# Discriminant Distance-Aware Representation on Deterministic Uncertainty Quantification Methods

**Jiaxin Zhang**
Intuit AI Research
jiaxin_zhang@intuit.com

**Kamalika Das**
Intuit AI Research
kamalika_das@intuit.com

**Sricharan Kumar**
Intuit AI Research
sricharan_kumar@intuit.com

## Abstract

Uncertainty estimation is a crucial aspect of deploying dependable deep learning models in safety-critical systems. In this study, we introduce a novel and efficient method for deterministic uncertainty estimation called Discriminant Distance-Awareness Representation (DDAR). Our approach involves constructing a DNN model that incorporates a set of prototypes in its latent representations, enabling us to analyze valuable feature information from the input data. By leveraging a distinction maximization layer over optimal trainable prototypes, DDAR can learn a discriminant distance-awareness representation. We demonstrate that DDAR overcomes feature collapse by relaxing the Lipschitz constraint that hinders the practicality of deterministic uncertainty methods (DUMs) architectures. Our experiments show that DDAR is a flexible and architecture-agnostic method that can be easily integrated as a pluggable layer with distance-sensitive metrics, outperforming state-of-the-art uncertainty estimation methods on multiple benchmark problems.

## 1 Introduction

Deep neural network (DNN) models play an important role in many safety-critical tasks, e.g., autonomous driving, or medical diagnosis. A key characteristic shared by these tasks is their risk sensitivity so that a confidently wrong prediction can lead to fatal accidents and misleading decisions. Therefore, it is of utmost importance to develop reliable and efficient uncertainty estimation methods that allow for the safe deployment in large-scale, real-world applications across computer vision and natural language processing (Zhang, 2021; Tran et al., 2022; Zhang et al., 2024).

However, naive DNN models do not deliver certainty estimates or suffer from over or under-confidence, i.e. are badly calibrated or assigned with high confidence to out-of-domain (OOD) inputs. This has led to the development of probabilistic approaches for uncertainty estimation in DNN models. Bayesian Neural Networks (BNNs) (Osawa et al., 2019; Wenzel et al., 2020) represent the dominant solution for quantifying uncertainty but exactly modeling the full posterior is often computationally intractable, and not scale well to complex tasks (Mukhoti et al., 2021a). Monte Carlo (MC) Dropout (Gal and Ghahramani, 2016), is simple to implement but its uncertainty is not always reliable while requiring multiple forward passes. Deep Ensembles (Lakshminarayanan et al., 2017) involves training multiple deep models from different initializations and a different data set ordering, which outperforms BNN but comes at the high expense of computational cost (Ovadia et al., 2019). A shared characteristic of these approaches is their high computational cost and large memory requirement. Thus, efficient and scalable methods for uncertainty estimation largely remain an open problem (Gawlikowski et al., 2023).

Recently, a set of promising works, named Deterministic Uncertainty Methods (DUMs) (Mukhoti et al., 2021a) emerged for estimating uncertainty with a single forward pass while treating its weights deterministically (Postels et al., 2021). These methods are prone to be efficient and scalable solutions to uncertainty estimation and out-of-distribution (OOD) detection problems. DUMs aim at learning informative latent representation of a model given that the distribution of latent representation should be representative of the input distribution. Then DUMs estimate uncertainty by replacing the final softmax layer with a distance-sensitive function. Specifically, DUQ (Van Amersfoort et al., 2020) defines the uncertainty as the distance between

Proceedings of the 27$^{\text{th}}$ International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238. Copyright 2024 by the author(s).

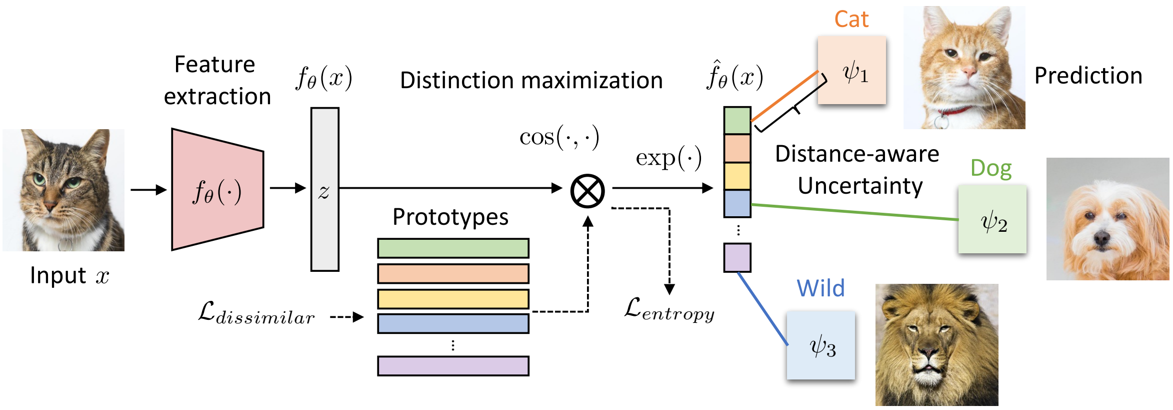

Figure 1: DDAR overview: an efficient, distance-aware and architecture-agnostic method for deterministic uncertainty estimation. DDAR learns a discriminative distance-aware latent representation by leveraging the learnable prototypes and distinction maximization layer. Combined with the RBF kernel, our method performs accurate uncertainty estimation and competitive OOD detection capabilities.

the model output and the closest centroid and proposes a novel centroid updating step based on Radial Basis Function (RBF) networks. DUE (van Amersfoort et al., 2021) built upon DUQ introduces deep kernel learning by using the inducing point approximation by incorporating a large number of inducing points without overfitting. SNGP (Liu et al., 2020) replaces the softmax layer with Gaussian processes (GP) with RBF kernel to extend distance awareness to the output layer. However, naive latent representations typically suffer from the *feature collapse* (Van Amersfoort et al., 2020) issue when OOD data are mapped to similar feature representations as in-distribution data, which makes OOD detection based on high-level representations impossible. To address the feature collapse issue, DUMs strongly rely on the regularization of latent representation with the ability to differentiate between in-distribution and out-of-distribution data. Otherwise, these methods have several essential challenges and weaknesses (Postels et al., 2021).

Specifically, DUMs mitigate the feature collapse issue through regularization techniques to mimic distances between latent representations to distances in the original input space. This is often achieved by adding constraints over the bi-Lipschitz constant, which enforces a lower and upper bound to expansion and contraction performed by a DNN model (Postels et al., 2021). The upper bound enforces *smoothness*, i.e., small changes in the input do not lead to large changes in the latent space and the lower bound enforces *sensitiveness*, i.e., different inputs are mapped to distinct latent spaces. Primarily, there are two methods to impose the bi-Lipschitz constraint: (1) *Gradient Penalty* (Van Amersfoort et al., 2020) directly constrains the gradient of input but leads to large computational cost due to backpropagation through the input's gradients; (2) *Spectral Normalization* (Miyato et al., 2018) normalizes the weights of each residual layers using their spectral norm, which is computationally more efficient compared with gradient penalty but requires the use of residual layers so it is not architecture-agnostic (van Amersfoort et al., 2021; Liu et al., 2020). Moreover, both regularization in distance-awareness representations have limitations in explicitly preserving sample-specific information. In other words, they may discard useful information in the latent representation depending on the underlying distance metric (Postels et al., 2020; Wu and Goodman, 2020; Franchi et al., 2022). Although the distance-awareness representation with the above regularization shows promising results, it does not explicitly preserve sample-specific information. In other words, it may discard useful information in its latent representation depending on the underlying distance metric. To fill the gap, this work aims to answer the following questions:

- *Can we build a simple and efficient uncertainty estimation method without feature collapse issue?*
- *Is that possible to learn a distance-aware representation while preserving sample-specific information?*
- *Is the feature extractor architecture-agnostic, with higher flexibility, not limited by residual layers?*

To answer these core questions, we develop DDAR (Discriminant Distance-Awareness Representation) - a novel method for deterministic uncertainty estimation, which is *efficient, distance-aware, and architecture-agnostic*. As shown in Fig. 1, we first build a DNN model imbued with a set of prototypes over its latent presentations. These prototypes allow us to better analyze useful feature information from the input data. Then we learn a discriminant distance-awareness representation (DDAR) by leveraging a distinction maximization layer over optimal trainable prototypes. DDAR is simpler, more efficient, and easy to use as a pluggable layer integrating with distance-sensitive metrics. We further propose adding two constrained losses to improve the informative and discriminative properties of the latent representation. We demonstrate that DDAR addresses the feature collapse by relaxing the Lipschitz constraint hindering the practicality of DUM architectures. Through several experiments on toy examples, image classification, and text OOD detection, DDAR shows superior performance over the state-of-the-art uncertainty estimation baseline methods, specifically single-forward pass methods. Compared with the ensemble-based methods, DDAR is also competitive but more computationally efficient.

## 2 Background

**Prototype Learning.** Prototype learning (Wen et al., 2016; Li et al., 2021; Gao et al., 2021) has been applied to feature extraction to build more discriminative features by compacting intra-class features and dispersing the inter-class ones. Specifically, few-shot learning studies are based on the prototypical networks (Snell et al., 2017) for their simplicity and competitive performance. Given a small support set of $N$ labeled examples $\mathcal{S} = \{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_N, y_N)\}$ where $\mathbf{x}_i \in \mathbb{R}^D$ is the feature vector nd $y_i$ is the label. Prototypical networks compute a *prototype*, $\mathbf{c}_k$ of each class through an embedding function $f_\theta : \mathbb{R}^D \to \mathbb{R}^M$. Each prototype is the mean vector of the embedded support points belonging to its class $k$:

$$\mathbf{c}_k = \frac{1}{|\mathcal{S}_k|} \sum_{(\mathbf{x}_i, y_i) \in \mathcal{S}_k} f_\theta(\mathbf{x}_i), i = 1, ..., K. \quad (1)$$

Thus embedded new query points are classified via a softmax over distance to class prototypes:

$$p_\theta(y = k|\mathbf{x}) \propto \exp(-d(f_\theta(\mathbf{x}), \mathbf{c}_k)) \quad (2)$$

where $d$ is a distance function: $\mathbb{R}^M \times \mathbb{R}^M \to [0, +\infty]$. The training is performed by minimizing the negative log-probability $\mathcal{J}(\theta) = -\log p_\theta(y = k|\mathbf{x})$ of the true class $k$ via stochastic gradient descent (SGD). The prototype networks can be easily extended to tackle zero-shot learning where each class comes with meta-data giving a high-level description of the class rather than a small number of labeled examples. The prototype for each class can be obtained by learning an embedding of the meta-data into a shared space.

**Deterministic Uncertainty Estimation.** DUQ (Van Amersfoort et al., 2020) builds on the RBF function which requires the preservation of input distances in the output space which is achieved using the gradient penalty. Compared with approximated GPs used in DUE (van Amersfoort et al., 2021) and SNGP (Liu et al., 2020) which rely on Laplace approximation with random Fourier feature and inducing point approximation, we prefer the simpler and more efficient RBF function as the distance metric for estimating uncertainty, which is defined as the distance between the model output and the class centroids:

$$\mathcal{K}_c(f_\theta, \psi_c) = \exp\left[-\frac{\|\mathbf{W}_c f_\theta(\mathbf{x}) - \psi_c\|_2^2}{n \cdot 2\sigma^2}\right], \quad (3)$$

where $f_\theta$ is the feature extractor parametrized by $\theta$, $\psi_c$ is the centroid for class $c$, $\mathbf{W}_c$ is a weight matrix with a length scale parameter $\sigma$, $n$ is the centroid size, and $\Psi = \{\psi_1, ...\psi_c\}$ is the class centroids. The loss function $\mathcal{L}_{RBF}$ is defined by the sum of binary cross entropy between a one-hot binary encoding of the label $y_c$ and each class kernel value $\mathcal{K}_c$:

$$\mathcal{L}_{RBF} = -\sum_c y_c \log(\mathcal{K}_c) + (1 - y_c) \log(1 - \mathcal{K}_c). \quad (4)$$

The training is performed by stochastic gradient descent on $\theta$ and $\mathcal{W} = \{\mathbf{W}_1, ..., \mathbf{W}_c\}$. However, the loss in Eq. (4) is prone to feature collapse without further regularization of DNN. Gradient penalty (Van Amersfoort et al., 2020) can address this issue but leads to large computational overhead as it requires differentiation of the gradients of the input with respect to the DNN parameters.

## 3 Methodology

In this section, we aim to address the three core questions by proposing a new DUM approach, based on a discriminative distance-aware representation that improves both scalability and flexibility. This is achieved by following the principle of DUMs of learning a sensitive and smooth representation but not by enforcing directly the Lipschitz constraint.

Specifically, we start with a theoretical analysis of feature collapse and understand the essential property of the Lipschitz function. We then propose learning optimal prototypes to better capture the distance-aware property and to improve the discriminative property with the help of the distinction maximization layer. Finally, we carefully design the discriminant loss with regularization to constrain the hidden representations to mimic distances from the input space. Our DDAR method is lighter, faster and only needs a single forward pass, while it can be used as a pluggable learning layer on any top of the feature extractor, which is architecture-agnostic.

### 3.1 Feature Collapse Issue

Most DUMs address the feature collapse issue through regularization methods for constraining the hidden representations to mimic distances from the input space. This is typically achieved by enforcing constraints over the Lipschitz constant of the DNN (Van Amersfoort et al., 2020; Liu et al., 2020). Specifically, given the feature extractor $f_\theta(\mathbf{x})$, the bi-Lipschitz condition implies that for any pair of inputs $\mathbf{x}_1$ and $\mathbf{x}_2$:

$$L_1\|\mathbf{x}_1 - \mathbf{x}_2\| \leq \|f_\theta(\mathbf{x}_1) - f_\theta(\mathbf{x}_2)\| \leq L_2\|\mathbf{x}_1 - \mathbf{x}_2\|. \quad (5)$$

where $L_1$ and $L_2$ are positive and bounded Lipschitz constants $0 < L_1 < 1 < L_2$. The lower Lipschitz bound $L_1$ deals with the sensitivity to preserve distances in the latent space thus avoiding feature collapse. The upper Lipschitz bound $L_2$ enforces the smoothness and robustness of a DNN by preventing over-sensitivity to perturbations in the input space of $\mathbf{x}$.

Typically, DUM approaches aim for bi-Lipschitz DNNs with small Lipschitz constants but this is sub-optimal based on the concentration theory (Boucheron et al., 2013).

**Theorem 1**. *Assume $\mathbf{x}$ is a set of random vectors, drawn from a Gaussian distribution $\mathcal{N}(0, \sigma^2\mathbf{I}_d)$ and let $f : \mathbb{R}^d \rightarrow \mathbb{R}$ be a Lipschitz function with Lipschitz constant $\tau$, then we have*

$$p(|f(\mathbf{x}) - \mathbb{E}(f(\mathbf{x}))| > s) \leq 2\exp(-\frac{s^2}{2\tau^2\sigma^2}) \quad (6)$$

*for all $s > 0$. That means the smaller the Lipschitz constant $\tau$ is, the more the concentration of the samples around their mean increases, which results in increased feature collapse.*

This motivates us to develop a new DUM strategy that does not rely on the network to comply with the Lipschitz constraint. In the meantime, the desired Lipschitz function will separate the dissimilar samples far from each other while retaining the similar samples as closely as possible.

### 3.2 Distance-aware Latent Representation

The DUM foundation is built upon the RBF function (or other kernel functions) which requires distance preserving (Van Amersfoort et al., 2020). Instead of focusing on preserving distance in the input space, we aim to deal with the distances in the latent space, named *distance-aware latent representation.* This is achieved by imposing a set of prototypes over the latent representations. Inspired by the prototype learning (Snell et al., 2017), we leverage these prototypes to better analyze features from the new queries (samples) in light of the knowledge acquired by the DNN from the training data. Unlike the prototypical networks that use fixed prototypes (mean vector of the embedded support points), we propose to learn the optimal prototypes (as learnable parameters) for improving distance awareness in the latent space.

Beyond the distance-aware property, the discriminative property is also critical to latent representation. The center loss (Wen et al., 2016) used in prototype learning can help DNN to build more discriminative features by compacting intra-class features and dispersing the inter-class ones. In this work, we propose to use a distinction maximization (DM) layer (Macêdo et al., 2022) as a hidden layer over latent representation. This way enables us to preserve the discriminative properties of the latent representations compared to placing DM as the last layer. Compared with the softmax layer, the DM layer shows competitive performance in classification accuracy, uncertainty estimation, and OOD detection, while maintaining deterministic neural network inference efficiency (Macêdo et al., 2021, 2022).

Let's define $\mathbf{z} \in \mathbb{R}^n$ to be the latent representation of $\mathbf{x}$ given feature extractor $f_\theta$, i.e., $\mathbf{z} = f_\theta(\mathbf{x})$, which is the input to the DM layer. Given a set of prototypes, $\mathcal{P} = \{\mathbf{p}_1, ..., \mathbf{p}_m\}$ of $m$ vectors that are trainable, we define a distinction maximization (DM) layer using cosine distance

$$\mathcal{D}_p(\mathbf{z}, \mathbf{p}_i) = \left[\frac{< \mathbf{z}, \mathbf{p}_1 >}{\|\mathbf{z}\|_2\|\mathbf{p}_1\|_2}, \cdots, \frac{< \mathbf{z}, \mathbf{p}_m >}{\|\mathbf{z}\|_2\|\mathbf{p}_m\|_2}\right]^\top. \quad (7)$$

The vectors $\mathbf{p}_i$ in Eq. (7) can help in better placing an input sample in the learned latent space using these prototypes as references. Also, it is flexible to assign an arbitrarily large number of prototypes such that a richer latent mapping is defined by a finer convergence of the latent space.

Then we apply the distinction maximization to the feature representation and subsequently use the exponential function as the activation function. This way aims to sharpen similarity values and thus facilitates the alignment of the data embedding to the prototypes. Thus the update latent representation $\tilde{f}(\theta)$ tends to be more distinctive

$$\tilde{f}_\theta(\mathbf{x}) = \exp(-\mathcal{D}_p(f_\theta(\mathbf{x}))) = \exp(-\mathcal{D}_p(\mathbf{z})). \quad (8)$$

Note that the vector weights $\mathbf{p}_m$ are optimized jointly with $\theta$ and $\mathbf{W}_c$ in Eq. (3) and $\mathbf{p}_m$ can also work as indicators for better analyzing informative patterns in the discriminant latent representation such that DDAR is distance preserving that satisfies the bi-Lipschitz function property.

**Proposition 1**. *Consider a hidden mapping $f \colon \mathcal{X} \rightarrow \mathcal{F}$, the discriminant latent representation $\tilde{f}_\theta(\mathbf{x})$ is a*

*Lipschitz function, which satisfies*

$$\|\exp(-\mathcal{D}_p(\mathbf{z}_1)) - \exp(-\mathcal{D}_p(\mathbf{z}_2))\| \leq \tau\|\mathbf{z}_1 - \mathbf{z}_2\|. \quad (9)$$

*Then $\tilde{f}$ is distance preserving with $\tau \in \mathbb{R}^+$.*

### 3.3 Discriminant Loss with Regularization Constraints

To better address feature collapse, we improve the discrimination of latent space by adding constraints to the binary cross entropy loss in Eq. (4). To fully leverage the benefits of learnable prototypes, we propose two measures to (1) avoid the collapse of all prototypes into a single prototype. (2) not rely only on a single prototype. Thus we first add a constraint to force the prototypes to be dissimilar using the negative sum of cosine distance:

$$\mathcal{L}_{dissimilar} = -\sum_{i<j} \frac{<\mathbf{p}_i, \mathbf{p}_j>}{\|\mathbf{p}_i\|_2\|\mathbf{p}_j\|_2}, \quad (10)$$

which avoids the collapse of all prototypes to a single prototype. Next, we constrain the latent representation $\mathcal{D}_p(f_\theta(\mathbf{x}))$ to not rely only on a single prototype by adding an entropy regularization loss:

$$\mathcal{L}_{entropy} = \sum_{k=1}^{n} \sigma(\mathcal{D}_p(\mathbf{z}))_k \cdot \log(\sigma(\mathcal{D}_p(\mathbf{z}))_k), \quad (11)$$

where $\sigma$ is the softmax layer, and the subscript index $k$ means the $k$-th coefficient of a tensor. Adding these two constraints in Eq. (10) and (11) to the RBF loss in Eq. (4), we can achieve more discriminative features by increasing the distance between prototypes and enlarging the dispersion of different prototype features. Therefore the total loss for training is defined by

$$\mathcal{L}_{total} = \mathcal{L}_{RBF} + \lambda(\mathcal{L}_{dissimilar} + \mathcal{L}_{entropy}), \quad (12)$$

where $\lambda \in [0, 1]$ is the coefficient weight of the constraints. To avoid additional hyperparameters, we only use one coefficient to evaluate the effect of regularization constraints. Naively we can introduce another parameter to adjust the weight between $\mathcal{L}_{dissimilar}$ and $\mathcal{L}_{entropy}$. Note that we use the distinctive latent representation $\tilde{f}_\theta(\mathbf{x})$ in Eq. (8) to compute the RBF loss $\mathcal{L}_{RBF}$ rather than the original latent representation $f_\theta(\mathbf{x})$ in Eq. (3). We name this proposed method as *Discriminant Distance-Aware Representation* (DDAR), where the training procedure is shown in Algorithm 1.

## 4 Experiments

We show the performance of DDAR in two dimensions, with the two-moon dataset, and show the effect of discriminant distance-aware property on addressing feature collapse issues. We further test the OOD detection performance on CIFAR-10/100 vs SVHN datasets compared with multiple SOTA baselines. To verify the DDAR capability on data modalities beyond images, we also evaluate the DDAR method on practical language understanding tasks using the CLINC benchmark dataset (Larson et al., 2019). We run all baseline methods in similar settings using publicly available codes and hyperparameters for related methods. Some of the results are reported by the literature such that we can directly compare them.

**Algorithm 1** The DDAR algorithm

```
**Requirements**:
  - Feature extractor $f_\theta : x \rightarrow \mathbb{R}_d$ with feature space dimensionality $d$ and deep neural network parameters $\theta$
  - Hyperparameters: number of prototypes $m$, loss weight $\lambda$, length scale $\sigma$ in RBF, learning rate $\eta$, batch size $b$
  - Training and testing datasets: in-distribution data $\mathbf{x}_{\text{in}}$ (e.g., CIFAR-10/100) and OOD data $\mathbf{x}_{\text{ood}}$ (e.g., SVHN)
**Initialize:** Prototype parameters $\{\mathbf{p}_1, ..., \mathbf{p}_m\}$ of $m$ vectors, $\mathbf{p}_i \in \mathbb{R}$ and RBF kernel weight matrix $\mathbf{W}_c$ (size $n \times d$)
**for** train step = 1 **to** max step **do**
    Extract the feature embedding $f_\theta(\mathbf{x})$
    Compute the discriminant embedding $\tilde{f}_\theta(\mathbf{x}) = \exp(-\mathcal{D}_p(f_\theta(\mathbf{x})))$ after the DM layer
    Calculate the RBF loss $\mathcal{L}_{RBF}$ using Eq. (3)
    Calculate the dissimilar loss $\mathcal{L}_{dissimilar}$ in Eq. (10)
    Calculate the entropy loss $\mathcal{L}_{entropy}$ using Eq. (11)
    Combine all loss terms for total loss $\mathcal{L}_{total}$ in Eq. (12), $\mathcal{L}_{total} = \mathcal{L}_{RBF} + \lambda(\mathcal{L}_{dissimilar} + \mathcal{L}_{entropy})$,
    Update DNN parameters $\theta$, RBF kernel weight matrix $\mathbf{W}_c$ and prototype parameters $\mathbf{p}$ via stochastic gradient descent:
      $(\theta, \mathbf{W}_c, \mathbf{p}) \leftarrow (\theta, \mathbf{W}_c, \mathbf{p}) + \eta * \nabla_{\theta,\mathbf{W}_c,\mathbf{p}}\mathcal{L}_{total}$
    Update centroids $\Psi = \{\psi_1, ..., \psi_C\}$ using an exponential moving average of the feature vectors belonging to that class
**end for**
```

### 4.1 Toy Example: Two Moons

To illustrate the DDAR method, we first consider the two moons benchmark where the dataset consists of two moon-shaped distributions separable by a nonlinear decision boundary. We use the scikit-learn implementation to draw 500 samples from each in-domain class (blue and orange dots). We use a deep feature extractor, ResFFN-12-128, which consists of 12 residual layers with 128 hidden neurons used by (Liu et al., 2020). The embedding size is 128, the dropout rate is 0.01. We use 64 prototypes for this case and train this task using Adam optimizer with a learning rate of 0.01, batch size of 64 and set the length scale $\sigma$ of 0.3 in Eq. (3), $\lambda$ of 0.1 in Eq. (12).

Fig. 2 shows the results of decision boundary and predictive uncertainty. DDAR performs the expected be-

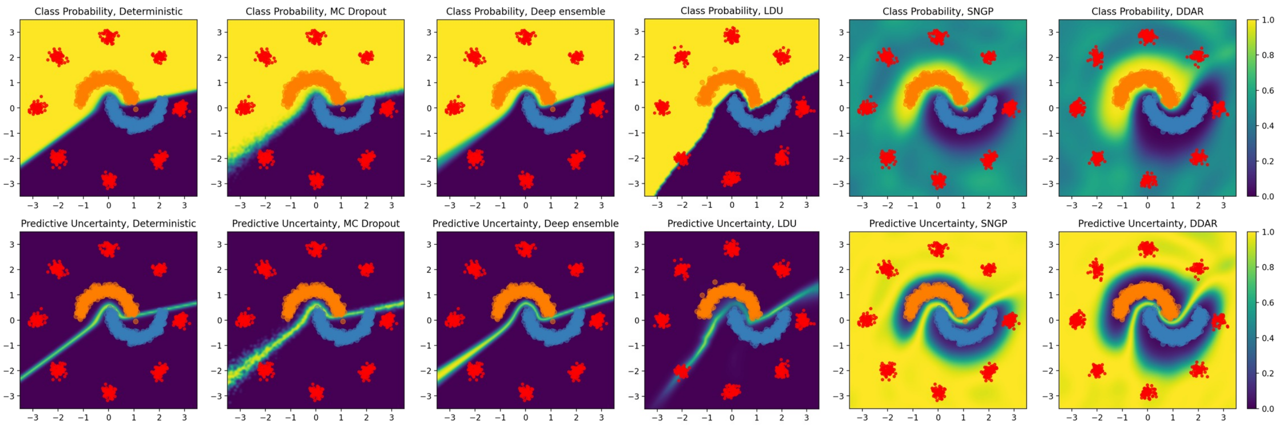

Figure 2: Uncertainty results of different baseline methods on the two moon 2D classification benchmarks. Yellow indicates uncertainty, while dark blue indicates certainty. The first row is the decision boundary (class probability) and the second row is the predictive uncertainty.

havior for high-quality predictive uncertainty: correctly classifiers the samples with high confidence (low uncertainty) in the region supported by the training data (dark blue color), and shows less confidence (high uncertainty) when samples are far away from the training data (yellow color), i.e., distance-awareness property. SNGP (Liu et al., 2020) is also able to maintain distance-awareness property via spectral normalization and shows a similar uncertainty surface. However, the other baseline methods, e.g., Deterministic (softmax) NN, MC Dropout (Gal and Ghahramani, 2016), and Deep Ensemble (Lakshminarayanan et al., 2017), quantify their predictive uncertainty based on the distance from the decision boundaries so only assign uncertainty along the decision boundary but certain elsewhere, which is not distance aware. They are overconfident since they assign high certainty to OOD samples even if they are far from the data. LDU (Macêdo et al., 2022) shows a similar overconfident result without leveraging OOD samples into training.

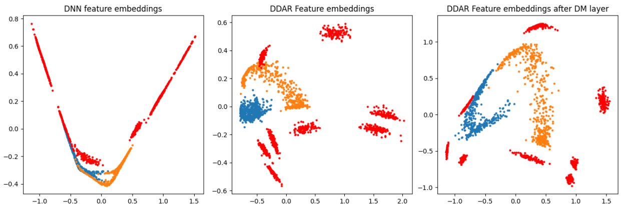

Figure 3: Addressing feature collapse. 2D project of feature embedding of regular DNN (left), DDAR embedding (middle), and DDAR embedding after the DM layer, trained on the two moons dataset.

**Feature Collapse**. Fig. 3 shows the PCA projection of the feature embedding of two moons dataset for feature collapse illustration. The objective for the regular DNN model introduces a large amount of distortion of the space, collapsing the two-class samples and OOD samples to a single line, making it almost impossible to use distance-awareness metric on these features. Specifically, the OOD samples move from a separated area in the input space on top of class data in the feature space, which fails in OOD detection tasks and results in unreliable predictive uncertainty estimation. In contrast, the feature embedding learned by the DDAR method allows a better disentangling of the latent space without overlapping the two classes. Furthermore, the learned embedding after the DM layer accurately maintains the relative distances of the two classes and OOD data.

### 4.2 OOD Detection: CIFAR-10/100 vs SVHN

**Baseline Methods**. We compare DDAR with two baselines for uncertainty estimation - MC Dropout (Gal and Ghahramani, 2016) (with 10 dropout samples) and Deep Ensemble (Lakshminarayanan et al., 2017) (with 10 models). We also include the softmax entropy of a regular DNN as a simple baseline. We choose three deterministic uncertainty methods (DUMs) - DUQ (Van Amersfoort et al., 2020), SNGP (Liu et al., 2020), and LDU (Macêdo et al., 2022) as representatives of distance-awareness (discriminative approaches) for uncertainty estimation. Also, we evaluate DDU (Mukhoti et al., 2021a) and MIR (Postels et al.) as representatives of informative representation (generative approaches (Winkens et al., 2020; Charpentier et al., 2020)), which fit a class-conditional GMM to their regularized latent representations and use the log-likelihood as an uncertainty proxy (Postels et al., 2021). For DDAR, we consider two cases with 256 (DDAR-256) and 64 (DDAR-64) prototypes respectively.

Table 1: OOD detection results: CIFAR-10 vs SVHN OOD.

| Method | Accuracy (↑) | ECE (↓) | AUROC (↑) |
|---|---|---|---|
| DNN (Softmax) | 94.3 ± 0.11 | 0.024 ± 0.003 | 0.928 ± 0.005 |
| MC Dropout (Gal and Ghahramani, 2016) | 95.7 ± 0.13 | 0.013 ± 0.002 | 0.934 ± 0.004 |
| Deep Ensemble (Lakshminarayanan et al., 2017) | **96.4 ± 0.09** | **0.011 ± 0.001** | **0.947 ± 0.002** |
| DUQ (Van Amersfoort et al., 2020) | 95.8 ± 0.12 | 0.027 ± 0.001 | 0.939 ± 0.007 |
| SNGP (Liu et al., 2020) | 95.7 ± 0.14 | 0.017 ± 0.003 | 0.940 ± 0.004 |
| LDU (Macêdo et al., 2022) | 95.5 ± 0.17 | 0.021 ± 0.002 | 0.945 ± 0.004 |
| DDU (Mukhoti et al., 2021a) | 95.1 ± 0.12 | **0.014 ± 0.002** | 0.936 ± 0.003 |
| MIR (Postels et al.) | 94.9 ± 0.15 | 0.021 ± 0.003 | 0.912 ± 0.005 |
| DDAR with 256 prototypes | **96.0 ± 0.12** | 0.015 ± 0.002 | **0.949 ± 0.003** |
| DDAR with 64 prototypes | 95.8 ± 0.13 | 0.016 ± 0.002 | **0.947 ± 0.003** |

Table 2: OOD detection results: CIFAR-100 vs SVHN OOD.

| Method | Accuracy (↑) | ECE (↓) | AUROC (↑) |
|---|---|---|---|
| DNN (Softmax) | 80.4 ± 0.11 | 0.082 ± 0.002 | 0.763 ± 0.011 |
| MC Dropout (Gal and Ghahramani, 2016) | 80.2 ± 0.22 | **0.031 ± 0.002** | 0.800 ± 0.014 |
| Deep Ensemble (Lakshminarayanan et al., 2017) | **82.5 ± 0.19** | 0.041 ± 0.002 | **0.832 ± 0.007** |
| DUQ (Van Amersfoort et al., 2020) | 79.7 ± 0.20 | 0.112 ± 0.002 | 0.777 ± 0.026 |
| SNGP (Liu et al., 2020) | **82.5 ± 0.16** | **0.030 ± 0.004** | 0.821 ± 0.019 |
| LDU (Macêdo et al., 2022) | 81.3 ± 0.15 | 0.052 ± 0.003 | 0.822 ± 0.003 |
| DDU (Mukhoti et al., 2021a) | 81.6 ± 0.14 | 0.029 ± 0.003 | 0.826 ± 0.009 |
| MIR (Postels et al.) | 80.9 ± 0.18 | 0.037 ± 0.002 | 0.788 ± 0.011 |
| DDAR with 256 prototypes | **82.5 ± 0.17** | 0.032 ± 0.002 | **0.829 ± 0.008** |
| DDAR with 64 prototypes | 82.0 ± 0.17 | 0.035 ± 0.002 | 0.826 ± 0.009 |

**Datasets**. We train the DDAR model on CIFAR-10 and CIFAR-100 image classification tasks. Following the benchmarking setup suggested in (Ovadia et al., 2019), we evaluate the model's predictive accuracy and expected calibration error (ECE) (Guo et al., 2017) under clean CIFAR-10/100 testing data. To evaluate the model's OOD performance, we choose the standard OOD task (Van Amersfoort et al., 2020) using SVHN as the OOD data for a model trained on CIFAR-10/100. We normalize the OOD datasets using the in-distribution training data (CIFAR-10/100) and use the Area Under the Reciever Operator Curve (AUROC) metric to report the OOD detection (image classification) performance.

**Model and Optimization**. Each baseline method shares the same ResFFN-12-128 architecture as used in two moons for training on CIFAR-10/100 datasets. Each method has a hyperparameter for the regularization of its latent representation, and we choose the hyperparameter so that it minimizes the validation loss. To compare the variation of each method, all results are averaged over 5 independent runs on an NVIDIA V100 GPU. We use 64 and 256 prototypes for each case and train the tasks using Adam optimizer with a learning rate of 0.01, and batch size of 128.

**Results**. To evaluate OOD detection performance, we use the standard setting based on training on CIFAR-10 and CIFAR-100 as in-distribution data and SVHN as OOD data. The performance of different baselines for CIFAR-10 and CIFAR-100 are shown in Table 1 and 2 (The top 2 results are highlighted in bold). Note that our DDAR method shows superior results, specifically on the AUROC and Accuracy metrics. With the increasing of prototypes, DDAR shows improved performance on OOD detection. This is because more prototypes increase the flexibility to model complex distributions of the discriminant latent representation.

Table 3: Ablation study of loss weight $\lambda$ on CIFAR-100.

| $\lambda$ | Accuracy (↑) | ECE (↓) | AUROC (↑) |
|---|---|---|---|
| 0.01 | 81.9 ± 0.18 | 0.043 ± 0.003 | 0.801 ± 0.012 |
| 0.05 | 82.1 ± 0.16 | 0.037 ± 0.002 | 0.817 ± 0.009 |
| 0.1 | **82.5 ± 0.17** | **0.032 ± 0.002** | **0.829 ± 0.008** |
| 0.5 | 82.4 ± 0.15 | 0.039 ± 0.003 | 0.824 ± 0.006 |
| 1.0 | **82.5 ± 0.17** | 0.035 ± 0.002 | 0.820 ± 0.007 |

Table 4: Effect of loss given $\lambda = 0.1$ on CIFAR-100.

| Loss | Accuracy (↑) | ECE (↓) | AUROC (↑) |
|---|---|---|---|
| $\mathcal{L}_d$ | 81.3 ± 0.13 | 0.041 ± 0.003 | 0.813 ± 0.009 |
| $\mathcal{L}_e$ | 82.0 ± 0.19 | 0.033 ± 0.002 | 0.822 ± 0.007 |
| $\mathcal{L}_d$ & $\mathcal{L}_e$ | **82.5 ± 0.17** | **0.032 ± 0.002** | **0.829 ± 0.008** |

**Ablation Studies**. We further investigate the effect of constrained loss terms and the corresponding weights on the OOD detection performance. Table 3 and Table 4 show the ablation studies on the choice of loss weight $\lambda$ and loss terms $\mathcal{L}_{dissimilar}$ ($\mathcal{L}_d$) and $\mathcal{L}_{entropy}$ ($\mathcal{L}_e$) on CIFAR-100 task respectively. Note that the loss weight $\lambda = 0.1$ is the best choice for this case and the two

Table 5: OOD detection results for BERT on CLINC dataset, averaged over 10 random trials.

| Method | Accuracy (↑) | ECE (↓) | AUROC (↑) |
|---|---|---|---|
| DNN (Softmax) | 96.5 ± 0.11 | 0.024 ± 0.002 | 0.897 ± 0.01 |
| MC Dropout (Gal and Ghahramani, 2016) | 96.1 ± 0.10 | 0.021 ± 0.001 | 0.938 ± 0.01 |
| Deep Ensemble (Lakshminarayanan et al., 2017) | **97.5 ± 0.03** | **0.013 ± 0.002** | 0.964 ± 0.01 |
| DUQ (Van Amersfoort et al., 2020) | 96.0 ± 0.04 | 0.059 ± 0.002 | 0.917 ± 0.01 |
| SNGP (Liu et al., 2020) | 96.6 ± 0.05 | **0.014 ± 0.005** | **0.969 ± 0.01** |
| DDAR with 256 prototypes | **97.5 ± 0.04** | **0.014 ± 0.002** | **0.969 ± 0.01** |
| DDAR with 64 prototypes | 96.3 ± 0.05 | 0.017 ± 0.002 | 0.961 ± 0.02 |

constrained losses contribute consistent improvements individually and together.

### 4.3 Conversational Language Understanding

To demonstrate the effectiveness of the proposed distance-awareness representation on data modalities beyond images, we further evaluate the DDAR method on practical language understanding tasks where uncertainty estimation is highly critical: detecting out-of-scope dialog intent (Larson et al., 2019; Vedula et al., 2019; Yaghoub-Zadeh-Fard et al., 2020; Zheng et al., 2020). For a dialog system (e.g., chatbot) built for in-domain services, it is essential to understand if an input natural utterance from a user is out-of-scope or in-scope. In other words, the model should know when it abstains from or activates one of the in-domain services. To this end, this problem can be formulated as an OOD detection problem where we consider training an intent understanding model to detect in-domain services or out-of-domain services.

We follow the problem setup (Liu et al., 2020) and use the CLINC out-of-scope intent detection benchmark dataset (Larson et al., 2019) which contains 150 in-domain services data with 150 training sentences in each domain, and 1500 natural out-of-domain utterances. We train a BERT model only on in-domain data and evaluate their predictive accuracy on the in-domain test data, their calibration, and OOD detection performance on the combined in-domain and out-of-domain data. The results are shown in Table 5. In this case, we only compare the baseline methods, including DNN, MC Dropout, Deep Ensemble, DUQ, and SNGP. As shown, consistent with the previous vision experiments, our DDAR method shows competitive performance, which outperforms other single model approaches and is close to the deep ensemble in prediction accuracy and to SNGP in confidence calibration and OOD AUROC.

## 5 Related Work

**Deterministic Uncertainty Methods.** Unlike the conventional uncertainty estimation methods, including Bayesian Neural Networks (BNNs) (Osawa et al., 2019; Wenzel et al., 2020), MC Dropout (Gal and Ghahramani, 2016), and Deep Ensemble (Lakshminarayanan et al., 2017), a promising line of work recently emerged for estimating uncertainties of a DNN with a single forward pass while treating its weights deterministically (Postels et al., 2021). By regularizing the hidden representations of a model, these methods represent an efficient and scalable solution to uncertainty estimation and to the related out-of-distribution (OOD) detection problem. In contrast to BNNs, Deterministic Uncertainty Methods (DUMs) quantify epistemic uncertainty using the distribution of latent representations (Alemi et al., 2018; Wu and Goodman, 2020; Charpentier et al., 2020; Mukhoti et al., 2021b; Charpentier et al., 2021) or by replacing the final softmax layer with a distance-sensitive function (Mandelbaum and Weinshall, 2017; Van Amersfoort et al., 2020; Liu et al., 2020; van Amersfoort et al., 2021). Note that there is another line of work, which proposes a principled approach for variance propagation in DNNs (Postels et al., 2019; Haußmann et al., 2020; Loquercio et al., 2020) but these approaches fundamentally differ from DUMs due to their probabilistic treatment of the parameters, even though they are efficient approaches and also relied on a single forward pass for uncertainty estimation.

**Addressing Feature Collapse.** The critical challenge in DUMs is how to address the feature collapse issue. Currently, there are two main paradigms - distance awareness and informative representations. The distance awareness avoids feature collapse by relating distances between latent representations to distance in the input space. The primary methods are to impose the bi-Lipschitz constraint by using a two-sided gradient penalty (Van Amersfoort et al., 2020) or spectral normalization (Liu et al., 2020; van Amersfoort et al., 2021; Mukhoti et al., 2021a). While distance-awareness achieves remarkable performance on OOD detection, it does not explicitly preserve sample-specific information. An alternative line of work addresses this challenge by learning informative representations (Alemi et al., 2018; Wu and Goodman, 2020; Postels et al., 2020), thus forcing discriminative models to preserve information in its hidden representations. While distance-awareness is based on the choice of a specific distance metric tying together input and latent space, informative representations incentivize a DNN to store more information about the input. There are several approaches in this paradigm, including contrastive learning (Chen et al., 2020; Wu and Goodman, 2020; Winkens et al., 2020), reconstruction regularization (Postels et al., 2020), entropy regularization (Charpentier et al., 2020), and invertible neural networks (Behrmann et al., 2019; Ardizzone et al., 2020; Nalisnick et al., 2019; Ardizzone et al., 2018).

## 6 Conclusions

In this work, we develop a novel DDAR method for deterministic uncertainty estimation. This is achieved by learning a discriminant distance-aware representation that leverages a distinction maximization layer over a set of learnable prototypes. Compared with the baseline DUMs, our DDAR is a simple and efficient method without the feature collapse issue while the feature extractor is architecture-agnostic with higher flexibility not limited by residual neural networks. Through several experiments on synthesis data, image classification, and text OOD detection benchmarks, we show that DDAR outperforms the different SOTA baselines in terms of prediction accuracy, confidence calibration, and OOD detection performance.

The limitation of this work is a lack of a deep theoretical understanding of feature collapse although the empirical improvements are clearly shown. We plan to dig into the theoretical propriety of feature collapse with better model interpretability and explainability. Future work could also investigate the scalability of DDAR on large-scale computer vision tasks, and large language models (LLMs).

### References

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

Lynton Ardizzone, Jakob Kruse, Sebastian Wirkert, Daniel Rahner, Eric W Pellegrini, Ralf S Klessen, Lena Maier-Hein, Carsten Rother, and Ullrich Köthe. Analyzing inverse problems with invertible neural networks. *arXiv preprint arXiv:1808.04730*, 2018.

Lynton Ardizzone, Radek Mackowiak, Carsten Rother, and Ullrich Köthe. Training normalizing flows with the information bottleneck for competitive generative classification. *Advances in Neural Information Processing Systems*, 33:7828–7840, 2020.

Jens Behrmann, Will Grathwohl, Ricky TQ Chen, David Duvenaud, and Jörn-Henrik Jacobsen. Invertible residual networks. In *International Conference on Machine Learning*, pages 573–582. PMLR, 2019.

Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration inequalities: A nonasymptotic theory of independence*. Oxford university press, 2013.

Bertrand Charpentier, Daniel Zügner, and Stephan Günnemann. Posterior network: Uncertainty estimation without ood samples via density-based pseudo-counts. *Advances in Neural Information Processing Systems*, 33:1356–1367, 2020.

Bertrand Charpentier, Oliver Borchert, Daniel Zügner, Simon Geisler, and Stephan Günnemann. Natural posterior network: Deep bayesian uncertainty for exponential family distributions. *arXiv preprint arXiv:2105.04471*, 2021.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

Gianni Franchi, Xuanlong Yu, Andrei Bursuc, Emanuel Aldea, Severine Dubuisson, and David Filliat. Latent discriminant deterministic uncertainty. *arXiv preprint arXiv:2207.10130*, 2022.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR, 2016.

Yizhao Gao, Nanyi Fei, Guangzhen Liu, Zhiwu Lu, and Tao Xiang. Contrastive prototype learning with augmented embeddings for few-shot learning. In *Uncertainty in Artificial Intelligence*, pages 140–150. PMLR, 2021.

Jakob Gawlikowski, Cedrique Rovile Njieutcheu Tassi, Mohsin Ali, Jongseok Lee, Matthias Humt, Jianxiang Feng, Anna Kruspe, Rudolph Triebel, Peter Jung, Ribana Roscher, et al. A survey of uncertainty in deep neural networks. *Artificial Intelligence Review*, 56(Suppl 1):1513–1589, 2023.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR, 2017.

Manuel Haußmann, Fred A Hamprecht, and Melih Kandemir. Sampling-free variational inference of bayesian neural networks by variance backpropagation. In *Uncertainty in Artificial Intelligence*, pages 563–573. PMLR, 2020.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

Stefan Larson, Anish Mahendran, Joseph J Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K Kummerfeld, Kevin Leach, Michael A Laurenzano, Lingjia Tang, et al. An evaluation dataset for intent classification and out-of-scope prediction. *arXiv preprint arXiv:1909.02027*, 2019.

Gen Li, Varun Jampani, Laura Sevilla-Lara, Deqing Sun, Jonghyun Kim, and Joongkyu Kim. Adaptive prototype learning and allocation for few-shot segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8334–8343, 2021.

Jeremiah Liu, Zi Lin, Shreyas Padhy, Dustin Tran, Tania Bedrax Weiss, and Balaji Lakshminarayanan. Simple and principled uncertainty estimation with deterministic deep learning via distance awareness. *Advances in Neural Information Processing Systems*, 33:7498–7512, 2020.

Antonio Loquercio, Mattia Segu, and Davide Scaramuzza. A general framework for uncertainty estimation in deep learning. *IEEE Robotics and Automation Letters*, 5(2):3153–3160, 2020.

David Macêdo, Tsang Ing Ren, Cleber Zanchettin, Adriano LI Oliveira, and Teresa Ludermir. Entropic out-of-distribution detection. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2021.

David Macêdo, Cleber Zanchettin, and Teresa Ludermir. Distinction maximization loss: Efficiently improving classification accuracy, uncertainty estimation, and out-of-distribution detection simply replacing the loss and calibrating. *arXiv preprint arXiv:2205.05874*, 2022.

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

Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations*, 2018.

Jishnu Mukhoti, Andreas Kirsch, Joost van Amersfoort, Philip HS Torr, and Yarin Gal. Deep deterministic uncertainty: A simple baseline. *arXiv e-prints*, pages arXiv–2102, 2021a.

Jishnu Mukhoti, Andreas Kirsch, Joost van Amersfoort, Philip HS Torr, and Yarin Gal. Deterministic neural networks with appropriate inductive biases capture epistemic and aleatoric uncertainty. *arXiv preprint arXiv:2102.11582*, 2021b.

Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, Dilan Gorur, and Balaji Lakshminarayanan. Hybrid models with deep and invertible features. In *International Conference on Machine Learning*, pages 4723–4732. PMLR, 2019.

Kazuki Osawa, Siddharth Swaroop, Mohammad Emtiyaz E Khan, Anirudh Jain, Runa Eschenhagen, Richard E Turner, and Rio Yokota. Practical deep learning with bayesian principles. *Advances in neural information processing systems*, 32, 2019.

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, David Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. *Advances in neural information processing systems*, 32, 2019.

Janis Postels, Hermann Blum, Cesar Cadena, Roland Siegwart, Luc Van Gool, and Federico Tombari. Quantifying aleatoric and epistemic uncertainty using density estimation in latent space.

Janis Postels, Francesco Ferroni, Huseyin Coskun, Nassir Navab, and Federico Tombari. Sampling-free epistemic uncertainty estimation using approximated variance propagation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2931–2940, 2019.

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

Janis Postels, Mattia Segu, Tao Sun, Luc Van Gool, Fisher Yu, and Federico Tombari. On the practicality of deterministic epistemic uncertainty. *arXiv preprint arXiv:2107.00649*, 2021.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30, 2017.

Dustin Tran, Jeremiah Liu, Michael W Dusenberry, Du Phan, Mark Collier, Jie Ren, Kehang Han, Zi Wang, Zelda Mariet, Huiyi Hu, et al. Plex: Towards reliability using pretrained large model extensions. *arXiv preprint arXiv:2207.07411*, 2022.

Joost Van Amersfoort, Lewis Smith, Yee Whye Teh, and Yarin Gal. Uncertainty estimation using a single deep deterministic neural network. In *International conference on machine learning*, pages 9690–9700. PMLR, 2020.

Joost van Amersfoort, Lewis Smith, Andrew Jesson, Oscar Key, and Yarin Gal. On feature collapse and deep kernel learning for single forward pass uncertainty. *arXiv preprint arXiv:2102.11409*, 2021.

Nikhita Vedula, Nedim Lipka, Pranav Maneriker, and Srinivasan Parthasarathy. Towards open intent discovery for conversational text. *arXiv preprint arXiv:1904.08524*, 2019.

Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VII 14*, pages 499–515. Springer, 2016.

Florian Wenzel, Kevin Roth, Bastiaan Veeling, Jakub Swiatkowski, Linh Tran, Stephan Mandt, Jasper Snoek, Tim Salimans, Rodolphe Jenatton, and Sebastian Nowozin. How good is the bayes posterior in deep neural networks really? In *International Conference on Machine Learning*, pages 10248–10259. PMLR, 2020.

Jim Winkens, Rudy Bunel, Abhijit Guha Roy, Robert Stanforth, Vivek Natarajan, Joseph R Ledsam, Patricia MacWilliams, Pushmeet Kohli, Alan Karthikesalingam, Simon Kohl, et al. Contrastive training for improved out-of-distribution detection. *arXiv preprint arXiv:2007.05566*, 2020.

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

Mohammad-Ali Yaghoub-Zadeh-Fard, Boualem Benatallah, Fabio Casati, Moshe Chai Barukh, and Shayan Zamanirad. User utterance acquisition for training task-oriented bots: a review of challenges, techniques and opportunities. *IEEE Internet Computing*, 24(3):30–38, 2020.

Jiaxin Zhang. Modern monte carlo methods for efficient uncertainty quantification and propagation: A survey. *Wiley Interdisciplinary Reviews: Computational Statistics*, 13(5):e1539, 2021.

Jiaxin Zhang, Sirui Bi, and Victor Fung. On the quantification of image reconstruction uncertainty without training data. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2072–2081, 2024.

Yinhe Zheng, Guanyi Chen, and Minlie Huang. Out-of-domain detection for natural language understanding in dialog systems. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:1198–1209, 2020.