Yandex Research

Uncertainty Estimation in Machine Learning

Andrey Malinin

11 March 2020

Overview of the Talk

1. Motivation: Why do we need Uncertainty Estimation?
2. Sources of Uncertainty in Predictions
3. Ensemble Approaches
4. Assessment of uncertainty quality
5. Ensemble Distribution Distillation

Overview of the Talk

1. **Motivation: Why do we need Uncertainty Estimation?**
2. Sources of Uncertainty in Predictions
3. Ensemble Approaches
4. Assessment of uncertainty quality
5. Ensemble Distribution Distillation

Why is Uncertainty important?

- Philosophical $\rightarrow$ "Scio me nihil scire" - Socrates
  - Intelligent agents must know that they don't know $\rightarrow$
  - Agents must understand the limits of their knowledge
- Intelligent behaviour depends on detecting novel situations
  - Animals display fear or **curiosity**
  - Humans ask questions
- Uncertainty must affect **actions** of an intelligent agent

Why is Uncertainty important?

- Machine Learning (ML) systems are being deployed to many applications $\rightarrow$
  - Image Classification / Segmentation
  - Speech Recognition
  - Machine Translation
  - Etc...
- In some applications, the cost of a mistake is **high** or consequence **fatal**
  - Medical Applications
  - Financial Applications
  - Self-driving vehicles
- Obtaining measures of uncertainty in predictions helps avoid mistakes!
  - Increases safety and reliability of ML system

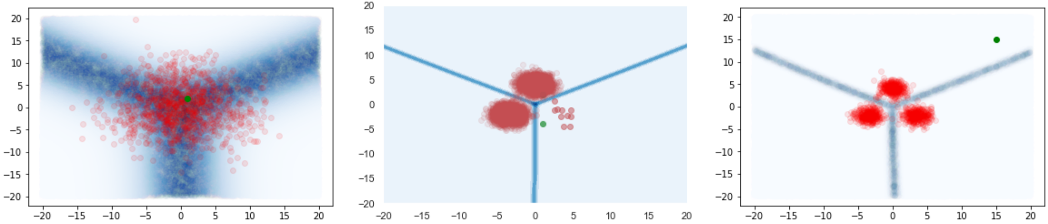
Scenario

- Given a deployed model and a test input $\boldsymbol{x}^*$ we wish to:
  - Obtain a prediction
  - Obtain a measure of **uncertainty in prediction**
- Take action based estimate of uncertainty
  - Reject prediction / stop decoding sentence
  - Modify policy / do exploration
  - Ask for human intervention
  - Use active learning

Overview of the Talk

1. Motivation: Why do we need Uncertainty Estimation?
2. **Sources of Uncertainty in Predictions**
3. Ensemble Approaches
4. Assessment of uncertainty quality
5. Ensemble Distribution Distillation

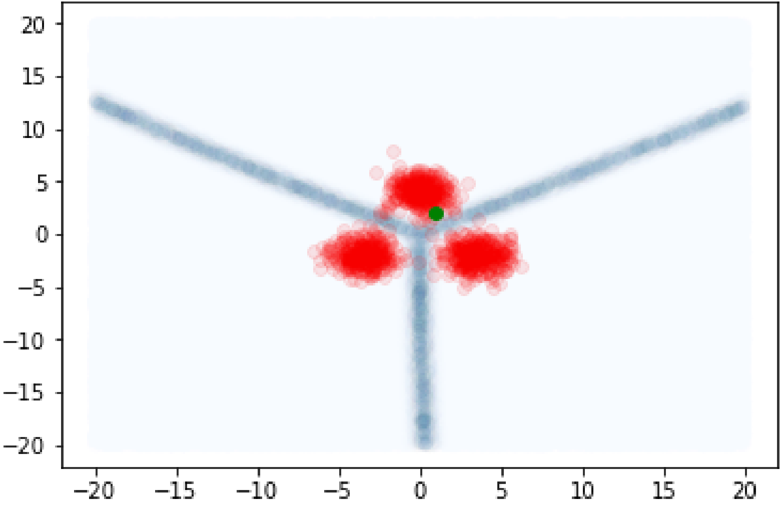
Sources of Uncertainty [Malinin, 2019]



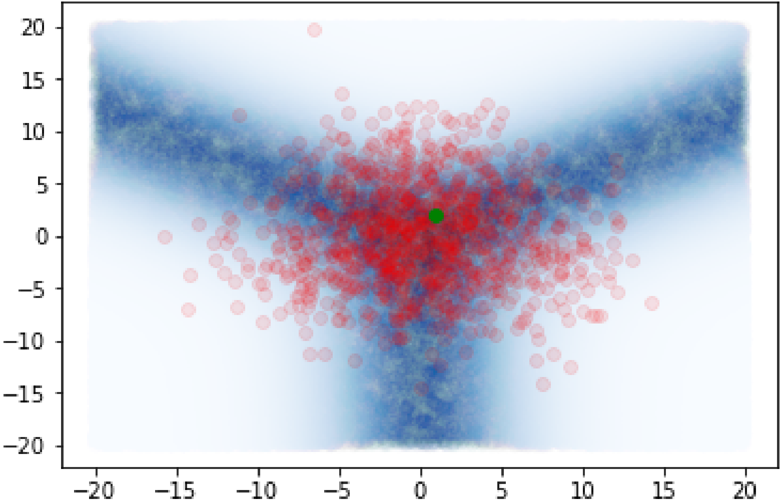
(a) Data Uncertainty (b) Knowledge Uncertainty (c) Knowledge Uncertainty

- Knowledge (epistemic) uncertainty refers to both:
  - Data Sparsity and Knowledge Uncertainty

Data (Aleatoric) Uncertainty

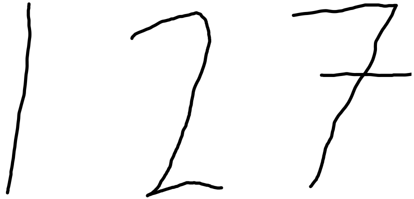


Data Uncertainty



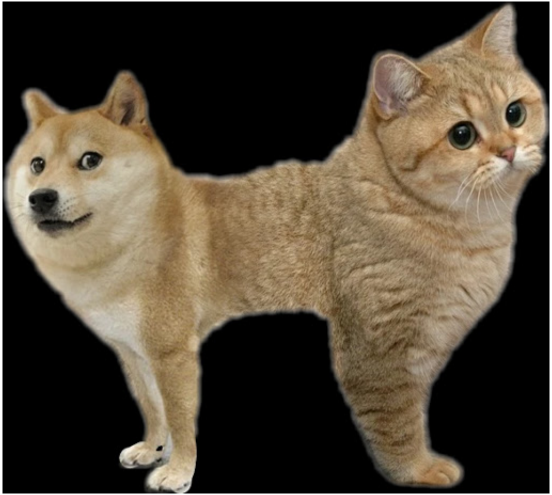
Data (Aleatoric) Uncertainty

- Distinct Classes



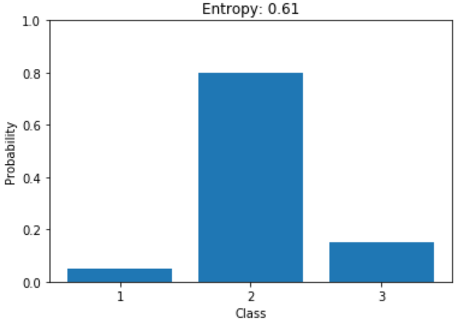
- Overlapping Classes

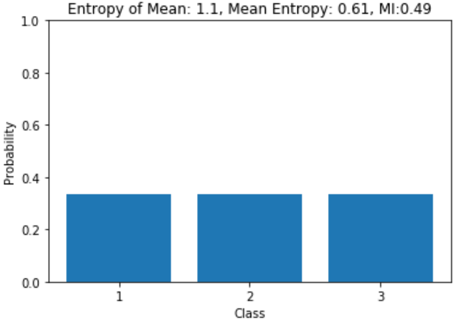
Data (Aleatoric) Uncertainty



Reminder - Entropy

$$\mathcal{H}[\mathrm{P}_{\mathtt{tr}}(y|\boldsymbol{x}^*)] = -\sum_{c=1}^{K} \mathrm{P}_{\mathtt{tr}}(y=\omega_c|\boldsymbol{x}^*) \ln \mathrm{P}_{\mathtt{tr}}(y=\omega_c|\boldsymbol{x}^*)$$





Data (Aleatoric) Uncertainty

- Data Uncertainty is the *entropy* of the *true data distribution* $\rightarrow$

$$\mathcal{H}[\mathrm{P}_{\mathtt{tr}}(y|\boldsymbol{x}^*)] = -\sum_{c=1}^{K} \mathrm{P}_{\mathtt{tr}}(y=\omega_c|\boldsymbol{x}^*) \ln \mathrm{P}_{\mathtt{tr}}(y=\omega_c|\boldsymbol{x}^*)$$

- Captured by the entropy of a model's posterior over classes $\rightarrow$

$$\mathcal{H}[\mathrm{P}(y|\boldsymbol{x}^*, \hat{\boldsymbol{\theta}})] = -\sum_{c=1}^{K} \mathrm{P}(y=\omega_c|\boldsymbol{x}^*, \hat{\boldsymbol{\theta}}) \ln \mathrm{P}(y=\omega_c|\boldsymbol{x}^*, \hat{\boldsymbol{\theta}})$$

- Data Uncertainty is captured as a consequence of Maximum Likelihood Estimation

Data Uncertainty

- Data Uncertainty is captured as a consequence of Maximum Likelihood Estimation

$$\begin{aligned}
\mathcal{L}(\boldsymbol{\theta},\mathcal{D}) &= \mathbb{E}_{\mathtt{p}_{\mathtt{tr}}(\boldsymbol{x},y)}\Big[-\sum_{c=1}^{K}\mathcal{I}(y=\omega_c)\ln \mathtt{P}(\hat{y}=\omega_c|\boldsymbol{x};\boldsymbol{\theta})\Big] \\
&= \mathbb{E}_{\mathtt{p}_{\mathtt{tr}}(\boldsymbol{x})}\Big[-\sum_{c=1}^{K}\mathtt{P}_{\mathtt{tr}}(y=\omega_c|\boldsymbol{x})\ln \mathtt{P}(\hat{y}=\omega_c|\boldsymbol{x};\boldsymbol{\theta})\Big] \\
&= \mathbb{E}_{\mathtt{p}_{\mathtt{tr}}(\boldsymbol{x})}\Big[\sum_{c=1}^{K}\mathtt{P}_{\mathtt{tr}}(y=\omega_c|\boldsymbol{x})\ln \frac{\mathtt{P}_{\mathtt{tr}}(y=\omega_c|\boldsymbol{x})}{\mathtt{P}(\hat{y}=\omega_c|\boldsymbol{x};\boldsymbol{\theta})} - \mathtt{P}_{\mathtt{tr}}(y=\omega_c|\boldsymbol{x})\ln \mathtt{P}_{\mathtt{tr}}(y=\omega_c|\boldsymbol{x})\Big] \\
&= \mathbb{E}_{\mathtt{p}_{\mathtt{tr}}(\boldsymbol{x})}\Big[\underbrace{\mathrm{KL}[\mathtt{P}_{\mathtt{tr}}(y|\boldsymbol{x})||\mathtt{P}(y|\boldsymbol{x};\boldsymbol{\theta})]}_{Reducible\ \ Loss} + \underbrace{\mathcal{H}[\mathtt{P}_{\mathtt{tr}}(y|\boldsymbol{x})]}_{Irreducible\ \ Loss}\Big]
\end{aligned}$$

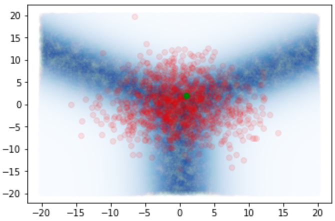
Data Uncertainty

- Data Uncertainty is captured as a consequence of Maximum Likelihood Estimation

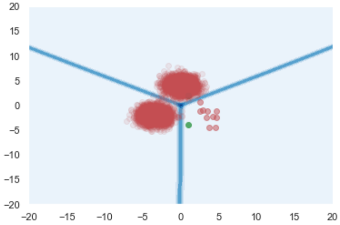
$$\mathcal{L}(\boldsymbol{\theta}, \mathcal{D}) = \mathbb{E}_{\mathrm{p_{tr}}(\boldsymbol{x})}\Big[\underbrace{\mathrm{KL}[\mathrm{P_{tr}}(y|\boldsymbol{x})||\mathrm{P}(y|\boldsymbol{x};\boldsymbol{\theta})]}_{\textit{Reducible Loss}} + \underbrace{\mathcal{H}[\mathrm{P_{tr}}(y|\boldsymbol{x})]}_{\textit{Irreducible Loss}}\Big]$$

- When loss $\mathcal{L}(\boldsymbol{\theta}, \mathcal{D})$ is minimized $\rightarrow$ Data Uncertainty is fully captured.
- The result is conditioned on
  - Sufficient training data $\rightarrow$ no over-fitting
  - Model $\mathrm{P}(y|\boldsymbol{x};\boldsymbol{\theta})$ powerful enough to fully capture $\mathrm{P_{tr}}(y|\boldsymbol{x})$

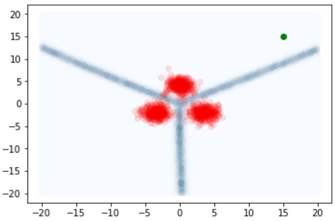
Sources of Uncertainty



(a) Data Uncertainty



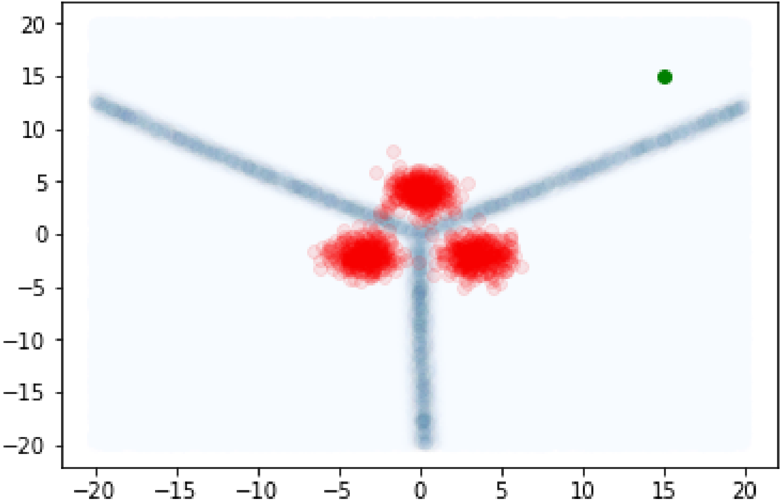
(b) Data Sparsity



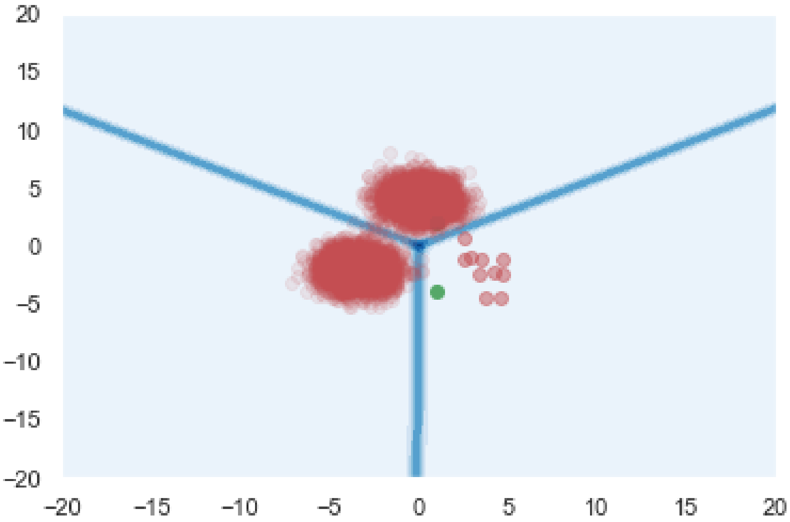
(c) Out-of-Distribution inputs

- Knowledge (epistemic) uncertainty refers to both:
  - Data Sparsity and Out-of-distribution inputs

Knowledge (Epistemic) Uncertainty

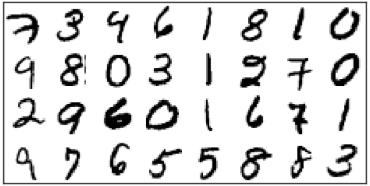
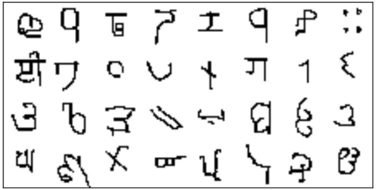


Knowledge (Epistemic) Uncertainty

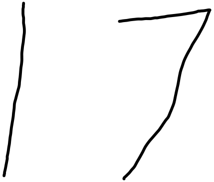
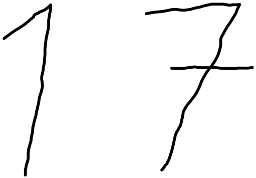


Knowledge (Epistemic) Uncertainty

- Unseen classes

- Unseen variations of seen classes

Knowledge (Epistemic) Uncertainty

Sources of Uncertainty

- Data Uncertainty $\rightarrow$ **Known-Unknown**
  - Class overlap (complexity of decision boundaries)
  - Homoscedastic and Heteroscedastic noise
- Knowledge Uncertainty $\rightarrow$ **Unknown-Unknown**
  - Test input in out-of-distribution region far from training data
  - Test input in out-of-distribution region of sparse training data
- Appropriate **action** depends on **source** of uncertainty
  - Separating sources of uncertainty requires **Ensemble approaches**

Overview

1. Motivation: Why do we need Uncertainty Estimation?
2. Sources of Uncertainty in Predictions
3. **Ensemble Approaches**
4. Ensemble Distribution Distillation?
5. Assessment of uncertainty quality

Ensemble Approaches

- Uncertainty in $\boldsymbol{\theta}$ captured by model posterior $\mathtt{p}(\boldsymbol{\theta}|\mathcal{D}) \rightarrow$

$$\mathtt{p}(\boldsymbol{\theta}|\mathcal{D}) = \frac{\mathtt{p}(\mathcal{D}|\boldsymbol{\theta})\mathtt{p}(\boldsymbol{\theta})}{\mathtt{p}(\mathcal{D})}$$

- Can consider an ensemble of models $\rightarrow$

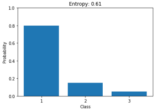
$$\{\mathtt{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta}^{(m)})\}_{m=1}^{M}, \ \boldsymbol{\theta}^{(m)} \sim \mathtt{p}(\boldsymbol{\theta}|\mathcal{D})$$

- Bayesian inference of $\mathtt{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta}) \rightarrow$

$$\mathtt{P}(y|\boldsymbol{x}^*, \mathcal{D}) = \mathbb{E}_{\mathtt{p}(\boldsymbol{\theta}|\mathcal{D})}\big[\mathtt{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta})\big] \approx \frac{1}{M}\sum_{m=1}^{M} \mathtt{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta}^{(m)}), \ \boldsymbol{\theta}^{(m)} \sim \mathtt{p}(\boldsymbol{\theta}|\mathcal{D})$$

Ensemble for certain in-domain input

$$\boldsymbol{\theta}^{(1)} \longrightarrow$$

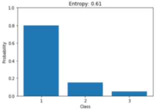


$\rightarrow$

$\mathrm{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta}^{(1)})$

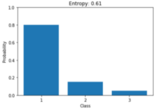
$\vdots \qquad \vdots$

$$\mathrm{p}(\boldsymbol{\theta}) + \mathcal{D} \rightarrow \mathrm{p}(\boldsymbol{\theta}|\mathcal{D}) \longrightarrow \quad \boldsymbol{\theta}^{(i)} \longrightarrow$$



$\rightarrow$

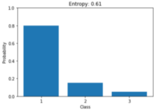
$\mathrm{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta}^{(i)})$



$\mathrm{P}(y|\boldsymbol{x}^*, \mathcal{D})$

$\vdots \qquad \vdots$

$$\boldsymbol{\theta}^{(N)} \longrightarrow$$

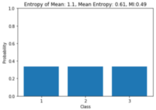


$\rightarrow$

$\mathrm{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta}^{(N)})$

Ensemble for uncertain in-domain input

$$\boldsymbol{\theta}^{(1)} \longrightarrow$$

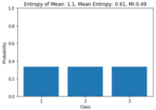


$\rightarrow$

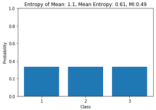
$\mathrm{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta}^{(1)})$

$\vdots \qquad \vdots$

$$\mathrm{p}(\boldsymbol{\theta}) + \mathcal{D} \rightarrow \mathrm{p}(\boldsymbol{\theta}|\mathcal{D}) \longrightarrow \quad \boldsymbol{\theta}^{(i)} \longrightarrow$$



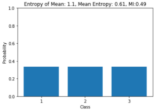
$\rightarrow$



$\mathrm{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta}^{(i)})$ $\qquad$ $\mathrm{P}(y|\boldsymbol{x}^*, \mathcal{D})$

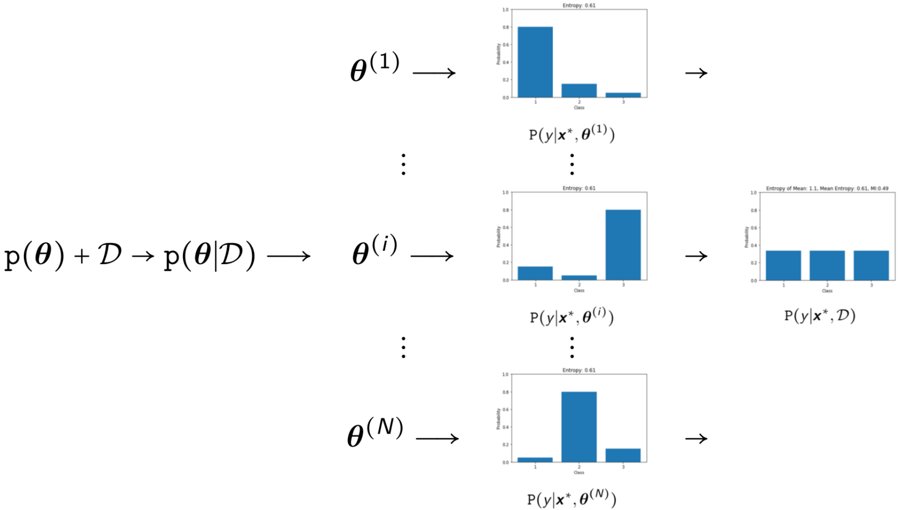
$\vdots \qquad \vdots$

$$\boldsymbol{\theta}^{(N)} \longrightarrow$$



$\rightarrow$

$\mathrm{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta}^{(N)})$



$\mathrm{p}(\boldsymbol{\theta}) + \mathcal{D} \rightarrow \mathrm{p}(\boldsymbol{\theta}|\mathcal{D}) \longrightarrow$
$\boldsymbol{\theta}^{(1)} \longrightarrow$
$\boldsymbol{\theta}^{(i)} \longrightarrow$
$\boldsymbol{\theta}^{(N)} \longrightarrow$
Entropy: 0.61
Probability
Class
$\mathrm{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta}^{(1)})$
$\mathrm{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta}^{(i)})$
$\mathrm{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta}^{(N)})$
Entropy of Mean: 1.1, Mean Entropy: 0.61, MI:0.49
$\mathrm{P}(y|\boldsymbol{x}^*, \mathcal{D})$

Total Uncertainty

- Consider the entropy of the predictive posterior $\mathtt{P}(y|\boldsymbol{x}^*, \mathcal{D}) \rightarrow$

$$\mathcal{H}\big[\mathtt{P}(y|\boldsymbol{x}^*, \mathcal{D})\big] = \mathcal{H}\big[\mathbb{E}_{p(\boldsymbol{\theta}|\mathcal{D})}[\mathtt{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta})]\big]$$
$$\approx \mathcal{H}\Big[\frac{1}{M}\sum_{m=1}^{M}\mathtt{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta}^{(m)})\Big],\ \boldsymbol{\theta}^{(m)} \sim \mathtt{p}(\boldsymbol{\theta}|\mathcal{D})$$

- Measure of Total Uncertainty
    - Combination of Data uncertainty and **Knowledge uncertainty**

Expected Data Uncertainty

- Lets consider an ensemble of models $\{\mathrm{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta}^{(m)})\}_{m=1}^{M}, \ \boldsymbol{\theta}^{(m)} \sim \mathrm{p}(\boldsymbol{\theta}|\mathcal{D})$
  - Each model $\mathrm{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta}^{(m)})$ captures an different estimate of data uncertainty.
- Ensemble estimate of data uncertainty $\rightarrow$ Expected Data Uncertainty

$$\mathbb{E}_{p(\boldsymbol{\theta}|\mathcal{D})}\big[\mathcal{H}[\mathrm{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta})]\big] \approx \frac{1}{M}\sum_{m=1}^{M}\mathcal{H}\big[\mathrm{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta}^{(m)})\big], \ \boldsymbol{\theta}^{(m)} \sim \mathrm{p}(\boldsymbol{\theta}|\mathcal{D})$$

- Not the same as entropy of the predictive posterior $\mathrm{P}(y|\boldsymbol{x}^*, \mathcal{D})$

Knowledge Uncertainty

- If the predictions from the models are consistent

$$\underbrace{\mathcal{H}\big[\mathbb{E}_{\mathrm{p}(\boldsymbol{\theta}|\mathcal{D})}[\mathrm{P}(y|\boldsymbol{x}^*,\boldsymbol{\theta})]\big]}_{\text{Total Uncertainty}} - \underbrace{\mathbb{E}_{\mathrm{p}(\boldsymbol{\theta}|\mathcal{D})}\big[\mathcal{H}[\mathrm{P}(y|\boldsymbol{x}^*,\boldsymbol{\theta})]\big]}_{\text{Expected Data Uncertainty}} = 0$$

- If the predictions from the models are diverse

$$\underbrace{\mathcal{H}\big[\mathbb{E}_{\mathrm{p}(\boldsymbol{\theta}|\mathcal{D})}[\mathrm{P}(y|\boldsymbol{x}^*,\boldsymbol{\theta})]\big]}_{\text{Total Uncertainty}} - \underbrace{\mathbb{E}_{\mathrm{p}(\boldsymbol{\theta}|\mathcal{D})}\big[\mathcal{H}[\mathrm{P}(y|\boldsymbol{x}^*,\boldsymbol{\theta})]\big]}_{\text{Expected Data Uncertainty}} > 0$$

- Difference of the two is a measure of knowledge uncertainty

$$\underbrace{\mathcal{I}[y,\boldsymbol{\theta}|\boldsymbol{x}^*,\mathcal{D}]}_{\text{Knowledge Uncertainty}} = \underbrace{\mathcal{H}\big[\mathbb{E}_{\mathrm{p}(\boldsymbol{\theta}|\mathcal{D})}[\mathrm{P}(y|\boldsymbol{x}^*,\boldsymbol{\theta})]\big]}_{\text{Total Uncertainty}} - \underbrace{\mathbb{E}_{\mathrm{p}(\boldsymbol{\theta}|\mathcal{D})}\big[\mathcal{H}[\mathrm{P}(y|\boldsymbol{x}^*,\boldsymbol{\theta})]\big]}_{\text{Expected Data Uncertainty}}$$

Approximate Inference

- Variational Inference:
  - Bayes by Backprop [Blundell et al., 2015]
  - Probabalistic Backpropagation [Hernández-Lobato and Adams, 2015]
- Monte-Carlo Methods:
  - Monte-Carlo Dropout [Gal, 2016, Gal and Ghahramani, 2016]
  - Stochastic Gradient Langevin Dynamics [Welling and Teh, 2011]
  - Fast-Ensembling via Mode Connectivity [Garipov et al., 2018]
  - Stochastic Weight Averaging Gaussian (SWAG) [Maddox et al., 2019]
- Non-Bayesian Ensembles:
  - Bootstrap DQN [Osband et al., 2016]
  - Deep Ensembles [Lakshminarayanan et al., 2017]

Limitations

- Hard to guarantee diverse $\{\mathrm{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta}^{(m)})\}_{m=1}^{M}$ for OOD $\boldsymbol{x}^*$
- Diversity of ensemble depends on:
  - Selection of prior
  - Nature of approximations
  - Architecture of network
  - Properties and size of data
- Computationally expensive

Assessment of Uncertainty Quality

- Quality of uncertainty estimates of commonly assessed via
  - Log-likelihood of test data $\mathcal{D}_{tst} = \{\boldsymbol{x}^*_{(i)}, y^*_{(i)}\}$
  - Calibration (Expected Calibration Error)
- Test Log-likelihood $\rightarrow$

$$NLL = \frac{1}{N}\sum_{i=1}^{N} -\ln \mathrm{P}(y^*_{(i))}|\boldsymbol{x}^*_{(i)}, \mathcal{D})$$

- Calibration $\rightarrow$ Does confidence correspond to long-run accuracy?
- Informative quality statistics, but weakly related to application

Assessment of Uncertainty Quality

- Uncertainty should be assessed in the context of an **application**
- Threshold-based outlier detection $\rightarrow$
  - Misclassification Detection [Hendrycks and Gimpel, 2016]
  - Out-of-distribution input Detection
  - Adversarial Attack Detection [Malinin and Gales, 2019]
- Active Learning [Gal, 2016]
- Reinforcement Learning uncertainty-driven exploration [Osband et al., 2016]
- Other...

Assessment of Uncertainty Quality

- Uncertainty should be assessed in the context of an **application**
- Threshold-based outlier detection $\rightarrow$
  - **Misclassification Detection** [Hendrycks and Gimpel, 2016, Malinin, 2019]
  - **Out-of-distribution input Detection** [Malinin, 2019]
  - Adversarial Attack Detection [Malinin and Gales, 2019]
- Active Learning [Gal, 2016]
- Reinforcement Learning uncertainty-driven exploration [Osband et al., 2016]
- Other...

Misclassification and Out-of-Distribution detection

- Threshold-based detection $\rightarrow$

$$\mathcal{I}_T(\boldsymbol{x}) = \begin{cases} 1, & \mathcal{H}(\boldsymbol{x}) > T \\ 0, & \mathcal{H}(\boldsymbol{x}) \leq T \end{cases}$$

- If $\mathcal{I}_T(\boldsymbol{x}) = 1 \rightarrow$ outlier
- If $\mathcal{I}_T(\boldsymbol{x}) = 0 \rightarrow$ normal
- Evaluate performance using
  - Area under Precision-Recall Curve (AUPR) $\rightarrow$ Misclassification Detection
  - Area under ROC curve (AUROC) $\rightarrow$ Out-of-distribution Detection

ROC Curve

- ROC curve depicts true-positive vs. false-positive trade-off at various thresholds T

$$t_p(T) = \frac{1}{N_p}\sum_{i=1}^{N_p}\mathcal{I}_T(\boldsymbol{x}_p^{(i)}) \qquad f_p(T) = \frac{1}{N_n}\sum_{j=1}^{N_n}\mathcal{I}_T(\boldsymbol{x}_n^{(j)})$$

- Area under ROC Curve
  - Good for balanced datasets
  - Good performance $\rightarrow$ 100 %
  - Random performance $\rightarrow$ 50 %

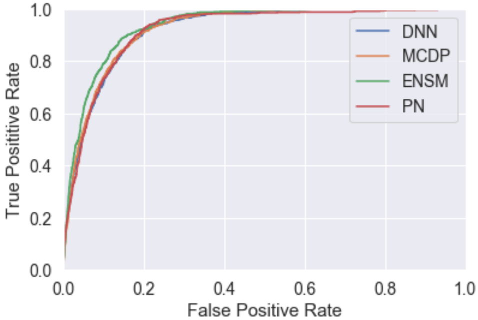
Precision-Recall Curve

- Curve depicts precision-recall trade-off at various thresholds T

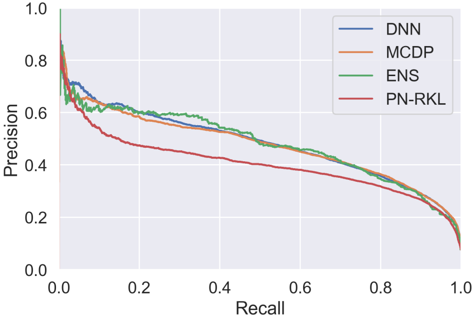
$$P(T) = \frac{\sum_{i=1}^{N_p} \mathcal{I}_T(\boldsymbol{x}_p^{(i)})}{\sum_{i=1}^{N_p} \mathcal{I}_T(\boldsymbol{x}_p^{(i)}) + \sum_{j=1}^{N_n} \mathcal{I}_T(\boldsymbol{x}_n^{(j)})} \qquad R(T) = \frac{1}{N_p} \sum_{i=1}^{N_p} \mathcal{I}_T(\boldsymbol{x}_p^{(i)})$$

- Area under Precision Recall Curve
    - Good for mis-balanced datasets
    - Good performance $\rightarrow$ 100%
    - Random performance $\rightarrow$ Classifier % Error

ROC and PR Curves

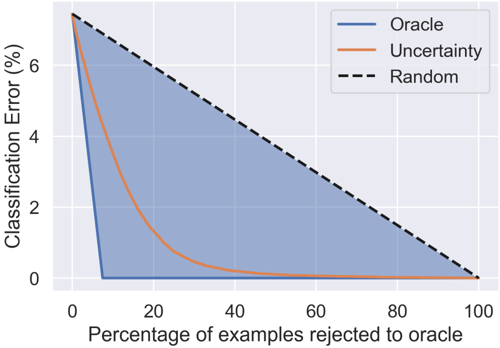


(a) ROC Curve

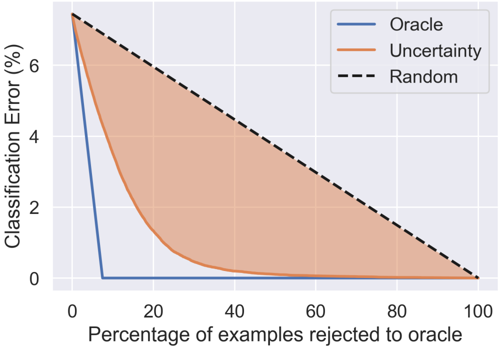


(b) PR Curve

Prediction Rejection Curve



(a) Shaded area is $AR_{\texttt{orc}}$.



(b) Shaded area is $AR_{\texttt{uns}}$.

Prediction Rejection Ratio

- Prediction Rejection Ratio summarizes Rejection Curve:

$$PRR = \frac{AR_{\texttt{uns}}}{AR_{\texttt{orc}}}$$

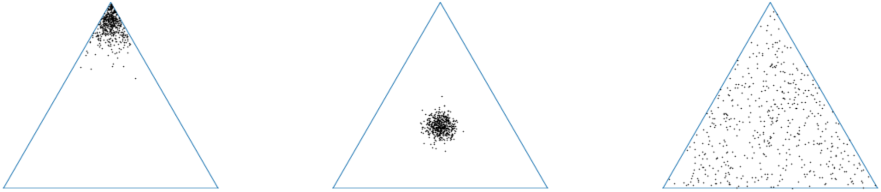
- Assesses misclassification detection independent of classification performance

Overview

1. Motivation: Why do we need Uncertainty Estimation?
2. Sources of Uncertainty in Predictions
3. Ensemble Approaches
4. Assessment of uncertainty quality
5. **Ensemble Distribution Distillation** [Malinin et al., 2019]

Distributions on a Simplex

- Ensemble $\{\mathrm{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta}^{(m)})\}_{m=1}^{M}$ can be visualized on a simplex



(a) Confident (b) Data Uncertainty (c) Knowledge Uncertainty

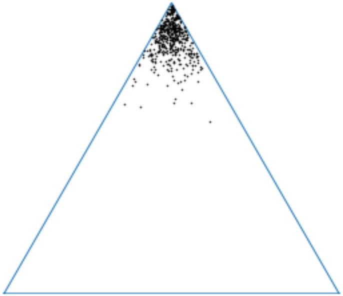
- Same as sampling from **implicit** Distribution over output Distributions

$$\mathrm{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta}^{(m)}) \sim \mathrm{p}(\boldsymbol{\theta}|\mathcal{D}) \equiv \boldsymbol{\mu}^{(m)} \sim \mathrm{p}(\boldsymbol{\mu}|\boldsymbol{x}^*, \mathcal{D})$$

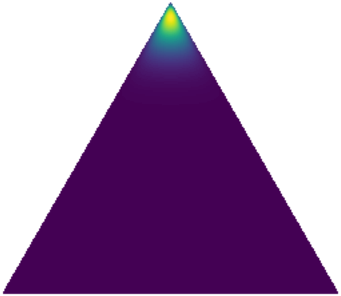
Distributions on a Simplex (cont)

- Expanding out $\boldsymbol{\mu}^{(m)} = \begin{bmatrix} \mathrm{P}(y = \omega_1) \\ \mathrm{P}(y = \omega_2) \\ \vdots \\ \mathrm{P}(y = \omega_K) \end{bmatrix}$, where each $\boldsymbol{\mu}^{(m)}$ is a point on a simplex.

Distribution over Distributions

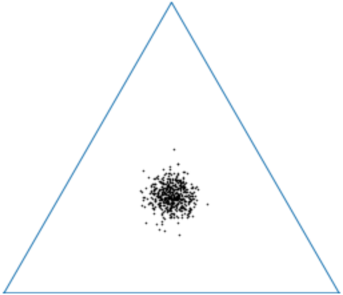


(a) $\{\boldsymbol{\mu}^{(m)}\}_{m=1}^{M}$

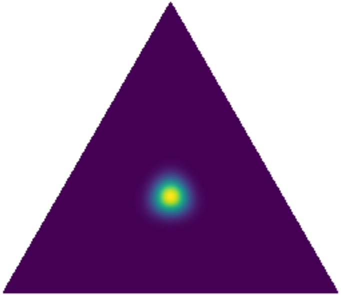


(b) $\mathrm{p}(\boldsymbol{\mu}|\boldsymbol{x}^*, \mathcal{D})$

Distribution over Distributions

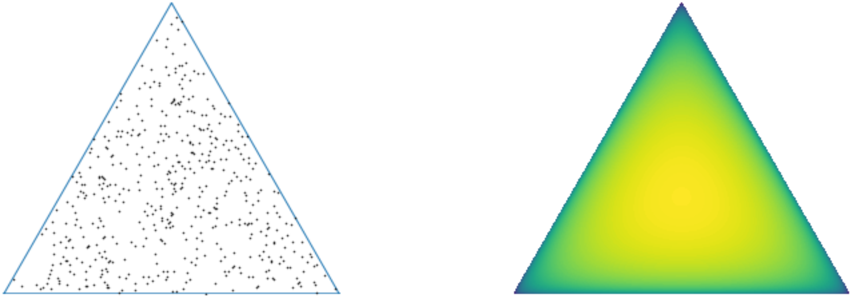


(a) $\{\boldsymbol{\mu}^{(m)}\}_{m=1}^{M}$



(b) $\mathrm{p}(\boldsymbol{\mu}|\boldsymbol{x}^{*}, \mathcal{D})$

Distribution over Distributions



(a) $\{\boldsymbol{\mu}^{(m)}\}_{m=1}^{M}$ (b) $\mathrm{p}(\boldsymbol{\mu}|\boldsymbol{x}^*, \mathcal{D})$

Prior Networks [Malinin and Gales, 2018]

- **Explicitly** model $\mathrm{p}(\boldsymbol{\mu}|\boldsymbol{x}^*, \mathcal{D})$ using a Prior Network $\mathrm{p}(\boldsymbol{\mu}|\boldsymbol{x}^*; \hat{\boldsymbol{\theta}})$

$$\mathrm{p}(\boldsymbol{\mu}|\boldsymbol{x}^*; \hat{\boldsymbol{\theta}}) \approx \mathrm{p}(\boldsymbol{\mu}|\boldsymbol{x}^*, \mathcal{D})$$

- Predictive posterior distribution is given by expected categorical

$$\mathrm{P}(y|\boldsymbol{x}^*; \hat{\boldsymbol{\theta}}) = \mathbb{E}_{\mathrm{p}(\boldsymbol{\mu}|\boldsymbol{x}^*; \hat{\boldsymbol{\theta}})}\big[\mathrm{p}(y|\boldsymbol{\mu})\big] = \hat{\boldsymbol{\mu}}$$

Uncertainty Measures for Prior Networks

- Ensemble uncertainty decomposition:

$$\underbrace{\mathcal{I}[y, \boldsymbol{\theta} | \boldsymbol{x}^*, \mathcal{D}]}_{\text{Knowledge Uncertainty}} = \underbrace{\mathcal{H}[\mathbb{E}_{\mathrm{p}(\boldsymbol{\theta}|\mathcal{D})}[\mathrm{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta})]]}_{\text{Total Uncertainty}} - \underbrace{\mathbb{E}_{\mathrm{p}(\boldsymbol{\theta}|\mathcal{D})}[\mathcal{H}[\mathrm{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta})]]}_{\text{Expected Data Uncertainty}}$$

- Prior Network uncertainty decomposition

$$\underbrace{\mathcal{I}[y, \boldsymbol{\mu} | \boldsymbol{x}^*; \hat{\boldsymbol{\theta}}]}_{\text{Knowledge Uncertainty}} = \underbrace{\mathcal{H}\big[\mathbb{E}_{\mathrm{p}(\boldsymbol{\mu}|\boldsymbol{x}^*;\hat{\boldsymbol{\theta}})}[\mathrm{P}(y|\boldsymbol{\mu})]\big]}_{\text{Total Uncertainty}} - \underbrace{\mathbb{E}_{\mathrm{p}(\boldsymbol{\mu}|\boldsymbol{x}^*;\hat{\boldsymbol{\theta}})}\big[\mathcal{H}[\mathrm{P}(y|\boldsymbol{\mu})]\big]}_{\text{Expected Data Uncertainty}}$$

Ensemble Distillation (EnD) [Hinton et al., 2015, Korattikara et al., 2015]

- Ensembles are computationally expensive
  - Distill an ensemble into a **single** model

$$\{\mathrm{P}(y|\boldsymbol{x}, \boldsymbol{\theta}^{(m)})\}_{m=1}^{M} \rightarrow \mathrm{P}(y|\boldsymbol{x}, \hat{\boldsymbol{\theta}})$$

- Minimize KL-divergence to mean of ensemble:

$$\mathcal{L}(\hat{\boldsymbol{\theta}}, \mathcal{D}) = \mathbb{E}_{\mathrm{p}(\boldsymbol{x})}\Big[\mathrm{KL}\big[\mathbb{E}_{\hat{\mathrm{p}}(\boldsymbol{\theta}|\mathcal{D})}[\mathrm{P}(y|\boldsymbol{x}, \boldsymbol{\theta})]||\mathrm{P}(y|\boldsymbol{x}, \hat{\boldsymbol{\theta}})\big]\Big]$$

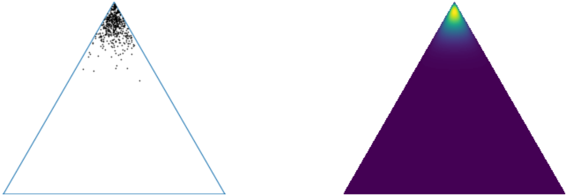
- Computational Performance gain
- Robustness to Adversarial Attack (Defensive Distillation)

Ensemble Distillation (EnD)

- EnD $\rightarrow$ model captures only *mean* of ensemble
- Diversity of ensemble is lost $\rightarrow$
  - Cannot separate measures of uncertainty
- Solution $\rightarrow$ Ensemble Distribution Distillation

Ensemble Distribution Distillation (End^2) [Malinin et al., 2019]

- Distill an ensemble into a **single** Prior Network



$$\left\{\mathrm{P}(y|\boldsymbol{x}, \boldsymbol{\theta}^{(m)})\right\}_{m=1}^{M} \quad \rightarrow \quad \mathrm{p}(\boldsymbol{\mu}|\boldsymbol{x}; \hat{\boldsymbol{\theta}})$$

- Goal $\rightarrow$ Maximum information extraction from ensemble.

Ensemble Distribution Distillation (End^2)

- Parameterize a Dirichlet distribution using Neural Network:

$$\mathrm{p}(\boldsymbol{\mu}|\boldsymbol{x};\boldsymbol{\theta}) = \mathtt{Dir}(\boldsymbol{\mu};\boldsymbol{\alpha}), \quad \boldsymbol{\alpha} = \boldsymbol{f}(\boldsymbol{x};\boldsymbol{\theta}), \quad \alpha_c > 0$$

- Training data are ensemble predictions for every input:

$$\mathcal{D} = \left\{\left\{p(y|\boldsymbol{x}^{(i)};\boldsymbol{\theta}^{(j)}), \boldsymbol{x}^{(i)}\right\}_{j=1}^{N}\right\}_{i=1}^{M} \sim \hat{\mathrm{p}}(\boldsymbol{\mu},\boldsymbol{x})$$

- Train via Maximum Likelihood:

$$\mathcal{L}(\boldsymbol{\theta},\mathcal{D}) = -\mathbb{E}_{\hat{\mathrm{p}}(\boldsymbol{x})}\left[\mathbb{E}_{\hat{\mathrm{p}}(\boldsymbol{\mu}|\boldsymbol{x})}[\ln \mathrm{p}(\boldsymbol{\mu}|\boldsymbol{x};\boldsymbol{\theta})]\right]$$

Ensemble Distribution Distillation: Image Classification

Dataset	Individual	Ensemble	EnD	EnD^2
CIFAR-10	8.0	**6.2**	6.7	6.9
CIFAR-100	30.4	**26.3**	28.2	28.0
TinyImageNet	41.8	**36.6**	38.5	37.3

Table: Classification Performance (% Error).

Dataset	Individual	Ensemble	EnD	EnD^2
CIFAR-10	84.6	**86.8**	85.1	85.7
CIFAR-100	72.5	**75.0**	74.0	74.0
TinyImageNet	70.8	**73.8**	72.6	72.7

Table: Misclassification detection performance (% PRR).

Ensemble Distribution Distillation: OOD Detection

Test OOD Dataset	Model	CIFAR-10		CIFAR-100	
		Total Unc.	Knowledge Unc.	Total Unc.	Knowledge Unc.
LSUN	Individual	91.3	-	75.6	-
	EnD	89.0	-	76.5	-
	EnD^2	94.4	**95.3**	83.5	86.9
	Ensemble	94.5	94.4	82.4	**88.4**
TIM	Individual	88.9	-	70.5	-
	EnD	86.9	-	70.0	-
	EnD^2	91.3	**91.8**	76.4	79.3
	Ensemble	**91.8**	91.4	76.6	**81.7**

Table: OOD detection performance (% AUC-ROC) for CIFAR-10 and CIFAR-100 models.

Take away points

- Uncertainty is important $\rightarrow$
  - Philosophically and practically necessary
- Sources of Uncertainty $\rightarrow$
  - Data Uncertainty and Knowledge Uncertainty
- Uncertainty Estimation via Ensembles $\rightarrow$
  - Theoretically motivated separation of uncertainty sources
  - Computationally Expensive $\rightarrow$ use Ensemble Distribution Distillation
- Uncertainty quality can be assessed via
  - Test-set Negative Log-Likelihood
  - PRR for Misclassification Detection
  - ROCAUC for OOD detection
  - ... and other applications...
- New area - lots of research opportunities!

Thank You!

Any questions?

[Blundell et al., 2015] Blundell, C., Cornebise, J., Kavukcuoglu, K., and Wierstra, D. (2015).
Weight uncertainty in neural networks.
arXiv preprint arXiv:1505.05424.

[Gal, 2016] Gal, Y. (2016).
Uncertainty in Deep Learning.
PhD thesis, University of Cambridge.

[Gal and Ghahramani, 2016] Gal, Y. and Ghahramani, Z. (2016).
Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning.
In *Proc. 33rd International Conference on Machine Learning (ICML-16)*.

References II

[Garipov et al., 2018] Garipov, T., Izmailov, P., Podoprikhin, D., Vetrov, D. P., and Wilson, A. G. (2018).
Loss surfaces, mode connectivity, and fast ensembling of dnns.
In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 31*, pages 8789–8798. Curran Associates, Inc.

[Hendrycks and Gimpel, 2016] Hendrycks, D. and Gimpel, K. (2016).
A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks.
http://arxiv.org/abs/1610.02136.
arXiv:1610.02136.

[Hernández-Lobato and Adams, 2015] Hernández-Lobato, J. M. and Adams, R. (2015).
Probabilistic backpropagation for scalable learning of bayesian neural networks.
In *International Conference on Machine Learning*, pages 1861–1869.

[Hinton et al., 2015] Hinton, G., Vinyals, O., and Dean, J. (2015).
Distilling the knowledge in a neural network.
In *NIPS Deep Learning and Representation Learning Workshop*.

References IV

[Korattikara et al., 2015] Korattikara, A., Rathod, V., Murphy, K. P., and Welling, M. (2015).
Bayesian dark knowledge.
In Cortes, C., Lawrence, N. D., Lee, D. D., Sugiyama, M., and Garnett, R., editors, *Advances in Neural Information Processing Systems 28*, pages 3438–3446. Curran Associates, Inc.

[Lakshminarayanan et al., 2017] Lakshminarayanan, B., Pritzel, A., and Blundell, C. (2017).
Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles.
In *Proc. Conference on Neural Information Processing Systems (NIPS)*.

References V

[Maddox et al., 2019] Maddox, W., Garipov, T., Izmailov, P., Vetrov, D. P., and Wilson, A. G. (2019).
A simple baseline for bayesian uncertainty in deep learning.
CoRR, abs/1902.02476.

[Malinin, 2019] Malinin, A. (2019).
Uncertainty Estimation in Deep Learning with application to Spoken Language Assessment.
PhD thesis, University of Cambridge.

[Malinin and Gales, 2018] Malinin, A. and Gales, M. (2018).
Predictive uncertainty estimation via prior networks.
In *Advances in Neural Information Processing Systems*, pages 7047–7058.

References VI

[Malinin and Gales, 2019] Malinin, A. and Gales, M. (2019).
Reverse kl-divergence training of prior networks: Improved uncertainty and adversarial robustness.
arXiv preprint arXiv:1905.13472.

[Malinin et al., 2019] Malinin, A., Mlodozeniec, B., and Gales, M. (2019).
Ensemble distribution distillation.
arXiv preprint arXiv:1905.00076.

[Osband et al., 2016] Osband, I., Blundell, C., Pritzel, A., and Van Roy, B. (2016).
Deep exploration via bootstrapped dqn.
In Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems 29*, pages 4026–4034. Curran Associates, Inc.

References VII

[Welling and Teh, 2011] Welling, M. and Teh, Y. W. (2011).
Bayesian Learning via Stochastic Gradient Langevin Dynamics.
In *Proc. International Conference on Machine Learning (ICML)*.