Detecting Adversaries Through Bayesian Approximations in Deep Learning

Thomas Pinder

Lancaster University

Objectives

The aims of this project can be broken out into four following sequential items:

- Testing the performance of Bayesian Convolutional Neural Networks (BCNNs) and compare against standard CNNs.
- Investigate the effects of adversaries using Fast Gradient Sign Method (FGSM) to perturb images [1].
- Derive uncertainty estimates from a BCNN's prediction in order to detect an adversary.
- Investigate the plausibility of using such methods in the Deep Q-Network (DQN) of a reinforcement learning agent.

Introduction

Despite state-of-the-art predictive capabilities, neural networks are incredibly brittle, and it takes only the slightest deviation away from their known manifold to fool such a classifier. These off-manifold examples are termed adversaries and are deliberately crafted to exploit the blind spots held in a neural network's structure [2].

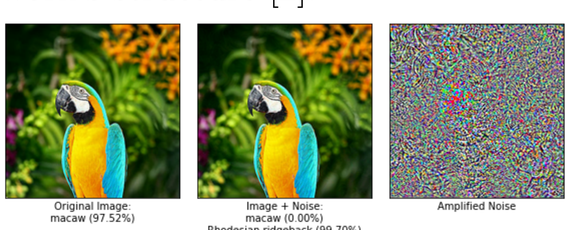

Figure 1: Adversarial Example with $\epsilon = 0.05$.

Existing attempts to defend against adversaries have included: *normalising* the input image's pixel values, stochastically pruning the network's weights and ensembling classifiers. In all cases, the defence method of choice comes at the cost of reduced accuracy, thus reducing the potency of neural networks in classification tasks. Unfortunately, the solution is not as simple as choosing a new classifier, such as SVMs or random forests, as it has been shown that adversarial examples are transferable across all classifiers.

Bayesian Neural Network Approximations

Bayesian neural networks are historically intractable to compute due to having to optimise millions of posterior distributions. Gal and Ghahramani were able to show that by introducing dropout layers into every layer of a neural network and keeping the dropout on at test time, the entire network converged to a deep Gaussian process [3].

By sampling from a trained network with dropout, we can consider each prediction through the network as a stochastic sample. Multiple samples are then equivalent to sampling from the network's predictive posterior distribution. The final classification of the network is the predictive mean of all our samples, which can be calculated as

$$\mathbb{E}(\boldsymbol{y}^*) = \frac{1}{T}\Sigma_{t=1}^{T}\hat{\boldsymbol{y}}_t^* \qquad (1)$$

Sampling from the predictive posterior of a deep Gaussian process allows us to leverage properties of a Gaussian process such as uncertainty. Similar to the predictive mean, we can get this uncertainty through the network's second moment, the sample variance, plus the model's precision. Figure 2 shows the advantage of these uncertainty estimates over softmax *uncertainties*.

Key Result

Uncertainty estimates enable adversarial detection with an adversarial example's uncertainty being greater than an unperturbed image's 75-96% of the time.

Fast Gradient Sign Method

FGSM creates adversarial examples in a fast and effective way using

$$\boldsymbol{\eta} = \epsilon \ \text{sign}(\Delta_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y)). \qquad (2)$$

$\boldsymbol{\eta}$ is then added to the original image's pixel values to craft an adversary, as in Figure 1.

Uncertainty vs. Softmax

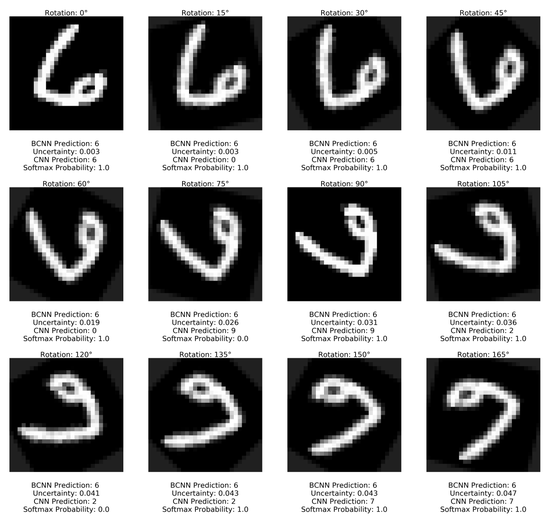

Figure 2: Uncertainty estimates as an MNIST digit is rotated

Experimental Steps

The following steps were done for the MNIST and Chest X-Ray (Figure 3) datasets:

- Construct an adversary using FGSM.
- Test the accuracy when adversaries are present at ϵ values $0.1, 0.2, \ldots 0.9$.
- Extract uncertainties from the BCNN pre and post adversarial perturbation.
- Use BEST, a Bayesian t-test equivalent, to test for a difference in uncertainty values [4].

Results

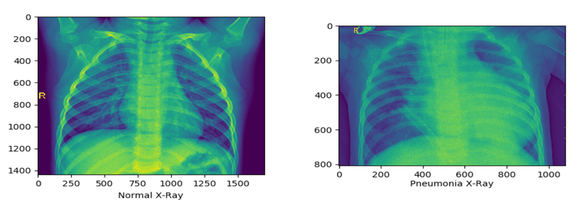

Figure 3: Chest X-Ray Dataset

Adversaries crafted using FGSM ($\epsilon = 0.05$) reduced a BCNN's accuracy from 93.2% to 25%, a value less than random guessing.

Results Continued

Through a Metropolis-Hastings MCMC, we can derive posteriors for the uncertainty estimates made on adversarial examples and their unperturbed counterparts.

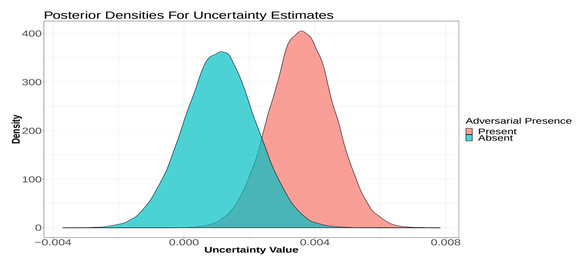

Figure 4: Posterior distributions of uncertainty estimates.

For the chest X-Ray dataset, there is a 0.75 probability that an adversarial example's uncertainty estimate will be larger than that of unperturbed images. Similarly, the probability is 0.96 for the MNIST dataset.

Concerning baseline accuracies, BCNNs had higher accuracy, recall and precision metrics than a standard CNN; however, their performance declined more severely in the presence of adversaries.

Conclusions & Further Work

- Adversaries are devastating, however, through uncertainty estimation, we have a way to detect them.
- Training, a DQN with a BCNN, is a viable option and should be investigated further to detect adversaries in reinforcement learning problems.
- Further work should be done to test the suitability of BCNN on more intelligent adversaries.

References

[1] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy.
Explaining and Harnessing Adversarial Examples.
12 2014.

[2] Szegedy, Christian and Zaremba, Wojciech and Sutskever, Ilya and Bruna, Joan and Erhan, Dumitru and Goodfellow, Ian and Fergus, Rob.
Intriguing properties of neural networks.
arXiv preprint arXiv:1312.6199, 2013.

[3] Yarin Gal and Zoubin Ghahramani.
Dropout as a bayesian approximation: Representing model uncertainty in deep learning.

[4] John K. Kruschke.
Bayesian estimation supersedes the t test.
Journal of Experimental Psychology: General, 142(2):573–603, 5 2013.