Ref. Ares(2022)6752235 - 30/09/2022

A European Cancer Image Platform Linked to Biological and Health Data for Next-Generation Artificial Intelligence and Precision Medicine in Oncology

Deliverable D6.2:

Methods to assess errors and procedures to address uncertainty

Reference	D6.2_ EuCanImage_UB_v1
Lead Beneficiary	UB
Author(s)	Kaisar Kushibar, Zohaib Salahuddin
Dissemination level	Public
Type	Report
Official Delivery Date	30 September 2022
Date of validation of the WP leader	30 September 2022
Date of validation by the Project Coordinator	30 September 2022
Project Coordinator Signature	

EuCanImage is funded by the European Union's H2020 Framework
Under Grant Agreement No 952103

1. Version log

Issue Date	Version	Involved	Comments
26/09/2022	V0.1	Kaisar Kushibar	Initial draft
28/09/2022	V0.2	Zohaib Salahuddin, Anais Emelie	Revision and correction
30/09/2022	V1	Anais Emelie & Karim Lekadir	Revised and corrected final version.

2. Executive Summary

In this deliverable, we present the progress on task 6.2 "Addressing bias, uncertainty and error in AI solutions for cancer imaging" that started on M12 and will be finalised on M36. Overall, the task has been split into two sub-tasks: 1) error and bias assessment; and 2) uncertainty estimation. A remarkable progress has been made on the second task as the problem does not depend on the availability of carefully curated and detailed annotated data from the clinical partners. Commonly used state of the art approaches for the uncertainty estimation have been reviewed, implemented, and analysed for the breast cancer use case of the EuCanImage project. As for the first sub-task, although the clinical data is not available, great progress has been achieved in collaboration with the AI and Clinical working groups to define the requirements, evaluation metrics, and identification of potential biases not only on macro factors such as demographics, socio-economic status, etc., but also on micro level variables such as biological reports. A framework of an action plan has been designed to carry out error and bias assessment. In particular, validation, data separation, benchmarking, and visualisation in the OpenEBench platform. Further progress on error assessment will be carried out as the data from the clinical partners become available until the end of the task timeline.

Table of Contents

Acronyms

Name	Abbreviation
Artificial Intelligence	AI
Virtual Research Environment	VRE
Out of distribution	OOD
Convolutional Neural Network	CNN
Layer Ensembles	LE
Deep Ensembles	DE
Monte-Carlo Dropout	MCDropout
Stein Variational Gradient Descent	SVGD
Stochastic Weight Averaging Gaussian	SWAG
Bayes by backprop	BBB
Breast Cancer Digital Repository	BCDR

1 Introduction

In this document, we present the completed and the ongoing progress as well as future work on error assessment and uncertainty estimation in Artificial Intelligence (AI) developed within the EuCanImage project. Throughout the timeline of this task (T6.2), the following aspects have been researched and methodologies have been proposed. First, the importance of the task was investigated by looking into the areas where the modern AI failures could be alarming, especially in healthcare. Secondly, we identified the methods for error assessment and estimating the uncertainties that are common in AI-based decision-making processes. Moreover, the sources of uncertainties were investigated. Thirdly, approaches for detection of these errors and uncertainties have been explored. The common approaches from the literature were reviewed and their advantages and drawbacks were identified. Lastly, uncertainty and error elimination methods are currently under investigation, which will be completed and tested with the data collected from the clinical partners by the end of the designated timeline of M36 of this task.

1.1 Importance of error and uncertainty assessment in AI

The surge of Artificial Intelligence (AI) based methods in healthcare has brought a new spectrum of applications that re-identified the approaches tackled in Medical Image Analysis. Decision making models are usually trained in a supervised manner with expert annotated data. These models learn complex functions that map imaging and/or non-imaging features extracted from clinical data. The learned functions serve as a decision-making process for a given new data point. For complex problems that involve feature extraction from radiological images (i.e., radiomics) the feature space is usually high dimensional. For example, the radiomics feature extraction tool (presented in D5.2) extracts a minimum of 105 features from a region of interest (e.g., lesion, anatomical structure, etc.) categorised into shape, intensity, and texture (Figure 1).

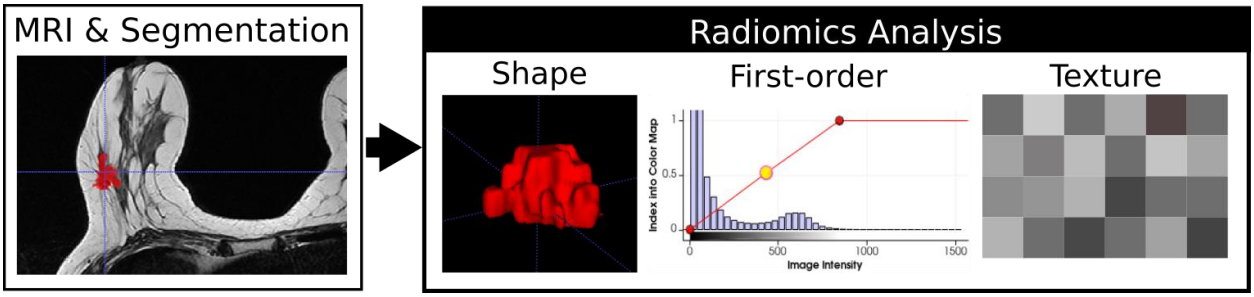

Figure 1: Overview of the three types of radiomics features extracted from Breast MRI images and corresponding lesion segmentation masks.

These radiomics features are then put into an N-dimensional vector that acts as a point in the N-dimensional hyperspace. The purpose of AI is to learn an arbitrary function that can separate different classes (e.g., benign vs malignant tumours) in the hyperspace so that for the new N-dimensional points the function is still able to identify to which class it belongs (Figure 2).

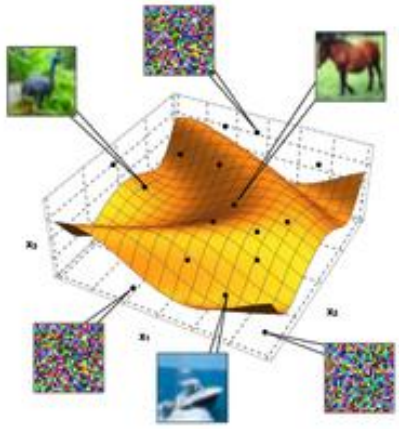

Figure 2: Supervised learning in high-dimensional feature space to create a decision function that can separate different classes. Each point is an N-dimensional feature vector describing a single data point. The orange sheet is a learned function that separates the data points into different classes. Image courtesy of [1].

There is no closed form solution to identify the mapping function and the number of solutions can be infinite. Taking this into account, the AI solutions are error-prone as it is not possible to learn a function that can give a correct solution at all times. Therefore, it is crucial to identify the ways to detect, assess, and eliminate errors in AI models, particularly, in healthcare applications.

In the initial stages of development we assess errors with the ground truth on retrospective data collected during the EuCanImage project. However, for the data that has no ground truth, e.g. in real clinical applications or prospective use-cases, it is not possible to perform such an assessment. Although the initial error assessment done in EuCanImage is crucial, further error detection on new unseen data (e.g. from new centres) helps to maintain actuality and continuous development to the EuCanImage platform. In this case, uncertainty estimation in AI comes to hand as a tool to provide confidence measures to the users (clinicians) for all its decisions as well as detecting potential erroneous outputs by AI tools. In the following sections we describe the two paradigms – error assessment and uncertainty estimation – from retrospective and prospective points of views.

2 Methods

As aforementioned, the error assessment and uncertainty estimation are done: 1) with the data collected during the project lifetime with corresponding ground truths; and 2) for the AI models on existing as well as potential new data that may not have ground truths, respectively. In the following sections we describe the pipelines and approaches for each step.

2.1 Methods for error assessment

The overall framework for error assessment is shown in Figure 3. It contains three interconnected parts: 1) Data storage (WP3); 2) AI Virtual Research Environment – AI-VRE (WP5); and 3) OpenEBench – benchmarking platform (WP6).

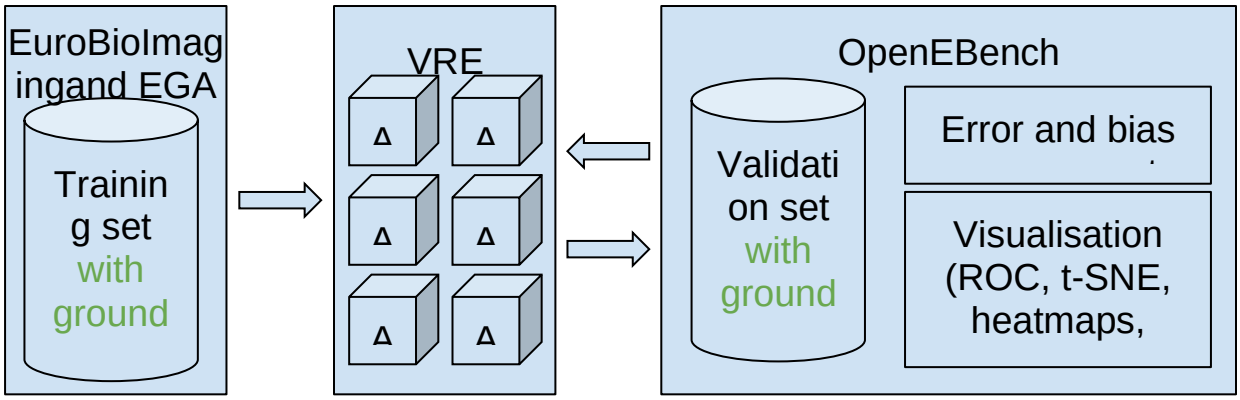

Figure 3: Pipeline for error and bias assessment on retrospective unseen data with ground truth.

The training set includes imaging and non-imaging data with corresponding ground truths for the use-cases of EuCanImage stored in EuroBioImaging Archive and European Genome-phenome archive. The Virtual Research Environment (VRE) hosts AI algorithms for each use-case trained with the data from the data storage. The VRE tools are validated on hold-out sets and benchmarked in the OpenEBench platform for error and bias assessment.

For fair assessment of AI models, a validation set is separated from the collected data that represent real-world distribution in terms of cancer prevalence, imaging modalities, and protocols. In collaboration with the clinical and AI working groups (WG2 and WG5), relevant evaluation metrics and variables that pose potential bias are identified. Table 1 shows the categorised metrics per objective that could be classification, segmentation, detection (spatial localisation using bounding boxes), explainability, and uncertainty. Currently, more discussion is on process to identify more precise requirements per use cases. For example, in breast cancer classification from screening mammograms (use-case 8), answering questions such as "Is it required to localise lesions or indicating the lesion presence is enough?" or "Is detection of two adjacent lesions as a single lesion should be penalised during evaluation?". Such clarifications will help AI developers to prioritise metrics that are really relevant in clinical practice.

The initial set of biases are defined per use case as shown in Table 2. More analysis to enlist biases for the clinical data will be performed when the data annotation and collection process is complete. However, the process workflow is already specified in collaboration with the AI, Validation, and Clinical working groups as shown in Figure 4 and this framework will be used as a basis for the bias and error detection, analysis, and mitigation. In some cases, where the bias mitigation is not possible, the AI model will be flagged for these biases to warn the users for potentially unfair treatment of some samples with specific characteristics.

Table 1: Overview of general evaluation metrics for Classification, Segmentation, Detection, Explainability, and Uncertainty.

General Classification Metrics	**General Segmentation Metrics**	**Detection**	**Explainability**	**Uncertainty**
TPR/Sensitivity/Recall	Dice Index	Intersection Over Union (IoU)	Qualitative Assessment	Negative Log Likelihood

TNR/Specificity	Surface Dice Index	Boundary Intersection Over Union (Uncertainity Aware)	System Causability Scale (SCS)	Brier's score
PPV/Precision	Jaccard Index	False Positives Per Image	*In silico* trial for usefulness of explanations in clinical practice	Entropy
NPV	Hausdorff Distance	FROC Curve (AUC-FROC)		Mutual Information
Accuracy	Hausdorff Distance 95 percentile	Average Precision at various thresholds (alpha= 0.1 to 0.75)		Variance
F1 Score	Average Symmetric Surface Distance	Sensitivity at various thresholds (alpha = 0.1 to 0.75)		Calibration Curves
Balanced Accuracy	Normalised Surface Distance			
Cohen's Kappa	Modified Hausdorff Distance			
Weighted Cohen's Kappa	Average Distance (2D)			
Matthews Correlation Coefficient				
AUC Receiver Operating Characteristic Curve				
AUC Precision Recall Curve				

Table 2: Overview of potential biases per use case identified within the AI Working Group.

Use-case	**AI WG Biases**
1.Liver diagnosis (CT) •Classification (benign/malignant) ?	•Age •Geography •Lesion size •Differences in the image acquisition •Coexistence of HCC with other abnormalities (like hemangioma)
2. Liver Diagnosis (MRI) •Classification (benign/malignant) ?	•Age •Geography •Lesion size •Differences in the image acquisition •coexistence of HCC with other abnormalities (like hemangioma)
3. Colorectal metastasis detection (CT) •Detection (localization)	•Age •Geography •Lesion size •Differences in image acquisition. •coexistence of metastasis with other abnormalities (like hemangioma)
4. Mesorectal lymph node metastasis identification (MRI) •Detection (localization) •Classification (metastasis present, not present)	•Age •Geography •Differences in the image acquisition • presence of other types of lymph nodes
5. Therapy response prediction based on primary imaging (for staging and restaging) (MRI) •Classification (no response, partial response, complete response)	•Age •Geography •Lesion size •Differences in the image acquisition •presence of other pathology in the pelvic region
6. Molecular subtype classification in invasive ductal breast carcinoma (MG) •Classification (Luminal A, Luminal B, HER2 positive, triple negative)	•Age •Geography •Lesion size •Breast composition •Differences in the image acquisition protocols •Aesthetic Implants •Occult lesions
7. Treatment Response Prediction (Breast MRI) •Segmentation (Instance) •Classification (no response, partial response, complete response)	•Age •Geography •Lesion size •Image acquisition differences
8. Breast Screening (MG) •Segmentation (Instance) •Classification (normal, benign, malignant)	•Breast composition •Age •Geography •Lesion size •Differences in the image acquisition protocols •Aesthetic Implants •Occult lesions

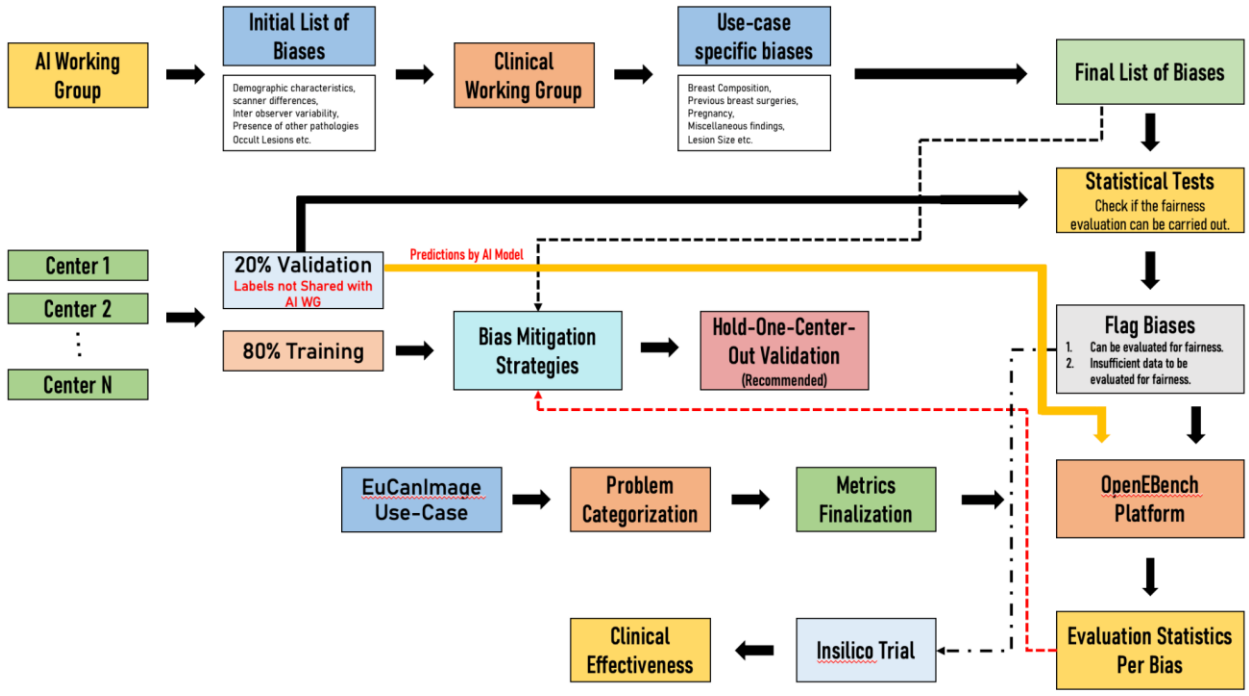

Figure 4: Bias and error assessment, visualisation, and mitigation framework.

Thorough error analysis is often more efficient by visualisation. Therefore, the OpenEBench platform is under development for the EuCanImage use cases. Different visualisations based on evaluation metrics defined in Table 1 can be useful to evaluate the algorithms. However, more systematic approaches are required to analyse the error occurrences. Inspired by the open source "responsible AI toolbox" by Microsoft [2], we will incorporate error assessment dashboards to the OpenEBench platform. The following techniques will be included:

1) Decision Tree: Discover subjects with high error rates across multiple features using the binary tree visualisation. This helps to investigate indicators such as error rate, error coverage, and data representation for each discovered cohort.

2) Error Heatmap: Once the hypotheses are formed on the most impactful features for failure, using the Error Heatmap helps to further investigate how one or two input features impact the error rate across subjects.

After identifying subjects with higher error rates, one can debug and explore these subjects further. Furthermore, data exploration and model explanation can be useful to gain deeper insights about the model or the data. The model explanation and interpretability is addressed in T5.6 and is expected to be completed by M48.

2.2 Methods for uncertainty estimation

As it was shown in Figure 1, the error assessment is done with the data collected from the EuCanImage clinical partners with their corresponding ground truths. Although this analysis is useful for benchmarking and error correction in the AI tools deployed in the VRE, there must be a mechanism that detects anomalies for the new cases without ground truths. The pipeline for such a scenario is illustrated in Figure 5. Uncertainty estimation is a good

procedure to employ for analysing the failure cases, detecting out-of-distribution (OOD) samples, and providing confidence measures to the AI prediction.

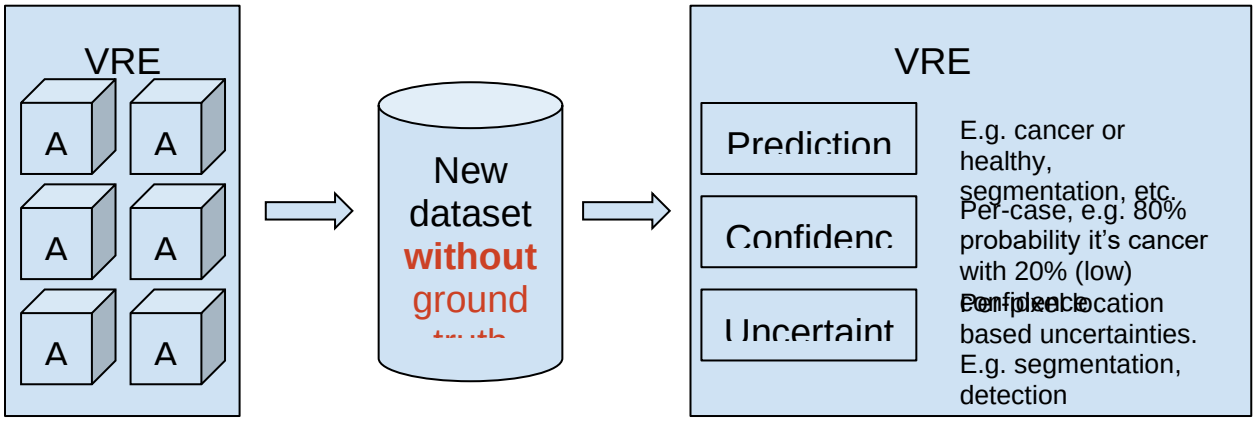

Figure 5: Pipeline for uncertainty estimation on prospective unseen data without ground truths.

Since the beginning of the task, we have done a deep analysis of the state-of-the-art methods for enabling uncertainty estimation in AI. In particular, the applications of Deep Learning (DL) as they are notoriously known for their overconfident predictions and silent failures for OOD data. In the following sections we describe the types and sources of uncertainties in DL, state-of-the-art techniques to estimate uncertainties, and the EuCanImage developed tool that mitigates the drawbacks of the existing approaches.

2.2.1 Types of uncertainties in AI

There are two types of uncertainties: 1) epistemic – inherent to the model; and 2) aleatoric – inherent to the noise in data. The epistemic uncertainty arises from the lack of knowledge in the model, i.e. due to the unseen data during training. Hence, this type of uncertainty can be fully eliminated if the model is trained with an infinite amount of data. The aleatoric uncertainty has two sub-categories: 1) heteroscedastic – corrupt data such as motion artefacts or radiofrequency spikes in MRI; 2) homoscedastic – constant noise that is caused by the sensor. The aleatoric uncertainty cannot be eliminated but can be detected. In EuCanImage, we mainly focused on epistemic uncertainty estimation as it is directly linked to detecting failure cases and providing confidence intervals on new test subjects.

2.2.2 State of the art approaches and their limitations

Figure 6 shows different categories of uncertainty estimation methods reviewed during this task. They are divided mainly into two categories based on their fundamental differences.

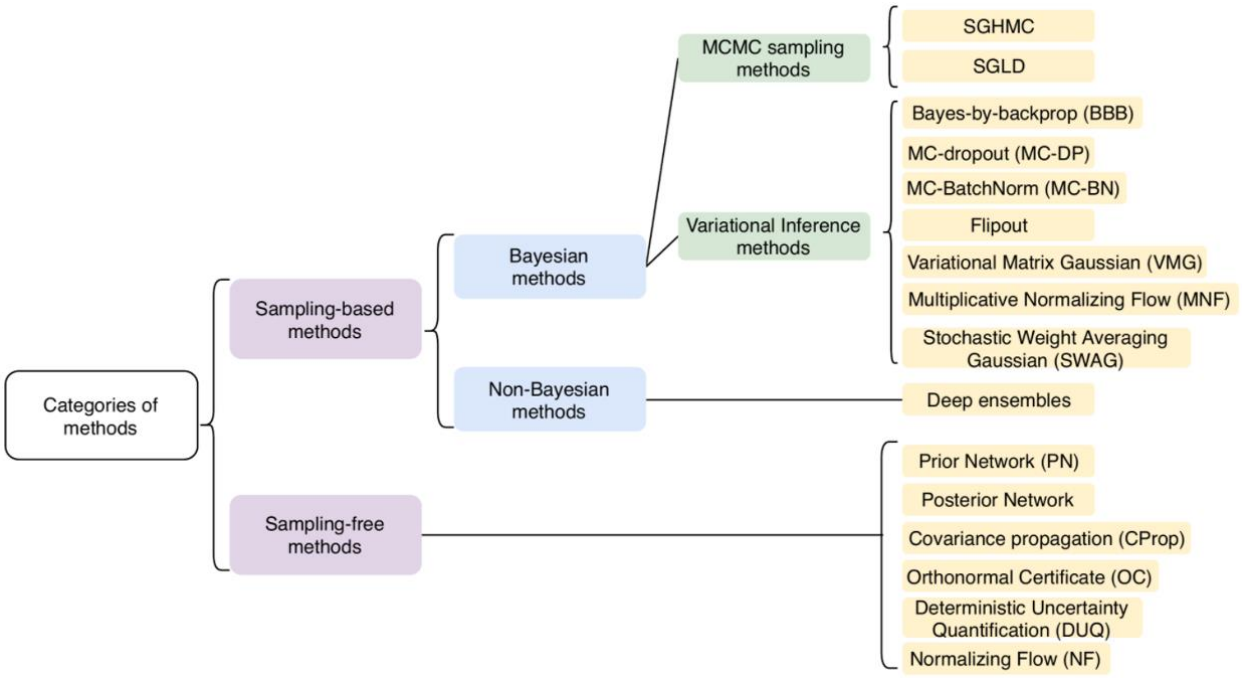

Figure 6: Uncertainty estimation methods in the literature categorised into different classes of techniques.

The main difference of these DL based techniques from the regular DL methods is that the latter ones are deterministic where each parameter in the network is a point estimate (a single number). The deterministic nature of the methods does not allow estimating the model uncertainties. Therefore, the research focus has gone towards building stochastic (Bayesian) neural networks.

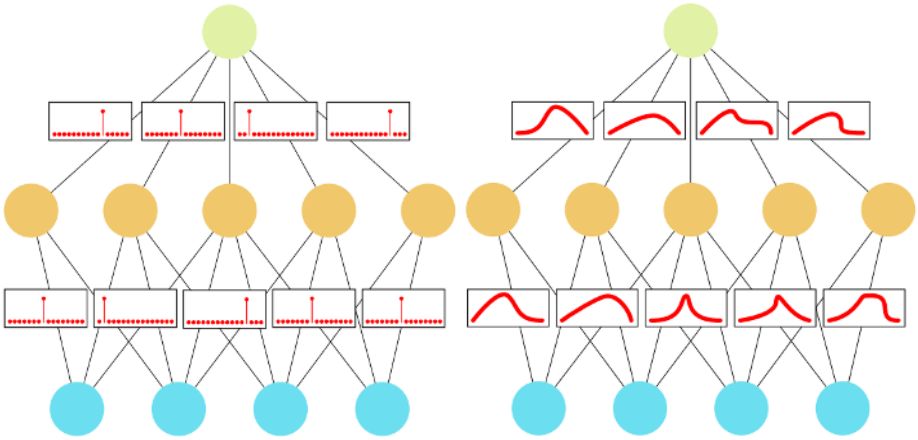

Figure 7: On the left: deterministic neural network with weights described as point estimates. On the right: stochastic neural network where the weights are distributions characterised by a tractable probability distribution family (usually Gaussian). Figure courtesy of [8]

Overall, the main idea of uncertainty estimation in deep learning is to learn the posterior distribution of the weights given a training dataset (Figure 7).

From the methods in the literature, we have evaluated the commonly used Bayesian and non-Bayesian approaches such as Monte-Carlo Dropout (MCDropout), Stochastic Weight Averaging Gaussian (SWAG), Bayes-by-Backprop (BBB), Stein Variational Gradient Descent (SVGD), and Deep Ensembles (DE) [3, 4, 5, 6, 7]. We assessed their advantages and limitations in the breast cancer use case (UC8), specifically, for breast mass segmentation in mammograms. Table 3 summarises the advantages and limitations of the commonly used state of the art methods for uncertainty estimation in DL.

Table 3: Advantages and limitations of commonly used state of the art uncertainty estimation techniques in the literature.

Method	**Advantages**	**Limitations**
MCDropout	Simple, scalable	Multi-pass, hyperparameter tuning is required
SWAG	Intuitive, elegant formulation	Multi-pass, slow convergence, data hungry
BBB	Good calibration, solid mathematical background	Multi-pass, slow convergence, data hungry
SVGD	Intuitive, elegant formulation, best calibration	Multi-network, parallel training, slow convergence
DE	Best calibration, intuitive, scalable	Multi-network

As can be seen, most methods require multiple passes at test time to get final prediction and to estimate the uncertainties. Slow convergence and data hungry nature of some methods make them impossible to use in medical image analysis where the data annotation process is costly. The best method currently available is the DE that requires training multiple networks, which is inefficient. Taking these into account, we proposed a new framework that takes all the advantages and eliminates the limitations of these commonly used techniques.

2.3 Proposed uncertainty estimation method

Our method is called Layer Ensembles (LE) and it is inspired by the state of the art DE [7] for uncertainty estimation as well as a more recent work [9] that estimates example difficulty through prediction depth. We introduced how LE can be used to obtain a single image-level uncertainty metric that is more useful for some tasks compared to the commonly used pixel-wise variance, entropy, and mutual information (MI) metrics.

In LE, we attach a prediction head after each layer output in the network as shown in Figure 8. We used a CNN following the U-Net architecture with different modules in the decoder and encoder blocks. LE is architecture agnostic and the choice of U-Net was due to its wide use and high performance on different medical imaging tasks.

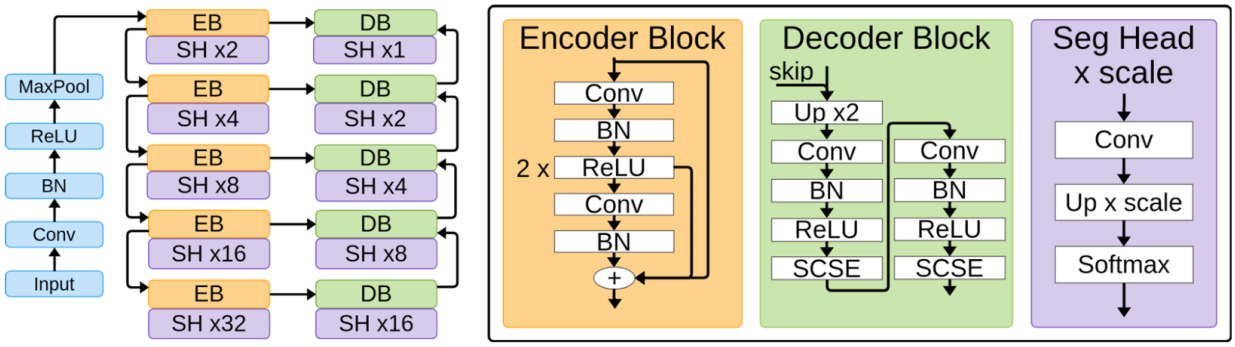

Figure 8: Layer Ensembles framework. In this example, LE is built on top of U-Net like architecture. Encoder Block (EB) in orange and Decoder Block (DB) in green have internal structures as depicted in the boxes below with corresponding colours. Ten Segmentation Heads (SH) are attached after each layer output with an up-scaling factor depending on the depth of the layer. SCSE - Squeeze and Excitation attention module. BN - Batch Normalisation.

As it was mentioned, DE has been used widely in the literature for epistemic uncertainty estimation. The original method assumes a collection of M networks with different initialisation trained with the same data. Then, the outputs of each of these M models can be used to extract uncertainty measurements (e.g. variance). As we have shown in Figure 8, ten segmentation heads were added after each layer. Then, LE is a compound of M sub-networks of different depths. Since each of the segmentation heads is randomly initialised, it is sufficient to cause each of the sub-networks to make partially independent errors. The outputs from each of the segmentation heads can then be combined to produce final segmentation and estimate the uncertainties, similarly to DE. Hence, LE is an approximation to DE, but using only one network model.

LE can also be viewed as stacked networks where the parameters of a network f_t is shared by f_{t+1} for all t in [0, N), where N is the total number of outputs. This sequential connection of sub-networks allows us to observe the progression of segmentation through the outputs of each segmentation head. We can measure the agreement between the adjacent layer outputs – e.g. using the Dice coefficient – to obtain a layer agreement curve. Depending on the network uncertainty, the agreement between layers will be low, especially in the early layers (Figure 9). We proposed to use the Area Under Layer Agreement curve (AULA) as an image-level uncertainty metric. Figure 10 demonstrates that AULA is a good uncertainty measure to detect poor segmentation quality by evaluating the fraction of remaining images with poor segmentation after a fraction of poor quality segmentation images are flagged for manual correction. We considered DSCs below 0.90 as poor quality.

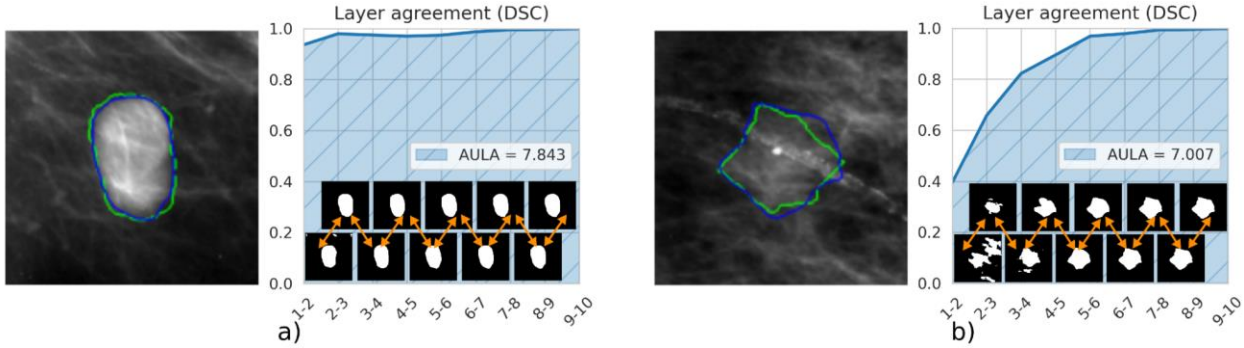

Figure 9: Layer Agreement curve. a) A high contrast lesion: large AULA and low uncertainty. b) A low contrast lesion and calcification pathology is present: small AULA and higher uncertainty. Arrows represent the correspondence between layers 1 and 2, 2 and 3, etc. DSC -- Dice Similarity Coefficient. Green contours are ground truths.

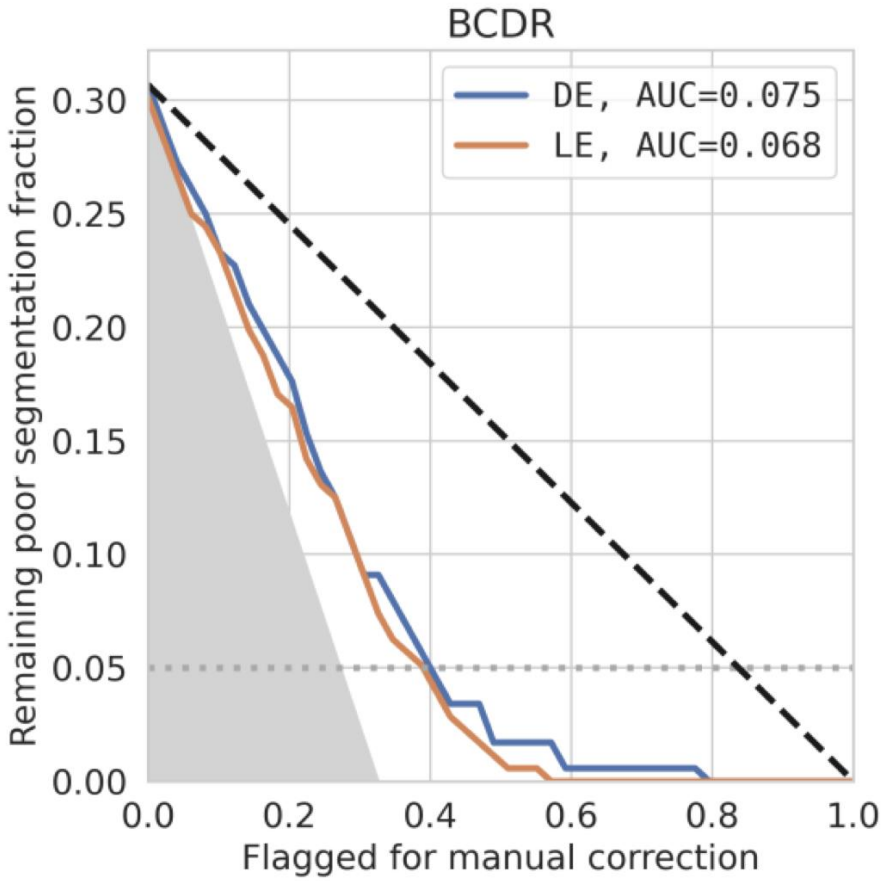

Figure 10: Segmentation quality control for DE and LE. The following are averaged indicators for: random flagging (dashed black); remaining 5% of poor segmentations (dotted grey); and ideal line (grey shaded area).

Table 4 compares the segmentation performance of LE with DE, and Plain models in terms of Dice Similarity Coefficient (DSC) and Modified Hausdorff Distance (MHD). Two-sided paired t-test is used to measure statistically significant differences with $\alpha = 0.05$. LE performs similarly to DE for both DSC and MHD metrics and significantly outperforms a deterministic counterpart model (Plain). The NLL (calibration performance metric) of LE is significantly better compared to others ($p < 0.001$).

*Table 4: Segmentation and confidence calibration performance for Plain U-Net, DE, and LE on publicly available BCDR mammogram dataset. The values for Dice Similarity Coefficient (DSC), Modified Hausdorff Distance (MHD), and Negative Log-Likelihood (NLL) are given as mean(std). The best values are in bold. Statistically significant differences compared to LE are indicated by *.*

	BCDR – breast mass segmentation		
Method	DSC ↑	MHD ↓	NLL ↓
Plain	*0.865(0.09)	*1.429(1.72)	*2.312(1.35)
DE	0.870(0.09)	1.373(1.76)	*0.615(0.54)
LE	**0.872(0.084)**	**1.317(1.692)**	**0.306(0.25)**

Moreover, as can be seen in Figure 11, DE's uncertainty maps are overconfident, while LE manages to highlight the difficult areas. We believe that having such meaningful heatmaps is more helpful for the clinicians (e.g. for manual correction).

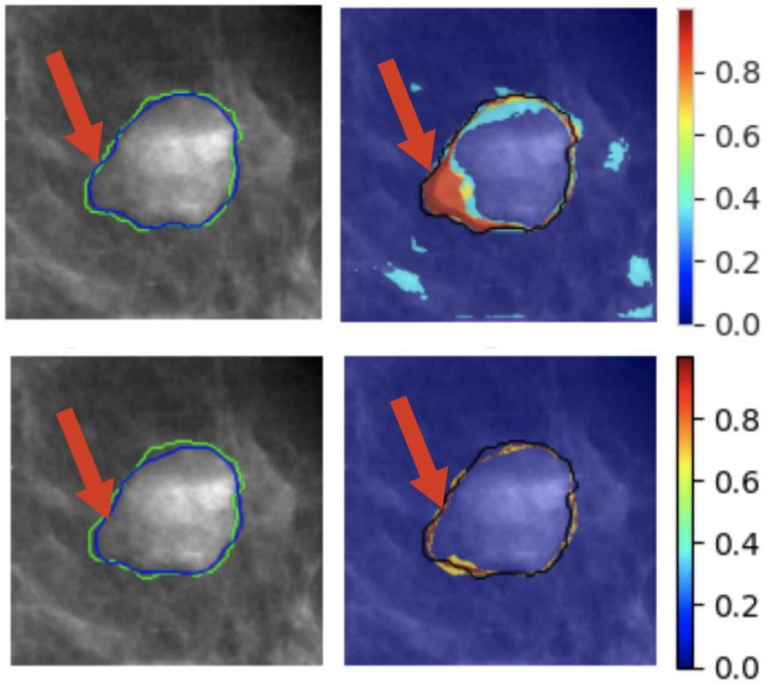

Figure 11: Examples of visual uncertainty heatmaps based on variance for high uncertainty areas (red arrows) using LE (top) and DE (bottom) for breast mass segmentation. Black and green contours correspond to ground truth.

The computational gain for LE compared to DE was substantial, both in training and testing due to the single network and single pass nature of LE. For training, we started measuring time after one epoch to let the GPU warm-up. Then, we captured the training (including backprop) and test times as seconds per batch. The averaged times were 0.99, 0.20, and 0.18 for DE, LE, and Plain, respectively. Similarly, the testing times were 0.240, 0.047, 0.045 for DE, LE, and Plain, respectively. These results show that LE allows much efficient training and testing compared to DE and a similar speed to the Plain approach.

2.4 Layer Ensembles Framework for building on top of other methods

We prepared an open-source tool available at https://github.com/pianoza/LayerEnsembles.

The steps are shown below:

1. Load any model

```
from torchvision.models import resnet18
architecture = resnet18(weights=None, num_classes=2)
```

2. Import the LayerEnsembles wrapper and the task Enum (e.g., segmentation, classification, regression)

```
from methods.layer_ensembles import LayerEnsembles
from utils import Task
```

3. Get the names of all the layers in your model

```
all_layers = dict([*architecture.named_modules()])
intermediate_layers = []
for name, layer in all_layers.items():
    if '.relu' in name:
        intermediate_layers.append(name)
```

The name signature (".relu") can be changed to any other component e.g., .bn or .conv the '.' is to include only sub-modules (exclude stem of the network).

4. Init LayerEnsembles with the names of the intermediate layers to use as outputs

```
model = LayerEnsembles(architecture, intermediate_layers)
# Dummy input to get the output shapes of the layers
x = torch.randn(1, 1, 128, 128)
output = model(x)
out_channels = []
for key, val in output.items():
    out_channels.append(val.shape[1])
# Set the output heads with the number of channels of the output layers
model.set_output_heads(in_channels=out_channels,task=Task.SEGMENTATION,
classes=2)
```

5. Check the output shapes

```
outputs = model(x)
print(len(outputs))
for layer, out in outputs.items():
    print(layer, out.shape)
```

6. Training goes as usual and the `outputs` is a dictionary with tensor values corresponding for each output head name as keys. Thus, we calculate the `total_loss` as the sum of each output head and then backpropagate.

```
model.train()
total_loss = 0
outputs = model(x)
losses = [criterion(output, target) for _, output in outputs.items()]
for loss in losses:
    total_loss = total_loss + loss
total_loss.backward()
optimizer.step()
```

The loss functions can be modified easily and the framework allows working around on how the total loss is calculated.

7. In testing, the output `list` contains predictions from each head. You can combine them in any way you like (e.g., averaging, STAPLE).

3 Future work

Visualisation and benchmarking tools will be developed in the VRE using the OpenEBench platform. The hold-out datasets collected during the EuCanImage project for validation will have bias-prone variables and evaluation metrics for each use-case. All the existing and future AI tools in the VRE will undergo thorough validation for error and bias detection using the approaches listed in Section 2.2. Error and uncertainty mitigation measures such as continual learning will be employed for the AI solutions that have been deemed to be unsatisfactory by the evaluation benchmark.

4 References

[1] Goldt, Sebastian, et al. "Modeling the influence of data structure on learning in neural networks: The hidden manifold model." *Physical Review X* 10.4 (2020): 041044.

[2] https://github.com/microsoft/responsible-ai-toolbox/blob/main/docs/erroranalysis-dashboard-README.md

[3] Gal, Yarin, and Zoubin Ghahramani. "Dropout as a bayesian approximation: Representing model uncertainty in deep learning." *international conference on machine learning*. PMLR, 2016.

[4] Maddox, Wesley J., et al. "A simple baseline for bayesian uncertainty in deep learning." *Advances in Neural Information Processing Systems* 32 (2019).

[5] Blundell, Charles, et al. "Weight uncertainty in neural network." *International conference on machine learning*. PMLR, 2015.

[6] Liu, Qiang, and Dilin Wang. "Stein variational gradient descent: A general purpose bayesian inference algorithm." *Advances in neural information processing systems* 29 (2016).

[7] Lakshminarayanan, Balaji, Alexander Pritzel, and Charles Blundell. "Simple and scalable predictive uncertainty estimation using deep ensembles." *Advances in neural information processing systems* 30 (2017).

[8] Jospin, Laurent Valentin, et al. "Hands-on Bayesian neural networks—A tutorial for deep learning users." *IEEE Computational Intelligence Magazine* 17.2 (2022): 29-48.

[9] Baldock, Robert, Hartmut Maennel, and Behnam Neyshabur. "Deep learning through the lens of example difficulty." *Advances in Neural Information Processing Systems* 34 (2021): 10876-10889.