Bachelor thesis

Artificial Intelligence

Radboud University

On the Reliability of the Uncertainty Quantified by a Convolutional Bayesian Neural Network

Author:
Alaa Elshamouty
s1020320

First supervisor:
Dr. M. Hinne
Artificial Intelligence
m.hinne@donders.ru.nl

Second supervisor:
Dr. L. Ambrogioni
Artificial Intelligence
l.ambrogioni@donders.ru.nl

June 18, 2021

Abstract

Deep learning (DL) is achieving a lot of breakthroughs in different fields, such as object detection, segmentation, and recognition [8, 15, 20]. However, DL still fails to reason about its decisions and hence is not widely used in safety critical applications yet [1]. Integrating Bayesian statistics with DL, e.g. Bayesian Neural Networks (BNN), provides versatility and the reasoning ability through decision confidence for DL. BNNs offer to accompany each decision taken with an uncertainty quantification metric that can be decomposed into two kinds of uncertainty: aleatoric and epistemic. Aleatoric uncertainty being the uncertainty caused due to inherent noise in the data, and epistemic uncertainty being the uncertainty due to improper model or lack of knowledge, i.e. lack of training data. A considerable amount of research underlies what these uncertainties capture and how they can be interpreted, however, not much on their reliability. Therefore, the research question is: *How reliable is the uncertainty quantified by a Bayesian Lenet[17], a convolutional bayesian neural network (CBNN) trained using Variational Inference?*

To answer the research question, seven hypotheses are proposed capturing the expected behaviour of the prediction accuracy, aleatoric uncertainty, and epistemic uncertainty, all quantified by the Bayesian Lenet using two different parameter sampling techniques, under small and large data shifts. In particular, small data shifts are approached by adding noise following Gaussian and Poisson distributions, or masking out parts of the inputs. Large data shifts are approached by passing data from a different dataset, e.g., training the network on recognizing digits and then passing alphabets or pictures of objects to it. Accordingly, the network's uncertainty is quantified, and visualized using Layer Wise Propagation approach. As a result of this study, a pipeline towards qualitatively understanding and visualizing the uncertainty quantified by the Bayesian Lenet model is constructed. This pipeline can be further used to explore the reliability of different uncertainty quantification methods and network architectures.

The hypotheses were qualitatively evaluated on two datasets, namely, MNIST and CIFAR10. Overall, results show that the uncertainty quantification method used in this work is able to detect small and large data shifts. However, out of the seven hypotheses, only one hypothesis holds, four partially hold, and two do not hold. It is observed that the reliability of the uncertainty quantification methods used is dependent on many factors, such as, the model's achieved training accuracy, the model parameters' sampling technique, dataset, and noise type.

Contents

Introduction

With the rising interest in self diving cars, medical image analysis, and security, researchers devoted time into improving image classification networks. CNN became the state of art for many image classification tasks [20, 15, 8]. In safety critical applications, it is beneficial to know how uncertain the network is with its prediction. Generally, this introduces the possibility for networks to indicate indecisiveness rather than being forced to, most likely incorrectly, classify. This can be applied, for example, to reduce the well known accidents in autonomous cars, often caused by object misclassification [6], by instead giving back control to the user. DL models that trade in stocks can also benefit from that uncertainty to recognize unfamiliar scenarios and avoid partaking in them. Initially, scholars deviated their attention from frequentest deep neural networks (DNN) to BNNs to reduce the DL problem of overfitting [1]. Instead of learning parameters as point estimates in DNNs, they are learned as distributions in BNNs allowing more versatility and less overly confident decisions by capturing uncertainty in the training data. But that approach resulted in the opportunity to accompany each decision with an uncertainty value which can be used as a step towards trustworthy and safe models. This lead to considerable research directed towards constructing Bayesian neural architectures [1, 11, 19, 4], alongside methods to quantifying the uncertainties captured by it [10, 14, 21, 16]. Not long afterwards, a growing research territory on understanding and interpreting the quantified uncertainties developed [3, 5, 19]. Within that territory remains a research gap in the reliability of the quantified uncertainties. Therefore, this thesis aims to understand and interpret an uncertainty quantification method, and answer the following research question:

How reliable is the uncertainty quantified by a Bayesian Lenet[17], a CBNN trained using Variational Inference?

To answer this question, the behaviour of the Bayesian Lenet's quantified predicted accuracy, aleatoric uncertainty, and epistemic uncertainty are qualitatively compared to their expected behaviour, posed as hypotheses, under small and large data shifts. As a result, a pipeline was constructed to asses the reliability of the uncertainty quantified by an already implemented Bayesian Lenet [19]. The same pipeline can then be later used to qualitatively test the reliability of the uncertainties quantified under different network architectures and different uncertainty quantification methods.

Outline

Section 2 introduces broad explanations on CNNs, BNNs, CBNNs, Variational Inference, and Bayesian Lenet, alongside the method used to quantify the uncertainties considered in this research. Section 3 provides a short overview of related work on interpreting and understanding the quantified uncertainties. Section 4 introduces the problem domain, the proposed hypotheses for testing reliability, and the pipeline for evaluating the hypotheses. Section 5 details the qualitative analysis that evaluates the hypotheses towards answering the research question. Finally, section 6 discusses the findings and future work.

Background Knowledge

This section briefly introduces the terms used in this paper onwards namely CNNs, BNNs, CBNNs, variational inference, Bayesian Lenet, and the uncertainty quantification method used.

2.1 Convolutional Neural Networks

A convolutional neural network (CNN) is a variation of DNN that is widely used for image classification since its accuracy far surpassed human accuracy in many classification tasks [12, 20]. It is inspired by the hierarchical structure of the primary visual pathway of the brain. The idea behind it is to learn the features/attributes of the image for classification. So for example, the feature extracted by the CNN from the training images of cats can be two ears, whiskers, four legs etc. [15]. It does so through kernels/filters of certain size and cell values that scans the image part by part. This filter, with its own values, slide over each pixel (starting from the top left) and multiply its values with the overlapped pixel values then sum it all up to a single value. This is a mathematical process known as convolution. Normally, an activation function, such as rectified linear unit (ReLu), is applied to the resulting feature matrix and then pooling is performed to reduce the dimensionality for less number of parameters and computations. For classification, the features are feedforwarded through a fully connected network so that now this fully connected network that used to learn and predict based on pixel values learns and predicts based on features.

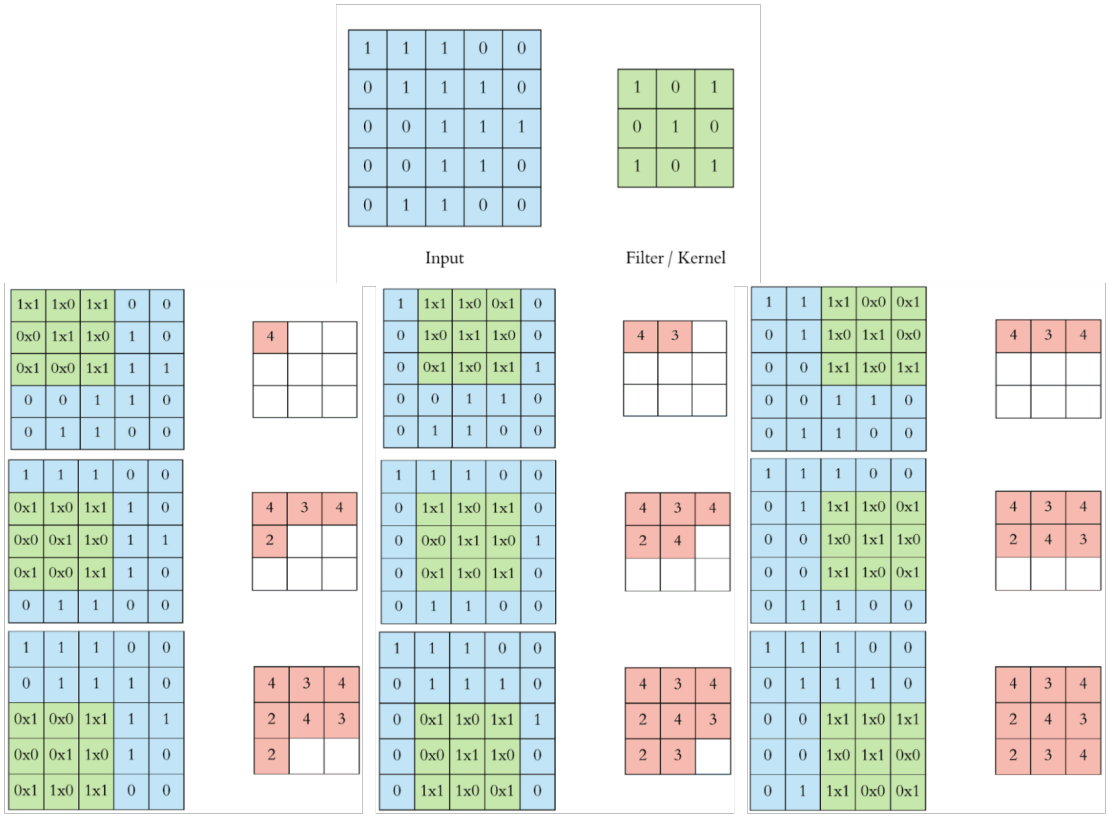

Figure 2.1: Multiplying filter/kernel values with the overlapping pixels and summing it to one value, taken from [7]

2.2 Bayesian Neural Networks

Despite DNN being widely used in multiple domains and being the state of art for majority of classification tasks, it is known to make overconfident decisions [19, 1]. Bayesian Neural Network (BNN) is developed as a technique to tackle this problem of over fitting by learning weights that are instead expressed as a distribution, $P(w|D)$ where w is the weight and D is training data [1]. Regularisation is introduced by placing priors upon the weights and computing the average across models during training. Good prior initialization can result in faster training convergence. Expectations over the learnt weight distribution is what is used to predict new unseen data:

$$P(\hat{y}|\hat{x}) = E_{P(w|D)}[P(\hat{y}|\hat{x}, w)] \tag{2.1}$$

where $\hat{y}$ is the unknown label, and $\hat{x}$ is the unknown input [1].

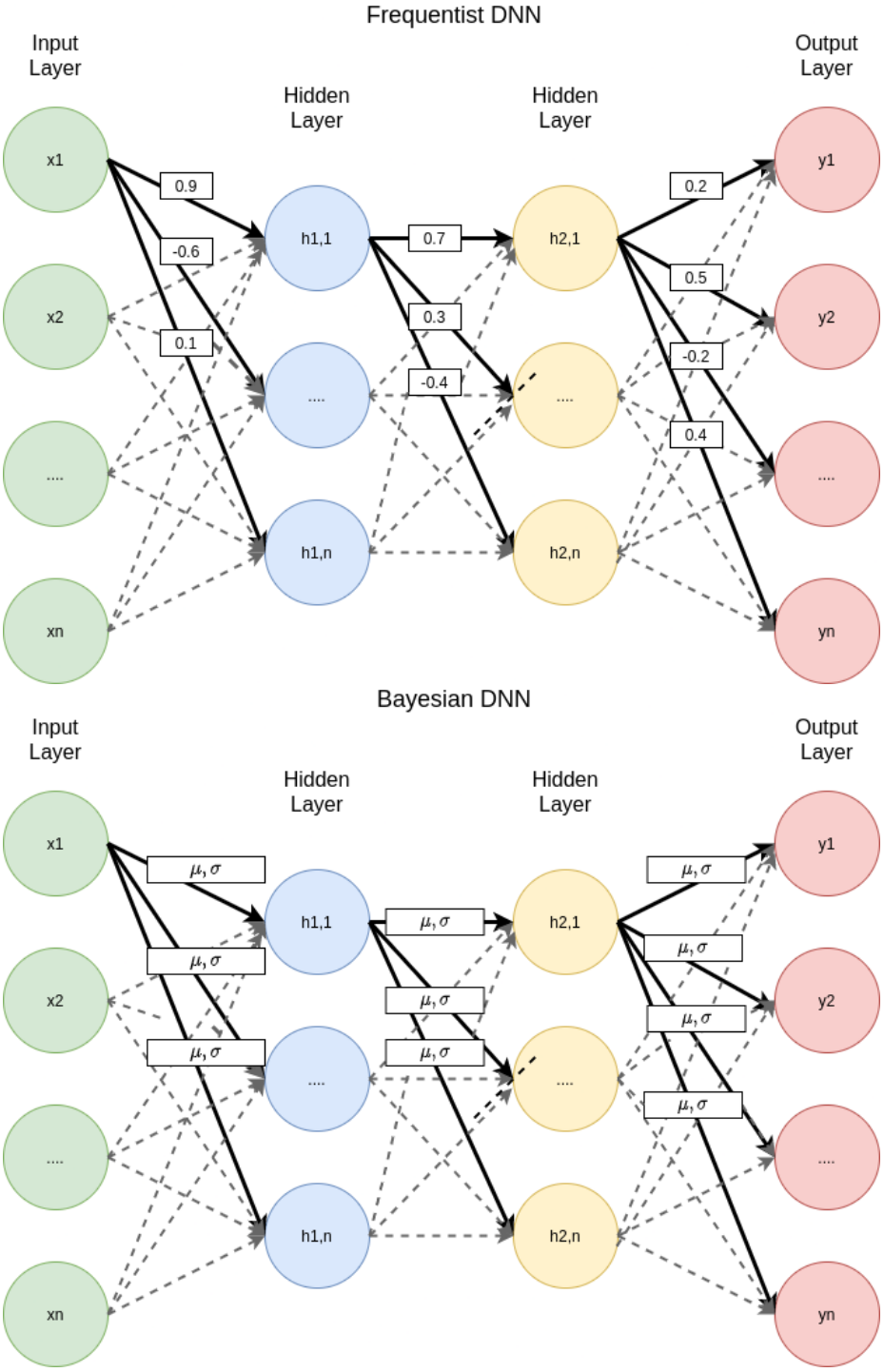

Figure 2.2: Frequentest neural network having point estimate parameters learnt vs Bayesian neural network having distribution parameters learnt.

2.2.1 Variational Inference

Taking expectation over posterior distribution of weights, $E_{P(w|D)}$ (2.1), is intractable as it is equivalent to taking ensemble over infinite number of neural networks [19, 1, 3]. One

approach, namely variational inference, is used to approximate that distribution as suggested by many[19, 1, 11]. This process is known as Bayes by Backprop. The idea is to find the parameters θ such that the Kullback-leibler(KL) divergence between the distribution of the weights given parameters, $q(w|\theta)$, and the true Bayesian posterior on the weights is minimized[1]. In this research project, variational learning on parameters of Gaussian distribution is used.

Reparametrisation trick is rewriting the Expectation over a distribution such that it is independent of the parameters. Local reparametrisation trick is translating the global uncertainty in the weights distribution to local uncertainty independent of the data [19]. That way, having parameters that are outside the distribution will not change the behaviour of the model.

2.2.2 Convolutional Bayesian Neural Network

Convolutional Bayesian Neural Network (CBNN) is a CNN except with parameters (weights and biases) expressed as distribution rather than point estimates. The same idea of modelling the weights in a BNN is projected on CNN by modelling the filter/kernel weight values and sampling from it alongside modelling the weights in the fully connected layers as well.

Frequentest vs Bayesian approach of Convolutional Neural Networks

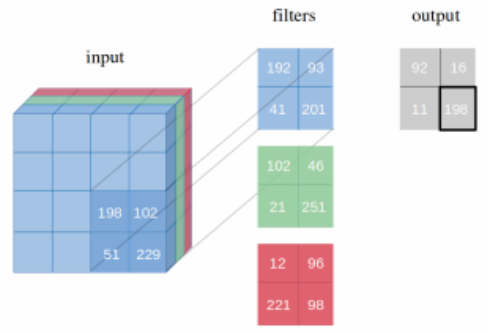

Figure 2.3: CNN with filters represented as point estimate

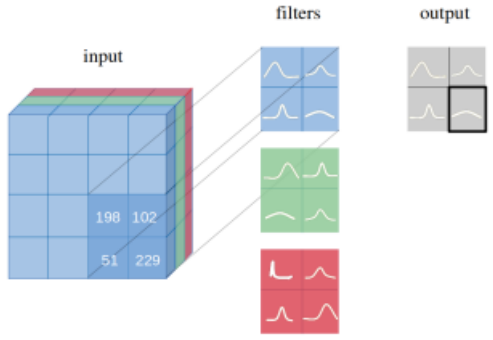

Figure 2.4: CNN with filters represented as distributions

Figure 2.5: Taken from [1]

2.2.3 Bayesian Lenet

Lenet (Fig. 2.6) is a CNN architecture used for digit recognition [17]. The network consists of two main parts:

- The convolution part for feature learning
- The fully connected part for classification

The Bayesian approach of Lenet is learning parameter distributions for both the filter kernels in the convolution part and the weight parameters in the fully connected part instead of point estimates.

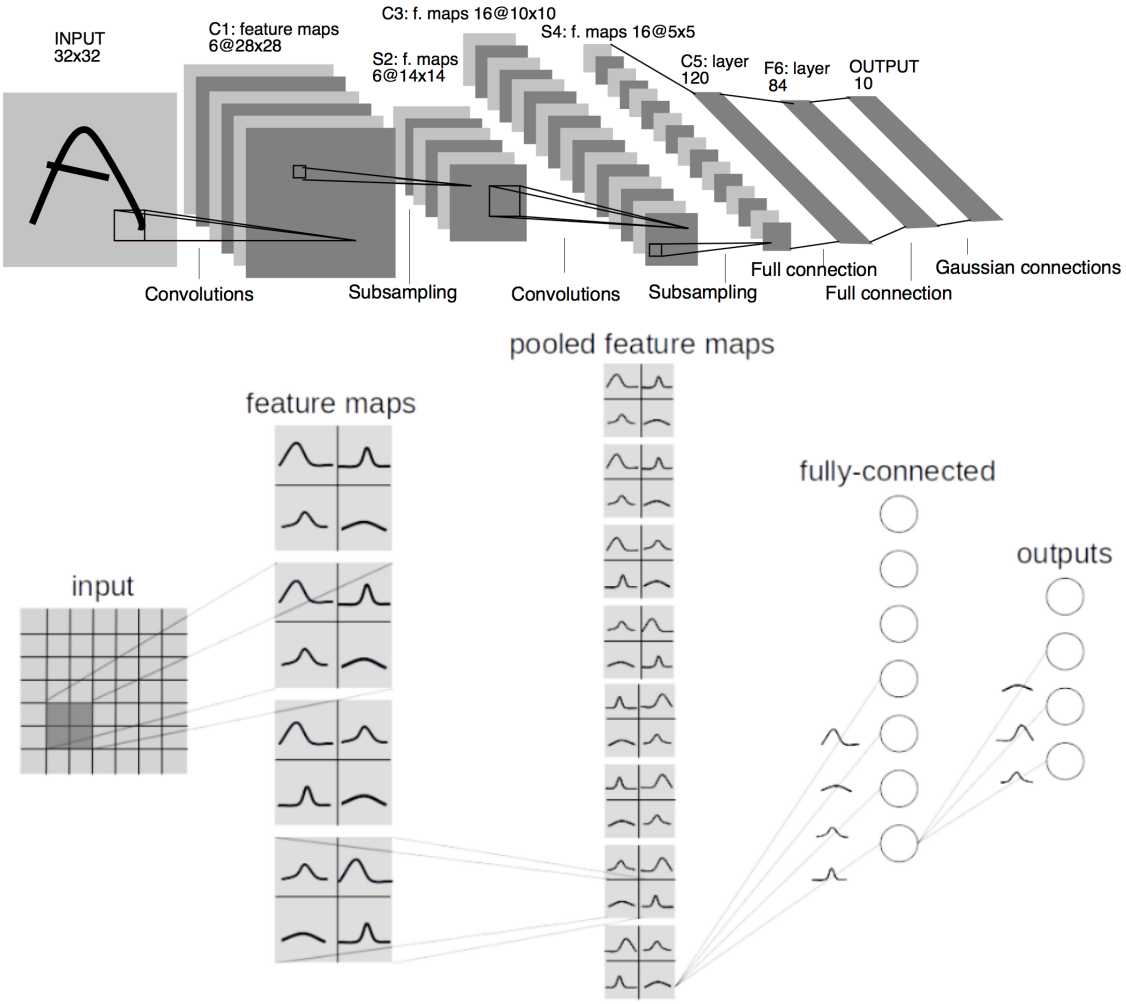

Figure 2.6: The Lenet 5 model kernels and weight parameters are sampled from their learnt distributions. Taken from [17]

To forward pass an image, the model parameters are sampled from their learnt distributions, demonstrated in Fig. 2.7. The model will then output a value for each of the 10 output labels. To capture the uncertainty of the model, the model parameters are sampled many times, Θ, and for each set of those sampled parameters the same image is passed to the network. The decision of the network will be the label with maximum predicted output mean $\underset{y}{\texttt{argmax}}\ \mathbb{E}[y \mid x, \Theta]$.

2.3 Uncertainty Quantification

This section provides an overview on the aleatoric and epistemic uncertainties, and how they are calculated in this research.

2.3.1 Aleatoric and Epistemic Uncertainty

In machine learning, the uncertainty sources are data and model. In Bayesian modelling, we consider two main types of uncertainties that could be modelled[19, 9]:

- Aleatoric Uncertainty: uncertainty caused in the prediction due to noise in the data.
- Epistemic Uncertainty: model uncertainty that is due to improper model or lack of data.

Aleatoric captures uncertainty concerning information that the data cannot explain. There are two types of aleatoric uncertainty: homoscedastic and heteroscedastic. Homoscdeastic is when the noise/unknown information is constant for all samples, e.g. measurement errors or malfunctioning sensor introducing noise to data. Heteroscedastic is when the noise is different across data. In this study, heteroscedastic aleatoric is investigated since the noise values added to the images are sampled from distributions and hence can be different from one input image to another. Epistemic noise describes what the model does not know mainly due to lack of training data or its inappropriateness. Therefore, increasing the training dataset can decrease the epistemic uncertainty but not the aleatoric. It still remains difficult to capture the uncertainties and quantify them as they overlap and require confinement of the model producing it [13]. Some research is done to distinguish them apart. For example, Robin et al. propose a binary classification method that can predict and quantify the aleatoric and epistemic uncertainty such that there is a clear distinction between them. However, it requires an extensive framework setup and hence is used for specific cases [18].

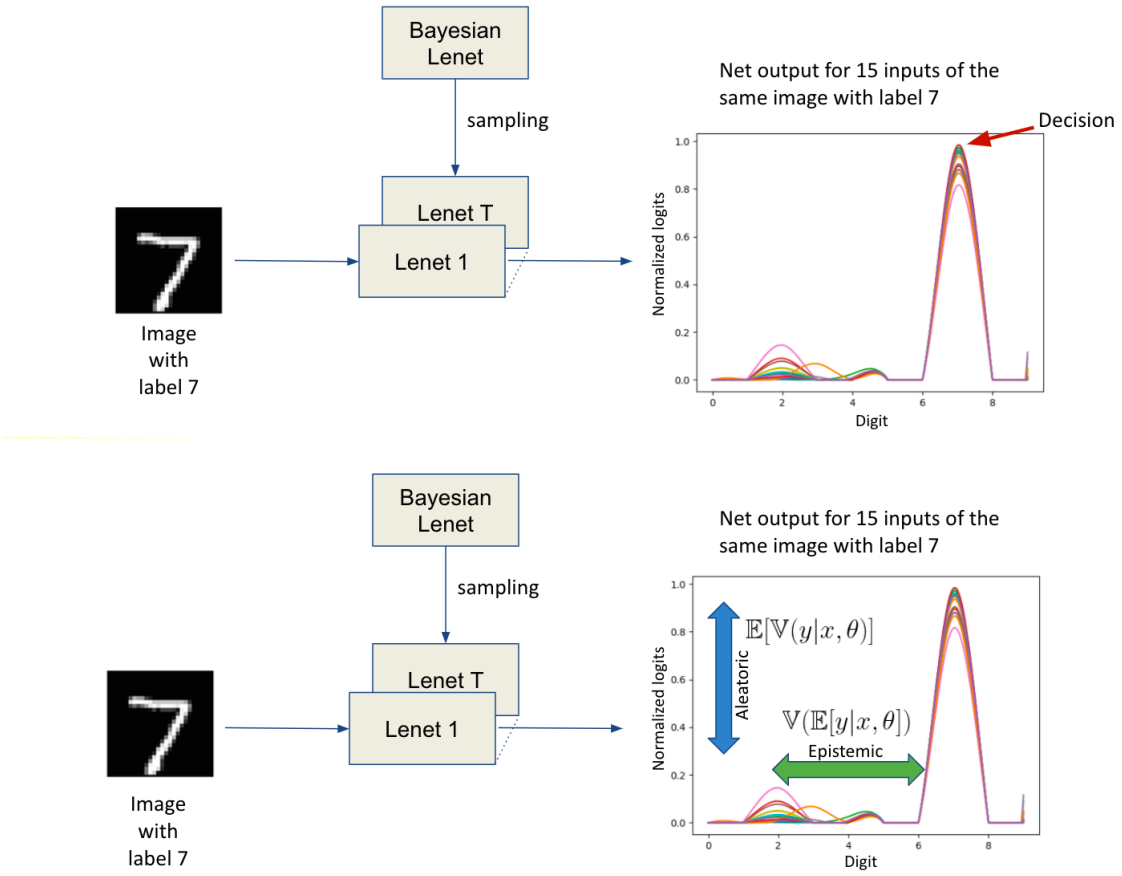

Figure 2.7: Bayesian Lenet forward pass with labelled uncertainties. Aleatoric captures the mean of the variances in each digit. Epistemic captures the variance across the mean of predicted values.

2.3.2 Uncertainty Quantification Method

There are different ways proposed by researchers to quantify uncertainty. For this study, Davis J. et al. method ([9]) is used due to its simplicity and intuitiveness. This subsection will demonstrate how their work was used to quantify the uncertainty of the Bayesian Lenet model used.

The parameters of the trained Bayesian Lenet is now represented as distributions. Instead of each parameter having a point estimate, parameters have a μ and σ values, since Gaussian kernels are used. During a forward pass, each parameter is sampled from a Gaussian distribution with their respective μ and σ values. Now if an image is passed, the parameters are sampled to predict on the image, but if the same image is passed again the parameters sampled may differ. That difference is used to determine the uncertainty of the model captured during the training phase. Therefore, to calculate the total predictive uncertainty of one prediction ($V(y|x)$), the model receives the same image a number of times and the total predictive variance in the values of the output layer is used to quantify the model's uncertainty. An example is shown in figure 2.7 where the same image is passed to the Bayesian Lenet with sampled parameters, but T times. T in this study is always set to 15, indicating that the set of parameters are sampled 15 times for every forward pass. Davis

J. et al. show that the total predictive variance can then be decomposed into aleatoric and epistemic uncertainty using the law of variance [9].

$$V(y|x) = \underbrace{V(E[y|x,\Theta])}_{\text{epistemic}} + \underbrace{E[V(y|x,\Theta)]}_{\text{aleatoric}}$$

where y is model output, x is model input, and θ is model sampled parameters. $V(E[y|x,\Theta])$ is variance of predicted mean that captures the variance across the model's mean predicted output over the T passes. $E[V(y|x,\Theta)]$ is the mean of predicted variance that captures the average variance across each label output over the T passes. T is the number of times the model parameters are sampled and hence also represents the number of times an image is predicted by the model.
There are two ways of sampling defined in this study:

- Random Sampling: sampling all the model parameters T times for every forward pass. Meaning, during the first forward pass, the model sampled parameters will differ from the sampled parameters during the future forward passes.
- Fixed Sampling: sampling all the model parameters T times only once, and fix the samples for all forward passes. Meaning, during the first forward pass, the model sampled parameters will be the same as the model sampled parameters during all future forward passes.

Random sampling will capture the total model's uncertainty across different images. Fixed sampling does not capture the full model uncertainty as it only samples from the learnt distributions once, but comparing the uncertainties in different images can be considered more valid. That is because during fixed sampling, the images uncertainty is produced by the same model with the same set of parameters. Whereas in random sampling, the images uncertainties are produced by the same model but with different set of parameters. Therefore, all LRPs performed in this study are visualizing the uncertainties produced by a fixed sampled model.

Related Work

Kumar et al. propose a Bayesian CNN architecture and convolutional operations on the mean and variance [19]. The network is trained using Bayes By Backprop which approximates the true posterior using variational inference. They compare their proposal to frequentest inference and incorporate a measurement for both aleatoric and epistemic uncertainties and regularisation. They study the effect of those uncertainties mainly on the training accuracy of the model. The work in this thesis is built on top of their network architecture. The uncertainty, however, will be quantified differently.

Antoran et al. propose CLUE, a method of interpreting uncertainty estimates by searching for low uncertainty points of an input and answer the question "How should we change an input such that it makes our model less uncertain?"[3]. The authors use a generative model to ensure plausibility of the input configurations in the explanations. In their work, the authors state "bayesian neural networks are able to provide reliable uncertainty estimates together with their predictions" and they base their work off that. This research project's aim is to support evidence to this claim qualitatively.

Chai [5] investigates the contribution of individual features on predictive uncertainty, epistemic uncertainty, and aleatoric uncertainty in BNN. The author's approach measures the change in uncertainty when a known feature of an input is given as opposed to an unknown feature. The contribution to this study would be to investigate the effect of added noise in the dataset as opposed to without noise while targeting on not only feature level, but also pixel level.

Layer Wise Relevance Propagation is an algorithm used to visualize the contribution of nodes in each layer to the output node for each output node. The implementation of M.Böhle et al. is extended to visualize the uncertainty of BNNs in this thesis [2].

Approach

To test for reliability of the uncertainty quantified by the Bayesian Lenet, a set of hypothesis is used to examine whether the uncertainty quantified by the model behaves as expected, based on definitions and research, under a defined domain. This chapter discusses the domain, hypotheses, and the research phases of this study.

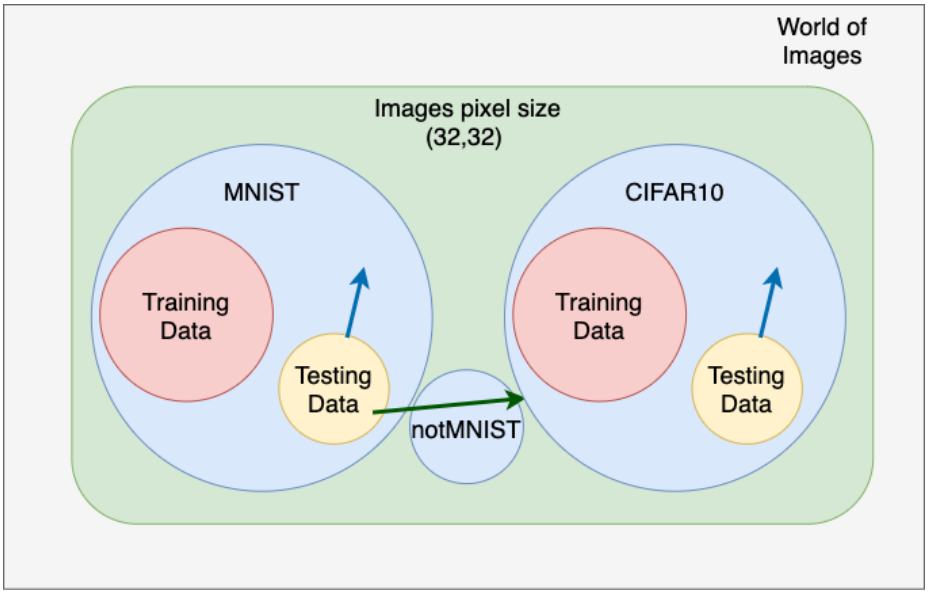

Figure 4.1: Illustration of research domain. The blue arrows display the data shifts that the aleatoric uncertainty is expected to capture. The green arrow displays the data shift that the epistemic uncertainty is expected to capture.

4.1 Research Domain

The Bayesian Lenet network can handle images of pixel size (32,32). Therefore, in our world of images, only images of pixel size (32,32) are representing the research. Within those images, there are two subsets namely MNIST and CIFAR10. Their training data subset consists of 10,000 images each and will be used to train the network under the same network configurations. The testing data subsets consists of 1,000 images each and will be manipulated to examine the reliability of the uncertainty quantified by the model. Given the aleatoric and epistemic uncertainties definitions discussed in section 2.3.1, the aleatoric will be tested by shifting both testing data subsets using random noise but still remain within the MNIST and CIFAR10 subset respectively. This shift is represented by the blue

arrow in figure 4.1. Whereas for epistemic, the model trained on MNIST will be tested on a small data shift, i.e. notMNIST dataset, and a large data shift, i.e. CIFAR10 testing data subset. This is represented by the green arrow in figure 4.1.

4.2 Hypotheses

4.2.1 Behaviour of Accuracy

Hypotheses 1,2 are used to study the behaviour of the model's prediction accuracy under small and large data shifts.

Hypothesis 1. Accuracy decreases with increasing noise levels

Hypothesis 2. The lower the accuracy the higher the uncertainty

In general, introducing such data shifts should degrade the model performance. There are however three possible scenarios:

1. The model is resilient to random noise and hence prediction accuracy is constant across different noise levels. In such scenario, studying the aleatoric uncertainty using data shifts of noise will be ineffective as it will remain stable across different noise levels.

2. The model prediction accuracy increases across different noise levels. This will be considered as irregular behaviour and the network's appropriateness for this study will be questioned.

3. The prediction accuracy decreases with increasing noise. This is the ideal scenario to continue the study only if the model accuracy does not drop less than chance level as soon as noise is introduced. Because otherwise, the model's uncertainties will be unstable to study under a guessing behaviour.

Therefore H1 is to ensure the model is reasonably affected by noise and hence can be used to study the behaviour of its uncertainty quantification under different noise levels. H2 is used to check for scenarios where the model's accuracy is decreasing and so does its uncertainty. This is an undesirable behaviour especially in safety critical applications where the goal is to have the model's uncertainty higher under lower prediction accuracy than higher prediction accuracy.

4.2.2 Behaviour of Aleatoric Uncertainty

Hypotheses 3, 4, 5 are used to study the behaviour of aleatoric uncertainty under small and large data shifts.

Hypothesis 3. Aleatoric uncertainty increases with increasing independent noise levels

Hypothesis 4. Aleatoric uncertainty increases with increasing box size for dependent noise

Hypothesis 5. Aleatoric uncertainty is insignificantly affected by large data shifts

In H3, independent noise levels are noise per pixel values sampled from a distribution. In this study, two distributions are used namely Gaussian and Poisson. For Gaussian noise, for each σ in the range $(0, 1)$ with stepsize 0.05, each pixel value in the original image is added to a noise value sampled from a normal distribution of $\mu = 0$ and σ.

$$\begin{aligned} \sigma &= \{0.05k | k \in \{0, .., 1\}\} \\ noise_{(n,i,j)} &\sim N(0, \sigma_n) \qquad \forall n \in \{0, ..., |\sigma|\}, \forall i, j \in \{0, .., 31\} \\ & \qquad\qquad noise_n \in \mathbb{R}^{32\times 32} \\ img &= img + noise_n \qquad img \in \mathbb{R}^{32\times 32} \end{aligned}$$

For Poisson noise, each pixel noise value is sampled from a Poisson distribution with a rate value sampled from a Gaussian distribution with $mu = 0$ for each sigma value in range $(0, 1)$ with step size 0.05.

$$\begin{aligned} \sigma &= \{0.05k | k \in \{0, .., 1\}\} \\ \lambda_{(n,i,j)} &\sim N(0, \sigma_n) \qquad \forall n \in \{0, ..., |\sigma|\}, \forall i, j \in \{0, .., 31\} \\ & \qquad\qquad \lambda_n \in \mathbb{R}^{32\times 32} \\ noise_n &\sim Poisson(\lambda_{(n)}) \qquad noise_n \in \mathbb{R}^{32\times 32} \\ img &= img + noise_n \qquad img \in \mathbb{R}^{32\times 32} \end{aligned}$$

In H4, dependent noise refers to masking input with boxes in random locations for each box size in the range [10,32] with step size 3. Figure 4.2 shows examples of an image with different noise types and levels. In H5, since aleatoric is about inherent noise in the image and epistemic is about the lack of knowledge of the model, it is expected that introducing a dataset the model is not trained to classify should be captured by the epistemic and not the aleatoric. Therefore, the aleatoric uncertainty is expected to be insignificantly affected by such data shifts.

4.2.3 Behaviour of Epistemic Uncertainty

Hypotheses 6 and 7 are used to study the behaviour of epistemic uncertainty under small and large data shifts.

Hypothesis 6. Epistemic uncertainty is insignificantly affected by noise

Hypothesis 7. Epistemic uncertainty increases with large data shifts

In H6, noise refers to the independent and dependent noise explained in section 4.2.2. Since the aleatoric is expected to capture such noise, the epistemic uncertainty is expected to be insignificantly affected by it. In H7, large data shifts imply the model predicting on a dataset it is not trained to classify. According to the epistemic uncertainty definition discussed in section 2.3.1, the epistemic should be able to capture the data shift and hence increase.

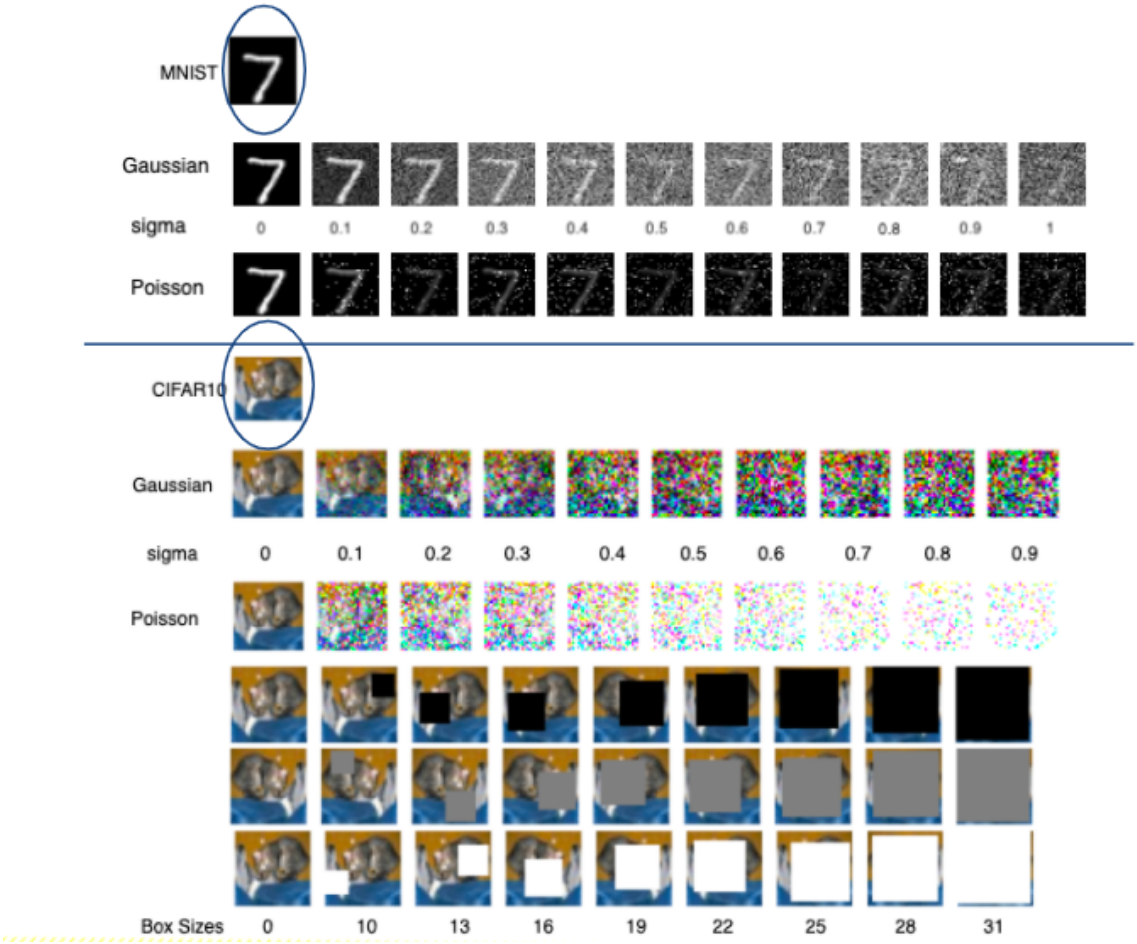

Figure 4.2: Example of imaged with Gaussian and Poisson noise distributions, and input masking

4.3 Pipeline

There are three main phases in this project (Fig.4.3):

- phase 1: Train the Bayesian Lenet on the training dataset
- phase 2: Pass altered test dataset to the network for prediction and uncertainty quantification
- phase 3: Evaluate uncertainties quantified using the seven hypotheses discussed in section 4.2

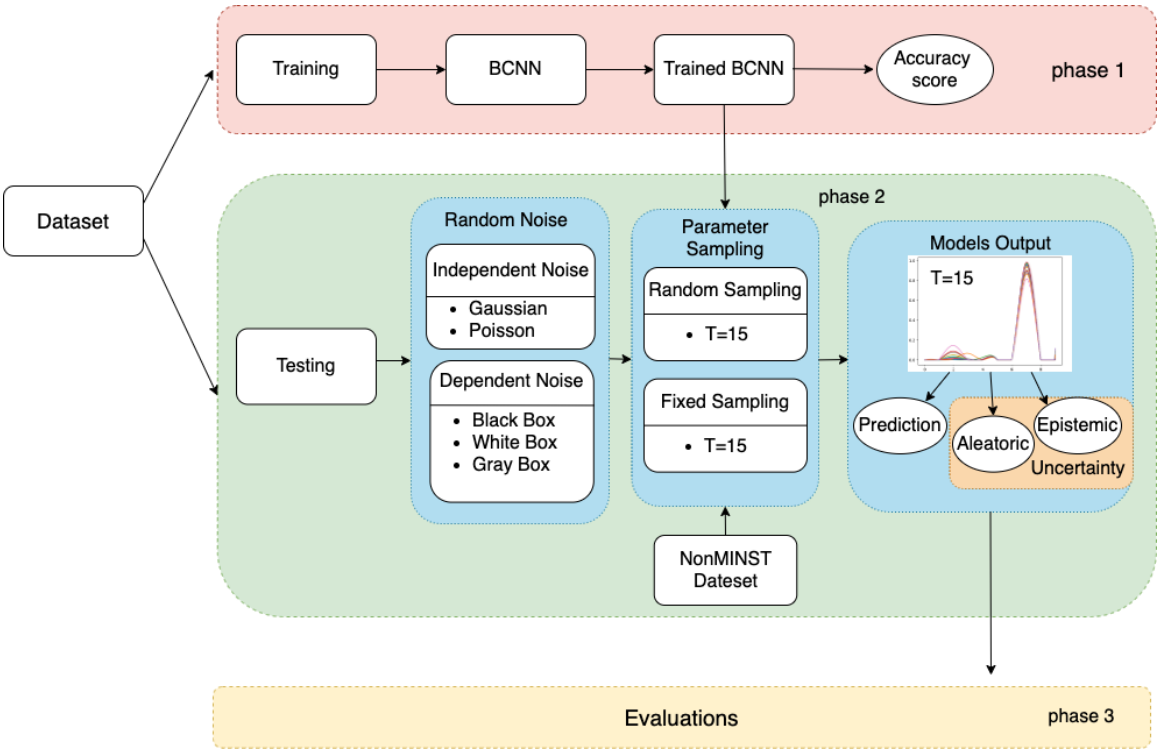

Figure 4.3: Three phases of the research plan. Phase 1 is training the Bayesian Lenet. Phase 2 is shifting the test dataset. Phase 3 is evaluating the behaviour of the aleatoric and epistemic under the data shifts.

4.3.1 Phase 1: Training

In this phase, two different models are trained. One model is trained on MNIST (an easy dataset), and another on CIFAR10 (a difficult dataset). Both models were trained under the same conditions, ie same training dataset size, same number of epochs, same network architecture including priors, etc. The model learnt the MNIST dataset much better than the CIFAR10 with a training accuracy score of 98.9% and 64% respectively, as shown in figure 4.4.

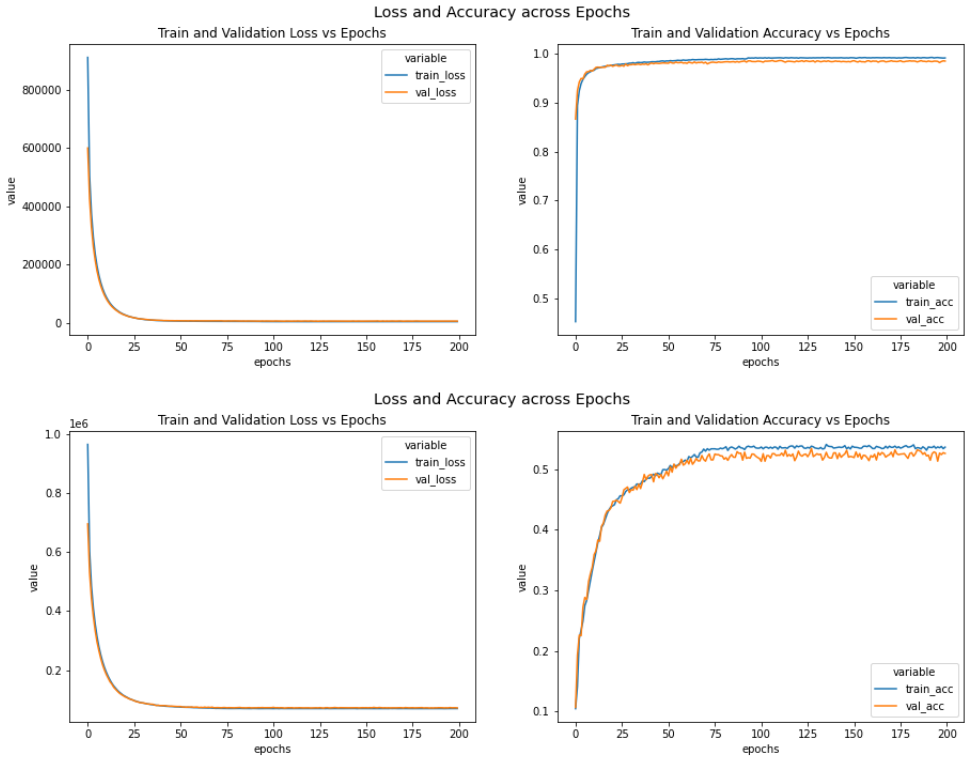

Figure 4.4: Training performance of model on MNIST and CIFAR10 respectively

4.3.2 Phase 2: Data Shifting

To examine aleatoric uncertainty, independent noise, namely Gaussian and Poisson, and dependent noise, box covering features, are added to the testing dataset of both MNIST and CIFAR10. To examine epistemic, notMNIST and CIFAR10 are passed as testing dataset to the model trained on MNIST10. notMNIST and CIFAR10 represent a small data shift and large data shift respectively, as shown in Fig. 4.5.

Figure 4.5: Example images of MNIST, notMNIST, and CIFAR10

4.3.3 Phase 3: Evaluation

After phase 2, the accuracy, aleatoric uncertainty, and epistemic uncertainty will be evaluated based on the seven hypotheses discussed in section 4.2. To deepen the understanding of where the model captured aleatoric and epistemic uncertainty in the image, LRP is used

to visualize them. An example with how to interpret the plots is discussed in the next subsection.

Visualizing Uncertainty

To further aid the examination, LRP technique is used to visualize the uncertainties captured in an image. An example in Fig. 4.3.3 is shown, where a Bayesian Lenet model trained on CIFAR10 predicts an image of a ship. The model predicted it as a car with prediction accuracy of 39.24, aleatoric uncertainty of 0.010049, and epistemic uncertainty of 0.010049. The uncertainty values on their own are of no use, they only hold a meaning when compared to the uncertainty values of other images. In CIFAR10, there are 10 output labels. *Evd.[label]*, the evidence of output label value, visualizes the features that contributed to this label node. In the ship example, *EVD. car* subplot shows the features that contributed most to its predicted output. Across the 10 labels, no features contributed to the output labels of animals. Some features contributed to the output values in vehicle label, more in car and ship than in truck and plane. *Pred. Evd.*, prediction evidence, is the evidence of predicted output label value overlaid with uncertainty value for each pixel. This means the brighter the pixel is, the more the variation in its output value indicating a higher uncertainty value in that pixel than less bright pixels. In the ship example, the *Pred. Evd.* demonstrates how the aleatoric uncertainty is more captured in the pixels around the ship, whereas the epistemic uncertainty was more captured in the pixels of the ship itself.

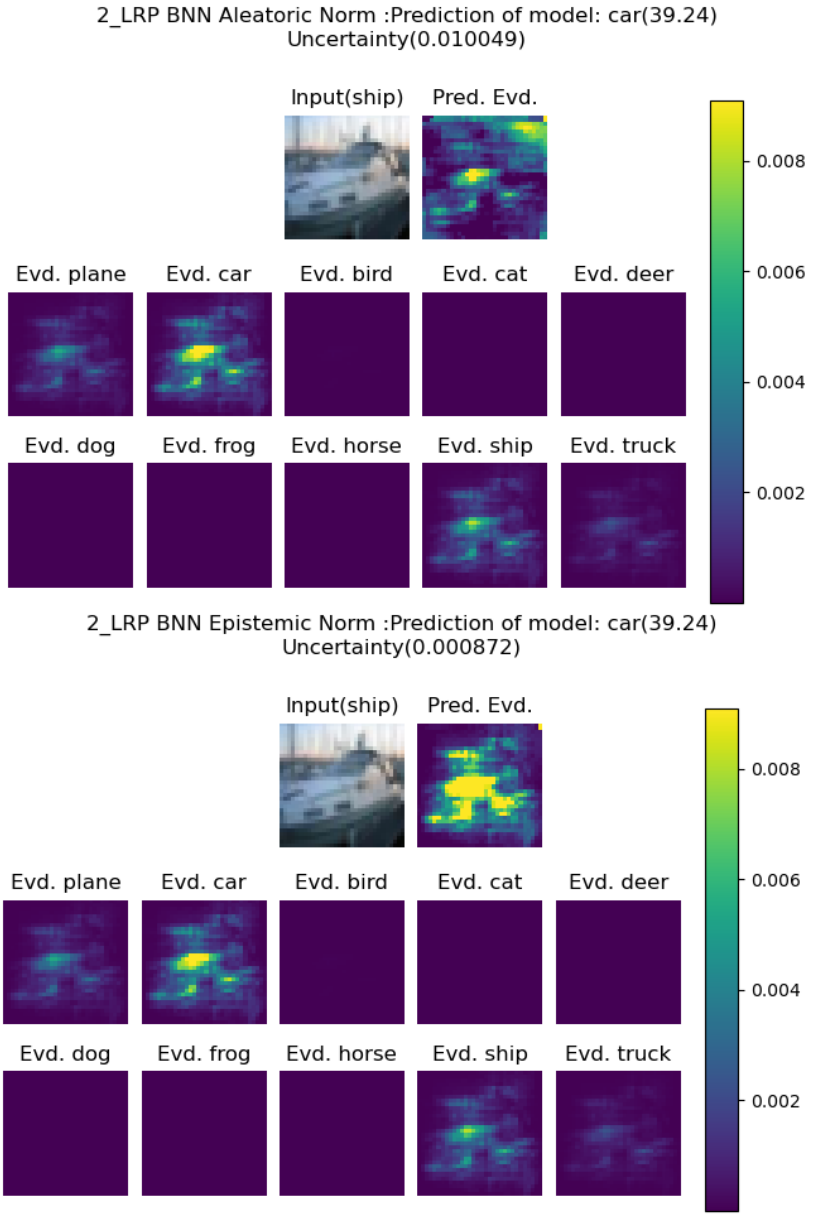

Figure 4.6: Visualization example of aleatoric and epistemic uncertainty of Bayesian Lenet trained on CIFAR10 when tested with an image of a ship.

Evaluation

This section discusses the results of the evaluation phase, i.e. phase 3. The model's prediction accuracy, aleatoric uncertainty, and epistemic uncertainty are studied under data shifts and evaluated qualitatively to verify the hypotheses in section 4.2.

5.1 Metrics

In the evaluation phase, three metrics are studied:

- Accuracy
- Aleatoric Uncertainty
- Epistemic Uncertainty

The reliability of the model will be based on the evaluation of those three metrics under the following hypotheses:

H1: Accuracy decreases with increasing noise levels

H2: The lower the accuracy the higher the uncertainty

H3: Aleatoric uncertainty increases with increasing independent noise levels

H4: Aleatoric uncertainty increases with increasing box size for dependent noise

H5: Aleatoric uncertainty is insignificantly affected by large data shifts

H6: Epistemic uncertainty is insignificantly affected by noise

H7: Epistemic uncertainty increases with large data shifts

5.2 Results

This section details the qualitative analysis that evaluates the hypotheses towards answering the research question.

5.2.1 Behaviour of Accuracy

Results of hypotheses 1 and 2 are discussed below.

H1: Accuracy decreases with increasing noise levels

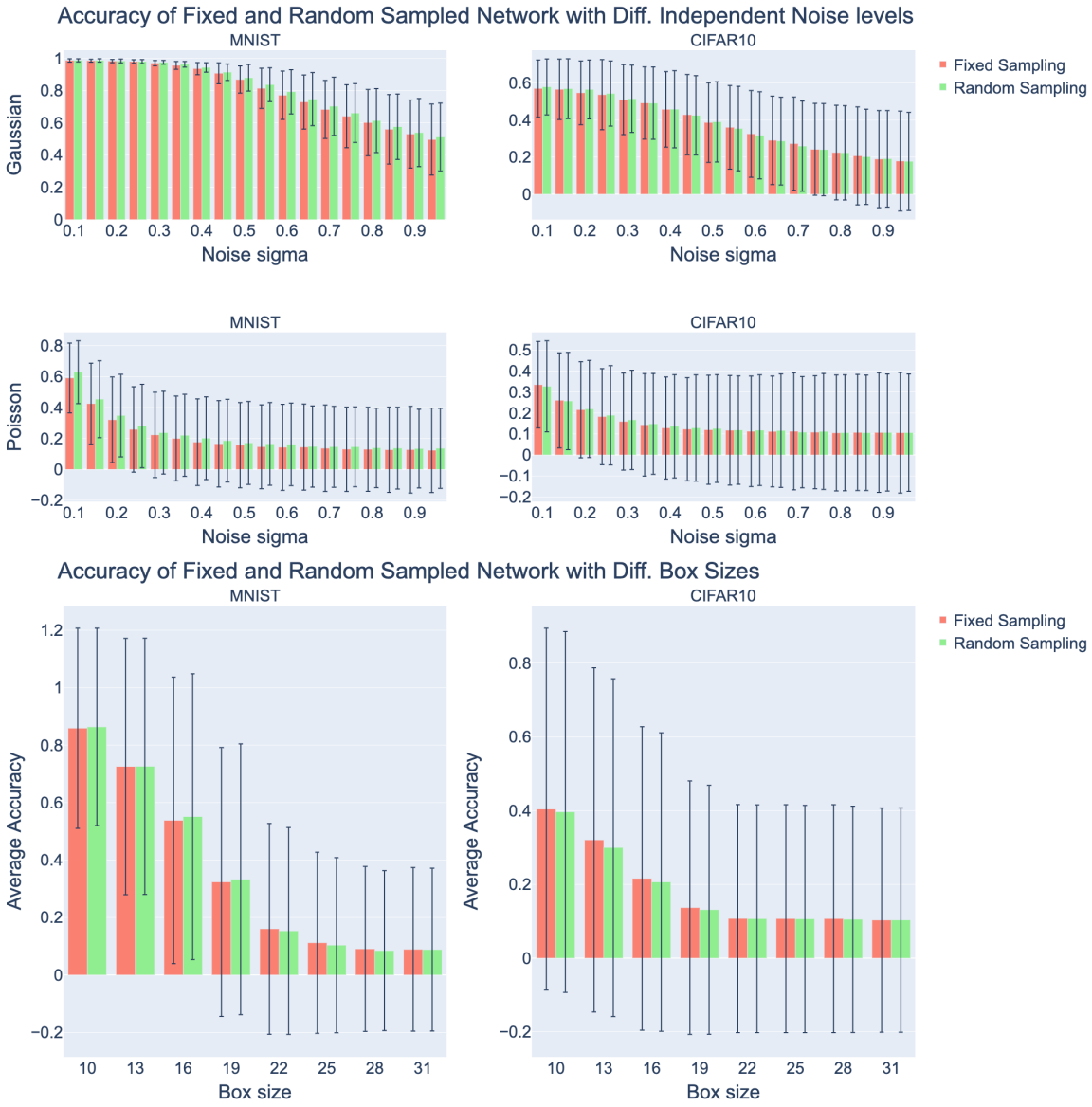

Figure 5.1: Accuracy vs independent and dependent noise for fixed and random sampling techniques and the both datasets MNIST and CIFAR10

As seen in Fig. 5.1, there is no significant difference in accuracy between random and fixed sampling. It would be expected that the fixed sampling will result in a lower accuracy on average due to fixating the model parameters after the first sample, and hence not being as versatile anymore. However, the ensemble sampled from both sampling techniques predicts the same output but outputs different uncertainties. This can imply that there is a low standard deviation over the learnt parameter distributions, and hence sampling once is as effective as re-sampling for every forward pass. However, this claim is a hypothesis that requires further investigation and so left as future work.

Average accuracy of images with Poisson noise is lower than images with Gaussian, and

drops quicker across increasing noise levels. That is because the model's parameters are sampled from Gaussian distributions (future work hypothesis). That means the model is trained to capture and account for Gaussian noise in the training dataset. Given that the test datasets tend to have extreme noise levels with high noise levels, the model performed still slightly better than chance with Gaussian noise regardless of the sampling technique.

The standard error represents the deviation in accuracy across different images. At the beginning this error was low with low noise levels and increased across higher noise levels. This means at a higher noise levels, some images were predicted better than others in presence of random noise.

Accuracy of the CIFAR(64%) dataset is drastically lower than MNIST(98%). This is expected since the model's training performance score was worse when trained on CIFAR10 than on MNIST.

To sum up, accuracy does decrease with increasing noise. Since the model seems to be affected by the noise, it is valid to study the behaviour of the uncertainties under different noise levels. Gaussian noise accuracy is higher than Poisson noise can be the effect of the model being more resilient to the type of noise the model parameters' distribution is based on, but more extensive evaluation for proving this claim is future work.

H2: The lower the accuracy the higher the uncertainty

There are case scenarios in which the the model has low uncertainty despite being wrong. This can be observed in Fig. 5.2 where the epistemic uncertainty increases across increasing noise levels, where as the accuracy across noise levels decrease in Fig. 5.1. On top of that, the standard error decreases at high noise levels. Those cases are a limitation to the reliability of the uncertainties quantified by the model specially in safety critical applications. To further investigate why the standard error is low for low accuracies, the predicted labels of the network for wrong decisions are plotted. The model had to predict on notMNIST and CIFAR10 despite being trained on MNIST only. It was observed that the model seems to have an "escape code". Meaning, the model always picks the same label when in doubt (shown and explained in A.3). It could be that the model could not learn that label during training and so resembled it as noise. Therefore, the images it cannot predict are treated as noise and hence correlated with that label. Further investigation to see the average prediction accuracy of that label during training is required to confirm this.

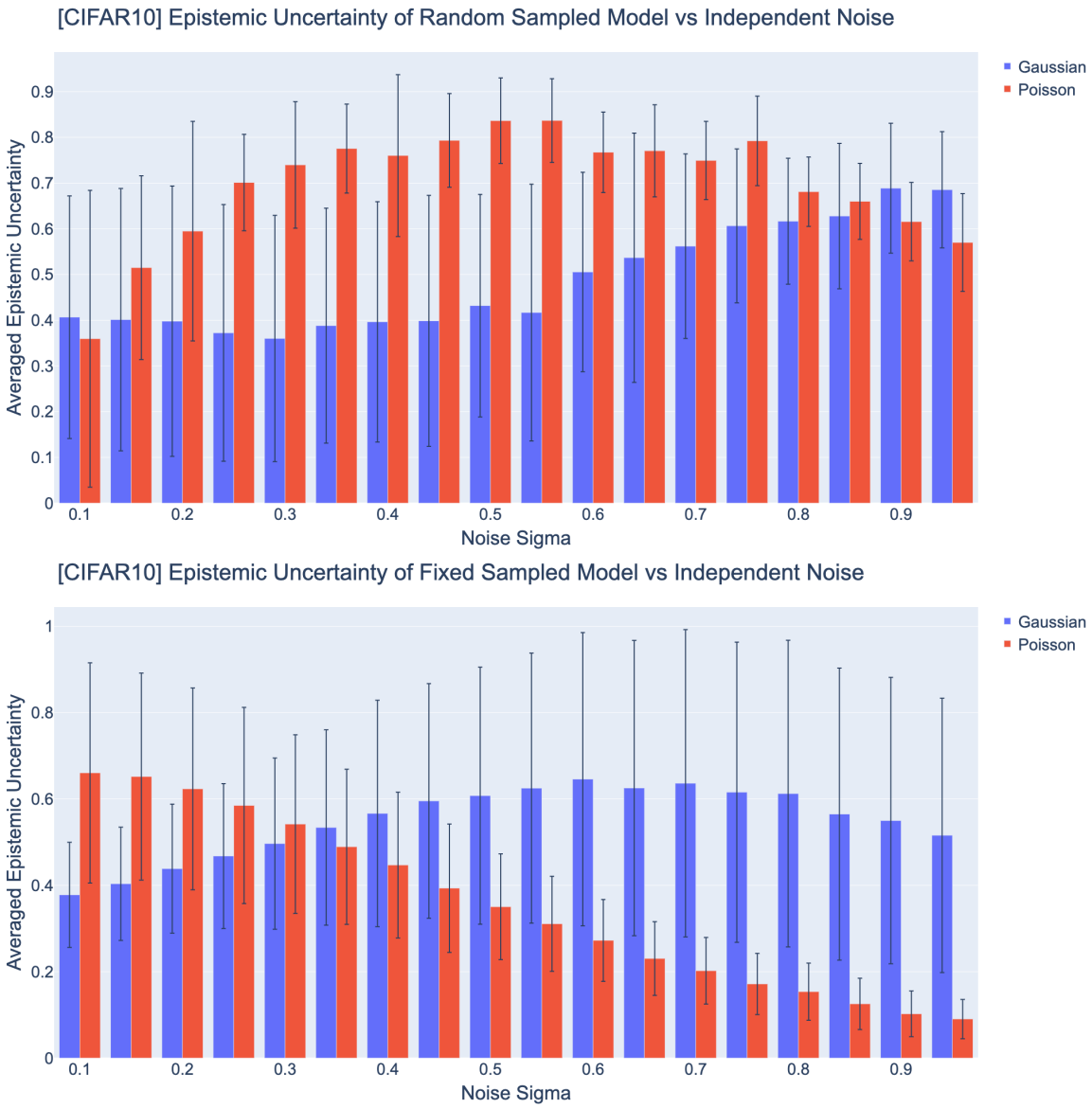

Figure 5.2: Epistemic uncertainty across noise levels for CIFAR10, random and fixed sampling. There is a decreasing behaviour in the uncertainty despite a decrease in accuracy shown in figure 5.1

5.2.2 Behaviour of Aleatoric Uncertainty

H3: Aleatoric uncertainty increases with increasing independent noise levels

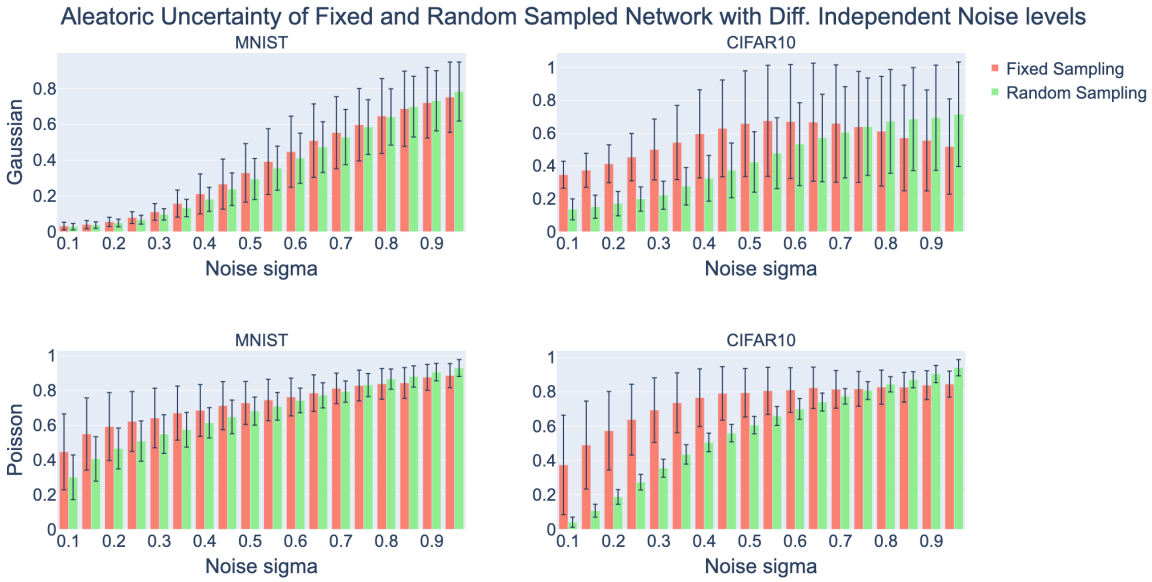

Figure 5.3: Aleatoric uncertainty vs independent noise levels for random,fixed sampling, and both datasets MNIST and CIFAR10

This hypothesis is true except for one case scenario of passing CIFAR10 with Gaussian noise to a model that underwent fixed sampling. For that scenario, the Gaussian uncertainty seemed to increase until noise level 0.7 (65%) then decreased slightly again to 52%. This is the same corner case as mentioned above (section 5.2.1) where even though the accuracy noise levels 0.7 and higher decreased (30% to 18%), the uncertainty decreased as well.

Overall, the aleatoric uncertainty displays a proper behaviour of increasing with increasing noise levels for random sampling techniques for both datasets. On the other hand, it behaved differently when the model parameter sampling is fixated to the first sample (Fixed sampling technique). The uncertainty decreased with noise sigmas > 0.7 for CIFAR10.

H4: Aleatoric uncertainty increases with increasing box size for dependent noise

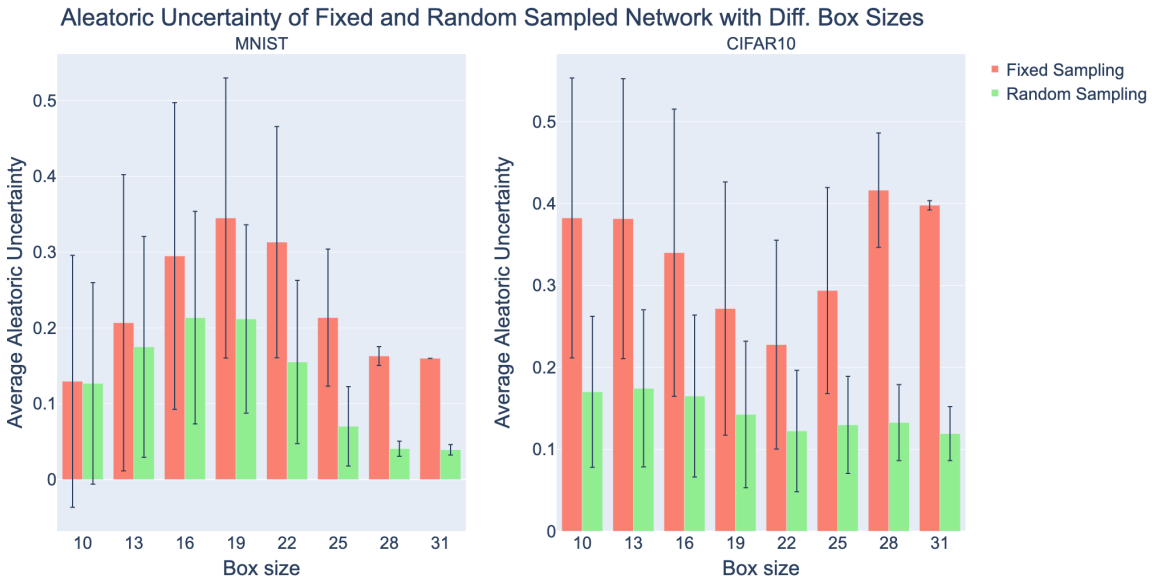

Figure 5.4: Aleatoric uncertainty vs dependent noise levels for random,fixed sampling, and both datasets of MNIST and CIFAR10

This hypothesis is shown in Fig. 5.4 to be wrong. Aleatoric doesnt have a defined pattern with increased box size. It sometimes increases then decreases like in MNIST dataset with random sampling, and in others decreases then increases like in Cifar10 dataset with fixed sampling.

Overall there is no significant difference between the three colours on the aleatoric uncertainty except for the gray box having a higher aleatoric uncertainty average in CIFAR10 fixed (see appendix A.2). This was also the case for upcoming hypothesis, therefore only black boxes are used for evaluation. A better defined priors can result in a better trained network. Studying that effect is left as future work.

Aleatoric uncertainty in this case is dependent on which features are covered and which features are not. This can be seen through the LRPs in Fig. 5.5. The same input image with box size (22,22) in different location resulted in different model prediction and aleatoric uncertainty values.

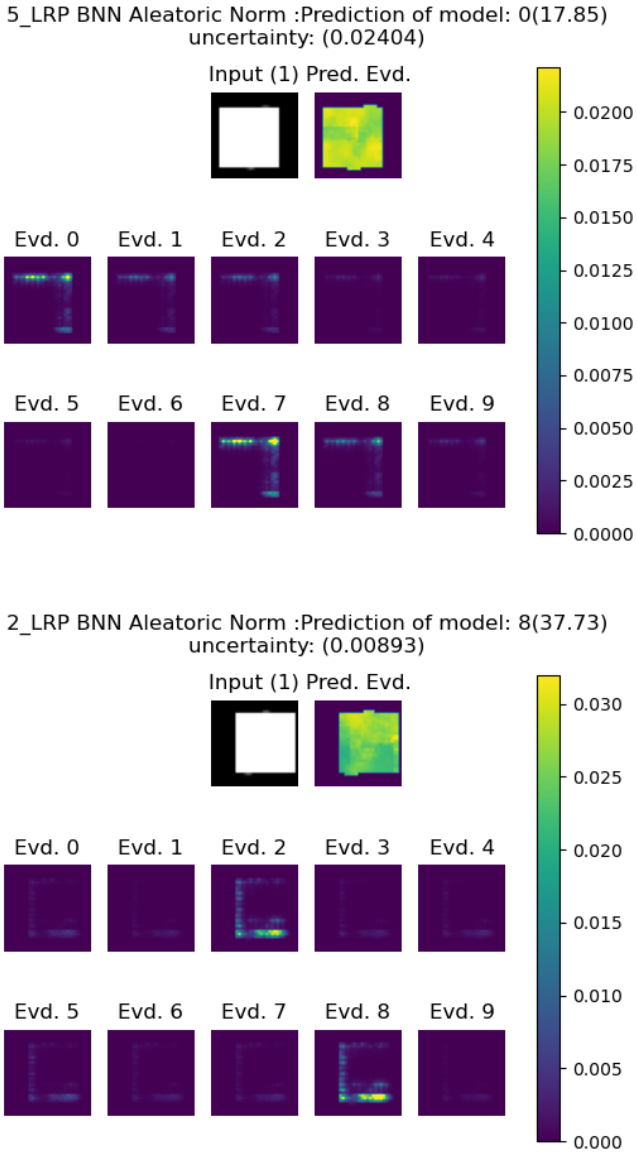

Figure 5.5: LRPs of two similar input images with boxes of same size in different locations resulting in different model predictions and aleatoric uncertainty values

H5: Aleatoric uncertainty is insignificantly affected by large data shifts

This hypothesis is shown to be incorrect in Fig. 5.6. Passing notMNIST to model trained on MNIST resulted in average aleatoric uncertainty higher than MNIST. And passing Cifar10 to model trained on MNIST also resulted in a higher average aleatoric uncertainty than MNIST, but lower than nonMNIST. This means aleatoric treats smaller data shifts (notMNIST) as noise and is less affected by larger data shifts (CIFAR10) in random sampling. But treats both data shifts as noise in fixed sampling (can be seen in the LRP visualization in 5.7).

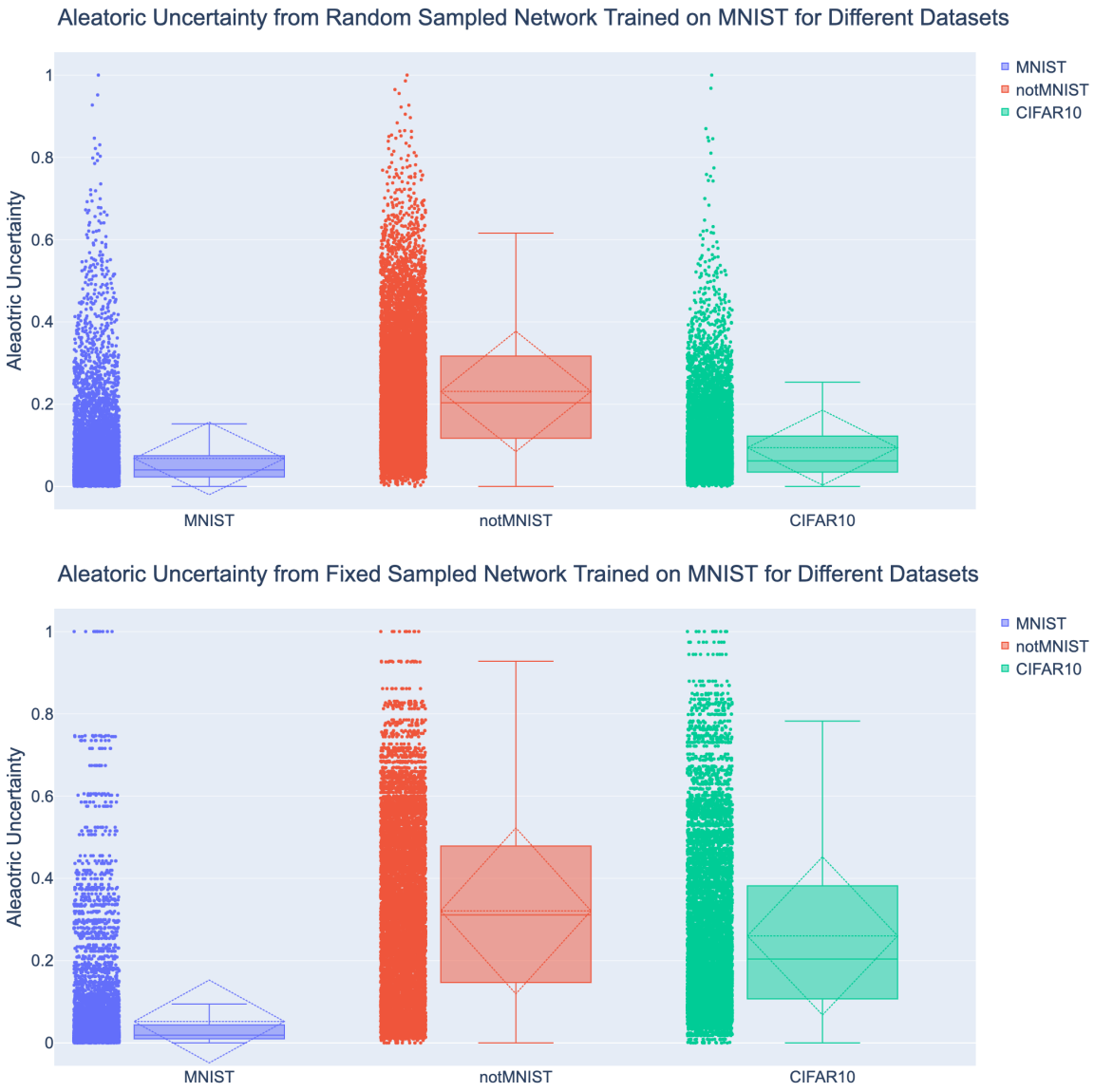

Figure 5.6: Aleatoric uncertainty vs data shifts

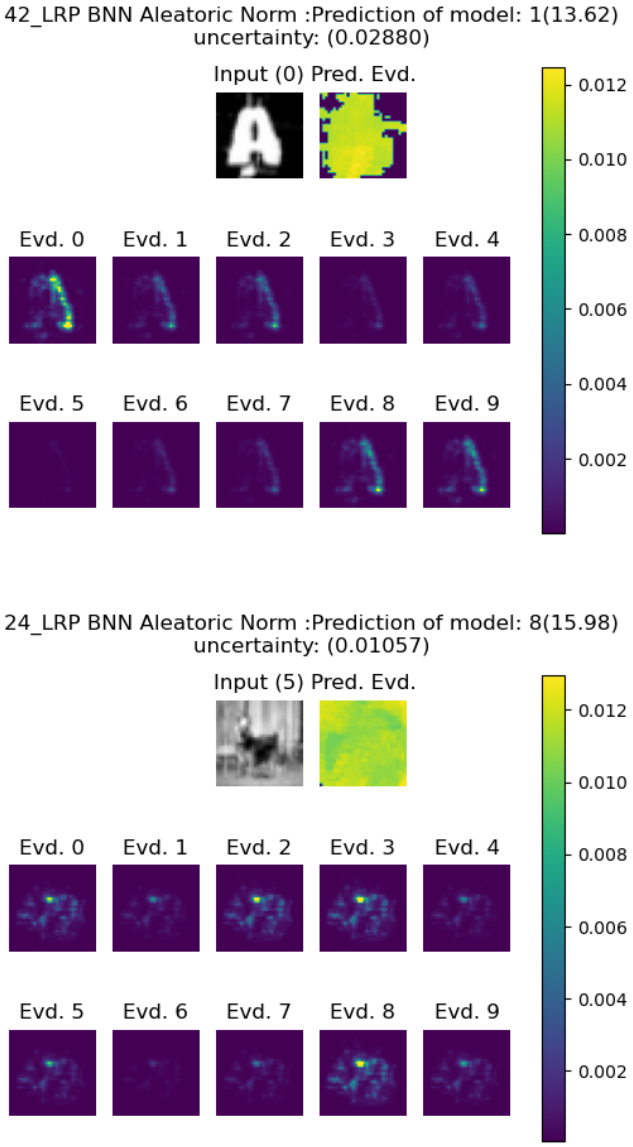

Figure 5.7: The model with fixed sampling treats the unknown features as noise

5.2.3 Behaviour of Epistemic Uncertainty

H6: Epistemic uncertainty is insignificantly affected by noise

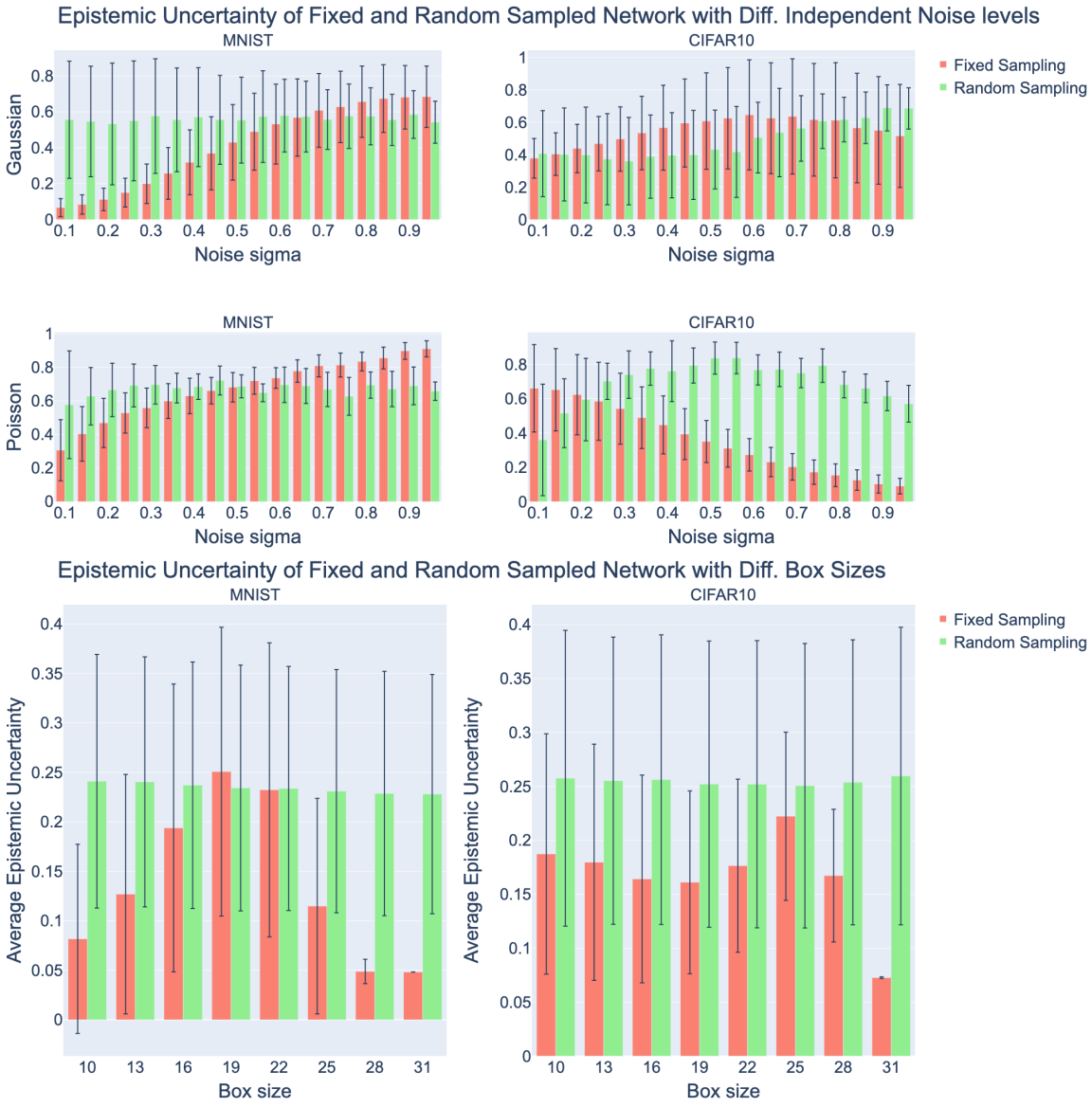

Figure 5.8: Epistemic uncertainty vs dependent noise for random and fixed sampling, and for both datasets MNIST and CIFAR10

For independent noise, this hypothesis holds for MNIST images under independent noise under random sampling. It does not hold for the CIFAR10 images and Fixed sampling on MNIST images, as shown in Fig. 5.8.

For dependent noise, the hypothesis only holds for random sampling in both datasets, as shown in Fig. 5.8.

Overall, for random sampling H6 holds except for CIFAR10 dataset. For fixed sampling, it does not hold at all. For CIFAR10 dataset, the training accuracy is low which means

the model did not capture enough knowledge about it. Hence, the epistemic level on its own is higher than MNIST dataset where the accuracy of training is higher. Therefore, an undefined behaviour in CIFAR10 is observed. This means, having a training accuracy and standard deviation of learnt parameters' distributions above a certain threshold (finding that threshold is future work) is needed for stable uncertainty quantification that is less based on a guessing network.

H7: Epistemic uncertainty increases with large data shifts

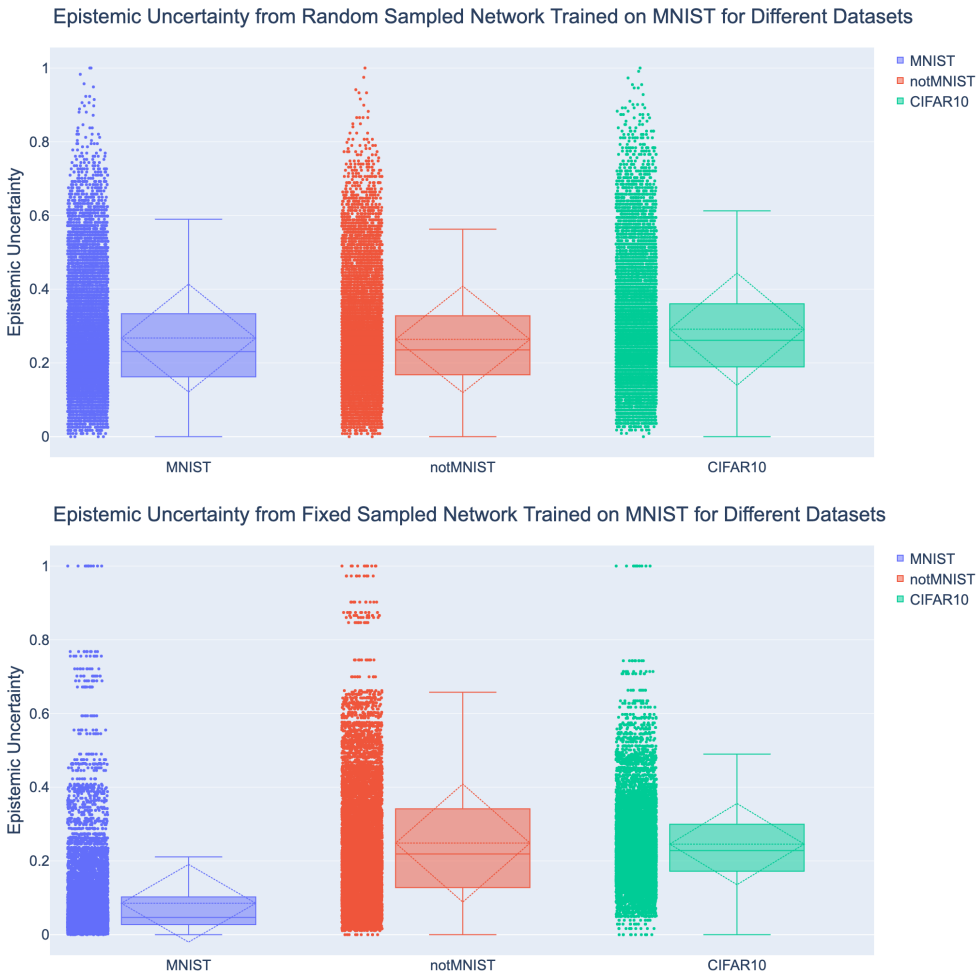

Figure 5.9: Epistemic uncertainty vs data shifts for random and fixed sampling, and for both datasets MNIST and CIFAR10

With Random sampling, the epistemic uncertainty is slightly affected by the data shift. Notmnist epistemic uncertainty is similar to MNIST since it was captured as noise by aleatoric due to small data shift. Whereas, CIFAR10 has slightly higher epistemic uncertainty than MNIST due to a larger data shift.

With Fixed sampling, the epistemic is affected by the datashift. However, it seems to pick up the CIFAR10 dataset features as well as the notMNISt with similar average uncertainty.

Conclusion

BNNs allow more versatility and less overly confident decisions by capturing uncertainty in the training data, than when using DNNs. Having the parameters sampled from a distribution for every forward pass gives the opportunity to quantify the uncertainty it captured during training. This thesis aims to understand and interpret an uncertainty quantification method, and evaluate its reliability by studying its behaviour under shifted data.

Overall, uncertainty quantification method is able to detect small and large data shifts but the model does not quantify the uncertainty reliably as hypothesis 4,5 do not hold and 2,3,6,7 partially holds. However, the results demonstrates that to have more reliable uncertainty quantification, a well trained model is needed, and random sampling is preferred over fixed sampling for calculating uncertainty due to it capturing more of the uncertainty the model captured during training. Aleatoric and epistemic uncertainty seem to overlap when independent or dependent noise is introduced, which requires further investigation. This was already observed and indicated in previous research [18, 22]. To avoid such overlap, calculating uncertainty with a closed form solution during the forward pass instead of estimating it with sampling techniques can be investigated. As well as this, there is a resemblance observed between the model being more resilient to noise distributions that the model parameter kernel is based on, which requires further investigation. To examine that, the same pipeline can be used for a network that uses Poisson kernels. Then images with Gaussian noise should have on average lower accuracy score than images with Poisson noise. Another examination is required to study whether a training accuracy and standard deviation of learnt parameters' distributions above a certain threshold is needed for a reliable uncertainty quantification. In addition, ways to compute that threshold.

In general, it is observed that the reliability of the uncertainty quantification methods used is dependent on many factors, such as, the model's achieved training accuracy, the model parameters' sampling technique, dataset, and noise type. Therefore, the uncertainty quantification method used is not ready to be used in safety critical applications since further investigations are required to understand its behaviour. The pipeline introduced in this work can be used to further asses other uncertainty quantification methods and/or different network configurations.

Bibliography

[1] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra, *Weight uncertainty in neural network*, International Conference on Machine Learning, PMLR, 2015, pp. 1613–1622.

[2] Moritz Böhle, Fabian Eitel, Martin Weygandt, and Kerstin Ritter, *Layer-wise relevance propagation for explaining deep neural network decisions in mri-based alzheimer's disease classification*, Frontiers in aging neuroscience **11** (2019), 194.

[3] Javier Antorán Cabiscol, *Understanding uncertainty in bayesian neural networks*, Master of Philosophy (University of Cambridge) (2019).

[4] Ginevra Carbone, Matthew Wicker, Luca Laurenti, Andrea Patane, Luca Bortolussi, and Guido Sanguinetti, *Robustness of bayesian neural networks to gradient-based attacks*, arXiv preprint arXiv:2002.04359 (2020).

[5] Lucy R Chai, *Uncertainty estimation in bayesian neural networks and links to interpretability*, Master of Philosophy (University of Cambridge) (2018).

[6] Joan Claybrook and Shaun Kildare, *Autonomous vehicles: No driver... no regulation?*, Science **361** (2018), no. 6397, 36–37.

[7] Arden Dertat, *Applied deep learning - part 4: Convolutional neural networks*, (2017).

[8] Arden Dertat, *Applied deep learning-part 4: convolutional neural networks*, Toward DataScience,[Online]. Available: https://towardsdatascience. com/applied-deeplearning-part-4-convolutional-neural-networks-584bc134c1e2 (2017).

[9] Davis J. et al., *Quantifying uncertainty in deep learning systems*, AWS Professional Services (2020).

[10] CG Lee X Wang M Rong G Li, L Yang, *A bayesian deep learning rul framework integrating epistemic and aleatoric uncertainties*, IEEE Transactions on Industrial Electronics (2020).

[11] Alex Graves, *Practical variational inference for neural networks*, Advances in neural information processing systems, Citeseer, 2011, pp. 2348–2356.

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

[13] Eyke Hüllermeier and Willem Waegeman, *Aleatoric and epistemic uncertainty in machine learning: A tutorialintroduction*, CoRR **abs/1910.09457** (2019).

[14] Eyke Hüllermeier and Willem Waegeman, *Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods*, Machine Learning **110.3** (2021), 457–506.

[15] P. Karkare, *Convolutional neural networks simplified*, (2019).

[16] Alex Kendall and Yarin Gal., *What uncertainties do we need in bayesian deep learning for computer vision?*, arXiv preprint arXiv:1703.04977 (2017).

[17] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner, *Gradient-based learning applied to document recognition*, Proceedings of the IEEE **86** (1998), no. 11, 2278–2324.

[18] Robin Senge, Stefan Bösner, Krzysztof Dembczyński, Jörg Haasenritter, Oliver Hirsch, Norbert Donner-Banzhoff, and Eyke Hüllermeier, *Reliable classification: Learning classifiers that distinguish aleatoric and epistemic uncertainty*, Information Sciences **255** (2014), 16–29.

[19] Kumar Shridhar, Felix Laumann, and Marcus Liwicki, *A comprehensive guide to bayesian convolutional neural network with variational inference*, arXiv preprint arXiv:1901.02731 (2019).

[20] Karen Simonyan and Andrew Zisserman, *Very deep convolutional networks for large-scale image recognition*, arXiv preprint arXiv:1409.1556 (2014).

[21] et al. Stahl, Niclas, *Evaluation of uncertainty quantification in deep learning*, International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (2020).

[22] Niclas Ståhl, Göran Falkman, Alexander Karlsson, and Gunnar Mathiason, *Evaluation of uncertainty quantification in deep learning*, International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, Springer, 2020, pp. 556–568.

Appendix

A.1 Dependent Noise

The figure below demonstrates how the behaviour of the uncertainties do not differ across boxes with different colours.

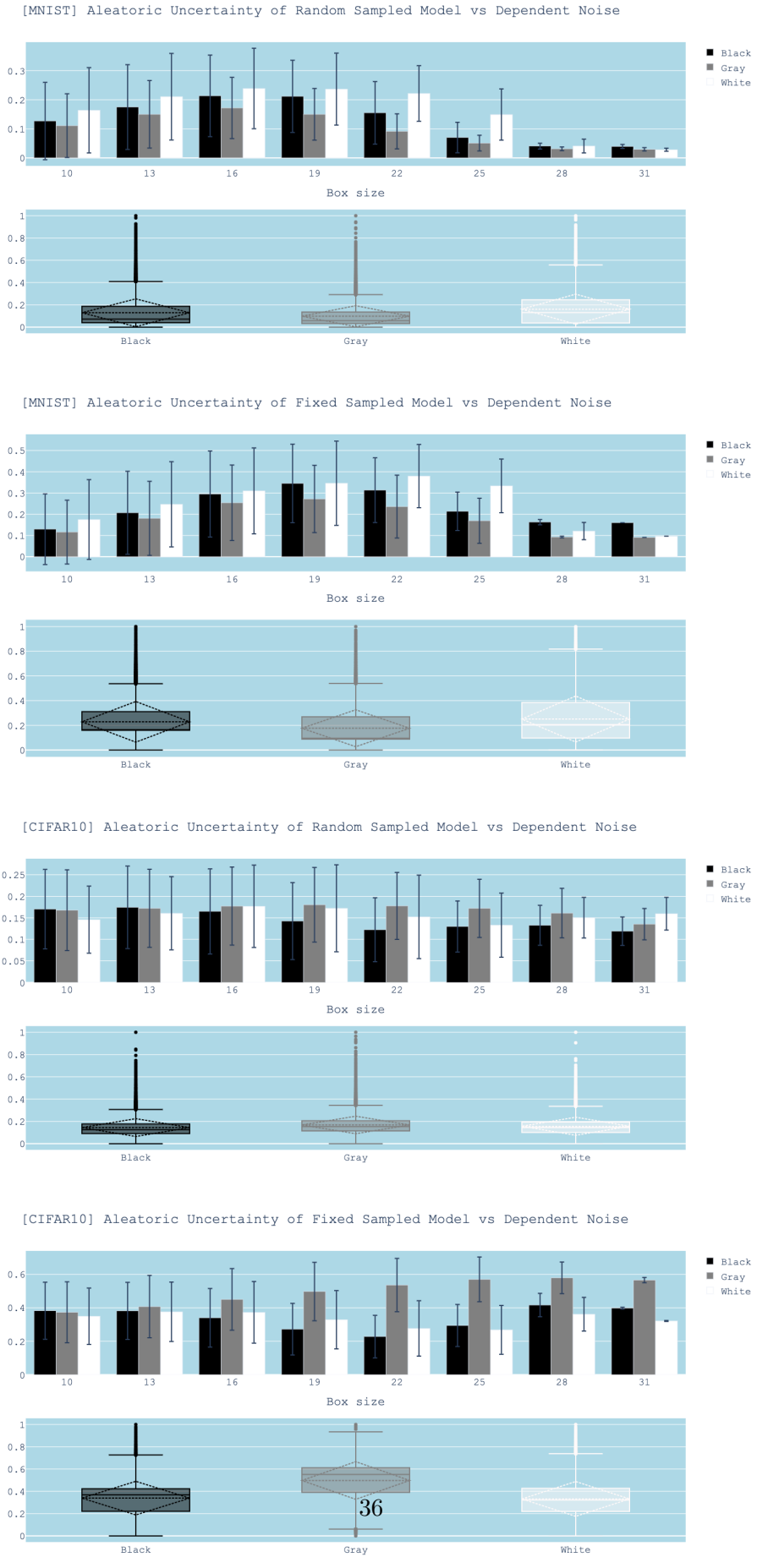

Figure A.1: Aleatoric uncertainty vs dependent noise for different box colours

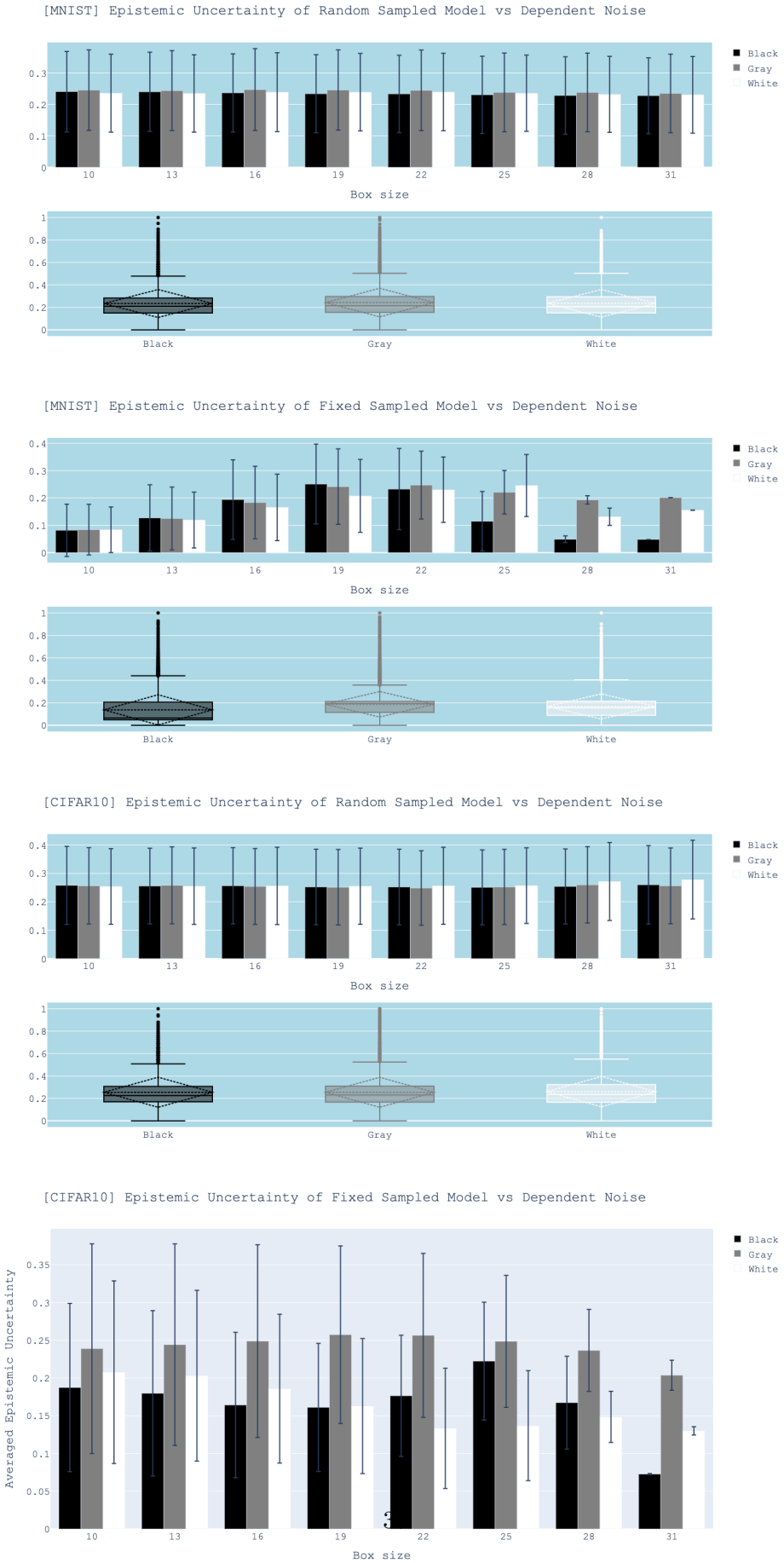

Figure A.2: Epistemic uncertainty vs dependent noise for different box colours

A.2 Escape Code

The model trained on MNISt predicts in notMNIST and CIFAR10. It is observed that the model significantly uses one label more than the rest.

Figure A.3: Labels 4 and 8 were mostly predicted labels when the model was incorrectly classifying.