Proceedings of Machine Learning Research 158:181–195, 2021 Machine Learning for Health (ML4H) 2021

Early Exit Ensembles for Uncertainty Quantification

Lorena Qendro* [1] LQ223@CL.CAM.AC.UK
Alexander Campbell* [1, 2] AJRC4@CL.CAM.AC.UK
Pietro Liò [1]
Cecilia Mascolo [1]
[1] *University of Cambridge, United Kingdom*
[2] *The Alan Turing Institute, United Kingdom*

Abstract

Deep learning is increasingly used for decision-making in health applications. However, commonly used deep learning models are deterministic and are unable to provide any estimate of predictive uncertainty. Quantifying model uncertainty is crucial for reducing the risk of misdiagnosis by informing practitioners of low-confident predictions. To address this issue, we propose early exit ensembles, a novel framework capable of capturing predictive uncertainty via an implicit ensemble of early exits. We evaluate our approach on the task of classification using three state-of-the-art deep learning architectures applied to three medical imaging datasets. Our experiments show that early exit ensembles provide better-calibrated uncertainty compared to Monte Carlo dropout and deep ensembles using just a single forward-pass of the model. Depending on the dataset and baseline, early exit ensembles can improve uncertainty metrics up to 2×, while increasing accuracy by up to 2% over its single model counterpart. Finally, our results suggest that by providing well-calibrated predictive uncertainty for both in- and out-of-distribution inputs, early exit ensembles have the potential to improve trustworthiness of models in high-risk medical decision-making.

Keywords: Uncertainty, Medical Imaging, Deep learning, Robustness, Early Exit, Out-Of-Distribution Detection

1. Introduction

Deep learning achieves state-of-the-art performance on a variety of tasks within the medical field such as classification (Esteva et al., 2017), segmentation (Perslev et al., 2019), and monitoring (Chan et al., 2019). For the majority of methods, however, the main focus is on improving accuracy without any consideration of predictive uncertainty. Particularly in medical imaging, uncertainty quantification is critical since the input distributions are often shifted from the training distribution due to different hardware and data collection protocols (Mårtensson et al., 2020; Amodei et al., 2016). In such scenarios, a model with well-calibrated uncertainty is able to indicate if a prediction should be trusted (Ovadia et al., 2019). Such a model could inform clinicians on how its performance may degrade in different deployment settings as well as when a human-in-the-loop is required to analyze uncertain samples (García Rodríguez et al., 2020; Xia et al., 2021).

Predictive uncertainty is defined as a probability distribution over multiple predictions on a single sample. Bayesian neural networks (BNNs) can naturally quantify such uncertainty via the estimation of the posterior over model weights using techniques such as variational inference (MacKay, 1992; Graves, 2011; Blundell et al., 2015). However, BNNs tend to be unstable and prohibitively slow to train, as well as parameter inefficient. More recently, Monte Carlo dropout uses dropout during inference as an efficient approximation to BNNs by creating an implicit ensemble of networks (Gal and Ghahramani, 2015). It is well known that ensembles of neural networks (NNs) improve prediction and uncertainty calibration (Hansen and Salamon, 1990). In particular deep ensembles (Lakshminarayanan et al., 2017), which train explicit ensemble members with different random initializations, have been shown to outperform approximate Bayesian methods (Ovadia et al., 2019). However, these approaches consist of training and running multiple models which can be unfeasible in real-world scenarios.

* These authors contributed equally

We propose a novel framework, early exit ensembles[1], that addresses the practical limitations of the aforementioned approaches for uncertainty quantification. The main contributions of our paper are:

- We provide a new interpretation of early exits as an implicit ensemble for uncertainty quantification. Our approach can be easily applied to any feed-forward deep learning architecture.
- We compare early exit ensembles to state-of-the-art deep learning uncertainty baselines in a series of experiments on the task of classification using real-world medical imaging datasets. Our evaluation includes data from different modalities (images and timeseries) as well as different model architectures (shallow and deep).
- Based on our experiments, early exit ensembles can provide significantly better uncertainty quantification compared to baselines such as Monte Carlo dropout and deep ensembles. On some datasets, negative log-likelihood, brier score and expected calibration error are improved up to 2×. Furthermore, on out-of-distribution data, early exit ensembles can achieve a 77% higher predictive entropy compared to the state-of-the-art.
- Finally, our approach enables training all ensemble members jointly in a single model, as well as providing uncertainty estimations from a single forward-pass of input data. Early exit ensembles therefore can be applied to a wide range of real-world applications in need of efficient uncertainty quantification.

2. Related Work

Uncertainty in deep learning. Several approaches exist for uncertainty quantification in deep learning. By setting a prior over model parameters, BNNs capture uncertainty via the estimation of the posterior distribution (MacKay, 1992; Blundell et al., 2015). To overcome the issue of slow training in BNNs (Graves, 2011), Monte Carlo dropout (MCDrop) approximates Bayesian inference by randomly dropping out weights during testing (Gal and Ghahramani, 2015). Despite the simplicity of Monte Carlo dropout, recent work using ensembles of NNs have demonstrated better calibrated results (Ovadia et al., 2019). In particular, deep ensembles (Lakshminarayanan et al., 2017), where individual networks are trained with different weight initialization, achieve state-of-the-art results by leveraging diverse models and their local optima (Fort et al., 2019). Hyper deep ensembles (Wenzel et al., 2020) and neural ensembles search (Zaidi et al., 2020) have further improved performance via using parameter search to build maximally diverse ensemble members, however, building and maintaining multiple models can be expensive.

Uncertainty in medical imaging. For quantifying uncertainty in medical imaging tasks, the most commonly used techniques are Monte Carlo dropout and deep ensembles (Rahman et al., 2021; Abdar et al., 2021). Few studies employ pure Bayesian methods such as BNNs on medical data (Zhao et al., 2018; Lotter et al., 2021) due to the former techniques being simpler to implement. To date, the majority of applications of uncertainty quantification in deep learning on medical data have focused on images such as cancerous skin lesions (Abdar et al., 2021), diabetic retinopathy (Leibig et al., 2017), and brain magnetic resonance imaging (Zhao et al., 2018). As of yet, there is no study of uncertainty quantification in deep learning that covers multiple medical imaging modalities (e.g. timeseries and images) across different architectures.

Early exit neural networks. Within the field of efficiency in deep learning, early exits are a class of conditional computation models that exit once a criterion (e.g., sufficient accuracy) is satisfied in order to save on computation (Wang et al., 2019; Li et al., 2019). Recent research leverages early exit predictions to dynamically change a neural network computation graph at test time (Nan and Saligrama, 2017; Montanari et al., 2019, 2020). The two most common use cases for early exits rely on either completing the full early exit inference before making a decision (Huang et al., 2017; Teerapittayanon et al., 2016) or applying a gating mechanism before the exit points in the backbone architecture (Bolukbasi et al., 2017). Under a different interpretation, the early exit paradigm can be used to mitigate the problem of model overthinking (Kaya et al., 2019) where the representations learnt in earlier layers can be more accurate than those learnt in the later layers which can sometimes not generalize well. As of yet, the connection between early exit model architectures and uncertainty remains unexplored.

[1] Code available at https://github.com/ajrcampbell/early-exit-ensembles

3. Methods

We introduce a novel framework, early exit ensembles, which gives early exit NNs a new probabilistic interpretation as an implicit ensemble. Herein, we present the methodology required to transform any feed-forward NN into an efficient ensemble of weight sharing sub-networks from which uncertainty can be quantified.

Problem formulation. Consider a classification problem where $\mathbf{x} \in \mathbb{R}^D$ denotes a D-dimensional input and $y \in \{1, \dots, C\}$ a corresponding discrete target taking one of C classes. We seek to learn a NN that can model the probabilistic predictive distribution $p_\theta(y|\mathbf{x})$ over the ground truth labels, where θ are model parameters. This predictive distribution is obtained via the softmax transform $\sigma(\cdot)$ of unnormalized log probabilities $f_\theta(\mathbf{x}) \in \mathbb{R}^C$.

Neural networks. By definition of a NN, $f_\theta(\cdot)$ consists of blocks of differential operations (e.g., convolution) (Goodfellow et al., 2016). We assume therefore that $f_\theta(\cdot)$ can be decomposed into a composition of B blocks such that

$$f_\theta(\mathbf{x}) = (f^{(B)} \circ f^{(B-1)} \circ \dots \circ f^{(1)})(\mathbf{x}) \tag{1}$$

where $(f^{(i)} \circ f^{(j)})(\mathbf{x}) = f_{\theta_i}(f_{\theta_j}(\mathbf{x}))$ denotes function composition for $i \neq j$ and $\theta = \cup_{i=1}^{B} \theta_i$ represents the union of each blocks parameters. Let $\mathbf{h}^{(i)} \in \mathbb{R}^{K_i \times D_i}$ denote the intermediary output of the i-th block having K_i features of dimension $D_i \leq D$ such that $\mathbf{h}^{(i)} = f_{\theta_i}(\mathbf{h}^{(i-1)})$ for $1 \leq i \leq B-1$, and $\mathbf{h}^{(0)} = \mathbf{x}$.

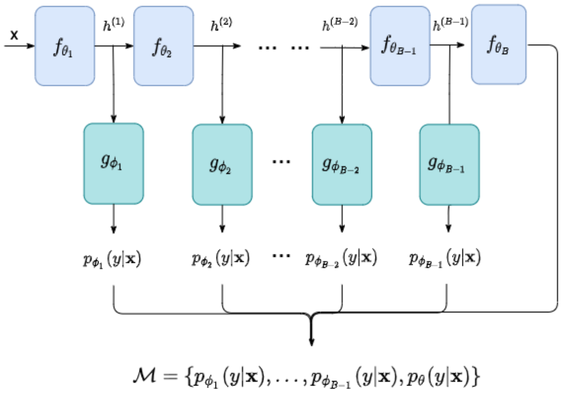
3.1. Early Exit Ensembles

With only minor architectural modifications, any multi-layered feed-forward NN can be converted into an implicit ensemble of networks by adding early exits. We define an early exit block as a NN $g_{\phi_i}(\cdot)$ with parameters ϕ_i which takes as input the intermediary output $\mathbf{h}^{(i)}$ from the i-th block of the backbone NN $f_\theta(\cdot)$. We let each exit block learn a probabilistic predictive distribution $p_{\phi_i}(y|\mathbf{x})$ such that

$$\begin{aligned} p_{\phi_i}(y|\mathbf{x}) &= \sigma(g_{\phi_i}(\mathbf{h}^{(i)})) \\ &= \sigma((g^{(i)} \circ f^{(i-1)} \circ \dots \circ f^{(1)})(\mathbf{x})) \end{aligned} \tag{2}$$

for $1 \leq i \leq B-1$. As such, any NN is able to output a set $\mathcal{M}$ containing up to $B-1$ probabilistic distributions from early exits blocks, in addition to the standard output from its final block

$$\mathcal{M} = \{p_{\phi_1}(y|\mathbf{x}), \dots, p_{\phi_{B-1}}(y|\mathbf{x}), p_\theta(y|\mathbf{x})\} \tag{3}$$



x
f_{θ_1}
$h^{(1)}$
f_{θ_2}
$h^{(2)}$
$h^{(B-2)}$
$f_{\theta_{B-1}}$
$h^{(B-1)}$
f_{θ_B}
g_{ϕ_1}
g_{ϕ_2}
$g_{\phi_{B-2}}$
$g_{\phi_{B-1}}$
$p_{\phi_1}(y|\mathbf{x})$
$p_{\phi_2}(y|\mathbf{x})$
$p_{\phi_{B-2}}(y|\mathbf{x})$
$p_{\phi_{B-1}}(y|\mathbf{x})$
$\mathcal{M} = \{p_{\phi_1}(y|\mathbf{x}), \dots, p_{\phi_{B-1}}(y|\mathbf{x}), p_\theta(y|\mathbf{x})\}$

Figure 1: Representation of an early exit ensemble. Each f_{θ_i} represents a block from the backbone NN. Exit block g_{ϕ_i} take as inputs intermediary output $h^{(i)}$ shared by the backbone (and previous exits) and learns a predictive distribution $p_{\phi_i}(y|\mathbf{x})$.

where $|\mathcal{M}| = B$. In practice, the number of exit blocks and therefore the ensemble size $|\mathcal{M}|$ is a hyperparameter bounded above by B and bounded below by a combination of factors such as computational cost (Kaya et al., 2019) and quality of the uncertainty estimates (Ovadia et al., 2019). Figure 1 provides a visual representation of an early exit ensemble.

Predictive uncertainty. During inference, a single forward pass of a NN with early exits produces a set of ensemble predictions $\mathcal{M}$. The overall prediction of the ensemble can be computed as the mean of a categorical distribution obtained from averaging the predictions from the individual exits

$$p_{\theta_\mathcal{M}}(y|\mathbf{x}) \approx \frac{1}{|\mathcal{M}|}\left(p_\theta(y|\mathbf{x}) + \sum_{i=1}^{B-1} p_{\phi_i}(y|\mathbf{x})\right) \tag{4}$$

where $\theta_\mathcal{M} = \theta \cup \phi$ and $\phi = \cup_{i=1}^{B-1} \phi_i$. Although other methods of aggregating the output of the ensemble exist (e.g. majority voting, geometric mean, or weighted average), previous works have shown that for uncertainty quantification the arithmetic mean works well in practice (Gal and Ghahramani, 2015; Lakshminarayanan et al., 2017).

3.2. Early Exit Blocks

The exit blocks are built to satisfy two important characteristics that have been shown to make a good ensemble: accuracy and diversity (Perrone and Cooper, 1992; Granitto et al., 2005). Since early exit blocks exit from different points in the backbone architecture, their predictions are naturally based off features learnt from structurally distinct architectures thereby promoting diversity. However, the exits from earlier blocks in the backbone inherit intermediary outputs with weak representational capacity which can negatively impact the overall accuracy.

Block architecture. To improve the accuracy of earlier exits, as well as further increase ensemble diversity, we propose inversely increasing the learning feature capacity of each block relative to its exit position in the backbone model. More formally, we introduce a learning capacity factor $\gamma \geq 0$ which increases the number of features K_i of the i-th intermediary output $\mathbf{h}^{(i)} \in \mathbb{R}^{K_i \times D_i}$ as follows:

$$K_i^\gamma = (\sqrt{1+\gamma})^{B-i} K_B, \quad 1 \leq i \leq B-1 \tag{5}$$

where K_B is the number of features in the last block defined by the backbone. The learning capacity factor changes the architecture of each exit block as:

$$g_{\phi_i}(\mathbf{h}^{(i)}) = \begin{cases} \mathbf{W}_2^{(i)} \rho(\mathbf{W}_1^{(i)} s(\mathbf{h}^{(i)})), & \gamma > 0 \\ \mathbf{W}_3^{(i)} s(\mathbf{h}^{(i)}), & \gamma = 0 \end{cases} \tag{6}$$

where $s(\cdot)$ denotes global average pooling used to reduce the spatial dimension D_i of each feature, $\rho(\cdot)$ is an activation function, and $\mathbf{W}_1^{(i)} \in \mathbb{R}^{K_i^\gamma \times K_i}$, $\mathbf{W}_2^{(i)} \in \mathbb{R}^{C \times K_i^\gamma}$ and $\mathbf{W}_3^{(i)} \in \mathbb{R}^{C \times K_i}$ are weights of linear layers (biases are omitted for clarity of notation). Clearly, when $\gamma > 0$ an exit block has the ability to learn more complex relations between features via the extra linear layer.

3.3. Training with Early Exists

In order to ensure early exit ensemble members are both accurate and diverse, we optimize a composite classification and diversity loss function:

$$\mathcal{L} = \mathcal{L}_C + \beta \mathcal{L}_D \tag{7}$$

where $\mathcal{L}_C$ is the classification loss, $\mathcal{L}_D$ is the diversity loss, and $\beta \geq 0$ represents the relative weight of the diversity loss. This procedure allows for an efficient training of the whole ensemble in one go.

Classification loss. The classification loss is the sum of individual losses of each exit in addition to the backbone. As such, each prediction propagates the error in relation to the ground truth label to the preceding exit blocks. More formally,

$$\mathcal{L}_C = L_{CE}(y, f_\theta(y|\mathbf{x})) + \sum_{i=1}^{B-1} \alpha_i L_{CE}(y, g_{\phi_i}(y|\mathbf{x})) \tag{8}$$

where $L_{CE}(\cdot,\cdot)$ is the cross-entropy loss function and $\alpha_i \in \{0,1\}$ is a weight parameter corresponding to the relative importance of each exit.

Diversity loss. In order to further increase ensemble diversity, correlation between exit block predictions must be minimal. To achieve this, we propose a diversity loss such as:

$$\mathcal{L}_D = \frac{1}{M} \sum_{i=1}^{B-1} \sum_{j \neq i} L_{CE}(g_{\phi_i}(y|\mathbf{x}), g_{\phi_j}(y|\mathbf{x})) \tag{9}$$

where $M = |\mathcal{M}|(|\mathcal{M}|-1)$. Since $\mathcal{L}_D$ minimizes the cross-entropy between all exit pairs it approximately maximizes pairwise mutual information (Boudiaf et al., 2020).

4. Experiments

We design experiments to verify the improvement of early exit ensembles in providing well-calibrated uncertainty quantification for in-distribution data compared to state-of-the-art deep learning ensemble baselines (Section 4.1). We then further verify the ability of early exit ensembles to detect out-of-distribution samples (ODD) (Section 4.2).

4.1. Uncertainty Estimation

Datasets. We evaluate our proposed early exit ensembles on three medical imaging classification datasets: (1) ECG heart attack (ECG5000) (Dau et al., 2019), (2) EEG epileptic seizure (EEG) (Dua and Graff, 2017), and (3) skin melanoma (ISIC2018) (Codella et al., 2019). Each dataset is paired with a different architecture (see Appendix B): FCNet for ECG5000, ResNet18 for EEG, and MobileNetV2 for ISIC2018. Table 1 contains a descriptive summary of each dataset. All datasets are split into 80% training, 10% validation and 10% testing maintaining the original class proportions. Further details are included in Appendix A.

Placement of exits. Following findings from recent works on the optimal number of ensemble members to consider for well-calibrated uncertainty (Ovadia et al., 2019; Qendro et al., 2021), we set the ensemble size $|\mathcal{M}| = 5$ for all models. Since for FCNet

Model			Dataset						
Name	**B**	θ	**Name**	**Type**	**Train**	**Valid**	**Test**	**C**	**D**
FCNet	5	0.2M	ECG5000	Time series	4050	450	500	2	1 × 140
ResNet18	9	3.8M	EEG	Time series	9315	1035	1150	5	1 × 178
MobileNetV2	18	3.1M	ISIC2018	Image	8268	919	1021	7	3 × 224 × 224

Table 1: Summary of models and datasets used for experiments. All datasets are split into 80% training, 10% validation and 10% testing maintaining the original class proportions. B is the number of exit blocks, θ is number of model parameters (millions), C is number of classes, and D is input data dimension.

this results in no choice of exit placement, we exit after all blocks. On the other hand, for ResNet18 we semantically group residual blocks based on number of hidden features (based on findings from different exit strategies detailed in Section 5.3). For MobileNetV2, we follow a similar strategy to Kaya et al. (2019) by placing exit points based on the overall number of floating-point operations (FLOPs) in the model. As such, we place exits points closest to 45%, 60%, 75% and 90% of the overall FLOPs for MobileNetV2.

Baselines. We compare early exit ensembles against its backbone without exists (Backbone), Monte Carlo dropout (MCDrop) (Gal and Ghahramani, 2015), deep ensembles (Deep) (Lakshminarayanan et al., 2017) and depth ensembles (Depth):

- **Backbone** - the unchanged implementations of FCNet, ResNet18, and MobileNetV2 representing a single model which outputs softmax probabilities over classes.

- **MCDrop** - an approximate Bayesian probabilistic approach implemented by adding dropout layers to each of the backbone architectures during training and activating them during inference. For fairness of comparison, dropout layers are added to the same places as the exit points. A total of 5 Monte Carlo samples are taken with a dropout rate of $p_{\mathrm{MC}} = 0.2$.

- **Deep** - a non-Bayesian probabilistic approach based on training an explicit ensemble of models with the same backbone architecture but trained with a different random weight initialization. A total of 5 models were used for the ensemble.

- **Depth** - our own baseline composed of an ensemble of the same backbone where each member has a different depth determined by the exit points. Similar to Deep, all ensemble members are trained separately. In contrast, each of the 5 models ranges from shallow to deep and is trained with the same weight initialization.

We omit the recently proposed batch ensembles (Wen et al., 2020) from our experiments since they do not improve upon deep ensembles in terms of uncertainty quantification.

Implementation. We use the data preprocessing and hyperparameters described in the original implementation of each model-dataset combination as the initial starting point for hyperparameter tuning. All models are trained with the Adam (Kingma and Ba, 2014) optimizer using default parameters, except for the learning rate which is tuned over $\{1e^{-2}, 1e^{-3}, 1e^{-4}, 1e^{-5}\}$. For MCDrop, the dropout rate is tuned over $\{0.2, 0.3, 0.5\}$. We empirically choose batch sizes and epochs from 50, 100, 200 and 100, 200, 250 respectively based on stability of training. To prevent overfitting, early-stopping is implemented with a patience of 5 based on the best validation accuracy. We train early exit ensembles with the best performing hyperparameters across all backbone models $\alpha_i = 1$ and $\beta = 0$ (see Section 5.5). Finally, all models were implemented using PyTorch (Paszke et al., 2019) and trained on a 4-GPU Linux server with 64GB memory.

Evaluation metrics. The classification performance of all models is evaluated based on class weighted F1, precision, and recall. For uncertainty quantification, we use negative log-likelihood (NLL), Brier score (BS) (Brier et al., 1950), and expected calibration error (ECE) (Naeini et al., 2015). NLL measures how likely it is to observe the test data given each trained model, BS measures the accuracy of predicted probabilities, and ECE measures model calibration in terms of the expected difference between accuracy and predicted confidence. A description of how these metrics are computed can be found in Appendix C.

Model	**F1** (↑)	**Precision** (↑)	**Recall** (↑)	**NLL** (↓)	**ECE** (↓)	**BS** (↓)
FCNet						
- Backbone	0.983 ± 0.010	0.983 ± 0.010	0.983 ± 0.010	0.059 ± 0.031	0.009 ± 0.005	0.026 ± 0.015
- MCDrop	0.987 ± 0.002	0.987 ± 0.002	0.987 ± 0.002	0.043 ± 0.019	0.011 ± 0.004	0.019 ± 0.004
- Depth	0.989 ± 0.007	0.989 ± 0.007	0.989 ± 0.007	0.036 ± 0.008	0.017 ± 0.007	0.018 ± 0.006
- Deep	0.989 ± 0.003	0.989 ± 0.003	0.989 ± 0.003	0.045 ± 0.020	0.014 ± 0.007	0.018 ± 0.005
- Early exit	**0.992 ± 0.005**	**0.992 ± 0.005**	**0.992 ± 0.005**	**0.024 ± 0.008**	**0.007 ± 0.001**	**0.009 ± 0.003**
ResNet18						
- Backbone	0.847 ± 0.012	0.848 ± 0.012	0.847 ± 0.012	0.432 ± 0.022	0.081 ± 0.007	0.233 ± 0.018
- MCDrop	0.844 ± 0.007	0.849 ± 0.004	0.844 ± 0.007	0.362 ± 0.012	0.045 ± 0.005	0.216 ± 0.011
- Depth	0.861 ± 0.011	0.862 ± 0.011	0.861 ± 0.011	0.318 ± 0.028	0.028 ± 0.003	0.194 ± 0.012
- Deep	**0.866 ± 0.009**	**0.866 ± 0.008**	0.866 ± 0.009	0.316 ± 0.024	0.028 ± 0.004	0.189 ± 0.011
- Early exit	0.865 ± 0.002	0.865 ± 0.002	**0.866 ± 0.002**	**0.306 ± 0.013**	**0.027 ± 0.006**	**0.189 ± 0.005**
MobileNetV2						
- Backbone	0.868 ± 0.005	0.869 ± 0.006	0.870 ± 0.003	0.512 ± 0.022	0.080 ± 0.008	0.209 ± 0.010
- MCDrop	0.865 ± 0.002	0.872 ± 0.001	0.861 ± 0.002	0.375 ± 0.016	0.041 ± 0.008	0.197 ± 0.008
- Depth	0.865 ± 0.011	0.873 ± 0.010	0.861 ± 0.011	0.495 ± 0.005	0.133 ± 0.017	0.237 ± 0.004
- Deep	**0.887 ± 0.011**	**0.888 ± 0.011**	**0.888 ± 0.009**	0.359 ± 0.019	0.037 ± 0.008	0.171 ± 0.007
- Early exit	0.885 ± 0.005	0.885 ± 0.004	0.886 ± 0.005	**0.357 ± 0.020**	**0.033 ± 0.010**	**0.170 ± 0.006**

Table 2: Results for classification and uncertainty metrics. All results are for the optimally tuned learning rate, batch size, and number of epochs for each model: FCNet ($1e^{-2}$, 200, 250), ResNet18 ($1e^{-3}$, 200, 200), MobileNetV2 ($1e^{-5}$, 100, 200). Results for MCDrop are for the optimal dropout rate $p_{\mathrm{MC}} = 0.2$. Results for Early Exit are for the weighted classification loss $\alpha_i = 1$, diversity loss $\beta = 0$ and learning capacity factor: FCNet (0.0), ResNet18 (0.2), MobileNetV2 (0.7). We mark in bold the best results. All results are the mean plus/minus standard deviation across 3 independent splits of the test dataset.

4.2. Out-of-Distribution Detection

Datasets. For OOD analysis, we use three datasets containing the same type of signal but from completely different distributions: ECG heart attack (ECG200) (Dau et al., 2019) is applied to FCNet, EEG Steady-State Visual Evoked Potential Signals (SSEVP) (Dua and Graff, 2017) is applied to ResNet18, and CIFAR-10 (Krizhevsky and Hinton, 2010) is applied to MobileNetV2.

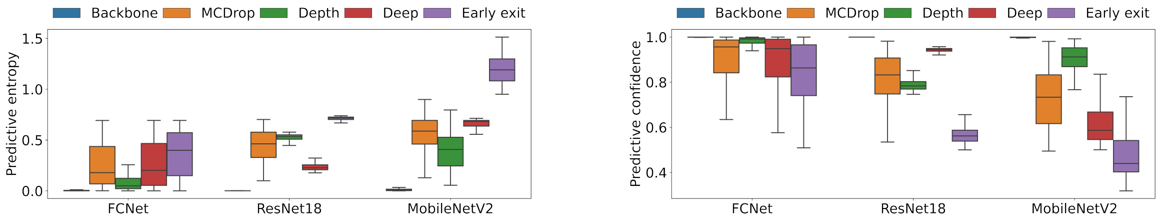
Evaluation metrics. OOD behavior is evaluated using predictive confidence (PC) and predictive entropy (PE) (see Appendix C). PC measures the probability of the top class prediction whilst PE measure the amount of information contained in the predictive distribution.

5. Results

We present the results of early exit ensembles on in-distribution (Section 5.1) and out-of-distribution (Section 5.2) experiments and compare them to the aforementioned baselines. Additionally, we analyze the effect of the number and position of exit blocks (Section 5.3), the learning capacity factor (Section 5.4), and diversity regularization as well as exit loss weighting (Section 5.5).

5.1. Uncertainty Estimation

Table 2 shows classification and uncertainty results on in-distribution test datasets. Our results show that the addition of early exits, and their joint training, improves accuracy compared to the backbone model across all datasets by an average of 1.5% as measured by F1 score. These findings are in line with existing work on early exit architectures (Teerapittayanon et al., 2016). Furthermore, Early Exit, Deep, Depth and MCDrop all display significantly better F1 score, precision and recall compared to Backbone across each of the three datasets. We attribute these results to a reduction in variance caused by averaging a set of NNs which individually have a high variance and low bias (Perrone and Cooper, 1992). For ResNet18 and MobileNetV2, Deep is on average 0.2% better than Early Exit across classification metrics. However, this marginal sacrifice in



(a) Predictive entropy (PE)

(b) Predictive confidence (PC)

Figure 2: Results for out-of-distribution sample detection

accuracy is negligible compared to the improvement gain in uncertainty metrics. For instance, Early Exit improves model ECE by an estimated 4% and 11% for ResNet18 and MobileNetV2, respectively.

Early Exit clearly outperforms MCDrop both in terms of classification and uncertainty metrics. In particular, BS is improved by an estimated 53%, 13%, and 14% for FCNet, ResNet18 and MobileNetV2, respectively. Such improvement translates into the predicted probability distributions over the classes more accurately reflecting the ground truth distribution in the data. Moreover, the generalizability of early exit ensembles is demonstrated by improvement in uncertainty quantification for both deep (ResNet18 and MobileNetV2) and shallow architectures (FCNet). In particular, for FCNet uncertainty metrics are roughly 2× less compared to Deep and MCDrop.

Finally, depth ensembles perform well across all metrics compared to MCDrop for both FCNet and ResNet18. In particular, Depth increase F1 score by an estimated 2% and improves NLL by 14% compared to MCDrop for ResNet18. These results demonstrate that when forming an ensemble, diversity in terms of different NN architecture depths, improves model calibration. Interestingly, Depth does not outperform Early Exit in either classification or uncertainty metrics. We attribute this difference to Early Exit having the benefit of exit blocks that are joint trained with a backbone architecture. As a result, Early Exit not only benefits from shared features from the backbone that are informative for the task but is also able to diversify these features at each each exit block allowing it to make accurate but independent predictions.

5.2. Out-of-Distribution Detection

Figure 2 shows PC and PE on out-of-distribution datasets. A well calibrated model should show low PC and high PE on out-of-distribution data. Our results clearly show across all datasets that Backbone is overly confident in making wrong predictions as reflected by an average PC of 0.99 and an PE entropy of 0.00. This demonstrates the risk of deploying deep learning models which are unable to provide proper uncertainty estimates potentially resulting in overconfident wrong diagnoses. Our proposed early exit ensemble framework provides the highest PE and lowest PC for out-of-distribution samples. In particular, Early Exit has an estimated 38%, 77% and 34% higher entropy for FCNet, MobileNetV2 and ResNet18 compared to their respective best performing baselines.

In contrast to previous works, Deep does not always out outperform MCDrop. For FCNet, Deep is comparable to MCDrop in terms of both average confidence and entropy. For ResNet18, MCDrop has a significantly lower PC (0.81 vs. 0.94) and PE (0.44 vs. 0.24). Depth ensembles do not show good calibration on OOD data having the worse performance for FCNet and MobileNet on both PC (0.97 and 0.90, respectively) and PE (0.10 and 0.40, respectively). While they showed acceptable uncertainty metrics in in-distribution, they cannot be trusted in OOD scenarios.

5.3. Effect of Number of Exits

The number of early exits, and therefore ensemble size $|\mathcal{M}|$, is bounded above by the number of blocks B in each backbone model architecture (see Table A). As such, each exit block is indexed $i \in \{1, \ldots, B-1\}$ which gives a combinatorial choice $\binom{B}{|\mathcal{M}|}$ of exist points and therefore ensemble arrangements. We investigate the effect of the number and placement of exits on predictive uncertainty using the ResNet18 backbone with $\gamma = 0.2$. Since for ResNet18, $B = 9$ and $|\mathcal{M}| = 5$ this yields 126 possible combinations of

Strategy	$\lvert\mathcal{M}\rvert$	**F1** (↑)	**ECE** (↓)	**BS** (↓)	**Exit index,** i	**Param. increase**
Pareto	$2+1$	0.853 ± 0.01	0.049 ± 0.01	0.201 ± 0.01	$\{4, 7\}$	9.8%
Last-k	$4+1$	0.849 ± 0.00	0.063 ± 0.01	0.219 ± 0.01	$\{5\ ,6, 7, 8\}$	23.3%
Semantic	$4+1$	$\mathbf{0.865 \pm 0.00}$	$\mathbf{0.027 \pm 0.01}$	$\mathbf{0.189 \pm 0.01}$	$\{2, 4, 6, 8\}$	15.2%
Computation	$6+1$	0.861 ± 0.01	0.057 ± 0.02	0.200 ± 0.01	$\{3, 5, 6, 7, 8\}$	25.6%
Residual	$8+1$	0.856 ± 0.01	0.032 ± 0.02	0.193 ± 0.02	$\{1, 2, 3, 4, 5, 6, 7, 8\}$	30.4%

Table 3: Effect of the number of exit blocks and exit strategy on accuracy and uncertainty for ResNet18 with $\gamma = 0.2$. Reported results are the mean plus/minus standard deviation across 3 independent splits of the test dataset. Ensemble size $\lvert\mathcal{M}\rvert$ includes the backbone in addition to each exit block (represented as $+1$). For *Last-k* we set $k = 4$. Parameter increase is measured as a percentage increase over the number of parameters from the backbone. Exit index match the indexes of the exit blocks in Figure 5.

ensembles. To limit this search space, we introduce and test the following exit strategies:

- *Pareto* (Pareto, 1964): Exit at blocks closest to 20% and 80% of the overall number of FLOPs.
- *Computation*: Exit at blocks closest to 15%, 30%, 45%, 60%, 75% and 90% of the overall number of FLOPs (Kaya et al., 2019).
- *Residual*: Exit at residual blocks.
- *Last-k*: Exit at the last k blocks.
- *Semantic*: Exit at blocks semantically grouped by their number of features.

Table 3 summarizes the results for the listed exit strategies. A small ensemble of 3 members in the case of *Pareto* does not perform well in terms of accuracy but still shows well-calibrated uncertainty comparable to *Computation* with 5 members (ECE 0.049 vs. 0.057). Interestingly, a higher number of exits does not guarantee better uncertainty quantification (*Residual* BS 0.193 vs. *Semantic* BS 0.189), nor does choosing to exit deeper in the case of *Last-k* (BS 0.219). The strategy *Semantic* clearly gives the best results in terms of accuracy and uncertainty metrics reinforcing our findings in Section 5.1 which employs the same strategy. We attribute this result to the fact that when exiting at semantically group blocks, low-level and high-level features are integrated by summation at residual connections therefore representing the points of maximum diversity in the backbone.

Figure 5 in Appendix D provides a detailed illustration of the ResNet18 backbone with exit blocks. Exit indexes in Table 3 match the index of the exit blocks in Figure 5.

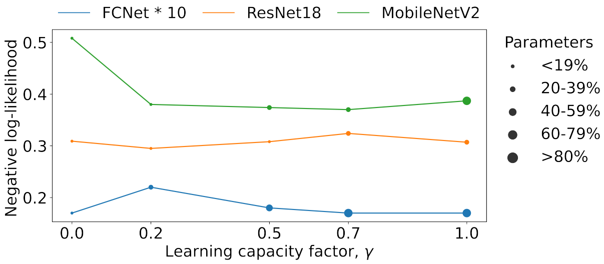


Figure 3: Effect of learning capacity factor γ on NLL as well the percentage increase in parameters over the backbone.

5.4. Effect of Learning Capacity Factor

In Figure 3, we show the effect of tuning the learning capacity factor γ for Early Exit in terms of NLL on each of the validation datasets. For FCNet, $\gamma = 0.0$ yields the best results for Early Exit in terms of lowest NLL and smallest percentage increase in parameters. On the other hand, for the deeper backbone architectures ResNet18 and MobileNetV2 on the larger datasets, higher learning capacity factors ($\gamma = 0.2$, $\gamma = 0.7$, respectively) yield the best results in terms of lowest NLL. Intuitively, for these harder classification problems, the earlier exit blocks require more features in order to make accurate predictions that improve the overall ensemble performance. For ResNet18 and MobileNetV2 this translates to a small increase of 1.15× and 1.38× respectively in the number of overall parameters. In contrast, Deep with $\lvert\mathcal{M}\rvert = 5$ always increases parameters by 5× regardless of the backbone and dataset. Additionally, unlike MCDrop, Early Exit can provide uncertainty quantification in one single forward pass.

5.5. Further analysis

Effect of weighting exits. We experiment with running early exit ensembles using a ResNet18 backbone setting $\gamma = 0$ and tuning only the importance weights from the classification loss (Equation 8). Without the extra learning capacity, the early exits with $\alpha_i = 1$ tend to have a high misclassification rate. However, penalizing the earlier exits, by setting $\alpha_i = 1/(B-1-i)$ for example, does not improve classification performance and worsens uncertainty metrics overall. We conclude the performance of early exit ensembles requires each individual member to be accurate in line with the findings of Perrone and Cooper (1992).

Effect of diversity loss. We further experiment with running all early exit models setting $\gamma = 0$ and tuning only the weight β of the diversity loss (Equation 7) over $\{0.3, 0.5, 1.0, 2.0\}$ using ResNet18. Intuitively, a higher value of β should allocate more importance to learning a diverse early exit ensemble. Results show that as β increases from 0.3 to 2.0, F1 score increases by 23.1% (from 0.85 to 0.86) and ECE decreases by 14.6% (from 0.047 to 0.041). Although performance improves, accuracy and calibration are still worse than the best result for early exit ensembles in Table 2.

Using uncertainty in patient-level decision making. During inference each ensemble technique outputs $|\mathcal{M}| = 5$ softmax predictions over classes. The extent of agreement/disagreement between these predictions can be used to quantify uncertainty. Figure 4 visualizes two outputs for Early Exit using MobileNetV2 on the ISIC2018 test dataset. It is clear from (a) that all exits are individually confident and correctly predict the same class $y = 0$ (melonocytic nevi). On the other hand, in (b) all the exits highly disagree predicting different classes to the true class $y = 1$ (melanoma) suggesting further investigation from an expert clinician.

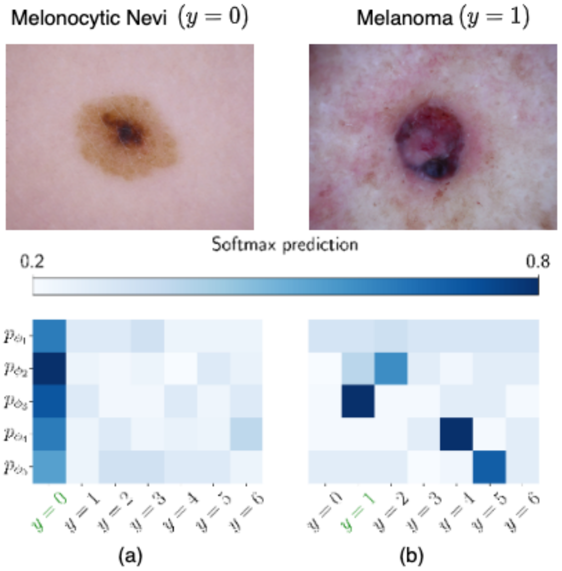


Figure 4: Early exit ensemble test predictions. In (a) all exits agree on the prediction. In (b) all exits highly disagree predicting different classes indicating the sample should sent to a clinician for further checks. Correct class in green.

6. Conclusion

We introduce, early exit ensembles, a novel framework for enabling uncertainty quantification in any feed-forward NN. At the core of our methodology is a new interpretation of early exit NNs as an implicit ensemble of weight sharing sub-networks from which predictive uncertainty can be estimated. We evaluate early exit ensembles using three state-of-the-art deep learning architectures applied to different medical imaging datasets. Our results show that, compared to competitive deep learning ensemble baselines, early exit ensembles can provide better calibrated uncertainty for both in-and out-of-distribution data. Moreover, our approach enables training all ensemble members jointly in a single model, as well as providing uncertainty estimations from a single forward-pass of input data. Both the ease of implementation and the computational efficiency of training and inference means that early exit can be applied to a wide range of real-world applications.

Future work. A limitation of early exit ensembles is the ensemble size being tied to the underlying backbone architecture. To overcome this issue, future work could add dropout and/or add multiple heads to the exit blocks to further increase ensemble size. Furthermore, new methodologies for enforcing diversity in order to boost ensemble performance is an interesting research direction. Finally, since early exit ensembles are efficient-by-design, future work could include an in-depth analysis of efficiency (e.g. FLOPs, runtime latency, and energy consumption) on resource-constrained devices specific to healthcare.

Acknowledgments

This work is supported by (1) Nokia Bell Labs through their donation for the Centre of Mobile, Wearable Systems and Augmented Intelligence, ERC Project 833296 (EAR), and (2) The Alan Turing Institute under the EPSRC grant EP/N510129/1.

References

Moloud Abdar, Maryam Samami, Sajjad Dehghani Mahmoodabad, Thang Doan, Bogdan Mazoure, Reza Hashemifesharaki, Li Liu, Abbas Khosravi, U Rajendra Acharya, Vladimir Makarenkov, et al. Uncertainty quantification in skin cancer classification using three-way decision-based bayesian deep learning. *Computers in biology and medicine*, page 104418, 2021.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.

Paschalis Bizopoulos, George I Lambrou, and Dimitrios Koutsouris. Signal2image modules in deep neural networks for eeg classification. In *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 702–705. IEEE, 2019.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural networks. *arXiv preprint arXiv:1505.05424*, 2015.

Tolga Bolukbasi, Joseph Wang, Ofer Dekel, and Venkatesh Saligrama. Adaptive neural networks for fast test-time prediction. *arXiv preprint arXiv:1702.07811*, 2017.

Malik Boudiaf, Jérôme Rony, Imtiaz Masud Ziko, Eric Granger, Marco Pedersoli, Pablo Piantanida, and Ismail Ben Ayed. A unifying mutual information view of metric learning: cross-entropy vs. pairwise losses. In *European Conference on Computer Vision*, pages 548–564. Springer, 2020.

Glenn W Brier et al. Verification of forecasts expressed in terms of probability. *Monthly weather review*, 78(1):1–3, 1950.

Justin Chan, Thomas Rea, Shyamnath Gollakota, and Jacob E Sunshine. Contactless cardiac arrest detection using smart devices. *arXiv preprint arXiv:1902.00062*, 2019.

Saket S Chaturvedi, Kajol Gupta, and Prakash S Prasad. Skin lesion analyser: An efficient seven-way multi-class skin cancer classification using mobilenet. In *International Conference on Advanced Machine Learning Technologies and Applications*, pages 165–176. Springer, 2020.

Noel Codella, Veronica Rotemberg, Philipp Tschandl, M Emre Celebi, Stephen Dusza, David Gutman, Brian Helba, Aadi Kalloo, Konstantinos Liopyris, Michael Marchetti, et al. Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic). *arXiv preprint arXiv:1902.03368*, 2019.

Hoang Anh Dau, Anthony Bagnall, Kaveh Kamgar, Chin-Chia Michael Yeh, Yan Zhu, Shaghayegh Gharghabi, Chotirat Ann Ratanamahatana, and Eamonn Keogh. The ucr time series archive. *IEEE/CAA Journal of Automatica Sinica*, 6(6): 1293–1305, 2019.

Dheeru Dua and Casey Graff. Uci machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Andre Esteva, Brett Kuprel, Roberto A Novoa, Justin Ko, Susan M Swetter, Helen M Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *nature*, 542(7639):115–118, 2017.

Stanislav Fort, Huiyi Hu, and Balaji Lakshminarayanan. Deep ensembles: A loss landscape perspective. *arXiv preprint arXiv:1912.02757*, 2019.

Yarin Gal and Zoubin Ghahramani. Bayesian convolutional neural networks with bernoulli approximate variational inference. *arXiv preprint arXiv:1506.02158*, 2015.

Carlos García Rodríguez, Jordi Vitrià, and Oscar Mora. Uncertainty-based human-in-the-loop deep learning for land cover segmentation. *Remote Sensing*, 12(22):3836, 2020.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

Pablo M Granitto, Pablo F Verdes, and H Alejandro Ceccatto. Neural network ensembles: evaluation of aggregation algorithms. *Artificial Intelligence*, 163 (2):139–162, 2005.

Alex Graves. Practical variational inference for neural networks. In *Advances in neural information processing systems*, pages 2348–2356, 2011.

Lars Kai Hansen and Peter Salamon. Neural network ensembles. *IEEE transactions on pattern analysis and machine intelligence*, 12(10):993–1001, 1990.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Gao Huang, Danlu Chen, Tianhong Li, Felix Wu, Laurens van der Maaten, and Kilian Q Weinberger. Multi-scale dense networks for resource efficient image classification. *arXiv preprint arXiv:1703.09844*, 2017.

Yigitcan Kaya, Sanghyun Hong, and Tudor Dumitras. Shallow-deep networks: Understanding and mitigating network overthinking. In *International Conference on Machine Learning*, pages 3301–3310. PMLR, 2019.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Alex Krizhevsky and Geoff Hinton. Convolutional deep belief networks on cifar-10. *Unpublished manuscript*, 40(7):1–9, 2010.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, pages 6402–6413, 2017.

Christian Leibig, Vaneeda Allken, Murat Seçkin Ayhan, Philipp Berens, and Siegfried Wahl. Leveraging uncertainty information from deep neural networks for disease detection. *Scientific reports*, 7(1): 1–14, 2017.

En Li, Liekang Zeng, Zhi Zhou, and Xu Chen. Edge ai: On-demand accelerating deep neural network inference via edge computing. *IEEE Transactions on Wireless Communications*, 19(1):447–457, 2019.

William Lotter, Abdul Rahman Diab, Bryan Haslam, Jiye G Kim, Giorgia Grisot, Eric Wu, Kevin Wu, Jorge Onieva Onieva, Yun Boyer, Jerrold L Boxerman, et al. Robust breast cancer detection in mammography and digital breast tomosynthesis using an annotation-efficient deep learning approach. *Nature Medicine*, 27(2):244–249, 2021.

David JC MacKay. *Bayesian methods for adaptive models*. PhD thesis, California Institute of Technology, 1992.

Gustav Mårtensson, Daniel Ferreira, Tobias Granberg, Lena Cavallin, Ketil Oppedal, Alessandro Padovani, Irena Rektorova, Laura Bonanni, Matteo Pardini, Milica G Kramberger, et al. The reliability of a deep learning model in clinical out-of-distribution mri data: a multicohort study. *Medical Image Analysis*, 66:101714, 2020.

Alessandro Montanari, Mohammed Alloulah, and Fahim Kawsar. Degradable inference for energy autonomous vision applications. In *Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp/ISWC '19, 2019. ISBN 978-1-4503-6869-8. doi: 10.1145/3341162.3349337. URL http://doi.acm.org/10.1145/3341162.3349337.

Alessandro Montanari, Manuja Sharma, Dainius Jenkus, Mohammed Alloulah, Lorena Qendro, and Fahim Kawsar. eperceptive: energy reactive embedded intelligence for batteryless sensors. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*, pages 382–394, 2020.

Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. Obtaining well calibrated probabilities using bayesian binning. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

Feng Nan and Venkatesh Saligrama. Adaptive classification for prediction under a budget. In *Advances in neural information processing systems*, pages 4727–4737, 2017.

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, David Sculley, Sebastian Nowozin, Joshua V Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. *arXiv preprint arXiv:1906.02530*, 2019.

Vilfredo Pareto. *Cours d'économie politique*, volume 1. Librairie Droz, 1964.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703*, 2019.

Michael P Perrone and Leon N Cooper. When networks disagree: Ensemble methods for hybrid neural networks. Technical report, BROWN UNIV PROVIDENCE RI INST FOR BRAIN AND NEURAL SYSTEMS, 1992.

Mathias Perslev, Michael Hejselbak Jensen, Sune Darkner, Poul Jørgen Jennum, and Christian Igel. U-time: A fully convolutional network for time series segmentation applied to sleep staging. *arXiv preprint arXiv:1910.11162*, 2019.

Lorena Qendro, Sangwon Ha, René de Jong, and Partha Maji. Stochastic-shield: A probabilistic approach towards training-free adversarial defense in quantized cnns. *arXiv preprint arXiv:2105.06512*, 2021.

Zillur Rahman, Md Sabir Hossain, Md Rabiul Islam, Md Mynul Hasan, and Rubaiyat Alim Hridhee. An approach for multiclass skin lesion classification based on ensemble learning. *Informatics in Medicine Unlocked*, 25:100659, 2021.

Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

Surat Teerapittayanon, Bradley McDanel, and Hsiang-Tsung Kung. Branchynet: Fast inference via early exiting from deep neural networks. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 2464–2469. IEEE, 2016.

Zhiguang Wang, Weizhong Yan, and Tim Oates. Time series classification from scratch with deep neural networks: A strong baseline. In *2017 International joint conference on neural networks (IJCNN)*, pages 1578–1585. IEEE, 2017.

Zizhao Wang, Wei Bao, Dong Yuan, Liming Ge, Nguyen H Tran, and Albert Y Zomaya. See: Scheduling early exit for mobile dnn inference during service outage. In *Proceedings of the 22nd International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pages 279–288, 2019.

Yeming Wen, Dustin Tran, and Jimmy Ba. Batchensemble: an alternative approach to efficient ensemble and lifelong learning. *arXiv preprint arXiv:2002.06715*, 2020.

Florian Wenzel, Jasper Snoek, Dustin Tran, and Rodolphe Jenatton. Hyperparameter ensembles for robustness and uncertainty quantification. *arXiv preprint arXiv:2006.13570*, 2020.

Tong Xia, Jing Han, Lorena Qendro, Ting Dang, and Cecilia Mascolo. Uncertainty-aware covid-19 detection from imbalanced sound data. *arXiv preprint arXiv:2104.02005*, 2021.

Sheheryar Zaidi, Arber Zela, Thomas Elsken, Chris Holmes, Frank Hutter, and Yee Whye Teh. Neural ensemble search for performant and calibrated predictions. *arXiv preprint arXiv:2006.08573*, 2020.

Gengyan Zhao, Fang Liu, Jonathan A Oler, Mary E Meyerand, Ned H Kalin, and Rasmus M Birn. Bayesian convolutional neural network based mri brain extraction on nonhuman primates. *Neuroimage*, 175:32–44, 2018.

Appendix A. Datasets

ECG heart attack (ECG5000) (Dau et al., 2019). A dataset of univariate timeseries of ECG signals of length 140 extracted from a single patient. Each signal falls into one of 5 classes which are combined to make two labels: Normal (N) and Abnormal (R-on-T, PVC, SP, UB). The original train and test datasets are combined creating a dataset of size 5000 which is re-split maintaining class proportions.

EEG epileptic seizure (EEG) (Dua and Graff, 2017). A dataset of univariate timeseries of single-channel EEG signal of length 178 extracted from 500 patients. Each signal falls into one of 5 classes: normal patient eyes open, normal patient eyes closed, tumor patient healthy area, tumor patient tumor area, epileptic patient seizure. The original train and test datasets are combined creating a dataset of size 11500 which is re-split maintaining patient and class proportions. During training, a combination of Gaussian noise, signal shift, and polarity inversion is applied with probability 0.5.

Skin melanoma (ISIC2018) (Codella et al., 2019) A dataset of multi-source dermatoscopic images of common pigmented skin lesions. Each image falls into one of 7 classes: melanocytic nevi, melanoma, benign keratosis-like lesions, basal cell carcinoma, actinic keratoses, vascular lesions, and dermatofibroma. The original train and validation datasets from Task 3 are combined creating a dataset of size 10208 which is re-split maintaining class proportions. All images are resized to 224 × 224 and then normalized using the training dataset mean and standard deviation. During training, a combination of Gaussian noise, horizontal and vertical flips, and jitter is applied to each image with probability 0.5.

Appendix B. Backbone architectures

FCNet (Wang et al., 2017) A fully convolutional network consisting of 4 blocks each containing a 1D convolution, batch normalization (BN), and Rectified Linear Unit (ReLU) activation. The output of the fourth block is averaged over the time dimension using global average pooling (GAP) and fed to a 1D convolutional layer with filter length 1. Convolutional layers all have 128 filters of length 8, 5, 5, and 3 all with a stride of 1 and zero padding to preserve the length of each time series input.

ResNet18 (He et al., 2016; Bizopoulos et al., 2019) A residual convolutional network composed of 8 blocks containing a 1D convolution, BN, and ReLU activation repeated twice. Convolutions in consecutive pairs of blocks have filters 64, 128, 256, 512 with stride 1, 2, 2, 2 all with length 3. The output of these blocks are fed to a GAP and a fully connected (FC) layer. Identity residual connections exist between consecutive blocks with the same number of filters. Downsample residual connections exist between blocks with different filter numbers.

MobileNetV2 (Sandler et al., 2018; Chaturvedi et al., 2020) A convolution neural network mobile architecture composed of 17 inverted residual blocks with bottleneck layers. Each inverted residual block contains a 2D convolution of size 1x1, a 2D depthwise convolution of size 3x3, a Rectified Linear Unit 6 (ReLU6), and finally a 2D convolution of size 1x1 followed by ReLU6.

Appendix C. Uncertainty metrics

C.1. Negative log-likelihood

Negative log-likelihood (NLL) measures how likely it is to observe the data under model. NLL is defined

$$\begin{aligned}\mathrm{NLL} = -\sum_{c\in\{1,\ldots,C\}} & \mathbb{1}(y=c)\log p_\theta(y=c|\mathbf{x}) \\ & +(1-\mathbb{1}(y=c))\log(1-p(y=c|\mathbf{x}))\end{aligned} \tag{10}$$

where $\mathbb{1}(\cdot)$ is the indicator function.

C.2. Brier score

Brier score (BS) measures the accuracy of predicted probabilities. BS is defined

$$\mathrm{BS} = \sum_{c\in\{1,\ldots,C\}} (p_\theta(y=c|\mathbf{x}) - \mathbb{1}(y=c))^2. \tag{11}$$

C.3. Predictive confidence

Predictive confidence (PC) is the probability of the top class prediction. PC is defined:

$$\mathrm{PC} = \max_{c\in\{1,\ldots,C\}} p_\theta(y=c|\mathbf{x}) \tag{12}$$

C.4. Predictive entropy

Predictive entropy (PE) measures the average amount of information in the predicted distribution.

PE is defined:

$$\text{PE} = -\sum_{c\in\{1,\ldots,C\}} p_\theta(y=c|\mathbf{x})\log p_\theta(y=c|\mathbf{x}) \quad (13)$$

C.5. Expected calibration error

Expected calibration error (ECE) measures the expected difference (in absolute value) between accuracies and the predicted confidences on samples belonging to different confidence intervals. ECE is defined:

$$\text{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{n}|\text{acc}(B_m) - \text{conf}(B_m)| \quad (14)$$

where accuracy and confidence for bin B_m are

$$\text{acc}(B_m) = \frac{1}{|B_m|}\sum_{n\in B_m} \mathbb{1}(\hat{y}_n = y_n)$$
$$\text{conf}(B_m) = \frac{1}{|B_m|}\sum_{n\in B_m} \text{PC}_n$$

such that $\hat{y}_n = \arg\max_{c\in\{1,\ldots,C\}} p_\theta(y_n = c|\mathbf{x}_n)$ is the predicted class, M is the number of bins of size $1/M$, and bin B_m covers the interval $(\frac{m-1}{M}, \frac{m}{M}]$.

Appendix D. Early exit ensemble on ResNet18

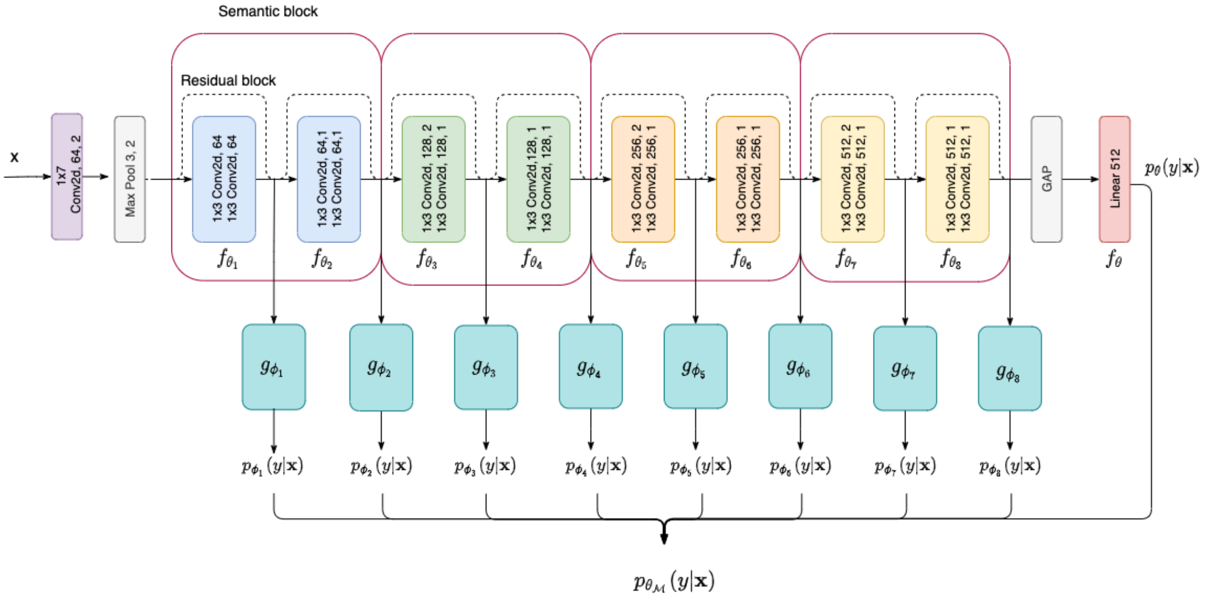


Figure 5: Representation of an early exit ensemble applied to a ResNet18 backbone decomposed into $B = 8$ blocks. The *Residual* strategy is used to create an ensemble of size $|\mathcal{M}| = 8 + 1 = 9$. For the *Semantic* strategy (best performing) only exits after each semantic block are included in the ensemble $|\mathcal{M}| = 4+1 = 5$. Block indexes match the exit indexes i in Table 3.