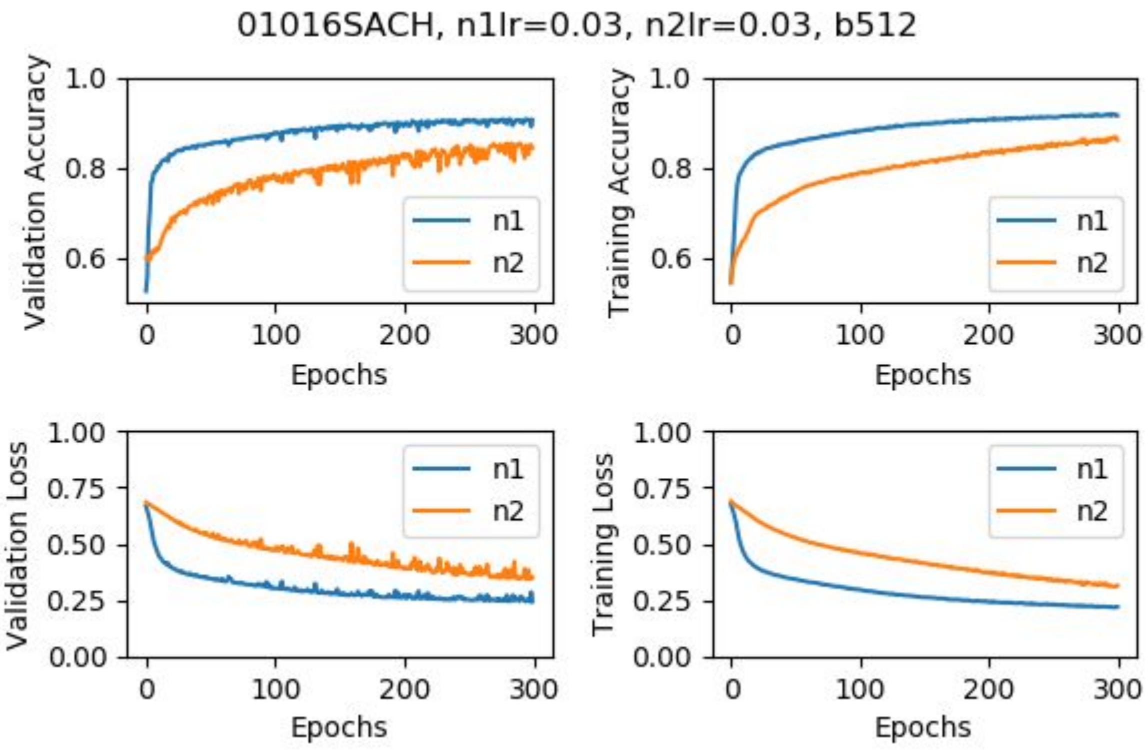
- Batch size, learning rates, and epochs were found.

Testing

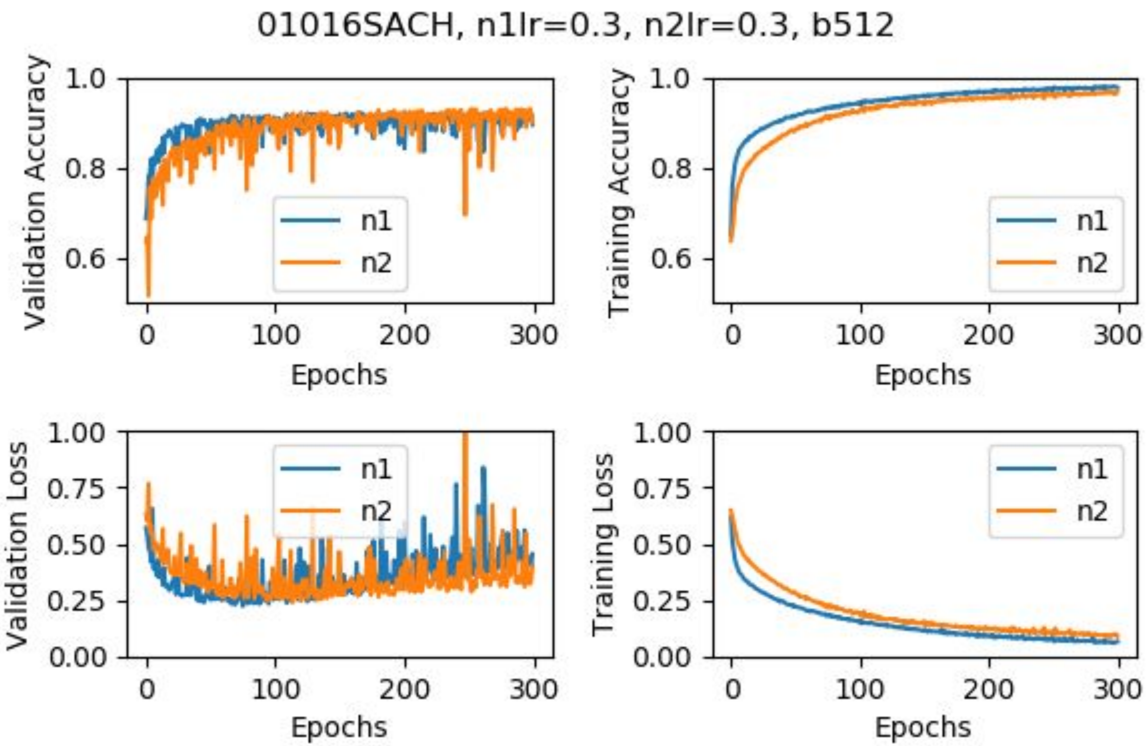
- For each model, test on patient that was left out.

- Score based on accuracy of classifications, and Dice score.

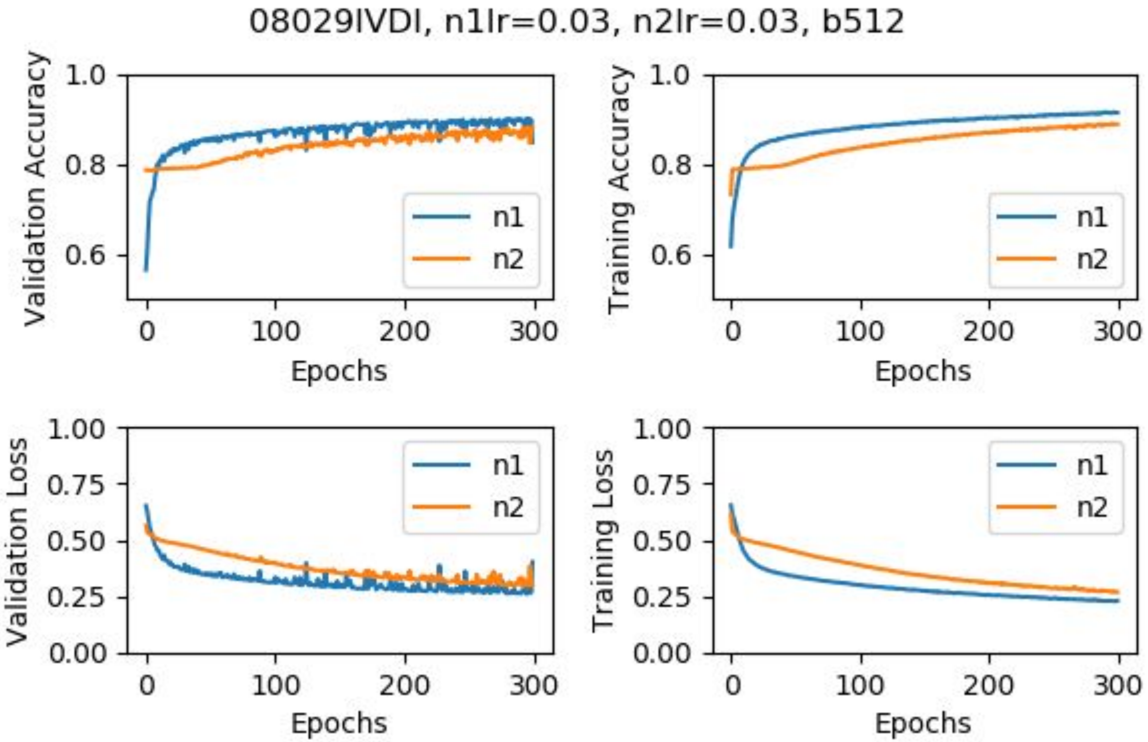
Training and Validation Performance



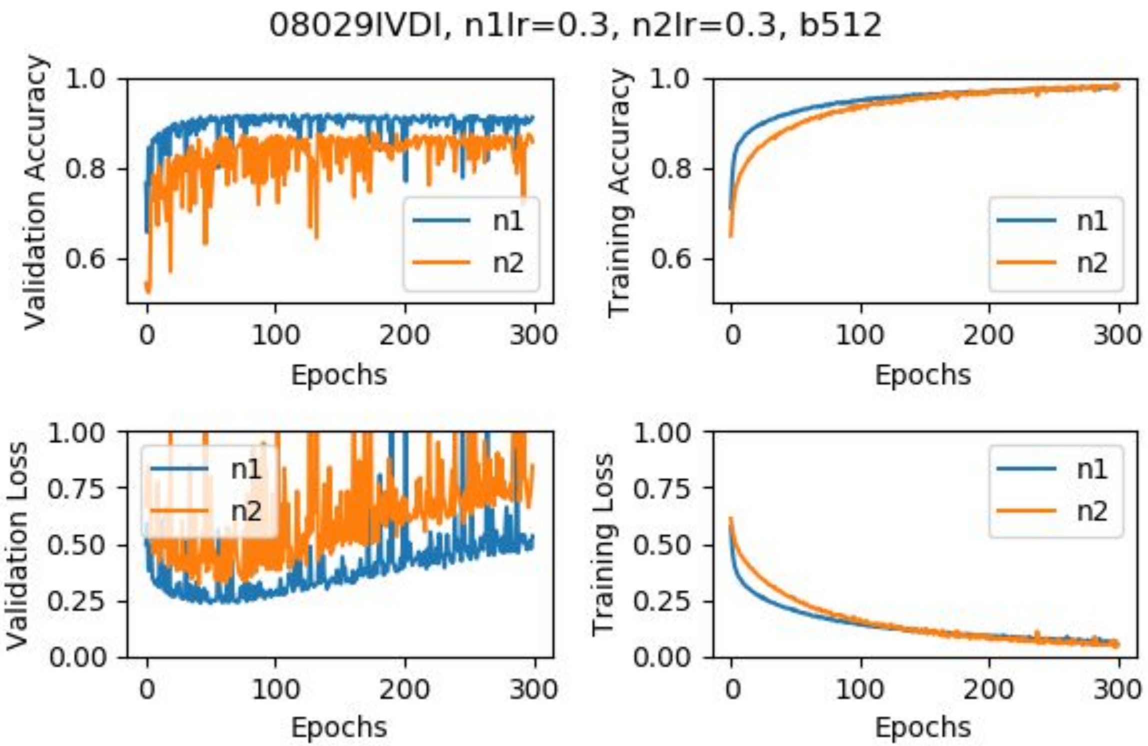
Training and Validation Performance



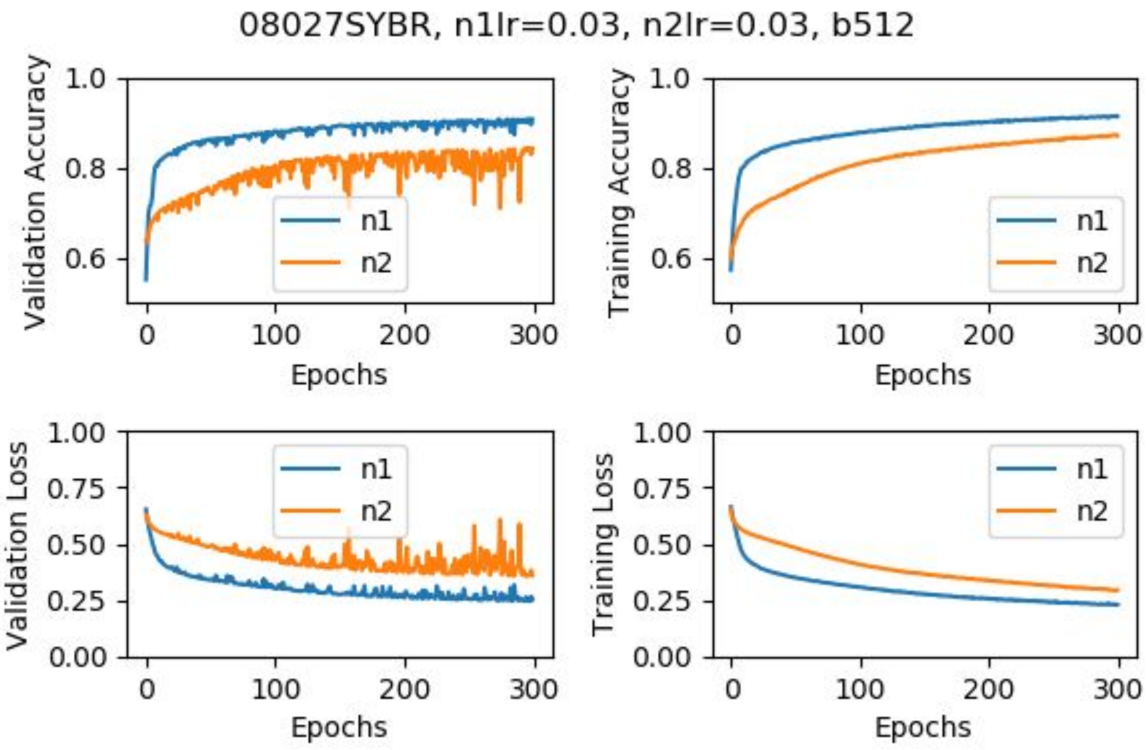
Training and Validation Performance



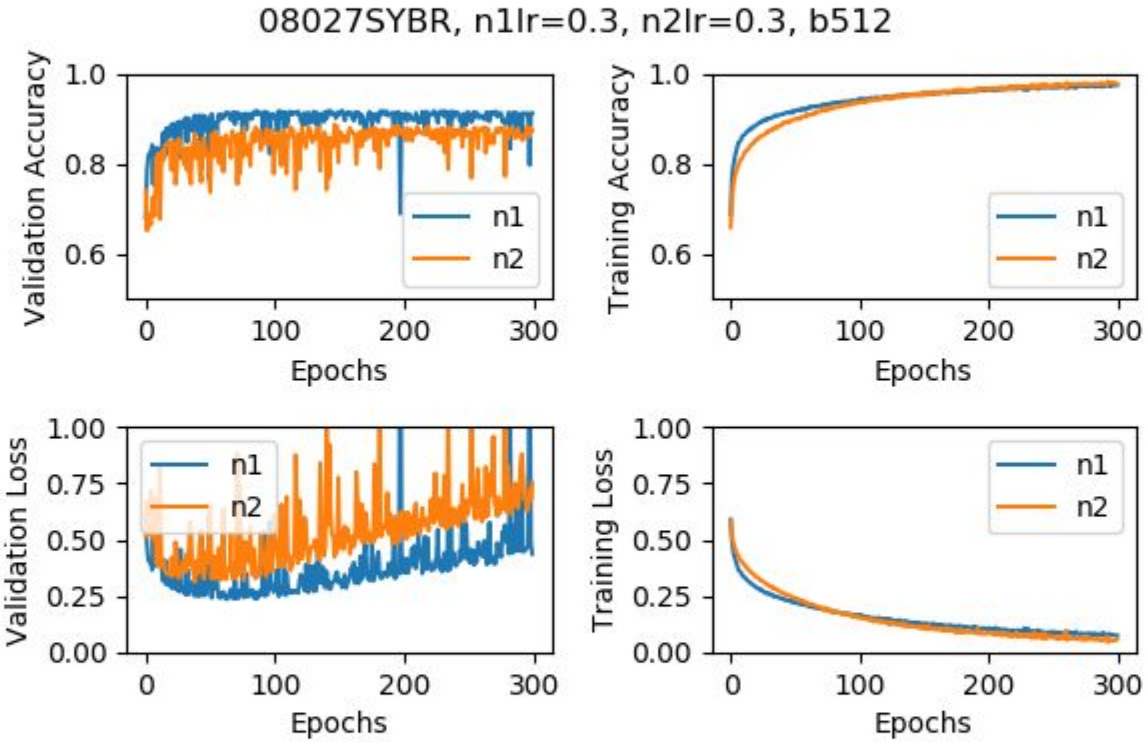
Training and Validation Performance



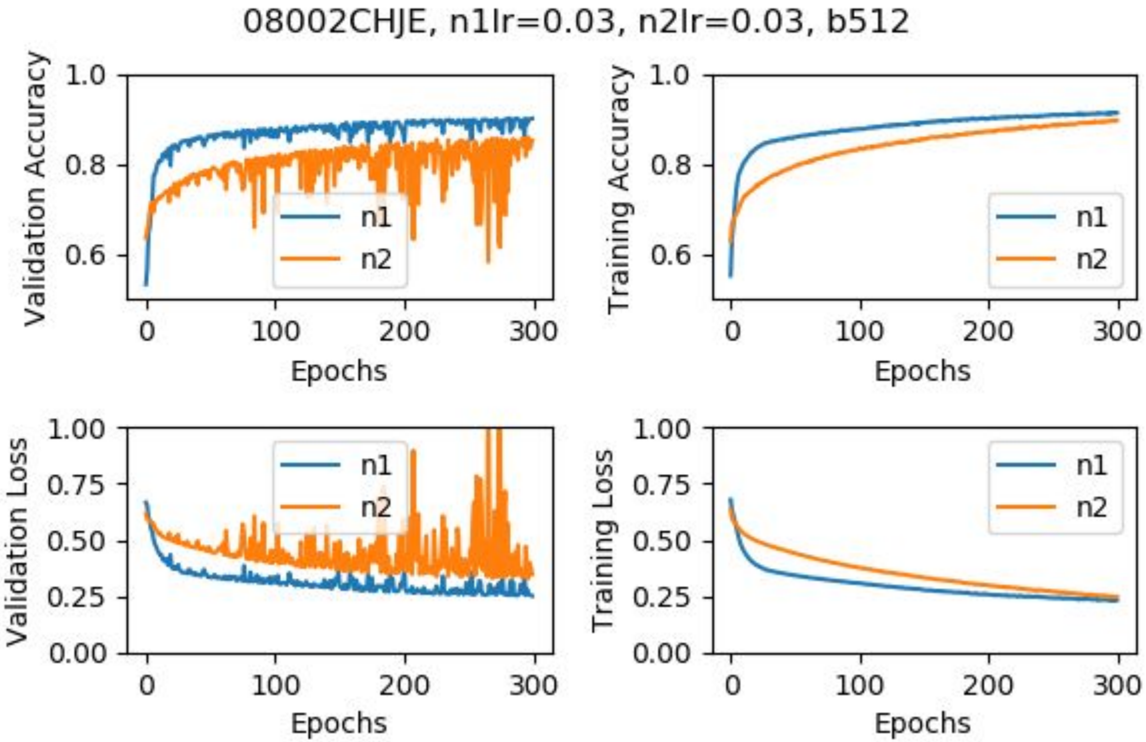
Training and Validation Performance



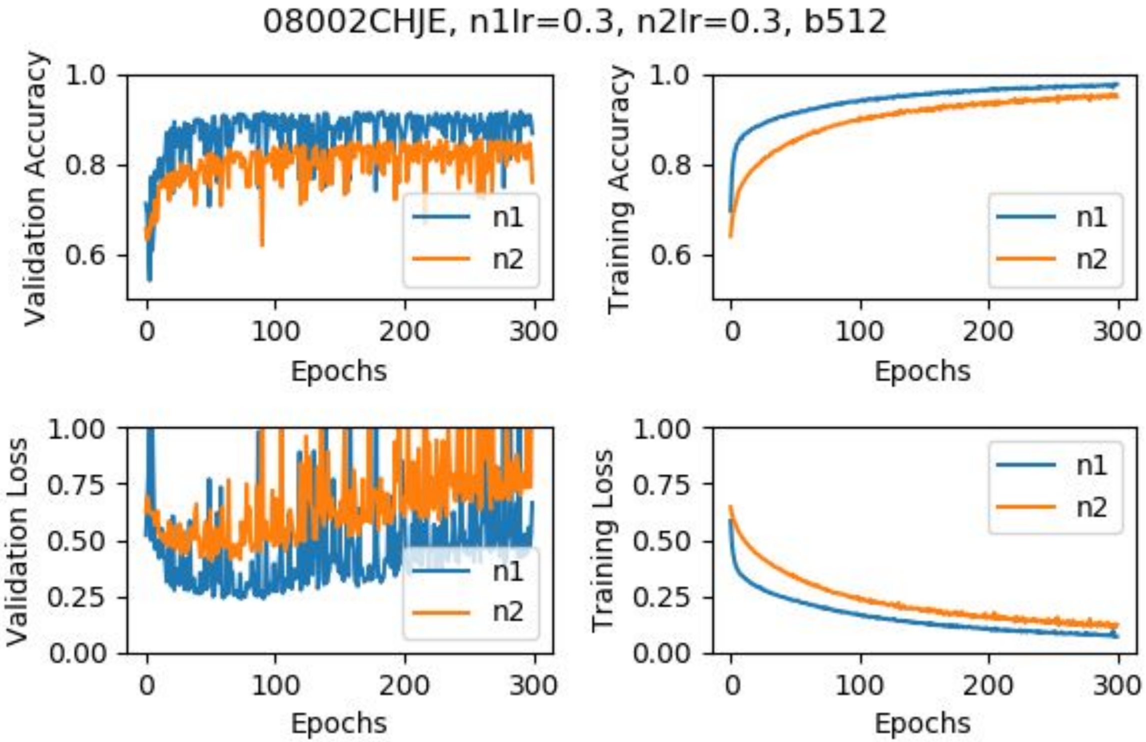
Training and Validation Performance



Training and Validation Performance



Training and Validation Performance



Dice Score

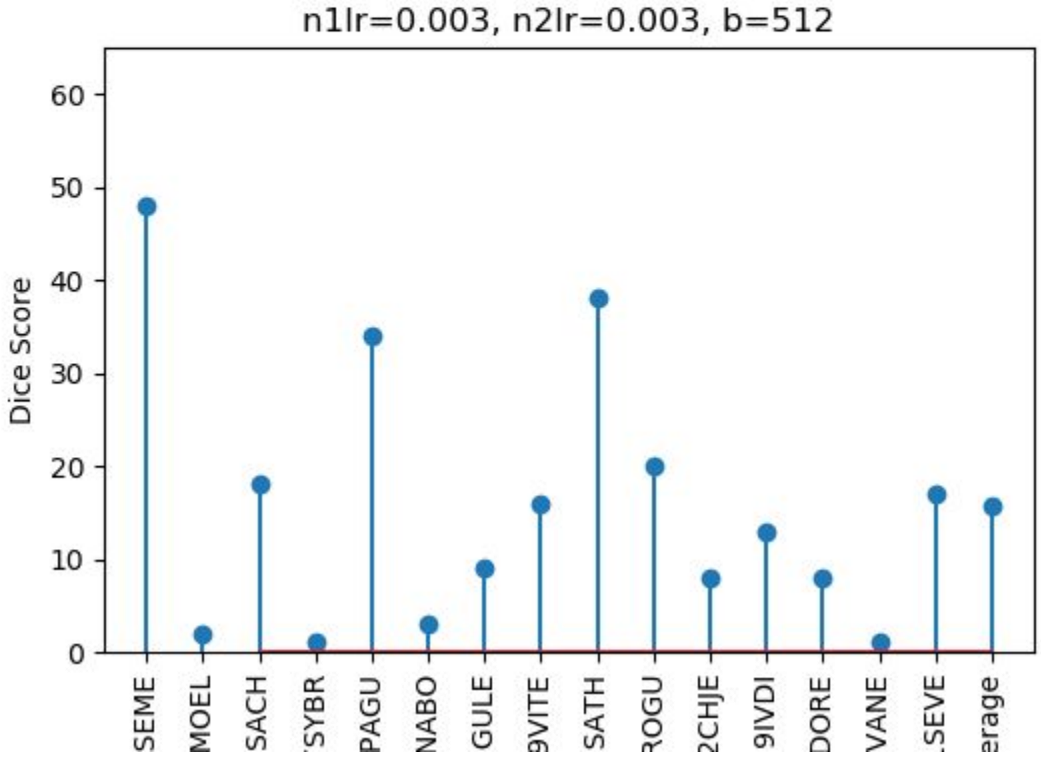
$$DSC = \frac{2 \times TP}{FN + FP + 2 \times TP} \times 100$$

Best Average Dice Score

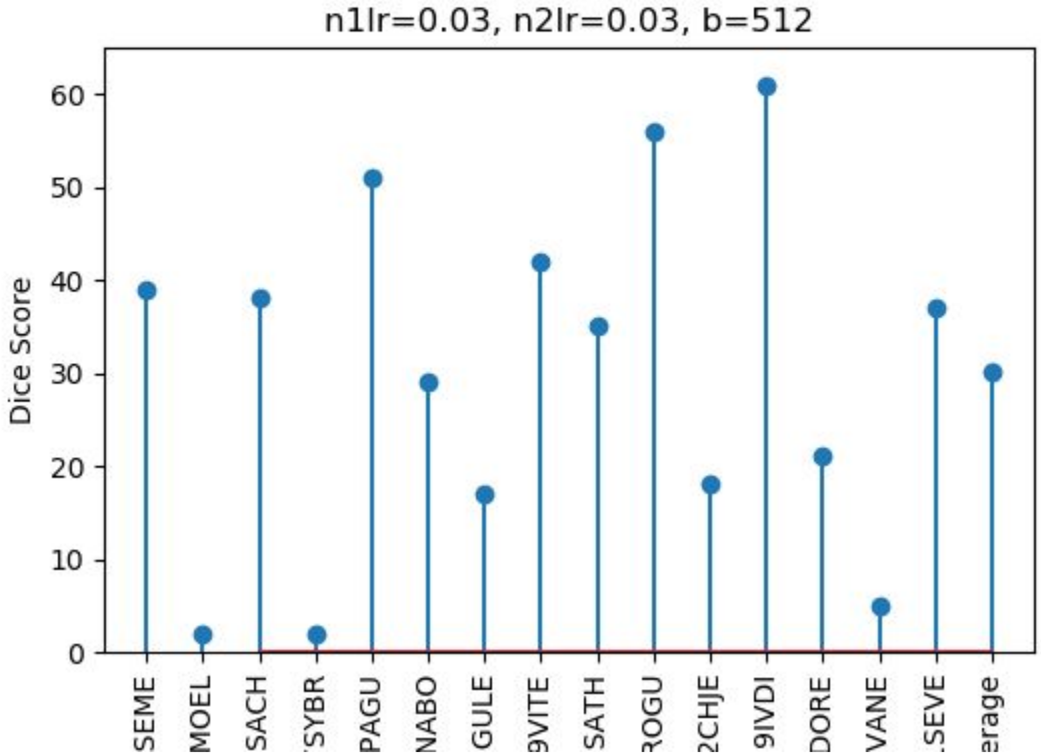
TABLE II: Test Results for N1LR=0.03, N2LR=0.03, batch=512

	num pixels	accuracy	false positives	false negatives	dice
07043SEME	1061417	0.98532	0.0108628	0.00381754	39
07001MOEL	986032	0.925387	0.0743181	0.000295122	2
01016SACH	4774673	0.831046	0.168325	0.000629572	38
08027SYBR	3543293	0.905221	0.0947158	6.29358e-05	2
01038PAGU	5445523	0.987371	0.00466273	0.00796673	51
07010NABO	924395	0.993595	0.00531591	0.00108936	29
01042GULE	4908547	0.949261	0.0342712	0.0164676	17
01039VITE	5228366	0.963937	0.0223169	0.0137462	42
07003SATH	876275	0.964754	0.00463781	0.030608	35
08037ROGU	3858673	0.983629	0.00937758	0.0069936	56
08002CHJE	3994067	0.944291	0.0552855	0.000423128	18
08029IVDI	3523149	0.969521	0.026912	0.00356698	61
07040DORE	982787	0.98828	0.00973151	0.00198822	21
01040VANE	3930109	0.970369	0.0295905	4.07113e-05	5
08031SEVE	3565957	0.988017	0.0116451	0.000337918	37
Average	3.17355e+06	0.956667	0.0374645	0.0058689	30.2

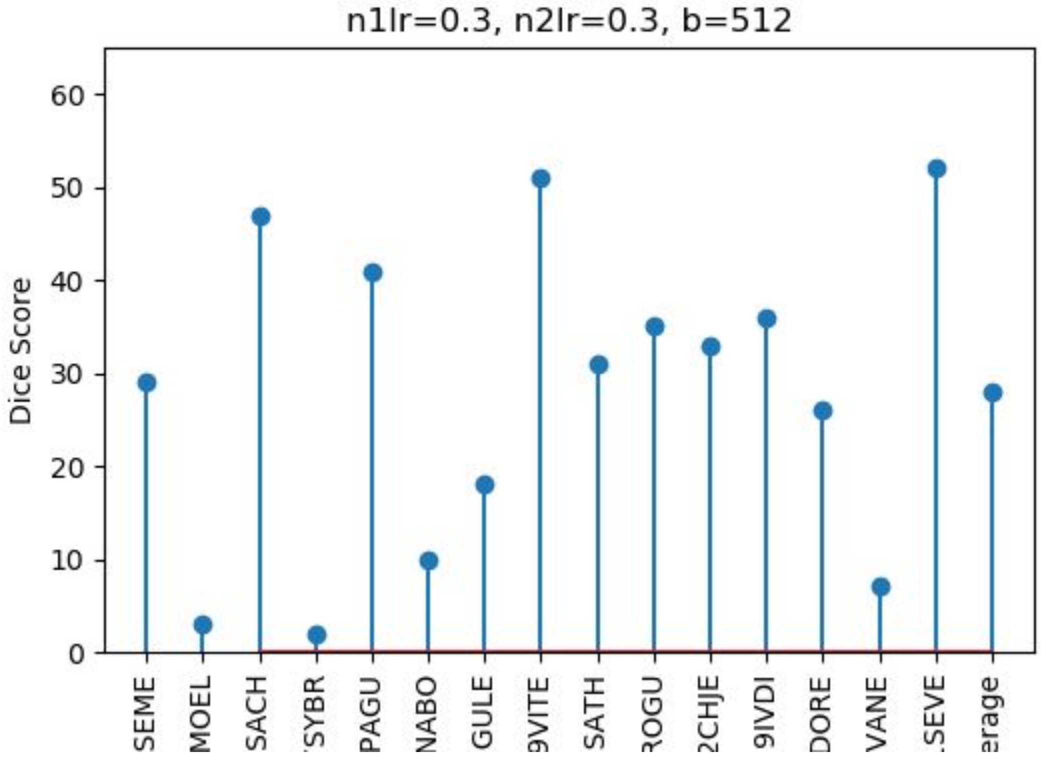
Dice Score Stem Plot Performance



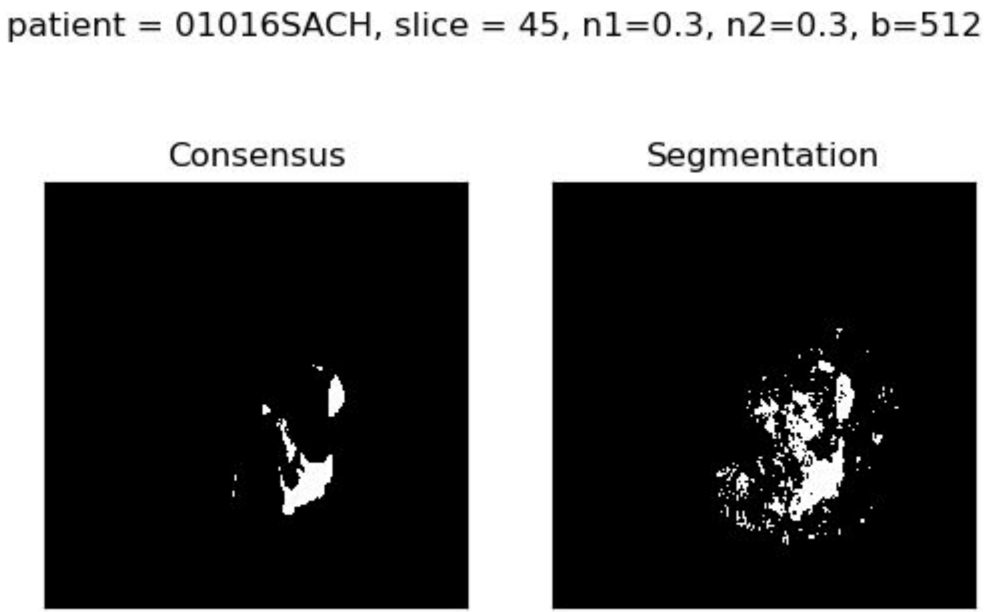
Dice Score Stem Plot Performance



Dice Score Stem Plot Performance

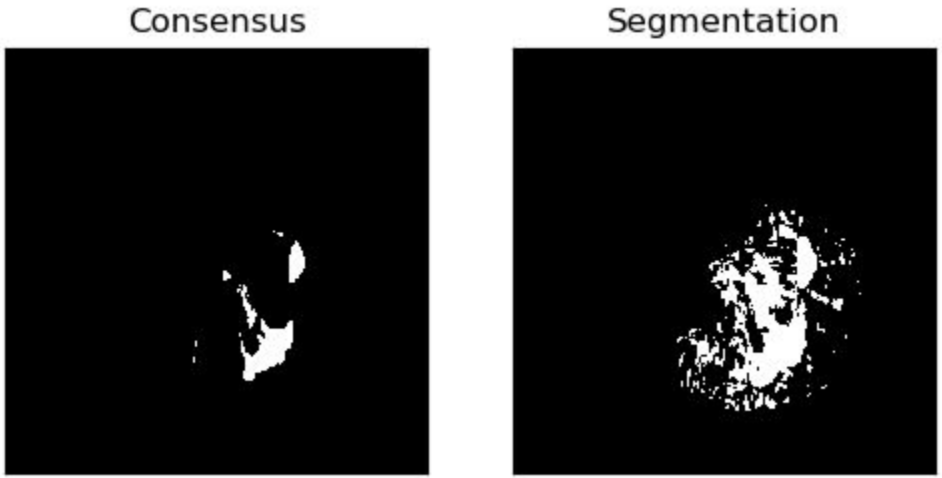


Results

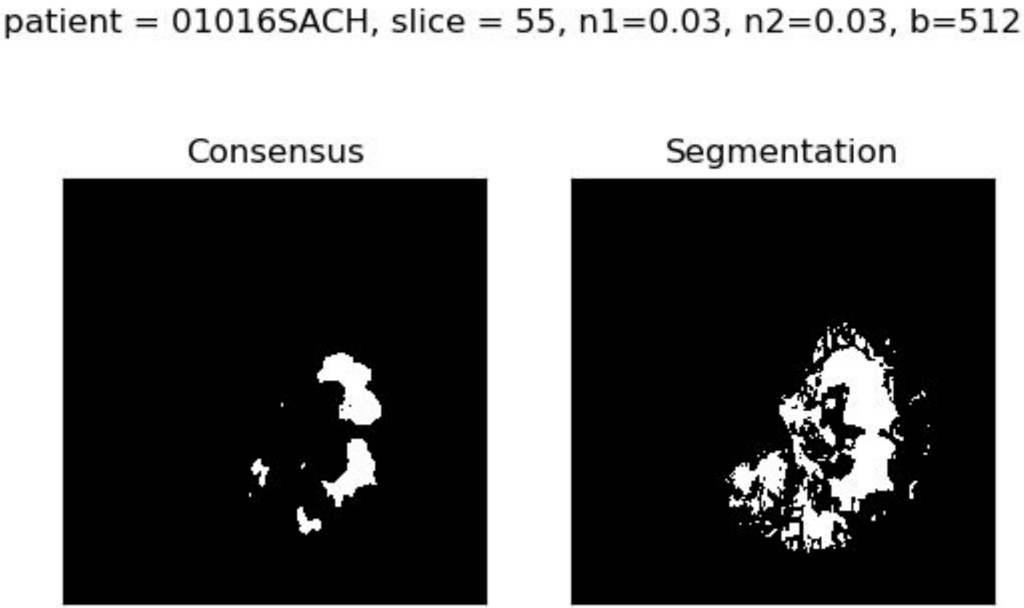

patient = 01016SACH, slice = 45, n1=0.3, n2=0.3, b=512
Consensus
Segmentation

Results

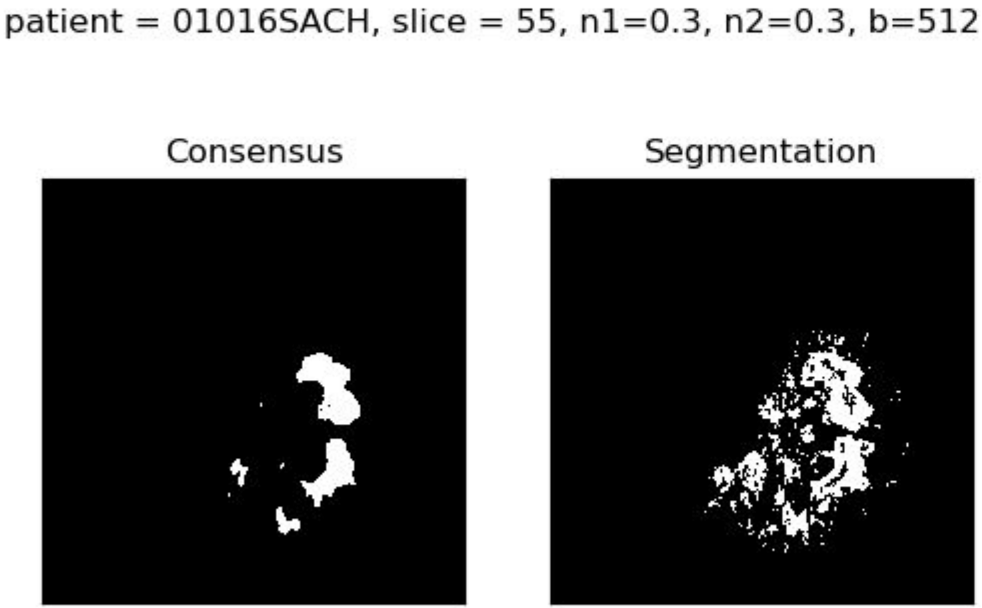
patient = 01016SACH, slice = 45, n1=0.03, n2=0.03, b=512



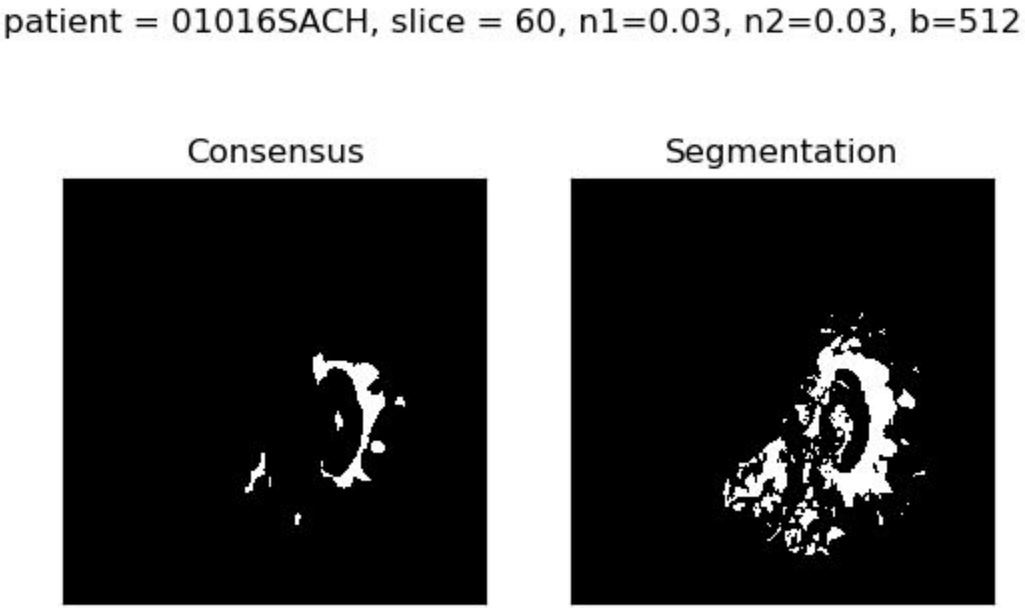
Results


patient = 01016SACH, slice = 55, n1=0.03, n2=0.03, b=512
Consensus
Segmentation

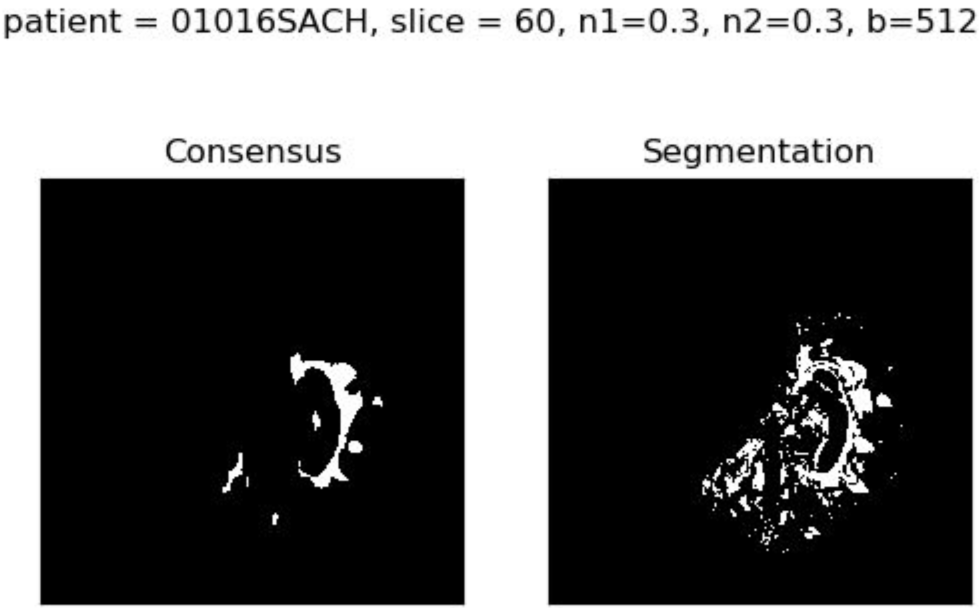
Results


patient = 01016SACH, slice = 55, n1=0.3, n2=0.3, b=512
Consensus
Segmentation

Results

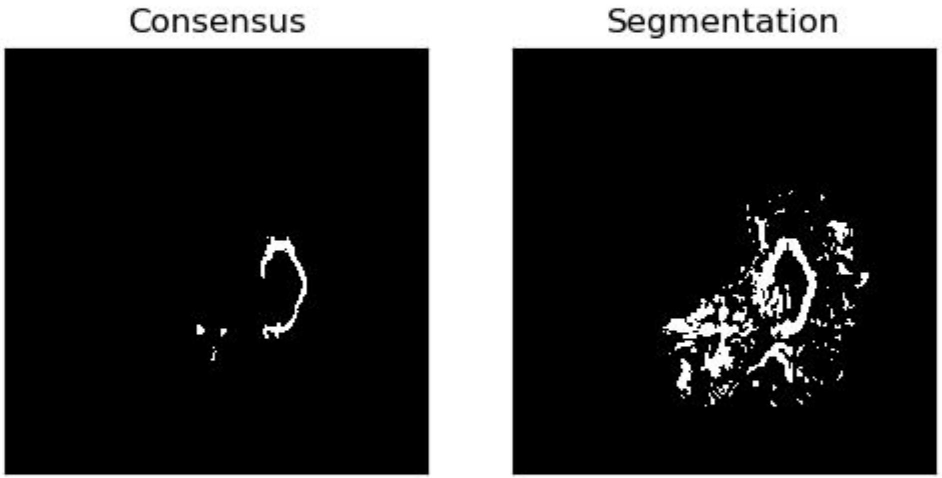

patient = 01016SACH, slice = 60, n1=0.03, n2=0.03, b=512
Consensus
Segmentation

Results

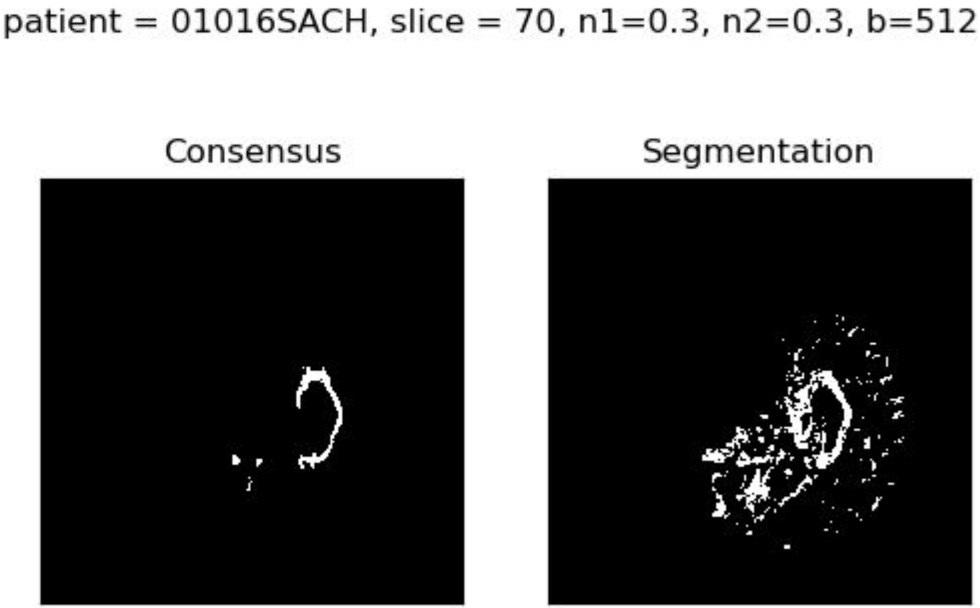

patient = 01016SACH, slice = 60, n1=0.3, n2=0.3, b=512
Consensus
Segmentation

Results

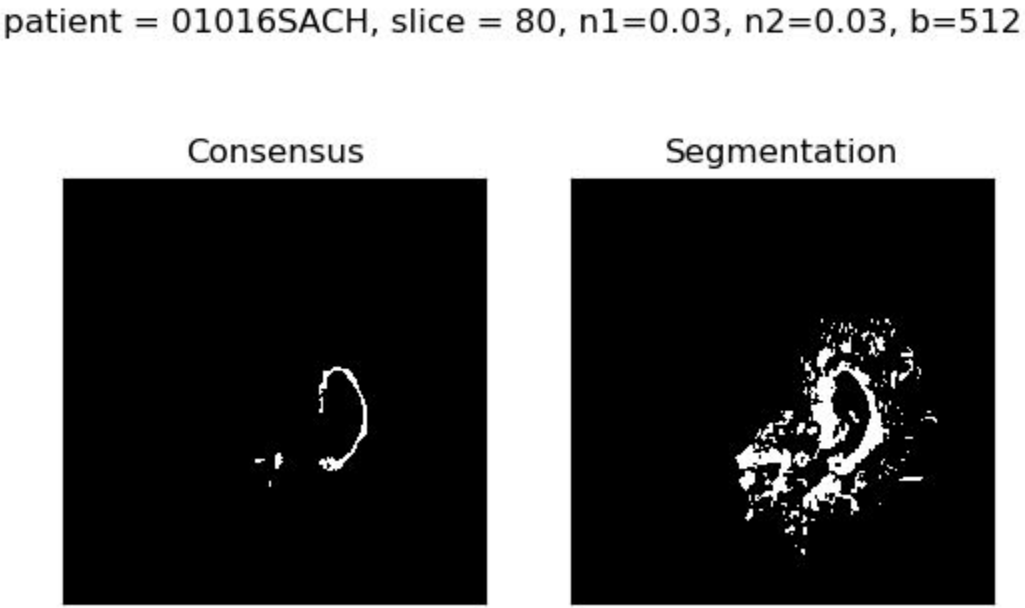
patient = 01016SACH, slice = 70, n1=0.03, n2=0.03, b=512



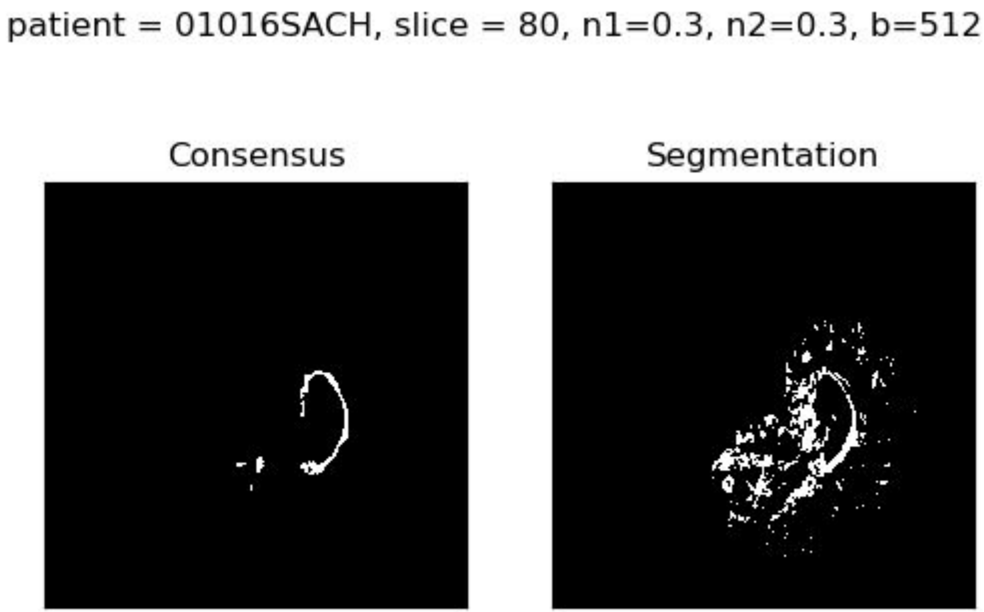
Results


patient = 01016SACH, slice = 70, n1=0.3, n2=0.3, b=512
Consensus
Segmentation

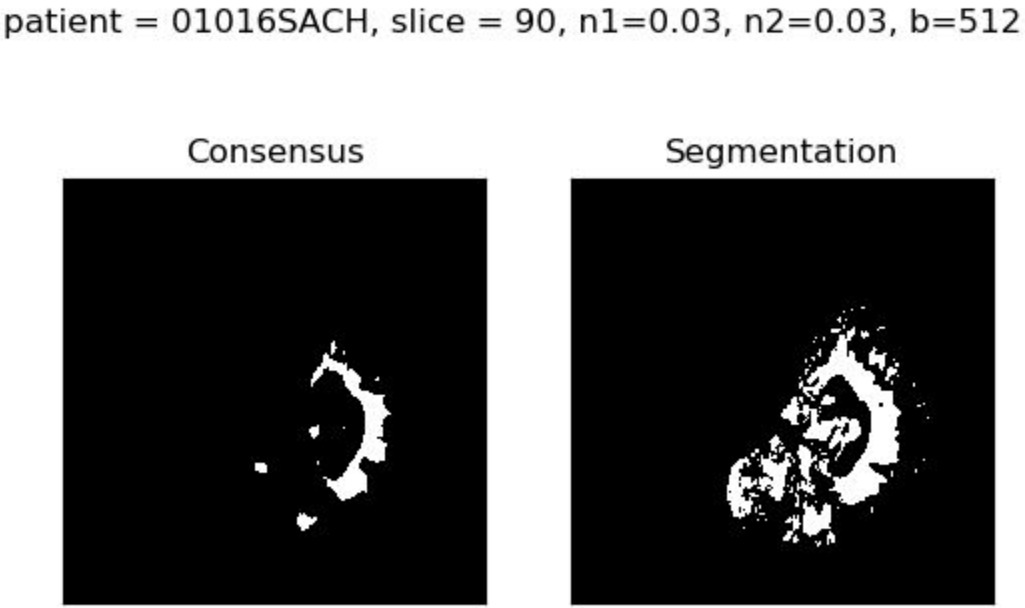
Results


patient = 01016SACH, slice = 80, n1=0.03, n2=0.03, b=512
Consensus
Segmentation

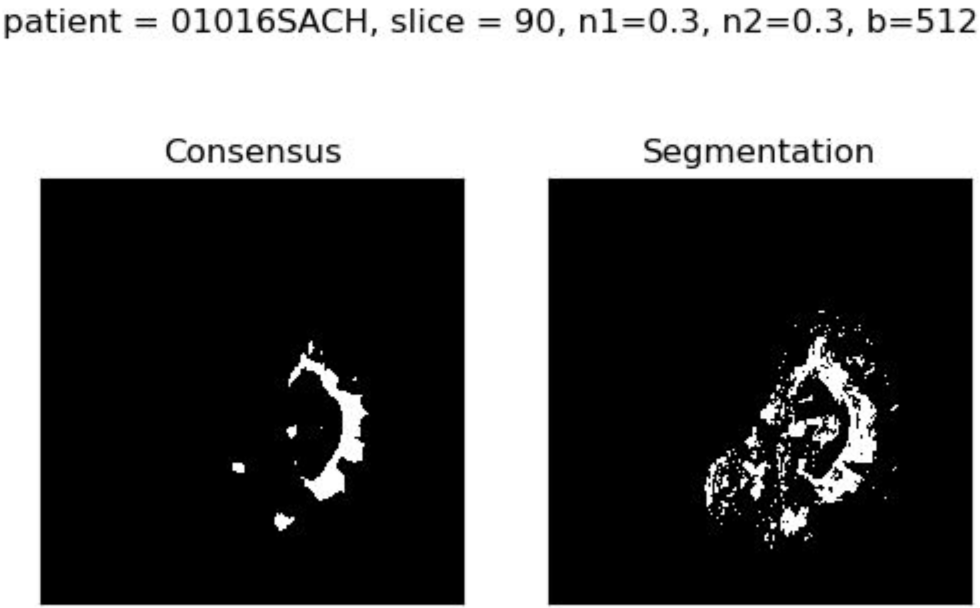
Results


patient = 01016SACH, slice = 80, n1=0.3, n2=0.3, b=512
Consensus
Segmentation

Results

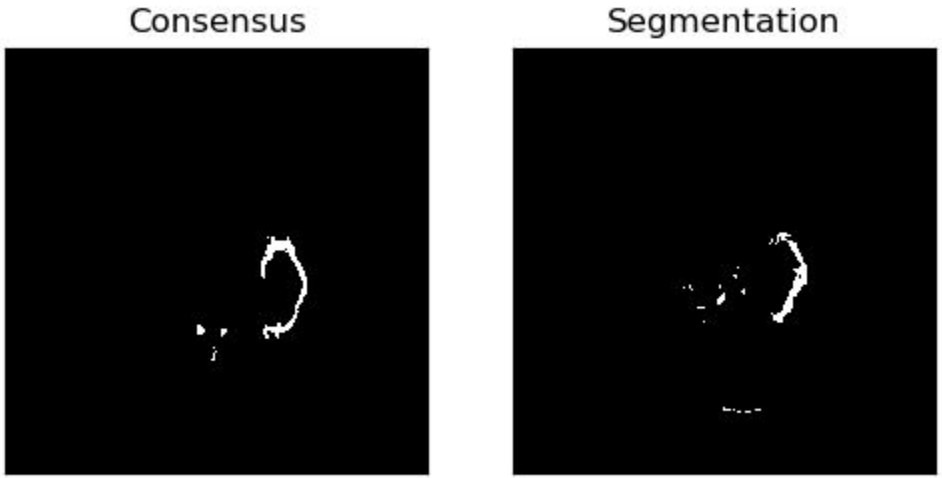

patient = 01016SACH, slice = 90, n1=0.03, n2=0.03, b=512
Consensus
Segmentation

Results

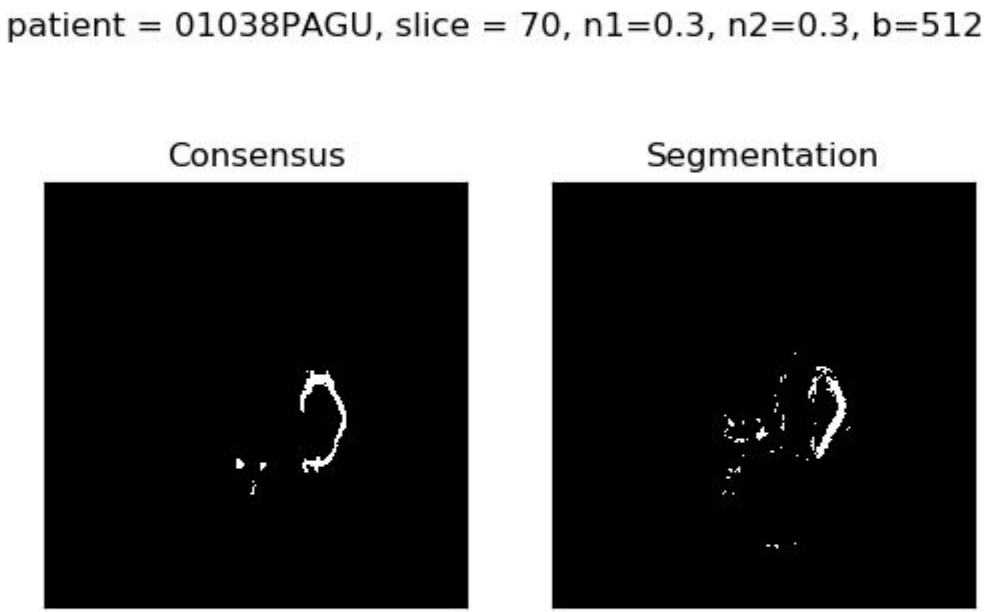

patient = 01016SACH, slice = 90, n1=0.3, n2=0.3, b=512
Consensus
Segmentation

Results

patient = 01038PAGU, slice = 70, n1=0.03, n2=0.03, b=512



Results


patient = 01038PAGU, slice = 70, n1=0.3, n2=0.3, b=512
Consensus
Segmentation

Questions?