Evaluating Scalable Bayesian Deep Learning Methods for Robust Computer Vision. Fredrik K. Gustafsson, Martin Danelljan, Thomas B. Schon

Abhishek Sabnis
ME18BTECH11049
IIT Hyderabad.

April 30, 2022

Introduction

Although DL technique works, it's most of the time unclear why deep learning works. The majority of these models fail to properly capture the uncertainty inherent in their predictions. Estimating this predictive uncertainty can be crucial, for example in automotive application. If the car is unsure where there is a pedestrian on the road, we would expect it to let the driver take charge.

While different scalable methods recently have emerged, no extensive comparison has been performed in a real-world setting. This paper proposes a comprehensive evaluation framework for scalable uncertainty estimation methods in deep learning.

Types of Uncertainty

There are two major different types of uncertainty in deep learning: epistemic uncertainty and aleatoric uncertainty.

Epistemic Uncertainty: Accounts for uncertainty in the DNN model parameters, due to limited training data

Aleatoric Uncertainty: It is the uncertainty arising from the natural stochasticity of observations. Aleatoric uncertainty cannot be reduced even when more data is provided.

Types of Uncertainty

In many computer vision applications, this aleatoric uncertainty can be effectively estimated by letting a DNN directly output the conditional distribution $p(y|x)$ of the target given the input.

But this does not capture epistemic uncertainty, as information about the uncertainty in the model parameters is disregarded. This often leads to highly confident predictions that are incorrect, especially for inputs x that are not well-represented by the training distribution.

Estimating Aleatoric Uncertainty

We now formulate equation to estimate Aleatoric Uncertainty of DNN model.

In classification problems, aleatoric uncertainty is commonly captured by predicting a categorical distribution $p(y|x,)$. This is implemented by letting the DNN predict logit scores $f_\theta(x) \in R^C$, which are then normalized by a Softmax function.

$$\begin{aligned} p(y|x,\theta) &= Cat(y; s_\theta(x)), \\ s_\theta(x) &= Softmax(f_\theta(x)). \end{aligned} \tag{1}$$

Estimating Aleatoric Uncertainty

Given a training set of i.i.d. sample pairs $D = \{X, Y\} = \{(x_i, y_i)\}_{i=1}^{N}, (x_i, y_i) \sim p(x, y)$, the data likelihood is obtained as $p(Y|X, \theta) = \prod_{i=1}^{N} p(y_i|x_i, \theta)$.

The MLE of the model parameters, θ_{MLE}, is obtained by minimizing negative log likelihood $-\sum_i logp(y_i|x_i, \theta)$. For Categorical model this is equivalent to minimizing Cross entropy loss.

Estimating Aleatoric Uncertainty

In regression, the most common approach is to let DNN directly predict targets, $y^* = f_\theta\ (x^*)$. The parameters $\hat{\theta}$ is learned by minimizing L^2 loss. But this does not capture aleatoric uncertainty. Instead, we predict the distribution $p(y|x, \theta)$, similar to classification case.

$$\begin{aligned} p(y|x, \theta) &= N(y; \mu_\theta(x), \sigma^2_\theta(x)), \\ f_\theta(x) &= [\mu_\theta(x), log\sigma^2_\theta(x)]. \end{aligned} \tag{2}$$

The model parameters are learnt by minimizing negative log likelihood $-\sum_i logp(y_i|x_i, \theta)$

Estimating Epistemic Uncertainty

While the above model captures the uncertainty in data, we are still yet to capture the uncertainty arising from model parameter θ. The aim is to utilize the posterior distribution $p(\theta|D)$, which is obtained from data likelihood and a chosen prior $p(\theta)$ by applying Bayes Theorem.

$$\begin{aligned} p(y^*|x^*, D) &= \int p(y^*|x^*, \theta)p(\theta|D)d\theta, \\ &= \frac{1}{M}\sum_{i=1}^{M} p(y^*|x^*, \theta^{(i)}), \theta^{(i)} \sim p(\theta|D). \end{aligned} \tag{3}$$

where (x^*, y^*) is test data set and D is the given training data.

Estimating Epistemic Uncertainty

For the Categorical model, $\hat{p}(y^*|x^*, D) = Cat(y^*; \hat{s}(x^*))$, where $\hat{s}(x^*) = \frac{1}{M}\sum_{i=1}^{M} s_{\theta(i)}(x^*)$ and $s_{\theta(i)}(x^*)$ is the softmax of logits function.

For the Regression model, we use uniformly weighted mixture of Gaussian model to calculate $\hat{p}(y^*|x^*, D)$. We approximate this mixture with a single Gaussian as follows -

Estimating Epistemic Uncertainty

$$
\begin{aligned}
p(y^*|x^*, D) &= \frac{1}{M}\sum_{i=1}^{M} p(y^*|x^*, \theta^{(i)}) \\
&= \frac{1}{M}\sum_{i=1}^{M} N(y^*; \mu_{\theta^{(i)}}(x^*), \sigma^2_{\theta^{(i)}}(x^*)) \\
&= N(y^*; \hat{\mu}(x^*), \hat{\sigma}^2(x^*))
\end{aligned} \tag{4}
$$

where $\hat{\mu}(x) = \frac{1}{M}\sum_{i=1}^{M} \mu_{\theta^{(i)}}(x)$,

$\hat{\sigma}^2(x) = \frac{1}{M}\sum_{i=1}^{M}((\mu_{\theta^{(i)}}(x) - \hat{\mu}(x))^2 + \sigma^2_{\theta^{(i)}}(x)))$

Popular Framework

Among scalable methods of Bayesian inference, MC dropout and Ensembling are the widely employed methods.

While scalable techniques for epistemic uncertainty estimation recently have emerged, the research community however lacks a common and comprehensive evaluation framework for such methods.

The paper propose an evaluation framework that actually enables a conclusive ranking of the compared methods.

MC Dropout

Modeling uncertainty with Monte Carlo dropout works by running multiple forward passes trough the model with a different dropout masks every time.To derive the uncertainty for one sample x, we collect the predictions of T inferences with different dropout masks.

By computing the average and the variance of this sample we get an ensemble prediction, which is the mean of the models posterior distribution for this sample and an estimate of the uncertainty of the model regarding x.

Ensembling

We create a parametric model $p(y|x, \theta)$ of the conditional distribution using a DNN, and learn multiple point estimates $\hat{\theta}^{(m)}$ by repeatedly minimizing MLE with random initialization. We then average over the parametric models to obtain the following predictie distribution -

$$\hat{p}(y^*|x^*) = \frac{1}{M} \sum_{m=1}^{M} p(y^*|x^*, \theta^m) \tag{5}$$

Evaluation Metrics

Area Under the Sparsification Error curve (AUSE)

We evaluate the methods in terms of the Area Under the Sparsification Error curve (AUSE) metric. The metric reveals how well the estimated uncertainty can be used to sort predictions from worst (large true prediction error) to best (small prediction error). We compute AUSE in terms of Root Mean Squared Error (RMSE) and based on all pixels in the entire evaluation dataset.

Evaluation Metrics

Area Under the Calibration Error curve (AUCE)

Since our models output the mean μ and variance σ^2 of a Gaussian distribution for each pixel, we can construct pixel-wise prediction intervals $\mu \pm \Phi^{-1}(\frac{p+1}{2})\sigma$ of confidence level $p \in [0, 1]$, where Φ is the CDF of the standard normal distribution.

When computing the proportion of pixels for which the prediction interval covers the true target $y \in R$, we expect this value, denoted by $\hat{p}$, to equal $p \in [0, 1]$ for a perfectly calibrated model.

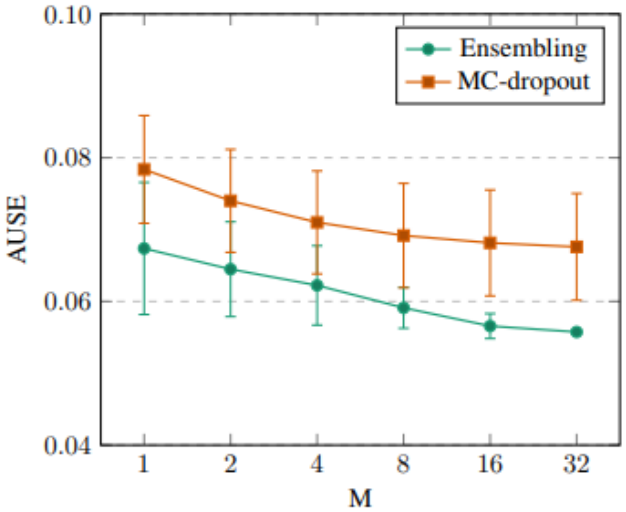
We compute the absolute error with respect to perfect calibration, $|p - \hat{p}|$, for 100 values of $p \in [0, 1]$ and use the area under this curve as our metric, which we call Area Under the Calibration Error curve (AUCE).

Depth Completion

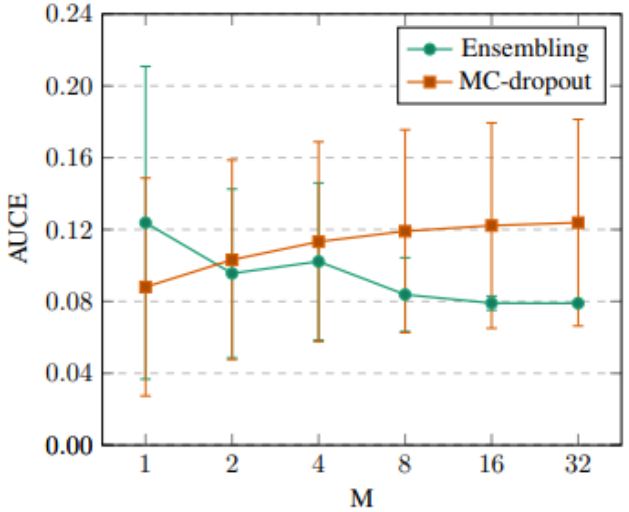
In depth completion, we are given an image $x_{img} \in R^{h \times w \times 3}$ from a forward-facing camera and an associated sparse depth map $x_{sparse} \in R^{h \times w}$

DNN model based on Resnet 34 architecture is used. Gaussian model (2) is employed in the final layer, outputting both μ and $log\sigma^2$. We now train the model using both MC dropout and Ensembling techniques.

Depth Completion



(a) AUSE (↓).



(b) AUCE (↓).

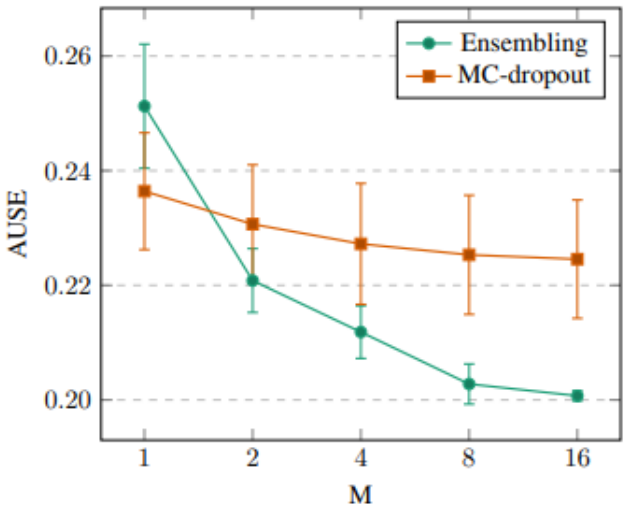
Figure: The plots show a comparison of ensembling and MC-dropout in terms of AUSE, AUCE on the depth completion validation dataset, for different number of samples M

Street-Scene Semantic Segmentation

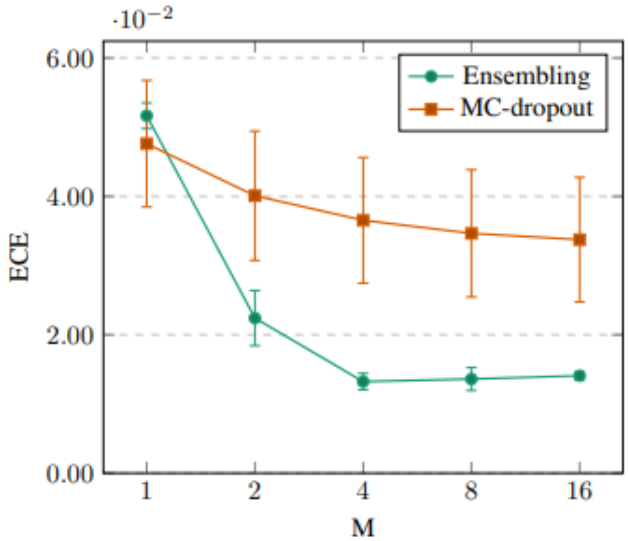
We are given an image $x_{img} \in R^{h \times w \times 3}$ from a forward-facing camera The goal is to predict y of size h $\times$ w, in which each pixel is assigned to one of C different class label.

The input image x is processed by a ResNet101. The feature maps is further processed and upsampled using bilinear interpolation. The conventional Categorical model (1) is used for each pixel. Finally trained using MC Dropout and Ensembling techniques.

Street-Scene Semantic Segmentation



(a) AUSE (↓).



(b) ECE (↓).

Figure: The plots show a comparison of ensembling and MC-dropout in terms of AUSE, AUCE on the Street-scene semantic segmentation validation dataset, for different number of samples M

Conclusion

Provided the first properly extensive and conclusive comparison of ensembling and MC-dropout, the results of which demonstrates that ensembling consistently provides more reliable and practically useful uncertainty estimates.

We observe that the metrics clearly improve as functions of M for both ensembling and MC-dropout, demonstrating the importance of epistemic uncertainty estimation.

MC-dropout has a large design-space compared to ensembling. But the success of ensembling is due to the random initialization, to capture the important aspect of multi-modality present in the posterior distribution.

References

Evaluating Scalable Bayesian Deep Learning Methods for Robust Computer Vision. Fredrik K. Gustafsson, Martin Danelljan, Thomas B. Schon¨
Department of Information Technology, Uppsala University, Sweden
Computer Vision Lab, ETH Zurich, Switzerland