Uncertainty Quantification for Classification

Sanika Padegaonkar
Electrical Engineering
IIT Bombay
Mumbai, India
20d070069@iitb.ac.in

***Abstract*—This report offers a comprehensive synthesis of the existing literature on Uncertainty Quantification and Estimation (UQE) in the context of classification tasks. It explores the diverse sources of uncertainty inherent in classification models and surveys the array of techniques employed for quantifying and estimating this uncertainty. Furthermore, the paper reports the outcomes of experiments conducted utilizing Test Time Augmentation (TTA) and offers detailed insights into these findings. By systematically reviewing the literature, this paper elucidates various techniques for uncertainty quantification in classification tasks and underscores the efficacy of TTA in enhancing classification performance.**

***Index Terms*—uncertainty quantification, estimation, classification, test time augmentation (TTA), aleatoric uncertainty, epistemic uncertainty.**

I. Introduction

Classification tasks are fundamental in various fields of machine learning and data analysis, ranging from image recognition to medical diagnosis. However, alongside the predictions they provide, classification models inherently come with a level of uncertainty. Understanding and quantifying this uncertainty is crucial for assessing the reliability of model predictions and making informed decisions based on them. Uncertainty Quantification and Estimation (UQE) techniques aim to address this challenge by providing measures of uncertainty associated with classification models.

The importance of uncertainty quantification in classification cannot be overstated. While classification models are typically trained to provide discrete predictions, they often operate in domains where the underlying data is inherently uncertain. For instance, in medical diagnosis, a classification model may need to predict whether a patient has a particular disease based on symptoms and test results. However, these symptoms and test results may not always be conclusive, leading to uncertainty in the model's predictions. Similarly, in image recognition tasks, environmental factors such as lighting conditions or occlusions may introduce uncertainty into the classification process.

Uncertainty in classification can arise from various sources, including the inherent stochasticity of the data (**aleatoric uncertainty**) and the model's inability to perfectly represent the underlying data distribution (**epistemic uncertainty**). Aleatoric uncertainty stems from the inherent variability in the observed data and is irreducible even with infinite amounts of data. On the other hand, epistemic uncertainty arises from the model's lack of knowledge about the true underlying data distribution and can be reduced with more data or improved model architectures. By quantifying and understanding both types of uncertainty, we can make more informed decisions about the **reliability of classification predictions** and better understand the limitations of our models.

In this report, a comprehensive review and synthesis of the existing literature on Uncertainty Quantification for Classification is presented. We delve into the various sources of uncertainty in classification models, including aleatoric and epistemic uncertainty, and examine the techniques used to quantify and estimate these uncertainties. Additionally, we explore the application of **Test Time Augmentation (TTA)** as a means to improve classification performance and provide detailed explanations for the observed results.

Through this review, we aim to provide insights into the state-of-the-art techniques for uncertainty quantification in classification tasks and offer a deeper understanding of the role of uncertainty in classification model predictions. Ultimately, our goal is to contribute to the development of more reliable and interpretable classification models.

II. Prior Work

A. Sources of Uncertainty in Machine Learning - A Statisticians' View [1]

It's fascinating to witness the strides made in machine learning and deep learning, which now allow us to tackle questions that were once considered inconceivable. However, beyond the realm of pure prediction, which supervised machine learning algorithms excel at, it has become evident that quantifying uncertainty is equally important. While initial concepts and ideas in this direction have surfaced in recent years, this paper takes a conceptual approach to examine potential sources of uncertainty.

Taking the perspective of a statistician, the authors delve into the concepts of aleatoric and epistemic uncertainty, which are commonly associated with machine learning. The aim of this paper is to formalize these two types of uncertainty and demonstrate that sources of uncertainty are diverse and not always easily decomposed into aleatoric and epistemic categories. By drawing parallels between statistical concepts and uncertainty in machine learning, the authors also highlight the role of data and how they influence uncertainty.

B. A Review of Uncertainty Quantification in Deep Learning: Techniques, Applications and Challenges [2]

This review paper differs from previous ones in the field of Uncertainty Quantification (UQ) by focusing on the most recent articles that quantify uncertainty in Artificial Intelligence (AI), specifically Machine Learning (ML) and Deep Learning (DL).

The study aims to explore how UQ can impact real-world cases by providing reliable results. By reviewing recent articles, the paper identifies various approaches to quantifying uncertainty in AI and highlights the potential impact of addressing uncertainty in ML and DL.

Furthermore, the paper identifies key challenges and important advancements in existing methods, providing valuable insights for future research in UQ within the ML and DL domains.

By summarizing recent studies and highlighting important trends, this review paper provides valuable input for future researchers working on UQ in ML and DL.

Some UQ methods mentioned in this paper are as follows:

1) Bayesian Deep Learning/Bayesian Neural Networks: Bayesian Neural Networks (BNNs) refers to extending standard networks with posterior inference in order to control overfitting. From a broader perspective, the Bayesian approach uses the statistical methodology so that everything has a probability distribution attached to it, including model parameters (weights and biases in neural networks).

2) Monte Carlo Dropout: Monte Carlo methods, or Monte Carlo experiments, are a broad class of computational algorithms that rely on repeated random sampling to obtain numerical results. It is difficult to compute the exact posterior inference, but it can be approximated. In this regard, Monte Carlo (MC) is an effective method.

Nonetheless, it is a slow and computationally expensive method when integrated into a deep architecture. To combat this, MC (MC) dropout has been introduced, which uses dropout as a regularization term to compute the prediction uncertainty.

3) Makov Chain Monte Carlo (MCMC): Markov Chain Monte Carlo (MCMC) is a method used to approximate complex posterior distributions. It starts by randomly drawing z_0 from the distribution $q(z_0)$ or $q(z_0|x)$. Then, it applies a stochastic transition to z_0, where z_t is drawn from $q(z_t|z_{t-1}, x)$. This transition is repeated for T iterations, and the resulting random variable converges in distribution to the exact posterior.

However, MCMC has its drawbacks. One challenge is determining the sufficient number of iterations T required for convergence, as it is often unknown. Additionally, MCMC algorithms can take a long time to converge, especially for high-dimensional or complex posterior distributions. This can make MCMC impractical for certain applications, particularly when computational resources are limited. Despite these limitations, MCMC remains a widely used and valuable tool for Bayesian inference in various fields.

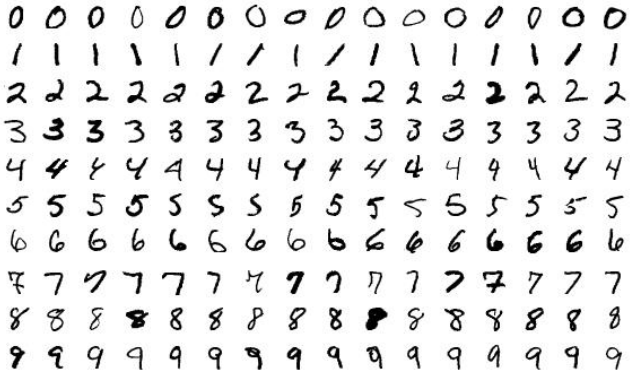

Fig. 1. MNIST Dataset

III. Datasets Used

A. MNIST

The MNIST database (Modified National Institute of Standards and Technology database) is a large collection of handwritten digits. It has a training set of 60,000 28x28 examples, and a test set of 10,000 examples. It is a subset of a larger NIST Special Database 3 (digits written by employees of the United States Census Bureau) and Special Database 1 (digits written by high school students) which contain monochrome images of handwritten digits. The digits have been size-normalized and centered in a fixed-size image. The original black and white (bilevel) images from NIST were size normalized to fit in a 20x20 pixel box while preserving their aspect ratio. The resulting images contain grey levels as a result of the anti-aliasing technique used by the normalization algorithm. the images were centered in a 28x28 image by computing the center of mass of the pixels, and translating the image so as to position this point at the center of the 28x28 field.

B. CIFAR10

The CIFAR-10 dataset consists of 60000 32x32 colour images in 10 classes, with 6000 images per class. There are 50000 training images and 10000 test images.

The dataset is divided into five training batches and one test batch, each with 10000 images. The test batch contains exactly 1000 randomly-selected images from each class. The training batches contain the remaining images in random order, but some training batches may contain more images from one class than another. Between them, the training batches contain exactly 5000 images from each class.

IV. Work Done

The aim of our experiments is to determine whether true prediction probabilities are provided by the softmax activation function. If not, the level of uncertainty associated with predictions will be assessed. Entropy is used as a measure of uncertainty in these experiments.

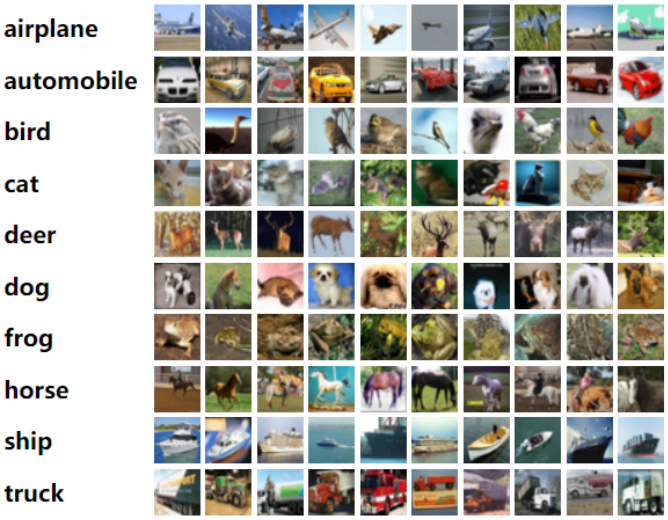

Fig. 2. CIDAR10 Dataset

A. Training

ResNet18 was trained on two different datasets: MNIST and CIFAR10. For the **MNIST** dataset, the model achieved a best validation accuracy of **99.2%** after **15 epochs** of training.

On the **CIFAR10** dataset, ResNet18 achieved its best accuracy of **81.59%** after training for **100 epochs**.

B. Test Time Augmentation

Test Time Augmentation (TTA) is a technique used to improve the performance of a trained model during the inference phase. Instead of presenting the model with the regular, "clean" test images only once, TTA applies random augmentations to these images and presents them to the model multiple times. The model then makes predictions on each of these augmented images, and the final prediction for each test image is determined by averaging or taking the mode of the predictions made on all corresponding augmented images. This approach helps improve the model's accuracy by considering variations of the test images, making predictions more robust.

TTA was performed on the above mentioned datasets and the entropy of the predictions on augmented samples of the same image was observed.

A single test sample was augmented 15 times and mode of these 15 predictions was considered as the final prediction.

C. Entropy Calculation

Softmax was applied on the 15 predictions of augmented images, giving us 15 probabilities. Entropy was calculated using these probabilities.

D. Defining 'good' and 'bad' predictions

In traditional classification tasks, higher entropy typically indicates higher uncertainty, as it signifies that the model is less confident in its prediction. However, when it comes to Test-Time Augmentation (TTA), where predictions are made on augmented samples, the interpretation of entropy is somewhat inverted.

In standard classification scenarios, the goal is to have the model make predictions with high confidence, which usually means low entropy. High confidence in a prediction means that the model can clearly distinguish between different classes, resulting in a low entropy value.

However, when employing TTA, the objective is slightly different. Instead of making predictions based on a single input sample, TTA generates multiple augmented versions of the same sample and makes predictions on each of these augmented samples. The idea is to improve model performance by averaging predictions across these different augmentations.

In the context of TTA, we want the predictions to be consistent across the augmented samples. If the model makes consistent predictions across all the augmented samples, it suggests that the model is robust to small variations in the input data. Consequently, in TTA, a good prediction is one where the model's predictions across the augmented samples are similar, resulting in higher entropy.

In summary, while lower entropy typically indicates better predictions in traditional classification tasks, in the case of TTA, higher entropy is indicative of more reliable predictions. Therefore, in TTA, we would consider predictions with **higher entropy as more reliable and less uncertain**.

E. Dropping samples

To improve the accuracy of the model's predictions, a dropout mechanism is applied to a fraction of the test set containing **bad predictions**, i.e., images with **lower entropy** (as we want the same prediction on all augmented samples) compared to others. This is achieved by arranging the test set images in ascending order of their entropies and then dropping a certain percentage of the images, determined by the drop rate. By increasing the drop rate from 10% to 90%, the expectation is to increase the overall accuracy, as the dropout mechanism removes the most uncertain predictions, allowing the model to focus on the more confident ones.

F. Results on MNIST

G. Results on CIFAR10

Conclusion: These results prove our supposition that eliminating 'bad' or 'uncertain' predictions from our dataset improves accuracy.

V. Learnings

This project has provided me with a deeper understanding of the different sources of uncertainty in machine learning and the various methods used to measure it. Additionally, I was introduced to the concept of test time augmentation and learned how it can enhance the robustness of predictions. I also learnt about entropy in machine learning and the behaviour of entropy curves in different settings.

VI. Conclusion

In conclusion, a conceptual perspective on Uncertainty Quantification for Classification has been provided in this paper, highlighting the importance of quantifying uncertainty in

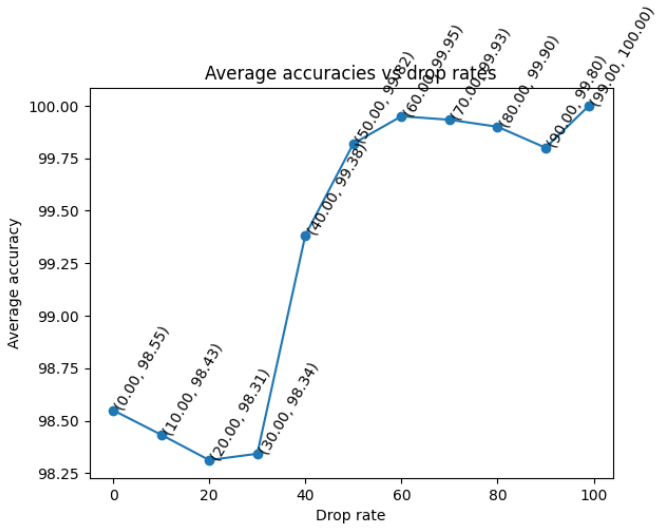

Fig. 3. Average accuracy vs Drop rate

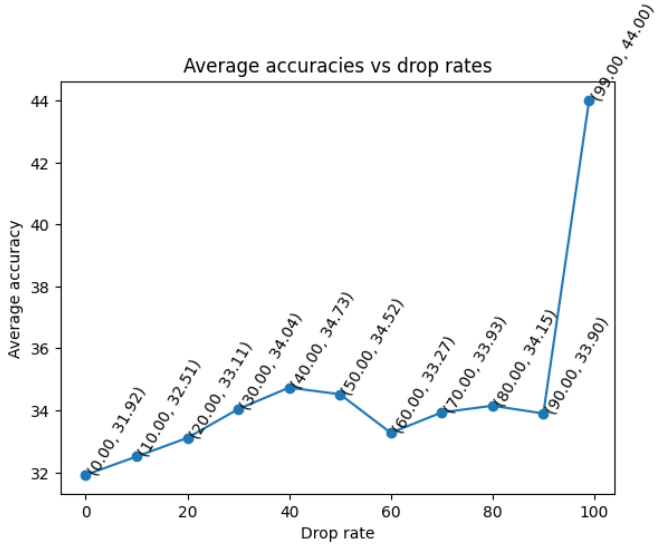

Fig. 4. Average accuracy vs Drop rate

machine learning models beyond pure prediction. By adopting a statistical viewpoint, the concepts of aleatoric and epistemic uncertainty were discussed, demonstrating their relevance in the context of machine learning.

Various sources of uncertainty in classification models were explored, and techniques used to quantify and estimate these uncertainties were examined. Additionally, the application of Test Time Augmentation (TTA) as a means to improve classification performance was investigated, and detailed explanations for the observed results were provided.

Through this review, a deeper understanding of uncertainty quantification in classification tasks has been contributed, providing insights into state-of-the-art techniques and their applications. By acknowledging and quantifying uncertainty, more reliable and interpretable classification models can be developed, ultimately advancing the field of machine learning and data analysis.

REFERENCES

[1] Gruber, C., Schenk, P. O., Schierholz, M., Kreuter, F., & Kauermann, G. (2023). Sources of Uncertainty in Machine Learning–A Statisticians' View. arXiv preprint arXiv:2305.16703.

[2] Abdar, M., Pourpanah, F., Hussain, S., Rezazadegan, D., Liu, L., Ghavamzadeh, M., ... & Nahavandi, S. (2021). A review of uncertainty quantification in deep learning: Techniques, applications and challenges. Information fusion, 76, 243-297.

[3] Sensoy, M., Kaplan, L., & Kandemir, M. (2018). Evidential deep learning to quantify classification uncertainty. Advances in neural information processing systems, 31.

[4] Jospin, L. V., Laga, H., Boussaid, F., Buntine, W., & Bennamoun, M. (2022). Hands-on Bayesian neural networks—A tutorial for deep learning users. IEEE Computational Intelligence Magazine, 17(2), 29-48.