HWA: AVERAGING HYPERPARAMETERS IN BAYESIAN NEURAL NETWORKS LEADS TO BETTER GENERALIZATION.

Anonymous authors
Paper under double-blind review

ABSTRACT

Bayesian Deep Learning presents itself as the most useful tool for adding uncertainty estimation to traditional Deep Learning models that only produce point estimates predictions as outputs. Confidence of the model and the predictions at inference time are left alone. Applying randomness and Bayes Rule to the weights of a deep neural network is a step towards achieving this goal. Current state of the art optimization method for training a Bayesian Neural Network are relatively slow and inefficient, compared to their deterministic counterparts. In this paper, we propose HWA (Hyperparameters Weight Averaging) algorithm that leverages the averaging procedure of Polyak and Ruppert in order to train faster and achieve a better accuracy. We develop our main algorithm using the simple averaging heuristic and demonstrate its effectiveness on the space of the hyperparameters of the neural networks random weights. Numerical applications confirm the empirical benefits of our method.

1 INTRODUCTION

Our main contributions read as follows:

- ff
- ff

The remaining of the paper is organized as follows.

2 RELATED WORK

Stochastic Averaging:

Variational Inference:

Posterior Prediction:

3 Hyperparameters Averaging in Bayesian Neural Networks

Algorithm 1 HWA: Hyperparameters Weight Averaging

```
**Input:** Trained hyperparameters $\hat{\mu}_\ell$ and $\hat{\sigma}$. LR bounds $\gamma_1$ and $\gamma_2$. Cycle length $c$.
Initialize the hyperparameters of the weights and $\mu_\ell = \hat{\mu}_\ell$ and $\mu_\ell^{HWA} = \mu_\ell$.
**for** $k = 0, 1, ...$ **do**
  $\gamma \leftarrow \gamma(k)$ (Cyclical LR for the iteration)
  $\mu_\ell^{k+1} \leftarrow \mu_\ell^k - \gamma \nabla \mathcal{L}(\mu_\ell^k)$ (regular SVI update)
  **if** $\text{mod}(k, c) = 0$ **then**
    $n_{\text{models}} \leftarrow k/c$ (Number of models to average)
    $\mu_\ell^{HWA} \leftarrow \frac{n_{\text{models}} \mu_\ell^{HWA} + \mu_\ell^{k+1}}{n_{\text{models}} + 1}$
    $\mu_\ell^{HWA} \leftarrow \frac{n_{\text{models}} \mu_\ell^{HWA} + \mu_\ell^{k+1}}{n_{\text{models}} + 1}$
  **end if**
**end for**
```

4 Numerical Experiments

5 Conclusion

REFERENCES

Joshua V. Dillon, Ian Langmore, Dustin Tran, Eugene Brevdo, Srinivas Vasudevan, Dave Moore, Brian Patton, Alex Alemi, Matthew D. Hoffman, and Rif A. Saurous. Tensorflow distributions. *CoRR*, abs/1711.10604, 2017. URL `http://arxiv.org/abs/1711.10604`.

A Appendix

You may include other additional sections here.