MARIUS GADE
ADVISOR: ALVARO FERNANDEZ-QUILEZ

Robust prostate segmentation with statistical guarantees

Master thesis 2024

Computer Science – Master of Science Degree Programme

Department of Electrical Engineering and Computer Science

Faculty of Science and Technology

Universitetet i Stavanger

Abstract

The segmentation of the prostate from MRI provides valuable data to diagnose prostate cancer. Yet this task is tedious when done manually. Artificial Intelligence (AI) can take over it, such that doctors and experts can focus on other tasks. While the introduction of AI in MRI segmentations might prove useful, lack of trust in AI can pose a problem for its use. We avoid this by applying a technique to quantify uncertainty in prostate segmentation. This way, we apply guardrail measures in our predictions.

In this thesis, we will train a segmentation network called not-new U-net (nnUNet) with data from the publicly available prostateX dataset, composed of patients diagnosed with prostate cancer. Specifically, we train a single model nnUNet and a 5-fold ensemble nnUNet. We then use the predicted probabilities to train a conformal classifier. When the conformal classifier is trained, we predict new results where the model only returns predictions where it is 85%, 90%, 95% and 99% certain. The certainty level are called alphas. We apply a quality control, where the code will simulate a situation where an expert looks over the uncertain points, and classifies them correctly.

After the classifiers has been trained with the internal prostateX dataset, we externally evaluate by testing the already trained model on different datasets. The datasets are from Stavanger University Sykehus (SUS) and the publicly available P158 dataset. In this case, the datasets have both patients diagnosed with cancer and healthy patients. When both the non-conformal models and conformal models have been tested, we calculate segmentation metrics to check the results and quantify the effect of conformal prediction and the quality-control method in the external sets.

The results show that applying the conformal classifier improves the segmentation prediction. All the different alphas showcases improvement for the segmentation, both for the internal and external datasets. The segmentation gets further improved when applying quality control.

Acknowledgements

I would like to thank my supervisor, Alvaro Fernandez-Quilez for invaluable assistance during this project.

Contents

Chapter 1

Introduction

1.1 Background and Motivation

The main focus of this project is to segment the prostate given the MRI from patients at risk of prostate cancer. The main motivation to use AI for such a serious condition, is that an accurate prostate segmentation is important because the diagnosis of a patient can depend on measures extracted from it. For example, if the volume extracted from a segmentation is wrong, the patient can be misdiagnosed. Currently, there are AI systems that can segment the prostate from MRI. However, those systems don't incorporate measures to increase trust and usability by the clinician[1]. In addition, the current models may have trouble adapting to data annotated by different experts. As they may differ from how the training data was annotated.[2].

An answer to this problem is conformal prediction [3]. Conformal prediction is a technique to quantify the uncertainty of a prediction. In this project, we use this technique to obtain AI predictions with a guaranteed pre-defined error rate. For example, we only want predictions where the AI is 90% certain it is correct. This way, we can flag cases in which the AI might fail, and the expert needs to check the output.

This project is a continuation of a previous project by Nguyen et al.[4]. This thesis performs a more exhaustive external testing, as we split into patients both diagnosed with cancer and healthy. The conformal prediction gets tested with different alpha values to check where the model predicts more accurately. This project also processes the images differently, as Nguyen et al.[4] crops the images to focus on the prostate, while we run the model on the entire picture.

1.2 Objectives

The main objectives are

- O1: To train and test a nnUNet model with our 'internal data' and obtain results and predicted probabilities for a single fold case and with an ensemble.
- O2: To test the previously trained nnUNet with external datasets.
- O3: To estimate the expected calibration error (ECE) and calculate different segmentation metrics for both the single fold and ensemble using both the internal and external data.
- O4: To apply a conformal prediction with a mondrian classifier to the single nnUNet model.
- O5: To estimate the Expected Calibration Error (ECE) and calculate different segmentation metrics to quantify the effect of conformal prediction on the single nnUNet model.
- O6: To design an automatic quality-control method based on the single nnUNet model and the mondrian based conformal classifier.

1.3 Outline

In chapter 2, the background for the project is introduced together with the motivation of the project, different technologies and knowledge required to develop the project. In addition, chapter 2 contains information about related works. In chapter 3, we introduce the requirements of the project and the steps that we followed to accommodate the requirements. It will go through the process of training and validating an nnUNet using the prostateX MRI, as well as how to predict the test prostateX data. The steps to train a conformal classifier will be explained, and how to predict using the conformal classifier with confidence level 85%, 90%, 95%, 99%. We do this both with and without quality control. When everything is predicted, the metrics will be calculated for the results in chapter 4.

In chapter 4, the results of the project are presented. These results are then discussed and contextualized in Chapter 5, in relation to the previously introduced related work. Chapter 6 will conclude the thesis and mention some possible future directions of the project.

Chapter 2

Background

2.1 Clinical

2.1.1 MRI and Prostate

The prostate is a small gland that's part of the male reproductive system that is located below a males bladder and in front of their rectum[5]. The Magnetic Resonance Image (MRI) is a medical imaging technique used in radiology to form pictures of the anatomy and mechanism inside someones body[6]. MRI scanners use magnetic fields, gradients and radio waves to generate images of the organs of the body [6]. MRI is used in hospitals and clinics for medical diagnosis, cancer staging and follow up to disease[6]. Compared to other images such as computed tomography[7], MRI provides better contrast in images of soft tissue, like the brain or abdomen. An MRI of the prostate can help diagnose prostate cancer (PC)[6]. We can see an example of the MRI of an prostate and surrounding structures alongside a Ground Truth Mask (GTM) of the isolated prostate in figure 2.1.

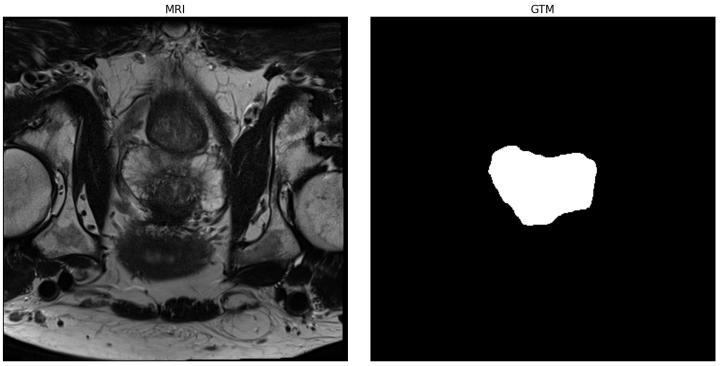


Figure 2.1: An image showcasing a MRI (left image) and the GTM highlighting its prostate (right image).

2.2 System

2.2.1 Convolutional Neural Networks (CNN)

CNN is a regularized type of feed-forward neural network[8] that learns feature engineering by itself via optimizations[9]. A CNN uses regularized weight over fewer connections[9]. Here is an example, for each neuron in a fully connected layer, 10 000 weights would be required for processing and image of size 100x100. By applying cascaded convolution kernels, only 25 neurons are required to process 5x5 sized tiles of the picture[9]. See figure 2.2 for a visualisation.

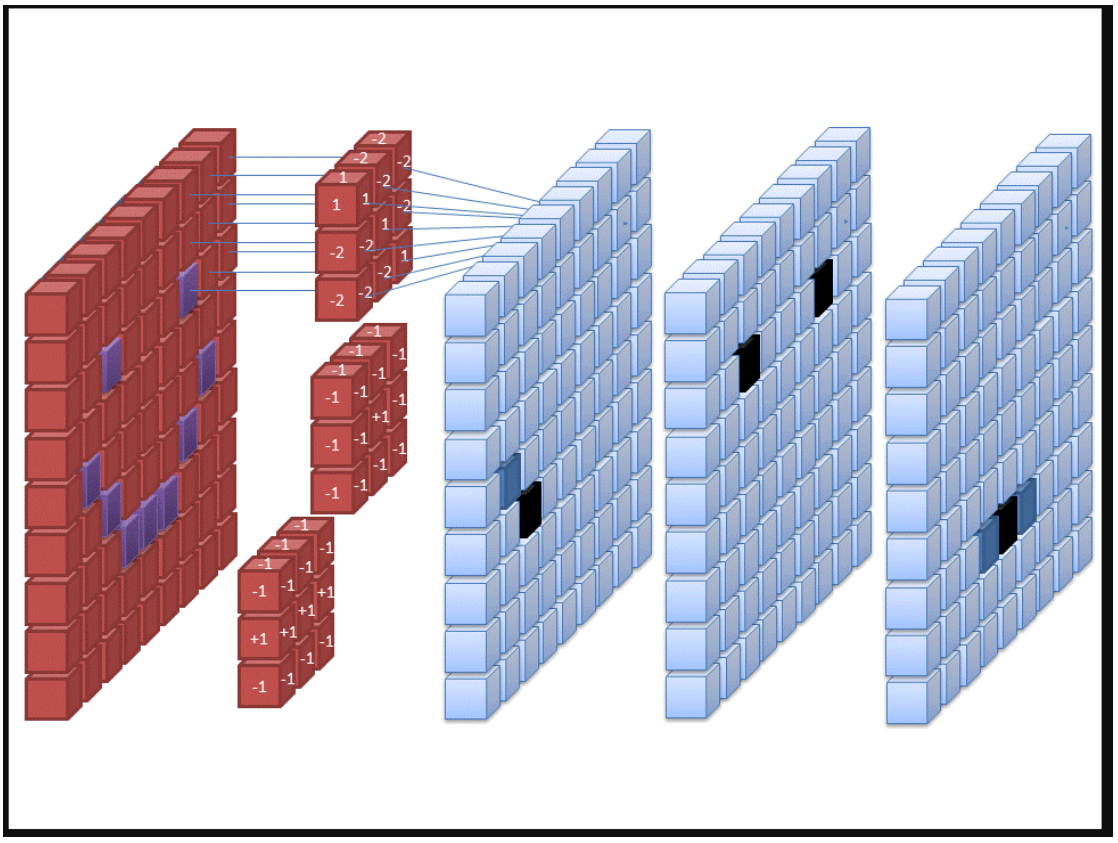


Figure 2.2: An illustration of CNN created by Cecbur from [10] that are using the Creative Commons Commons Attribution-Share Alike 4.0 International licence.

While a regular Neural network have three types of layers: Input, Hidden and Output, the CNN is an extended version, featuring an input, convolutional, pooling, dense and output[11]. The convolutional layer consists of a set of learnable filters, having small width and heights but the same depth as that of the input volume[11]. For example, if we run convolution on an 34x34x3 image, the possible size of filters are YxYx3, where Y can be 3, 5 or 7, which is smaller compared to the original image dimensions, which again reduces calculation time[11]. When we forward pass, the filters gets slip across the

whole input volume step by step where each step is called a stride[11]. As we slide our filters, a 2-D output for each filter gets created and stacked together. As a result, the output volume have a depth equal to the number of filters [11].

The pooling layer is inserted in the CNN periodically and its main function is to reduce the size of volume to make the computation faster and prevent overfitting[11].

U-Net

A U-Net is a CNN that was developed for biomedical image segmentation[12] at the University of Freiburg[13]. The U-Net architecture stems from 'fully convolution network' proposed by Long, Shelhamer and Darrel[14]. The main idea is to supplement a usual contracting network by successive layers, where the pooling layers are replaced by upsampling([15]) operators[13]. These layers increase the resolution of the output, which makes a CNN learn to assemble precise output based on the information provided[13].

In the U-Net, there is a large number of feature channels in the upsampling([15]), which is what allows the network to generate context information to the higher resolution layers[13]. This makes the expansive path symmetric to the contracting part, and yields a u-shaped architecture[13]. The contracting part is a typical convolutional network that consist of repeated application of convolutions, each followed by a rectified linear unit[16] and max pooling operations[13]. The spatial information becomes reduced while the feature information is increased[13]. The expansive pathway combines the feature and spatial information through up-convolutions and concatenations with high-resolution features from the contracting path[13].

2.2.2 no-new U-Net (nnUNet)

nnUNet is a semantic segmentation method that automatically adapts to a given dataset [17]. nnUNet offers multiple tools for an AI researcher developing segmentation methods, for example:

- A baseline algorithm to compete against [17]
- Act as a method development framework to test your contribution on a large number of datasets whiteout fine tuning every pipeline [17]
- Provides a strong starting point for further dataset-specific optimizations [17]

nnUNet is a tool built for semantic segmentation, that can handle 2D and 3D images with arbitrary input modalities. It relies on supervised learning, which means we have to provide training cases to our application for it to work. nnUNet will then analyze the provided training cases and create a 'fingerprint'[18]. nnUNet then creates several U-Net configurations for each dataset.

The nnUNet can either create a 2D U-net, a 3D U-Net that operates on high image resolution or a 3D U-Net cascade. The nnUNet configures its segmentation pipelines based on a three step-recipe [18]:

- Fixed Parameters
- Rule-based parameters
- Empirical parameters

2.2.3 Conformal prediction

Conformal prediction is a technique for quantifying uncertainty in AI systems[3]. Conformal predictions, given an input will estimate a prediction interval for regression system, or a set of classes for classification systems[3].

The steps the conformal prediction takes is as follows. First, we identify an appropriate score function to measure the discrepancy between model output ỹ and label y[3]. In terms of our classification problem, a good score function might be $1 - y_I$ where y_I is the predicted logits for the true classic[3]. This way, the predicted logits will be greater than some threshold[3].

Then we compute ẽ as the (1-a) quantile of the scores $(s_1, ..., s_n)$ where $s_n = (x_n, y_n)$[3]. In the full conformal prediction method, we train m models, where m is the number of possible values Y_{n+1} could take[19]. We then use model prediction and the (1-a) quantile to construct a prediction set[3].

Conformal predictors has the advantage that it is mathematically guaranteed to provide valid predictions, when new examples are independent and identically distributed to the training examples[20]. In comparison to an ensemble, conformal prediction is a prediction built upon statistical guarantees. In simpler terms, an ensemble will always predict one class, even if it is uncertain if it is correct. Conformal will return none or multiple predictions if it is uncertain.

2.3 Environment

2.3.1 Unix cluster and SSH

A unix cluster is a system that interconnects two or more computers using additional network and software technology to make a single virtual or logical server[21]. We can compute bigger dataset and process more information on a much larger scale using this method.

Slurm is a resource manager and job scheduler, which allows us to queue up for resources from the unix cluster and use them for our jobs.[22].

2.3.2 Anaconda/Miniconda

Miniconda is a small bootstrap version of anaconda[23]. Anaconda is a distribution of the python and R programming languages for scientific computing, that aims to simplify package management and deployment[24]. The package versions are managed by conda, which allows users to install different versions of binary software and the required dependencies[25]. Conda also allows for version limitations, so the versions doesn't change with the overall python environment outside the conda system[25].

2.4 Segmentation metrics

2.4.1 Dice similarity coefficient

The Dice similarity coefficient (DCS) or dice score, is a method which measures the similarity between two datasets, usually binary datasets[26]. With segmentation's, the dice score can be used to evaluate similarities between the predicted segmentation and a GTM[26]. It ranges from 0, which means no overlap, to 1, which means perfect overlap[26].

The dice score is calculated with the following formula

$$DSC = 2 * X/(Y + Z)$$

where X is the number of common elements in the two sets, Y is the number of elements in the predicted segmentation's and Z is the number of elements in the GTM segmentation's[26].

2.4.2 Hausdorff distance

The Hausdorff distance (HS) measures how far two subsets of a metric space are from each other[27]. Two sets are close in the Hausdorff distance if every point of both set A is close to some other point in set B[27]. The Hausdorff distance represents the furthest distance a point in one subset have to another point in the other subset[27].

The Hausdorff distance is calculated with the following formula

$$Hd(X,Y) = max\{sup(d(x,Y)), sup(d(Xmy))\}$$

Where sup is the supremum operator[28], $d = inf(d(a,b))$ which quantifies the distance between a point a in subset X to the set B which is a subset or equal to X[27]. Inf is the infimum operator[28].

In summary in use of segmentations, the Hausdorff distance is the maximum distance between a positive point in the predicted segmentations, to any point in the GTM segmentations sharing the same label.

2.4.3 Average Surface distance

The average surface distance (ASD) is the average of all distance from points of boundary in the segmentations to the boundary of the ground truth, as well as the opposite direction[29].

2.4.4 Relative Volume Difference

The relative volume difference (RVD) measures the absolute size difference of the regions, as fractions of the size of the reference[29].

2.4.5 Absolute volume Difference

The absolute volume difference (AVD) is the total volume of the prediction divided by the total volume of the GTM.

$$AVD(X,Y) = Total\_volume\_X/Total\_volume\_Y$$

The formula above is when X is the predicted segmentation and Y is the GTM segmentation.

2.5 Uncertainty metrics

2.5.1 Brier loss score

The brier loss score is a scoring rule that measures the accuracy of predicted probabilities[30]. The brier loss score can be considered a cost function, as it measures the mean squared error between the predicted probabilities of the different points and the actual points[30]. The lower the brier score is, the better the calibration[30].

2.5.2 Validity and efficiency

Validity and efficiency quantifies the quality for the uncertainty prediction. The formula to calculate these are

$$Validity = correct\_preds/total\_preds$$

$$Efficiency = Single\_preds/total\_preds$$

2.5.3 Expected calibration error and calibration curve

The Excpected Calibartion Error (ECE) is a measure of how well a model's predicted probabilities are[31]. The ECE are calculated by taking the weighted average over the absolute difference between accuracy and confidence over M-bins[31].

The calibration curve or reliability diagram is a plot that showcases the fraction of positive classes over the mean of the predicted probabilities[19].

2.6 Related works

2.6.1 Estimating diagnostic uncertainty in artificial intelligence assisted pathology using conformal prediction

The study "Estimating diagnostic uncertainty in artificial intelligence assisted pathology using conformal prediction" by Olsson et al. [20] is a study that very closely resembles our own. The main objective of the study is to showcase the usage of conformal prediction by demonstrating how it changes the results of the AI's predictions.

They used 7788 prostate biopsies from 1192 men in the STHLM3 study for training, and 3059 biopsies from 676 men for testing. The study uses the training data to train the

model. Then creates 6 different test set to evaluate the system and conformal prediction. These test set were created to split the test data into sets containing different attributes, with the first two test sets coming from the same scanner and laboratory, the third test set comes from an external scanner, and the fourth and fifth test sets coming from two different scanners and laboratories, and test set 6 showing unusual morphological patterns that the model have limited or no previous exposure to. By differentiating in scanners and labs, the study will show the uncertainty of a model when it encounters data it isn't accustomed to, and how the use of conformal prediction might help reduce the errors created by unknown attributes to the data.

The first test set were used to detecting cancer and give it a ISUP grading. The set were being used to show the difference in predictions, when using conformal prediction and not, and how it changes depending on the confidence level. With a confidence level of 99% the system only categorized on cancer case of being benign, and giving us a 78% overall efficiency. This means that 78% of the predictions were single predictions, while 22% where multiple predictions, which become flagged for human intervention as it is uncertain. In comparison, the system without conformal predictions committed 14 errors of cancer detection. In the study, they also experimented with two other confidence levels. The results at a confidence level of 67% was 98 errors were committed, 10 empty predictions and 72 multiple predictions, with 174 correct predictions. At a confidence level of 80%, there were 62 errors, 7 empty and 153 multiple, so an decrease in errors and empty predictions but an increase in predictions needing human assistance. In comparison to the predictions with CP, the AI without CP committed 117 errors for ISUP grading. The study also shows that it is possible with a mix confidence level, the previous tests showed the CP produced a higher error rate for ISUP grading 1. With this information, Olsson et al. [20] set the confidence level to 85% for ISUP 1 and 67% for ISUP 2-5. The systems predictions then consist of 20% errors and an overall efficiency at 47%.

The study also shows how predictions alter when using segmentations from different equipment than the model was trained for. The study shows that using different equipment will make the CP act different, granting a higher error rate all in all.

With test set 6, Olsson et al. [20] have multiple different different mimics of prostate cancer and sub-types. This is meant to show the CP will be able to make predictions with low error rate, when it hasn't seen it before, but at the cost of more empty and multiple predictions, creating a dynamic where the CP predicts clear cut cases with high reliability, while experts work on more challenging cases.

Compared to our study, the study conducted by Olsson et al. [20] is about the effectiveness of CP, similar to this work. The main differences come in the way we approach the task

and the application area. The underlying model we use will be different from how Olsson et al. [20] did it. We will use nnUNet as the model, while they used two ensemble of convolutional deep neural networks. The CP implementation won't be to different and both our studies use a mondrian classifier. While they use CP for pathology, we will use it with MRI. Our project also looks at prostate volume, not if the patients have cancer.

2.6.2 Revisiting prostate segmentation in magnetic resonance imaging (MRI): On model transferability, degradation and PI-RADS adherence

In the study by Fernandez-Quilez et al. [32], they want to investigate the effect of scanner and prostate MRI acquisition when compared to PI-RADSv2.1 technical standards in the performance of a deep learning prostate segmentation model. The model is trained with data from one center, longitudinally evaluated at the same institution and when transferred to other institutions[32].

In the study, Fernandez-Quilez et al. [32] used nnUNet for prostate MRI segmentation, trained using data from the prostateX dataset. It was trained using 204 patients and tested with 30 patients, using a different protocol in a different time[32]. To test the transferability of the model, they used 248 patients sequences from 5 different institutions, acquired with different scanners[32].

The results showed that the model presented a significant degradation for the whole segmentation[32]. It had a significant higher performance in centers adhering to PI-RADS v2.1 when compared to those that did not[32].

This paper formed the starting point of this thesis, as one of the main contributors for this paper is my advisor. Therefore, much of the file transferring and nnUNet training have inspiration from this. Our thesis also allows to modernize some of the aspects of the previous thesis, like upgrading from nnUNetv1 to nnUNetv2. In addition, we explore the effect of uncertainty to mitigate dataset differences, as explored in the external evaluation.

2.6.3 Uncertainty quantification in prostate segmentation

The thesis by Ngyuen et al.[4], builds upon the paper by Fernandez-Quilez et al. [32] by quantifying uncertainty in the predictions. He does this by adding a conformal classifier, and evaluates it through dice scores an relative volume distance, and compare these values to before the conformal classifier was applied [4]. The data that Ngyuen et al.[4] used is MRI prostate from prostateX, SUS and patient158.

Ngyuen et al.[4] resized his images to be the same, in case any images was different in size. He then uses the prostateX results from the nnUNet and splits them into a test and training set. He then crops the training images to just focus on the part of the image that has the prostate[4]. When he finished training the model, he used it on both the test set for prostateX, and the both the Stavanger University Sykehus (SUS)[32] and patient158[33] datasets[4]. The results shows that the validity and efficiency for the model achieves percentages ranging from 94.24% to 99.34%.

This thesis is a direct continuation of Ngyuen et al's.[4] thesis, where we create the nnUNet from scratch and use the results from a single model nnUNet in a conformal classifier, while Ngyuen et al.[4] didn't create nor train the nnUNet. This results in more control of the pipeline, as we know exactly what happens during the process. The model will also be trained with the full image instead of cropping it, and are testing with different alpha values. The external data is split by characteristic, which Nguyen et al.[4] did not do during his thesis, testing to see if the segmentation predictions changes with different characteristics.

Chapter 3

Data and methods

3.1 Dataset

The dataset is a set of patients who are being checked for prostate cancer using MRI(2.1.1). The patients are split between two main folders. The internal data is used for training and testing, while the external data is used for external evaluation of our model.

The nnUNet works with a specific folder structure, where we define a path to the raw data, pre-processed data and results[34]. The raw data folder will include a different folder called Dataset**ID** where the ID can be any number, in this project it is 101, showcased in figure 3.1. The nnUNet will the be able to find the right data, by using this folder ID to configure the dataset correctly. The nnUNet will then mostly do the rest by itself, with some configurations you can give the jobs.

The nnUNet does different common steps in an automatic way, like pre-processing and post-processing. The only step prior to training is to split the data into a mapping structure the nnUNet understands, so it can find train, test and true labels.

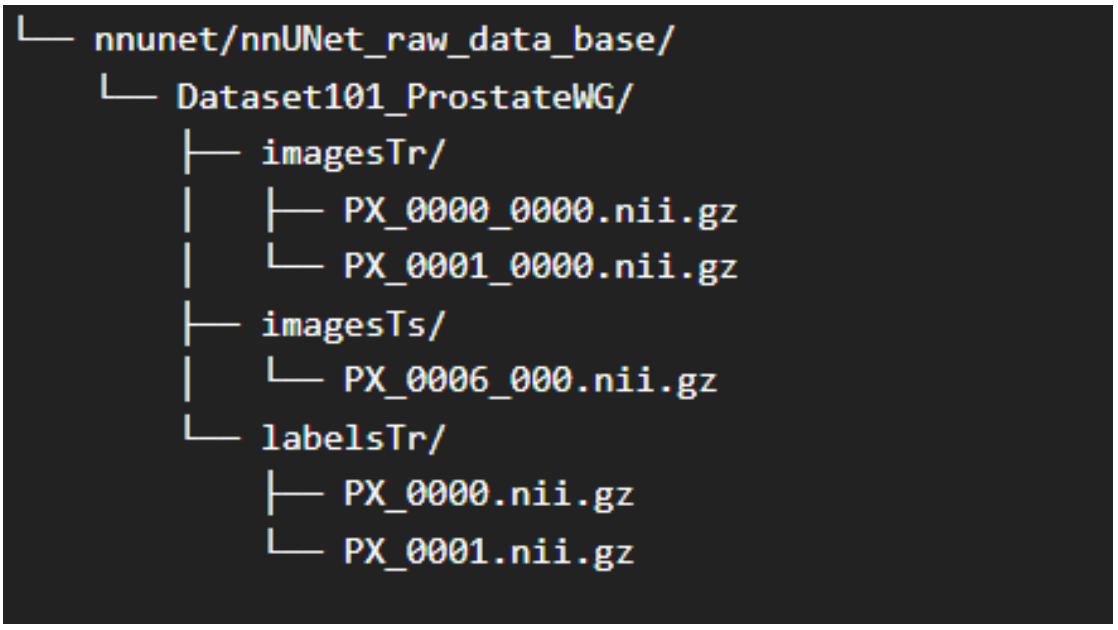


Figure 3.1: An example of how to structure the folders. Created using tree.nathanfriend.io.

3.1.1 Internal data

The internal dataset is the backbone of the entire project, as this data will be used to train both the nnUNet and the conformal classifier. The internal data is solely made up by the prostateX dataset. The prostateX dataset is set of prostate MRI studies, all including T2-weighted (T2w), proton density-weighted (PD-W), dynamic contrast enhanced (DCE) and diffusion weighted (DW) imaging [35]. The images were acquired on two different type of Siemens 3T MR scanners, the MAGNETOM Trio and Skyar[35].

In this work, we focus on T2w images, which are acquired using a turbo spin echo sequence and have a resolution of around 0.5 mm in plane and a slice thickness of 3.6 mm[35]. The prostate MRI was performed at the Radbound University Medical Center in the prostate MR Reference Center under supervision of Professor Dr. Barentsz in Nijmegen, Netherlands[35]. The dataset was collected and curated for research in computer aided diagnosis of prostate MR under supervision of Dr. Huisman at Radbound University Medical Center[35]. The dataset contains 204 patients. Some data characteristics can be seen at table 3.1. In the characteristics datasets, we assume that all the patients counted for in with/without significant cancer, have been diagnosed with cancer.

Status	Count
Patients under or equal 35ml	10 (4.902%)
Patients between 34 and 50ml	36 (17.6471%)
Patients over or equal 50ml	158 (77.451%)
Patients with significant cancer	65 (32.5%)
Patients without significant cancer	135 (67.5%)

Table 3.1: Table for ProstateX characteristics. Table shows data for patients that had clinical information available at the time of the study.

3.1.2 External dataset

The external dataset is split up into two different folders representing different methods of getting the MRI. The external data are from Stavanger University Sykehus (SUS)[32] and from a publicly available dataset Prostate158 (P158)[33]. These datasets are exclusively being used as test data, to see how our pipeline will work with data that have different origins compared to the internal training data. Both of the datasets are split between patients with cancer and healthy patients, to test for differences in regards to clinical characteristics for the patients.

SUS

The MRI from SUS were acquired with an in-plane resolution of 0.5mm x 0.5mm and a slice thickness of 3.0mm[4]. The SUS folder are split into 41 patients with cancer, and 7 healthy patients. Some data characteristic can be seen at table 3.2

Status	Count
Patients under or equal 35ml	6 (12.5%)
Patients between 34 and 50ml	11 (22.9167%)
Patients over or equal 50ml	29 (60.4167%)
Patients with significant cancer	16 (39.0244%)
Patients without significant cancer	25 (60.976%)
People with positive cancer	41 (85.4167%)
People with negative cancer	7 (14.5833%)

Table 3.2: Table for SUS characteristics.

P158

The P158 dataset consists of expert annotated biparametric 3T prostate MRI comprising of T2w sequences with apparent diffusion coefficient maps[33]. The P158 patients are split between 83 patients diagnosed with cancer and 56 healthy patients. In this work, we only use the T2w. Some data characteristic can be seen at table 3.3.

Status	Count
Patients under or equal 35ml	1 (0.7194%)
Patients between 34 and 50ml	12 (8.6331&)
Patients over or equal 50ml	126 (90.6475%)
Patients with significant cancer	83 (100%)
Patients without significant cancer	0 (0%)
People with positive cancer	83 (59.7122%)
People with negative cancer	56 (40.2878%)

Table 3.3: Table for P158 characteristics.

3.2 Methods

Training and queuing jobs for deep learning models

A nnUNet is a rather computationally expensive model to train, for that reason, we use the unix cluster system that is part of UiS, where the slurm system is used to queue jobs for the cluster. The gpu's on the unix clusters consists of eight 40 GB Nvidia Tesla A100, that is divided equally between two different machines[22].

We use SSH to connect to the machine ssh1 as a proxy, and then connect to gorina11 from ssh1. The gorina11 uses the slurm system to reserve resource that are needed for training the model. By creating a sh file, the sbatch command can be used to reserve GPU's, activate virtual environments and run python scripts. If there aren't enough resources available, the sbatch command will queue up the request, such that when the resources become available, the model can start training. The sbatch allows us to leave the terminal as well. The model can then be trained overnight, if it takes a long time to train. A flowchart of the running process can be see at figure 3.2.

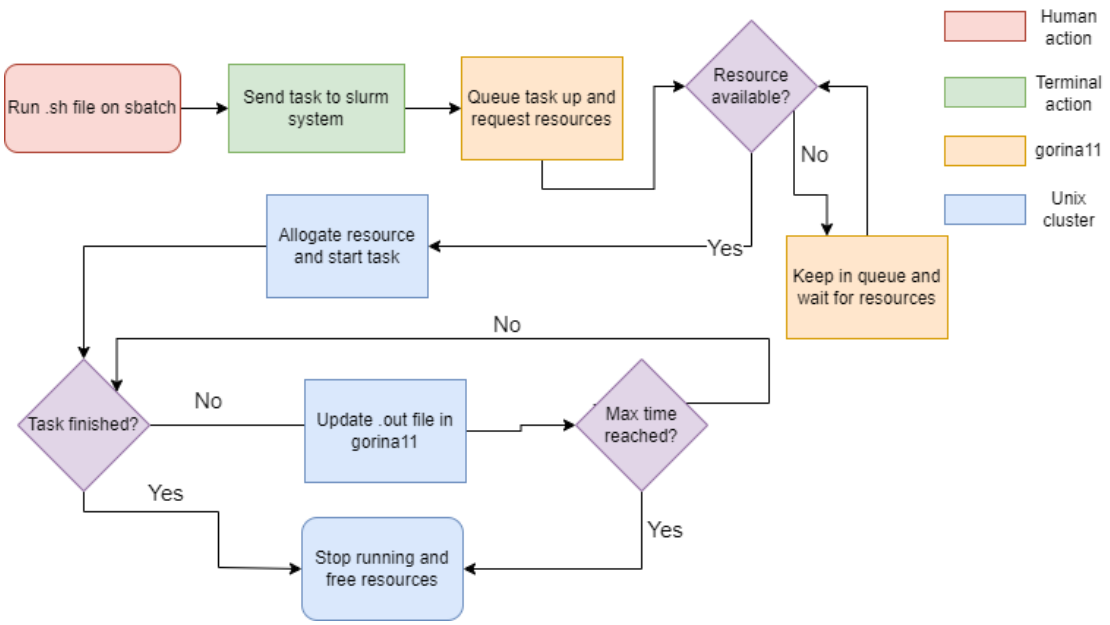


Figure 3.2: Flowchart of the slurm process.

The UiS unix cluster is available to every student at UiS. As such, the main environment on the machine can change at anytime. For example, updating python and libraries to new versions, thus making code that uses previous versions not work. To make sure that the script runs smoothly during the duration of the study, a virtual environment on the unix cluster is needed. The virtual environment gets generated and managed using Miniconda[23]. Miniconda is a free minimal installer for conda that comes with python and the packages python and conda depends on[23]. By creating a virtual python environment with miniconda, packages and libraries needed to get the scripts running can be downloaded and remain unchanged. Even if the main environments for the unix cluster updates and changes, with the virtual environment, the script will always run on the same configurations and system versions.

3.2.1 Part 1: Segmentation by nnUNet

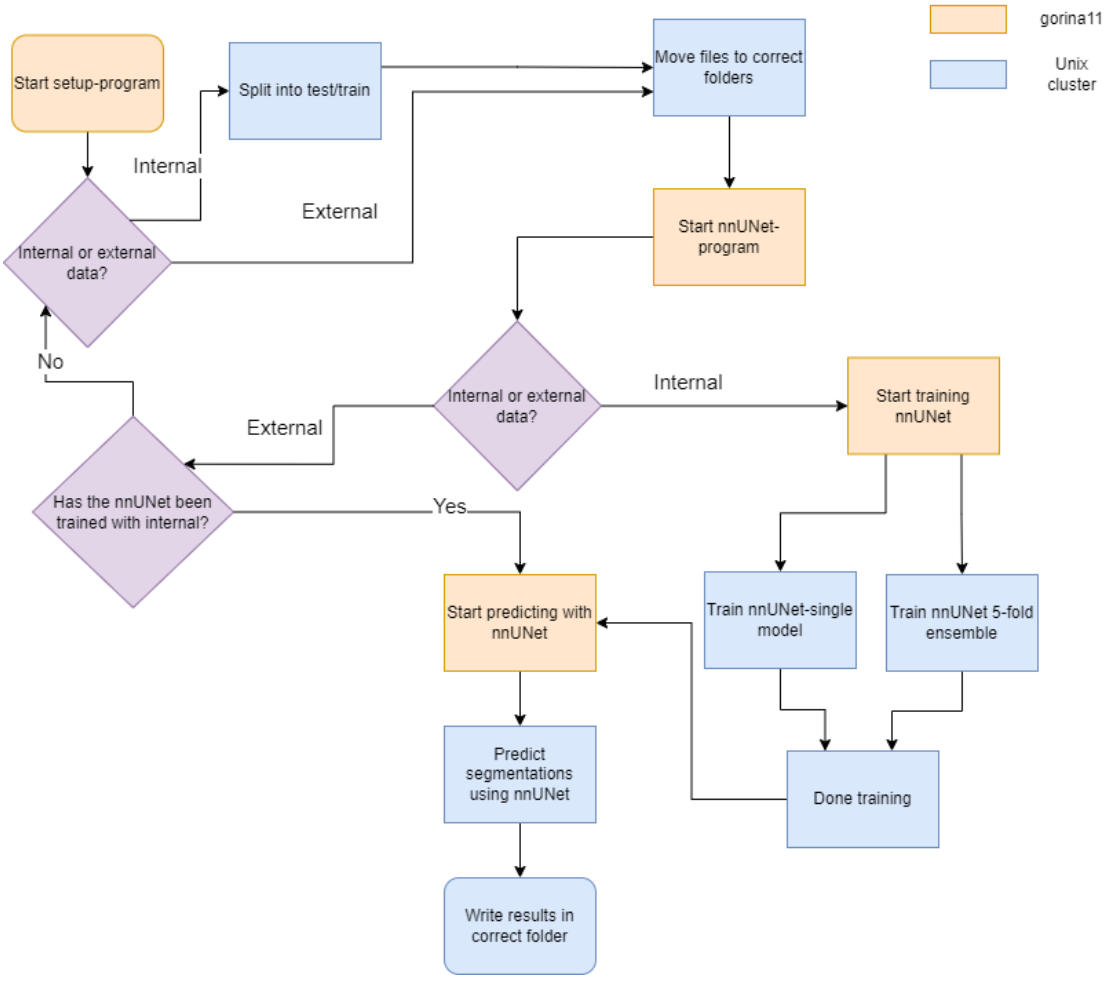


Figure 3.3: Flowchart of the nnUNet process.

The first step of the project is to train a segmentation network based on the internal dataset, using nnUNetv2. At the start, the datasets is split into internal and external, with each containing the prostate MRI, and the mask of the prostate gland (see figure 2.1), each in a separate folder in the internal and external folder.

The first task is to develop the code to create folders and move the data into these folders. The code will create the raw data folder, pre-processed folder and a results folder. The data is then split into test and train, and put into different folders called imagesTr and imagesTs in the raw data folder. The GTM of the segmentation will be necessary, so it moves the GTM of the training data in a folder called labelsTr. The current folder structure was created to be used by nnUNetv1, so the folder setup won't work for nnUNetv2. nnUNetv2 has a function called convert_old_nnUNet_dataset, which transforms the folder structure into something that can work for nnUNetv2.

The process will start with using nnUNet's plan and pre-processing method, which takes the data from the raw_data folder, and goes through nnUNet's pre-processing. When it is finished, it will put the pre-processed image in the proprocessed folder. When the pre-processing is done, the training of the nnUNet can start.

The nnUNet is trained with 5 fold cross-validation. The nnUNet uses 1000 epochs for each fold, to make it more accurate. When all folds have been trained, the predict method will be used on the test data. In terms of the project, the nnUNet model will predict whether a pixel in the MRI is a prostate or not. When the model predicts using all 5 folds, an ensemble will be used to compare the 5 results, and return one result from the 5-fold ensemble. To compare, a single model will also be trained, where all the data gets used to train 1 model, instead of spreading it into 5 models for an ensemble. The conformal classifier will be trained on the single model predicted probabilities later in the project, where it will predict the same test segmentations. Later, when using the conformal classifier, the results will be compared by calculating the metrics for all models and datasets. The whole process is visualized in figure 3.3.

3.2.2 Part 2: Conformal prediction

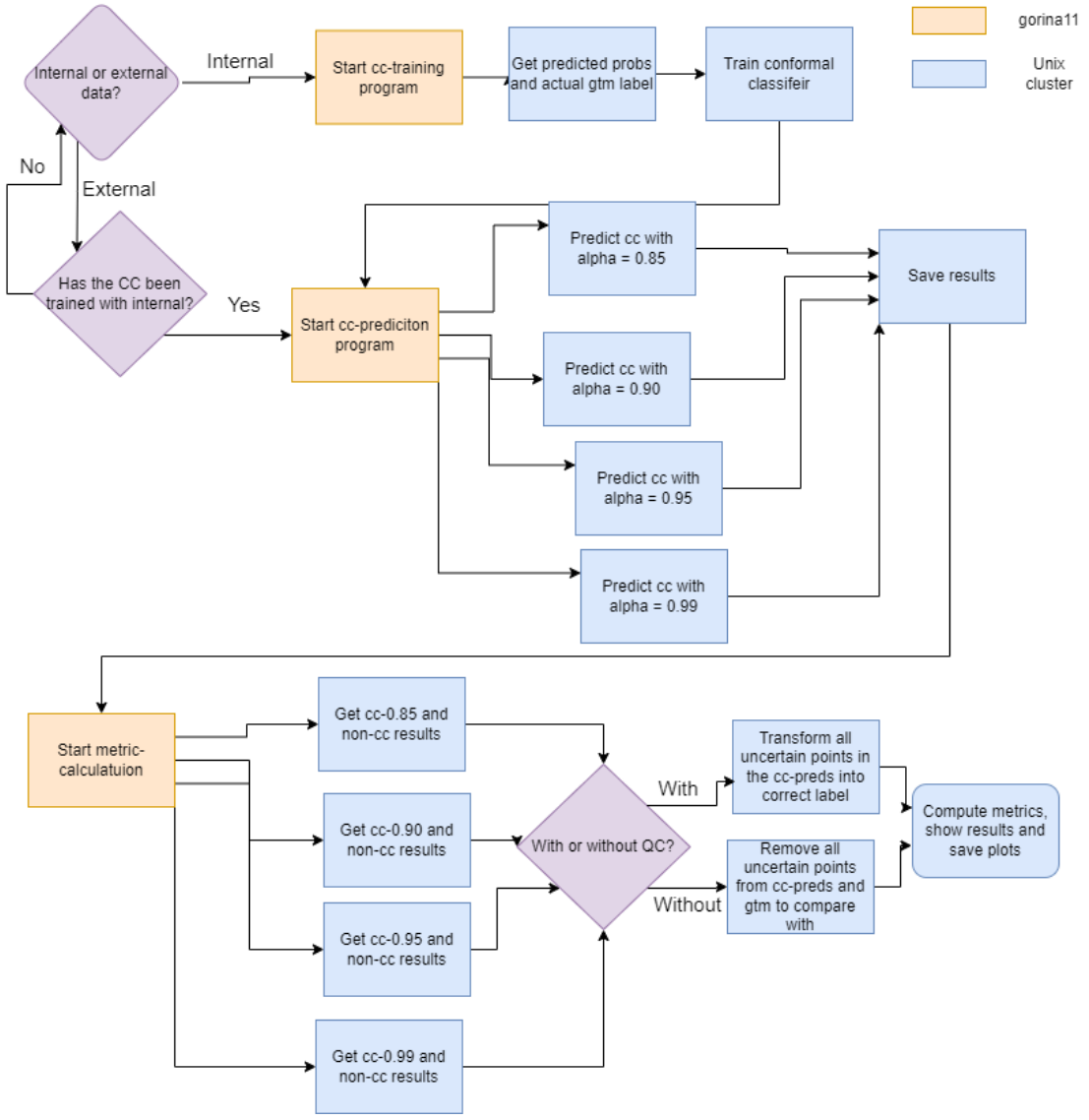


Figure 3.4: Flowchart of the CC process.

Once the nnUNet is trained, we obtain the predicted probabilities for the training set (validation) and for the results. The conformal classifier uses the validation predicted probabilities from the single-model to train.

Before training the conformal classifier, the data needs to be in the correct shape. Instead of having a 3-dimensional array, a 1-dimensional list is required, so the predicted probabilities gets flattened into a list where each element is [prob of value being 0, prob of value being 1] and a list containing the GTM's, where each element is the true label of the segmentation, either 0 or 1. These two list and a list of the model classes, which here is just [0, 1], is utilized to create our alphas/loss function by using the hinge function from crepes library[36]. The hinge function computes the non-conformity

score for the conformal classifier. Non-conformity score measures how much each record doesn't conform with the rest of the data[37]. The hinge function uses 1 - predicted probability for the positive label class (1) to compute the non-conformity score, so that if the predicted probability for the correct class is 1, the non-conformity score would be 0.

When applied to a model of our choice, which here is the nnUNet, conformal prediction will only return predictions that the model is a percentage confident are correct. The conformal prediction will return 4 different cases of predictions, either [1, 0] where it isn't a prostate gland or [0, 1] where it is a prostate gland. The last two cases are called multiple predictions ([1, 1]) and empty predictions ([0, 0]), these types of predictions happen when the model is not certain whether the point is a prostate gland or not. When this happens, in a real life scenario, an expert would be called to check the prostate MRI. This creates a dynamic interaction environment where the nnUNet will predict more common cases, while the experts can focus on the real difficult ones.

When the conformal classifier is trained, the performance is evaluated. For each confidence level (alpha = 0.85, alpha = 0.90, alpha = 0.95, alpha = 0.99) we obtain the dice score, total volume, Hausdorff distance, global surface distance, relative volume distance, absolute volume distance, brier score, efficiency and validity of the conformal classifier (cc) and the non conformal classifier (no-cc) models (single nnUNet and 5-fold ensemble).

Metric calculations

When calculating the metrics, we do not account for the points where the classifier is uncertain. New lists are created where these points and the correspondent point from the GTM list are removed. This way, only the points the GTM is certain of is calculated. The only difference is how this is done with the Hausdorff distance and the global surface distance, because of how the functions from medpy work[38]. Medpy is a library for medical image processing in python. Instead of removing them completely, the uncertain predictions and their correspondent true label are turned into 0, this way, the shape stays the same, but the uncertain points doesn't add any difference to the calculation. For this study, we use the ConformalClassifier from the library crepes[36]. A visualisation of the training of the conformal classifier can be seen in figure 3.4.

3.2.3 Part 3: External evaluation

The external datasets is structured differently from the internal dataset. The external dataset is split into two different facilities: Stavanger University Hospital (SUS)[32] and P158[33]. Each of these also splits the data into patients that have cancer and patients

that doesn't have cancer. Inside each patient folder, there are two folders, one for the T2w data and one for the GTM, instead of the entire dataset being split into data and GTM. A new nnUNet model won't be trained for the external evaluation, but will be tested on the model trained with the internal dataset. It is the same situation with conformal classification. A new conformal classifier will not be trained with the external data, but the same one used in part 2. The only difference is that the labels in the P158 GTM must change. The labels for P158 is by default [0, 1, 2] where 1 and 2 represents different zones of the prostate. The volume and size of the prostate is what's necessary for the project. With this in mind, every value that is 2 in the GTM gets changed to 1. When the predicted segmentations and GTM have the correct format for the project, the metrics gets calculated. When this is done, the results will show how good the models are with external evaluation. The visualisation for this process can be seen in figure 3.4.

3.2.4 Part 4: Quality control

The quality control in this project simulates when an expert checks over the predicted segmentations, that is predicted by the conformal classifier and corrects it. Whenever the code encounters a point which the conformal classifier is uncertain (empty/multiple predictions) during the calculation of the metrics, the code will look at which class the label is from the GTM and apply the correct label to the point.

When quality control is applied, the calculation of the metrics becomes much simpler. As the images doesn't change shape as with non-quality control, as all points in the image gets included, the data can just be put in the functions to calculate the metrics. There are some functions where flattening the lists are needed, but it doesn't get more complicated than that. The visualisation for this can be seen in figure 3.4.

Chapter 4

Results

To compare the different model states, from ensemble, single model, conformal classifier and quality control, different metrics will be calculated. The main metrics used is DSC, HD, ASD, RVD and AVD to check segmentations. For the uncertainty quantification, the metrics used are ECE, calibration curve, brier loss score, efficiency and validity to check calibrations. While the ensemble will only be used once, we will see improvements in the single model when adding the conformal prediction method.

4.1 Experimental Results

The results will show how specific metrics change depending on the model that is used. The goal for this is to see the improvements from adding conformal to the single model, and to see how it compares to the ensemble model.

The results will be shown in the form of tables and plots. The tables will be split into internal dataset and external evaluations. Each dataset has two tables, one for the segmentation's and one for the conformal as well as multiple plots.

The images at 4.1, 4.4, 4.7, 4.10 and 4.13 have three colors. These colors represents what the classifier has predicted the point to be. The colors are black = [1, 0] for not prostate, white = [0, 1] for prostate and red = [0, 0] where it is uncertain. The classifier can also predict [1, 1], but in our cases, the model did not predict uncertainties as [1, 1].

4.1.1 Internal evaluation

Model	DSC	HD	ASD	RVD	AVD
5-fold ensemble	0.8198 ± 0.0593	7.7628 ± 4.8219	0.5734 ± 0.1661	1.73 ± 9.02	99.07 ± 8.82
Single model	0.8197 ± 0.0619	7.8389 ± 4.8219	0.5915 ± 0.1723	1.56 ± 9.03	99.24 ± 8.83
CC-85%	0.9064 ± 0.0576	9.0660 ± 5.8333	0.1318 ± 0.1138	-1.41 ± 4.01	101.59 ± 4.15
CC-90%	0.9027 ± 0.0595	9.033 ± 5.828	0.1394 ± 0.1194	-1.32 ± 4.09	101.52 ± 4.23
CC-95%	0.8990 ± 0.0585	8.8286 ± 5.7431	0.1588 ± 0.1270	-1.51 ± 4.33	101.73 ± 4.51
CC-99%	0.8712 ± 0.0641	8.0996 ± 5.1566	0.3127 ± 0.1921	-4.28 ± 5.65	104.83 ± 6.33
Quality control-85%	0.9523 ± 0.0516	5.1682 ± 3.3636	0.1379 ± 0.0997	-1.33 ± 3.46	101.48 ± 3.59
Quality control-90%	0.9505 ± 0.0509	5.264 ± 3.407	0.1438 ± 0.0993	-1.27 ± 3.55	101.420 ± 3.670
Quality control-95%	0.9456 ± 0.0500	5.4798 ± 3.4246	0.1649 ± 0.1056	-1.46 ± 3.81	101.63 ± 3.97
Quality control-99%	0.9097 ± 0.0576	6.1474 ± 3.8590	0.3166 ± 0.1732	-4.10 ± 5.34	104.60 ± 5.97

Table 4.1: Results for ProstateX segmentations. Results are presented as $Mean \pm SD$

Model	Validity	Efficiency	Brier loss score	ECE
5-fold ensemble	-	-	0.40 ± 0.19	0.00394563
Single model	-	-	0.42 ± 0.18	0.00401208
CC-85%	50.63 ± 0.22	50.70 ± 0.24	0.12 ± 0.09	0.00142969
CC-90%	50.60 ± 0.24	50.66 ± 0.25	0.12 ± 0.08	0.00138796
CC-95%	50.57 ± 0.24	50.64 ± 0.26	0.13 ± 0.09	0.00145699
CC-99%	50.56 ± 0.24	50.67 ± 0.27	0.21 ± 0.11	0.00229643
Quality control-85%	-	-	0.10 ± 0.08	-
Quality control-90%	-	-	0.11 ± 0.08	-
Quality control-95%	-	-	0.12 ± 0.08	-
Quality control-99%	-	-	0.21 ± 0.11	-

Table 4.2: Results for ProstateX calibration. Results are presented as $Mean \pm SD$

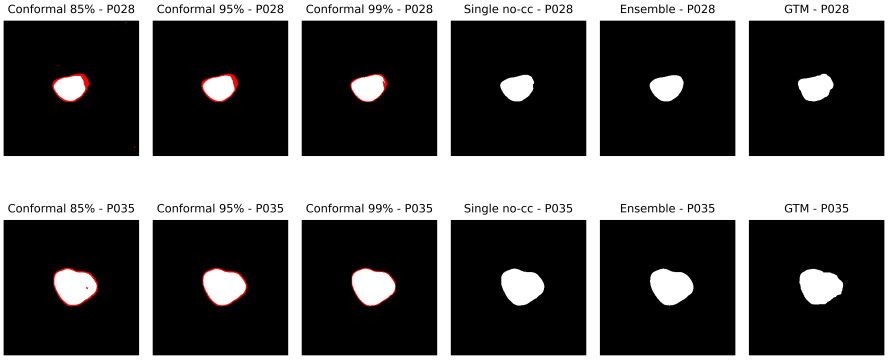
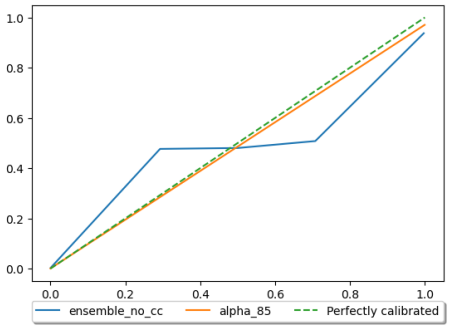
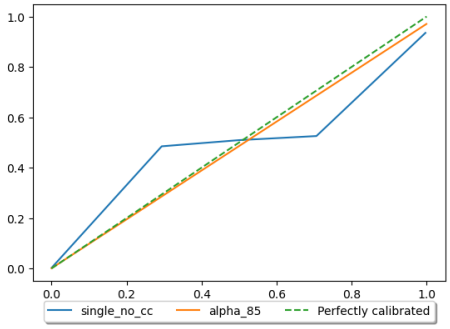


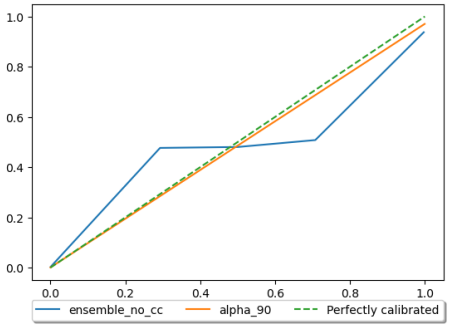
Figure 4.1: Prediction of prostate gland given by the conformal classifier, from the prostateX dataset



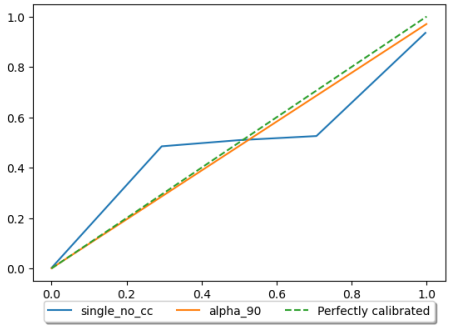
(a) Calibration curve of prostateX with an 0.85 confidence compared to ensemble model



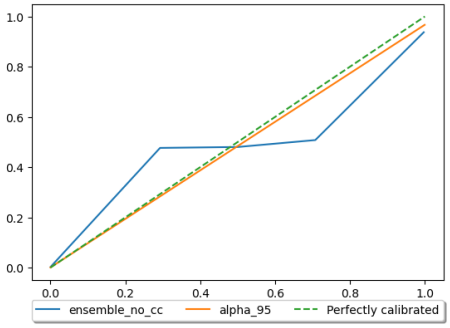
(b) Calibration curve of prostateX with an 0.85 confidence compared to single model



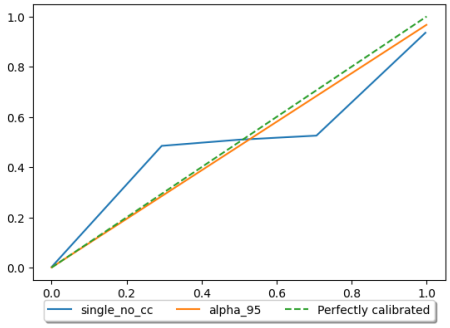
(c) Calibration curve of prostateX with an 0.90 confidence compared to ensemble model



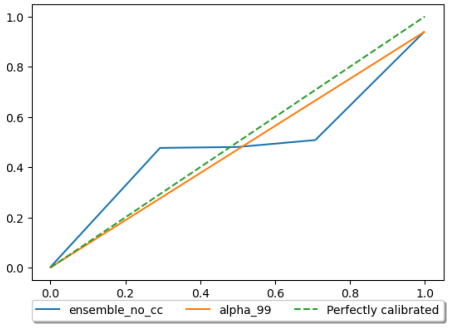
(d) Calibration curve of prostateX with an 0.90 confidence compared to single model



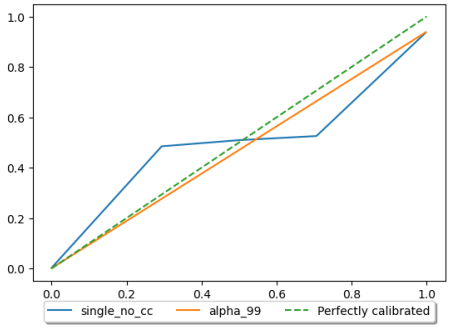
(e) Calibration curve of prostateX with an 0.95 confidence compared to ensemble model



(f) Calibration curve of prostateX with an 0.95 confidence compared to single model

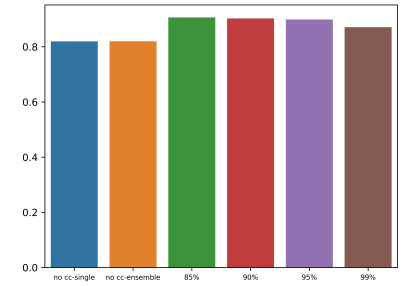


(g) Calibration curve of prostateX with an 0.99 confidence compared to ensemble model

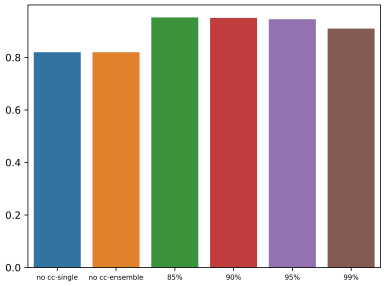


(h) Calibration curve of prostateX with an 0.99 confidence compared to single model

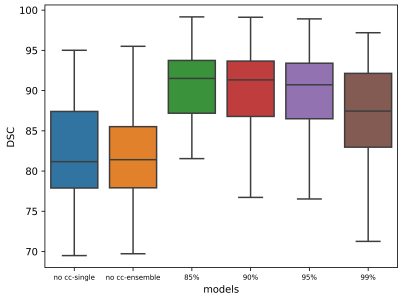
Figure 4.2: ProstateX calibration curves



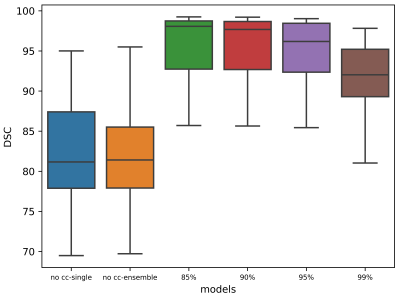
(a) DSC barplot of prostateX without QC



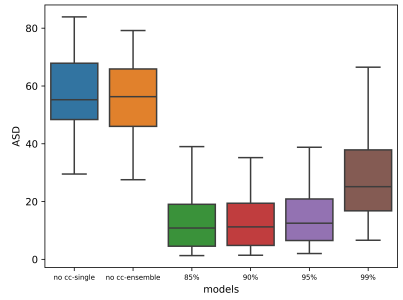
(b) DSC barplot of prostateX with QC



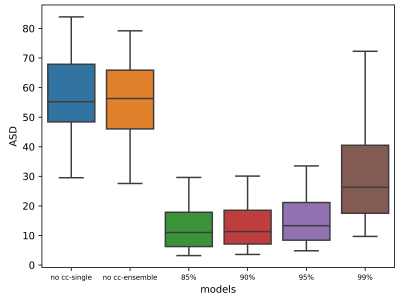
(c) DSC boxplot of prostateX without QC



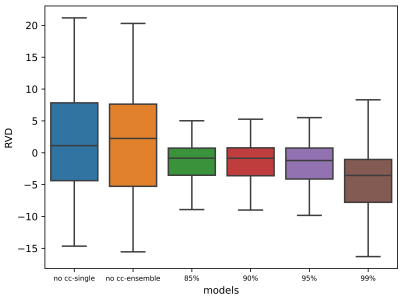
(d) DSC boxplot of prostateX with QC



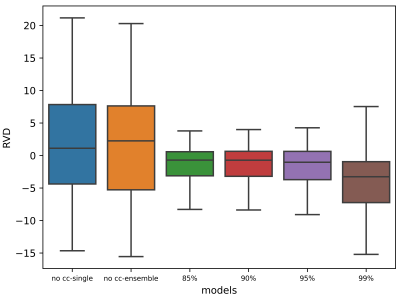
(e) ASD boxplot of prostateX without QC



(f) ASD boxplot of prostateX with QC



(g) RVD boxplot of prostateX without QC



(h) RVD boxplot of prostateX with QC

Figure 4.3: Visual depiction of key metrics: DSC, RVD and ASD for prostateX

4.1.2 External evaluation

SUS-cancer

Model	DSC	HD	ASD	RVD	AVD
5-fold ensemble	0.7797 ± 0.1087	13.2991 ± 16.4288	1.5724 ± 3.7821	4.80 ± 10.92	96.69 ± 12.80
Single model	0.7659 ± 0.1105	13.183 ± 15.831	1.6791 ± 3.9219	5.33 ± 11.03	96.16 ± 12.12
CC-85%	0.9072 ± 0.0829	11.6830 ± 11.2703	0.3462 ± 1.5008	-1.27 ± 3.25	101.41 ± 3.59
CC-90%	0.9015 ± 0.0886	11.6675 ± 11.4185	0.3686 ± 1.5925	-1.28 ± 3.31	101.42 ± 3.63
CC-95%	0.8941 ± 0.0899	11.7199 ± 13.5241	0.5183 ± 2.3615	-1.64 ± 3.78	101.83 ± 4.26
CC-99%	0.8596 ± 0.1013	12.5479 ± 15.2462	1.1131 ± 3.5695	-6.19 ± 6.95	107.35 ± 10.33
Quality control-85%	0.974 ± 0.033	5.6394 ± 11.3145	0.2551 ± 1.1011	-0.93 ± 2.07	100.99 ± 2.21
Quality control-90%	0.9726 ± 0.0331	5.8027 ± 11.4800	0.2751 ± 1.1822	-0.94 ± 2.13	101.00 ± 2.27
Quality control-95%	0.9638 ± 0.0369	6.4917 ± 13.8540	0.4109 ± 1.8751	-1.25 ± 2.60	101.34 ± 2.83
Quality control-99%	0.9075 ± 0.0800	10.1144 ± 15.7034	1.0151 ± 3.3475	-5.36 ± 5.96	106.18 ± 8.33

Table 4.3: Results for SUS-cancer segmentations. Results are presented as $Mean \pm SD$

Model	Validity	Efficiency	Brier loss score	ECE
5-fold ensemble	-	-	0.31 ± 0.13	0.00256632
Single model	-	-	0.31 ± 0.13	0.00270147
CC-85%	50.38 ± 0.20	50.38 ± 0.20	0.03 ± 0.02	0.00029722
CC-90%	50.349 ± 0.220	50.36 ± 0.22	0.03 ± 0.02	0.00029688
CC-95%	50.33 ± 0.23	50.349 ± 0.240	0.04 ± 0.03	0.00035355
CC-99%	50.34 ± 0.25	50.40 ± 0.26	0.10 ± 0.06	0.00098402
Quality control-85%	-	-	0.02 ± 0.02	-
Quality control-90%	-	-	0.02 ± 0.02	-
Quality control-95%	-	-	0.03 ± 0.02	-
Quality control-99%	-	-	0.10 ± 0.06	-

Table 4.4: Results for SUS-cancer calibration. Results are presented as $Mean \pm SD$

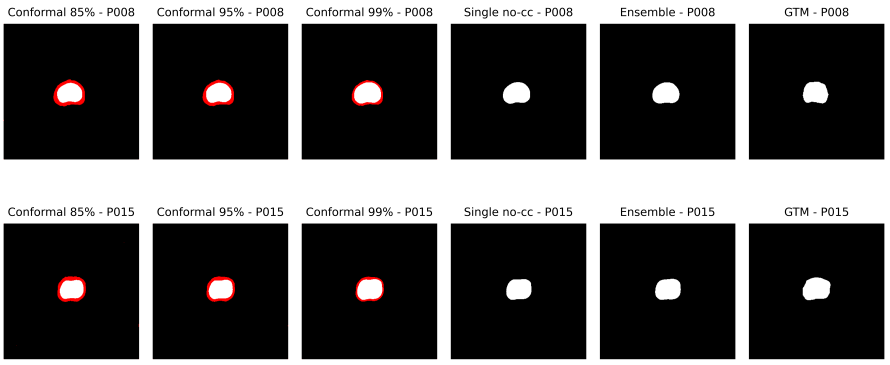
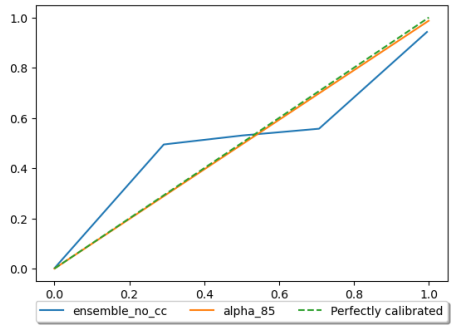
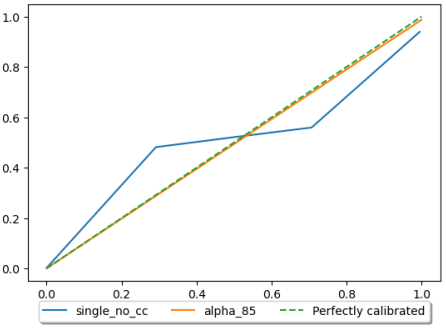


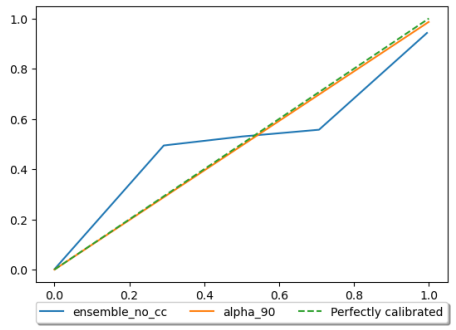
Figure 4.4: Prediction of the prosatete gland given by the conformal classifier, from the SUS dataset with cancer



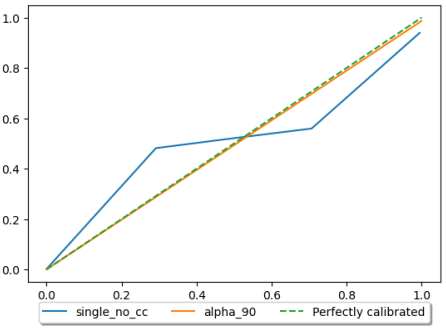
(a) Calibration curve of SUS-cancer with an 0.85 confidence compared to ensemble model



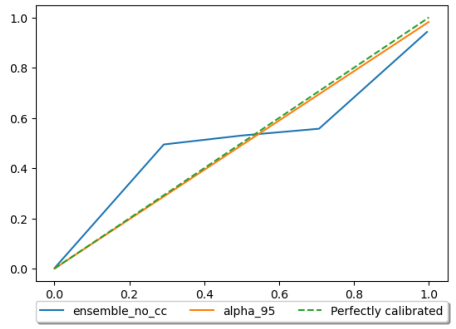
(b) Calibration curve of SUS-cancer with an 0.85 confidence compared to single model



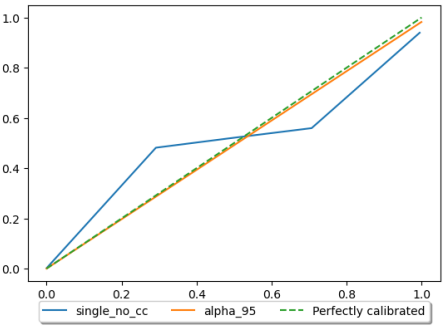
(c) Calibration curve of SUS-cancer with an 0.90 confidence compared to ensemble model



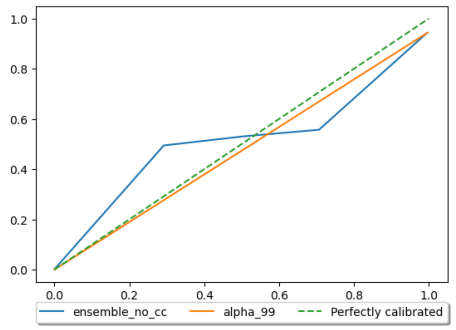
(d) Calibration curve of SUS-cancer with an 0.90 confidence compared to single model



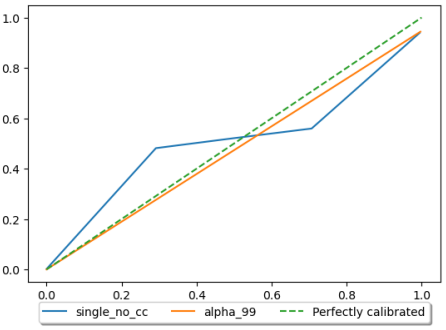
(e) Calibration curve of SUS-cancer with an 0.95 confidence compared to ensemble model



(f) Calibration curve of SUS-cancer with an 0.95 confidence compared to single model

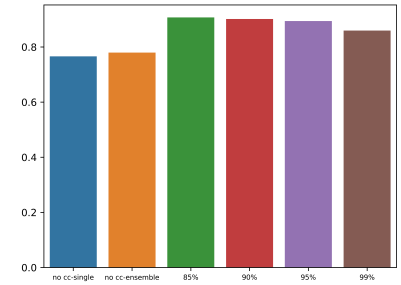


(g) Calibration curve of SUS-cancer with an 0.99 confidence compared to ensemble model

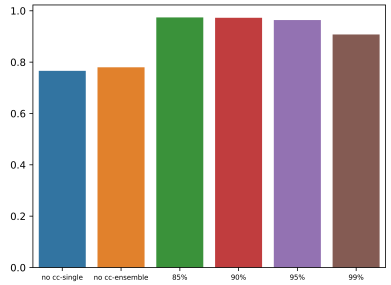


(h) Calibration curve of SUS-cancer with an 0.99 confidence compared to single model

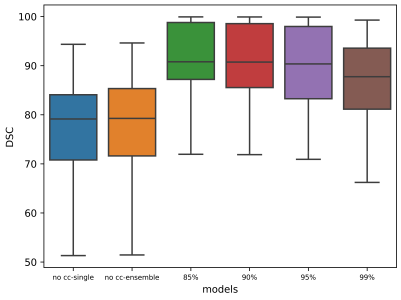
Figure 4.5: SUS-cancer calibration curve



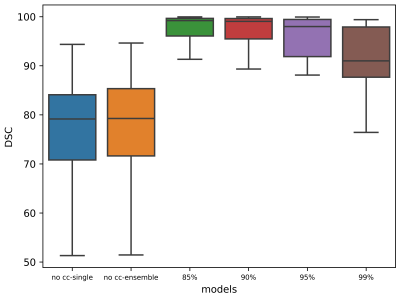
(a) DSC barplot of SUS-cancer without QC



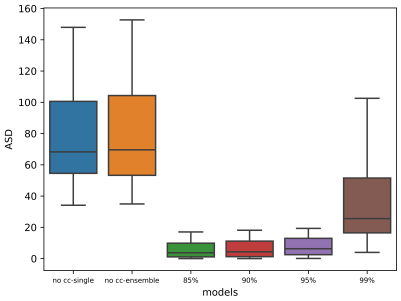
(b) DSC barplot of SUS-cancer with QC



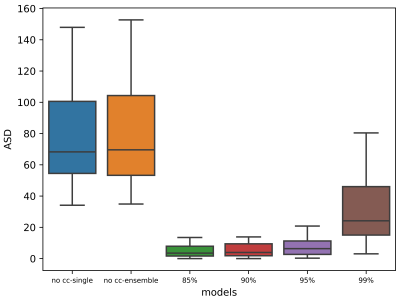
(c) DSC boxplot of SUS-cancer without QC



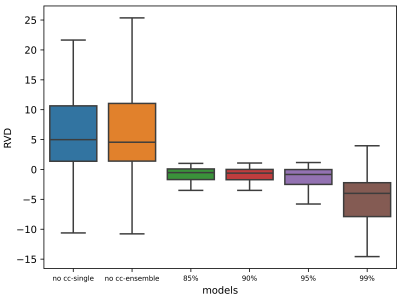
(d) DSC boxplot of SUS-cancer with QC



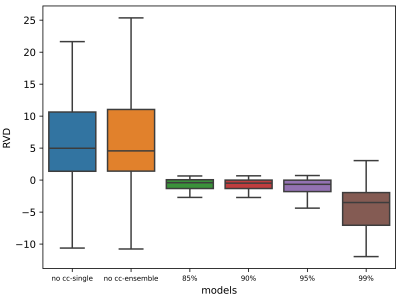
(e) ASD boxplot of SUS-cancer without QC



(f) ASD boxplot of SUS-cacncerwith QC



(g) RVD boxplot of SUS-cancer without QC



(h) RVD boxplot of SUS-cancer with QC

Figure 4.6: Visual depiction of key metrics: DSC, RVD and ASD for SUS-cancer

SUS-negative

Model	DSC	HD	ASD	RVD	AVD
5-fold ensemble	0.8317 ± 0.0737	6.8658 ± 2.3229	0.6332 ± 0.1025	6.03 ± 4.28	94.46 ± 3.84
Single model	0.8104 ± 0.0531	6.9059 ± 2.3039	0.6457 ± 0.0732	7.21 ± 4.42	93.44 ± 3.96
CC-85%	0.9438 ± 0.0550	9.1158 ± 3.5521	0.0494 ± 0.0370	-0.71 ± 0.75	100.72 ± 0.76
CC-90%	0.9435 ± 0.0550	9.0400 ± 3.5647	0.0528 ± 0.0420	-0.70 ± 0.84	100.72 ± 0.84
CC-95%	0.9361 ± 0.0584	8.5101 ± 3.3603	0.0695 ± 0.0592	-0.85 ± 1.10	100.87 ± 1.12
CC-99%	0.9327 ± 0.0597	7.1684 ± 2.5156	0.1951 ± 0.1087	-3.32 ± 2.11	103.48 ± 2.26
Quality control-85%	0.9858 ± 0.0209	3.1590 ± 1.8953	0.0396 ± 0.0216	-0.53 ± 0.55	100.54 ± 0.56
Quality control-90%	0.9846 ± 0.0206	3.3166 ± 1.9273	0.0436 ± 0.0229	-0.52 ± 0.63	100.53 ± 0.63
Quality control-95%	0.9809 ± 0.0211	3.5033 ± 2.0032	0.0569 ± 0.0296	-0.66 ± 0.85	100.68 ± 0.86
Quality control-99%	0.9658 ± 0.0231	4.9759 ± 2.0593	0.1818 ± 0.9220	-2.93 ± 1.91	103.06 ± 2.04

Table 4.5: Results for SUS-negative segmentation. Results are presented as $Mean \pm SD$

Model	Validity	Efficiency	Brier loss score	ECE
5-fold ensemble	-	-	0.30 ± 0.11	0.00248617
Single model	-	-	0.32 ± 0.12	0.0026801
CC-85%	50.48 ± 0.21	50.49 ± 0.21	0.03 ± 0.11	0.00026722
CC-90%	50.44 ± 0.20	50.45 ± 0.21	0.03 ± 0.02	0.00026794
CC-95%	50.42 ± 0.20	50.44 ± 0.21	0.03 ± 0.02	0.00033286
CC-99%	50.45 ± 0.21	50.49 ± 0.23	0.09 ± 0.05	0.00083488
Quality control-85%	-	-	0.02 ± 0.01	-
Quality control-90%	-	-	0.02 ± 0.02	-
Quality control-95%	-	-	0.03 ± 0.02	-
Quality control-99%	-	-	0.09 ± 0.05	-

Table 4.6: Results for SUS-negative calibration. Results are presented as $Mean \pm SD$

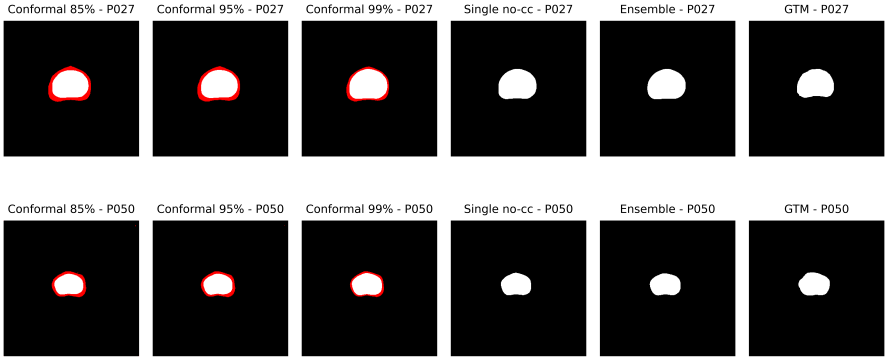
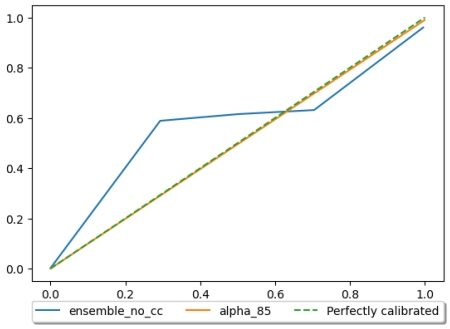
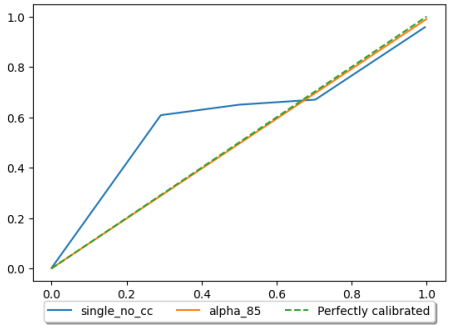


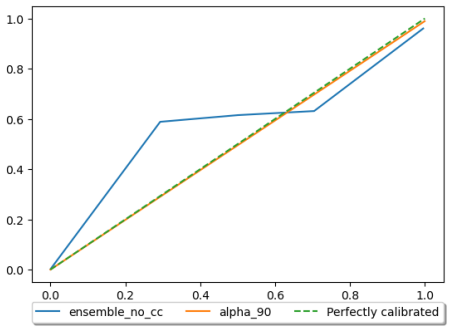
Figure 4.7: Prediction of the prostate gland given by the conformal classifier, from the SUS dataset without cancer



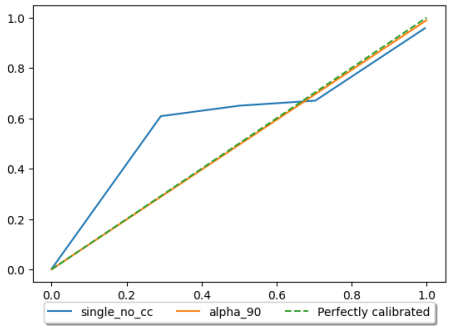
(a) Calibration curve of SUS-negative with an 0.85 confidence compared to ensemble model



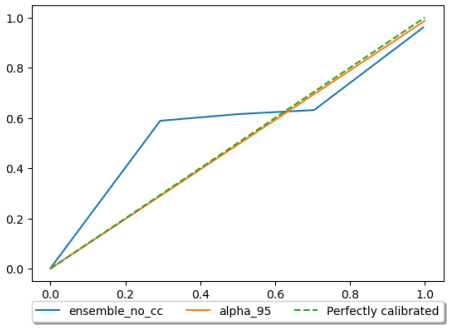
(b) Calibration curve of SUS-negative with an 0.85 confidence compared to single model



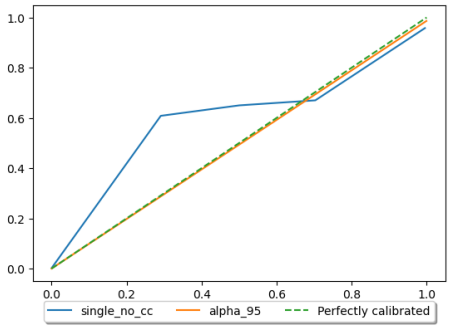
(c) Calibration curve of SUS-negative with an 0.90 confidence compared to ensemble model



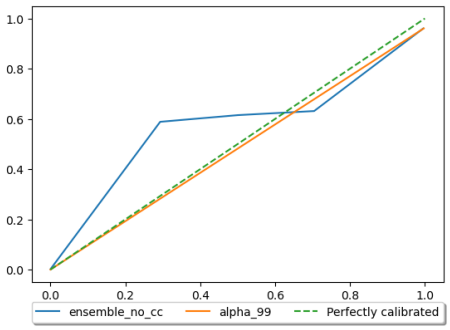
(d) Calibration curve of SUS-negative with an 0.90 confidence compared to single model



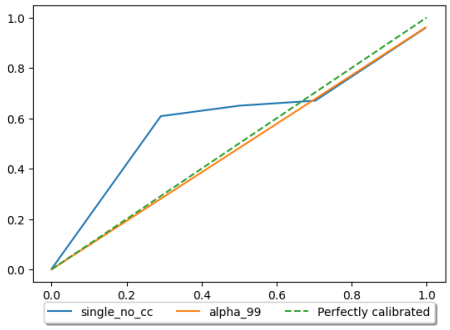
(e) Calibration curve of SUS-negative with an 0.95 confidence compared to ensemble model



(f) Calibration curve of SUS-negative with an 0.95 confidence compared to single model

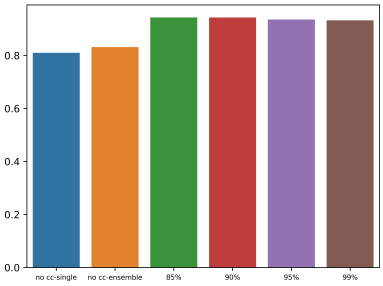


(g) Calibration curve of SUS-negative with an 0.99 confidence compared to ensemble model

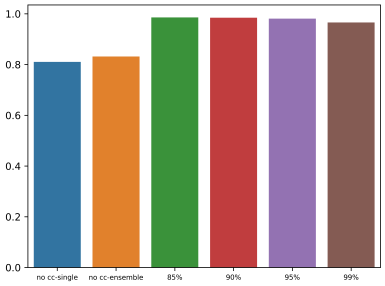


(h) Calibration curve of SUS-negative with an 0.99 confidence compared to single model

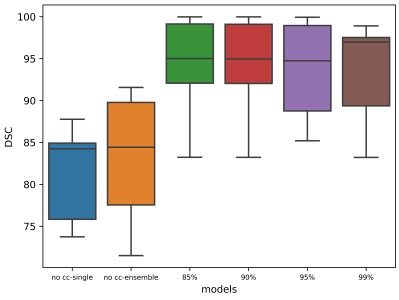
Figure 4.8: SUS-negative calibration curves



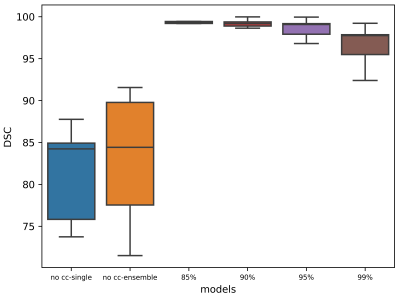
(a) DSC barplot of SUS-negative without QC



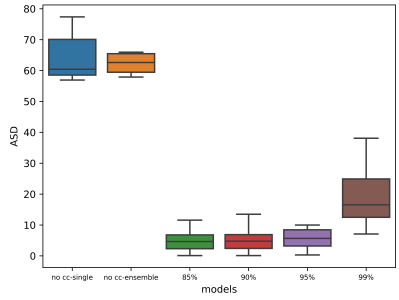
(b) DSC barplot of SUS-negative with QC



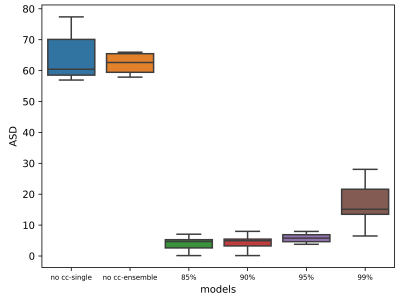
(c) DSC boxplot of SUS-negative without QC



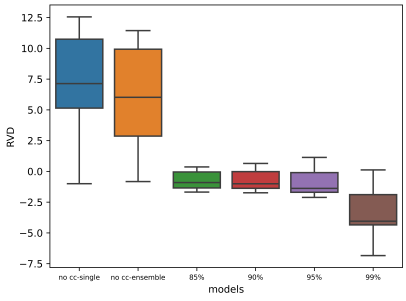
(d) DSC boxplot of SUS-negative with QC



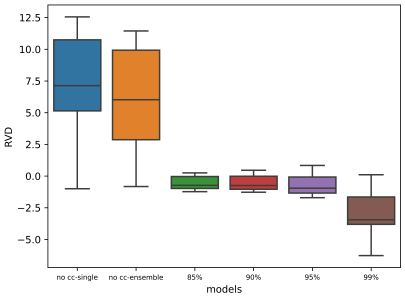
(e) ASD boxplot of SUS-negative without QC



(f) ASD boxplot of SUS-cacncerwith QC



(g) RVD boxplot of SUS-negative without QC



(h) RVD boxplot of SUS-negative with QC

Figure 4.9: Visual depiction of key metrics: DSC, RVD and ASD for SUS-negative

P158-cancer

Model	DSC	HD	ASD	RVD	AVD
5-fold ensemble	0.5437 ± 0.1539	15.5410 ± 10.6747	1.4607 ± 0.7253	123.21 ± 409.97	68.11 ± 21.68
Single model	0.5632 ± 0.1539	14.1448 ± 10.5269	1.4217 ± 0.5845	75.54 ± 207.75	73.18 ± 20.70
CC-85%	0.6883 ± 0.2330	18.5365 ± 15.9380	0.0373 ± 0.1724	195.23 ± 1577.80	89.11 ± 21.83
CC-90%	0.6791 ± 0.2323	18.5713 ± 15.8028	0.0383 ± 0.1732	151.71 ± 1152.82	88.35 ± 22.47
CC-95%	0.6722 ± 0.2268	17.3795 ± 13.8855	0.0445 ± 0.1708	35.57 ± 156.85	88.44 ± 22.05
CC-99%	0.6962 ± 0.1881	13.8175 ± 11.8033	0.2057 ± 0.1775	17.23 ± 97.12	96.18 ± 16.95
Quality control-85%	0.9659 ± 0.0393	5.9009 ± 5.6375	0.1216 ± 0.1517	2.3 ± 3.7	97.87 ± 3.18
Quality control-90%	0.9619 ± 0.0417	6.1433 ± 5.7157	0.1390 ± 0.1717	2.69 ± 4.30	97.53 ± 3.62
Quality control-95%	0.9540 ± 0.0456	6.3977 ± 5.7359	0.1656 ± 0.1984	3.17 ± 5.18	97.14 ± 4.28
Quality control-99%	0.9012 ± 0.0695	7.4263 ± 5.6265	0.3556 ± 0.2554	1.48 ± 8.62	99.15 ± 7.26

Table 4.7: Results for P158-cancer segmentation. Results are presented as $Mean \pm SD$

Model	Validity	Efficiency	Brier loss score	ECE
5-fold ensemble	-	-	2.13 ± 1.42	0.01646629
Single model	-	-	2.00 ± 1.19	0.0155795
CC-85%	50.60 ± 0.49	50.74 ± 0.44	0.24 ± 0.03	0.00222658
CC-90%	50.55 ± 0.47	50.70 ± 0.42	0.25 ± 0.31	0.00228943
CC-95%	50.54 ± 0.48	50.70 ± 0.43	0.27 ± 0.32	0.00245664
CC-99%	50.74 ± 0.56	51.00 ± 0.55	0.5 ± 0.4	0.00422945
Quality control-85%	-	-	0.16 ± 0.20	-
Quality control-90%	-	-	0.18 ± 0.23	-
Quality control-95%	-	-	0.21 ± 0.26	-
Quality control-99%	-	-	0.43 ± 0.33	-

Table 4.8: Results for P158-cancer calibration. Results are presented as $Mean \pm SD$

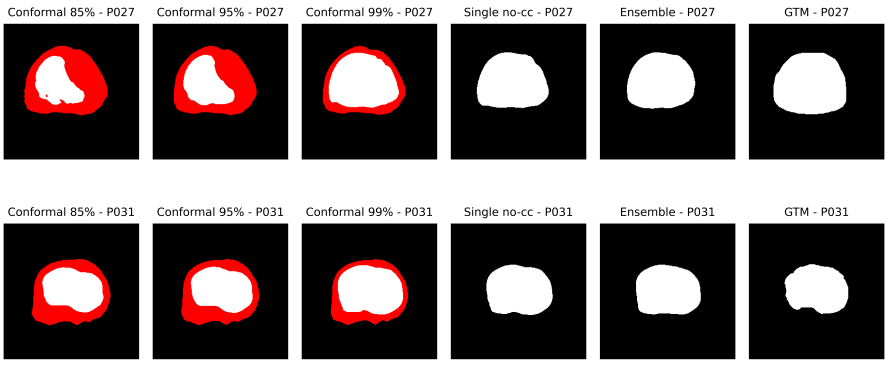
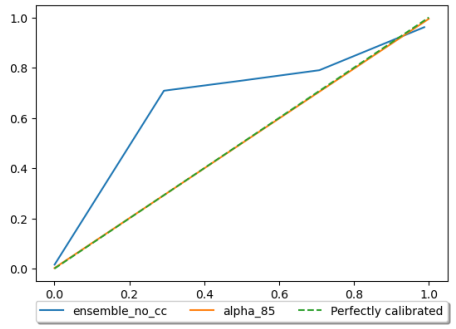
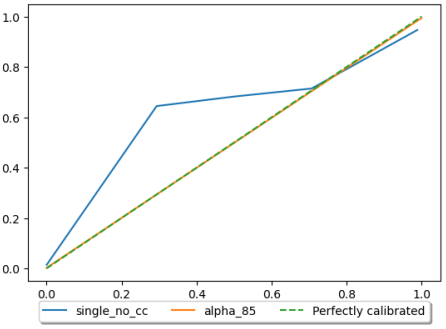


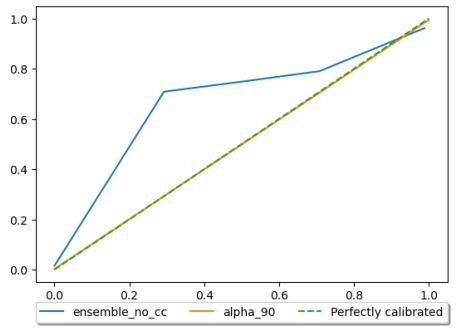
Figure 4.10: Prediction of prostate gland given by the conformal classifier, from the P158 dataset with cancer



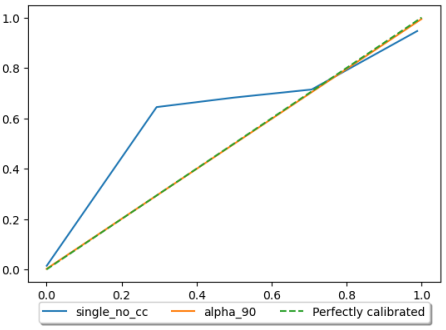
(a) Calibration curve of P158-cancer with an 0.85 confidence compared to ensemble model



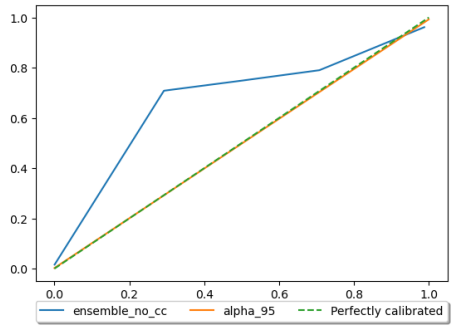
(b) Calibration curve of P158-cancer with an 0.85 confidence compared to single model



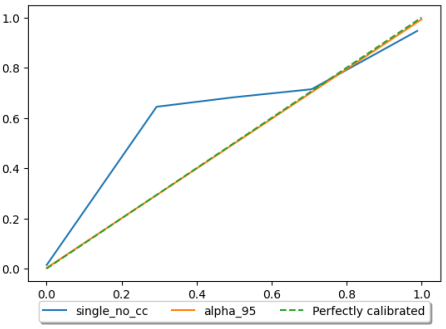
(c) Calibration curve of P158-cancer with an 0.90 confidence compared to ensemble model



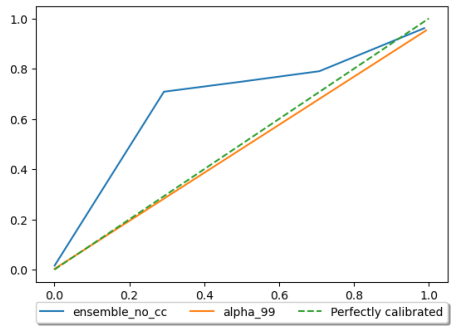
(d) Calibration curve of P158-cancer with an 0.90 confidence compared to single model



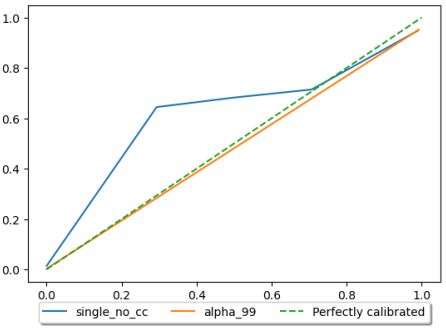
(e) Calibration curve of P158-cancer with an 0.95 confidence compared to ensemble model



(f) Calibration curve of P158-cancer with an 0.95 confidence compared to single model

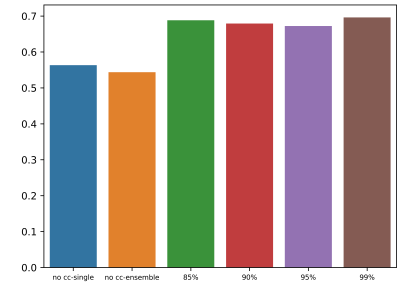


(g) Calibration curve of P158-cancer with an 0.99 confidence compared to ensemble model

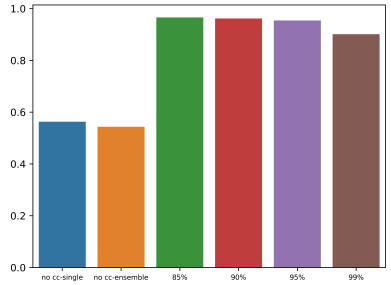


(h) Calibration curve of P158-cancer with an 0.99 confidence compared to single model

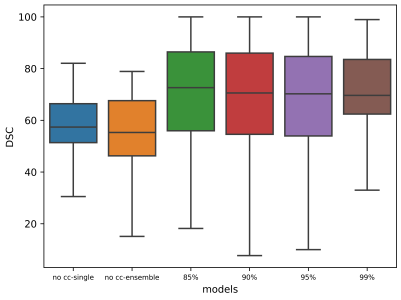
Figure 4.11: P158-cancer calibration curves



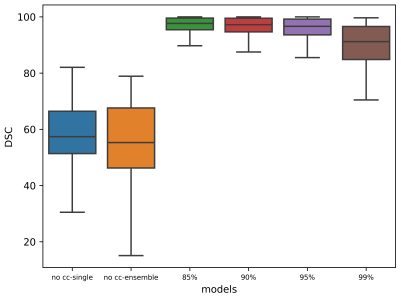
(a) DSC barplot of P158-cancer without QC



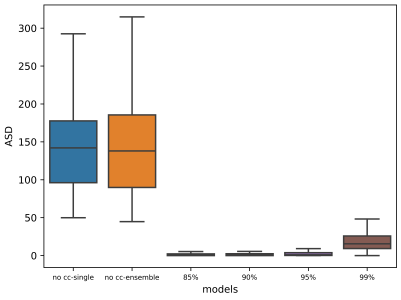
(b) DSC barplot of P158-cancer with QC



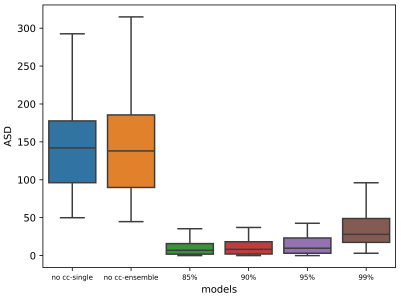
(c) DSC boxplot of P158-cancer without QC



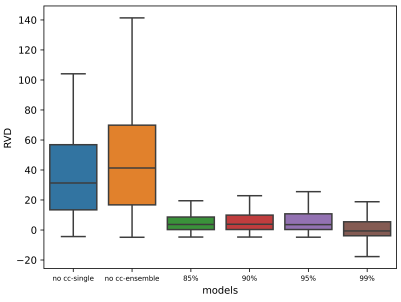
(d) DSC boxplot of P158-cancer with QC



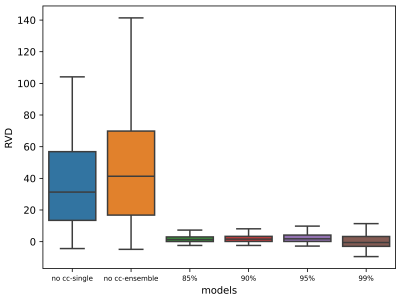
(e) ASD boxplot of P158-cancer without QC



(f) ASD boxplot of P158-cacncerwith QC



(g) RVD boxplot of P158-cancer without QC



(h) RVD boxplot of P158-cancer with QC

Figure 4.12: Visual depiction of key metrics: DSC, RVD and ASD for P158-cancer

P158-negative

Model	DSC	HD	ASD	RVD	AVD
5-fold ensemble	0.6095 ± 0.1588	15.1926 ± 1.4473	1.4473 ± 0.7837	50.58 ± 43.72	76.67 ± 18.83
Single model	0.6205 ± 0.1060	14.4737 ± 10.2345	1.4441 ± 0.6751	30.67 ± 33.50	80.82 ± 17.51
CC-85%	0.7272 ± 0.1588	18.1172 ± 14.0839	0.0290 ± 0.0454	6.99 ± 13.76	94.61 ± 9.06
CC-90%	0.7159 ± 0.1529	18.1100 ± 13.9563	0.0308 ± 0.0470	7.57 ± 14.61	94.22 ± 9.43
CC-95%	0.7149 ± 0.1521	17.3712 ± 13.2520	0.0407 ± 0.0547	7.39 ± 14.26	94.33 ± 9.42
CC-99%	0.7215 ± 0.1159	14.4842 ± 11.2794	0.2569 ± 0.2116	0.67 ± 11.16	100.36 ± 9.48
Quality control-85%	0.9617 ± 0.0465	6.030 ± 2.921	0.1530 ± 0.1698	1.76 ± 2.99	98.35 ± 2.77
Quality control-90%	0.9584 ± 0.0483	6.1619 ± 2.9843	0.1697 ± 0.2204	1.99 ± 3.26	98.14 ± 2.99
Quality control-95%	0.9520 ± 0.0525	6.3727 ± 3.0553	0.1999 ± 0.2527	2.18 ± 3.74	97.99 ± 3.42
Quality control-99%	0.9018 ± 0.0681	7.9658 ± 4.5090	0.4403 ± 0.3311	-0.72 ± 6.60	101.15 ± 6.50

Table 4.9: Results for P158-negative segmentation. Results are presented as $Mean \pm SD$

Model	Validity	Efficiency	Brier loss score	ECE
5-fold ensemble	-	-	2.57 ± 2.39	0.01925756
Single model	-	-	2.45 ± 2.07	0.0186994
CC-85%	50.93 ± 0.48	51.11 ± 0.47	0.34 ± 0.54	0.00279603
CC-90%	50.88 ± 0.47	51.05 ± 0.46	0.35 ± 0.56	0.00284424
CC-95%	50.86 ± 0.47	51.05 ± 0.47	0.38 ± 0.59	0.00311162
CC-99%	51.04 ± 0.48	51.41 ± 0.59	0.71 ± 0.71	0.00615668
Quality control-85%	-	-	0.22 ± 0.34	-
Quality control-90%	-	-	0.24 ± 0.37	-
Quality control-95%	-	-	0.28 ± 0.42	-
Quality control-99%	-	-	0.62 ± 0.59	-

Table 4.10: Results for P158-negative calibration. Results are presented as $Mean \pm SD$

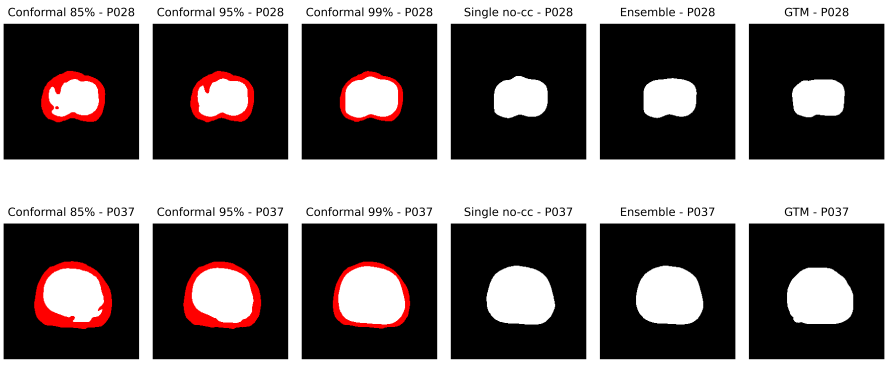
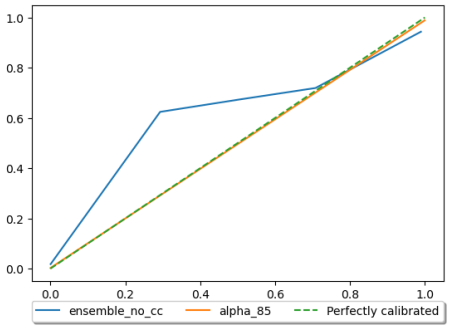
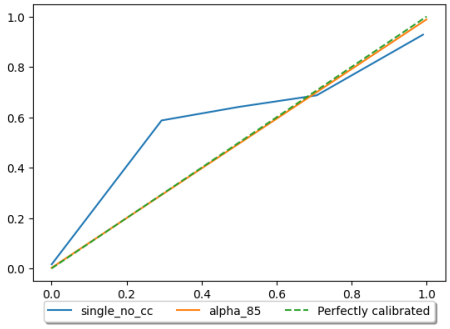


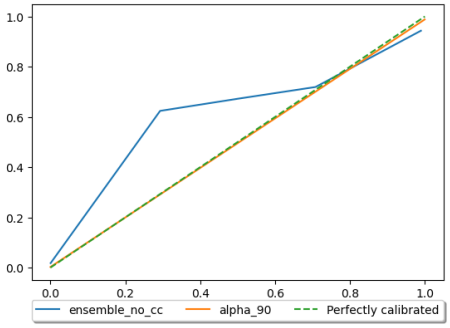
Figure 4.13: Prediction of prostate gland given by the conformal classifier, from the P158 dataset without cancer



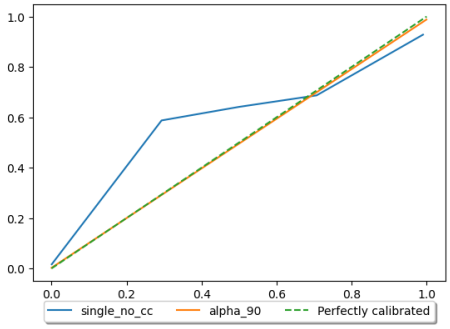
(a) Calibration curve of P158-negative with an 0.85 confidence compared to ensemble model



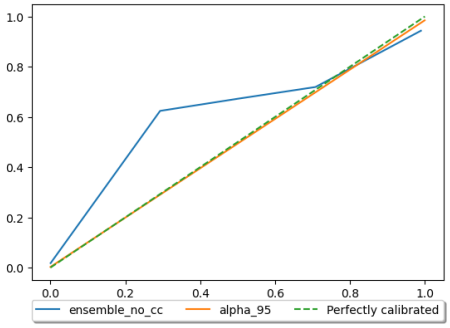
(b) Calibration curve of P158-negative with an 0.85 confidence compared to single model



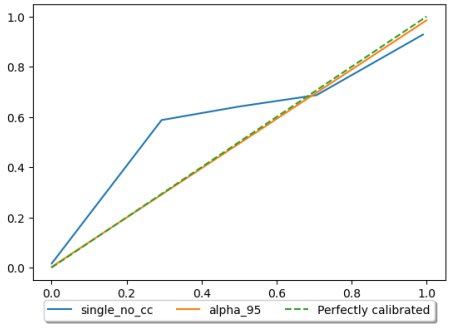
(c) Calibration curve of P158-negative with an 0.90 confidence compared to ensemble model



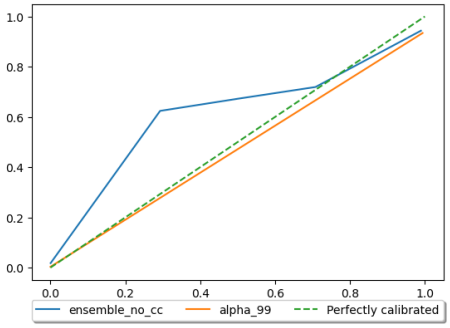
(d) Calibration curve of P158-negative with an 0.90 confidence compared to single model



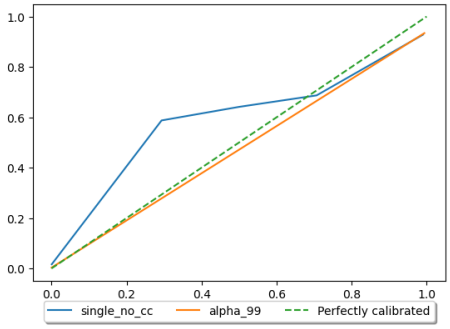
(e) Calibration curve of P158-negative with an 0.95 confidence compared to ensemble model



(f) Calibration curve of P158-negative with an 0.95 confidence compared to single model

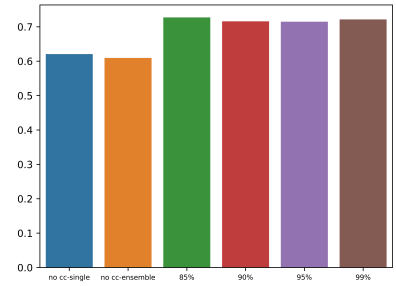


(g) Calibration curve of P158-negative with an 0.99 confidence compared to ensemble model

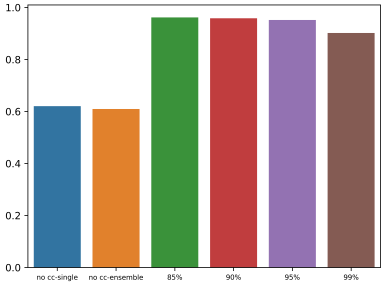


(h) Calibration curve of P158-negative with an 0.99 confidence compared to single model

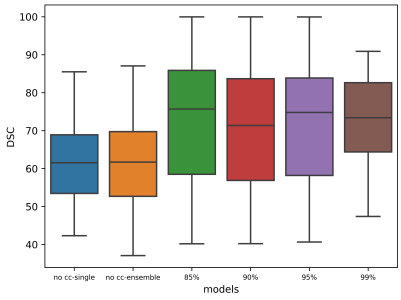
Figure 4.14: P158-negative calibration curve



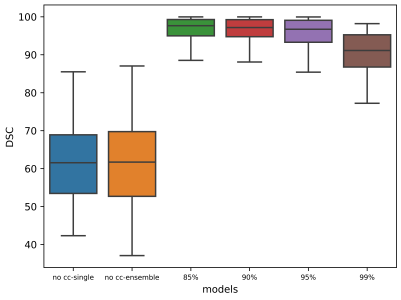
(a) DSC barplot of P158-negative without QC



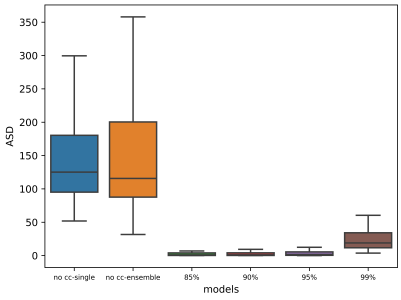
(b) DSC barplot of P158-negative with QC



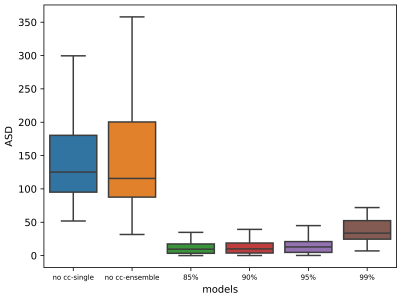
(c) DSC boxplot of P158-negative without QC



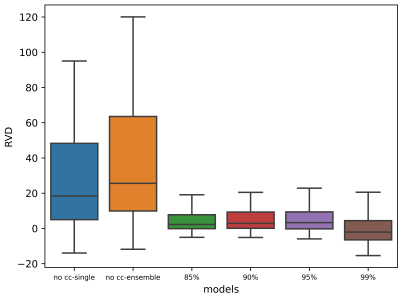
(d) DSC boxplot of P158-negative with QC



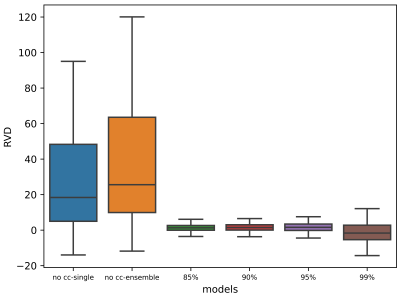
(e) ASD boxplot of P158-negative without QC



(f) ASD boxplot of P158-cacncerwith QC



(g) RVD boxplot of P158-negative without QC



(h) RVD boxplot of P158-negative with QC

Figure 4.15: Visual depiction of key metrics: DSC, RVD and ASD for P158-negative

Chapter 5

Discussion

5.1 ProstateX

The results for the internal dataset at table 4.1 showcase how the segmentation improves when we use the conformal classifier(2.2.3). With the non-conformal models, the DSC is nearly identical, with 5-fold ensemble DSC = 0.8198 and Single model DSC = 0.8197. When comparing the worst non-cc model with the best conformal classifier without quality control, there is an increase in 0.0866 for the DSC, from a DSC = 0.8197 for single model to DSC = 0.9064 for CC-85%. This shows that the conformal classifier increases the accuracy of its prediction **when it is certain about it**. It seems that the outliers is further away the smaller the alpha is, as the Hausdorff distance is bigger on every conformal classifier without quality control, for example with CC-85% HD = 9.0660 against 5-fold ensemble HD = 7.7628. Regarding the RVD, the non-cc models have a slightly bigger volume compared to the GTM as single-model RVD = 1.56 and ensemble model RVD = 1.73. The different CC's both with and without quality control have negative value, which means the predicted segmentation is smaller than the gtm, with the quality control version being closer to the original volume.

When adding the quality control to the different alphas of the conformal classifier, there is another increase for the DSC, from CC-85% DSC = 0.9064 to Quality control-85% = 0.9523, which is expected as every uncertain point gets checked and becomes the correct label. The pattern where the conformal classifier is the best at alpha = 85% confidence is the same, and it decreases in accuracy when we increase the confidence level.

The table at 4.2 showcases the effectiveness of conformal prediction, and compares them to the non-cc models. First is the ECE, we can see that the conformal classifier reduces classification error, from single model ECE = 0.00401208 against CC-90% = 0.00138796.

There is a reduction in the brier score loss as well, from single model brier score = 0.42 to CC-85% brier score = 0.12. The validity and efficiency shows that the conformal classifiers manages to make a certain prediction for about half of the predictions it made, which could be improved upon.

The combination of conformal classifier with expertise makes the accuracy of the segmentation predictions much better. Even without the quality control the DSC improves quite a bit, showing the effectiveness of conformal prediction on the internal dataset.

5.2 SUS-cancer

The results of the SUS patients segmentations with cancer can be seen at table 4.3. The no-cc models is worse for this dataset than for the internal prostateX dataset in every metric. This will also show the effectiveness with the conformal classifier, as every metric improves with the conformal classifier. We can see that the DSC becomes very similar to the DSC of the prostateX conformal classifier, with the DSC for the 85% models being 0.9064 for prostateX and 0.9072 for SUS-cancer. The jump between 95% confidence and 99% confidence is a lot bigger here than in the prostateX, taking a bigger decrease in accuracy in the predictions. From a DSC = 0.9072 for 85% confidence to DSC = 0.8596 for 99% confidence.

In this case, the HD and SD is worse in the SUS-cancer compared to the prostateX set. The HD for prostateX doesn't go above 10, but for SUS-cancer, it is consistently above 10. For the HD, this shows that the model has some outliers in its predictions that is even further away than in the prostateX. Standard Deviation increases between the datasets, for example with DSC SD for prostateX = 0.0593 against DSC SD for SUS-cancer = 0.1087. This increase happens with every metric over every model. When quality control gets implemented, the DSC is only 3% off from being perfect, with it going from DSC = 0.9072 for CC-85% to DSC = 0.974 for Quality control-85%. The HD is decreased tremendously, going down from HD = 11.6830 for CC-85% to HD = 5.6394 for quality control-85%. The ASD doesn't change drastically, but has a slight decrease and improvement, same with the RVD.

In table 4.4 for the calibration, the validity and efficiency doesn't change that much. The Brier loss score seems to be a little better for non-cc models in this dataset than for prostateX. The Brier score loss is better for the different conformal classifiers as well, going from 0.31 for both non-cc models to around 0.03-0.04 for every model except CC-99% which have a Brier score = 0.10. The ECE seems to be better than in the

SUS-cancer than with the prostateX, reducing the calibration error from prostateX CC-85% ECE = 0.00142969 to SUS-cancer CC-85% ECE = 0.00029722.

This showcases some very interesting properties, as in many aspects, it seems the model, which was trained with prostateX, is better at predicting segmentations from SUS than from prostateX. Yet its not perfect, as the HD is bigger than with the prostateX, telling us that while it manages to predict the prostate pretty well, it predicts some outliers outside the prostate.

5.3 SUS-negative

Table 4.5 showcases the segmentation results from the SUS patients that are healthy. The DSC and HD of this is the best out of all the datasets, with the ensemble having a DSC = 0.8317 and HD = 6.8658. The ASD in this table are a little bit worse, with ASD = 0.6332 for the ensemble model. The RVD and the AVD for the non-cc models is worse than both the prostateX and the SUS-cancer, having RVD above 6 and AVD below 96 for both models.

For the cc-models, the RVD is actually better than for both SUS-cancer and prostateX, with CC-90% RVD = -0.7. Both SUS-datasets follow the same pattern of the model getting worse when the confidence level increases, but it is important to note that for SUS-negative, it doesn't have such a huge decrease as with both SUS-cancer and prostateX. When the quality control is implemented, there is an improvement in almost every metric, while the AVD have very minor changes.

The table at 4.6 shows how good the conformal calibration is. The calibration is very similar to the SUS-cancer set, which makes sense, as they are the same dataset split into two.

5.4 Patient158-Cancer

Table 4.7 shows the segmentation results for the P158 dataset patients with cancer. The model struggles a lot more with this dataset than the previous ones, with a pretty significant decrease in accuracy for the predictions, as showcased by all the metrics. The non-cc models have significant lower DSC than the previous datasets, with the single model DSC = 0.5632 for P158 with cancer. The models are showing a huge decrease in similarity between the predicted segmentations and the GTM-segmentations. The HD showcases outliers pretty far from the prostate, with the biggest being 5-fold ensemble with

an HD = 15.5410. Interestingly, the ASD doesn't seem to change that much compared to the SUS datasets. The RVD and AVD showcases some of the biggest differences, with a significant increase in volume difference between the predicted segmentation and GTM segmentations, with the predicted segmentations having RVD = 123, compared to previous RVD which were below 10.

When the CC gets applied, the predictions have a pretty significant improvement, yet is still worse than when we use the models on the other datasets. Something worth noting is that the conformal classifier doesn't have a obvious decrease when the alpha is increased as with the other models. The worst DSC is with confidence 95% = 0.6722, with 99% having a the biggest DSC = 0.6962.

With the quality control, predictions have become much better. In the other datasets, the appliance of quality control didn't necessarily change the results tremendously, as the other datasets had pretty good cc models, the scores didn't change massively. But in this dataset, applying quality control have a tremendous effect upon the conformal classifier, as the DSC goes from 0.6883 up 0.9659 for 85% confidence, and every other metric becomes better.

Table 4.8 shows that the metrics showcases mostly expected things compared to table 4.7. With an worse ECE and brier score compared to the other datasets. The validity and efficiency stays the same, showing that the amount of uncertainties are persistent.

5.5 Patient158-Negative

Table 4.9 shows segmentation results for healthy P158 patients. Just as with P158 which have cancer, the no-cc models struggle more with this one, but does a better job here than with P158-cancer. This dataset is very similar to the P158-cancer set in almost every regards. It is worth noting that the RVD is better in the dataset with healthy patients than in those with cancer, with RVD for CC-90% = 0.67 for healthy P158-patients and RVD for CC-90% = 17.23 for P158-patients with cancer. This is probably because the models had a easier time with the P158-negative, as the other metrics are also better here than with the P158-cancer.

There are no significant changes in table 4.10, compared to the table 4.8, except that the loss score for the no-cc models are a little bit bigger.

5.6 Calibration

We can see in figure groups 4.2, 4.5, 4.8, 4.11 and 4.14 the calibration curves for the different datasets. With every dataset, there is a clear improvement of the calibration curve when the conformal classifier is applied. It is important to note that the calibration curve for the cc is only **when it is certain about it**.

5.7 Limitations

5.7.1 Alternatives to nnUNet

During this project, we have only tested nnUNet. There exist multiple different libraries that trains and uses different versions of U-Net for predictions and classifications. The segmentations models library[39] has 4 different models to use, U-net, FPN, Linknet and PSPNet. The main reasoning why we choose nnUNet instead of other libraries like segmentation models, was because of the automatic calibration nnUNet does for you, as well as the ease of implementing it. The nnUNet is also considered the standard architecture for prostate gland segmentation.

5.7.2 Alternatives to crepes for conformal classifier

One library we considered during the project was Puncc[40]. This model comes with state-of-the-art conformal predictions algorithms. As with any conformal predictions model, Puncc can be used with any predictive model to provide rigorous uncertainty estimations. However, we decided to continue with crepes library. Crepes[36] is a python package that implements conformal classifiers, regressors and predictive systems on top of any classifier and regressor. The main reasoning Crepes was chosen is because of the ease of implementing it, as Nguyen et al.[4] worked with this library. The crepes conformal classifier can also work on the results from the model, while Puncc wraps around the model when we use it. This makes Puncc not work for our project, as nnUNet gets started in the terminal, and not as an object in python.

Conformal prediction is defined as an online transductive framework (OTF)[20]. In an OTF, the conformity score is calculated with all available data[20]. This score get calculated for each new example to make predictions on[20]. This makes us retrain the underlying machine learning model for every calibration example, and every test example[20]. While OTF is is better since it uses all available data, it is also to time expensive to do[20]. We instead use a fixed model, or a inductive offline framework.

CP will instead have one model built from a training set and applied to a test set[20]. Inductive CP is more computationally effective, and the method of training is more align with normal models, have a training set to train, a calibration set to validate and a test set to test the model[20].

Chapter 6

Conclusions

6.1 Summary

During this project, we wanted to test an conformal classifier to see if it can improve the results of a single fold model. The results shows that the conformal classifier has better results, granting better predictions where it is certain. The combination of an experts intervention and the conformal predictions showcases an even better prediction possibility.

In particular, the results showcases how good the conformal classifier is for segmenting prostates in MRI-images. While the amount of uncertainties should be improved upon, the fact that it predicts uncertainties can increase trust and usability for the clinicians.

This project is to confirm if the conformal predictions is better than an 5-fold ensemble, as it is cheaper to train and doesn't take as much time as an 5-fold nnUNet ensemble. The study shows the potential of the model when it is certain about its predictions. With the DSC above 0.9 on both the prostateX and SUS dataset and a DSC above 0.7 on every dataset except the P158 patients with cancer. The results shows that conformal classifier performs better than the 5-fold ensemble.

The biggest outlier is the P158 dataset, as it seems the models have much more difficulty to predict an accurate segmentation of these MRI's compared to the the other two datasets. The most probable theory is that the classifiers struggle more with the two zones the MRI have, as the original GTM's segments the points into either 0, 1 or 2 for no prostate and two different zones of the prostate. When the nnUNet and conformal classifier tries to predict these, it might struggle as it is very different from what it is trained with. The only difference is with the QC models, as the uncertainties gets changed into correct labels.

Every dataset has some outliers, as the HD never approximates to 0, while most cases are 5 or above. While most of these may just be one single point in the image, and wont really apply to the prediction of the prostate gland, it showcases that the model doesn't predict perfectly. Yet, this one metric doesn't decide if the model is bad or not, but need to be looked at with the other metrics first. The combination of conformal classifier and quality control showcases the true potential of this model, as all datasets gets a DSC over 0.9 when quality control is applied.

6.2 Future Directions

Based on the results and development, we consider the work to have fulfilled the most important objectives of this project. To be able to compare an 5-fold ensemble model and single model with conformal predictions, showing that the conformal classifier works better than an 5-fold ensemble.

For future additions to this project, implementing a more broad quality control system is recommended. The current quality control is just a comparison method, to see how the results change if all the uncertain predictions turn into the correct label, simulating a situation where an expert needs to check these points. Realistically, it would not be productive of the experts to check every segmentation to fix what the model predicted to be uncertain.

A future quality control should check some defined metric, or maybe even check the efficiency/validity of the predictions, and through these metrics, we can decide on what action should be taken. An idea might be to decide to apply human intervention after checking the validity and efficiency, to try to simulate an expert when the model can't predict properly.

There may be other methods to create a quality control, as the focus of the project was about the comparison between 5-fold ensemble and conformal classifier. There are many possibilities for quality control that are not accounted for in this project, which might potentially improve the segmentation's even further.

Appendix A

Poster

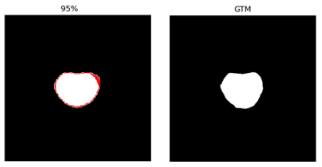
Quantifying uncertainty in nnUNet-segmentations

Introduction

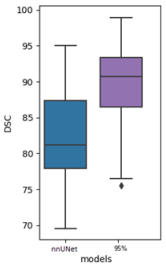
- Deep learning in medical fields
- Segmentations, U-net
- Lacks certainty
- Safer predictions.

Data and methods

- ProstateX
  - 204-patients
  - Clinically significant
- nnUNet for segmentations
- Quantify uncertainty
- Compare results against each other



Model	DSC	HD	ASD	RVD	ABS
nnUNet	0.8197 ± 0.0619	7.8389 ± 48219	0.5915 ± 0.1723	1.56 ± 9.03	99.24 ± 8.83
95%	0.8990 ± 0.0585	8.8286 ± 5.7431	0.1588 ± 0.1270	-1.51 ± 4.33	101.73 ± 4.51



Conclusion

- Quantifying uncertainty improves results
- Gives safer enviroment by classifying as uncertain

Alvaro Fernandez-Quilez, Tobias Nordstom, Trygve Eftestol, Andreas Bremset Alvestad, Fredrik Jaderling, Svein Reidar Kjosavik, and Martin Eklund. Revisiting prostate segmentation in magnetic resonance imaging (mri): On model transferability, degradation and pi-rads adherence. medRxiv, pages 2023–08, 2023.

Kevin Mekhaphan Nguyen. Uncertainty quantification in prostate segmentation. Master's thesis, uis, 2023.

Isensee, F., Jaeger, P. F., Kohl, S. A., Petersen, J., & Maier-Hein, K. H. (2021). nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. Nature methods, 18(2), 203-211.

Figure A.1: Poster from poster presentation

Bibliography

[1] Hamid A Jalab and Ali M Hasan. Magnetic resonance imaging segmentation techniques of brain tumors: A review. *Archives of Neuroscience*, 6(Brain Mapping), 2019.

[2] Binyan Hu and AK Qin. Deep learning for medical image segmentation with imprecise annotation. *arXiv preprint arXiv:2402.07330*, 2024.

[3] Shuo Li. An introduction to conformal prediction. https://towardsdatascience.com/conformal-prediction-4775e78b47b6, 2021. Last updated 20/08/2021.

[4] Kevin Mekhaphan Nguyen. Uncertainty quantification in prostate segmentation. Master's thesis, uis, 2023.

[5] Cleveland clinic. Prostate, 2022. URL https://my.clevelandclinic.org/health/body/23965-prostate#overview. [Online; accessed 04-June-2024].

[6] Wikipedia contributors. Magnetic resonance imaging — Wikipedia, the free encyclopedia, 2024. URL https://en.wikipedia.org/w/index.php?title=Magnetic_resonance_imaging&oldid=1219184259. [Online; accessed 2-May-2024].

[7] Wikipedia contributors. Ct scan — Wikipedia, the free encyclopedia, 2024. URL https://en.wikipedia.org/w/index.php?title=CT_scan&oldid=1225573559. [Online; accessed 4-June-2024].

[8] Wikipedia contributors. Feedforward neural network — Wikipedia, the free encyclopedia, 2024. URL https://en.wikipedia.org/w/index.php?title=Feedforward_neural_network&oldid=1221157607. [Online; accessed 4-June-2024].

[9] Wikipedia contributors. Convolutional neural network — Wikipedia, the free encyclopedia, 2024. URL https://en.wikipedia.org/w/index.php?title=Convolutional_neural_network&oldid=1220698821. [Online; accessed 2-May-2024].

[10] Wikimedia Commons. File:3 filters in a convolutional neural network.gif — wikimedia commons, the free media repository, 2024. URL https://commons.wikimedia.org/w/index.php?title=File:3_filters_in_a_Convolutional_Neural_Network.gif&oldid=845744467. [Online; accessed 2-May-2024].

[11] GeeksforGeeks. Introduction to convolution neural network, 2024. URL https://www.geeksforgeeks.org/introduction-convolution-neural-network/. [Online; accessed 2-May-2024].

[12] Wikipedia contributors. Image segmentation — Wikipedia, the free encyclopedia, 2024. URL https://en.wikipedia.org/w/index.php?title=Image_segmentation&oldid=1222875710. [Online; accessed 5-June-2024].

[13] Wikipedia contributors. U-net — Wikipedia, the free encyclopedia, 2024. URL https://en.wikipedia.org/w/index.php?title=U-Net&oldid=1215987653. [Online; accessed 2-May-2024].

[14] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[15] Wikipedia contributors. Upsampling — Wikipedia, the free encyclopedia, 2024. URL https://en.wikipedia.org/w/index.php?title=Upsampling&oldid=1226490716. [Online; accessed 5-June-2024].

[16] Wikipedia contributors. Rectifier (neural networks) — Wikipedia, the free encyclopedia, 2024. URL https://en.wikipedia.org/w/index.php?title=Rectifier_(neural_networks)&oldid=1221547062. [Online; accessed 5-June-2024].

[17] Fabian Isensee, Paul F Jaeger, Simon AA Kohl, Jens Petersen, and Klaus H Maier-Hein. nnu-net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature methods*, 18(2):203–211, 2021.

[18] F Isensee and the rest of the contributors. Welcome to the new nnu-net! https://github.com/MIC-DKFZ/nnUNet, 2023. Last updated 25/09/2023.

[19] Xiang J. A brief introduction to uncertainty calibration and reliability diagrams. https://towardsdatascience.com/introduction-to-reliability-diagrams-for-probability-calibration-ed785b3f5d44, 2020. Last uvisited 14/05/2024.

[20] Henrik Olsson, Kimmo Kartasalo, Nita Mulliqi, Marco Capuccini, Pekka Ruusuvuori, Hemamali Samaratunga, Brett Delahunt, Cecilia Lindskog, Emiel AM Janssen,

Anders Blilie, et al. Estimating diagnostic uncertainty in artificial intelligence assisted pathology using conformal prediction. *Nature communications*, 13(1):7761, 2022.

[21] The open group. The unix® operating system: A robust, standardized foundation for cluster architectures, 2020. URL https://unix.org/whitepapers/cluster.html. [Online; accessed 4-June-2024].

[22] Jayachander Surbiryala . Slurm quickstart, 2023. URL https://gitlab.ux.uis.no/unix/gpu/-/blob/main/README.md. [Online; accessed 4-June-2024].

[23] Anaconda . Miniconda, 2024. URL https://docs.anaconda.com/free/miniconda/index.html. [Online; accessed 4-June-2024].

[24] Wikipedia contributors. Anaconda (python distribution) — Wikipedia, the free encyclopedia, 2024. URL https://en.wikipedia.org/w/index.php?title=Anaconda_(Python_distribution)&oldid=1225512987. [Online; accessed 4-June-2024].

[25] Wikipedia contributors. Conda (package manager) — Wikipedia, the free encyclopedia, 2024. URL https://en.wikipedia.org/w/index.php?title=Conda_(package_manager)&oldid=1224754148. [Online; accessed 4-June-2024].

[26] OECD. Dice score. https://oecd.ai/en/catalogue/metrics/dice-score, 2024. Last visited 10/05/2024.

[27] Wikipedia contributors. Hausdorff distance — Wikipedia, the free encyclopedia, 2024. URL https://en.wikipedia.org/w/index.php?title=Hausdorff_distance&oldid=1213656182. [Online; accessed 10-May-2024].

[28] Wikipedia contributors. Infimum and supremum — Wikipedia, the free encyclopedia, 2024. URL https://en.wikipedia.org/w/index.php?title=Infimum_and_supremum&oldid=1212840409. [Online; accessed 4-June-2024].

[29] Varduhi Yeghiazaryan and Irina Voiculescu. Family of boundary overlap metrics for the evaluation of medical image segmentation. *Journal of Medical Imaging*, 5(1): 015006–015006, 2018.

[30] Wikipedia contributors. Brier score — Wikipedia, the free encyclopedia, 2024. URL https://en.wikipedia.org/w/index.php?title=Brier_score&oldid=1194280415. [Online; accessed 15-May-2024].

[31] Maja Pavloic. Expected calibration error (ece): A step-by-step visual explanation. https://towardsdatascience.com/expected-calibration-error-ece-a-step-by-step-visual-explanation-with-python-cod 2023. Last uvisited 14/05/2024.

[32] Alvaro Fernandez-Quilez, Tobias Nordstom, Trygve Eftestol, Andreas Bremset Alvestad, Fredrik Jaderling, Svein Reidar Kjosavik, and Martin Eklund. Revisiting prostate segmentation in magnetic resonance imaging (mri): On model transferability, degradation and pi-rads adherence. *medRxiv*, pages 2023–08, 2023.

[33] Lisa C Adams, Marcus R Makowski, Günther Engel, Maximilian Rattunde, Felix Busch, Patrick Asbach, Stefan M Niehues, Shankeeth Vinayahalingam, Bram van Ginneken, Geert Litjens, et al. Prostate158-an expert-annotated 3t mri dataset and algorithm for prostate cancer detection. *Computers in Biology and Medicine*, 148: 105817, 2022.

[34] F Isensee and the rest of the contributors. nnu-net dataset format. https://github.com/MIC-DKFZ/nnUNet/blob/master/documentation/dataset_format.md, 2023. Last updated 05/12/2023.

[35] Geert Litjens, Oscar Debats, Jelle Barentsz, Nico Karssemeijer, and Henkjan Huisman. Prostatex challenge data. *The cancer imaging archive*, 10:K9TCIA, 2017.

[36] Henrik Boström. crepes: a python package for generating conformal regressors and predictive systems. In Ulf Johansson, Henrik Boström, Khuong An Nguyen, Zhiyuan Luo, and Lars Carlsson, editors, *Proceedings of the Eleventh Symposium on Conformal and Probabilistic Prediction and Applications*, volume 179 of *Proceedings of Machine Learning Research*. PMLR, 2022.

[37] Zachary Warnes. How to add uncertainty estimation to your models with conformal prediction. https://towardsdatascience.com/how-to-add-uncertainty-estimation-to-your-models-with-conformal-prediction-a5acdb86ea05 2021. Last uvisited 14/05/2024.

[38] Oskar Maier. Metric measures, 2024. URL https://loli.github.io/medpy/reference/metric.html. [Online; accessed 10-June-2024].

[39] Pavel Iakubovskii. Segmentation models. https://github.com/qubvel/segmentation_models, 2019.

[40] Mouhcine Mendil, Luca Mossina, and David Vigouroux. Puncc: a python library for predictive uncertainty calibration and conformalization. In *Conformal and Probabilistic Prediction with Applications*, pages 582–601. PMLR, 2023.