On the Calibration of Deterministic Epistemic Uncertainty

Janis Postels [*1] **Mattia Segu** [*1 2] **Tao Sun** [1] **Luc Van Gool** [1] **Fisher Yu** [1] **Federico Tombari** [3 4]

Abstract

A set of novel approaches for estimating epistemic uncertainty in deep neural networks with a single forward pass has recently emerged as a valid alternative to Bayesian Neural Networks. On the premise of informative representations, these deterministic uncertainty methods (DUMs) achieve strong performance on detecting out-of-distribution (OOD) data while adding negligible computational costs at inference time. However, so far it remains unclear whether DUMs are well calibrated and can seamlessly scale to real-world applications - both prerequisites for their practical deployment. We firstly provide a taxonomy of DUMs, evaluate their calibration under continuous distributional shifts and their performance on OOD detection for image classification tasks. Then, we extend the most promising DUMs to semantic segmentation and demonstrate that DUMs scale to realistic vision tasks. We show that the practicality of current methods, despite their efficacy on OOD detection, is undermined by their poor calibration under realistic distributional shift.

1. Taxonomy for Deterministic Uncertainty Quantification

We propose a taxonomy of existing Deterministic Uncertainty Methods (DUMs). To quantify epistemic uncertainty deterministically, one needs to assume that the distribution of the hidden representations of a neural network is representative of the input distribution. However, discriminative models suffer from the fundamental problem of feature collapse. Thus, we firstly categorize DUMs according to the strategy adopted to mitigate feature collapse (Sec. 1.1). Another important dimension along which DUMs deviate is the method used for uncertainty estimation (Sec. 1.2).

[1]Computer Vision Lab, ETH Zurich, Zurich, Switzerland [2]Max Planck ETH Center for Learning Systems [3]Google Inc, Zurich, Switzerland [4]TUM Munich, Munich, Germany. Correspondence to: Janis Postels <jpostels@ethz.ch>, Mattia Segu <segum@ethz.ch>.

Preliminary work. Under review by the ICML 2021 Workshop on Uncertainty and Robustness in Deep Learning. Do not distribute.

Feature Collapse. Discriminative models can learn to discard large part of their input information, as exploiting spurious correlations may lead to better performance on the training data distribution (Peters et al., 2017). Such learned invariant representations extend to out-of-distribution (OOD) data, resulting in a collapse of OOD embeddings to in-distribution features and harming domain shift modeling (Segù et al., 2020). This problem, known as *feature collapse* (Van Amersfoort et al., 2020), makes OOD detection based on high-level representations impossible.

1.1. Regularization of Hidden Representations

We group DUMs according to the strategy used for regularizing hidden representations to mitigate feature collapse. We identify two main paradigms - distance awareness (Sec. 1.1.1) and informative representations (Sec. 1.1.2).

1.1.1. Distance Awareness

The fundamental idea of distance-aware hidden representations is to avoid feature collapse by enforcing distances between latent representations to mirror distances in the input space. This can be achieved by constraining the Lipschitz constant, as it enforces a lower and an upper bound to expansion and contraction performed by an underlying neural network. A lower bound is associated with *sensitivity*, since it enforces distances in the input space to translate into distances in the latent space and provides a solution to feature collapse. Moverover, an upper bound enforces *smoothness* of the hidden representations, *i.e.* small changes in the input do not result in large changes in the latent space. More formally, given any pair of inputs x_1 and x_2 the following lower and upper bounds must hold for the resulting activation of a feature extractor f_θ with parameters θ: $c_1||x_1 - x_2||_I \leq ||f_\theta(x_1) - f_\theta(x_2)||_F \leq c_2||x_1 - x_2||_I$. c_1 and c_2 denote respectively the lower and upper bound for the Lipschitz constant, and $||\cdot||_I$ and $||\cdot||_F$ are the chosen metrics in the input and feature space respectively. While there exist other approaches, *e.g.* (Mandelbaum & Weinshall, 2017; Obukhov et al., 2021), recent proposals have primarily adopted two methods to impose the bi-Lipschitz constraint and make the feature extractor distance preserving, *i.e.* gradient penalty (Gulrajani et al., 2017) as in DUQ (Van Amersfoort et al., 2020) or spectral normalization (Miyato et al., 2018) as in SNGP (Liu et al., 2020), DUE (van Amersfoort

et al., 2021) and DDU (Mukhoti et al., 2021). We refer to the supplement for a detailed description of gradient penalty, spectral normalization and their limitations.

1.1.2. Informative Representations

While methods enforcing distance awareness according to a predefined distance metric achieve remarkable performance in OOD detection, they do not explicitly preserve sample-specific information. Consequently, they may discard useful information about the input or act overly sensitive. An alternative line of work proposes to avoid feature collapse by learning informative representations (Alemi et al., 2018; Wu & Goodman, 2020; Postels et al., 2020; Nalisnick et al., 2019; Ardizzone et al., 2018; 2020), thus forcing discriminative models to preserve information beyond what is required by its target task independent of the choice of an underlying distance metric. Notably, while representations that are aware of distances in the input space are arguably also informative, both categories remain fundamentally different in their approach to feature collapse. While distance-awareness is based on the choice of a specific distance metric tying together input and latent space, informative representations enforce a constraint on the distribution of hidden representations. We identify three distinct families of approaches to enforce informative representations - contrastive learning (Wu & Goodman, 2020), reconstruction regularization for Maximally Informative Representations (MIR) (Postels et al., 2020), and invertible neural networks (Nalisnick et al., 2019; Ardizzone et al., 2020). We refer to the supplementary materials for a details description of those.

1.2. Uncertainty Estimation

We distinguish two directions to quantifying uncertainty based on such regularized representations. While *generative approaches* use the likelihood produced by an explicit generative model of the distribution of hidden representations as uncertainty proxy, *discriminative methods* directly use the predictions based on regularized representations to quantify uncertainty. We refer to the appendix for a detailed description and associated DUMs.

2. Evaluation of Deterministic Uncertainty

We first benchmark DUMs on popular image classification datasets (Sec. 2.1). Further, we evaluate how such techniques hold their promises on harder tasks, *e.g.* semantic segmentation, and successfully propose extended versions to dense prediction tasks of most representative DUMs (Sec. 2.2). Finally, in Sec. 2.2.3 we show the poor calibration of DUMs under realistic distributional shift.

Baselines. We compare DUMs with two baselines for epistemic uncertainty - Monte-Carlo (MC) dropout (Gal & Ghahramani, 2016) and deep ensembles (Lakshminarayanan et al., 2017). While these baselines are expected to predict reasonable uncertainty, they require more computations than DUMs. We also report the performance of the softmax entropy as a baseline deterministic uncertainty method.

OOD detection metrics. We follow prior work (Liu et al., 2020; Postels et al., 2020) and compute Area Under the Receiver Operating Characteristic (AUROC) and Area Under the Precision-Recall curve (AUPR) between test data that originates from the same distribution as the training data and data originating from another dataset (Sec. 2.1.1).

Calibration metric. Traditional metrics, such as Expected Calibration Error (ECE) (Naeini et al., 2015) and Brier score (BRIER, 1950), are not applicable to some DUMs. Therefore, we measure the calibration of an uncertainty estimate under continuous distributional shifts using the Relative Area Under the Lift Curve (rAULC) (see supplement).

2.1. Image Classification

Datasets. We train DUMs on MNIST (LeCun, 1998), FashionMNIST (Xiao et al., 2017)), CIFAR-10 (Krizhevsky et al., 2014) and SVHN (Netzer et al., 2011)). Further, when evaluating OOD detection, we also evaluate on Omniglot (Lake et al., 2015) for models trained on MNIST/FashionMNIST and on STL-10 (Coates et al., 2011) for models trained on CIFAR10/SVHN.

Models and optimization. For the experiments on MNIST and Fashion-MNIST, we employ a multilayer perceptron (MLP) as feature extractor with 3 hidden layers of 100 dimensions each and ReLU activation functions. When training on CIFAR-10 and SVHN, we use a ResNet-18 (He et al., 2016) as backbone. Each DUM is associated with a particular hyperparameter for the regularization of its hidden representations. We choose the hyperparameter such that it minimizes the validation loss in the experiments on OOD detection and calibration. However, this section also includes an analysis of the sensitivity of calibration and OOD detection performance to the hyparameter based on DUMs using distinct regularization techniques (Fig. 4). All experiments were run 5 times. We refer to the supplement for a detailed description on the optimization procedure for each DUM.

Continuous distributional shifts. We evaluate calibration of DUMs on continuously shifted test data. We apply rotations (MNIST/FashionMNIST) in steps of 20° from 0° to 180°. On SVHN/CIFAR10 we add Gaussian noise of increasing standard deviation σ to the test data. We vary the σ in steps of 0.05 from 0 to 0.25 (on normalized data).

2.1.1. Results

Calibration under continuous distributional shift. Tab. 1 shows accuracy on the test set and calibration performance

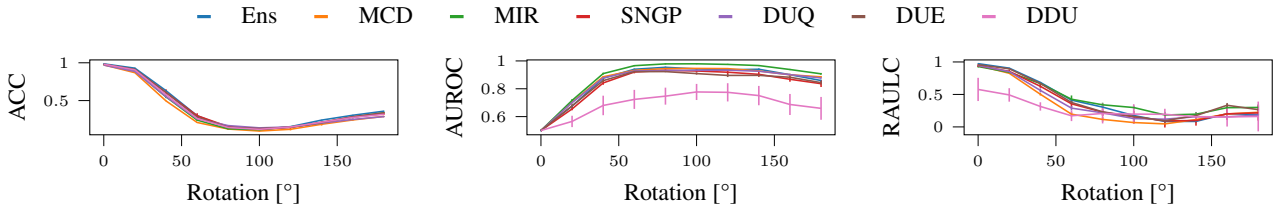


Figure 1. We compare the behaviour of various DUMs and the baselines under continuous distributional shifts. Therefore, we plot accuracy (LEFT), AUROC (CENTER) and rAULC (RIGHT) against the perturbation magnitude for a MLP trained on MNIST. We perturbe MNIST using rotations. The AUROC is computed against unperturbed test data.

in terms of rAULC. Fig. 1 shows accuracy, AUROC and rAULC against the rotation angle applied to the test data. Except DUQ, prone to training instabilities, all DUMs achieve similar predictive performance as MC dropout, deep ensembles and standard softmax models. In terms of uncertainty calibration, we observe that DUE, SNGP and MIR compete with established uncertainty estimation methods (*e.g.* MC dropout) across most datasets and both types of distributional shifts. Consequently, these approaches do not only fare well at detecting OOD data but also deliver well calibrated uncertainties under synthetic distributional shifts. However, this is not sufficient to justify their use for real-world safety-critical applications. We refer to Sec. 2.2.3 for an analysis of calibration under realistic distributional shift.

Sensitivity to Regularization Strength and OOD Detection. It is important to understand the impact of the regularization parameter on the calibration and OOD detection performance. We refer to the appendix for an empirical analysis - where we find that enforcing distance aware representations does not show any evidence for correlating with either - and for an ablation on OOD detection for DUMs.

2.2. Semantic Segmentation

Prior work on DUMs were motivated from a practical perspective by their fast inference speed. However, no practical extension of such methods to dense prediction tasks has been proposed. This section evaluates whether DUMs scale to large vision tasks and compares their behaviour under realistic distributional shifts with MC dropout. We choose MIR (Postels et al., 2020) and SNGP (Liu et al., 2020), both demonstrating good quality of uncertainty for image classification, and adapt them to semantic segmentation.

We consider semantic segmentation as a multidimensional classification problem, where for each pixel of the output mask represents an independent classification problem. Given an image $\mathbf{x}$ with n pixels $\mathbf{y} = \{y_1, \cdots, y_n\}$, the predictive distribution factorizes according to $p(\mathbf{y} \mid \mathbf{x}) = p(y_1 \mid \mathbf{x})p(y_2 \mid \mathbf{x}) \cdots p(y_n \mid \mathbf{x})$. We use a global uncertainty estimate for the output map as our distributional shifts act on a global scale (see Sec. 2.2.1) and, thus, use the average pixel-level uncertainty as an image-level uncertainty.

While for image classification we simulated domain shift through synthetic distributional shifts, *i.e.* additive Gaussian noise and rotation, for semantic segmentation we are interested in challenging current methods with realistic continuous distributional shifts along natural directions. Sec. 2.2.1 describes the dataset we collected for this purpose. In Sec. 2.2.2 we report architectural choices and we detail how we extended the chosen DUMs to dense prediction tasks. Results and findings are discussed in Sec. 2.2.3.

2.2.1. Dataset

To benchmark our model on data with realistically and continuously changing environment, we collect a synthetic dataset for semantic segmentation. We use the CARLA Simulator (Dosovitskiy et al., 2017) for rendering the images and segmentation masks (see supplement) with classes found in CityScape dataset (Cordts et al., 2016).

Training data. Data is collected from four towns in CARLA. We produce 32 sequences from each town with randomly generated vehicles and pedestrians. Every sequence has has 500 frames with a sampling rate at 10 FPS. We uniformly sample a validation set from this.

OOD data. We use the time-of-the-day as the parameter for continuously changing the distribution. Visual examples and details on data collection are in the appendix. The time-of-the-day is parametrized by the Sun's altitude angle, where $90°$ means the mid-day (training data) and the $0°$ means the dust/dawn. We produce samples with altitude angles from $90°$ to $15°$ by steps of $5°$, and $15°$ to $-5°$, where the environment changes sharply, in $1°$ steps.

2.2.2. Method Details

Backbone. We adopt Dilated ResNet (DRN) (Yu & Koltun, 2016; Yu et al., 2017) as semantic segmentation backbone. DRN introduces dilated convolutions to the ResNet, effectively increasing the receptive field without increasing the number of layers or parameters. This improves the spatial accuracy of DRN, achieving satisfactory results on CityScapes (Cordts et al., 2016). We adopt the variant DRN-

Method	MNIST		FashionMNIST		CIFAR10		SVHN	
	Acc	rAULC	Acc	rAULC	Acc	rAULC	Acc	rAULC
Softmax	98.1 ± 0.1	54.1 ± 3.4	88.1 ± 0.2	7.1 ± 2.9	84.5 ± 0.2	43.4 ± 17.4	93.2 ± 0.2	84.2 ± 0.5
Dropout	97.2 ± 0.1	59.2 ± 0.8	87.8 ± 0.2	**48.4 ± 1.2**	86.0 ± 0.2	**65.5 ± 0.8**	94.0 ± 0.1	82.2 ± 0.1
Ensemble	98.6 ± 0.2	59.4 ± 1.4	88.9 ± 0.2	24.3 ± 1.5	**88.9 ± 0.2**	60.6 ± 3.2	95.2 ± 0.2	84.7 ± 1.4
DUE	**98.1 ± 0.1**	59.0 ± 2.0	88.6 ± 0.2	24.5 ± 3.1	84.6 ± 1.9	59.6 ± 3.2	88.5 ± 7.8	81.6 ± 4.5
DUQ	96.9 ± 0.1	61.4 ± 1.2	87.4 ± 0.4	41.8 ± 1.5	77.2 ± 0.9	37.2 ± 13.2	91.5 ± 0.3	79.2 ± 3.4
DDU	**98.1 ± 0.1**	38.3 ± 9.0	**89.3 ± 0.1**	33.5 ± 9.7	85.5 ± 0.2	34.3 ± 10.9	**94.8 ± 0.2**	67.7 ± 7.3
SNGP	97.7 ± 0.4	58.9 ± 4.6	86.4 ± 1.4	34.7 ± 3.7	85.5 ± 0.3	64.6 ± 3.0	94.0 ± 0.2	**84.5 ± 0.6**
MIR	97.6 ± 0.2	**72.4 ± 1.1**	87.4 ± 0.3	47.4 ± 3.7	85.3 ± 0.2	55.0 ± 5.7	94.1 ± 0.2	71.9 ± 3.1

Table 1. We evaluate the in-domain test accuracy (Acc, %) and calibration performance (rAULC, %) on MNIST, FashionMNIST, CIFAR10 and SVHN datasets. The backbones for all methods are MLP for MNIST/FashionMNIST and ResNet-18 for CIFAR10/SVHN.

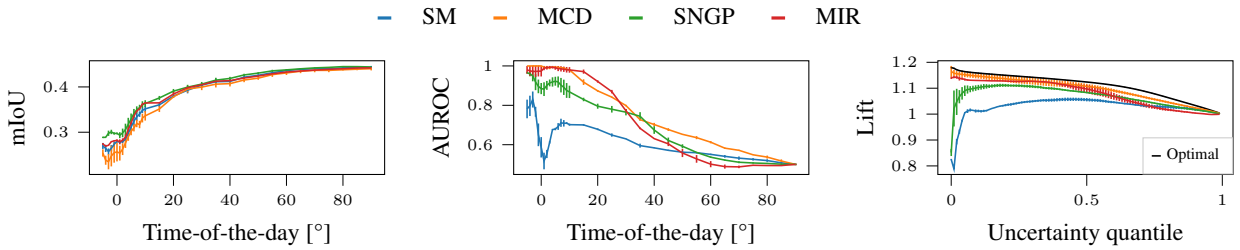


Figure 2. We compare the mean Intersection over Union (mIoU) (left), AUROC (center) and rAULC (right) of softmax (SM), MC dropout (MCD), SNGP and MIR under realistic continuous distributional shifts. The AUROC is computed against unperturbed test data. While the softmax entropy as a measure of uncertainty clearly fails in this scenario, DUMs (SNGP/MIR) yield reasonable uncertainty using a similar computational budget at identical predictive performance (mean Intersection over Union (mIoU)). However, MC dropout, with its larger computational footprint, provides better uncertainty estimates, especially in terms of calibration (Lift).

A-50. All experiments were run 3 times.

Adapting SNGP and MIR to semantic segmentation. We refer to the supplement for a description of adapting SNGP and MIR to semantic segmentation.

2.2.3. Results

Continuous Distribution Shift. We evaluate the calibration of DUMs under realistic distributional shifts against MC dropout and softmax entropy (deterministic baseline). Tab. 2 shows quantitative results. All methods yield similar performance in terms of mean Intersection over Union (mIoU) and pixel accuracy. However, while DUMs yield a considerable improvement over the softmax entropy both in OOD detection and calibration using a similar computational budget, these methods still fall short of established approaches, such as MC dropout, in terms of calibration under challenging realistic distributional shifts. This finding is further supported by the results of Fig. 2, which shows how DUMs behave poorly in terms of calibration (see Lift curves) against severe distributional shift.

Method		Softmax	Dropout	SNGP	MIR
In-domain Testset	Acc (%)	93.0	92.9	93.2	93.0
	mIoU (%)	44.3	44.0	44.4	44.2
Time-of-the-Day	RAULC (%)	25.9	**91.2**	66.7	72.5

Table 2. Calibration results on semantic segmentation.

3. Conclusion

While DUMs show good OOD detection performance and are interesting for practical applications in need of efficient uncertainty quantification, we find them struggling regarding calibration under realistic distributional shifts. Despite promising results on toy datasets, this does not extrapolate to more realistic scenarios (Sec. 2.2). We find that, while DUMs outperform the softmax entropy, MC dropout clearly outperforms DUMs regarding uncertainty calibration. Another desirable property for such methods would be that the strength of the feature space regularization correlates with the quality of OOD detection. However, this is not verified for Lipschitz regularization by our investigation (see supplement). As expected, the lack of correlation also extends to calibration performance. We hope that our findings will foster future research on making these promising family of methods better calibrated and more broadly applicable.

References

Alemi, A. A., Fischer, I., and Dillon, J. V. Uncertainty in the variational information bottleneck. *arXiv preprint arXiv:1807.00906*, 2018.

Ardizzone, L., Kruse, J., Wirkert, S., Rahner, D., Pellegrini, E. W., Klessen, R. S., Maier-Hein, L., Rother, C., and Köthe, U. Analyzing inverse problems with invertible neural networks. *arXiv preprint arXiv:1808.04730*, 2018.

Ardizzone, L., Mackowiak, R., Rother, C., and Köthe, U. Training normalizing flows with the information bottleneck for competitive generative classification. *Advances in Neural Information Processing Systems*, 33, 2020.

Behrmann, J., Grathwohl, W., Chen, R., Duvenaud, D., and Jacobsen, J. Invertible residual networks. arxiv e-prints. *arXiv preprint arXiv:1811.00995*, 2018.

Blum, H., Sarlin, P.-E., Nieto, J., Siegwart, R., and Cadena, C. The fishyscapes benchmark: Measuring blind spots in semantic segmentation. *arXiv preprint arXiv:1904.03215*, 2019.

BRIER, G. W. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950.

Burt, D., Rasmussen, C. E., and Van Der Wilk, M. Rates of convergence for sparse variational gaussian process regression. In *International Conference on Machine Learning*, pp. 862–871. PMLR, 2019.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

Coates, A., Ng, A., and Lee, H. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 215–223. JMLR Workshop and Conference Proceedings, 2011.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Daunizeau, J. Semi-analytical approximations to statistical moments of sigmoid and softmax mappings of normal variables. *arXiv preprint arXiv:1703.00091*, 2017.

Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., and Koltun, V. CARLA: An open urban driving simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*, pp. 1–16, 2017.

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

Gal, Y., Islam, R., and Ghahramani, Z. Deep bayesian active learning with image data. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1183–1192. JMLR. org, 2017.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. Improved training of wasserstein gans. *arXiv preprint arXiv:1704.00028*, 2017.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Hensman, J., Matthews, A., and Ghahramani, Z. Scalable variational gaussian process classification. In *Artificial Intelligence and Statistics*, pp. 351–360. PMLR, 2015.

Jacobsen, J.-H., Smeulders, A., and Oyallon, E. i-revnet: Deep invertible networks. *arXiv preprint arXiv:1802.07088*, 2018.

Krizhevsky, A., Nair, V., and Hinton, G. The cifar-10 dataset. *online: http://www. cs. toronto. edu/kriz/cifar. html*, 55:5, 2014.

Lake, B. M., Salakhutdinov, R., and Tenenbaum, J. B. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

Lakshminarayanan, B., Pritzel, A., and Blundell, C. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in neural information processing systems*, pp. 6402–6413, 2017.

LeCun, Y. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Liu, J. Z., Lin, Z., Padhy, S., Tran, D., Bedrax-Weiss, T., and Lakshminarayanan, B. Simple and principled uncertainty estimation with deterministic deep learning via distance awareness. *Conference on Neural Information Processing Systems*, 2020.

Mandelbaum, A. and Weinshall, D. Distance-based confidence score for neural network classifiers. *arXiv preprint arXiv:1709.09844*, 2017.

Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018.

Mukhoti, J., Kirsch, A., van Amersfoort, J., Torr, P. H., and Gal, Y. Deterministic neural networks with appropriate inductive biases capture epistemic and aleatoric uncertainty. *arXiv preprint arXiv:2102.11582*, 2021.

Naeini, M. P., Cooper, G., and Hauskrecht, M. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015.

Nalisnick, E., Matsukawa, A., Teh, Y. W., Gorur, D., and Lakshminarayanan, B. Do deep generative models know what they don't know? In *International Conference on Learning Representations*, 2018.

Nalisnick, E., Matsukawa, A., Teh, Y. W., Gorur, D., and Lakshminarayanan, B. Hybrid models with deep and invertible features. In *International Conference on Machine Learning*, pp. 4723–4732. PMLR, 2019.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading digits in natural images with unsupervised feature learning. 2011.

Obukhov, A., Rakhuba, M., Liniger, A., Huang, Z., Georgoulis, S., Dai, D., and Van Gool, L. Spectral tensor train parameterization of deep learning layers. In *International Conference on Artificial Intelligence and Statistics*, pp. 3547–3555. PMLR, 2021.

Oord, A. v. d., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Peters, J., Janzing, D., and Schölkopf, B. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017.

Postels, J., Blum, H., Strümpler, Y., Cadena, C., Siegwart, R., Van Gool, L., and Tombari, F. The hidden uncertainty in a neural networks activations. *arXiv preprint arXiv:2012.03082*, 2020.

Rasmussen, C. E. Gaussian processes in machine learning. In *Summer school on machine learning*, pp. 63–71. Springer, 2003.

Rosca, M., Weber, T., Gretton, A., and Mohamed, S. A case for new neural network smoothness constraints. *arXiv preprint arXiv:2012.07969*, 2020.

Sedghi, H., Gupta, V., and Long, P. M. The singular values of convolutional layers. *arXiv preprint arXiv:1805.10408*, 2018.

Segù, M., Tonioni, A., and Tombari, F. Batch normalization embeddings for deep domain generalization. *arXiv preprint arXiv:2011.12672*, 2020.

Singla, S. and Feizi, S. Bounding singular values of convolution layers. *arXiv preprint arXiv:1911.10258*, 2019.

Titsias, M. Variational learning of inducing variables in sparse gaussian processes. In *Artificial intelligence and statistics*, pp. 567–574. PMLR, 2009.

Van Amersfoort, J., Smith, L., Teh, Y. W., and Gal, Y. Uncertainty estimation using a single deep deterministic neural network. In *International Conference on Machine Learning*, pp. 9690–9700. PMLR, 2020.

van Amersfoort, J., Smith, L., Jesson, A., Key, O., and Gal, Y. Improving deterministic uncertainty estimation in deep learning for classification and regression. *arXiv preprint arXiv:2102.11409*, 2021.

Vuk, M. and Curk, T. Roc curve, lift chart and calibration plot. *Metodoloski zvezki*, 3(1):89, 2006.

Wu, M. and Goodman, N. A simple framework for uncertainty in contrastive learning. *arXiv preprint arXiv:2010.02038*, 2020.

Xiao, H., Rasul, K., and Vollgraf, R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Yu, F. and Koltun, V. Multi-scale context aggregation by dilated convolutions. In *International Conference on Learning Representations (ICLR)*, 2016.

Yu, F., Koltun, V., and Funkhouser, T. Dilated residual networks. In *Computer Vision and Pattern Recognition (CVPR)*, 2017.

4. Appendix

We here provide extensive implementation details, additional results, comparisons and ablation studies. In particular, we provide additional results both for image classification Sec. 4.6.1 and for semantic segmentation Sec. 4.6.2. We provide an in-depth explanation of the Lipschitz regularization techniques used (Sec. 4.3). We report training Sec. 4.7 and implementation details Sec. 4.8. Moreover, we explain in detail how uncertainty is derived for image classification Sec. 4.8.3 and semantic segmentation Sec. 4.8.4 . Finally, we provide details on the data collection process and examples from the sequences collected for the semantic segmentation task Sec. 4.9.

4.1. Taxonomy

Tab. 3 shows an overview of the taxonomy introduced in Sec. 1.

4.2. Regularization Methods Enforcing Informative Representations

Contrastive learning (Oord et al., 2018; Wu & Goodman, 2020) has emerged as an approach for learning representations that are both informative and discriminative. This is utilized by Wu *et al.* (Wu & Goodman, 2020), who build on top of SimCLR (Chen et al., 2020) to regularize hidden representations for a discriminative task by proposing an auxiliary contrastive loss on distributions rather than instances.

Reconstruction regularization (Postels et al., 2020) instead forces the intermediate activations to embody a complete representation of the input space. This is achieved by adding a decoder branch fed with the activations of a given layer to reconstruct the input. Since this method aims at producing Maximally Informative Representations, we term it MIR.

Invertible Neural Networkss (INNs) (Jacobsen et al., 2018; Ardizzone et al., 2018; Nalisnick et al., 2019; Ardizzone et al., 2020), built via a cascade of homeomorphic layers, cannot discard information except at the final classification stage. Consequently, the mutual information between input and hidden representation is retained. Interestingly, Behrmann *et al.* (Behrmann et al., 2018) showed that a ResNet is invertible if its Lipschitz constant is lower than 1, meaning that invertible ResNets both possess highly-informative representations and satisfy distance awareness.

4.3. Lipschitz Regularization

Gradient Penalty. First introduced to regularize the Lipschitz constant in GAN training (Gulrajani et al., 2017), a two-sided gradient penalty is used as an additional loss term to enforce detectability in the feature space of changes in the input by DUQ (Van Amersfoort et al., 2020). Given a layer g, regularising the L_2 norm of the Jacobian J enforces a Lipschitz constraint at least locally for a small perturbation ϵ around x, *i.e.* $g(x+\epsilon) - g(x) \simeq J_g(x)\epsilon \leq ||J(x)||^2||\epsilon||^2$. The following two-sided gradient penalty can thus be defined: $\lambda \left[||\nabla_x \sum_c K_c||_2^2 - 1\right]^2$, where λ is the regularization strength, $||\cdot||_2$ is the L_2 norm, the target Lipschitz constant is 1 and $K_c(x)$ is short for the Radial Basis Function (RBF) kernel $K_c(x, e_c) = exp\left[-\frac{1}{n}||W_c f_\theta(x) - e_c||_2^2/(2\sigma^2)\right]$. e_c is the centroid for the class c, W_c is a learnable weight matrix for the class c and σ an hyperparameter called lengthscale.

Spectral Normalization. A more efficient technique to constrain the Lipschitz constant is Spectral Normalization (SN) (Miyato et al., 2018). For each layer $g : \mathbf{h}_{in} \rightarrow \mathbf{h}_{out}$, SN effectively constrain its Lipschitz norm $||g||_{Lip} = sup_{\mathbf{h}} sn(\nabla g(\mathbf{h}))$, where $sn(A)$ is the spectral norm - the L_2 matrix norm - of the matrix A, equivalent to its largest singular value. When applied, SN normalizes the spectral norm of the weights W of each layer to satisfy the soft-Lipschitz constraint $sn(W) = c$ (hard- if the Lipschitz constant $c = 1$): $W_{sn} = W/sn(W)$. A number of DUMs - SNGP (Liu et al., 2020), DUE (van Amersfoort et al., 2021) and DDU (Mukhoti et al., 2021) - relies on SN of the weight matrices to make the feature extractor distance preserving.

Note that principled approaches providing exact singular values in convolutional layers (Sedghi et al., 2018) result in prohibitive computational complexity; the spectral normalization approximations typically adopted by the methods previously described have been found to be sub-optimal *et al.* (Singla & Feizi, 2019), and its interaction with losses, architecture and optimization is yet to be fully understood (Rosca et al., 2020).

4.4. Uncertainty Estimation in Deterministic Epistemic Uncertainty.

Generative approaches estimate the distribution of hidden representations post-training and use the likelihood as uncertainty metric to detect OOD samples. Wu *et al.* (Wu & Goodman, 2020) propose a Deep Contrastive Uncertainty (DCU) method to train an additional deep network from the representation space to a distribution space, where the variance of the distribution is used as a confidence measure. MIR (Postels et al., 2020) fits a class-conditional GMM on the maximally informative latent space and DDU (Mukhoti et al., 2021) on its last hidden space. A special instance of the generative approaches are INNs as they directly estimate the training data distribution. This allows using the likelihood of the input data as a proxy of uncertainty. While this idea is appealing, it can lead to training difficulties, imposes strong constraints on the underlying feature extractor

DUMs		Uncertainty Estimation Method			
		Discriminative		Generative	
		Distance from class centroid	Gaussian Processes	Gaussian Mixture Models	Normalizing Flows
Target Property	Distance awareness	DCS DUQ,	SNGP DUE	DDU	-
	Informative representations	-	-	DCU, MIR	Invertible networks

Table 3. Taxonomy of DUMs. Methods are grouped according to the target property of the hidden representations (rows), and their uncertainty estimation method (columns). DCS (Mandelbaum & Weinshall, 2017), DUQ (Van Amersfoort et al., 2020), SNGP (Liu et al., 2020), DUE (van Amersfoort et al., 2021), DDU (Mukhoti et al., 2021), DCU (Wu & Goodman, 2020), MIR (Postels et al., 2020), Invertible networks (Ardizzone et al., 2018; Nalisnick et al., 2019; Ardizzone et al., 2020)

and in some instances even remains susceptible to OOD data (Nalisnick et al., 2018).

Discriminative approaches use the predictions based on regularized representations to directly assess confidence. Mandelbaum *et al.* (Mandelbaum & Weinshall, 2017) propose to use a Distance-based Confidence Score (DCS) to estimate local density at a point as the Euclidean distance in the embedded space between the point and its k nearest neighbors in the training set. Similarly, DUQ (Van Amersfoort et al., 2020) builds on Radial Basis Function (RBF) networks (LeCun et al., 1998) and propose a novel centroid updating scheme. Uncertainty is estimated as the distance between the model output and the closest centroid. DUMs adopting SN (Liu et al., 2020; van Amersfoort et al., 2021) typically opt to preserve the L_2 distance so to fit Gaussian processes (GPs) with RBF kernels on top of the learned feature space, extending distance awareness to the output layer. In particular, SNGP (Liu et al., 2020) relies on a Laplace approximation of the GP based on the random Fourier feature (RFF) expansion of the GP posterior (Rasmussen, 2003). DUE (van Amersfoort et al., 2021) leverages instead the inducing point approximation (Titsias, 2009; Hensman et al., 2015), allowing to pick an arbitrarily large number of inducing points without overfitting (Burt et al., 2019). The uncertainty is then derived respectively as the Dempster-Shafer metric (Liu et al., 2020) or the softmax entropy (van Amersfoort et al., 2021).

4.5. Metrics.

rAULC. In order to assess the calibration of uncertainty estimates under distributional shifts, we need to introduce a novel metric, since ECE (Naeini et al., 2015) and Brier score (BRIER, 1950) are limited to probabilistic forecasts (*i.e.* methods producing calibrated probabilities). Concretely, we calculate the Area Under the Lift Curve (AULC) (Vuk & Curk, 2006), which we obtain by ordering the predictions according to increasing uncertainty and plotting the performance (*e.g.* accuracy) of all samples with an uncertainty estimate smaller than a certain quantile of the uncertainty against the quantile of uncertainty. Formally, given a set of uncertainty quantiles $q_i \in [0, 1]$, $i \in [1, ..., N]$, with some quantile step width $0 < s < 1$ and the function $F(q_i)$ which returns the accuracy of all samples with uncertainty $u < q_i$, we define the AULC as $AULC = -1 + \sum_{i \in [1,...,N]} s \frac{F(q_i)}{F_R(q_i)}$. Here, $F_R(\cdot)$ refers to a baseline uncertainty estimate that corresponds to random guessing, and we subtract 1 to only measure the improvement over it. Note, if an uncertainty estimate that is anti-correlated with a models' performance the score can also be negative. To alleviate biasing towards better performing models, we further compute the rAULC by dividing the AULC by the AULC of a hypothetical (optimal) uncertainty estimation that perfectly orders samples according to model performance.

4.6. Additional Results

4.6.1. Image Classification

Calibration on CIFAR10. Fig. 3 shows accuracy, AUROC and rAULC against the standard deviation of additive Gaussian noise applied to the test data for various DUMs.

OOD Detection. OOD Detection. Tab. 4 shows quantitative results on detecting OOD data for DUMs, MC dropout, softmax entropy and deep ensembles trained on MNIST/FashionMNIST (similarly in the Tab. 5 for CIFAR10/SVHN). Among DUMs, SNGP and MIR demonstrate the best performance across a variety of scenarios. While DUMs confirm to be naturally suited for OOD detection, other established approaches (*e.g.* MC dropout) remain competitive.

Sensitivity to regularization strength. The hyperparameter associated with each DUM constitutes an important element in each method. It is important to know whether their performance in terms of OOD detection and calibration is sensitive to its choice. Intuitively, we expect the regularization technique to be positively correlated with OOD detection performance, since this is what most DUMs were designed for. Fig. 4 visualizes the calibration (rAULC) and OOD detection (AUROC) performance for strength

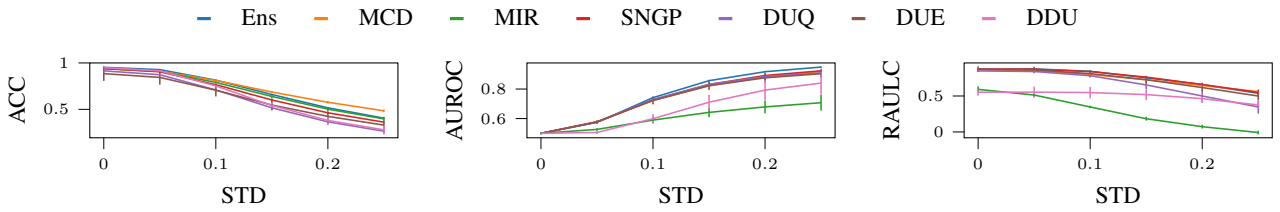


Figure 3. We compare the behaviour of various DUMs and the baselines under continuous distributional shifts. Therefore, we plot accuracy (LEFT), AUROC (CENTER) and rAULC (RIGHT) against the perturbation magnitude for a ResNet-18 trained on SVHN. We perturbe SVHN using additive Gaussian noise. The AUROC is computed against unperturbed test data.

Method	MNIST → F-MNIST		MNIST → Omniglot		F-MNIST → MNIST		F-MNIST → Omniglot	
	AUROC	AUPR	AUROC	AUPR	AUROC	AUPR	AUROC	AUPR
Softmax	89.0 ± 1.2	88.5 ± 1.4	95.1 ± 0.3	94.5 ± 0.4	73.2 ± 3.5	75.9 ± 2.6	75.9 ± 1.5	74.1 ± 1.6
Dropout	94.4 ± 0.9	92.0 ± 1.8	94.8 ± 0.2	92.0 ± 0.5	95.8 ± 0.2	93.6 ± 0.6	96.3 ± 0.2	94.3 ± 0.3
Ensemble	95.2± 1.0	91.4 ± 3.6	97.3 ± 0.4	95.3 ± 0.1	87.0 ± 1.5	81.4 ± 0.3	90.7 ± 1.1	87.6 ± 2.5
DUE	90.8 ± 2.3	90.9 ± 2.0	94.2 ± 0.2	92.8 ± 0.2	68.2 ± 1.9	65.7 ± 3.0	72.4 ± 1.7	65.8 ± 2.8
DUQ	90.2 ± 3.0	92.2 ± 2.4	93.8 ± 0.3	93.8 ± 0.4	95.1 ± 1.1	95.9 ± 1.0	94.7 ± 0.6	94.3 ± 0.7
DDU	83.9 ± 7.6	83.4 ± 7.9	75.2 ± 6.7	69.4 ± 9.5	90.8 ± 5.5	92.2 ± 4.8	90.6 ± 4.6	90.3 ± 4.6
SNGP	93.2 ± 1.2	94.6 ± 1.3	94.8 ± 0.7	93.9 ± 0.7	89.2 ± 1.1	87.9 ± 1.3	89.8 ± 1.7	85.5 ± 2.7
MIR	**97.0 ± 0.7**	**97.7 ± 0.5**	**97.3 ± 0.6**	**97.4 ± 0.5**	**99.0 ± 0.3**	**99.2 ± 0.2**	**97.9 ± 0.2**	**97.6 ± 0.4**

Table 4. Experiments for OOD detection (AUROC and AUPR, %). The baseline method and all DUMs use a MLP trained on MNIST/FashionMNIST and predict on another dataset. (F-MNIST = FashionMNIST)

of the regularization of MIR, SNGP and DUQ trained on MNIST (we provide similar plots for other datasets and methods in the supplement). Most interestingly, we find that DUMs regularizing the Lipschitz constant of the underlying feature extractor do not show evidence for a correlation between the regularization strength and the performance on OOD detection. One possible reason for this result is that Lipschitz-regularization is defined in the context of a particular predefined norm (*e.g.* L_2) (see Sec. 1). However, this norm does not necessarily represent meaningful distances on a particular dataset (*e.g.* images). On the contrary, in the case of enforcing informative representations using reconstruction regularization (MIR), we find evidence for a correlation between regularization strength and performance in terms of both OOD detection and calibration.

4.6.2. Semantic Segmentation

Weather conditions.

Examples of segmentation and uncertainty masks.

4.7. Training Details

We here provide training and optimization details for all the evaluated methods. All methods using spectral normalization do 1 power iteration. Hyperparameters were chosen to optimize the validation accuracy.

4.7.1. Digits Recognition

All methods trained on digits recognition datasets (MNIST/FashionMNIST) used a MLP as backbone with 3 hidden layers of 100 dimensions each and ReLU activation functions. We used a batch size of 128 samples and trained for 200 epochs. No data augmentation is performed.

Softmax and Deep ensembles. We used for the single softmax model the Adam optimizer with learning rate 0.003, and L_2 weight regularization 0.0001. When using ensembles, 10 models are trained from different random initializations.

MC dropout. We used for all baselines the Adam optimizer with learning rate 0.003, dropout rate 0.4 and L_2 weight regularization 0.0001.

DUE We trained DUE with the SGD optimizer with learning rate 0.01, L_2 weight regularization 0.0005, and a multi-step learning rate decay policy with decay rate 0.2 and decay steps at the epochs 60, 120, 160. We found the optimal SN coefficient to be 7, with the GP approximation using 10 (number of classes) inducing points initialized using k-means over 10000 samples.

DUQ We trained DUE with the SGD optimizer with learning rate 0.01, L_2 weight regularization 0.0001, and a multi-step learning rate decay policy with decay rate 0.3 and decay steps at the epochs 10, 20. Lengthscale for the RBF kernel is 0.1 and gradient penalty loss weight is 0.1.

Method	CIFAR10 → SVHN		CIFAR10 → STL10		SVHN → CIFAR10		SVHN → STL10	
	AUROC	AUPR	AUROC	AUPR	AUROC	AUPR	AUROC	AUPR
Softmax	83.1 ± 1.0	77.2 ± 1.8	67.4 ± 1.3	71.5 ± 1.2	93.1 ± 0.5	97.5 ± 0.2	93.6 ± 0.6	98.1 ± 0.2
Dropout	85.2 ± 0.9	76.5 ± 1.2	68.7 ± 0.3	72.8 ± 0.5	94.3 ± 0.4	98.0 ± 0.1	94.7 ± 0.3	98.5 ± 0.1
Ensemble	79.8 ± 2.9	97.6 ± 1.7	98.2 ± 1.8	85.8 ± 3.4	99.3 ± 1.1	97.7 ± 3.5	98.3 ± 0.5	96.8 ± 2.3
DUE	84.3 ± 4.7	77.4 ± 7.3	71.3 ± 2.6	75.5 ± 3.0	92.2 ± 3.3	97.0 ± 1.4	92.4 ± 3.5	97.7 ± 1.2
DUQ	76.8 ± 5.0	67.2 ± 6.3	65.1 ± 1.9	70.8 ± 1.0	90.1 ± 1.2	96.3 ± 0.4	91.1 ± 1.5	97.3 ± 0.4
DDU	69.0 ± 6.3	51.1 ± 7.4	68.9 ± 3.6	72.8 ± 3.2	73.4 ± 5.5	89.8 ± 2.4	76.7 ± 5.4	92.5 ± 2.0
SNGP	**85.3 ± 4.7**	**79.4 ± 6.5**	**76.1 ± 1.7**	**79.4 ± 1.5**	**95.8 ± 0.3**	**98.4 ± 0.1**	**96.4 ± 0.3**	**98.9 ± 0.1**
MIR	85.0 ± 6.0	71.4 ± 11.7	72.2 ± 2.1	76.3 ± 2.0	91.6 ± 1.2	97.0 ± 0.4	92.8 ± 1.2	97.9 ± 0.3

Table 5. Experiments for OOD detection's performance (AUROC and AUPR, %). The baselines and all DUMs use a ResNet-18 trained on CIFAR10/SVHN and predict on another dataset.

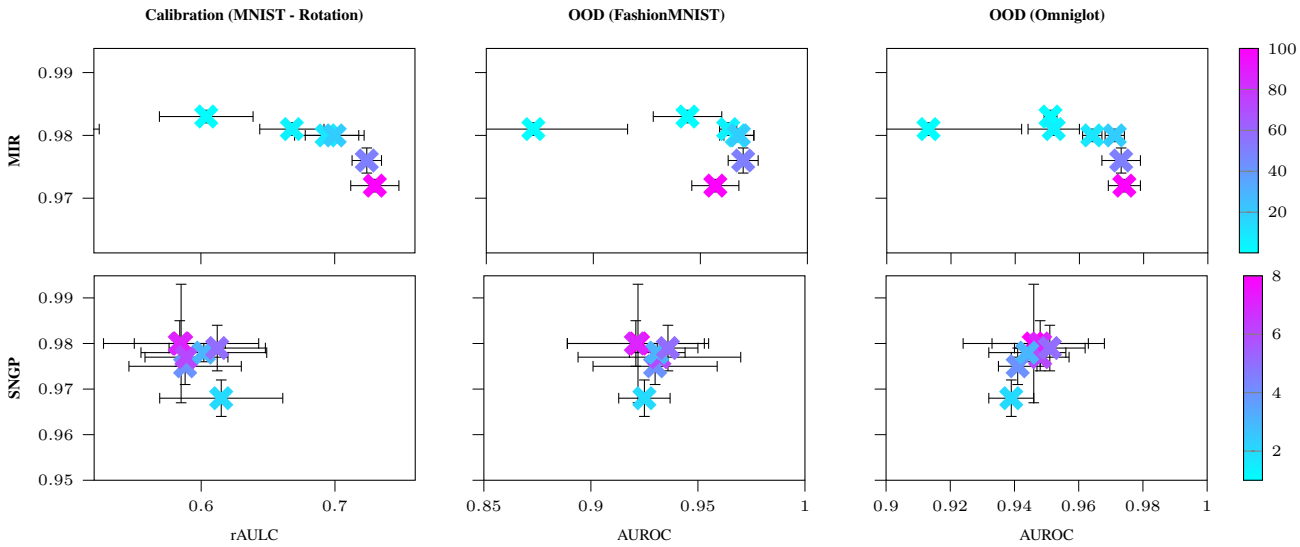


Figure 4. Trained on MNIST. Vertical axis: Test accuracy. Horizontal axis: rAULC (left), AUROC against FashionMNIST (center) and Omniglot (right) for MIR (top) and SNGP (bottom) using different regularization strength. For SNGP a larger hyperparameter corresponds to less regularization. For MIR we observe a correlation between regularization strength and performance.

DDU. We trained DDU with the Adam optimizer with learning rate 0.001, L_2 weight regularization 0.0001, and a multi-step learning rate decay policy with decay rate 0.2 and decay steps at the epochs 100, 200, 300. We found the optimal SN coefficient to be 6. The GMM fit on top of the pretrained feature extractor is trained for 100 epochs and is fit with 64 batches.

SNGP. We trained SNGP with the SGD optimizer with learning rate 0.05, L_2 weight regularization 0.0003, and a multi-step learning rate decay policy with decay rate 0.2 and decay steps at the epochs 60, 120, 160. We found the optimal SN coefficient to be 6, with the GP approximation using 10 hidden dimensions, lengthscale 2 and mean field factor 30.

MIR. We trained MIR with the Adam optimizer with learning rate 0.001, and L_2 weight regularization 0.0001. We found the optimal reconstruction loss weight to be 1.

4.7.2. Image Classification

When training on image classification datasets (CIFAR-10/SVHN), we use a ResNet-18 (He et al., 2016) as backbone. The dimensionality of the last feature space encoded with the ResNet backbone is 100 for all methods. We used a batch size of 128 samples and trained for 400 epochs. The training set is augmented with common data augmentation techniques. We apply random horizontal flips, random brightness augmentation with maximum delta 0.2 and random contrast adjustment with multiplier lower bound 0.8 and upper bound 1.2.

Softmax and Deep ensembles. We used for the single softmax model the Adam optimizer with learning rate 0.003, L_2 weight regularization 0.0001, and a multi-step learning rate decay policy with decay rate 0.2 and decay steps at the epochs 250, 300, 400. When using ensembles, 10 models are trained from different random initializations.

MC dropout. We used for all baselines the Adam optimizer with learning rate 0.003, dropout rate 0.3, L_2 weight regularization 0.0001, and a multi-step learning rate decay policy with decay rate 0.2 and decay steps at the epochs 250, 300, 400.

DUE We trained DUE with the SGD optimizer with learning rate 0.01, L_2 weight regularization 0.0005, dropout rate 0.1, and a multi-step learning rate decay policy with decay rate 0.2 and decay steps at the epochs 100, 200, 300. We found the optimal SN coefficient to be 7 for SVHN and 9 for CIFAR-10, with the GP approximation using 10 (number of classes) inducing points initialized using k-means over 10000 samples.

DUQ We trained DUE with the SGD optimizer with learning rate 0.01, L_2 weight regularization 0.0001, dropout rate 0.1, and a multi-step learning rate decay policy with decay rate 0.3 and decay steps at the epochs 200, 250. 300. Lengthscale for the RBF kernel is 0.1 and gradient penalty loss weight is 0.1.

DDU. We trained DDU with the Adam optimizer with learning rate 0.001, L_2 weight regularization 0.0001, dropout rate 0.3, and a multi-step learning rate decay policy with decay rate 0.2 and decay steps at the epochs 80, 120, 180. We found the optimal SN coefficient to be 7. The GMM fit on top of the pretrained feature extractor is trained for 100 epochs and is fit with 64 batches.

SNGP. We trained SNGP with the SGD optimizer with learning rate 0.05, L_2 weight regularization 0.0004, dropout rate 0.1, and a multi-step learning rate decay policy with decay rate 0.2 and decay steps at the epochs 100, 200, 300. We found the optimal SN coefficient to be 7, with the GP approximation using 10 hidden dimensions, lengthscale 2 and mean field factor 30.

MIR. We trained MIR with the Adam optimizer with learning rate 0.003, L_2 weight regularization 0.0001, dropout rate 0.1, and a multi-step learning rate decay policy with decay rate 0.2 and decay steps at the epochs 150, 200, 250, 300. We found the optimal reconstruction loss weight to be 1.

4.7.3. Semantic Segmentation.

When training on image classification datasets (CIFAR-10/SVHN), we use a ResNet-18 (He et al., 2016) as backbone. We used a batch size of 8 samples and trained for 200 epochs. Images are rescaled to size 200×320 The training set is augmented with common data augmentation techniques. All training samples are augmented with random cropping with factor 0.8. We apply random horizontal flips, random brightness augmentation with maximum delta 0.2 and random contrast adjustment with multiplier lower bound 0.8 and upper bound 1.2.

Softmax. We used for the single softmax model the Adam optimizer with learning rate 0.0004, L_2 weight regularization 0.0001, and a multi-step learning rate decay policy with decay rate 0.3 and decay steps at the epochs 30, 60, 90, 120.

MC dropout. We used for all baselines the Adam optimizer with learning rate 0.0004, dropout rate 0.4, L_2 weight regularization 0.0001, and a multi-step learning rate decay policy with decay rate 0.3 and decay steps at the epochs 30, 60, 90, 120.

SNGP. We trained SNGP with the SGD optimizer with learning rate 0.0002, L_2 weight regularization 0.0003, dropout rate 0.1, and a multi-step learning rate decay policy with decay rate 0.2 and decay steps at the epochs 20, 40, 60, 80, 100. We found the optimal SN coefficient to be 6, with the GP approximation using 128 hidden dimensions, lengthscale 2 and mean field factor 25.

MIR. We trained MIR with the Adam optimizer with learning rate 0.0002, L_2 weight regularization 0.0001, dropout rate 0.1, and a multi-step learning rate decay policy with decay rate 0.3 and decay steps at the epochs 30, 60, 90, 120. We found the optimal reconstruction loss weight to be 1.

4.8. Implementation Details.

All methods were re-implemented in Tensorflow 2.0. We payed attention to all the details reported in each paper and we run all experiments for each method multiple times to check for stochasticity, *i.e.* 5 times for classification and 3 times for segmentation. When an implementation was publicly available, we relied on it. This is the case for DUQ (https://github.com/y0ast/deterministic-uncertainty-quantification), SNGP (https://github.com/google/uncertainty-baselines/blob/master/baselines/imagenet/sngp.py) and DUE (https://github.com/y0ast/DUE).

SNGP. We follow the publicly available implementation of SNGP, which, compared to the implementation described in the original paper, proposes to further reduce the computational overhead of the GP approximation by replacing the Monte-Carlo averaging with the mean-field approximation (Daunizeau, 2017). This is especially relevant in large-scale tasks like semantic segmentation, were it is important to reduce the computational overload.

4.8.1. Image Classification

DUE. Please notice that only DUE uses a SN approximation also for the batch normalization layer. All other methods only restrict the Lipschitz constant of convolutional and fully connected layers.

MIR only differs from regular softmax models in its decoder module used for the reconstruction regularization

loss (Postels et al., 2020). When training MLP architectures the decoder is comprised of two fully-connected layer. The first has a ReLU activation function and 200 output neruons. The second has a linear activation function and its output dimensionality equals that of the models' input data. When training convolutional neural networks the decoder is comprised of four blocks of transpose convolutions, batch-normalization layers and ReLU activation functions that gradually upscale the hidden representations to the dimensionality of the input data. These four block are followed by a 1x1 convolution with linear activation function.

4.8.2. Semantic Segmentation

SNGP. DRN uses 1×1 convolutions at the last layer to map the latest feature map to the predicted segmentation mask. This works under the assumption that all pixels in the output mask are i.i.d. random variables. Following this intuition, we extend SNGP to semantic segmentation by fitting a $GP : \mathbb{R}^Z \to \mathbb{R}^C$ at pixel level that maps from the deep feature dimension z to the number of classes c. By keeping the GP kernel parameters shared across all pixels, we simulate a 1×1 convolution of GPs, *i.e.* $\sigma : (H \times W \times Z) \to (H \times W \times C)$, where σ is the convolution of GPs operation, H and W are respectively image height and width, Z is the number of latent features and C is the number of output classes. For details about the GP we refer to (Liu et al., 2020) or the supplement.

MIR. To estimate image-level uncertainty MIR requires fitting the distribution of hidden representations. We fit a Gaussian mixture model (GMM) with 20 components (*i.e.* number of classes) to each spatial location of the hidden representations using features extracted from the training data independently. This assumes that the distribution is translation invariant and factorizes along the spatial dimensions of the latent space which is similar to the procedure used in (Blum et al., 2019). Similar to image classification, MIR only differs from regular segmentation models in its decoder module used for the reconstruction regularization loss (Postels et al., 2020). The decoder module is comprised of a single point-wise feed forward layer that maps the hidden representations $\mathbf{z} \in \mathbb{R}^{W_z \times H_z \times C_z}$ to $\mathbf{z} \in \mathbb{R}^{W_z \times H_z \times 3}$. Subsequently, the result is bilinearly upsampled to the image resolution on which we compute the reconstruction loss.

4.8.3. Uncertainty Derivation.

We here provide details on the procedure to estimate uncertainty for the baseline methods. For details on the uncertainty derivation in DUMs, please refer to Sec. 1 of the main paper or to the original paper of each amalysed method.

Softmax. In case of the softmax baseline we estimate uncertainty using the entropy of the predictive distribution parameterized by the neural network. Given an input $\mathbf{x}$ the entropy H is given by $H(\mathbf{y}|\mathbf{x}) = \sum -p(\mathbf{y}|\mathbf{x}) \log(p(\mathbf{y}|\mathbf{x}))$ where $p(\mathbf{y}|\mathbf{x})$ are the softmax probabilities.

MC dropout and deep ensembles. We following (Gal et al., 2017) and compute epistemic uncertainty as the conditional mutual information between the weights $\mathbf{w}$ and the predictions $\hat{y}$. Given an input $\mathbf{x}$ and a set of weights $\mathbf{w}$ we observe the predictive distribution $p(\hat{y}|x, w)$. Then epistemic uncertainty u_{ep} is calculated by approximating the mutual information conditioned on the input $\mathbf{x}$:

$$\begin{aligned} u_{ep} &= I\left(\hat{y}, w|x\right) \\ &= H(\hat{y}|x) - H(\hat{y}|w, x) \\ &= E_{y\sim p(\hat{y}|x)}\left[-\log(p(\hat{y}|x))\right] - u_{al} \end{aligned}$$

Here, $p(\hat{y}|x) = \int \mathrm{d}w p(w) p(\hat{y}|x, w)$ is evaluated using a finite set of samples/ensemble members.

4.8.4. Uncertainty Derivation for Semantic Segmentation.

We here derive uncertainty estimates for each method for semantic segmentation. While for Softmax, MC dropout and SNGP we average pixel-level uncertainties under the assumption that all pixels are represented by i.i.d. variables, for MIR we rely on the log-likelihood of the estimated feature space distribution.

Uncertainty Averaging. In our experiments on continuous distributional shifts we want to estimate a global uncertainty for the output map and are not interested in pixel-level confidence. Therefore, we propose to approximate the uncertainty of the predicted segmentation masks as the average of all pixel-level uncertainties, i.e.,

$$\begin{aligned} H(\mathbf{y} \mid \mathbf{x}) &= \mathbb{E}_{\mathbf{x}}\left[-\int_{\mathbf{y}} p(\mathbf{y} \mid \mathbf{x}) \log p(\mathbf{y} \mid \mathbf{x}) \mathrm{d}\mathbf{y}\right] \\ &= \mathbb{E}_{\mathbf{x}}\left[-\int_{\mathbf{y}} p(\mathbf{y} \mid \mathbf{x}) \log p(y_1 \mid \mathbf{x}) \mathrm{d}\mathbf{y}\right] + \cdots \\ &+ \mathbb{E}_{\mathbf{x}}\left[-\int_{\mathbf{y}} p(\mathbf{y} \mid \mathbf{x}) \log p(y_n \mid \mathbf{x}) \mathrm{d}\mathbf{y}\right] \\ &\approx \frac{1}{n} \sum_{i=1}^{n} H(y_i \mid \mathbf{x}) \end{aligned} \tag{1}$$

MIR estimates epistemic uncertainty using the likelihood of hidden representations $\mathbf{z} \in \mathbb{R}^{W_z \times H_z \times C_z}$. Since $\mathbf{z}$ is high-dimensional in our experiments, we assume that it factorizes along W_z and H_z and is translation invariant. Formally, $p(\mathbf{z}) = \prod_i^{W_z} \prod_j^{H_z} p_\theta(\mathbf{z}_{ij})$ where $\mathbf{z}_{ij} \in \mathbb{R}^{W_z \times H_z}$ and θ is shared across W_z and H_z.

We parameterize p_θ with GG with $n = 10$ components where each component has a full covariance matrix. We fit the GMM on 100000 hidden representations ($\mathbf{z}_{ij} \in \mathbb{R}^{C_z}$)

randomly picked from the training dataset post-training. Since $C_z = 1024$ is still high-dimensional, we first apply PCA to reduce its dimensionality to 32.

In the dilated resnet architecture used for semantic segmentation the latent representation $\mathbf{z}$ is passed through a pointwise feedforward layer $f : \mathbb{R}^{W_z \times H_z \times C_z} \mapsto \mathbb{R}^{W_z \times H_z \times 3}$ and, subsequently, bilinearly upsampled to image resolution ($\mathbb{R}^{W \times H \times K}$) where K is the number of classes. When could estimate the global, *i.e.*image-level, uncertainty of an input, by providing the negative log-likelihood of the factorizing distribution. However, in order to also obtain pixel-wise uncertainties using MIR, we first compute the negative log-likelihood (*i.e.*epistemic uncertainty) associated with each latent representation $\mathbf{z}_{ij}$. Then, we bilinearly upsample the negative log-likelihoods and use the result as proxy for pixel-wise epistemic uncertainty. To obtain a global, *i.e.*image-level, uncertainty we apply the same averaging scheme used for SNGP, softmax and MC dropout.

4.9. Dataset

To benchmark our model on data with realistically and continuously changing environment, we collect a synthetic dataset for semantic segmentation. We use the CARLA Simulator (Dosovitskiy et al., 2017) for rendering the images and segmentation masks. The classes definition is aligned with the CityScape dataset (Cordts et al., 2016). In order to obtain a fair comparison, all the OOD data are sampled with the same trajectory and the environmental objects, except for the time-of-the-day or weather parameters.

In-domain data The data are collected from 4 towns in CARLA. We produce 32 sequences from each town. The distribution of the vehicles and pedestrians are randomly generated for each sequence. Every sequence has has 500 frames of data with the sampling rate at 10 FPS. From them we randomly sample the training and validation set.

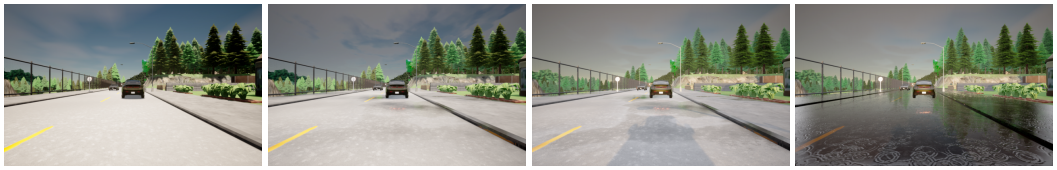
Out-of-domain data Here, we consider the time-of-the-day and the rain strength as the parameters for the continuous changing environment. In practice, these two parameters have the major influence for autonomous driving tasks.

The change of the time-of-the-day is illustrated in Fig. 5. The time-of-the-day is parametrized by the Sun's altitude angle, where $90°$ means the mid-day and the $0°$ means the dust or dawn. Here, we produce samples with the altitude angle changes from $90°$ to $15°$ by step of $5°$, and $15°$ to $-5°$ by step of $1°$ where the environment changes shapely. From these examples, we can confirm that the change of time-of-the-day leads to the major change in the lightness, color and visibility of the sky, roads and the buildings nearby. The effect of rain strength is demonstrated in Fig. 6. Here the cloudiness and, ground wetness and ground reflection are the main changing parameters.

(a) Mid-day (90°) (b) Afternoon (45°) (c) Afternoon (15°) (d) Evening (10°)

(e) Dust (2°) (f) Sunset (0°) (g) Sunset (-2°) (h) Night ($\leq -5^\circ$)

Figure 5. Changing of the time-of-the-day



(a) Clear (b) Cloudy (c) Small rain (d) Heavy rain

Figure 6. Changing of the weather