Syed Ashar Javed

Learning to learn

Blog About

Uncertainty Estimation in Deep Learning

Written on January 10, 2019

Neural networks have seen amazing diversification in its applications in the last 10 years. The effectiveness of deep learning as complex function approximators has allowed us to go past many benchmarks across domains. Models using some form of deep learning have been widely adopted for the real-world tasks, which has brought to the fore, a very important topic of model confidence. More often than not, the predictions from the deep network are used for some downstream decision-making. A semantic segmentation map produced by a CNN is used to plan future trajectories of the driverless cars. The credit scores from a model are used towards loan approval/denial decisions. Thus it makes sense to have reasonable estimates of uncertainty of our model's predictions. It is interesting to note that many papers from more than a couple of decades ago tried to solve this problem for neural networks through its Bayesian treatment. The first ideas behind Bayesian Neural Networks (BNNs) can be found as early as 1992-1995 in various works by David Mackay [1], Radford Neal [2] and Hinton and Van Camp[3]. A great keynote talk about the history of these ideas was given by Zoubin Ghaharmani in the NIPS 2016 workshop. More recently, Yarin Gal came up with a Bayesian interpretation of dropout based deep models which has resulted in a flurry of research into this area (not to mention funny comments like this post from Ferenc and the comical cartoon below!) In this post, I would like to summarize few interesting papers in the uncertainty estimation area in the recent literature.

Overview of methods

Before jumping into the specific papers, it's always nice to have some mental buckets into which all approaches can be categorized. In my limited knowledge, most works in uncertainty estimation in neural networks fall under two heads. The first belongs to the category of treating the network as a Bayesian model, having a prior distribution over its weights and using data to learn a posterior distribution. The problem of doing inference in such setups has been solved through either Markov Chain Monte Carlo (MCMC) based methods (like the ones by Mackay mentioned earlier) or variational methods (see this great lecture slide). Majority of papers follow this paradigm and these methods are often clubbed together as Bayesian Deep Learning. The second category, for the lack of a better name, is the non-Bayesian one. Many recent papers have explored ideas other than approximate Bayesian NNs, some trying to obtain a frequentist estimate of uncertainty while others try to enforce an explicit minimization of KL divergences of certain distributions of in-domain and out-of-domain samples while yet others use adversarial samples or contrastive samples to build uncertainty estimates. The key ideas from some of these papers is given below. The main intention here is to give a breadth of the kind of techniques being explored.

Types of uncertainty

It is also useful to quickly describe the different types of predictive uncertainties being estimated in these papers. The first is the model uncertainty or epistemic uncertainty which is a consequence of mis-specification of the model or its parameters, for some given data. The second is the data uncertainty or aleatoric uncertainty which is a result of complexity or noisy nature of the data. This itself is further divided into homoscedastic and heteroscedastic uncertainty where the former is constant across data samples while the latter changes with the inputs. Some works also address a third type of uncertainty called the distributional uncertainty, which is the uncertainty in prediction due to a change in the data distribution from train to test.

Papers

1. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning by Gal et al [4]

- First paper to formulate stochastic regularization techniques like dropout in deep learning as approximate Bayesian inference.
- Shows that training a neural network with dropout is equivalent to doing approximate variational inference in a probabilistic deep Gaussian process. This means that when dealing with the predictive distribution, we can simply have a Bernoulli (or other depending on the kind of stochastic noise being injected) distribution over the weights and then doing multiple forward passes through the network and averaging them will be the same as doing Monte Carlo integration to find the expected output value of the model under the predictive distribution (it's called Monte Carlo dropout).
- For a much more detailed (and amazingly lucid!) explanation, please see chapter 3.2 in Gal's thesis where he explains how variational inference in Bayesian NNs is the same as Monte Carlo dropout for various stochastic regularization techniques, and how moment matching can be used to obtain uncertainty estimates in this formulation.

2. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? by Kendall et al [5]

- A follow up paper by Kendall and Gal discussing two kinds of uncertainties, namely epistemic and aleatoric.

- Show how both uncertainty estimates can be obtained from the same model. Use Monte Carlo dropout for epistemic uncertainty as shown in the previous paper, and predict a variance term using the model itself to handle aleatoric uncertainty of each input. The usual regression loss function is extended with variance term which makes the variance high whenever the model outputs a very wrong value.
- Show some really nice (state-of-art) results on real-world semantic segmentation and depth regression datasets.

3. Weight Uncertainty in Neural Networks by Blundell et al [6]

- Another early work post deep learning and variational autoencoders (VAEs) which learns a distribution over neural network weights using variational inference, similar to techniques mentioned in Gal's thesis.
- Apply the reparameterization trick to get a variational approximation to the distribution over weights as opposed to distribution over hidden units as done in VAE papers. Use a scale Gaussian mixture as prior combined with a diagonal Gaussian posterior distribution. Also has a small interesting paragraph on why optimizing the prior distribution's parameters based on the data (empirical Bayes) does not work for their model.
- Show that their uncertainty estimates for the weights can be used to decide exlploration strategies in contextual bandits with Thomson sampling by modeling the conditional reward distribution using their neural network. Interesting idea!

4. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles by Lakshminarayanan et al [7]

- A departure from the usual Bayesian modeling for getting uncertainty estimates through ensembling and adversarial training. A side-effect of being non-Bayesian is that it involves no mathematical guarantees like previous work (this of course is not a general comment on non-Bayesian techniques).
- Define proper scoring rules which can be used to measure the quality of the predictive distribution, which it turns out, are many of the standard loss functions used for training neural networks. The scoring rules are then used with adversarial training to smooth the predictive distribution and ensembles are used as uniformly-weighted mixture models to get the predicted value (mean) and the variance associated with the prediction, by assuming the output conditional distribution to be a mixture of Gaussian.
- Also show empirically that another intuitive solution of using the ensemble to make multiple predictions and then using the empirical variance in output as measure of uncertainty does not provide good estimates (consistently underestimates it). The final method is fairly simple and is shown to give better estimates than Monte Carlo dropout.

5. Bayesian Uncertainty Estimation for Batch Normalized Deep Networks by Teye et al [8]

- Shows that just like Monte Carlo dropout, using batch normalization in neural networks can also be cast as doing approximate Bayesian inference. This is possible due to the stochasticity involved in sampling mini-batches for getting batch statistics in batch normalization.
- Similar to the analysis done in Gal's thesis, this paper too shows equivalence of the variational loss in a neural network and a network trained with Batch norm in all its layers. However, it also makes certain other assumptions about the nature of each batch norm layer, the units in each layer being uncorrelated and so on in order to cast the prior term in the objective as weight decay.

6. Accurate Uncertainties for Deep Learning Using Calibrated Regression by Kuleshov et al [9]

- Propose a method for obtaining well-calibrated uncertainty estimates for Bayesian NNs using the post-hoc recalibration motivated by Platt's scaling. Intuitively, a calibrated model means that for an input, whenever it predicts an output with a probability 0.7, then that output should occur 70% of the time. See their first couple of sections to read more about calibration and sharpness of model predictions.
- Train an auxiliary regression model to recalibrate uncertainty estimates. This model is formulated as estimating a CDF arrived at through earlier defined properties of calibrated models. This estimation is done on a separate calibration set of data to prevent overfitting and uses a prior work called isotonic regression model which is non-parametric and hence can learn the true distribution with enough iid data.
- The proposed method also works with probabilistic predictions such as the ones from a Bayesian NN (they show it for paper 1 and 4 along with others). Thus this method can be used with any black box methods to recalibrate their uncertainty estimates.

7. Reliable Uncertainty Estimates In Deep Neural Networks Using Noise Contrastive Priors by Hafner et al [10]

- Many earlier Bayesian NN methods use a standard normal prior over weights which imposes weight shrinkage in the form of weight decay in the final objective. Thus these priors are uninformative about the function class and data and only depend on the parameterization which can cause the posterior to generalize to out-of-distribution (OOD) samples not seen during training. This paper proposes a new contrastive prior which explicitly ensures high uncertainty for OOD samples.
- Since generating a OOD data means finding the complement of training distribution, which is tricky, the paper uses few key ideas. First is to approximate the OOD input using random contrastive noise (motivated by noise contrastive estimation). Another is to encourage high

uncertainty at data points close to the boundary of the training distribution and let this effect propagate through the OOD space.

- These contrastive data points are used during training and the prior KL loss term for these is added to the final objective, which can be interpreted as minimization of KL divergence on pseudo-data points from the OOD inputs. This kind of prior is used to extend the work from paper 3 from above and show good uncertainty estimates on small datasets.

References

1. A Practical Bayesian Framework for Backprop Networks ↩
2. Bayesian Learning for Neural Networks ↩
3. Keeping Neural Networks Simple by Minimizing the Description Length of the Weights ↩
4. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning ↩
5. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? ↩
6. Weight Uncertainty in Neural Networks ↩
7. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles ↩
8. Bayesian Uncertainty Estimation for Batch Normalized Deep Networks ↩
9. Accurate Uncertainties for Deep Learning Using Calibrated Regression ↩
10. Reliable Uncertainty Estimates In Deep Neural Networks Using Noise Contrastive Priors ↩

0 Comments **Learning to learn** 1 **Login**

♡ **Recommend** Tweet Share **Sort by Best**

Start the discussion…

LOG IN WITH

OR SIGN UP WITH DISQUS ?

Name

Be the first to comment.

ALSO ON **LEARNING TO LEARN**

REINFORCE vs Reparameterization trick – Syed Ashar Javed

3 comments • a year ago

Tom Diethe — Yes that's it, although in the case where it's not proper, exchange the integral for a limit process with uniform

Variational Inference and Expectation Maximization – Syed Ashar Javed

2 comments • 2 years ago

Ashar Javed — Welcome Gabriel!

Subscribe **Add Disqus to your siteAdd DisqusAdd** **Disqus' Privacy PolicyPrivacy PolicyPrivacy**