4th Symposium on Advances in Approximate Bayesian Inference, 2021 1–9

Bayesian OOD detection with aleatoric uncertainty and outlier exposure

Anonymous Authors
Anonymous Institution

Abstract

Typical Bayesian approaches to OOD detection use epistemic uncertainty. Surprisingly from the Bayesian perspective, there are a number of methods that successfully use *aleatoric* uncertainty to detect OOD points (e.g. Hendryks et al. 2018). In addition, it is difficult to use outlier exposure to improve a Bayesian OOD detection model, as it is not clear whether it is possible or desirable to increase posterior (epistemic) uncertainty at outlier points. We show that a generative model of data curation provides a principled account of aleatoric uncertainty for OOD detection. In particular, aleatoric uncertainty signals a specific type of OOD point: one without a well-defined class-label, and our model of data curation gives a likelihood for these points, giving us a mechanism for conditioning on outlier points and thus performing principled Bayesian outlier exposure. Our principled Bayesian approach, combining aleatoric and epistemic uncertainty with outlier exposure performs better than methods using aleatoric or epistemic alone.

1. Introduction

The most typical approach to Bayesian OOD distribution detection uses epistemic uncertainty (Lakshminarayanan et al., 2017; Malinin and Gales, 2018; Choi et al., 2018; Wen et al., 2019; Malinin et al., 2020; Postels et al., 2020). We have epistemic uncertainty when finite training data fails to pin down the classifier's ideal outputs in all regions of the input space (Der Kiureghian and Ditlevsen, 2009; Fox and Ülkümen, 2011; Kendall and Gal, 2017). Importantly, the amount of epistemic uncertainty will vary depending on how close a given test point is to the training data. Close to the training data, the classifier's predictive distribution is reasonably well-pinned-down and there is little epistemic uncertainty. In contrast, far from the training data, the classifier's predictive distribution is more uncertain, and this uncertainty can be used to detect OOD data. In contrast, aleatoric uncertainty is the irreducible output "noise" that is left over when there is no uncertainty in the parameters (e.g. because a lot of training data is available).

Some work using Bayesian epistemic uncertainty for OOD detection explicitly rejects the use of aleatoric uncertainty (Malinin and Gales, 2018; Malinin et al., 2020; Wen et al., 2019; Choi et al., 2018; Postels et al., 2020), while other work implicitly combines aleatoric and epistemic uncertainty by looking at the overall predictive entropy (Lakshminarayanan et al., 2017; Izmailov et al., 2021; Ovadia et al., 2019; Maddox et al., 2019). Surprisingly from the Bayesian perspective, there are a large number of methods that successfully use aleatoric uncertainty alone to detect OOD points (Hendrycks and Gimpel, 2016; Liang et al., 2017; Lee et al., 2017; Liu et al., 2018; Hendrycks et al., 2018). In addition, many of these methods can be *trained* on OOD points, in a process known as outlier exposure (OE).

© A. Authors.

When suitable OOD data is available, OE leads to dramatic increases in performance (Hendrycks et al., 2018). However, developing a principled Bayesian method using OE is difficult. In particular, a principled Bayesian formulation would involve treating the OE points as providing extra terms in the likelihood. However, it not currently clear how to create a likelihood for outlier points. Instead, current OE methods use a variety intuitively reasonable objectives, which have no interpretation as log-likelihoods and thus cannot be combined with Bayesian inference.

In this paper, we provide a principled account of how to incorporate aleatoric uncertainty and outlier exposure into Bayesian OOD detection methods. In particular, we consider a model of the curation process applied during the original creation of datasets such as CIFAR-10 and ImageNet. Critically, this curation process is designed to filter out a specific set of OOD points: *data points without a well-defined class label*. For simplicity, we will refer to these points as OOD for the remainder of the paper. For instance, if we try to classify an image of a radio as cat vs dog, there is no well-defined class-label, and we should not include that image in the training set. We model curation as a consensus-formation process. In particular, we give each the image to multiple human annotators: if the image has a well-defined label (Fig. 1 left and middle), they will all agree, consensus will be reached and the datapoint will be included in the dataset. In contrast, if the human annotators are given an image with an undefined class label (Fig. 1 right), all they can do is to choose randomly, in which case they disagree, consensus will not be reached and the datapoint will be excluded from the dataset. Critically, that random final choice corresponds to *aleatoric*, not epistemic uncertainty. If we ask a human to classify an image of a radio as cat vs dog, the issue certainly is not that the human annotator is uncertain about the radio's degree of "cat-ness" or "dog-ness". The issue is that we are forcing the human to answer a fundamentally nonsensical question, and the only reasonable response is to choose randomly. That random choice thus corresponds to *aleatoric* uncertainty, and thus aleatoric uncertainty can signal that the point is OOD, and has an undefined class-label. Our model gives a likelihood for being OOD (or having an "undefined class-label") in terms of the underlying classifier probabilities, allowing us to incorporate outliers in principled Bayesian inference. We find that our approach, incorporating OE and aleatoric uncertainty with Bayes performs better than a standard Bayesian approach without OE, and better than a standard aleatoric uncertainty based approach with OE (e.g. Hendrycks et al., 2018).

2. Background: A model for data curation

In the introduction, we briefly noted that different annotators will agree about the class label when that class label is well-defined, but will disagree for OOD inputs without a well-defined class-label (if only because they are forced to the label the image and the only thing they can do is to choose randomly). Interestingly, a simplified generative model which considers the probability of disagreement amongst multiple annotators has already been developed to describe the process of data curation (Aitchison, 2020, 2021). In data curation, the goal is to exclude any OOD images to obtain a high-quality dataset containing images with well-defined and unambiguous class-labels. Standard benchmark datasets in image classification have indeed been carefully curated. For instance, in CIFAR-10, graduate student annotators were instructed that "It's worse to include one that shouldn't be

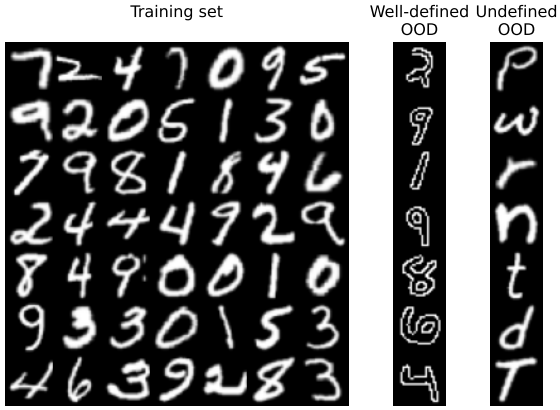

Figure 1: When training on MNIST (left), there is the potential for OOD images with a well-defined class-label (middle), and for images that simultaneously are OOD and have an undefined class-label (right).

included than to exclude one", then Krizhevsky et al. (2009) "personally verified every label submitted by the annotators". Similarly, when ImageNet was created, Deng et al. (2009) made sure that a number of Amazon Mechanical Turk (AMT) annotators agreed upon the class before including an image in the dataset.

Aitchison (2021) proposes a generative model of data curation that we will connect to the problem of OOD detection. Given a random input, X, drawn from $P(X)$, a group of S annotators (indexed by $s \in \{1, \ldots, S\}$) are asked to assign labels $Y_s \in \mathcal{Y}$ to X, where $\mathcal{Y} = \{1, \ldots, C\}$ represents the label set of C classes. If X is OOD, annotators are instructed to label the image randomly. We assume that if the class-label is well-defined, sufficiently expert annotators will all agree on the label, so consensus is reached, $Y_1 = Y_2 = \cdots = Y_S$, and the image will be included in the dataset. Any disagreement is assumed to arise because the image is OOD, and such images are excluded from the dataset. In short, the final label Y is chosen to be Y_1 if consensus was reached and `Undef` otherwise (Fig. 2B).

$$Y|\{Y_s\}_{s=1}^S = \begin{cases} Y_1 & \text{if } Y_1 = Y_2 = \cdots = Y_S \\ \texttt{Undef} & \text{otherwise} \end{cases} \tag{1}$$

From the equation above, we see that $Y \in \mathcal{Y} \cup \{\texttt{Undef}\}$, that is, Y could be any element from the label set $\mathcal{Y}$ if annotators come to agreement or `Undef` if consensus is not reached. Suppose further that all annotators are IID (in the sense that their probability distribution over labels given an input image is the same). Then, the probability of $Y \in \mathcal{Y}$ can be written as

$$\begin{aligned} \mathrm{P}\left(Y = y|X, \theta\right) &= \mathrm{P}\left(\{Y_s = y\}_{s=1}^S|X, \theta\right) \\ &= \textstyle\prod_{s=1}^S \mathrm{P}\left(Y_s = y|X, \theta\right) \\ &= \mathrm{P}\left(Y_s = y|X, \theta\right)^S = p_y^S(X) \end{aligned} \tag{2}$$

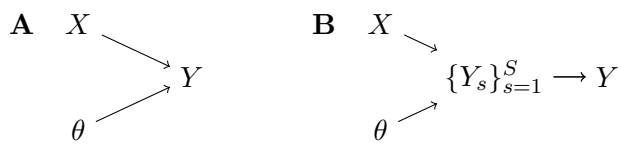

Figure 2: Graphical models under consideration. **A** The generative model for standard supervised learning with no data curation. **B** The generative model with data curation. (Adapted with permission from Aitchison, 2021).

where we have abbreviated the single-annotator probability as $p_y(X) = \mathrm{P}\left(Y_s{=}y|X,\theta\right)$. When consensus is not reached (noconsensus), we have:

$$\begin{aligned}\mathrm{P}\left(Y{=}\texttt{Undef}|X,\theta\right) &= 1 - \sum_{y\in\mathcal{Y}} \mathrm{P}\left(Y{=}y|X,\theta\right) \\ &= 1 - \sum_{y\in\mathcal{Y}} p_y^S(X). \end{aligned} \tag{3}$$

Notice that the maximum of Eq.(3) is achieved when the predictive distribution is uniform: $p_y = 1/C, \forall y \in \mathcal{Y}$, as can be shown using a Lagrange multiplier γ to capture the normalization constraint,

$$L = \left(1 - \textstyle\sum_y p_y^S\right) + \gamma\left(1 - \textstyle\sum_y p_y\right) \tag{4}$$

$$0 = \frac{\partial L}{\partial p_y} = -S p_y^{S-1} - \gamma \tag{5}$$

The value of p_y with maximal L is independent of y, so p_y is the same for all $y \in \mathcal{Y}$, and we must therefore have $p_y = 1/C$. In addition, the minimum of zero is achieved when one of the C classes has a probability $p_y(X) = 1$. Therefore, an input with high predictive (aleatoric) uncertainty is, *by definition*, an input with a high probability of disagreement amongst multiple annotators, which corresponds to being OOD.

3. Methods

We are able to form a principled log-likelihood objective by combining Eq.(2) for inputs with a well-defined class-label (denoted by $\mathcal{D}_{\text{in}}$) and Eq.(3) for OOD inputs without a well-defined class label (denoted by $\mathcal{D}_{\text{out}}$). However, this model was initially developed for cold-posteriors (Aitchison, 2021) and semi-supervised learning (Aitchison, 2020) where the noconsensus inputs were not known and were omitted from the dataset. In contrast, and following Hendrycks et al. (2018), we use proxy datasets for OOD inputs, and explicitly maximize the probability of inputs from those proxy datasets having being OOD (Eq. 8). Importantly, now that we explicitly fit the probability of an undefined class-label, we need to introduce a little more flexibility into the model. In particular, a key issue with the current model is that more annotators, S, implies a higher chance of disagreement hence implying more OOD images. Thus, arbitrary choices about the relative amount of training data with

well-defined and undefined class-labels might cause issues. To avoid any such issues, we modify the undefined-class probability by including a base-rate or bias parameter, c, which modifies the log-odds for well-defined vs undefined class-labels. In particular, we define the logits to be,

$$\ell_0 = c + \log\left(1 - \textstyle\sum_{y\in\mathcal{Y}} p_y^S(X)\right) \tag{6}$$

$$\ell_{y\in\mathcal{Y}} = \log p_y^S(X) \tag{7}$$

where $p_y(X)$ is the single-annotator probability output by the neural network.

$$\mathrm{P}\left(Y = \texttt{Undef}|X,\theta\right) = \frac{e^{\ell_0}}{e^{\ell_0} + \sum_{y\in\mathcal{Y}} e^{\ell_y}} \tag{8}$$

$$\mathrm{P}\left(Y = y|X,\theta\right) = \frac{e^{\ell_y}}{e^{\ell_0} + \sum_{y\in\mathcal{Y}} e^{\ell_y}} \tag{9}$$

with $c = 0$, this reverts to Eq.(2) and (3), while non-zero c allow us to modify the ratio of well-defined to undefined class-labels to match that in the training data.

Of course, we do not have the actual datapoints that were rejected during the data curation process, so instead $\mathcal{D}_{\text{out}}$ is a proxy dataset (e.g. taking CIFAR-10 as $\mathcal{D}_{\text{in}}$, we might use downsampled ImageNet with 1000 classes as $\mathcal{D}_{\text{out}}$). The objective is,

$$\mathcal{L} = \mathbb{E}_{\mathcal{D}_{\text{in}}}\left[\log \mathrm{P}\left(Y = y|X,\theta\right)\right] + \lambda \mathbb{E}_{\mathcal{D}_{\text{out}}}\left[\log \mathrm{P}\left(Y = \texttt{Undef}|X,\theta\right)\right] \tag{10}$$

where λ represents the relative quantity of inputs with undefined to well-defined class-labels. We use $\lambda = 1$ both for simplicity and because the inclusion of the bias parameter, c, should account for any mismatch between the "true" and proxy ratios of inputs with well-defined and undefined class-labels. In addition, we use a fixed value of $S = 10$ as is suggested by Aitchison (2021) and we learn c via backpropagation during training.

Lastly, since our objective is a well-defined likelihood function that jointly models $\mathcal{D}_{\text{in}}$ and $\mathcal{D}_{\text{out}}$, we can easily turn our model into a fully Bayesian one by adding a prior distribution on the neural network parameters, θ, and then perform approximate inference approaches (e.g. stochastic gradient Markov chain Monte Carlo) to estimate the posterior distribution over θ. The use of Bayesian inference in our approach allows us to incorporate both epistemic uncertainty and aleatoric uncertainty when detecting OOD samples and we will show in next section that combining two types of uncertainty together can lead to performance superior than using either of them alone.

4. Results

In this section, we demonstrate the effectiveness of our approach via large scale image classification experiments with CIFAR-10 and CIFAR-100 as $\mathcal{D}_{\text{in}}$, and downsampled ImageNet as our training $\mathcal{D}_{\text{out}}$. We considered two different baselines, in addition to our method. First, in "BNN", we followed the usual OOD detection procedure for Bayesian neural networks in training on our in-distribution dataset, $\mathcal{D}_{\text{in}}$, using SGLD, and ignored our OOD dataset, $\mathcal{D}_{\text{out}}$, as it is not clear how to incorporate these points into a classical Bayesian neural network. Second, in "OE", we used the non-Bayesian method of Hendrycks et al.

Dataset		FPR95 ↓		
$\mathcal{D}_{\text{in}}$	$\mathcal{D}_{\text{test}}$	BNN	OE	Ours
CIFAR-10	Gaussian	13.91±11.05	9.03±5.83	**0.00±0.0**
	Rad.	11.01±7.05	7.62±2.49	**0.00±0.0**
	Blob	35.16±5.19	33.25±10.17	**0.00±0.0**
	Texture	37.61±1.14	52.17±6.15	**25.19±3.93**
	SVHN	28.55±4.36	19.43±2.46	**12.28±2.28**
CIFAR-100	Gaussian	14.47±5.45	32.92±12.43	**0.00±0.00**
	Rad.	26.60±11.01	10.95±8.35	**0.00±0.00**
	Blob	29.95±3.59	36.62±16.30	**0.01±0.01**
	Texture	69.74±2.27	76.72±2.74	**58.43±2.09**
	SVHN	52.18±2.90	63.67±6.24	**42.70±5.57**

Table 1: Experimental results on a range of different datasets for FPR95 (see Appendix A.1 and Appendix A.2 for more details). Note the arrows indicate the "better" direction (i.e. so lower FPR95 is better). $\mathcal{D}_{\text{in}}$ represents the in-distribution dataset. $\mathcal{D}_{\text{test}}$ is the testing out-of-distribution dataset. (The results reported are mean and standard error computed over 6 runs of different random seeds.)

(2018), which trains on $\mathcal{D}_{\text{in}}$, and incorporates an objective that encourages uncertainty on $\mathcal{D}_{\text{out}}$. Our method uses a BNN, as in the BNN baseline, but additionally trains on $\mathcal{D}_{\text{out}}$ using the log-likelihood from Eq.(3) to increase uncertainty on those points. The network architecture is chosen to be a 40-2 Wide Residual Network (Zagoruyko and Komodakis, 2016) for all experiments. OE was trained directly using the code from Hendrycks et al. (2018). For BNN and our approach, we used Cyclical Stochastic Gradient MCMC (Zhang et al., 2020) to perform approximate inference over the network parameters. In addition, we used a temperature of 0.1 on the likelihood for the baseline BNN, so as to match $S = 10$ in the labelled likelihood for our model (Eq. 2 Aitchison, 2020). In addition, the OOD score is chosen to be the predictive distribution's total uncertainty for all experiments (see Appendix A.3).

Results Broadly, we found that our approach gave superior performance to the OE and BNN using FPR95 on a wide range of test datasets $\mathcal{D}_{\text{test}}$ that the model was not trained on (Table 1).

5. Conclusion

We developed a likelihood for undefined class-label samples (a subset of OOD points), and used it to integrate OE methods within principled Bayesian inference. The resulting Bayesian OE method gave superior performance to other methods, including pure aleatoric uncertainty and Bayesian methods without OE.

References

Laurence Aitchison. A statistical theory of semi-supervised learning. *arXiv preprint arXiv:2008.05913*, 2020.

Laurence Aitchison. A statistical theory of cold posteriors in deep neural networks. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=Rd138pWXMvG.

Hyunsun Choi, Eric Jang, and Alexander A Alemi. Waic, but why? generative ensembles for robust anomaly detection. *arXiv preprint arXiv:1810.01392*, 2018.

M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

Stefan Depeweg, Jose-Miguel Hernandez-Lobato, Finale Doshi-Velez, and Steffen Udluft. Decomposition of uncertainty in bayesian deep learning for efficient and risk-sensitive learning. In *International Conference on Machine Learning*, pages 1184–1193. PMLR, 2018.

Armen Der Kiureghian and Ove Ditlevsen. Aleatory or epistemic? does it matter? *Structural safety*, 31(2):105–112, 2009.

Craig R Fox and Gülden Ülkümen. Distinguishing two dimensions of uncertainty. *Essays in Judgment and Decision Making*, 2011.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. In *International Conference on Learning Representations*, 2018.

Pavel Izmailov, Sharad Vikram, Matthew D. Hoffman, and Andrew Gordon Wilson. What are bayesian neural network posteriors really like? In *ICML*, volume 139 of *Proceedings of Machine Learning Research*, pages 4629–4640. PMLR, 2021.

Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *NIPS*, pages 5574–5584, 2017.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. *Tech. report*, 2009.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *NIPS*, pages 6402–6413, 2017.

Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. *arXiv preprint arXiv:1711.09325*, 2017.

Shiyu Liang, Yixuan Li, and R Srikant. Principled detection of out-of-distribution examples in neural networks. *arXiv preprint arXiv:1706.02690*, pages 655–662, 2017.

Si Liu, Risheek Garrepalli, Thomas Dietterich, Alan Fern, and Dan Hendrycks. Open category detection with pac guarantees. In *International Conference on Machine Learning*, pages 3169–3178. PMLR, 2018.

Wesley J. Maddox, Pavel Izmailov, Timur Garipov, Dmitry P. Vetrov, and Andrew Gordon Wilson. A simple baseline for bayesian uncertainty in deep learning. *NeurIPS*, pages 13132–13143, 2019.

Andrey Malinin and Mark Gales. Predictive uncertainty estimation via prior networks. *arXiv preprint arXiv:1802.10501*, 2018.

Andrey Malinin, Bruno Mlodozeniec, and Mark Gales. Ensemble distribution distillation. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=BygSP6Vtvr.

Yuval Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Ng. Reading digits in natural images with unsupervised feature learning. *Feature Learning NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, D Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. *Advances in Neural Information Processing Systems*, 32:13991–14002, 2019.

Janis Postels, Hermann Blum, Yannick Strümpler, Cesar Cadena, Roland Siegwart, Luc Van Gool, and Federico Tombari. The hidden uncertainty in a neural networks activations. *arXiv preprint arXiv:2012.03082*, 2020.

Yeming Wen, Dustin Tran, and Jimmy Ba. Batchensemble: an alternative approach to efficient ensemble and lifelong learning. In *International Conference on Learning Representations*, 2019.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *BMVC*, 2016.

Ruqi Zhang, Chunyuan Li, Jianyi Zhang, Changyou Chen, and Andrew Gordon Wilson. Cyclical stochastic gradient mcmc for bayesian deep learning. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=rkeS1RVtPS.

Appendix A. Experiment details

A.1. Test datasets

At test time, we evaluate the models' OOD detection ability on a number of OOD datasets that the model was not trained on, as proposed in Hendrycks and Gimpel (2016) and as implemented by Hendrycks et al. (2018):

1. Isotropic zero-mean Gaussian noise with $\sigma = 0.5$
2. Rademacher noise where each dimension is -1 or 1 with equal probability.
3. Blobs data of algorithmically generated amorphous shapes with definite edges
4. Texture data (Cimpoi et al., 2014) of textural images in the wild.
5. SVHN (Netzer et al., 2011) which contains 32x32 colour images of house numbers.

A.2. OOD metric

OOD detection is in essence a binary classification problem. It is therefore sensible to use metrics for binary classification to evaluate a model's ability to detect OOD inputs. In particular, we adopt the false positive rate at $N\%$ true positive rate (FPRN), which computes the probability of an input being misclassified as having an undefined class-label (false positive) when at least $N\%$ of the true inputs with undefined class-labels are correctly detected (true positive). In practice, we would like to have a model with low FPR$N\%$ since an ideal model should detect nearly all inputs with undefined class-labels while raising as few false alarms as possible. In our experiments, we let $N = 95$.

A.3. OOD score

At test time, to distinguish between OOD and in-distribution examples, we need a score that measures the model's uncertainty. There are several model-specific choices. In our model one can choose to use the OOD probability (Eq. 3) as the score. Hendrycks and Gimpel (2016); Hendrycks et al. (2018) use the negative maximum softmax probability (Hendrycks and Gimpel, 2016). However, to ensure a fair comparison, we used one metric that makes sense for all methods considered, the total uncertainty (Depeweg et al., 2018). The total uncertainty is the entropy of the predictive distribution, marginalising over uncertainty in the neural network parameters, $\mathbb{H}\left[p(y \mid x^*)\right]$, which equals the sum of aleatoric uncertainty and epistemic uncertainty. Note that the total uncertainty from OE only contains aleatoric uncertainty since the model is fully deterministic. In contrast, the standard BNN and our BNN approach both have aleatoric and epistemic uncertainty. The key differences is that in our model, the aleatoric uncertainty is shaped by outlier exposure, whereas in a standard BNN, the aleatoric uncertainty is determined solely by the in-distribution data.