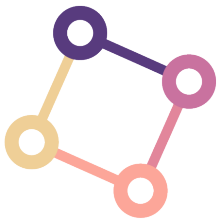
DENSITY ESTIMATION IN REPRESENTATION SPACE TO PREDICT MODEL UNCERTAINTY

Tiago Ramalho et al., arXiv preprint, 2019

VISION SEMINAR 2020/06/19



DAVIAN

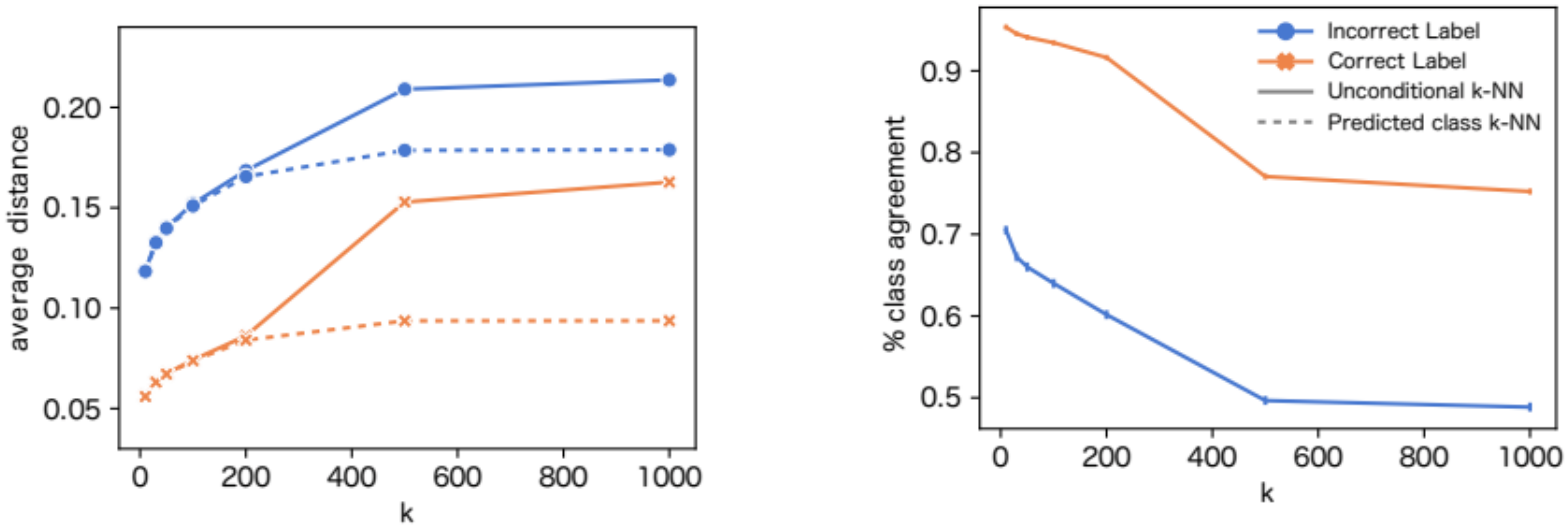
Data and Visual Analytics Lab

Overview

- **Motivations**
- **Methods**
- **Experiments**

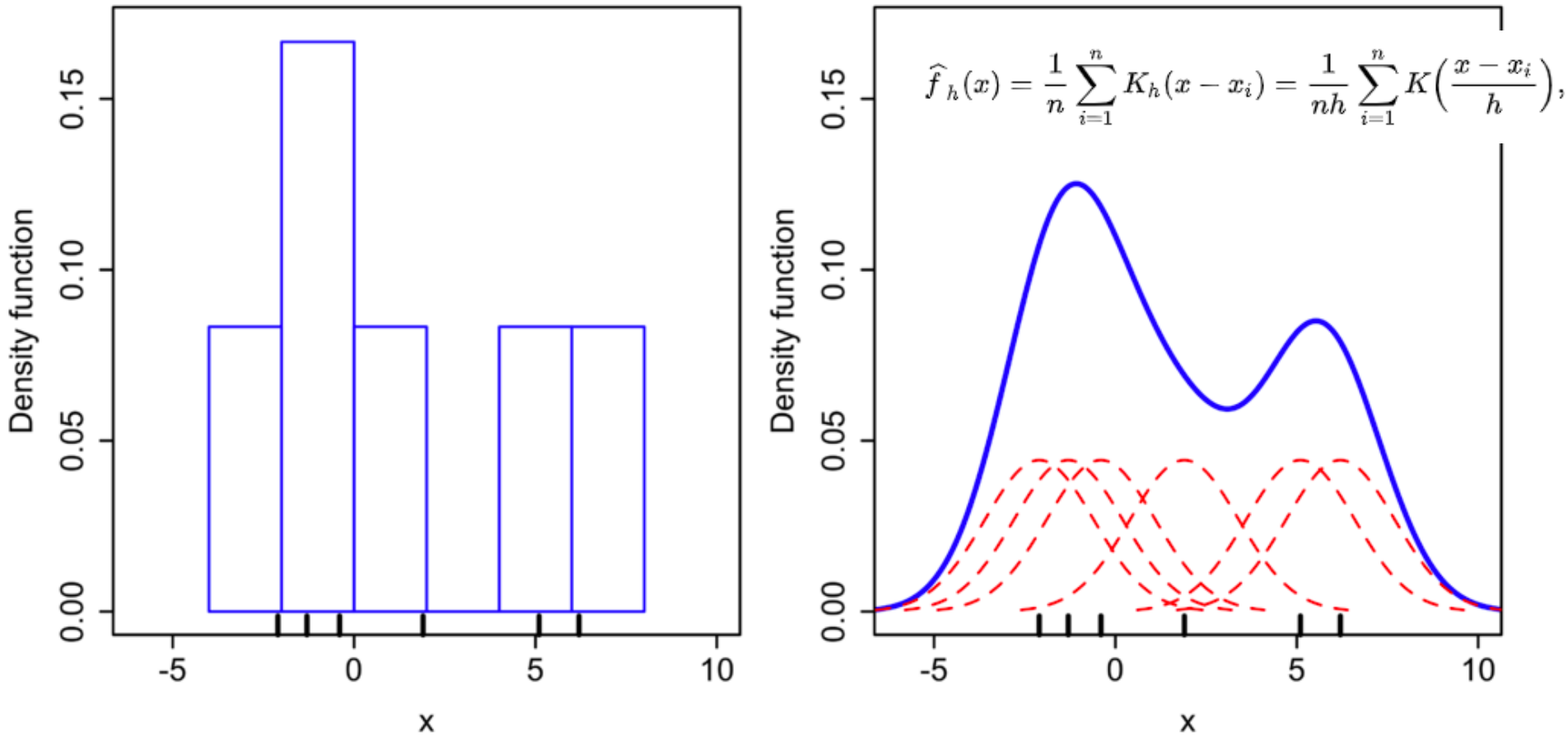
Motivations

- High confidence problem in modern neural networks. And out-of-distribution papers tackle this issue by measuring the uncertainty about predictions
- Measuring the uncertainty is the **estimating the density** using the neural networks
- This paper aims to **approximate the density of data point via neighborhood data.**



Quantifying the neighborhood statistics (ILSVRC2012 validation set) of the representation space of the last layer for Inception-ResNet-v2

Preliminary – Kernel Density Estimation



In this figure, K is a **gaussian kernel**

Methods: setting kernel differently

- The unconditional kernel estimate of x(i) with representation r(i)

$$P(x_i) \propto \sum_{j \in \mathcal{N}(r_i)}^{k} d(r_i, r_j)$$

- The class conditional kernel density estimate, which considers only neighbors with that predicted by the model

$$P(x_i|\hat{y}_i) \propto \sum_{j \in \mathcal{N}(r_i): y_j = \hat{y}_i}^{k} d(r_i, r_j)$$

Methods: Training with uncertainty estimation

- Uncertainty estimation with the neural networks

$$u(x_i) = g_\theta(\{(r_j, \mathbb{I}(y_j = \hat{y}_i)\}_{j \in \mathcal{N}(r_i)}, s(\hat{y}_i))$$

- Loss: We wish to minimize the uncertainty if the neighborhoods are in same class

In this dataset we minimize the binary cross entropy loss

$$\mathcal{L}[u, t] = -t_i \log(u(x_i)) \tag{5}$$

with the binary labels $t_i = 1$ if $y_i = \hat{y}_i$ and $t_i = 0$ otherwise.

Methods:Overview

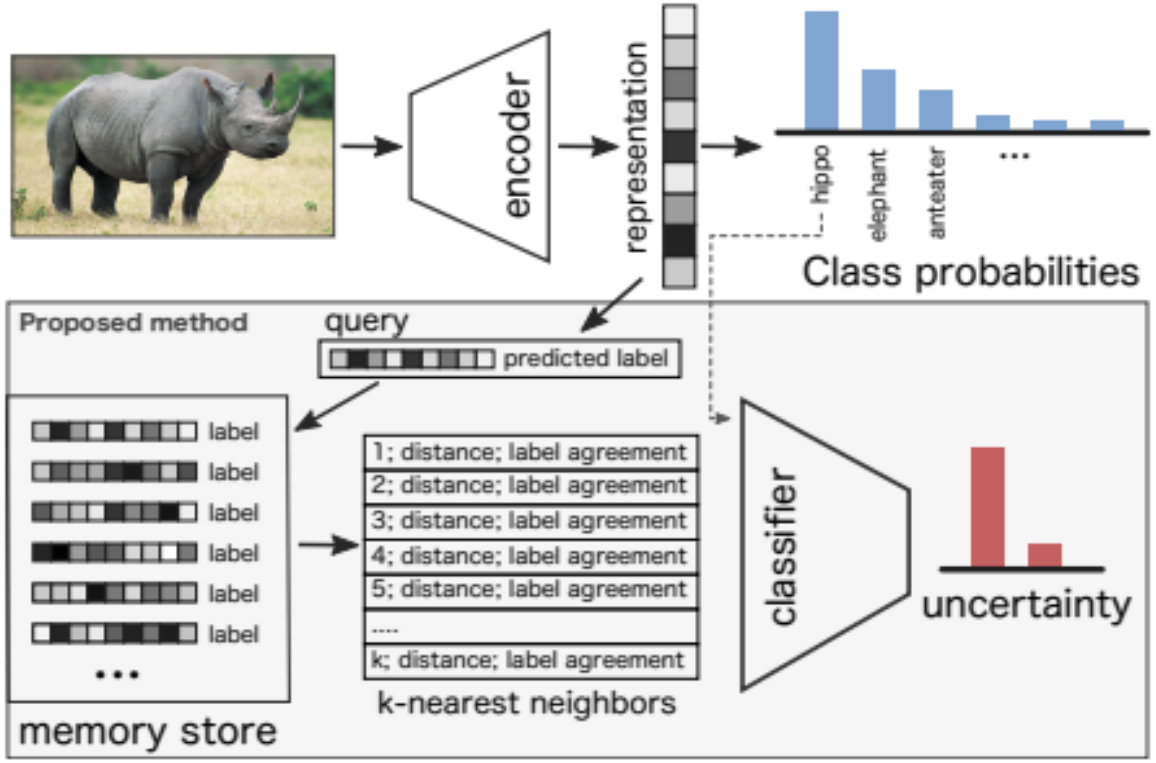


Figure 2: Schematic of the proposed model for uncertainty estimation. Top row, a deep neural network model is trained using the standard cross-entropy loss to produce class probabilities within a known set. The high level representation (before the logits layer) is additionally stored in a queriable memory database. When a new input is classified, its representation can be used to query the database, and information about its nearest neighbors is fed to the uncertainty model, which predicts the likelihood of the classification result being incorrect.

Methods: Training scheme

Algorithm 1 NUC training procedure

Require: k number of neighbors, $\mathcal{A}$ the set of all representations, θ model parameters, f a trained model

for epoch $\in$ number of epochs **do**
  for $x_i \in$ training set **do**
    $r_i \leftarrow \mathrm{f}^{N-1}(x_i)$
    $\hat{y}_i \leftarrow \mathbf{argmax}\ \mathrm{f}^{N}(r_i)$
    $t \leftarrow (y_i = \hat{y}_i)$
    $\{(r_j, y_j)\}_{j \in \mathcal{N}(r_i)} \leftarrow \mathbf{knn}(r_i, k, A)$
    $u(x_i) \leftarrow g_\theta(\{(r_j, \mathbb{I}(y_j = \hat{y}_i)\}_{j \in \mathcal{N}(r_i)}, s(\hat{y}_i))$
    $\theta_{t+1} \leftarrow \theta_t + \lambda \nabla_\theta \mathcal{L}\left[u(x_i), t\right]$
  end for
end for

Experiments (1/2)

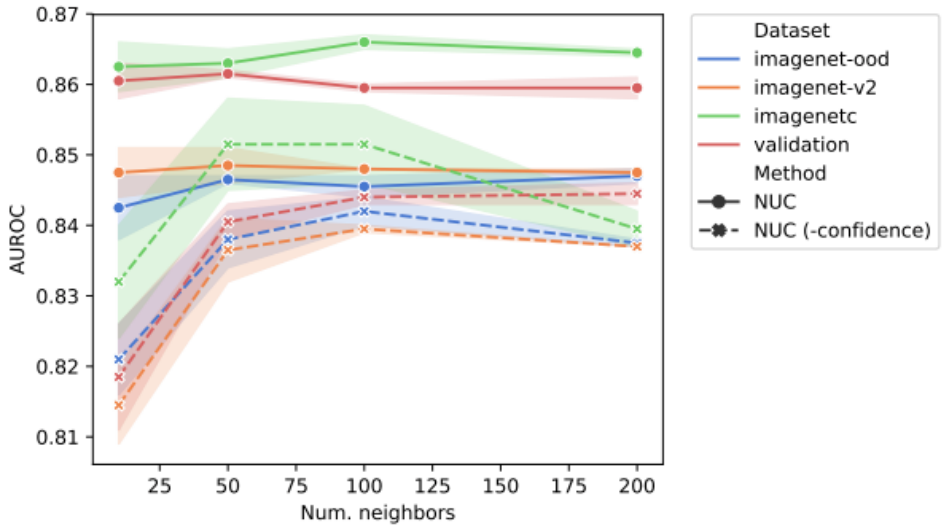


Figure 3: NUC performance as a function of number of neighbors k. Without taking the softmax prediction confidence into account, the model's performance is strongly dependent on k, as expected. For large values of k the performance slightly degrades as neighbors from potentially very far regions are being taken into account. The full model's performance, however, is relatively constant throughout different values of k.

Experiments: (2/2)

	$\text{ILSVRC2012}_{\text{valid.}}$	Imagenet-V2 [26]	Imagenet-C [10]
	AUROC / AUPR-Out / AUPR-In		
Softmax	0.844 / 0.587 / 0.945	0.826 / 0.682 / 0.896	0.848 / 0.812 / 0.852
Softmax†	0.848 / 0.590 / 0.948	0.829 / 0.681 / 0.899	0.849 / 0.811 / 0.854
Eq. 1	0.772 / 0.389 / 0.930	0.781 / 0.525 / 0.893	0.826 / 0.730 / 0.862
Eq. 2	0.773 / 0.398 / 0.931	0.781 / 0.541 / 0.892	0.829 / 0.743 / 0.865
Eq. 3	0.818 / 0.470 / 0.950	0.815 / 0.602 / 0.914	0.844 / 0.759 / 0.883
Mahalanobis	0.786 / 0.462 / 0.931	0.798 / 0.608 / 0.896	0.842 / 0.789 / 0.869
NUC	**0.862 / 0.600 / 0.959**	**0.845 / 0.690 / 0.923**	**0.862 / 0.820 / 0.889**

Table 1: Results for the in-distribution uncertainty quantification task. We predict classification mistakes (top-1) for the ILSVRC2012 validation set, Imagenet-V2 [26] new validation set and the Imagenet-C [10] dataset. We report the threshold independent metrics AUROC, AUPR-Out and AUPR-In [11]. Items marked with † used the calibration procedure described in [7].