Uncertainty Quantification in Machine Learning for Biosignal Applications - A Review

Ivo Pascal de Jong, Andreea Ioana Sburlea and Matias Valdenegro-Toro
Department of Artificial Intelligence, Bernoulli Institute, University of Groningen, The Netherlands

***Abstract*—Uncertainty Quantification (UQ) has gained traction in an attempt to fix the black-box nature of Deep Learning. Specifically (medical) biosignals such as electroencephalography (EEG), electrocardiography (ECG), electroocculography (EOG) and electromyography (EMG) could benefit from good UQ, since these suffer from a poor signal to noise ratio, and good human interpretability is pivotal for medical applications and Brain Computer Interfaces. In this paper, we review the state of the art at the intersection of Uncertainty Quantification and Biosignal with Machine Learning. We present various methods, shortcomings, uncertainty measures and theoretical frameworks that currently exist in this application domain. Overall it can be concluded that promising UQ methods are available, but that research is needed on how people and systems may interact with an uncertainty-model in a (clinical) environment.**

I. Introduction

Standard Machine Learning (ML) systems such as Random Forests, SVMs, and Neural Networks typically produce single-point estimates for their classification task. This means that it is not possible to get an estimate of how likely the model is to be correct for a given sample. The inability to know how well a model will actually perform once deployed is part of the black-box skepticism that hinders implementation of Machine Learning methods in clinical settings [1]. Uncertainty Quantification (UQ) attempts to address this problem by adapting Machine Learning systems to also predict a measure of confidence for a given prediction. Over the past years this has been gaining traction in Computer Vision [2], but it is still only lightly explored in Machine Learning tasks that focus on Biosignals.

Applications using Biosignals such as electroencephalography (EEG), electrocardiography (ECG), electromyography (EMG) and electrooculography (EOG) can gain particular benefits from uncertainty quantification. Their signals are sensitive to artefacts that could corrupt the prediction of a Machine Learning system in unexpected ways. Uncertainty Quantification methods may help here by recognizing that the data is corrupted and withholding a classification, rather than giving an erroneous response. Another argument for the importance of Uncertainty Quantification is that the human interpretation of the signal requires substantial time investment. Uncertainty Quantification may identify which samples really need an expert opinion, and which may be automatically classified with minimal risk of error. To give an order of scale to the human effort: sleep scoring a patients EEG recording of an overnight stay will typically take a neurologist about 2 hours [3]. A Machine Learning system that can automatically classify the majority of the overnight stay with high confidence while identifying the parts that it is uncertain on may reduce this. Figure 1 shows where the uncertainty quantification aspect might occur in a Machine Learning based medical biosignal analysis.

With the value that this direction of research can bring this review attempts to identify how Uncertainty Quantification methods should be used in biosignal applications. Answering this question directly is impossible, but by investigating and critically assessing the way research is currently being conducted we attempt to provide some adjustments to the current directions and suggests new avenues to be explored in the future. Moreover, we intend to provide an overview of currently common methods as an entryway for researchers new to the topic of UQ in Biosignal processing, together with a simplified end-to-end guide for implementing, applying and evaluating uncertainty.

In the rest of this section we explain how the literature review was performed to offer some usability, and we end the section with a thorough explanation of what uncertainty is in Subsection I-B. Section II discusses the different Machine Learning algorithms for quantifying uncertainty. A large part of this is devoted to Bayesian Neural Networks and its various implementations. It discusses the other available algorithms in decreasing other of prevalence. Section III provides different ways a numerical measure of uncertainty can be extracted from a predicted distribution generated by some of the uncertainty quantification methods. Then, Section IV describes different ways uncertainty has been used in the biosignal domain. Together, Sections II-IV form the main body of findings and reflect how uncertainty quantification has been used for biosignals. The paper closes with a guide for uncertainty quantification in Section V and some identified open challenges in Section VI, to close with a brief conclusion in Section VII.

A. Search Method

To ensure reproducibility we used a systematic review. A first search had a higher level structure of $((\text{Uncertainty Quantification} \wedge \text{Machine Learning}) \vee \text{Bayesian Neural Networks}) \wedge \text{Biosignals}$. However, it was found that a line of research [4]–[6] uses the term "Bayesian Neural Networks" erroneously to describe classical Neural Networks trained with Bayesian Regularization [7]. A second search was performed without the Bayesian Neural Networks disjunction.

To ensure good coverage of the review various synonyms and abbreviations were used for each term. Specifically for

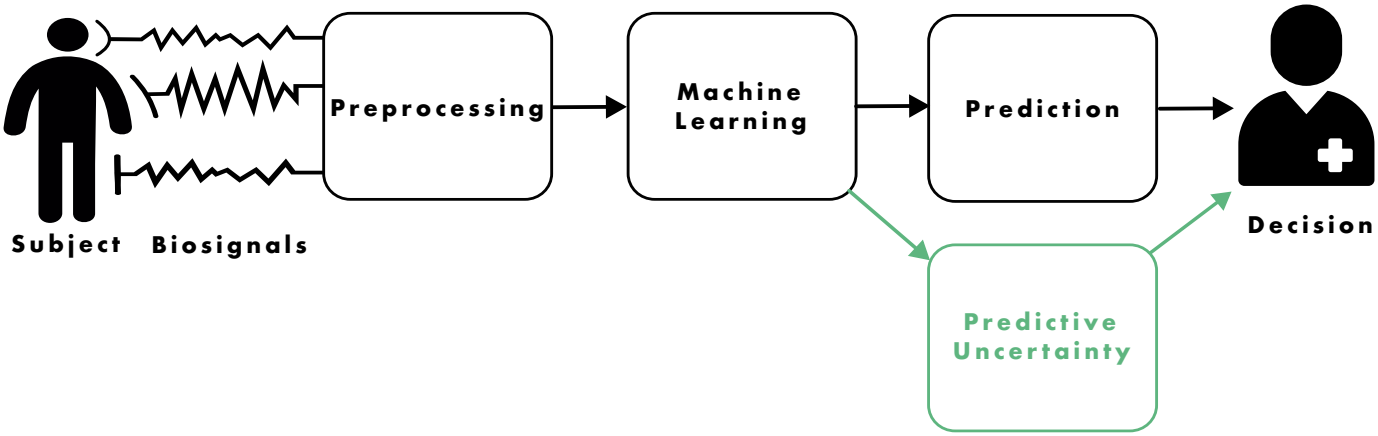


Fig. 1. Position of Uncertainty Quantification in medical biosignal interpretation. The uncertainty of a prediction may be considered by a clinician to improve diagnosis, or predictions with very low uncertainty may be accepted without human involvement.

the Machine Learning term several Neural Networks methods were used, and various Machine Learning models such as SVM, Random Forest and Fuzzy Logic. For the application domain we searched on the following terms: EEG, ECG, EOG, EMG, BCI and fNIRS, although no fNIRS papers were found. The choice of these terms was selected for the consistent modality, as each of them cover data from a set of time series from different locations.

Works that did not touch on the predictive uncertainty of a Machine Learning model, or that did not cover one of the relevant biosignals were rejected from the review. The two searches were applied to the databases: Web of Science, Scopus, IEEE Xplore and PsycINFO. Manual filtering by abstract and title resulted in a total of 59 papers, of which 35 met the criteria. 14 papers used the Bayesian Neural Networks term erroneously, five did not look at predictive uncertainty of a ML model, and two papers did not concern a relevant biosignal. Three papers looked at different biosignals, but were kept due to their interesting application of uncertainty quantification. The included and not included papers are visualised in Figure 2. The search covers any work before October 2023.

Figure 3 shows an overview of the results from this search. It shows that since 2020 there has been an increase in the use of Uncertainty Quantification, but overall there is not a lot of research in this intersection of Uncertainty Quantification and Biosignals.

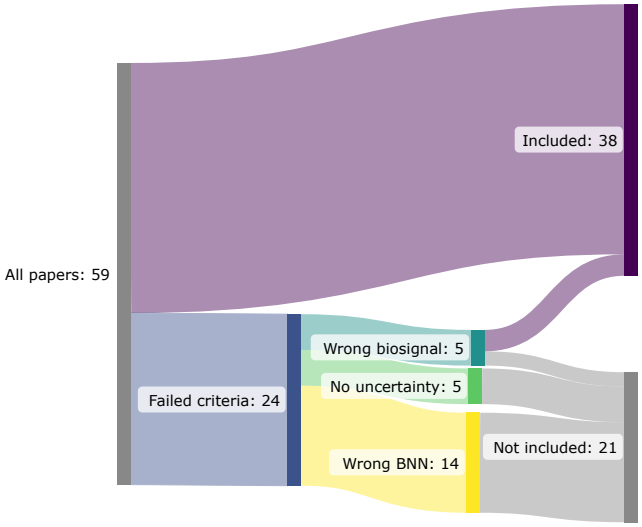


Fig. 2. The flow of papers that were covered in the systematic literature search, divided by rejection criteria.

B. Fundamentals of Predictive Uncertainty

Before going into the specific Machine Learning models that can quantify uncertainty for a given prediction, it is important to first understand what uncertainty really entails. Hüllermeier *et al.* [8] explains how predictive uncertainty can arise from two conceptual sources: aleatoric uncertainty and epistemic uncertainty. Aleatoric uncertainty[1] (also known as stochastic uncertainty) is the uncertainty that is inherent in the data. This kind of uncertainty cannot be reduced by having a better model, or by having the data be evaluated by a human expert. It comes from an imperfect predictive relationship between the features $\mathbf{X}$ and the to-be-predicted label $\mathbf{y}$. A toy example of aleatoric uncertainty is predicting the outcome of a coin toss. Even with arbitrarily many training samples, the uncertainty will not decrease.

In more detail, we may subdivide aleatoric uncertainty into three sources. Firstly, there may be noise in the features. This could be due to artifacts, sensor noise or sensor failure. The second is label noise. In virtually every dataset there will be training samples that have been mislabeled. For classifiers used to aid diagnosis this is largely due to ground truth labels that have been annotated with imperfect expert diagnosis. Third, there is the possibility that even noiseless features and noiseless labels simply do not have a completely predictive relationship. It may be inherently impossible to uncover certain fine-grained cognitive processes with complete certainty from

[1]Aleatoric is derived from the Latin word ”alea”, meaning dice

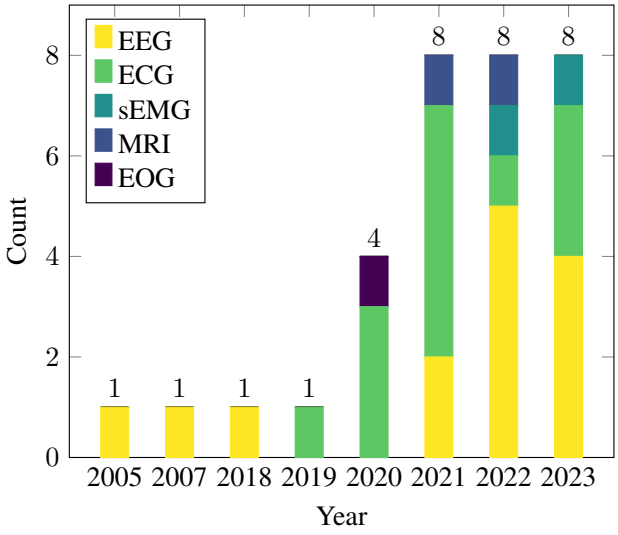


Fig. 3. Histogram of the number of papers per year using Uncertainty Quantification for each Biosignal. The search was conducted in September of 2023, so more papers for 2023 are expected. Overall this shows an increase in the number of Biosignal papers using Uncertainty Quantification. This shows an increase in the popularity of Uncertainty Quantification methods, although the amounts are still small.

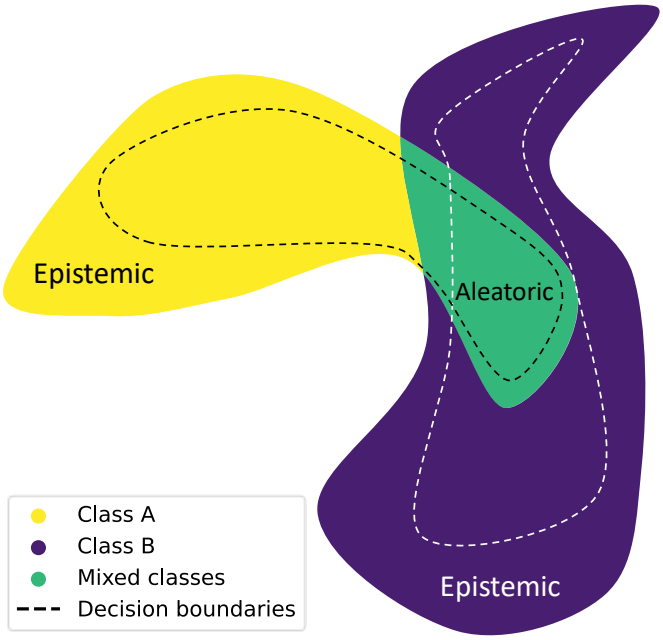


Fig. 4. Aleatoric and epistemic uncertainty in the 2D feature space of a classification tasks. The purple and yellow indicate the different class distributions, and the green area in which the classes overlap indicates aleatoric uncertainty. The dashed lines indicate a possible decision boundary. The part of the class distributions that fall outside of the decision boundary has not yet been learned. These parts show epistemic uncertainty.

electricity on the scalp, even if this task is absent of any noise in the recordings and has an infinite amount of perfectly labeled training samples.

Epistemic uncertainty (also known as model uncertainty) is the uncertainty that comes from a lack of knowledge. In this case a better model or a human expert would be able to make an accurate prediction. This kind of uncertainty may arise when a model is applied to data that is different from the data it was trained on, which is referred to as out-of-distribution [9]. Other causes of epistemic uncertainty include limited training data or model misspecification. Unlike aleatoric uncertainty, epistemic uncertainty does decrease with an increase in training samples. This property is explicitly used when epistemic uncertainty is measured by the Mutual Information between the model and a new hypothetical training sample in Section III-D

In biosignal contexts epistemic uncertainty commonly comes from limited labeled training data. It also arises when a model fails to generalise. This happens either across subjects, across recording hardware, and possibly even across clinics. Moreover, some tasks suffer from concept drifts where the signals that should be identified as a certain class may change over time or over contexts. In EEG research this can exhibit as cross-session variability.

The distinction between aleatoric and epistemic uncertainty is made clear in Figure 4, which shows how aleatoric and epistemic uncertainty arise in classification. In this case we see that in the area of feature space where both classes occur, aleatoric uncertainty arises. Epistemic uncertainty arises as the model cannot perfectly learn the distribution of the classes in feature space.

Van Gorp *et al.* [10] emphasises the need for this distinction in sleep stage classification, although this need also applies to other areas. They explain how aleatoric uncertainty should be addressed differently than epistemic uncertainty. If there is high aleatoric uncertainty for a given sample, then there is no use in asking a colleague for a second opinion, or for extra training on how to do sleep scoring. For a model, this means that more training samples will not improve performance. Instead the approach to improving model accuracy is to either collect cleaner data to improve the correlation, or to collect additional features such as age or known pathologies to make the classes more separable. For epistemic uncertainty more training data, better models, or asking an expert for help can improve performance.

In Section II we will discuss how these different kinds of uncertainties can present differently in some ML methods, which can be used to help inform what needs to be done to make a more confident prediction, or to decide how to respond to samples that are classified with aleatoric or with epistemic uncertainty.

1) Limitations of Aleatoric and Epistemic Uncertainty : Understanding aleatoric and epistemic uncertainty is sufficient to follow most of the current Uncertainty Quantification methods in the biosignal domain. However, it is important to know that is only a perspective, and not a fundamental property of uncertainty. The rest of this sections outlines some of the limitations of this perspective and shows where this perspective fails and what alternatives exist. While this is not required to understand the current methods, it may be helpful for the development of novel methods and offers a broadened concept of uncertainty.

Since the aleatoric-epistemic perspective is only a perspective on uncertainty there are other ways to look at uncertainty.

Some of these alternatives fit into the aleatoric-epistemic framework, but others do not. For example, in Section II-D1 we discuss Prior Networks, where the epistemic uncertainty is further split into *model uncertainty* and *distributional uncertainty*. Meanwhile Bishop *et al.* [11] makes a distinction between *discriminative* and *generative* models, where the former learns a decision boundary between the classes, the latter learns the class likelihood in feature space. Under these generative models samples with low likelihood for either class may be considered uncertain. However, this does not intuitively fit into either aleatoric or epistemic uncertainty. Bayesian Neural Networks finding an uncertainty decision boundary can be classified as discriminative models, whereas feature density methods can be classified as generative models for Uncertainty Quantification [12], [13].

We can explore the limitations of aleatoric and epistemic uncertainty with a toy example. Consider fitting a linear regression model to samples from a quadratic polynomial without any noise. With infinite training samples we can be certain about the optimal parameters for the linear model, thus resulting in zero epistemic uncertainty. The uncertainty that remains is aleatoric uncertainty, which is considered to be irreducible. However, while the uncertainty is not reducible through increased training samples, it is reducible by having a nonlinear model. Given this, we should be careful when saying aleatoric uncertainty is irreducible.

The problem here is that aleatoric and epistemic uncertainty are measured for a modelling task given a set of features and a model architecture, but this may not be the task we need to understand the uncertainty for. In a practical setting, we might want to know the uncertainty of a classification for an arrhythmia, which should be invariant to the feature extraction and model architecture chosen by the modeller. This shows that aleatoric and epistemic uncertainty can easily be confused.

2) Uncertainty in Terms of Evidence: One alternative perspective on uncertainty is discussed in the literature. Lin *et al.* [14], distinguishes between uncertainty from *vacuity* and from *dissonance*. This comes from the domain of Subjective Logic [15]. Here, vacuity is the absence of evidence for a prediction. Dissonance arises from conflicting evidence. Lin *et al.* [14] describes these in a context of evidence-based Machine Learning. Similar to the aleatoric and epistemic uncertainty one can use this distinction to make decisions on how to improve the quality of a model.

This perspective of uncertainty does not directly unify with aleatoric and epistemic uncertainty, but there are some similarities. Firstly, aleatoric uncertainty has a similarity with dissonance. In Figure 4 we see that the part of the feature space that fits both classes is considered aleatoric uncertainty. In terms of evidence, we may say there is evidence for both classes, and thus there is dissonance. With this it may be said that dissonance is likely to co-occur with aleatoric uncertainty. However, they are not perfectly identical, as there may be parts where there is evidence for both class, but the model is still epistemically uncertain about which classification should come from this.

Similarly, when there is a lack of evidence for either class there is vacuity, but it is also likely that these samples have epistemic uncertainty as no evidence has been learned yet. Again, there may be cases where vacuity and epistemic uncertainty do not co-occur, such that a model may in fact be very certain that there is no evidence.

With this starting point of how the aleatoric-epistemic perspective and the vacuity-dissonance perspective may be unified, we encourage experimental research in this area. Specifically, it would be interesting to quantify how strongly these concepts are related in existing datasets. Moreover, attempts to measure the epistemic and aleatoric uncertainty of the evidence for a classification may result in novel Machine Learning methods for Uncertainty Quantification that unify these perspectives of uncertainty. Specifically, new methods combining ideas from Bayesian Neural Networks (Section II-B) and Prior Networks (Section II-D1) could give predictions with more complete explanations of uncertainty.

II. Methods for Uncertainty Quantification

As most of the development of Uncertainty Quantification methods happens in the field of Computer Vision [2], it is no surprise that the Machine Learning models for which Uncertainty Quantification is defined are models that perform well in Computer Vision. As a result we find most works build on Neural Networks. Specifically this review found many Convolutional and Recurrent Neural Networks. On overview of the popluation of different Neural Network types is given in Figure 5.

With the vast majority of models being Neural Networks, the Uncertainty Quantification methods are also mostly intended for Neural Networks. An overview of the most common methods covered is given in Table I, although they are discussed in much more detail below.

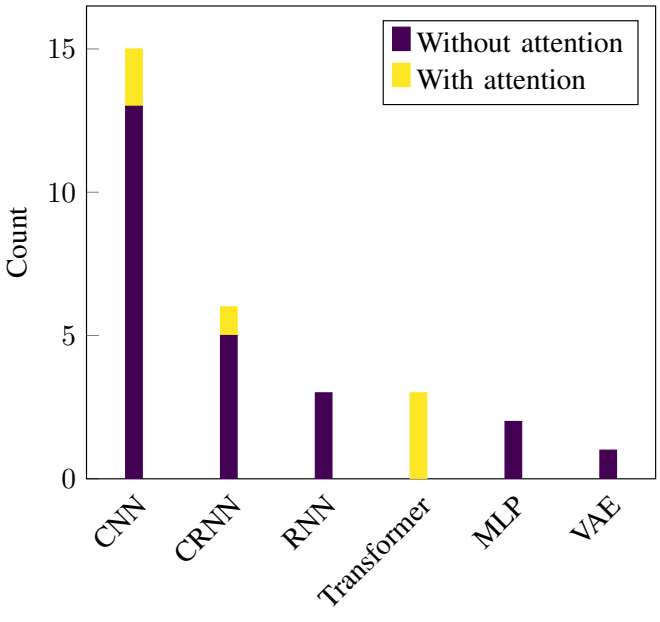


Fig. 5. Popularity of various Neural Network architectures in this review. Models with at least one convolutional or recurrent layer are respectively labeled CNN or RNN. Models with both are labeled as CRNN. Yellow indicates models with attention layers.

In accordance with the literature, this section mostly discusses Deep Learning methods for Uncertainty Quantification. First, the concept of Bayesian Neural Networks is explained, including the range of different implementations. Bayesian Neural Networks are the most common method for Uncertainty Quantification, and lends itself well to interpretation through the lens of aleatoric and epistemic uncertainty. Next, we will discuss some more common Uncertainty Quantification methods such as Varaitional Autoencoders, Direct Unceretainty Quantification and Gaussian Process Regression. We also discuss post-hoc uncertainty calibration methods, and end this section with a list of the less common and experimental methods for uncertainty quantification that have been used for Biosignals. All-together, this section gives a complete overview of all the Uncertainty Quantification methods that have been used for Biosignals.

A. Notation for Softmax Uncertainty

Standard Neural Networks give point-estimate predictions for a given sample. In regression, this prediction is a scalar with no indication of uncertainty or expected error. However, in classification with standard Neural Networks a Softmax activation function is often used such that the prediction is given as

$$p(y=c\,|\,x,\theta)=\frac{\exp(f_c^\theta(x))}{\sum_c^\prime \exp(f_{c'}^\theta(x))}. \tag{1}$$

Where f_c^θ predicts the logits for a given input x, as parameterized by θ. To ease notation we introduce the predicted probability of a class c as

$$p_c \coloneqq p(y=c\,|\,x,\,D), \tag{2}$$

which in the case of a standard Neural Network with parameters θ learned on dataset $D=\{\mathbf{X},\mathbf{y}\}$ simplifies to $p_c=p(y=c\,|\,x,\theta)$.

Before going into how uncertainty is modelled in Bayesian Neural Networks, it is important to be aware that predicting class probabilities, rather than directly predicting a class label already quantifies uncertainty. The uncertainty that arises here (e.g. $p_c=0.5$) can come from two possible sources. Firstly, it can come from samples in dataset D where the annotated ground truth is uncertain such that $p_c=0.5$. Alternatively, it is due to restricted degrees of freedom in the model architecture such that two similar samples in $\{x,x'=x+\epsilon\}\in\mathbf{X}$ with different labels $\{y,y'\neq y\}\in\mathbf{y}$ cannot be given a different classification. Both of these uncertainties are aleatoric uncertainty. Epistemic uncertainty cannot be captured in this framework. As such, generalization error, adversarial examples and out-of-distribution data will cause overconfident predictions.

B. Bayesian Neural Networks

Given a starting point of aleatoric uncertainty with softmax, we move towards quantifying epistemic uncertainty with Bayesian Neural Networks. The foundational difference is the way both methods look at learning the parameters. In the standard Neural Network the parameters θ is learned from the space of all possible sets of parameters Θ to minimize a loss function $\mathcal{L}(\theta,D)$. The loss function primarily measures the error between the predictions and the annotated ground truth. Under Bayesian Neural Networks, instead of considering a single optimized set of parameters θ, we consider a distribution of all possible sets of parameters in Θ. Since some parameters are more likely under dataset D than others, we also consider the likelihood of each set of parameters. This results in the integral

$$p_c=\int \underbrace{p(y=c\,|\,x,\theta)}_{\text{Aleatoric}}\underbrace{p(\theta|D)}_{\text{Epistemic}}\,d\theta. \tag{3}$$

.

From this the epistemic uncertainty as the probability distribution of the parameter vector $p(\theta|D)$ also becomes apparent. This integral combining aleatoric and epistemic uncertainty is referred to as the posterior predictive distribution.

Some approximations of Bayesian Neural Networks such as MC-Dropout and Ensembles are based on this equation. They sample multiple parameter vectors θ which are all trained to maximise $p(\theta|D)$ through e.g. the negative log likelihood. From each parameter vector predictions are made, resulting in a distribution over predicted probabilities. The variance between classes (or disagreement between samples models) now captures epistemic uncertainty.

To complete the picture of the Bayesian Neural Network, and to show why it is called "Bayesian", we take the dataset D as Random Variables $\{X,Y\}$ and deconstruct the posterior $p(\theta|D)$ with Bayes theorem as

$$p(\theta|X,Y)=\frac{p(Y|X,\theta)p(\theta)}{p(Y|X)}. \tag{4}$$

The evidence term $p(Y|X)$ is intractable[2]. Fortunately, it is a constant for a given dataset, so we can optimize θ only on the likelihood and the prior. The likelihood is determined by the model fit to the data and may be computer through a loss function. The prior $p(\theta)$ can be selected to match assumptions about the modelling task. For example, it has been used to explain regularization terms in a loss function such as L2-regularization where large weights are considered less likely than small weights [11]. This property also makes Bayesian Neural Networks well suited for injecting prior domain knowledge into a Machine Learning task. Unfortunately, none of the reviewed literature applied this property, despite the potential benefits this may have given the amount of domain knowledge available, and the often limited size of datasets.

The last implementation of Bayesian Neural Networks that we will discuss in this section is Variational Inference. Unlike the MC-Dropout and Ensembles which try to draw samples from $p(\theta|D)$, Variational Inference tries to learn an approximation of the actual distribution of $p(\theta|D)$.

The rest of this section explains these three methods in more detail to provide a conceptual understanding, and to show the limitations of each method.

[2]This would result in an integral for each parameter of the Neural Network such that $p(Y|X)=\int_{\theta_1}\int_{\theta_2}\dots\int_{\theta_D}p(Y|X,\theta)d\theta_1 d\theta_2\dots d\theta_D$ where D represents the number of dimensions of the parameter vector θ.

1) MC-Dropout: Dropout [16] has been a prominent regularization method in Deep Learning applications. During training with dropout, some nodes (either hidden or input) have some probability p to be dropped (i.e. activation set to 0). This adds noise to the training procedure and has been thoroughly shown to prevent overfitting in various Deep Learning domains.

Normally, the dropout is removed during inference to keep it deterministic and prevent dropping important information. MC-Dropout (Monte Carlo Dropout) [17] keeps this dropout during inference. Gal *et al.* [17] shows that each forward pass with dropout is a sample from $q(\theta) \approx p(\theta|D)$.

Dropout can be considered as a special probability distribution over parameter vectors, because dropping a node is equivalent to setting all the incoming or outgoing weights of that node to zero. With this we can think about the sampling of MC-Dropout as sampling from an unusual probability distribution over weights. Due to the training process, each of these samples is optimized to be as-likely-as-possible. When we then make predictions with MC-Dropout, it is effectively sampling from $q(\theta)$.

This approximate sampling of the posterior then results in a distribution of predictions. When samples from the posterior result in different predictions, this can be considered as epistemic uncertainty.

A commonly considered advantage of MC-Dropout is the simplicity with which it can be applied to a Deep Learning model. Many Deep Learning architectures are already trained with dropout, so MC-Dropout can easily be applied without even re-training the model. The big disadvantage however is that it takes many forward passes[3] for the MC-Dropout to capture the predictive distribution, making inference computationally expensive.

2) Ensembles: Although MC-Dropout is technically an ensemble of weight-sharing NNs [16], the idea of an Ensemble as an approximation of a Bayesian Neural Network originally takes the form of several independently trained Neural Networks following the same architecture and trained on the same data [20][4]. Only a limited number of models [5] is needed to achieve an acceptable approximation of the weight distribution. This keeps the computational cost relatively cheap compared to MC-Dropout at inference time, but performing the training several times and storing several models in memory may be prohibitively expensive.

An Ensemble may be interpreted as a small set of samples of the parameter distribution $p(\theta|D)$[23]. Each of these samples are trained to the data, so each sample should reflect a parameter vector with high posterior probability. While this matches the parameter distribution of the BNN a lot sparser than MC-Dropout, it may actually be more efficient as it samples only well optimized parameters.

[3] $T = 50$ is recommended, but anywhere from $T = 10$ to $T = 1001$ may be used. [17]–[19]

[4] Originally Deep Ensembles were introduced as a non-Bayesian method for UQ [20], but it has since been shown that it can be considered as a very coarse approximation of a BNN [21], [22].

[5] As an example: Lakshminarayanan *et al.* [20] uses an ensemble of 5 models.

Much like MC-Dropout, ensembles are conceptually simple, and intuitive to reason about. It aligns with human analogies where when all the models/people disagree, then there is a lot of (epistemic) uncertainty. Contrastingly, situations where all models/people agree must be very certain.

Xia *et al.* [18] shows that ensembles represent epistemic uncertainty under distributional shifts better than MC-Dropout, and that the accuracy of the predictions are also better. They do this on various Biosignal classification tasks such as auditory COVID-19 classification, respiratory abnormality detection and heart arrhythmia detection. By providing various forms of dataset shift, they concur with findings from computer vision and language models [24], suggesting that ensembles may be better at presenting epistemic uncertainty under dataset shifts.

3) Variational Inference: In variational inference (VI) the intractable posterior distribution $p(\theta|X, Y)$ is approximated with a simpler distribution $q_\omega(\theta)$. In the case of a Bayesian Neural Network $p(\theta|X, y)$ is an arbitrary high dimensional distribution with no known parametric form, where each dimension corresponds to a weight in the neural network. This makes it impossible to sample from directly. A possible approximation through $q_\omega(\theta)$ might say that each weight is a Gaussian distribution with a mean and a variance. The goal is then to optimize the parameters ω for the high-dimensional Gaussian, so that it is similar to the true posterior. With this, we can then sample models from $q_\omega(\theta)$ to predict class probabilities according to the integral in Equation 3.

In order to make a good approximation of the posterior VI needs to minimize the Kullback-Leibler divergence (KL-divergence) between the approximate distribution $q_\omega(\theta)$ and the true distribution $p(\theta|X, Y)$. The KL-divergence measures the distance between two distributions. In this case it is given as

$$KL(q_\omega(\theta) \,||\, p(\theta|X, Y)) = \int_\Theta q_\omega(\theta) \log \frac{q_\omega(\theta)}{p(\theta|\, X, Y)} d\theta. \quad (5)$$

This minimization task still contains the posterior distribution term $p(\theta|X, Y)$ which is intractable as discussed in Equation 4. By rearranging the KL-divergence into the evidence lower bound (ELBO) we instead get the maximization task [2]:

$$\text{ELBO}(\omega) := \int_\Theta q_\omega(\theta) \log p(Y|X, \theta) d\theta - KL(q_\omega(\theta) \,||\, p(\theta)) \quad (6)$$

The prior chosen for $p(\theta)$ may still be defined by the modeller, and can have an impact on the quality of the model. For the purposes of transfer learning, this prior may even be a learned distribution on another dataset (see [25]).

While Variational Inference is a better approximation of a Bayesian Neural Network than Ensemble-based methods, it is often much more expensive to train and do inference on. Moreover, implementing it introduces many new decisions to make. The form of the posterior approximation needs to be chosen, as well as the prior for its parameters. Moreover, measuring the evidence lower bound requires Monte-Carlo sampling from the approximated posterior. The number of samples to use is a balance between computational cost per

epoch, and the stochasticity of the gradient descent. Lastly, having many Bayesian layers in a Deep Bayesian Network can cause the loss to become numerically unstable.

Only one reviewed work tested Variational Inference and compared it against other methods of Uncertainty Quantification. Xia *et al.* [18] found that Variational Inference and Ensembles together showed the best classification accuracies for an arrhythmia detection task and on a separate respiratory abnormality detection task, but that Variational Inference performed poorly on an auditory COVID-19 detection task compared to Ensembles. They also found that all methods performed poorly at quantifying uncertainty under their synthetic dataset shift.

While Variational Inference introduces several challenges, the ability to directly use an informative prior to inject domain knowledge may have considerable advantages. Moreover, while using smaller Neural Networks, VI may prove to be the best approximation of a Bayesian Neural Network currently available.

C. Variational Autoencoders

Variational Autoencoders [26] are a specific type of neural network architecture. It has an encoder which receives a high-dimensional input x and encode it into a lower dimensional latent distribution $p(z \mid x, \theta)$. It does so by predicting a mean and a variance for each dimension of the latent distribution, from which latent representations $z \sim p(z \mid x)$ can be sampled. A decoder network then reconstructs the encoding back into the original dimensionality of the input to achieve $x' = f_{\theta'}(z \sim p(z|x, \theta)) \approx x$.

The VAE model is trained to minimize the difference between the input x and the reconstructed output x'. As a result, the latent distribution $p(z \mid x, \theta')$ should be a lower-level representation of the salient features that exist in the data. This works under the concept of *manifold learning* where many of the points on the high-dimensional input have near-zero likelihood, and that actually a lower-dimensional manifold should be able to capture the distribution of the actual data.

VAEs were originally intended a generative unsupervised learning models, and were not invented with Uncertainty Quantification in mind. However, because the latent representation is a distribution which can be sampled from, researchers have constructed various methods to extract uncertainty from that stochasticity. Belen *et al.* [27] uses a trained VAE on a dataset of segments of ECG with and without expert annotated atrial fibrillation. They then use the sampled latent representations as input for a multi-layer-perceptron to do the classification task as

$$p(y = c \mid p(z|x, \theta), \theta'). \tag{7}$$

This results in a distribution of probabilities, of which the variance is used to measure aleatoric uncertainty.

Van De Leur *et al.* [28] apply Principal Component Analysis to get a 2-dimensional visualization of the latent space as a method for interpretability. They show how various diagnoses would show in the latent representation, so that a sample on the boundary of two classes, or far away from any known classes can be qualitatively assessed as uncertain.

D. Direct Quantification of Uncertainty

In contrast to the previous methods which rely on stochasticity to quantify uncertainty, there is also a set of methods that aim to directly predict uncertainty as part of the model training task. The most intuitive form of this is heteroscedastic uncertainty quantification for regression [29]. In these models, the Neural Network not only attempts to learn a predicted regression value, but it has a separate output for the predicted error. This results in a prediction, paired with a measure of aleatoric uncertainty. Taking $\mu_\theta(x)$ as the predicted mean and $\sigma_\theta(x)$ as the predicted variance for a sample, the predicted value y is given as

$$y = \mathcal{N}(\mu_\theta(x), \sigma^2_\theta(x)). \tag{8}$$

Such a model is then trained with a loss function that optimizes both the predicted mean and the variance. The Gaussian Negative Log-Likelihood is the simplest, but alternatives have been proposed [29].

In this category where Neural Networks have a modified output structure to give uncertainty quantification we found two papers presenting two uncertainty quantification methods, which will be discussed below. The first is Prior Networks, which expands on the concepts of aleatoric and epistemic uncertainty. The second is Evidential Machine Learning, which looks at uncertainty in terms of vacuity and dissonance.

Both methods predict for c classes the parameters $\alpha_c > 0$ for a Dirichlet distribution. From this distribution, class probability vectors may be sampled such that each class probability is in the range $[0, 1]$ and that all class probabilities together sum to 1. Small values for all α_c result in plots similar to 7b and d, whereas large values result in plots similar to 7 a and c. The difference between the different α_c skew the distribution towards a class.

Through the Dirichlet distribution these models produce the same second-order uncertainty as found in the Bayesian Neural Networks, so that the predicted class probabilities are subject to uncertainty as well.

Below we will explain how Prior Networks and Evidential Machine Learning use Dirichlet distributions with two different perspectives on uncertainty.

1) Prior Networks: Prior networks [30] work within the framework of aleatoric and epistemic uncertainty, but make a further distinction beyond these two. Under the Bayesian Neural Network framework we consider the uncertainty due to generalization error, such as when the model is evaluated under out-of-distribution data, as part of the epistemic uncertainty. Prior networks instead add the term *distributional uncertainty*. This then gives

$$p_c = \int\int \underbrace{p(y = c \mid \mu)}_{\text{aleatoric}} \underbrace{p(\mu \mid x, \theta)}_{\text{distributional}} \underbrace{p(\theta \mid D)}_{\text{epistemic}} d\mu \, d\theta. \tag{9}$$

This formulation follows from Equation 3, but now models a specific uncertainty for the mismatch between the training and testing data. Prior Networks only specifically learns the distributional uncertainty and the aleatoric uncertainty, but not the epistemic uncertainty.

In practice, these prior networks for classification learn the parameters α_c for a C dimensional Dirichlet distribution. However, standard cross-entropy loss does not enforce a difference in *sharpness* for out-of-distribution and in-distribution samples. Therefore, prior networks are multi-task trained to get a *flat* prediction for out-of-distribution samples and a *sharp* prediction for in-distribution samples.

Naturally, this requires specific out-of-distribution samples, but the true out-of-distribution samples are unknown. Instead one may use synthetic samples [31] or real other datasets [32] as the out-of-distribution samples.

2) Evidential Machine Learning: Similar to Prior Networks, Evidential Machine Learning [33] uses a single-point model to predict the parameters for a c dimensional Dirichlet distribution. Unlike the Prior Network, it instead looks at uncertainty through the Dempster-Shafer Theory of Evidence (DST).

An Evidential Neural Network attempts to predict *evidence* for each class $c \in C$ as a C-dimensional output from a ReLU activation function. The value for each class gives the amount of *evidence* for that class. These values are then used as the parameters α_c for the Dirichlet distribution.

The uncertainty here is split into *vacuity* and *dissonance* [14], [32]. Vacuity is the absence of evidence causing uncertainty. Like standard Neural Networks with a Softmax activation function, Evidential Machine Learning assumes that exactly one class must be the ground truth. The absence of evidence for any of the classes would then result in a form of uncertainty referred to as vacuity. The opposite uncertainty is *dissonance*, which occurs when the model has found evidence for multiple classes, which is not in line with the assumption of mutual exclusivity.

Here we say that small α_c for all classes is a lack of evidence and is therefore vacuity, while large α_c for all classes is dissonance. This gives a more direct interpretation of the uncertainty in the Dirichlet distribution compared to Prior Networks which requires explicit in-distribution and out-of-distribution datasets to quantify uncertainty through the Dirichlet distribution. Meanwhile vacuity and dissonance are an emergent property of the model.

E. Gaussian Process Regression

Gaussian Process Regression [34], [35] is a non-parametric regression method that considers epistemic uncertainty. It assumes a Gaussian prior over the dependent variable Y. It also assumes that the samples in the training data D are drawn without measurement error. This leaves uncertainty in the regression between and outside training samples, and gives more certainty at points close to the training samples.

As more training samples get collected, the epistemic uncertainty will decrease. The assumption that data are drawn without measurement error does naturally lead to an inability to capture aleatoric uncertainty.

F. Post-hoc Calibration

Post-hoc calibration methods [36] look at uncertainty only in terms of the predicted probability for each class, and addresses how this may deviate from the observed probability. A class prediction with $p = 0.75$ should be correct 75% of the time, but this does not hold for standard softmax classification. By learning a mapping from the softmax output to observed probabilities post-hoc calibration methods provide a more directly interpretable and workable measure of uncertainty. Post-hoc calibration methods aim for an optimal calibration such that

$$p(y = c \mid p_c) = p_c. \tag{10}$$

The mismatch between the predicted probability and the observed probability is measured by the Expected Calibration Error (ECE) [36]. This is usually computed over bins of probability predictions, since most datasets will have one-hot encoded class labels.

While the ECE is often used simply as a measure of how well calibrated the predicted probabilities of a model are, the concept of binning probabilities and observing the *true probability* in a dataset can also be used to shift the predicted probabilities for a certain bin up/down depending on the calibration error.

Various methods for post-hoc probability calibration methods exist [36]. Temperature Scaling is the simplest (and often the most effective) method of post-hoc calibration, which determines the *softness* of the Softmax function. It does so by introducing a hyperparameter τ to get the scaled Softmax function

$$p(y = c \mid x, \theta) = \frac{\exp(\frac{f_c^\theta(x)}{\tau})}{\sum_c' \exp(\frac{f_{c'}^\theta(x)}{\tau})}. \tag{11}$$

While this can make the probabilities a better reflection of the true probability, which is a concern for clinical deployment of ML models [37], it would not provide any benefit to methods that use a threshold against the uncertainty to abstain from predicting a sample. Since calibrated probabilities are monotonically increasing with the uncalibrated uncertainties, moving the threshold will achieve the same decision boundary.

Calibration methods do not distinguish between aleatoric and epistemic uncertainty. Epistemic uncertainty can be accounted for when the calibration data is different than the training data. However, since calibration methods only calibrate homoscedastic uncertainty, it may fail to present substantial epistemic uncertainty on out-of-distribution data.

G. Non-standard UQ Methods

Above, a selection of common and well studied methods for Uncertainty Quantification is discussed. This does not cover all the UQ methods that were encountered in the review. Below we continue the description of uncertainty quantification methods with some non-standard encountered in the reviewed literature to provide an exhaustive presentation of UQ research on Biosignals. However, we consider these models to be more experimental or niche, and may not transfer as easily or reliably as some of the previous methods.

TABLE I
AN OVERVIEW OF THE DIFFERENT UQ METHODS DISCUSSED IN II. DUE TO THE REDUCTIVE NATURE OF THIS TABLE, IT SHOULD ONLY BE CONSIDERED IN CONJUNCTION WITH SECTION II. COMPUTATIONAL COST OF UQ METHODS IS QUALITATIVELY GROUPED INTO 3 CLASSES. *None* HAS NEGLIGIBLE ADDED COMPUTATIONAL COST. *Small* HAS SOME ADDED COMPUTATIONAL COST E.G. DUE TO SLOWER CONVERGENCE OR TRAINING STEPS BEING MORE COMPUTATIONALLY EXPENSIVE. *Large* INDICATES SUBSTANTIAL INCREASE IN COMPUTATIONAL COST, SUCH AS 5 TIMES THE TRAINING COST, OR 50 TIMES THE INFERENCE COST.

Method	Model Agnostic	Epistemic UQ	Aleatoric UQ**	Training Cost	Inference Cost
MC-Dropout [17]	NN only	✓		None	Large
Ensembles [20]	✓*	✓		Large	Small
Variational Inference [38]	NN only	✓		Large	Large
Variational Autoencoder [26]			✓	Small	Large
Prior Networks [30]	✓	***	✓	None	None
Evidential Machine Learning [33]	✓		✓	None	None
Gaussian Process Regression [39]		✓	✓	Small	Small
Post-hoc calibration [36]	✓		✓	None	None

*Requires bootstrapping for non-stochastic training procedures. May perform poorly without local minima.
**Aleatoric uncertainty may still show in classification with Softmax.
***Generalization uncertainty through out-of-distribution dataset

1) Bayesian Model Averaging with Reversible-Jump MCMC: Schetinin *et al.* [40] attempted to classify EEG artifacts using a method based on Bayesian Model Averaging. They use Markov-Chain Monte Carlo to sample changes to a decision tree. These changes are any of 4 types: adding a split in the tree, removing a split in the tree, changing the variable a split is focused on, or changing the rule of a split. These changes are accepted or rejected based on the likelihood given the data. This consists of how well a given change improves the training classification, as well as how likely it is given a set prior.

As a measure of uncertainty the authors consider the entropy in the leaf nodes. The authors showed that subtracting a non-stationary component from the power of the subdelta band improved the accuracy of their model, but since the dataset is not specified and no other models are shown it is not possible to assess the quality of the model, nor the resulting entropies.

Another reviewed work also used the entropy of the leaf-nodes in a decision tree as a measure of uncertainty, but this also lacked interpretation [41] .

2) Majorization-Minimization and Hierarchical Bayesian Modelling: Bekhti *et al.* [42] compares Majorization-minimization and Hierarchical Bayesian Modelling and shows how they are fundamentally the same. Unlike the majority of work found in this review which try to learn an arbitrary function $p(y|x, D) = f_\theta(x)$, this work starts with the assumption that observed EEG recordings X are a linear combination of underlying sources G connected through a known linear forward propagation matrix G, with some Gaussian noise E such that $M = GX+E$. This results in a multi-task regression where we need to learn an optimal matrix X that minimizes the E. Without considering regularization this results in the optimization

$$\hat{X} = \arg\min_X \frac{1}{2}||M - GX||_F^2. \tag{12}$$

Majorization-minimization solves this by taking a random initialization, fitting a Taylor expansion to the cost function at that point, and then using the X^t that minimizes that Taylor expansion as the next initialization. To avoid overfitting $l_{2,p}$-norm regularization is used. This has the added benefit of promoting sparse solutions.

They are able to show that the full maximum a posteriori estimate of a Hierarchical Bayesian Modelling approach can be re-derived as a Majorization-Minimization optimization problem. From this insight, authors propose a method of sampling multiple initialization for the MM optimization, resulting in multiple sparse solutions to the inverse problem.

Using the multiple sparse solutions, together with how well they minimize the objective function, the authors are able to present various source attribution to an observed EEG or MEG signal, together with a measure of how (un)certain each solution is.

3) Bayesian Moderated Outputs: Based on Mackay [43], Mohamed *et al.* [44] compare Bayesian Moderated Outputs to a standard Multi-Layer Perceptron for the task of epileptic activity classification in sleep EEG recordings. The concept of Bayesian Moderated Outputs is that instead of having a single optimal parameter vector $\hat{\theta}$, a more robust method will have a Gaussian distribution of parameters around an optimum $\Theta = \mathcal{N}(\hat{\theta}, s^2)$. The hypothesis is that the mean prediction over these different models provides a better representation of the predicted probability.

Unfortunately, this did not lead to apparent better performance than a maximum-likelihood trained Multi-Layer Perceptron [44]. This was observed by using a rejection threshold of 0.9 for both models. The Bayesian Moderated Outputs did achieve slightly higher accuracy (up to 1 percent-point), but at the cost of rejecting up to 15 percent-point more samples from classification.

This concept has further been explored by Maddox *et al.* [45].

4) Neural Stochastic Differential Equations: Wabina *et al.* [46] propose a novel method called Neural Stochastic Differential Equations to learn an electrical conductivity model of the head based on MRI. Such conductivity models can be

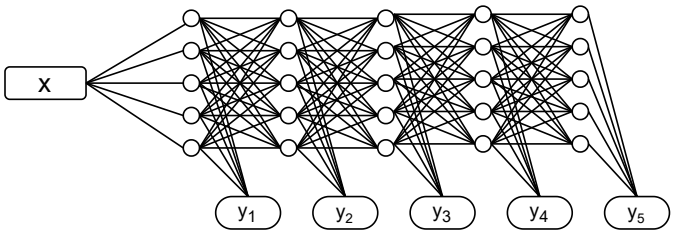


Fig. 6. A diagram showing the concept of Early Exit Ensembles [48]. There is a shared backbone network, from which Exit Branches make predictions. The architecture of the backbone, exit branches, and where those connect can be arbitrarily designed. Each exit branch makes an independent prediction. The distribution of predictions may be interpreted similar to a normal Ensemble.

used to inform the forward propagation of EEG signals as referred to in Section II-G2.

They use a class of Deep Neural Networks proposed in [47], which includes a split block consisting of a drift and a diffusion network to consider the Neural Network as a Stochastic Differential Equation. The drift network continues to attempt to optimize predictions, while the diffusion network predicts a heteroscedastic amount of Gaussian noise. The noise should be minimal for samples in the training distribution, and maximal for out-of-distribution samples. The result of the SDE-block can be sampled and passed through a final block of dense layers to reach a distribution of predictions. The complete Neural Network proposed is called SDE-Net.

An experiment on the Single Individual volunteer for Multiple Observations across Networks (SIMON) MRI dataset showed that SDE-Net outperformed Bayesian methods. However, the effect of epistemic uncertainty on the spread of the predictions and SDE-Net's ability to capture epistemic uncertainty is not investigated.

5) Early Exit Ensembles: As a quasi-ensembling method Campbell *et al.* [48] propose Early Exit Ensembles. Early exit ensembles work by taking any deep neural network and adding various *exit* branches to points of the network as illustrated in Figure 6. Each exit will have a global pooling operation and 2 dense layers. The idea is that each exit branch will try to learn to do the classification task (as an ensemble), but depending on the location on the *backbone* architecture they may learn on either lower or higher level features.

Like normal ensembling methods, the disagreement between the various classifiers corresponds to epistemic uncertainty. The advantage compared to normal ensembling is that the large amount of weight sharing can reduce the computational cost of training and inference, as well as the size of the model. The ways in which constructing an Early Exit Ensemble from an existing architecture affects the quality of the predicted uncertainty is an interesting avenue for research, which may be partly inspired by what is already known about early-exit neural networks (see [49], [50]).

6) Reconstruction Error: Martinez *et al.* [51] look at how to reconstruct an ECG signal based on bioimpedance recordings. Bioimpedance can be much easier to record, but also difficult for cardiologists to interpret. They propose a method where an Autoencoder uses the biosignals to construct the ECG morphology, but without correct amplitudes. Then a second autoencoder uses this amplitude-invariant data, and the original bioimpedance to reconstruct the ECG.

Since the amplitude corrected data should have the same morphology as the predicted ECG, any differences in morphology can be attributed to a generalization failure of the second autoencoder. Thus, the authors measure the Pearson correlation between the amplitude invariant and amplitude corrected data as a measure of uncertainty.

7) Fuzzy Logic: The systematic search found one work in the domain of fuzzy logic. In [52] they propose a method to automatically construct Fuzzy Cognitive Maps. These are directed graphs where each node represents a concept, and each weighted edge represents a causal relation with a strength $w_{i,j} \in [0, 1]$. Each node/concept will have an activation $A_i \in [0, 1]$ which spreads through the graph. Fuzzy Cognitive Maps aim to provide an interpretable medical decision support systems [53]. Sovatzidi *et al.* [52] propose a method for automatically constructing such a Fuzzy Cognitive Map for supporting diagnosis of depression based on EEG data. They cluster the frequency-band power in different scalp areas and use these clusters as the *concepts*, with one last *concept* being the level of depression.

They use a fuzzy clustering where each sample has partial membership of a cluster, resulting in partial activation of the corresponding concepts. The authors claim that this makes the classification system uncertainty-aware.

8) Assumed Density Filtering: Duan *et al.* [54] applies a more computationally affordable method for modelling data uncertainty called Assumed Density Filtering (ADF). Whereas Bayesian Neural Networks model a distribution for each weight, ADF takes a single-point solution for the weights, but has a distribution for the activations.

This is achieved by modelling the input as a Gaussian distribution around the single-point input features such that

$$\mathbf{z} = \mathcal{N}(x, \sigma^2). \tag{13}$$

Passing this as the input to a Neural Network results in distributions for each activation. Each activation is modelled by a mean and variance, where the variance corresponds to the uncertainty. This ultimately results in a mean (prediction) and variance (uncertainty) in the output. This method is intended to correspond to aleatoric uncertainty.

This can be applied to any neural network architecture. The assumed variance of the input can be optimised against the Negative Log Likelihood to get a better calibrated model.

For biosignals this corresponds well with sensor noise as this is explicitly modelled and propagated, but it may not work well with other sources of aleatoric uncertainty such as label errors.

Combined with a Bayesian Neural Network as done by Duan *et al.* [54] provides explicit modelling for both uncertainty of the model, and uncertainty of the biosignal recording.

9) Data Uncertainty Learning: As a method for aleatoric uncertainty, Data Uncertainty Learning [55] models uncertainty as a distribution in an embedding such that

$$p(z|x) = \mathcal{N}(x; \mu, \sigma^2 I). \tag{14}$$

Here a Neural Network learns an embedding as a Gaussian distribution. This method holds similarities to a Variational Autoencoder, as both methods learn a Gaussian distributed representation of the input.

The primary differences are the architecture and the learning task of the model. Where a VAE normally has structural symmetry between the encoder and decoder, using the embedding as a bottleneck, Data Uncertainty Learning has the embedding as the penultimate layer. For Data Uncertainty Learning the decoder is then replaced with a shallow classifier.

Deng *et al.* [56] applied this method with a Vision Transformer to predict seizures from EEG. The uncertainty in the embedding should then capture the uncertainty that is in the EEG recording. Since the uncertainty is modelled in a deep embedding it may represent more complicated uncertainty in the EEG signal. This goes beyond simple sensor noise, but may instead capture uncertainty from various artifacts with more complicated patterns.

Although Deng *et al.* [56] do not give a thorough evaluation of the uncertainty, they do show that the modelling of uncertainty improves the classifier as compared to a deterministic equivalent, with minimal additional computational cost. Next to a Gaussian embedding, they also explore a Laplacian embedding. The Laplacian embedding was still better than a deterministic model, but it performed worse than the Gaussian embedding model.

10) Miscellaneous methods: Two more uncertainty quantification methods we encountered, but they were sufficiently rare that they do not fit into the presented narrative. The first of these is Adaptive Stochastic Gradient Hamiltonian Monte Carlo, which Chetkin *et al.* [57] uses for Motor Imagery classification. This Bayesian Neural Network method assumes a parameterized distribution over each weight, but uses a Markov Chain to converge to the posterior distribution. They found that this worked better than an ensemble when applied to ShallowConvNet [58], but there was no statistically significant different when applied to EEGNet [59].

To deal with the large amount of data in the Temple University Hospital Seizure Corpus (TUSZ) [60] dataset, De Rooij *et al.* [61] used Kalman Filters to solve the least squares adaption of SVMs. Rather than optimizing the SVM for epilepsy classification against the whole dataset at once, they consider parts of the dataset to continually learn the parameters of the SVM. Since Kalman Filters allow for some uncertainty, this method should capture model uncertainty. However, the authors do not go into detail on how well the uncertainty quantification performs.

III. Uncertainty Measures for Distributions of Probabilities

Some of the uncertainty quantification methods when applied in classifications tasks produce a distribution over class probabilities. There are several different ways in which these distributions over class probabilities may be used to get a scalar measure of uncertainty. An overview of such methods is given in Table II. We refer to these quantifications as *Uncertainty Measures*.

We found that throughout the literature various authors define some measure of "uncertainty" and refer to that as the quantified uncertainty. In this section we review these different methods explaining their intuition and relation to aleatoric and epistemic uncertainty. For a comparison on how some of these methods compare in practice we refer the reader to Milanes-Hermosilla *et al.* [62] and Wabina *et al.* [46].

Some of these measures are affected by the agreement between different forward passes (e.g. variance), while others may be affected by the (un)certainty of the probabilities being predicted. The UQ methods that are able to capture epistemic uncertainty (e.g. BNNs) will present this in the variance between forward passes. For this reason, we will refer to the uncertainty over class probability as the epistemic uncertainty and the predicted probabilities as the aleatoric uncertainty, even though some methods will instead present aleatoric uncertainty in both (see Sections II-C, II-D and II-G8). It is therefore important to consider that in order to capture epistemic uncertainty (and thus have a good uncertainty measure for generalization error), both a UQ method able to capture epistemic uncertainty, and an uncertainty measure that measures this is required.

A set of Simplex plots in Figure 7 shows how aleatoric and epistemic uncertainty interact. These plots are generated by taking 3 Gaussian distributions to represent predicted logits. 100.000 samples are taken from these logits and passed through the Softmax function. The closeness to each vertex represent the predicted class probability. This provides an intuition of how aleatoric and epistemic uncertainty may present as predicted class probabilities. It becomes apparent that under high epistemic uncertainty, determining aleatoric uncertainty becomes difficult.

Below we provide more details to the various Uncertainty Measures presented in Table II, and provide more details on their properties.

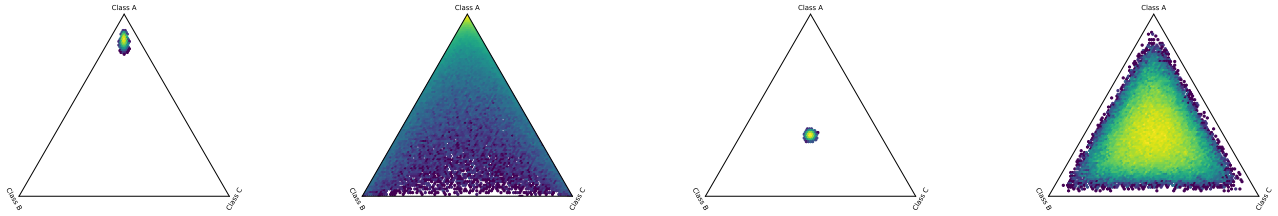
A. Class Probability

The standard method for measuring uncertainty in Neural Networks is the predicted Softmax probability of a classification. A (balanced) epilepsy classifier that gives the diagnosis of epilepsy with $p = 0.55$ is less certain than if it gives the diagnosis with $p = 0.97$.

This uncertainty measure typically captures aleatoric uncertainty. This uncertainty can either arise when the training data has annotations that reflect uncertainty, or when some samples in the training data have similar features, but different labels. Softmax probabilities are infamously overconfident in single-point neural networks, even when using a proper scoring loss function [36].

When multiple forward passes are made under a BNN the class probability is determined by the average of all forward passes. With T as the number of forward passes and $\bar{c}$ as the max probability class of the average probabilities ($\bar{c} = \arg\max_c T^{-1} \sum_t p_c$) we define the class probability as:

$$\mathbb{P}(p) \equiv T^{-1} \sum_T p_{\bar{c}} \tag{15}$$



(a) Low Aleatoric & Low Epistemic (b) Low Aleatoric & High Epistemic (c) High Aleatoric & Low Epistemic (d) High Aleatoric & High Epistemic

Fig. 7. Simplexes presenting different types of uncertainty. Epistemic uncertainty is presented by increased variance in the logits. Aleatoric uncertainty is presented by decreasing the difference between the means of the logits between classes. The points represent softmax probabilities determined by logits following a multivariate Gaussian $\mathcal{N}(\mu, \sigma^2)$. For high aleatoric uncertainty we set $\mu = [10, 10, 10]$, whereas for low we use $\mu = [10, 8, 8]$. For high epistemic uncertainty we set $\sigma^2 = [2, 2, 2]$, whereas for low we use $\sigma^2 = [0.01, 0.01, 0.01]$.

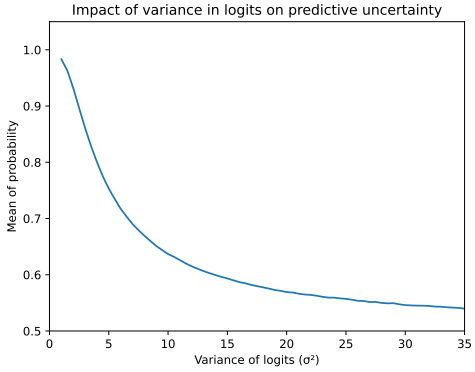


Fig. 8. Mean class probability decreases for higher variance in the logits. This illustration assumes binary classification. The logits are distributed as $\mathcal{N}([4, 0], [\sigma^2, \sigma^2])$. Only the first class is shown.

Or in a shorthand:

$$\mathbb{P}(p) \equiv \bar{p}_{\bar{c}} \tag{16}$$

Depending on the number of classes in a classification task, the range of this measure varies as $p \in [C^{-1}, 1]$, although some works choose to normalize this into $p \in [0, 1]$ [14]. Since high values are actually more certain here, it is more intuitive to consider the class probability as a "confidence" measure instead. To consider it in terms of uncertainty the negative is also sometimes used [14], [32].

a) The effect of variance on a Softmax probability: Most Uncertainty Quantification methods produce a distribution over probabilities, instead of a singular probability. In these cases, the mean of the predicted probabilities could then be considered as the most likely probability [63].

For approximations of Bayesian Neural Networks we can assume that the logits increase in variance as the epistemic uncertainty increases. The Softmax function (or a sigmoidal activation function) pushes high logits down into a $[0, 1]$ range, while lower logits are shifted less. As such, logits from a distribution with high variance will result in less confident probabilites. Figure 8 visualizes this effect. This means that the averaged class probabilities may actually also be affected by epistemic uncertainty. This provides a possible explanation of how mean class probability of a BNN can respond to epistemic uncertainty, and how it may prevent overconfidence.

Several of the reviewed works do discuss the difference between aleatoric and epistemic uncertainty, but they typically do not explore this interaction. Some works compare Bayesian Neural Networks to single-point networks [64], [65]. They find that the uncertainty becomes more precise (reduced Expected Calibration Error) by using MC-Dropout, but this may be because single-point classifiers are typically overconfident while BNNs will reduce confidence.

B. Variance

Various methods consider the variance or standard deviations of the posterior distributions as a measure of uncertainty [19], [37], [64], [66]–[68]. This works under the idea that epistemic uncertainty will lead to more disagreement between model samples.

Unfortunately, under multi-class classification it can be unclear which variance should be computed. Some implementations measure the variance over each class and either present all those variances to clinicians [37] or as features to another Machine Learning model [66]. One implementation seems to take the variance over the concatenated array of all class predictions [27]. This variance however then also goes down when the central tendency of the different classes becomes the same (as is expected under aleatoric uncertainty). While a good uncertainty measure may respond either to aleatoric uncertainty, epistemic uncertainty or both, it should not respond inversely to the different kinds of uncertainty.

Fiorillo *et al.* [64] presents the mean and variance for each class as a measure of how certain the model is for distinguishing the different classes. Through this they show that their model for sleep stage classification is more uncertain in predicting the sleep stage N1 (which it also performs the worse at), while it is more certain with the predictions for wakefulness. It is still unclear how the uncertainties interact within a single classification. When only a single measure of uncertainty can be used it is unclear whether that should be the variance of the predicted class, or if should it be the average variance over all classes.

To be specific, this leaves two possible measures for probability variance under multi-class predictions[6]:

[6]Note that this also applies to binary classifications that use two output nodes

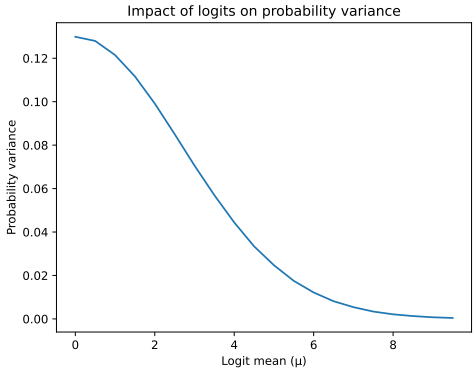


Fig. 9. Probability variance decreases as the logit mean increases. This illustration was generated by taking logits as $\mathcal{N}([\mu, 0], [2, 2])$. Here we see that probability variance (used as a measure of epistemic uncertainty) becomes smaller when aleatoric uncertainty decreases.

$$\mathbb{V}_{\bar{c}}(p) = T^{-1} \sum_{t} (p_{\bar{c}t} - \bar{p}_{\bar{c}})^2 \tag{17}$$

$$\mathbb{V}(p) = C^{-1} \sum_{c} T^{-1} \sum_{t} (p_{ct} - \bar{p}_{c})^2 \tag{18}$$

A similar effect as described in Section III-A0a occurs when applying variance uncertainty measures to the class probabilities. Figure 9 illustrates that the difference in the mean of the logits increases (less aleatoric uncertainty) the variance of the class probabilities decreases. As a result a decrease in aleatoric uncertainty can present as a perceived decrease in epistemic uncertainty. Future works should consider using the variance of the logits as described in [69] to get a more independent measure of epistemic uncertainty.

C. Predictive Entropy

Predictive entropy measures the total amount of uncertainty over the probabilities of all classes. This is also a method commonly used for single point Neural Networks. It is functionally equivalent to class probability for a binary classification task, but for more classes it also considers the amount of uncertainty remaining in the other classes.

Predictive Entropy[7] is given as:

$$\mathbb{H}_{\text{pred}}(p) = -\sum_{c} \bar{p}_c \log \bar{p}_c \tag{19}$$

When a UQ method that produces a distribution over predictions is used, the mean probability for each class is used (with similar consequences as discussed in III-A0a). Other variations include normalizing the entropy by dividing it by $\log(C)$ or taking $1 - \mathbb{H}_{\text{pred}}$ to get a confidence measure instead of an uncertainty measure [70].

As an example of how predictive entropy as an uncertainty measure can be functionally different from class probability consider the (mean) probability vectors $P_a = [0.6, 0.2, 0.2]$ and $P_b = [0.6, 0.35, 0.05]$. Both will have the class probability for the first class as 0.6, while their respective predictive entropies are 0.41 and 0.36. The predictive entropy (uncertainty) is lower in the second case, because it is more certain about the alternative classes. It is important to consider this effect when deciding between an uncertainty measure based on predictive entropy or on class probability.

[7]While the current work strictly defined this a predictive entropy, some works refer to this simply as entropy. Expected Entropy will sometimes also simply be referred to as entropy. In this work we consistently keep these distinct.

D. Disentangling Entropy

By capturing the total uncertainty, predictive entropy responds to both aleatoric and epistemic uncertainty. I.e. it is high when aleatoric uncertainty is high, or when epistemic uncertainty is high. It may be desirable to disentangle these methods.

The mutual information between a models parameters ω and a new labelled sample $\{x, y\}$ gives the amount of information gained by knowing that label of that sample, relative to what was already known by the models parameters. Since this may be considered equivalent to epistemic uncertainty [71] we get an intractable epistemic uncertainty measure:

$$I(\omega, y|D, x) = H[p(y|x, D)] - \mathbb{E}_{p(\omega|D)} H[p(y|x, \omega)] \tag{20}$$

This can be approximated by sampling from the posterior distribution:

$$\mathbb{I}(p) \approx \mathbb{H}_{\text{pred}}(p) + T^{-1} \sum_{t} \sum_{c} p_{ct} \log p_{ct} \tag{21}$$

These terms can be reordered as shown by Mukhoti *et al.* [13] into:

$$\underbrace{\mathbb{H}_{\text{pred}}(p)}_{\text{total}} \approx \underbrace{\mathbb{I}(p)}_{\text{epistemic}} \underbrace{- T^{-1} \sum_{t} \sum_{c} p_{\log p_{ct}}}_{\text{aleatoric if ID}} \tag{22}$$

The latter part may be considered an approximation of the Expected Entropy, which we refer to shorthandedly as the Expected Entropy, just as we refer to the approximation of the Mutual Information between the model parameters and a new sample as simply "Mutual Information".

Mukhoti *et al.* [13] brings forward an interesting observation about how Expected Entropy reflects aleatoric uncertainty. Since aleatoric uncertainty is learned on the training data, it is only well defined on data that is similar to the training data. As such, when samples are out of distribution and the epistemic uncertainty is high, Expected Entropy may be either high or low.

Wabina *et al.* [46] used Mutual Information as a method to do Bayesian Active Learning for EEG source localization. This method selected MRI training samples for a neural network based on which items the model was most uncertain about. One might expect that optimizing information gain would be ideal for this, but the authors found that it was actually outperformed by predictive entropy. It is unclear why this happens. Mukhoti *et al.* [13] suggests that when datasets have minimal aleatoric uncertainty in the training data, the predictive entropy may be dominated by the epistemic

TABLE II
AN OVERVIEW OF DIFFERENT UNCERTAINTY MEASURES THAT CAPTURE PREDICTIVE UNCERTAINTY/CONFIDENCE FROM A DISTRIBUTION OVER PROBABILITIES.

Name	Formula	Intuition	Ale UQ	Epi UQ
Class Probability [63]	$\mathbb{P}(p) = \bar{p}_{\bar{c}}$	Mean probability of predicted class	✓	✓
Predictive Entropy[62]	$\mathbb{H}_{\text{pred}}(p) = -\sum_c \bar{p}_c \log \bar{p}_c$	Uncertainty in mean prediction	✓	✓
Probability Variance [67]	$\mathbb{V}_{\bar{c}}(p) = T^{-1}\sum_t (p_{\bar{c}t} - \bar{p}_{\bar{c}})^2$	Variance of the predicted probability		✓
Expected Entropy [18], [71]	$\mathbb{H}_{\mathbb{E}}(p) = -T^{-1}\sum_t \sum_c p_{ct} \log p_{ct}$	Average uncertainty for each prediction	✓	
Mutual Information [62]	$\mathbb{I}(p) \approx \mathbb{H}_{\text{pred}}(p) - \mathbb{H}_E(p)$	Information gain from new sample		✓
Margin of Confidence [62]	$\mathbb{M}(p) = T^{-1}\sum_t p_{\bar{c}t} - \max_{c' \neq c} p_{c't}$	Average distance to second class	✓	?

We consider some number of forward passes $t \in T$. We denote some number of classes $c \in C$. A given probability for a class c on pass t is then p_{ct}. The average probability of a class c over all passes T is denoted $\bar{p_c}$. To denote the highest probability class after averaging over T we use $\bar{c}$. Lastly, $f_{\bar{c}}$ is the number of passes in T where $p_{\bar{c}t} = \max_c p_{ct}$.
The arrow behind each measure indicate the value where a measure is most certain.
Vacuity and Dissonance fall outside of this scope due to their different nature.

uncertainty. If this is happening, then the approximative nature of this disentanglement could explain why predicitive entropy outperformed mutual information.

1) Explaining Ad-Hoc Methods by Entropy: Disentangling predictive entropy into mutual information and expected entropy gives a well formalized understanding of how a distribution over probabilities may be deconstructed into separate aleatoric and epistemic uncertainty.

Epistemic uncertainty has been modelled by the variance or standard deviation of the predicted probability in 27.3% of the reviewed papers, but without evidence of why this reflects epistemic uncertainty. The intuition available is that disagreement between sampled models should be epistemic uncertainty.

Smith *et al.* [71] looks at the Taylor series of the Mutual Information and finds a similarity to probability variance. The first term of the Taylor expansion is equivalent to probability variance, up to a multiplicative constant. This shows that probability variance and mutual information have some functional similarities, and shows that probability variance therefore must also capture epistemic uncertainty.

To the best of the author's knowledge, there is no similar proof known to connect the Expected Entropy to a similar statistical measure of central tendency. Since the predictive entropy build on the mean prediction, it is unlikely that the mean would reflect aleatoric uncertainty, but rather a mixed uncertainty. Other measures of central tendency, such as the mode or the median, may reflect aleatoric uncertainty and have functional similarities to Expected Entropy. Further exploring this will allow a more intuitive measure to be constructed that can correspond to aleatoric uncertainty.

E. Margin of Confidence

Last but perhaps not least, Milanes-Hermosilla *et al.* [62] proposes Margin of Confidence as an intuitive uncertainty measurement. This rather ad-hoc measure looks at the average distance between the probability of the predicted class and the class with the next highest probability. Note that while the predicted class is taken over the average from the forward passes $\bar{c} = \arg\max_{c \in C} \bar{p}_c$, the second-highest is chosen on each sample. This means that in some forward passes, the second-highest probability $\max_{c' \neq c} p_{c't}$ is actually higher than the probability of the predicted class $p_{\bar{c}t}$.

In its full form the Margin of Confidence is given as:

$$\mathbb{M}(p) = T^{-1}\sum_t p_{\bar{c}t} - \max_{c' \neq c} p_{c't} \tag{23}$$

Milanes-Hermosilla *et al.* [62] used the Margin of Confidence to separate correctly and incorrectly classified predictions. They found that the Margin of Confidence had a greater Bhattacharyya distance between the correctly and incorrectly classified predictions than Mutual Information, Predictive Entropy and Probability Variance (following Equation 18).

IV. UNCERTAINTY USE CASES

The study of Uncertainty Quantification methods in Biosignals focuses on implementing these methods to solve some kind of problem, or to gain knowledge about the Biosignal or the underlying biological process. This separates the research on Uncertainty Quantification in Biosignals from other domains. While the Machine Learning methods may transfer well between Computer Vision research and Biosignal research, the actual problems being solves are very different. Therefore, the ways uncertainty are used may also be different.

Below we outline several common purposes that uncertainty quantification has been used for. We specifically look at what behaviour is expected of the uncertainty for different applications. We will see that different tasks have different requirements for how uncertainty should behave.

A. Rejection Methods

The most common use for estimating uncertainty is to be able to not make a prediction when the likelihood of that prediction being wrong is too high. 37% of papers in this review use a measured uncertainty to reject samples from the testing data.

Both aleatoric and epistemic uncertainty can contribute to a risk of predictions being wrong. A good estimation that considers both would be optimal. However, under epistemic uncertain the models prediction of aleatoric uncertainty may

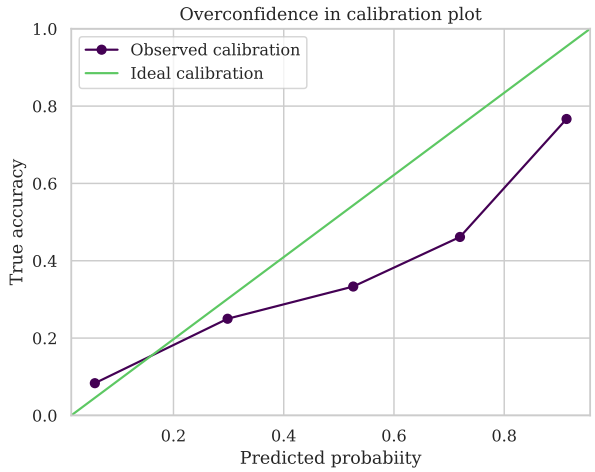


Fig. 10. Example calibration plot for an EEG Motor Imagery classifier. The predicted probability is consistently higher than the true probability of being correct. This means the model is overconfident. Note that a plot like this is only reliable with sufficient samples with different predicted probabilities.

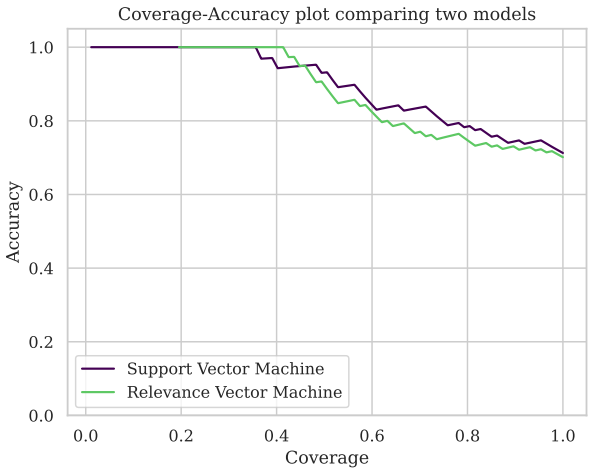


Fig. 11. Example plot showing the tradeoff between coverage and accuracy for two EEG Motor Imagery classifiers. The plot shows that both models have very similar accuracy without rejection (coverage at 1.0), but that the Support Vector Machine has a better accuracy-coverage trade-off.

also be wrong [13]. When the to-be-rejected samples are dissimilar to the training data, epistemic uncertainty measures may be preferred.

Provided that the uncertainty measure has any predictive power of the accuracy of a prediction, a rejection method will always improve accuracy. The cost of this is a reduction in the amount of samples the system makes a prediction for. The ratio of samples for which a prediction is made can be referred to as *coverage*[19].

An ideal uncertainty measure simultaneously maximizes coverage and predictive accuracy. It may do so by producing higher values when the probability of a classification being wrong is high, and lower values when that probability is low. Since the goal is to be able to separate those two groups by a decision threshold, the uncertainty does not need to match the probability of a classification being incorrect, nor is it constraint to expectations of being linearly related to the probability of the class being accurate.

Expected Calibration Error (ECE) is a common method for evaluating uncertainty quantification which measures the difference between a predicted probability for a classification and the actual observed probability on a validation set [72]. This is often visualised with a calibration plot as shown in Figure 10. While minimizing the Expected Calibration Error does satisfy the needs of making a good decision boundary, a consistently over or under confident uncertainty quantification can make an equally good decision boundary. As a result, a consistently overconfident model with good separability of correct and incorrect predictions still gets a bad ECE, while an appropriately confident model with poor separability of correct and incorrect predictions gets a better ECE. Moreover, a model which is always predicts $p(y) = 0.5$ is not a usable model, but will have a minimal ECE. Therefore, ECE should be used cautiously as a way to evaluate uncertainty calibration, but not of uncertainty quantification as a whole.

TABLE III
VARIOUS UNCERTAINTY USE CASES GROUPED IN THEIR REQUIRED TYPE OF UNCERTAINTY. IN GENERAL, METHODS THAT NEED EITHER ALEATORIC OR EPISTEMIC UNCERTAINTY MAY STILL DO WELL WITH A MIXTURE OF BOTH. REJECTION IS SPLIT INTO REJECTION WHEN THE DATA IS IN-DISTRIBUTION (ID), OR OUT-OF-DISTRIBUTION (OOD), RELATIVE TO THE TRAINING DATA.

Aleatoric	Epistemic	Both
Feature Rejection (ID)	Active Learning Model Pruning Data Augmentation Rejection (OOD)	Interpretability Social Bias Soft Voting

1) Uncertainty as a Classification Task: A common technique used to evaluate uncertainty quantification for rejection methods is setting a threshold against uncertainty and observing an increase in accuracy and a decrease in coverage [14], [19], [32], [44], [64], [70]. This framework considers uncertainty as a tool to improve classification performance, instead of having uncertainty as an inherent goal. While some works set a single threshold against uncertainty [32], [44], [64] a range of thresholds is preferred [14], [19], [70], as the right balance between coverage and accuracy is typically not well established. Moreover, when different uncertainty quantification methods are compared against a threshold they might decrease in coverage while increasing accuracy, which makes them harder to compare. Instead, coverage-accuracy plots as visualised in Figure 11 may be used to assess the reject-performance of a model. By going over all possible thresholds, this plot shows the options for balancing coverage and accuracy, which may be used for comparing models.

The alternative framework is to consider uncertainty as a classification task, where the goal is to classify whether a prediction will be correct or incorrect [14], [31], [62], [65], [73], [74]. This results in the common classification metrics such as uncertainty accuracy, uncertainty precision and uncertainty recall being used in this domain, although other

common classification metrics such as the ROC-curve should be considered [75]. Unlike the accuracy/coverage framework, this perspective keeps the quality of uncertainty and the quality of predictions as orthogonal.

Two disadvantages of having uncertainty as a classification task can be observed. First is that it does not show the trade-off between accuracy and coverage. When the focus is on implementing a model that complies with certain needs this might be what one wants to evaluate the model on. Instead it gives more insight into the performance of individual components, which could be more helpful for knowing what to optimize. Lin *et al.* [14] solves this dilemma by presenting both frameworks. The second disadvantage is that the precision of uncertainty can never approach 100%. In the ideal case it separates the predictions that are certainly correct from the predictions that are randomly guessed. However, it is evaluated on separating the correct from the incorrect predictions. As a result, the upper limit of uncertainty precision is lower than 100%. Jahmunah *et al.* [31] performed 10-class myocardial infarction classification with ECG data. As artificially added noise increased, the uncertainty accuracy flatted out at 88.5%. Considering the aforementioned, this may be closer to optimal uncertainty quantification than the uncertainty accuracy suggests.

When uncertainty is considered as a classification task it may be worth considering normalizing the accuracy to control for this upper limit.

2) Choice of Uncertainty Measure: Fiorillo *et al.* [64] explored the choice of using class probability or probability variance as an uncertainty measure. They found the accuracy improved most under class probability. Similarly, 5 out of 9 other papers using uncertainty as a rejection method used measures that observe a mixture of aleatoric and epistemic uncertainty, while only 1 used an epistemic method, and 1 an aleatoric method. The last one used Margin of Confidence, for which is it unknown whether it responds to aleatoric uncertainty, epistemic uncertainty, or both. Overall it can be concluded that most works use mixed uncertainties for rejection methods.

While it is clear that mixed uncertainty measures are commonly preferred, it may still be worth considering separated uncertainty measures. By using a specifically aleatoric and a specifically epistemic measure, one may be able to set different rejection thresholds against different kinds of uncertainty. This makes evaluation more tricky, but it can improve usability in situ. When reducing aleatoric uncertainty by collecting another recording is very expensive, but reducing epistemic uncertainty by getting a second opinion from an expert or another model is cheap, it is desirable to make different decisions for the different causes of uncertainty.

3) The Rejected Samples: In rejection methods it is worth contemplating what happens to the samples that are rejected. In medical applications a common expectation may be to have a clinician re-asses the data, but this would only be valuable under epistemic uncertainty. Under aleatoric uncertainty a re-recording of the electrodes would be needed instead [27], or additional information about the patient. Implementations where predictions are made and used in real-time require a well-considered behaviour for rejected cases. In the current literature this consideration seems to be missing or limited.

B. Uncertainty for Interpretability

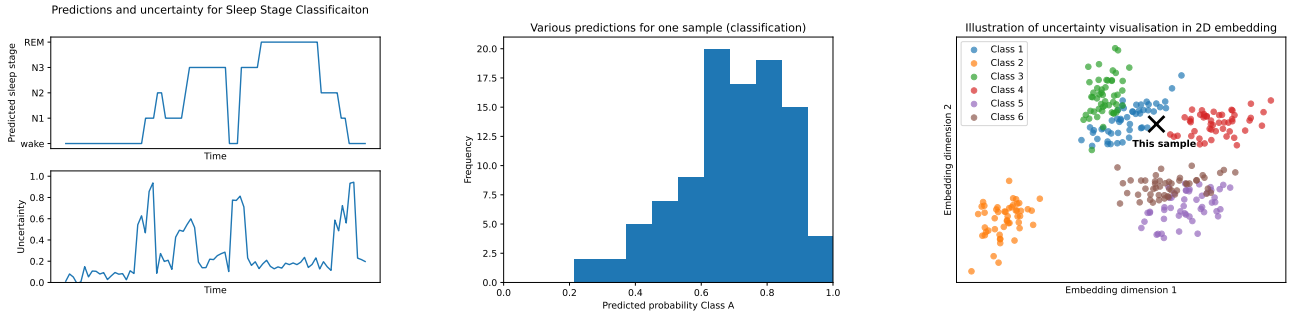
Uncertainty is sometimes proposed as a method to alleviate a part of the black-box problem of Neural Networks [70]. Standard Neural Networks only give a single point prediction, but do not show any insights about how it got to this prediction, or how likely it is that a prediction is correct. This is a legitimate objection to the adoption of Deep Learning systems in medical applications, but methods that detect uncertainty may alleviate this [37]. By presenting aleatoric uncertainty a model is able to show that a given prediction may not be correct. Epistemic uncertainty can show this specifically when the data to be classified is dissimilar to the data a model is trained on, for example due to changes in recording equipment or protocol.

Determining what good communication of a quantified uncertainty is can be difficult. Some works use uncertainty to get an estimate of the scientific validity of outcomes from a Neural Network [35], [42]. In this case, interpretation is not very time-sensitive, and the purpose of interpretation may vary widely. Research on scientific visualization of uncertainty is available [76], [77], but is not interweaved with the reviewed literature and does not demonstrate how to present aleatoric and epistemic uncertainty.

Clinical interpretation of uncertainty can be even more critical, as it may affect the quality of a diagnosis or the adoptability of Machine Learning methods. For some ECG applications time-sensitivity is given as a factor affecting manual diagnosis [31], so the interpretation of an uncertain prediction may be subject to time constraints in such cases.

In standard classification tasks an accepted way of presenting a quantified uncertainty is by reporting an accurate class probability. A predicted class probability that accurately corresponds to the true probability of a class (even under epistemic uncertainty) can be mathematically interpreted and gives a well-defined and well understood measure of uncertainty. Expected Calibration Error has been used to capture this goal in a metric [18], [48], [63]. However, people may prefer non-numerical representations of probability [78]. These methods will aggregate a range of probabilities under one label. This means that small deviations on each sample have no effect, but a few samples with large difference may be a problem. In general, there is a need for more research on how people interpret measures of epistemic and aleatoric uncertainty from UQ methods, so that future research can focus on optimizing the appropriate interpretations, rather than appropriate numerical outcomes.

1) Visualizations of Uncertainty: Within the interpretability domain there are opportunities to create task specific visualizations of uncertainty to improve interpretability. Three different generalised approaches are shown in Figure 12. This makes a generalised observation that these visualisations may deliver uncertainty and prediction separately, together, or may even leave both to be decided by the user. In practice, the specific task for the visualisation impacts the way it should be implemented. Below we discuss how visualisations have been done in the reviewed literature.



(a) Explicit prediction and uncertainty (b) Explicit uncertainty and implicit prediction (c) Implicit prediction and uncertainty

Fig. 12. Three different general approaches to visualising uncertainty. The first plot specifically shows a separated prediction and a degree of uncertainty. The second plot explicitly shows the uncertainty, but leaves the predicted probability to be determined by the user. The third plot shows an embedding of the sample, but does not explicitly give a prediction nor an uncertainty.

Bekhti *et al.* [42] proposes a Markov Chain Monte Carlo approach for Majority Minimization to solve the inverse problem. The MCMC sampling results in multiple sparse solutions, where the agreement between solutions is interpreted as uncertainty. By presenting a heatmap of the source localization solutions on 3D brain renderings they allow the reader to interpret the level of uncertainty based on the relative density and the total spread of solutions. This also allows readers to involve their prior knowledge about neuroanatomy implicitly by contrasting the certainty of the predictions against prior knowledge.

As an alternative, Gill *et al.* [79] uses a CNN with MC-Dropout to classify lesional voxels in patients with focal cortical dysplasia. The results are presented by a map of class probability voxels (predictive uncertainty) and a separate map of probability variance voxels (epistemic uncertainty). It is then up to the user to combine these two sources of information.

Phan *et al.* [70] shows a method to support EEG-based sleep classification. They show a timeseries of the predictive entropy, the stacked class probabilities (also predictive uncertainty) and the classifications above each other. To improve readability they highlight the parts where confidence drops below a given threshold. This is used to show how uncertainty is highest during stage transitions.

A more generalizable method is given by Costabal *et al.* [35], who present a histogram of the whole distribution of class probabilities. This allows readers to intuitively asses central tendencies, spread and skew.

The design of a visualization is very task dependent, and may have an impact on the effectiveness of quantified uncertainty. More research in how these visualization can be optimized, especially in (emergency) medicine is needed and may guide how uncertainty should be quantified. It is currently unknown how aleatoric and epistemic uncertainty may be differently interpreted, or if a vacuity-dissonance framework is more interpretable.

One interesting approach to dealing with this is suggested by Van De Leur *et al.* [28], where a VAE embedding of an ECG is reduced to 2 dimensions using Principal Component Analysis. A user is presented with the embeddings of known diagnoses. This allows the user to determine a measure of uncertainty based on a more fluid notion of vacuity, dissonance, aleatoric or epistemic uncertainty. By not trying to quantify uncertainty, but instead allowing the user to assess uncertainty, they aim to make a diagnosis more interpretable.

C. Uncertainty as a Feature

Another interesting, but rather task-dependent, use of uncertainty is as a feature for subsequent Machine Learning tasks. For example, Stoean *et al.* [66] attempts to detect presymptomatic spinocereballar ataxia type 2 using electrooculography. They observe the saccadic eye movements in healthy, sick, and presymptomatic participants. Healthy participants show a sudden eye movement with nearly instant acceleration and deceleration. Sick participants can show more chaotic movement with slower acceleration and speed. Presymptomatic participants can show a decrease in control, speed and rate of acceleration. Since there is a lot of variation between participants and each saccade, 85 saccades are recorded for each participant, and classified with an ensemble of Deep Neural Networks using MC-Dropout. The 3 class probabilities and the 3 class standard deviations for all 85 saccades were used for a decision tree classifier. The system was able to classify sick and healthy participant quite well, and performed acceptably at classifying presymptomatic participants.

When uncertainty is used as a feature for another Machine Learning model the constraints of what a good uncertainty is are loosened. The main difference is that the uncertainty no-longer needs to be a single value, but can instead be given by multiple uncertainty measures. As such, the choice of using a epistemic and a aleatoric uncertainty measure can provide more information. Depending on the choice for a subsequent Machine Learning model, even the requirement of the measure being monotonically in/decreasing with uncertainty may be lifted.

D. Uncertainty to Control Social Bias

As fairness and negative social biases are a growing concern in Machine Learning, Zanna *et al.* [67] present a rather unique usecase for uncertainty quantification. They propose a Multi-Task Learning method using Uncertainty Quantification to

reduce social bias while classifying periods of anxiety from ECG features. The bias mitigation strategy uses a multi-task branch that attempts to classify whether the samples belongs to a person from an unprivileged demographic group.

The model is trained for 100 epochs, with the weights being saved every 5 epochs. After training, the model with the highest average epistemic uncertainty (probability variance) on the demographic-classification and the lowest average uncertainty on the anxiety-classification is selected. The model performing poorly at demographic classification should not have features in the latent representation to capture demographic classification. The authors showed that this minimized bias, but this did come at a loss in model performance.

While this method is still somewhat ad-hoc, it paves the way for future methods in minimizing social bias through uncertainty quantification. Future research may focus on forms of adversarial training, so that an anxiety model will try to optimize the anxiety classification while under an ongoing constraint of having no features that may be used to infer the demographic class. The different effects of aleatoric and epistemic uncertainty are also worth exploring here.

E. Bayesian Active Learning

The last usage for uncertainty quantification encountered in the review is in Bayesian Active Learning. Under Active Learning training samples are iteratively selected by the epistemic uncertainty that the model has about that sample [80]. These methods are proposed for situations where insufficient labelled training data is available, and manual labelling of data is expensive. To reduce the labelling cost, Active Learning starts with a model trained on very little data, and observes the uncertainty it has on the unlabelled data. The most uncertain samples are then manually labelled by an *Oracle*: a system that produces the ground truth labels. This Oracle can (in the domain of Biosignals) be the expert annotations, but may also be additional testing to establish a better ground truth such as an MRI scan[8].

Gal *et al.* [80] produces a proof-of-concept of Bayesian Active Learning with the MNIST dataset of handwritten digit classification. They use MC-Dropout as the uncertainty quantification methods, and then compare different uncertainty measures as criteria to select training samples with (this is called the Acquisition Function).

They found that variation ratio (based on the rate of times predicted class is most-likely) performed best, followed by Mutual Information[9] and predictive entropy. When repeating the setup on skin cancer classification they found that variation ratio was consistently the same value, and therefore not usable. Mutual Information and predictive entropy still performed well.

Wabina *et al.* [46] reproduced this comparison of acquisition functions for solving *the inverse problem*. However, while Gal *et al.* [80] evaluated the performance by the minimum number of training samples needed to achieve a given accuracy, Wabina *et al.* [46] evaluated on the quality of the model predictions after training with all data but limiting the learning by early stopping. They find predictive entropy to perform the best, but they do not focus on reducing the number of samples.

[8]This is only valuable if the experts have aleatoric uncertainty about the Biosignal recording

[9]In Active Learning Mutual Information is referred to as Bayesian Active Learning by Disagreement (BALd)

F. Miscellaneous use cases for uncertainty

Two works propose novel ways to use uncertainty for Brain Computer Interfaces. As part of their UNCER model, Duan *et al.* [54] uses uncertainty to assess the quality of data augmentation. They consider data augmentation as a method to reduce uncertainty to unseen corruptions.

For a P300 speller Ma *et al.* [81] look at model uncertainty, not only in terms of how it affects predictive uncertainty, but also in what it says about the model. They argue that weights with a poor signal-to-noise ratio are redundant. With this method they were able to prune 75% of the weights without decreasing the F1 score. In the single-point model any amount of pruning would result in a (slight) decrease in F1 score.

Additionally, Ma *et al.* [81] used the predicted probability for a special soft-voting strategy. In P300 spellers each letter is flashed several times, and a classifier tries to identify a P300 wave. By using the probability of a P300 wave their Bayesian CNN outperformed an equivalent single-point model. This strategy of voting with probabilities, rather than with discretised predictions is similar to Soft Voting in Machine Learning ensembles.

V. Guideline for Adding Uncertainty Quantification

The review covered various methods for obtaining quantified uncertainties and presented methods which people have been using uncertainty for. Based on these findings, we aim to conclude a guideline on how to implement uncertainty quantification for a Machine Learning task on Biosignal data. Unfortunately, there is no singular solution or decision tree that works best for all cases. Nonetheless, we provide an outline below of decisions to make for researchers using a Machine Learning system for a Biosignal that are interested in using Uncertainty Quantification. The descriptive study of UQ for Biosignals in the previous section cannot result in a complete guideline. Instead, the remainder of this section gives prescriptive advice. These instructions should be taken with a critical eye and may be subject to disagreement. Still, it is intended to provide a starting point from which better methodologies may be constructed.

We start with considering the cost of adding Uncertainty Quantification to a Machine Learning task. After this the first step will cover the uncertainty quantification methods, which is mostly guided by your choice of Machine Learning model and computational constraints. Second is the choice of uncertainty measure, which is chosen on the constraints of the uncertainty usecase, and whether the task concerns aleatoric or epistemic uncertainty. The last step is the evaluation. Depending on the uncertainty usecase, different evaluation methods align best with the specific goal. Lastly, we discuss some sanity checks to validate that the uncertainty quantification works as intended.

A. Cost of Uncertainty

Knowing when your models predictions are likely to be wrong, and a hint of why they might be wrong, can be quite valuable. However, there is always a price to pay.

For MC-Dropout and Ensembles this price is computational cost. MC-Dropout requires many forward passes, so the cost of inference might increase 100 times. Ensembles require training several models, which means training cost may increase 5 times. At inference, this also requires having enough memory for 5 models.

However, these methods do not result in a decrease in model accuracy. MC-Dropout converges to the prediction a single-point model would have made after 100 forward passes [69], and Ensembles have been researched extensively with the aim of improving model accuracy [82].

Methods that optimise a model for uncertainty (such as Variational Inference, Prior Networks, Evidential Machine Learning and Variational Autoencoders) are at risk of decreased model accuracy. Since the model is now optimised towards two tasks simultaneously, this may have a negative effect on the predictive performance. However, this is not guaranteed as multi-task learning leverages a similar mechanism to improve predictive performance [83].

Post-hoc calibration does not directly have a substantial computational cost, nor does it directly affect the model predictions. However, doing post-hoc calibration requires data to do the calibration on, which generally cuts into the data available for training or testing.

B. Choice of Uncertainty Method

Without a goal to specifically explore novel uncertainty quantification methods, simply following the most commonly used methods can give a good starting point. Figure 13 shows that most of the Biosignal papers using Uncertainty Quantification use MC-Dropout. This is probably because it is easy to implement, and can be applied to any Neural Network using dropout without retraining.

A practical downside to using MC-Dropout is that it substantially increases the inference cost. 100 forward passes is a good starting point to capture the predictive distribution, but that increases inference cost up to 100 times. This may be acceptable when doing offline inference or when the model is small, but can be prohibitive when inference is done under computational constraints. Moreover, MC-Dropout is only an approximation of a Bayesian Neural Network. Depending on how it is applied this may be a rather poor approximation.

Ensembles are the next common method. The advantage relative to MC-Dropout is that it requires fewer forward passes to get a decent approximation of the predictive distribution. 5 models in an ensemble is a good starting point, increasing inference cost only 5 times, but also increasing model storage size 5 times, and increasing training cost 5 times.

For large Neural Networks either of these approaches may be prohibitively expensive. In that case direct Uncertainty Quantification methods such as Evidential Deep Learning or Prior Networks may be preferable, although it is not certain whether they capture epistemic uncertainty as well as BNNs do. Alternatively, Early Exit Ensembles may be considered as a computationally affordable solution to deep ensembles.

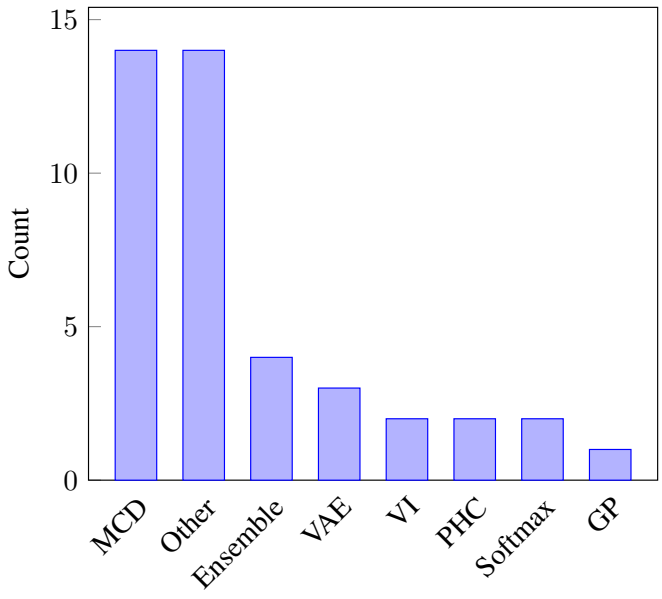


Fig. 13. Popularity of various Uncertainty Quantification methods in this review. Several papers presented multiple Uncertainty Quantification methods. MC-Dropout and Ensembles are the most common. A large group of Other consists of 14 different methods. MCD is MC-Dropout, VAE is Variational Autoencoder, VI is Variational Inference, PHC is Post-Hoc Calibration, and GP is a Gaussian Process.

When the Neural Networks are particularly small and training data is sparse one may opt for Variational Inference. Variational Inference can provide a better approximation of a Bayesian Neural Network than MC-Dropout, but it is prohibitively expensive and may suffer from numerical instability on Deep Neural Networks [84]. The impact of injecting domain knowledge through informative priors is not known. Note that while MC-dropout and Ensembles can be trained with regular loss functions used for single-point Neural Networks, most UQ methods require loss functions that consider the uncertainty such as the VI loss in Equation 6 or variance attenuated negative log-likelihood [29].

If the base-model of choice is not a Neural Network this review finds insufficient previous research to suggest a method of uncertainty quantification. Two studies use the entropy in a decision tree as a measure of uncertainty, but this does not capture epistemic uncertainty, is often quite coarse and may be dependent on the choice of hyperparameters of the training process. It is unclear whether the Kalman Filter SVM [61] result in good uncertainties. Instead we recommend considering Bayesian methods for standard Machine Learning models such as Bayesian Linear Discriminant Analysis and Relevance Vector Machines as explained by Prince [85].

C. Choice of Uncertainty Measure

For regression problems, the current review only covers two papers, which is insufficient to make data-driven recommendations on a good choice of uncertainty measure. Instead we recommend from our experience two measures of uncertainty

for regression: the variance of the prediction, or the 95% Confidence Interval. Measures of variance may be well suited for rejection systems, as they present a scalar uncertainty which can be thresholded against. Confidence Intervals may be preferable for human interpretation as it gives a notion of likely possible values.

For classification problems however, the choices for uncertainty measures may be discussed more extensively. For rejection methods the, expected entropy, or mutual information may be good options. Expected entropy and mutual information can be useful when only aleatoric or only epistemic uncertainty is to be considered, or when different behaviour is expected for each kind of uncertainty. Even when there is no interest in disentangling uncertainty, expected entropy or predictive entropy may perform poorly when there is a lot of epistemic uncertainty, as the predictions of aleatoric uncertainty are then made by an uncertain model.

However, the entropy based measures may be more difficult to interpret by people. If human interpretation is the goal instead consider using statistical measures to represent the predictive distribution. A well calibrated mean probability that accounts for epistemic uncertainty may be easy to interpret. A visualization of the probability variance (e.g. by color) can be used to present epistemic uncertainty. Alternatively, for completeness the entire distribution of class probabilities can be presented.

When the uncertainty measure is used for active learning it would intuitively make most sense to use a measure of epistemic uncertainty. However, Wabina *et al.* [46] found that predictive entropy, which captures both aleatoric and epistemic uncertainty, works best.

D. Evaluating Uncertainty Quantification

Whenever Uncertainty Quantification is considered as a tool to improve the outcome of a larger system, rather than as its own end-goal, the evaluation methods may need to be adjusted to the purpose for which uncertainty is used. Below take in each section a given uncertainty usecase, and discuss how to evaluate the uncertainty quantification for that usecase.

1) Rejection: If uncertainty is used in order to reject difficult samples, the impact of uncertainty on the larger system may be directly measured with a coverage-accuracy plot as in [14], [70]. These systems all depend on setting a threshold, which is usually arbitrary. Therefore, it is better to create a plot that shows the outcome for all possible thresholds by plotting the coverage against the accuracy. Showing the coverage and accuracy only for a single threshold makes it hard to compare models when the distribution of the uncertainty measure shifts.

However, these coverage-accuracy plots do not give direct insights into the Uncertainty Quantification performance per se. Gaining more insights into this may help improve the large system, rather than only evaluate it. For this, it may be worth casting the uncertainty as a classification task, so that regular classification metrics may be used. Be aware that this is typically an unbalanced task, where again the cost of false-positives and false-negatives is not well defined, so ROC curves may be a preferred approach. Since a perfect uncertainty measure is not able to provide perfect classification (as described in Section IV-A1), it may be worth adjusting the metrics to give a more directly interpretable evaluation of the uncertainty.

For both of these cases, it is worthwhile to use a good baseline to assess whether the Uncertainty Quantification method actually provides an improvement. Setting a threshold against a standard Neural Network with Softmax as uncertainty gives a fair baseline.

2) Interpretation: While the rejection usecase does not demand a well-calibrated measure of uncertainty, that may important for interpretation by a person. In this case the best approximation that can be given is that a predicted probability should align with the true probability. This can be measured by the Expected Calibration Error, which is therefore an acceptable metric for evaluating an uncertainty that needs to be directly interpreted.

However, giving too many significant figures of a probability may give a false sense of precision, so it is possible that similar probabilities can be put in larger bins, which may even be mapped to natural language. In that case, the Expected Calibration Error is not ideal, as many small errors can have a substantial contribution to this metric, but may not actually affect the presented uncertainties. Instead, Maximum Calibration Error may be used, as this would ignore the small calibration errors and only focus on the large differences.

To really get a thorough understanding of what works best for interpretability human evaluation and user studies are needed. Both for the general problem of using uncertainty quantifying ML models, as well as for specific user groups and specific tasks. For supporting interpretability in medical decision making user studies should focus on the specific medical discipline of the user.

3) Intermediary Features: When uncertainty is used as an intermediate, for example as a feature for a different model, or as an acquisition function for Active Learning, it can be hard to identify which properties are required for an optimal uncertainty measure.

Expected Calibration Error or the quality of uncertainty as a classifier may be used as a proxy for the quality of the uncertainty, but this is not specific to the usecase. Instead, the uncertainty method should be evaluated on the impact it has on the performance of the larger system.

For any case of using uncertainty, it may be good to perform some sanity checks to ensure the uncertainty is behaving as intended [86]. For systems that are expected to measure epistemic uncertainty, one may try to create out-of-distribution data, and validate whether the epistemic uncertainty increases. To observe the quality of aleatoric uncertainty, one may look at the samples in the training data that are classified with high aleatoric uncertainty, to assess whether they align with the intuitions for aleatoric uncertainty. Alternatively, aleatoric uncertainty may be evaluated with relevant and realistic induced noise in the training data.

VI. Open Challenges

We identify several open challenges to the Biosignal application domain for uncertainty quantification. Overall, while

uncertainty quantification has been gaining traction, there are still some obstacles for adoption and some under-explored areas. This paper attempted to address two of these obstactles already. By providing an outline of how to add Uncertainty Quantification to a biosignal classifier in Section V we invite more researchers to incorporate Uncertainty Quantification methods into their Neural Networks.

Many previous papers used ad-hoc measures of uncertainty which generally seemed to behave quite well. Section III provides an overview of these uncertainty measures, and explains how they correspond to aleatoric and epistemic uncertainty. This should encourage researchers to distinguish between aleatoric and epistemic uncertainty in their Uncertainty Quantification, in an attempt to encourage Machine Learning applications that are conscious of the distinction between aleatoric and epistemic uncertainty in the model environment.

Many issues and opportunities still exist that are not solved in the currently available literature. Below we outline some of the open challenges identified, and we specifically encourage research projects that study Uncertainty Quantification models in situ.

A. Interpretability of Uncertainty

This review found 13 papers where the quantified uncertainty was explicitly or implicitly intended to be interpreted by a person, but none of them connected the uncertainty to thorough studies of how different representations affect uncertainty. Gill *et al.* [79] - for example - makes a visualization distinguishing predictive and epistemic uncertainty in FCD lesions detection, but it is not known how well such a visualization helps a clinician with identifying the true lesions and the false positives.

Previous research about how well clinicians can interpret probabilistic tests exists [87], [88], but that is currently not tied to the way Uncertainty Quantification research is conducted. Research on what makes a well-interpretable (disentangled) uncertainty is needed.

B. Biosignals Are Not Images

Bayesian Neural Networks cover the majority of uncertainty quantification methods encountered in this review. These methods have been popularized in Computer Vision, where Deep Neural Networks are dominating the state-of-the-art.

While Deep Learning has been gaining popularity and generating good results on large datasets [89], its infamy for requiring large amounts of training data means many Biosignal models prefer shallower Machine Learning systems such as Support Vector Machines [90] and Linear Discriminant Analysis [91]. This review did not find uncertainty quantification for such models, although they do exist (see Prince [85]). More research implementing uncertainty quantification on shallow models is needed, preferably with the ability to disentangle aleatoric and epistemic uncertainty, but minimally with the ability to capture a mixture of aleatoric and epistemic uncertainty. Bayesian alternatives for both methods are available [11], but they are not yet used in the biosignal literature.

C. Appropriate Benchmarks for Uncertainty

Xia *et al.* [18] offers some benchmark data. They do this by introducing noise to existing biosignal datasets with the intention that uncertainty should go up as dataset shift makes the accuracy go down. While this is a good starting point, the type of introduced noise may not be reflective of real dataset shifts that may be observe when UQ models are implemented in practice. Instead, there is a need for datasets that realistically capture the aleatoric and epistemic uncertainty they may be encountered when biosignal models are deployed in practice.

Epistemic uncertainty presents most realistically in cross-subject generalizability, rare comorbidities, or unusual erroneous recordings. By tailoring a dataset with these sources of epistemic uncertainty, we can improve the construct validity of UQ research.

For aleatoric uncertainty we may make annotations in existing datasets that indicate ambiguity. For sleep stage classification this can be done by focusing on the inter operator variability. We may also annotate artifacts in training datasets that would corrupt the discernibly of classes.

D. Vacuity-Dissonance and Aleatoric-Epistemic

Two frameworks for understanding uncertainty were encountered. The most common is the distinction between aleatoric (data) and epistemic (knowledge) uncertainty. However, the vacuity (absence of class features) and dissonance (contradicting class features) distinction could provide a more directly interpretable disentangling of uncertainty. It is not clear how these frameworks interact, and clarifying this may provide a more complete understanding of the uncertainty a model encounters.

Future research may explore their interactions, their differences, and other interpretations of uncertainty that may be useful for biosignal classification tasks.

E. The Needs of Clinicians

Elul *et al.* [37] discusses the needs of clinicians in three concepts: estimating uncertainty, handling unknown classes, and detecting a failure to generalize.

Under the aleatoric-epistemic uncertainty framework, the *estimating uncertainty* corresponds to aleatoric uncertainty, while both out-of-distribution unknown and known classes fall under epistemic uncertainty. In order to better address the clinical concerns, each of these problems may be addressed uniquely. While the path towards this is not known, the unification of aleatoric-epistemic and vacuity-dissonance uncertainties may provide a starting point.

F. Using Uncertainty for Biosignal Applications

67.6% of papers reviewed use uncertainty either for presenting a confidence with a prediction, or for rejecting difficult samples. However, there is an unknown number of other possible things that uncertainty quantification may be used for that need exploring.

A promising purpose is to use epistemic and aleatoric uncertainty in an online setting while recording a biosignal. An

increase in uncertainty may correspond with artefacts in the data, making uncertainty an artefact detector with possibly better properties than normal artefact classifiers. One advantage is that it may only detect artefacts that are obstructing a good classification, allowing it to tolerate artefacts in channels or at timepoints where they do not pose a problem for the specific task.

Moreover, there may be many possible artefacts for some setups, so creating a dataset to train an artefact classifier may require a very large dataset. Even then, some artefacts may be too rare to occur in the dataset, while they could still arise in a recording. Similarly, there may be many anomalies in clean data that actually correspond with a physiological anomaly. These anomalies may specifically be markers for a (rare) condition that needs to be detected. This makes artefact-classification a difficult task for a dedicated model, but since epistemic uncertainty does not need artefact-datasets, it has the potential to outperform dedicated artefact classifiers.

There may be many more unexplored opportunities to use estimated uncertainties when these uncertainty-enabled models are integrated in a task environment. Perhaps in a neurorehabilitation BCI the uncertainty may be used to support the patient in improving their movement attempts, or in situations where the labels may be erroneous an uncertainty measure is able detect mislabeled training samples.

Holistic models for multi-task diagnosis on Biosignals may also be benefited by uncertainty quantification, or may even produce novel techniques to quantify uncertainty.

G. Informative Priors

Variational Inference gives a modeller the option to specify a prior $p(\theta)$. This prior may be very helpful in training good Bayesian Neural Networks when data is limited. Efforts to cast domain knowledge into a probability distribution for $p(\theta)$ may be non-trivial, but has the potential to improve these models.

Alternatively, the prior $p(\theta)$ may also be learned on datasets similar to the task at hand [25].

H. Rejected Samples

We see that several works reject difficult samples to improve accuracy. In medical diagnosis systems the assumption is that these difficult samples may be offered to a diagnostician, so that there quality of diagnosis may not be compromised by mistakes in the Neural Network. However, it is unclear what the resulting diagnostic performance of the system combining the doctor and the Neural Network would be. Both may find the same samples difficult, or completely different samples.

In Breast Cancer and Tubercolosis screening some theoretical work with historical data has been done [92]. Similar research may be done within the biosignal domain as a step towards implementing models with Uncertainty Quantification in the medical biosignal domain.

VII. Conclusion

This review finds that Uncertainty Quantification methods for Neural Networks are increasingly being used in Biosignal domain, but that there are some hurdles to overcome.

By providing a clarification about how uncertainty measures relate to aleatoric and epistemic uncertainty, and by providing an end-to-end guideline on how to add uncertainty quantification to a Biosignal classifying Neural Network we aim to make uncertainty quantification more accessible to researchers working with EEG, ECG, sEMG and EOG.

Many areas still remain to be explored. Uncertainty Quantification methods should be further studied in situ, where clinicians may perform specific actions based on predicted uncertainty. To this end, studies that investigate the performance of a (clinical) environment containing an uncertainty-estimating model are needed.

References

[1] J. He, S. L. Baxter, J. Xu, J. Xu, X. Zhou, and K. Zhang, "The practical implementation of artificial intelligence technologies in medicine," *Nature medicine*, vol. 25, no. 1, pp. 30–36, 2019.

[2] M. Abdar, F. Pourpanah, S. Hussain, *et al.*, "A review of uncertainty quantification in deep learning: Techniques, applications and challenges," *Information Fusion*, vol. 76, pp. 243–297, 2021.

[3] A. Malhotra, M. Younes, S. T. Kuna, *et al.*, "Performance of an automated polysomnography scoring system versus computer-assisted manual scoring," *Sleep*, vol. 36, no. 4, pp. 573–582, 2013.

[4] R. Chai, G. R. Naik, S. H. Ling, Y. Tran, A. Craig, and H. T. Nguyen, "Channels selection using independent component analysis and scalp map projection for EEG-based driver fatigue classification," in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, ISSN: 1558-4615, Jul. 2017, pp. 1808–1811. DOI: 10.1109/EMBC.2017.8037196.

[5] Rifai Chai, Y. Tran, G. R. Naik, *et al.*, "Classification of EEG based-mental fatigue using principal component analysis and Bayesian neural network," *Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual International Conference*, vol. 2016, pp. 4654–4657, Aug. 2016, Place: United States Publisher: [IEEE], ISSN: 2694-0604. DOI: 10.1109/EMBC.2016.7591765.

[6] Rifai Chai, M. R. Smith, T. N. Nguyen, Sai Ho Ling, A. J. Coutts, and H. T. Nguyen, "Comparing features extractors in EEG-based cognitive fatigue detection of demanding computer tasks," *Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual International Conference*, vol. 2015, pp. 7594–7597, 2015, Place: United States Publisher: [IEEE], ISSN: 2694-0604. DOI: 10.1109/EMBC.2015.7320150.

[7] F. Burden and D. Winkler, "Bayesian regularization of neural networks," *Artificial neural networks: methods and applications*, pp. 23–42, 2009.

[8] E. Hüllermeier and W. Waegeman, "Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods," *Machine Learning*, vol. 110, pp. 457–506, 2021.

[9] J. Yang, K. Zhou, Y. Li, and Z. Liu, "Generalized out-of-distribution detection: A survey," *arXiv preprint arXiv:2110.11334*, 2021.

[10] H. van Gorp, I. A. M. Huijben, P. Fonseca, R. J. G. van Sloun, S. Overeem, and M. M. van Gilst, "Certainty about uncertainty in sleep staging: A theoretical framework," English, *Sleep*, vol. 45, no. 8, Aug. 2022, Place: Cary Publisher: Oxford Univ Press Inc WOS:000814108500001, ISSN: 0161-8105. DOI: 10.1093/sleep/zsac134. (visited on 01/16/2023).

[11] C. M. Bishop and N. M. Nasrabadi, *Pattern recognition and machine learning*. Springer, 2006, vol. 4.

[12] J. Van Amersfoort, L. Smith, Y. W. Teh, and Y. Gal, "Uncertainty estimation using a single deep deterministic neural network," in *International conference on machine learning*, PMLR, 2020, pp. 9690–9700.

[13] J. Mukhoti, A. Kirsch, J. van Amersfoort, P. H. Torr, and Y. Gal, "Deep deterministic uncertainty: A simple baseline," *arXiv e-prints*, arXiv–2102, 2021.

[14] Y. Lin, R. Palaniappan, P. De Wilde, and L. Li, "Reliability Analysis for Finger Movement Recognition With Raw Electromyographic Signal by Evidential Convolutional Networks," English, *Ieee Transactions on Neural Systems and Rehabilitation Engineering*, vol. 30, pp. 96–107, 2022, Place: Piscataway Publisher: Ieee-Inst Electrical Electronics Engineers Inc WOS:000748370800010, ISSN: 1534-4320. DOI: 10.1109/TNSRE.2022.3141593. (visited on 01/31/2023).

[15] A. Jøsang, *Subjective logic*. Springer, 2016, vol. 4.

[16] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[17] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *international conference on machine learning*, PMLR, 2016, pp. 1050–1059.

[18] T. Xia, J. Han, and C. Mascolo, "Benchmarking uncertainty quantification on biosignal classification tasks under dataset shift," in *Multimodal AI in healthcare: A paradigm shift in health intelligence*, Springer, 2022, pp. 347–359.

[19] R. Harper and J. Southern, "A Bayesian Deep Learning Framework for End-To-End Prediction of Emotion From Heartbeat," English, *Ieee Transactions on Affective Computing*, vol. 13, no. 2, pp. 985–991, Jun. 2022, Place: Piscataway Publisher: Ieee-Inst Electrical Electronics Engineers Inc WOS:000804643000033, ISSN: 1949-3045. DOI: 10.1109/TAFFC.2020.2981610. (visited on 01/16/2023).

[20] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," *Advances in neural information processing systems*, vol. 30, 2017.

[21] T. Pearce, F. Leibfried, and A. Brintrup, "Uncertainty in neural networks: Approximately bayesian ensembling," in *International conference on artificial intelligence and statistics*, PMLR, 2020, pp. 234–244.

[22] A. G. Wilson and P. Izmailov, "Bayesian deep learning and a probabilistic perspective of generalization," *Advances in neural information processing systems*, vol. 33, pp. 4697–4708, 2020.

[23] L. V. Jospin, H. Laga, F. Boussaid, W. Buntine, and M. Bennamoun, "Hands-on bayesian neural networks - a tutorial for deep learning users," *IEEE Computational Intelligence Magazine*, vol. 17, no. 2, pp. 29–48, 2022.

[24] Y. Ovadia, E. Fertig, J. Ren, *et al.*, "Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift," *Advances in neural information processing systems*, vol. 32, 2019.

[25] R. Shwartz-Ziv, M. Goldblum, H. Souri, *et al.*, "Pretrain your loss: Easy bayesian transfer learning with informative priors," *Advances in Neural Information Processing Systems*, vol. 35, pp. 27 706–27 715, 2022.

[26] D. P. Kingma, M. Welling, *et al.*, "An introduction to variational autoencoders," *Foundations and Trends® in Machine Learning*, vol. 12, no. 4, pp. 307–392, 2019.

[27] J. Belen, S. Mousavi, A. Shamsoshoara, and F. Afghah, "An Uncertainty Estimation Framework for Risk Assessment in Deep Learning-based AFib Classification," in *2020 54th Asilomar Conference on Signals, Systems, and Computers*, ISSN: 2576-2303, Nov. 2020, pp. 960–964. DOI: 10.1109/IEEECONF51394.2020.9443466.

[28] R. R. Van De Leur, B. Hompot, R. J. Hassink, P. A. Doevendans, and R. Van Es, "Interpretable uncertainty estimation for automated triage of 12-lead electrocardiogram using deep convolutional neural networks," English, *European Heart Journal*, vol. 42, pp. 3163–3163, Oct. 2021, Place: Oxford Publisher: Oxford Univ Press WOS:000720456903464, ISSN: 0195-668X. (visited on 01/31/2023).

[29] M. Seitzer, A. Tavakoli, D. Antic, and G. Martius, "On the pitfalls of heteroscedastic uncertainty estimation with probabilistic neural networks," *arXiv preprint arXiv:2203.09168*, 2022.

[30] A. Malinin and M. Gales, "Predictive uncertainty estimation via prior networks," *Advances in neural information processing systems*, vol. 31, 2018.

[31] V. Jahmunah, E. Y. K. Ng, R.-S. Tan, S. L. Oh, and U. R. Acharya, "Uncertainty quantification in DenseNet model using myocardial infarction ECG signals," English, *Computer Methods and Programs in Biomedicine*, vol. 229, p. 107 308, Feb. 2023, Place: Clare Publisher: Elsevier Ireland Ltd WOS:000905967100001, ISSN: 0169-2607. DOI: 10.1016/j.cmpb.2022.107308. (visited on 01/31/2023).

[32] Y. Lin, R. Palaniappan, P. De Wilde, and L. Li, "Robust Long-Term Hand Grasp Recognition With Raw Electromyographic Signals Using Multidimensional Uncertainty-Aware Models," *IEEE Transactions*

on Neural Systems and Rehabilitation Engineering, pp. 1–1, 2023, Conference Name: IEEE Transactions on Neural Systems and Rehabilitation Engineering, ISSN: 1558-0210. DOI: 10.1109/TNSRE.2023.3236982.

[33] M. Sensoy, L. Kaplan, and M. Kandemir, "Evidential deep learning to quantify classification uncertainty," *Advances in neural information processing systems*, vol. 31, 2018.

[34] E. Schulz, M. Speekenbrink, and A. Krause, "A tutorial on gaussian process regression: Modelling, exploring, and exploiting functions," *Journal of Mathematical Psychology*, vol. 85, pp. 1–16, 2018.

[35] F. S. Costabal, K. Matsuno, J. Yao, P. Perdikaris, and E. Kuhl, "Machine learning in drug development: Characterizing the effect of 30 drugs on the QT interval using Gaussian process regression, sensitivity analysis, and uncertainty quantification," English, *Computer Methods in Applied Mechanics and Engineering*, vol. 348, pp. 313–333, May 2019, Place: Lausanne Publisher: Elsevier Science Sa WOS:000462472400012, ISSN: 0045-7825. DOI: 10.1016/j.cma.2019.01.033. (visited on 01/31/2023).

[36] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *International conference on machine learning*, PMLR, 2017, pp. 1321–1330.

[37] Y. Elul, A. A. Rosenberg, A. Schuster, A. M. Bronstein, and Y. Yaniv, "Meeting the unmet needs of clinicians from AI systems showcased for cardiology with deep-learning-based ECG analysis," English, *Proceedings of the National Academy of Sciences of the United States of America*, vol. 118, no. 24, e2020620118, Jun. 2021, Place: Washington Publisher: Natl Acad Sciences WOS:000668851100003, ISSN: 0027-8424. DOI: 10.1073/pnas.2020620118. (visited on 01/31/2023).

[38] M. D. Hoffman, D. M. Blei, C. Wang, and J. Paisley, "Stochastic variational inference," *Journal of Machine Learning Research*, 2013.

[39] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006, vol. 2.

[40] V. Schetinin and C. Maple, "A Bayesian model averaging methodology for detecting EEG artifacts," 2007, pp. 499–502. DOI: 10.1109/ICDSP.2007.4288628.

[41] R. Hagan, C. Gillan, and F. Mallett, "Comparison of machine learning methods for the classification of cardiovascular disease," English, *Informatics in Medicine Unlocked*, vol. 24, 2021, ISSN: 2352-9148. DOI: 10.1016/j.imu.2021.100606.

[42] Y. Bekhti, F. Lucka, J. Salmon, and A. Gramfort, "A hierarchical Bayesian perspective on majorization-minimization for non-convex sparse regression: Application to M/EEG source imaging," English, *Inverse Problems*, vol. 34, no. 8, p. 085 010, Aug. 2018, Place: Bristol Publisher: Iop Publishing Ltd WOS:000436948900001, ISSN: 0266-5611. DOI: 10.1088/1361-6420/aac9b3. (visited on 01/16/2023).

[43] D. J. C. Mackay, "Bayesian methods for adaptive models," in California Institute of Technology, 1992, ch. 5.2.

[44] N. Mohamed, D. Rubin, and T. Marwala, "Detection of epileptiform activity in human EEG signals using Bayesian neural networks," in *IEEE 3rd International Conference on Computational Cybernetics, 2005. ICCC 2005.*, Apr. 2005, pp. 231–237. DOI: 10.1109/ICCCYB.2005.1511578.

[45] W. J. Maddox, P. Izmailov, T. Garipov, D. P. Vetrov, and A. G. Wilson, "A simple baseline for bayesian uncertainty in deep learning," *Advances in neural information processing systems*, vol. 32, 2019.

[46] R. S. Wabina and C. Silpasuwanchai, "Neural stochastic differential equations network as uncertainty quantification method for EEG source localization," *Biomedical physics & engineering express*, Nov. 2022, Place: England Publisher: IOP Publishing Ltd, ISSN: 2057-1976. DOI: 10.1088/2057-1976/aca20b.

[47] L. Kong, J. Sun, and C. Zhang, "Sde-net: Equipping deep neural networks with uncertainty estimates," *arXiv preprint arXiv:2008.10546*, 2020.

[48] A. Campbell, L. Qendro, P. Lió, and C. Mascolo, "Robust and Efficient Uncertainty Aware Biosignal Classification via Early Exit Ensembles," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, ISSN: 2379-190X, May 2022, pp. 3998–4002. DOI: 10.1109/ICASSP43922.2022.9746330.

[49] S. Teerapittayanon, B. McDanel, and H.-T. Kung, "Branchynet: Fast inference via early exiting from deep neural networks," in *2016 23rd International Conference on Pattern Recognition (ICPR)*, IEEE, 2016, pp. 2464–2469.

[50] A. Montanari, M. Sharma, D. Jenkus, M. Alloulah, L. Qendro, and F. Kawsar, "Eperceptive: Energy reactive embedded intelligence for batteryless sensors," in *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*, 2020, pp. 382–394.

[51] J. Martinez, A. Akbari, K. Sel, and R. Jafari, "Strategic Attention Learning for Modality Translation," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, ISSN: 2379-190X, May 2020, pp. 1030–1034. DOI: 10.1109/ICASSP40776.2020.9053515.

[52] G. Sovatzidi, M. Vasilakakis, and D. Iakovidis, "Constructive Fuzzy Cognitive Map for Depression Severity Estimation," *Studies in Health Technology and Informatics*, vol. 294, pp. 485–489, 2022. DOI: 10.3233/SHTI220506.

[53] A. Amirkhani, E. I. Papageorgiou, A. Mohseni, and M. R. Mosavi, "A review of fuzzy cognitive maps in medicine: Taxonomy, methods, and applications," *Computer methods and programs in biomedicine*, vol. 142, pp. 129–145, 2017.

[54] T. Duan, Z. Wang, S. Liu, Y. Yin, and S. N. Srihari, "UNCER: A framework for uncertainty estimation and reduction in neural decoding of EEG signals," en, *Neurocomputing*, vol. 538, p. 126 210, Jun. 2023, ISSN:

09252312. DOI: 10.1016/j.neucom.2023.03.071. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0925231223003156 (visited on 09/20/2023).

[55] J. Chang, Z. Lan, C. Cheng, and Y. Wei, "Data uncertainty learning in face recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 5710–5719.

[56] Z. Deng, C. Li, R. Song, X. Liu, R. Qian, and X. Chen, "EEG-based seizure prediction via hybrid vision transformer and data uncertainty learning," en, *Engineering Applications of Artificial Intelligence*, vol. 123, p. 106 401, Aug. 2023, ISSN: 09521976. DOI: 10.1016/j.engappai.2023.106401. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0952197623005857 (visited on 09/20/2023).

[57] E. I. Chetkin, S. L. Shishkin, and B. L. Kozyrskiy, "Bayesian opportunities for brain–computer interfaces: Enhancement of the existing classification algorithms and out-of-domain detection," *Algorithms*, vol. 16, no. 9, p. 429, 2023.

[58] R. T. Schirrmeister, J. T. Springenberg, L. D. J. Fiederer, *et al.*, "Deep learning with convolutional neural networks for eeg decoding and visualization," *Human brain mapping*, vol. 38, no. 11, pp. 5391–5420, 2017.

[59] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance, "Eegnet: A compact convolutional neural network for eeg-based brain–computer interfaces," *Journal of neural engineering*, vol. 15, no. 5, p. 056 013, 2018.

[60] I. Obeid and J. Picone, "The temple university hospital eeg data corpus," *Frontiers in neuroscience*, vol. 10, p. 196, 2016.

[61] S. De Rooij, K. Batselier, and B. Hunyadi, "Enabling large-scale probabilistic seizure detection with a tensor-network kalman filter for ls-svm," English, in *ICASSPW 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing Workshops, Proceedings*, ser. ICASSPW 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing Workshops, Proceedings, Green Open Access added to TU Delft Institutional Repository 'You share, we take care!' – Taverne project https://www.openaccess.nl/en/you-share-we-take-care Otherwise as indicated in the copyright section: the publisher is the copyright holder of this work and the author uses the Dutch legislation to make this work public. ; 2023 IEEE International Conference on Acoustics, Speech and Signal Processing Workshops, ICASSPW 2023 ; Conference date: 04-06-2023 Through 10-06-2023, United States: Institute of Electrical and Electronics Engineers (IEEE), 2023. DOI: 10.1109/ICASSPW59220.2023.10193615.

[62] D. Milanes-Hermosilla, R. T. Codorniu, R. Lopez-Baracaldo, *et al.*, "Monte Carlo Dropout for Uncertainty Estimation and Motor Imagery Classification," English, *Sensors*, vol. 21, no. 21, p. 7241, Nov. 2021, Place: Basel Publisher: Mdpi WOS:000719045200001. DOI: 10.3390/s21217241. (visited on 01/16/2023).

[63] A. Borovac, T. P. Runarsson, G. Thorvardsson, and S. Gudmundsson, "Calibration of Automatic Seizure Detection Algorithms," in *2022 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, ISSN: 2473-716X, Dec. 2022, pp. 1–6. DOI: 10.1109/SPMB55497.2022.10014868.

[64] L. Fiorillo, P. Favaro, and F. D. Faraci, "DeepSleepNet-Lite: A Simplified Automatic Sleep Stage Scoring Model With Uncertainty Estimates," English, *Ieee Transactions on Neural Systems and Rehabilitation Engineering*, vol. 29, pp. 2076–2085, 2021, Place: Piscataway Publisher: Ieee-Inst Electrical Electronics Engineers Inc WOS:000709073200001, ISSN: 1534-4320. DOI: 10.1109/TNSRE.2021.3117970. (visited on 01/16/2023).

[65] A. O. Aseeri, "Uncertainty-Aware Deep Learning-Based Cardiac Arrhythmias Classification Model of Electrocardiogram Signals," English, *Computers*, vol. 10, no. 6, p. 82, Jun. 2021, Place: Basel Publisher: Mdpi WOS:000665560000001, ISSN: 2073-431X. DOI: 10.3390/computers10060082. (visited on 01/31/2023).

[66] C. Stoean, R. Stoean, M. Atencia, *et al.*, "Automated Detection of Presymptomatic Conditions in Spinocerebellar Ataxia Type 2 Using Monte Carlo Dropout and Deep Neural Network Techniques with Electrooculogram Signals," English, *Sensors*, vol. 20, no. 11, p. 3032, Jun. 2020, Place: Basel Publisher: Mdpi WOS:000552737900025. DOI: 10.3390/s20113032. (visited on 01/31/2023).

[67] K. Zanna, K. Sridhar, H. Yu, and A. Sano, "Bias Reducing Multitask Learning on Mental Health Prediction," in *2022 10th International Conference on Affective Computing and Intelligent Interaction (ACII)*, ISSN: 2156-8111, Oct. 2022, pp. 1–8. DOI: 10.1109/ACII55700.2022.9953850.

[68] N. Strodthoff, P. Wagner, T. Schaeffter, and W. Samek, "Deep Learning for ECG Analysis: Benchmarks and Insights from PTB-XL," English, *Ieee Journal of Biomedical and Health Informatics*, vol. 25, no. 5, pp. 1519–1528, May 2021, Place: Piscataway Publisher: Ieee-Inst Electrical Electronics Engineers Inc WOS:000649625200018, ISSN: 2168-2194. DOI: 10.1109/JBHI.2020.3022989. (visited on 01/31/2023).

[69] M. Valdenegro-Toro and D. S. Mori, "A deeper look into aleatoric and epistemic uncertainty disentanglement," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, IEEE, 2022, pp. 1508–1516.

[70] H. Phan, K. Mikkelsen, O. Y. Chen, P. Koch, A. Mertins, and M. De Vos, "SleepTransformer: Automatic Sleep Staging With Interpretability and Uncertainty Quantification," English, *Ieee Transactions on Biomedical Engineering*, vol. 69, no. 8, pp. 2456–2467, Aug. 2022, Place: Piscataway Publisher: Ieee-Inst Electrical Electronics Engineers Inc WOS:000838529800011, ISSN: 0018-9294. DOI: 10.1109/TBME.2022.3147187. (visited on 01/16/2023).

[71] L. Smith and Y. Gal, "Understanding measures of uncertainty for adversarial example detection," *arXiv preprint arXiv:1803.08533*, 2018.

[72] A. Niculescu-Mizil and R. Caruana, "Predicting good probabilities with supervised learning," in *Proceedings of the 22nd international conference on Machine learning*, 2005, pp. 625–632.

[73] M. Barandas, L. Famiglini, A. Campagner, *et al.*, "Evaluation of uncertainty quantification methods in multi-label classification: A case study with automatic diagnosis of electrocardiogram," en, *Information Fusion*, vol. 101, p. 101 978, Jan. 2024, ISSN: 15662535. DOI: 10.1016/j.inffus.2023.101978. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1566253523002944 (visited on 09/20/2023).

[74] D. Milanés-Hermosilla, R. Trujillo-Codorniú, S. Lamar-Carbonell, *et al.*, "Robust Motor Imagery Tasks Classification Approach Using Bayesian Neural Network," en, *Sensors*, vol. 23, no. 2, p. 703, Jan. 2023, ISSN: 1424-8220. DOI: 10.3390/s23020703. [Online]. Available: https://www.mdpi.com/1424-8220/23/2/703 (visited on 09/20/2023).

[75] X. Huang, J. Yang, L. Li, H. Deng, B. Ni, and Y. Xu, "Evaluating and boosting uncertainty quantification in classification," *arXiv preprint arXiv:1909.06030*, 2019.

[76] G.-P. Bonneau, H.-C. Hege, C. R. Johnson, *et al.*, "Overview and state-of-the-art of uncertainty visualization," *Scientific Visualization: Uncertainty, Multifield, Biomedical, and Scalable Visualization*, pp. 3–27, 2014.

[77] K. Potter, P. Rosen, and C. R. Johnson, "From quantification to visualization: A taxonomy of uncertainty visualization approaches," *IFIP advances in information and communication technology*, vol. 377, p. 226, 2012.

[78] S. Renooij and C. Witteman, "Talking probabilities: Communicating probabilistic information with words and numbers," *International Journal of Approximate Reasoning*, vol. 22, no. 3, pp. 169–194, 1999.

[79] R. S. Gill, H.-M. Lee, B. Caldairou, *et al.*, "Multicenter Validation of a Deep Learning Detection Algorithm for Focal Cortical Dysplasia," English, *Neurology*, vol. 97, no. 16, E1571–E1582, Oct. 2021, Place: Philadelphia Publisher: Lippincott Williams & Wilkins WOS:000708601400019, ISSN: 0028-3878. DOI: 10.1212/WNL.0000000000012698. (visited on 01/16/2023).

[80] Y. Gal, R. Islam, and Z. Ghahramani, "Deep bayesian active learning with image data," in *International conference on machine learning*, PMLR, 2017, pp. 1183–1192.

[81] R. Ma, H. Zhang, J. Zhang, *et al.*, "Bayesian Uncertainty Modeling for P300-Based Brain-Computer Interface," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 31, pp. 2789–2799, 2023, ISSN: 1534-4320, 1558-0210. DOI: 10.1109/TNSRE.2023.3286688. [Online]. Available: https://ieeexplore.ieee.org/document/10153625/ (visited on 09/20/2023).

[82] O. Sagi and L. Rokach, "Ensemble learning: A survey," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 8, no. 4, e1249, 2018.

[83] Y. Zhang and Q. Yang, "A survey on multi-task learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 12, pp. 5586–5609, 2021.

[84] N. Brosse, C. Riquelme, A. Martin, S. Gelly, and É. Moulines, "On last-layer algorithms for classification: Decoupling representation from uncertainty estimation," *arXiv preprint arXiv:2001.08049*, 2020.

[85] S. Prince, *Computer Vision: Models Learning and Inference*. Cambridge University Press, 2012.

[86] M. Valdenegro-Toro, "Exploring the limits of epistemic uncertainty quantification in low-shot settings," *arXiv preprint arXiv:2111.09808*, 2021.

[87] O. Kostopoulou, K. Arora, and B. Pálfi, "Using cancer risk algorithms to improve risk estimates and referral decisions," *Communications Medicine*, vol. 2, no. 1, p. 2, 2022.

[88] B. Pálfi, K. Arora, and O. Kostopoulou, "Algorithm-based advice taking and clinical judgement: Impact of advice distance and algorithm information," *Cognitive research: principles and implications*, vol. 7, no. 1, p. 70, 2022.

[89] S. Somani, A. J. Russak, F. Richter, *et al.*, "Deep learning and the electrocardiogram: Review of the current state-of-the-art," *EP Europace*, vol. 23, no. 8, pp. 1179–1191, 2021.

[90] I. Kawashima and H. Kumano, "Prediction of mind-wandering with electroencephalogram and non-linear regression modeling," *Frontiers in Human Neuroscience*, vol. 11, p. 365, 2017.

[91] Y.-C. Yeh, W.-J. Wang, and C. W. Chiou, "Cardiac arrhythmia diagnosis method using linear discriminant analysis on ecg signals," *Measurement*, vol. 42, no. 5, pp. 778–789, 2009.

[92] K. Dvijotham, J. Winkens, M. Barsbey, *et al.*, "Enhancing the reliability and accuracy of ai-enabled diagnosis via complementarity-driven deferral to clinicians," *Nature Medicine*, pp. 1–7, 2023.