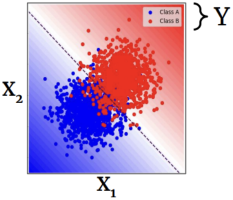
Uncertainty Quantification in Deep Learning

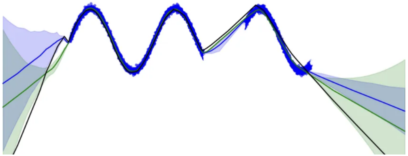
Dissertation Defense

Paris Dauphine - Université PSL
Nathaniel Cogneaux

What do we mean by uncertainty?

When can we trust the model's predictions?





- **Classification:** Output label along with its confidence
- **Regression:** Output mean along with its variance

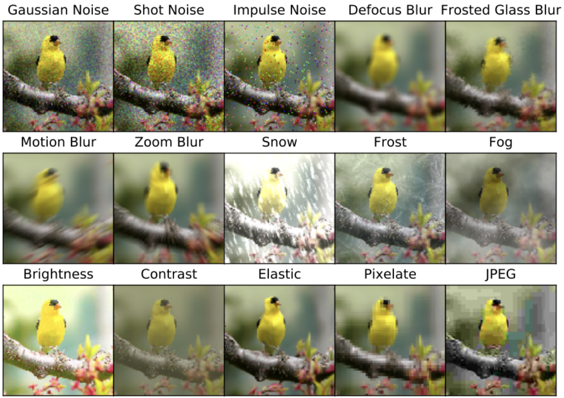
Usual assumption in machine learning:

$$\mathbb{P}_{\text{test}}(y, x) = \mathbb{P}_{\text{train}}(y, x)$$

In reality:

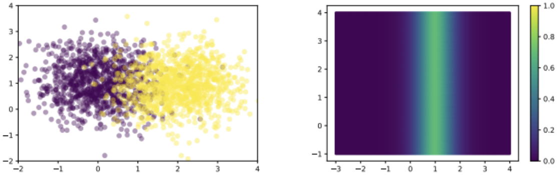
$$\mathbb{P}_{\text{test}}(y, x) \neq \mathbb{P}_{\text{train}}(y, x)$$

What do we mean by uncertainty?

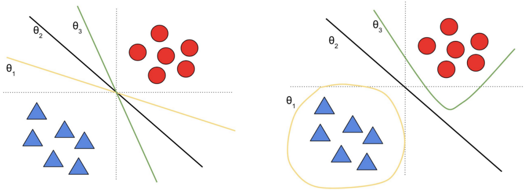


ImageNet-C, common corruptions and perturbations (Hendrycks and Dietterich [2019])

Different types of uncertainties

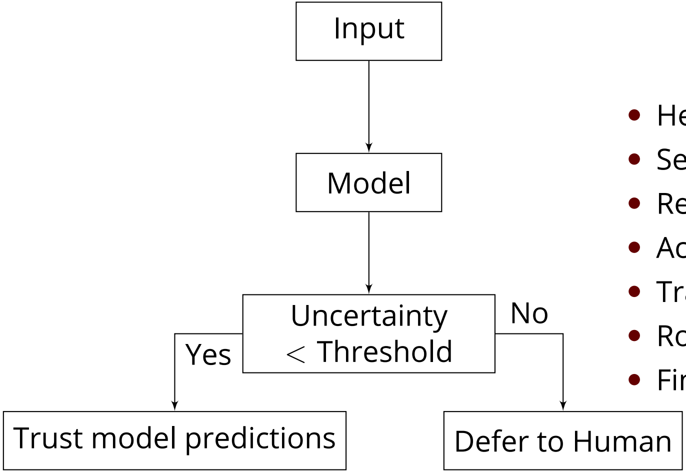


Aleatoric Uncertainty: Class overlap causing ambiguous decision boundaries in classification.



Epistemic Uncertainty: Illustrations of model uncertainty in classification.

Applications



- Healthcare
- Self-driving cars
- Reinforcement learning
- Active inference
- Transfer learning
- Robotics
- Finance and risk assessment

Overview & Contributions

- **Literature Review:** Covers Bayesian, ensemble, EDL, and post-hoc UQ methods
- **Research Gap:** Current methods are computationally expensive and require architectural modifications, highlighting the need for a post-hoc and efficient model-agnostic solution.
- **Proposed Solution:** Introduces a multi-output module for efficient UQ in pre-trained models, without retraining.
- **Results:** Achieves near state-of-the-art performance on MNIST and CIFAR datasets, with reduced computational costs.

Bayesian Modelling

Usual models yields only a **single** prediction $\Rightarrow$ Bayesian approach: define the model likelihood $\mathbb{P}(y|x,\omega)$.
The goal is to find the best set of parameters ω such that

$$\omega^* = \arg\max_{\omega} \mathbb{P}(\omega \mid x, y)$$

This is equivalent to

$$= \arg\min_{\omega} -\log \mathbb{P}(y \mid x, \omega) - \log \mathbb{P}(\omega)$$

Bayesian Modelling

Posterior distribution, $\mathbb{P}(\omega|X, Y)$ obtained by applying Bayes' theorem:

$$\mathbb{P}(\omega|X, Y) = \frac{\mathbb{P}(Y|X,\omega)\mathbb{P}(\omega)}{\mathbb{P}(Y|X)}.$$

Then, for a given test sample x^*, the class label with respect to $\mathbb{P}(\omega|X, Y)$ can be predicted by:

$$\mathbb{P}(y^*|x^*, X, Y) = \int \mathbb{P}(y^*|x^*,\omega)\mathbb{P}(\omega|X, Y)d\omega.$$

Basics of disentanglement:

$$PU = EU + AU$$

Uncertainty disentanglement in Bayesian Modelling

$$\mathbb{P}(y^*|x^*, X, Y) = \int \underbrace{\mathbb{P}(y^*|x^*, \omega)}_{\text{Aleatoric}} \underbrace{\mathbb{P}(\omega|X, Y)}_{\text{Epistemic}} d\omega.$$

With entropy (Gal and Ghahramani [2016]):

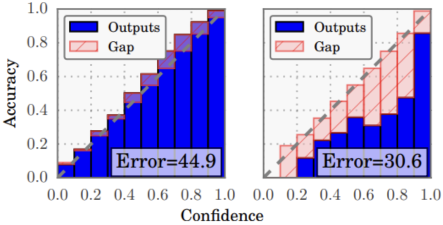
$$\mathbb{H}[y^*|x^*, D_{tr}] - \mathbb{E}_{\mathbb{P}(\omega|D_{tr})}[\mathbb{H}(y^*|\omega, x^*)] = I(y^*, \omega|x^*, D_{tr})$$

With the law of total variance (Depeweg et al. [2018]):

$$\sigma^2(y^*|x^*, D_{tr}) = \sigma^2_{\mathbb{P}(\omega|D_{tr})}(\mathbb{E}[y^*|\omega, x^*]) + \mathbb{E}_{\mathbb{P}(\omega|D_{tr})}[\sigma^2(y^*|\omega, x^*)]$$

In practice: posterior is intractable

Calibration Errors & Proper Scoring Rules



Expected Calibration Error (ECE):

$$\mathrm{ECE} = \sum_{b=1}^{B} \frac{n_b}{N} \left|\mathrm{acc}(b) - \mathrm{conf}(b)\right|$$

Maximum Calibration Error (MCE):

$$\mathrm{MCE} = \max_{b\in\{1,\dots,B\}} \left|\mathrm{acc}(b) - \mathrm{conf}(b)\right| .$$

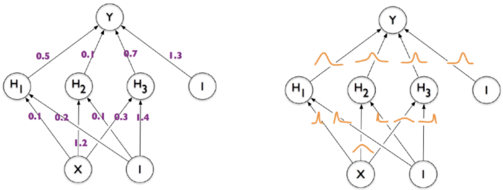
Negative Log-Likelihood (NLL):
The NLL is a proper scoring rule for probabilistic models:

$$\mathrm{NLL} = -\sum_{n=1}^{N} \log \mathbb{P}(y_n|\mathbf{x}_n, \omega)$$

Brier Score (BS):
Quadratic penalty for difference between predicted probabilities and outcomes (Gneiting and Raftery [2007]):

$$\mathrm{BS} = \frac{1}{|\mathcal{Y}|} \sum_{y\in\mathcal{Y}} \left[\mathbb{P}(y|\mathbf{x}_n, \omega) - \delta(y - y_n)\right]^2$$

Bayesian Neural Networks (BNNs)



$$\mathbb{H}[y^* | x^*, D_{tr}] - \mathbb{E}_{q(\omega)}[\mathbb{H}(y^* | \omega, x^*)] = I(y^*, \omega | x^*, D_{tr})$$

$$\mathrm{Var}(y) \approx \underbrace{\frac{1}{T}\sum_{t=1}^{T} \hat{y}_t^2 - \left(\frac{1}{T}\sum_{t=1}^{T} \hat{y}_t\right)^2}_{\text{Epistemic}} + \underbrace{\frac{1}{T}\sum_{t=1}^{T} \hat{\sigma}_t^2}_{\text{Aleatoric}} .$$

Variational Inference:

$$q_\theta(\omega) = \mathcal{N}(\omega | \mu, \Sigma) = \prod_{i=1}^{D} \mathcal{N}(\omega_i | \mu_i, \sigma_i)$$

$$KL(q_\theta(\omega) \parallel \mathbb{P}(\omega | X, Y)) = \int q_\theta(\omega) \log \frac{q_\theta(\omega)}{\mathbb{P}(\omega | X, Y)} \, d\omega$$

Monte Carlo Dropout:

$$\mathrm{Var}(y) \approx \underbrace{\sigma^2}_{\text{Aleatoric}} + \underbrace{\frac{1}{T}\sum_{t=1}^{T} f_{\hat{\omega}_t}(x)^T f_{\hat{\omega}_t}(x) - \left(\frac{1}{T}\sum_{t=1}^{T} f_{\hat{\omega}_t}(x)\right)^2}_{\text{Epistemic}}$$

Predictive Mean/Variance:

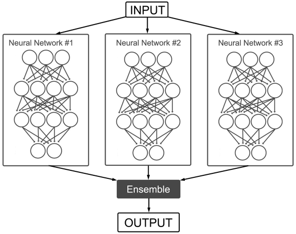
$$[\hat{y}, \hat{\sigma}^2] = f^{\hat{\omega}}(x), \quad \mathcal{L}_{BNN}(\theta) = \frac{1}{D}\sum_{i} \left(\frac{1}{2}\hat{\sigma}_i^{-2} \|y_i - \hat{y}_i\|^2 + \frac{1}{2}\log \hat{\sigma}_i^2\right)$$

Ensembles for Uncertainty Quantification

Ensembles Overview:
Ensemble methods aggregate multiple models with different parameter settings to improve robustness and capture uncertainty (Dietterich [2000]). Examples include:

- **Monte Carlo Dropout** (Gal and Ghahramani [2016]) approximates Bayesian inference by applying dropout during training and inference.
- **Bagging** (Breiman [1996]) trains models on bootstrap samples to reduce variance.
- **Deep Ensembles** (Lakshminarayanan et al. [2017]) trains multiple neural networks independently to capture uncertainty.

Deep Ensembles consistently outperform other UQ methods



Ensemble Prediction:

$$\hat{y} = \frac{1}{M}\sum_{i=1}^{M} f^{\omega^{(i)}}(\mathbf{x})$$

Uncertainty Decomposition (Variance):

$$\mathrm{Var}(\hat{y}) = \underbrace{\frac{1}{M}\sum_{i=1}^{M} \sigma^{2,\omega^{(i)}}(\mathbf{x})}_{\text{Aleatoric}} + \underbrace{\frac{1}{M}\sum_{i=1}^{M} \mu^{\omega^{(i)}}(\mathbf{x})^2 - \left(\frac{1}{M}\sum_{i=1}^{M} \mu^{\omega^{(i)}}(\mathbf{x})\right)^2}_{\text{Epistemic}}$$

Uncertainty Decomposition (Entropy):

$$\mathbb{H}(\hat{y}) = \underbrace{\mathbb{E}[\mathbb{H}(\hat{y}|\omega)]}_{\text{Aleatoric}} + \underbrace{\mathbb{I}(\hat{y};\omega)}_{\text{Epistemic}}$$

Hierarchical Methods - Evidential Deep Learning

Evidential Deep Learning (EDL):

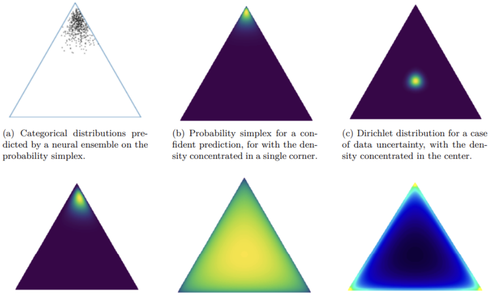
- Predicts a distribution over class probabilities using the Dirichlet distribution.
- Neural networks predict concentration parameters (α) for the Dirichlet.
- Produces both the mean prediction and uncertainty estimate simultaneously.
- Final prediction is derived from the mean of the Dirichlet-distributed probabilities.

Key Formula:

$$\alpha = \exp\left(f_\omega(\mathbf{x})\right), \quad \pi_k = \frac{\alpha_k}{\alpha_0}, \quad \hat{y} = \arg\max_{k \in \mathcal{K}} \pi_k$$

Uncertainty disentanglement:

$$I\left[y, \pi \mid \mathbf{x}, \mathcal{D}\right] = \mathbb{H}\left[\mathbb{E}_{\mathbb{P}(\pi \mid \mathbf{x}, \mathcal{D})}\left[\mathbb{P}(y \mid \pi)\right]\right] - \mathbb{E}_{\mathbb{P}(\pi \mid \mathbf{x}, \mathcal{D})}\left[\mathbb{H}\left[\mathbb{P}(y \mid \pi)\right]\right]$$



(a) Categorical distributions predicted by a neural ensemble on the probability simplex.

(b) Probability simplex for a confident prediction, for with the density concentrated in a single corner.

(c) Dirichlet distribution for a case of data uncertainty, with the density concentrated in the center.

(d) Dirichlet distribution for a case of model uncertainty, with the density spread out more.

(e) Dirichlet for a case of distributional uncertainty, with the density spread across the whole simplex.

(f) Alternative approach to distributional uncertainty called representation gap, with density concentrated along the edges.

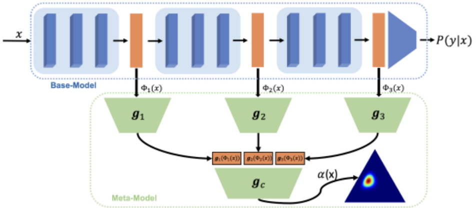
Post-hoc Single-Pass Uncertainty Quantification methods

Key Methods:

- **Conformal Prediction:** Provides prediction intervals but can't distinguish between aleatoric and epistemic uncertainty [Shafer and Vovk, 2008].
- **Temperature Scaling:** Adjusts softmax outputs using a temperature parameter for better calibration [Guo et al., 2017].
- **Bayesian Meta-Modeling:** Improves uncertainty quantification without retraining, capturing both total and epistemic uncertainty [Shen et al., 2022].

Challenges:

- Both conformal prediction and temperature scaling assume consistent data distribution.
- Conformal methods often produce overly wide prediction intervals in high-dimensional spaces.
- Bayesian meta-model approaches, while promising, still face challenges in capturing second-order uncertainty [Bengs et al., 2023].



Meta-Model Structure [Shen et al., 2022].

$$I(y, \pi \mid \Phi(\mathbf{x})) = \mathcal{H}\left(\mathbb{E}[\mathbb{P}(y \mid \pi)]\right) - \mathbb{E}[\mathcal{H}(\mathbb{P}(y \mid \pi))]$$

Method Description

Context:

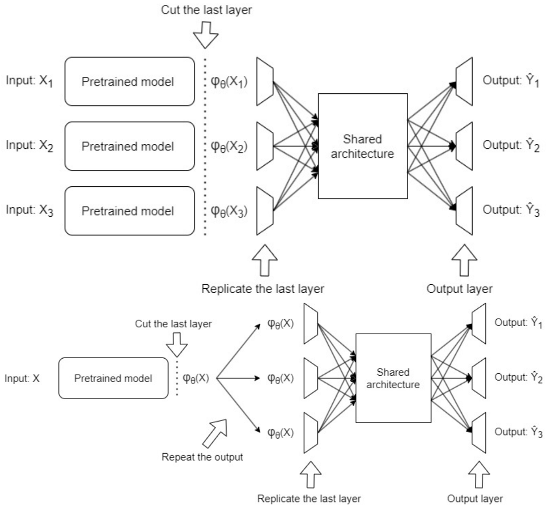
- The base model $h_{II} \circ \varphi$ maps input $\mathcal{X}$ to predicted label distributions $\mathbb{P}_B(y|\varphi(x))$.
- To improve uncertainty estimation, a meta-model is created on top of the base model without retraining it.
- The last layer is duplicated M times, creating an ensemble of input heads $h_i(\varphi(x))$.
- These heads are processed through a shared fully connected layer and produce logits, which are turned into probabilities with softmax.

Training:

- The module is trained on penultimate layer features, mapping multiple inputs to multiple outputs at once.
- The loss function minimizes a sum of log-likelihoods, regularized by $R(\omega)$.
- It learns a joint distribution between penultimate layer activations and predicted classes.

Inference:

- At evaluation, activations $\varphi(x')$ are repeated M times.
- Each head approximates $\mathbb{P}_\omega(y_i|\varphi(x'))$, and the final output is averaged across the M heads.
- This produces predictions similar to Deep Ensembles, and uncertainty is estimated via the variance of these outputs.



Uncertainty Quantification and Disentanglement

Key Idea:
Each of the M softmax outputs from the model heads contributes to uncertainty estimation. Uncertainty is decomposed into:

- **Aleatoric Uncertainty:** Ambiguity in data, reflected when heads predict confidently but differently.
- **Epistemic Uncertainty:** Lack of knowledge or insufficient training, captured by high entropy across all heads.

Key Points:

- At inference, we get M softmax outputs from the heads:

$$\{\mathbf{p}^m\}_{m=1}^M = \left\{(p_1^m, p_2^m, \ldots, p_K^m)\right\}_{m=1}^M ,$$

- The mean prediction across heads is $\bar{\mathbf{p}}$, and the total predictive uncertainty is the entropy of $\bar{\mathbf{p}}$.
- Epistemic uncertainty is represented by the average entropy, while aleatoric uncertainty is quantified by the KL divergence between each head's output and the mean.

Mean Softmax Prediction:

$$\bar{\mathbf{p}} = \frac{1}{M}\sum_{m=1}^{M} \mathbf{p}^m, \quad \bar{p}_i = \frac{1}{M}\sum_{m=1}^{M} p_i^m$$

Total Predictive Uncertainty:

$$\mathbb{H}(\bar{\mathbf{p}}) = -\sum_{i=1}^{K} \bar{p}_i \log \bar{p}_i$$

Entropy of m-th Head:

$$\mathbb{H}(\mathbf{p}^m) = -\sum_{i=1}^{K} p_i^m \log p_i^m$$

Uncertainty Decomposition:

$$\mathbb{H}(\bar{\mathbf{p}}) = \underbrace{\frac{1}{M}\sum_{m=1}^{M} \mathbb{H}(\mathbf{p}^m)}_{\text{Epistemic Uncertainty}} + \underbrace{\frac{1}{M}\sum_{m=1}^{M} \mathrm{KL}(\mathbf{p}^m \| \bar{\mathbf{p}})}_{\text{Aleatoric Uncertainty}}$$

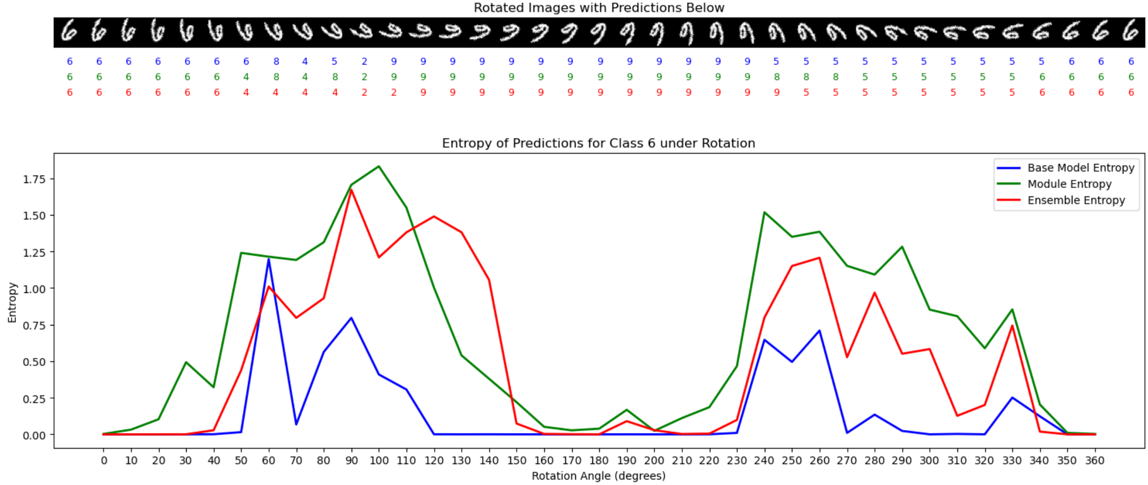
Cifar10 dataset

Model	Accuracy (%)	NLL (%)	ECE (%)	cA (%)	cNLL (%)	cECE (%)	Parameters	Forward Passes
Multi-Output k heads compared to Base Model								
Multi-Output 3 heads	+0.07	-12.91	-23.90	+0.27	-16.23	-14.55	36.5M	1
Multi-Output 5 heads	+0.05	-16.95	-35.66	+0.38	-20.65	-21.98	36.5M	1
Multi-Output 7 heads	+0.07	-18.80	-48.53	+0.26	-23.57	-28.30	36.5M	1
Multi-Output 10 heads	+0.09	-20.41	-62.50	+0.39	-27.11	-34.94	36.6M	1
Baselines compared to Deterministic								
BatchEnsemble (size=4)	+0.31	-13.99	-23.37	+2.23	-7.62	-18.95	36.6M	4
Hyper-BatchEnsemble (size=4)	+0.31	-20.75	-60.17	-	-	-	73.1M	4
MIMO	+0.42	-22.01	-56.28	+0.66	-6.96	-26.80	36.5M	1
Rank-1 BNN (Gaussian, size=4)	+0.31	-19.87	-65.40	+0.79	-7.69	-47.71	36.6M	4
Rank-1 BNN (Cauchy, size=4)	+0.52	-24.53	-60.17	+4.41	-20.67	-41.18	36.6M	4
SNGP	+0.00	-15.72	-69.57	+2.42	-5.73	-49.02	37.5M	1
SNGP, with AugMix	**+0.94**	**-35.22**	**-80.53**	**+13.07**	-68.57	**-90.20**	37.5M	1
SNGP, with MC Dropout (size=10)	-0.10	-17.61	-65.29	+1.58	-7.62	-46.41	37.5M	10
SNGP, with BatchEnsemble (size=4)	+0.21	-20.12	-73.94	+2.00	-7.62	-47.71	37.5M	4
SNGP Ensemble (size=4)	+0.70	-31.45	-78.36	+3.39	-7.62	-51.63	150M	4
Monte Carlo Dropout (size=1)	-0.10	+0.62	+4.33	-7.31	+1.90	+8.50	36.5M	1
Monte Carlo Dropout (size=30)	+0.10	-8.81	-17.84	+1.58	-1.10	-9.77	36.5M	30
Monte Carlo Dropout, improved (size=30)	+0.21	-27.04	-78.40	+3.66	**-68.82**	-55.54	36.5M	30
Ensemble (size=4)	+0.63	-28.30	-56.72	+3.49	-7.62	-43.14	146M	4
Hyper-deep ensemble (size=4)	+0.63	-24.53	-60.17	+4.02	-7.62	-48.37	146M	4
Variational inference (sample=1)	-1.35	+32.70	+25.32	-6.32	+39.05	+18.30	73M	1
Posterior Network	-3.02	+126.42	+385.28	-1.18	+51.43	+33.51	36.6M	1

Cifar100 dataset

Model	**Accuracy (%)**	**NLL (%)**	**ECE (%)**	**cA (%)**	**cNLL (%)**	**cECE (%)**	**Parameters**	**Forward Passes**
Multi-Output k heads compared to Base Model								
Multi-Output 3 heads	+0.77	-24.83	-41.34	+2.89	-29.16	-32.48	36.9M	1
Multi-Output 5 heads	+0.96	-28.30	-57.54	+3.17	-33.67	-42.42	37.4M	1
Multi-Output 7 heads	+0.66	<u>-29.05</u>	-66.54	+2.75	-35.46	-48.52	38.0M	1
Multi-Output 10 heads	+0.53	**-29.79**	-77.72	+2.98	<u>-38.01</u>	-56.99	39.2M	1
Baselines compared to Deterministic								
BatchEnsemble (size=4)	+2.63	-21.14	-69.08	+1.73	-5.19	-37.66	36.6M	4
Hyper-BatchEnsemble (size=4)	+2.63	-22.51	-76.70	-	-	-	36.6M	4
MIMO	+2.75	-21.14	-74.33	+2.33	+3.56	-46.03	36.5M	1
Rank-1 BNN (Gaussian, size=4)	+1.88	-20.91	-79.00	+2.43	+3.67	-51.05	36.6M	4
Rank-1 BNN (Cauchy, size=4)	+3.26	-21.20	<u>-86.00</u>	<u>+6.41</u>	-24.44	-40.59	36.6M	4
SNGP	+0.50	-7.80	-76.66	+0.50	-25.19	-61.34	37.5M	1
SNGP, with AugMix	+0.97	-5.91	-71.99	**+14.53**	**-52.63**	**-77.43**	37.5M	1
SNGP Ensemble (size=4)	+2.13	-24.00	**-87.15**	+5.43	-24.44	<u>-62.10</u>	150M	4
Monte Carlo Dropout (size=1)	-0.25	-0.94	-41.51	-7.74	+7.41	-15.93	36.5M	1
Ensemble (size=4)	<u>+3.64</u>	-23.89	-75.49	+2.90	-16.11	-43.84	146M	4
Hyper-deep ensemble (size=4)	**+4.02**	-25.31	-74.30	+3.00	-24.44	-46.44	146M	4
Variational inference (sample=1)	-2.51	+7.89	+13.88	-8.13	+17.78	+13.39	73M	1
Heteroscedastic	+0.50	-5.14	-31.12	+0.24	-2.88	-25.73	37M	1
Heteroscedastic Ensemble (size=4)	+2.38	-23.67	-69.65	+1.75	-7.38	-56.89	148M	4

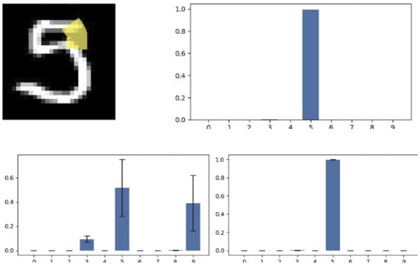
OOD detection



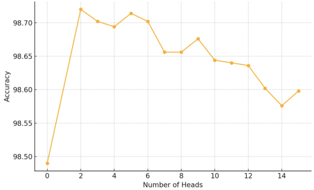
Understanding the method

Model	nb epochs	training time	nb parameters
CIFAR-10			
Multi-Output 3 heads	36.60 ± 11.94	0h 35m 38s ± 0h 11m 19s	36,510,380
Multi-Output 5 heads	61.80 ± 15.44	1h 13m 20s ± 0h 17m 57s	36,526,440
Multi-Output 7 heads	75.00 ± 10.99	1h 11m 33s ± 0h 10m 22s	36,544,100
Multi-Output 10 heads	90.00 ± 11.73	1h 39m 41s ± 0h 13m 20s	36,573,590
CIFAR-100			
Multi-Output 3 heads	20.20 ± 3.31	0h 22m 0s ± 0h 3m 20s	36,919,880
Multi-Output 5 heads	29.20 ± 7.19	0h 42m 8s ± 0h 10m 34s	37,368,480
Multi-Output 7 heads	46.40 ± 8.28	0h 55m 2s ± 0h 9m 56s	37,977,080
Multi-Output 10 heads	60.20 ± 9.57	1h 23m 45s ± 0h 11m 13s	39,189,980

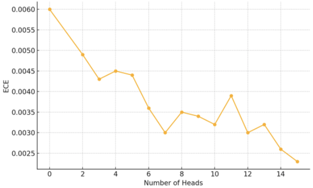
Dataset	Learning Rate (LR)	L2 Weight Decay	Batch Size	Optimizer
CIFAR	0.0001	0.0005	$16 \times$ num_heads	Adam
MNIST	0.0005	0	$16 \times$ num_heads	Adam



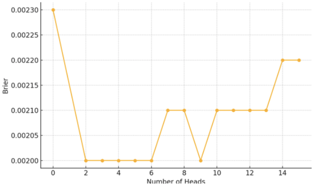
Trade-off between accuracy and calibration



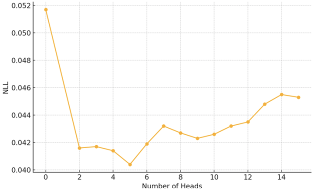
Accuracy with the Number of Heads



ECE with the Number of Heads



Brier Score with the Number of Heads



NLL with the Number of Heads

Conclusion

- **Post-hoc Uncertainty Estimation:** A meta-model technique introduced on top of pre-trained models.
- **Model-Agnostic and Efficient:** No need for additional data or retraining, while achieving near state-of-the-art results.
- **Strong Performance:** Demonstrated on MNIST, CIFAR-10, CIFAR-100, and corrupted datasets with minimal computational overhead.
- **Scalability:** Efficiently disentangles uncertainty using output disagreements, ensuring applicability in real-world settings.
- **Future Work:** Requires further testing on diverse datasets, especially for out-of-distribution (OOD) detection.
- **Numerical Uncertainty:** Addressing numerical errors in high-dimensional optimization and real-time systems is crucial.

References I

V. Bengs, E. Hüllermeier, and W. Waegeman. On second-order scoring rules for epistemic uncertainty quantification. *arXiv preprint arXiv:2301.12736*, 2023.

L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

S. Depeweg, J. M. Hernández-Lobato, F. Doshi-Velez, and S. Udluft. Decomposition of uncertainty in bayesian deep learning for efficient and risk-sensitive learning. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pages 1192–1201. PMLR, 2018.

T. G. Dietterich. Ensemble methods in machine learning. In J. Kittler and F. Roli, editors, *Multiple Classifier Systems*, pages 1–15. Springer, Berlin, Heidelberg, 2000.

Y. Gal and Z. Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, pages 1050–1059. PMLR, 2016.

References II

T. Gneiting and A. E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.

C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger. On calibration of modern neural networks. *arXiv preprint arXiv:1706.04599*, 2017.

D. Hendrycks and T. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *Proceedings of the 7th International Conference on Learning Representations*, 2019. doi: 10.48550/arXiv.1903.12261.

B. Lakshminarayanan, A. Pritzel, and C. Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 6402–6413, 2017.

G. Shafer and V. Vovk. A tutorial on conformal prediction. *Journal of Machine Learning Research*, 9(Mar):371–421, 2008.

References III

M. Shen, Y. Bu, P. Sattigeri, S. Ghosh, S. Das, and G. Wornell. Post-hoc uncertainty learning using a dirichlet meta-model. *arXiv preprint arXiv:2212.07359*, 2022. URL `https://doi.org/10.48550/arXiv.2212.07359`. Accepted by AAAI 2023.