Understanding Measures of Uncertainty for Adversarial Example Detection

Lewis Smith
Department of Engineering Science
University of Oxford
Oxford, United Kingdom

Yarin Gal
Department of Computer Science
University of Oxford
Oxford, United Kingdom

Abstract

Measuring uncertainty is a promising technique for detecting adversarial examples, crafted inputs on which the model predicts an incorrect class with high confidence. There are various measures of uncertainty, including predictive entropy and mutual information, each capturing distinct types of uncertainty. We study these measures, and shed light on why mutual information seems to be effective at the task of adversarial example detection. We highlight failure modes for MC dropout, a widely used approach for estimating uncertainty in deep models. This leads to an improved understanding of the drawbacks of current methods, and a proposal to improve the quality of uncertainty estimates using probabilistic model ensembles. We give illustrative experiments using MNIST to demonstrate the intuition underlying the different measures of uncertainty, as well as experiments on a real-world Kaggle dogs vs cats classification dataset.

1 INTRODUCTION

Deep neural networks are state of the art models for representing complex, high dimensional data such as natural images. However, neural networks are not robust: there exist small perturbations to the input of the network which produce erroneous and over-confident classification results. These perturbed inputs, known as adversarial examples (Szegedy et al., 2013), are a major hurdle for the use of deep networks in safety-critical applications, or those for which security is a concern.

One possible hypothesis for the existence of adversarial examples is that such images lie off the manifold of natural images, occupying regions where the model makes unconstrained extrapolations. If this hypothesis were to hold true, then one could detect adversarial perturbation by measuring the distance of the perturbed input to the image manifold.

Hypothetically, such distances could be measured using nearest neighbour approaches, or by assessing the probability of the input under a density model on image space. However, approaches based on geometric distance are a suboptimal choice for images, as pixel-wise distance is a poor metric for perceptual similarity; similarly, density modelling is difficult to scale to the high dimensional spaces found in image recognition.

Instead, we may consider proxies to the distance from the image manifold. For example, the model uncertainty of a *discriminative* Bayesian classification model should

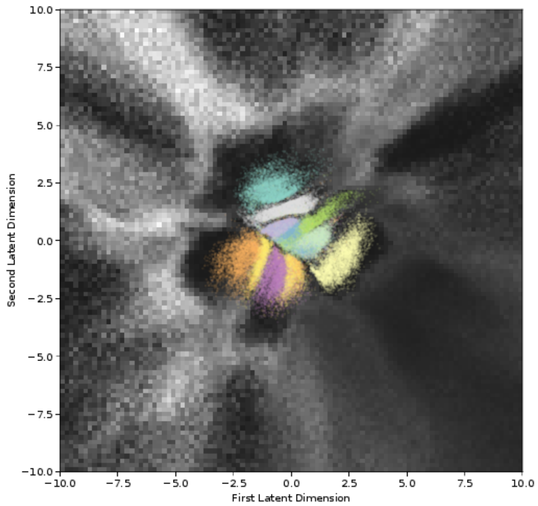

Figure 1: Uncertainty of a standard dropout network trained on MNIST, as measured by *mutual information*, visualized in the latent space obtained from a variational autoencoder. Colours are classes for each encoded training image. The background shows uncertainty, calculated by decoding each latent point into image space, and evaluating the mutual information between the decoded image and the model parameters. A lighter background corresponds to higher uncertainty.

be high for points far away from the training data, for a high-capacity model such as a deep network. Under the hypothesis that adversarial examples lie far from the image manifold, i.e. the training data, such uncertainty could be used to identify an input as adversarial.

The uncertainty of such models is not straightforward to obtain. Numerical methods for integrating the posterior, such as Markov Chain Monte Carlo, are difficult to scale to large datasets (Gal, 2016). As a result, approximations have been studied extensively. For example, approximate inference in Bayesian neural networks using dropout is a computationally tractable technique (Gal & Ghahramani, 2016) which has been widely used in the literature (Leibig et al., 2017; Gal, 2016). Dropout based model uncertainty can be used for the detection of adversarial examples, with moderate success (Li & Gal, 2017; Feinman et al., 2017; Rawat et al., 2017).

However, existing research has mostly overlooked the effect of the chosen *measure for uncertainty quantification*. Many such measures exist, including mutual information, predictive entropy and softmax variance. (Li & Gal, 2017) for example use expected entropy, (Rawat et al., 2017) use mutual information, whereas (Feinman et al., 2017) estimate the variance of multiple draws from the predictive distribution (obtained using dropout). Further, to date, research for the identification of adversarial examples using model uncertainty has concentrated on toy problems such as MNIST, and has not been shown to extend to more realistic data distributions and larger models such as ResNet (He et al., 2015).

In this paper we examine the differences between the various measures of uncertainty used for adversarial example detection, and in the process provide further evidence for the hypothesis that model uncertainty could be used to identify an input as adversarial. More specifically, we illustrate the differences between the measures by projecting the uncertainty onto lower dimensional spaces (see for example Fig. 1). We show that the softmax variance can be seen as an approximation to the mutual information (section 3.2), explaining the effectiveness of this rather ad-hoc technique. We show that some measures of uncertainty do not distinguish between non-adversarial off-manifold images (for example image interpolations) and adversarial inputs. We highlight ways in which dropout fails to capture the full Bayesian uncertainty by visualizing gaps in model uncertainty in the latent space (Section 4.2), and use this insight to propose a simple extension to dropout schemes to be studied in future research. We finish by demonstrating the effectiveness of dropout on the real-world ASIRRA (Elson et al., 2007) cats and dogs classification dataset (Section 4.3). Code for the experiments described in this paper is available online[1].

2 BACKGROUND

2.1 BAYESIAN DEEP LEARNING

A deep neural network (with a given architecture) defines a function $f : \mathcal{X} \mapsto \mathcal{Y}$ parametrised by a set of weights and biases $\omega = \{\mathbf{W}_l, \mathbf{b}_l\}_{l=1}^{L}$. These parameters are generally chosen to minimize some loss function $E : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbb{R}$ on the model outputs and the target outputs over some dataset $\mathcal{D} = \{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^{N}$ with $\mathbf{x} \in \mathcal{X}$ and $\mathbf{y} \in \mathcal{Y}$. Since neural networks are highly flexible models with many degrees of freedom, a regulariser is often added to the loss, giving

$$\hat{\omega} = \arg\min_{\omega} \sum_i E(f(\mathbf{x}_i; \omega), y) + \lambda \sum_l ||\mathbf{W}_l||^2 \quad (1)$$

for the common choice of an L_2 regulariser with weight decay λ.

In Bayesian approaches, rather than thinking of the weights as fixed parameters that are optimised over, we treat them as random variables, and so we place a prior distribution $p(\omega)$ over the weights of the network. If we also have a likelihood function $p(y \mid \mathbf{x}, \omega)$ that gives the probability of $\mathbf{y} \in \mathcal{Y}$ given the model parameters and an input to the network, we can conduct inference given a dataset by marginalizing the parameters. Such models are known as *Bayesian neural networks*.

If the prior on the weights is a zero mean Gaussian with diagonal covariance, and the loss of the network is the negative log likelihood (so $p(\mathbf{y} \mid \omega, \mathbf{x}) = e^{-E(f(\mathbf{x}),\mathbf{y})}$) then the optimised solution in equation 1 corresponds to a mode of the posterior over the weights.

Ideally we would integrate out our uncertainty by taking the expectation of the predictions over the posterior, rather than using this point estimate. For neural networks this can only be done approximately. Here we discuss one practical approximation, variational inference with dropout approximating distributions.

2.2 VARIATIONAL INFERENCE

Variational inference is a general technique for approximating complex probability distributions. The idea is to approximate the intractable posterior $p(\omega \mid \mathcal{D})$ with a simpler approximating distribution $q_\theta(\omega)$. By applying Jensen's inequality to the Kullback-Leibler divergence between the approximating distribution and the true pos-

[1] https://github.com/lsgos/uncertainty-adversarial-paper

terior, we obtain the log-evidence lower bound $\mathcal{L}_{VI}$

$$\mathcal{L}_{VI} := \int q_\theta(\omega) \log p(\mathcal{D} \mid \omega) d\omega - D_{KL}(q_\theta \mid\mid p(\omega)).$$

Since the model evidence is a constant independent of the parameters of q_θ, maximizing $\mathcal{L}_{VI}$ with respect to θ will minimize the KL divergence between q and the model posterior. The key advantage of this from a computational perspective is that we replace an integration problem with an optimisation problem, maximising a parametrised function, which can be approached by standard gradient based techniques.

For neural networks, a common approximating distribution is *dropout* (Srivastava et al., 2014) and it's variants. In the variational framework, this means the weights are drawn from

$$\begin{aligned} &\mathbf{W}_l = \mathbf{M}_l \cdot \text{diag}([\mathbf{z}_{l,j}]_{j=1}^{K_l}) \\ &\quad \text{where } \mathbf{z}_{l,j} \sim \text{Bernoulli}(p_l),\ l = 1..L, j = 1..K_{l-1} \end{aligned}$$

for a network with L layers, where the dimension of each layer is $K_i \times K_{i-1}$, and the parameters of q are $\theta = \{\mathbf{M}_l, p_l \mid l = [1..L]\}$. Informally, this corresponds to randomly setting the outputs of units in the network to zero (or zeroing the rows of the fixed matrix $\mathbf{M}_l$). Often the layer dropout probabilities p_i are chosen as constant and not varied as part of the variational framework, but it is possible to learn these parameters as well (Gal et al., 2017). Using variational inference, the expectation over the posterior can be evaluated by replacing the true posterior with the approximating distribution. The dropout distribution is still challenging to marginalise, but it is readily sampled from, so expectations can be approximated using the Monte Carlo estimator

$$\begin{aligned} \mathbb{E}_{p(\omega|\mathcal{D})}[f^\omega(x)] &= \int p(\omega|\mathcal{D}) f^\omega(x) \mathrm{d}\omega \\ &\simeq \int q_\theta(\omega) f^\omega(x) \mathrm{d}\omega \\ &\simeq \frac{1}{T} \sum_{i=1}^{T} f^{\omega_i}(x),\ \omega_{1..T} \sim q_\theta(\omega). \quad (2) \end{aligned}$$

2.3 ADVERSARIAL EXAMPLES

Works by (Szegedy et al., 2013) and others, demonstrating that state-of-the-art deep image classifiers can be fooled by small perturbations to input images, have initiated a great deal of interest in both understanding the reasons for why such adversarial examples occur, and devising methods to resist and detect adversarial attacks. So far, attacking has proven more successful than defence; a recent survey of detection methods by (Carlini & Wagner, 2017a) found that, with the partial exception of the method based on dropout uncertainty analysed by (Feinman et al., 2017), all other investigated methods could be defeated straightforwardly.

There is no precise definition of when an example qualifies as 'adversarial'. The most common definition used is of an input $\mathbf{x}_{adv}$ which is close to a real data point $\mathbf{x}$ as measured by some L_p norm, but is classified wrongly by the network with high score. Speaking more loosely, an adversarially perturbed input is one which a human observer would assign a certain class, but for which the network would predict a different class with a high score.

It is notable that there exists a second, related, type of images which have troubling implications for the robustness of deep models, namely meaningless images which are nevertheless classified confidently as belonging to a particular class (see, for example, Nguyen et al. (2015)). That such images can be found reveals another shortcoming of neural networks from the point of view of uncertainty, since they are far from all training data by any reasonable metric (based on either pixel-wise or perceptual distance). We refer to these as 'rubbish class examples' or 'fooling images' following (Nguyen et al., 2015) and (Goodfellow et al., 2014).

Several possible explanations for the existence of adversarial examples have been proposed in the literature (Akhtar & Mian, 2018). These include the idea, proposed in the original paper by (Szegedy et al., 2013), that the set of adversarial examples are a dense, low probability set like the rational numbers on $\mathbb{R}$, with the discontinuous boundary somehow due to the strong non-linearity of neural networks. Contrary to that, (Goodfellow et al., 2014) proposed that adversarial examples are partially due of the intrinsically linear response of neural network layers to their inputs. (Tanay & Griffin, 2016) have proposed that adversarial examples are possible when the decision boundaries are strongly tilted with respect to the vector separating the means of the class clusters.

Many of these ideas are consistent with the idea that the training data of the model lies on a low dimensional manifold in image space, the hypothesis we build upon in this paper.

2.4 MEASURES OF UNCERTAINTY

There are two major sources of uncertainty a model may have:

1. *epistemic* uncertainty is uncertainty due to our lack of knowledge; we are uncertain because we lack understanding. In terms of machine learning, this corresponds to a situation where our model parameters are poorly determined due to a lack of data, so

our posterior over parameters is broad.

2. *aleatoric* uncertainty is due to genuine stochasticity in the data. In this situation, an uncertain prediction is the best possible prediction. This corresponds to *noisy* data; no matter how much data the model has seen, if there is inherent noise then the best prediction possible may be a high entropy one (for example, if we train a model to predict coin flips, the best prediction is the max-entropy distribution $P(\text{heads}) = P(\text{tails})$).

In the classification setting, where the output of a model is a conditional probability distribution $P(y|x)$ over some discrete set of outcomes $\mathcal{Y}$, one straight-forward measure of uncertainty is the entropy of the predictive distribution

$$H[P(y|x)] = -\sum_{y\in\mathcal{Y}} P(y|x)\log P(y|x). \tag{3}$$

However, the predictive entropy is not an entirely satisfactory measure of uncertainty, since it does not distinguish between epistemic and aleatoric uncertainties. However, it may be of interest to do so; in particular, we want to capture when an input lies in a region of data space where the model is poorly constrained, and distinguish this from inputs near the data distribution with noisy labels.

An attractive measure of uncertainty able to distinguish epistemic from aleatoric examples is the information gain between the model parameters and the data. Recall that the mutual information (MI) between two random variables X and Y is given by

$$\begin{aligned} I(X,Y) &= H[P(X)] - \mathbb{E}_{P(y)}H[P(X \mid Y)] \\ &= H[P(Y)] - \mathbb{E}_{P(x)}H[P(Y \mid X)]. \end{aligned}$$

The amount of information we would gain about the model parameters if we were to receive a label y for a new point x, given the dataset $\mathcal{D}$ is then given by

$$I(\omega, y \mid \mathcal{D}, x) = H[p(y \mid x, \mathcal{D})] - \mathbb{E}_{p(\omega|\mathcal{D})}H[p(y \mid x, \omega)] \tag{4}$$

Being uncertain about an input point x implies that if we knew the label at that point we would gain information. Conversely, if the parameters at a point are already well determined, then we would gain little information from obtaining the label. Thus, the MI is a measurement of the model's *epistemic* uncertainty.

In the form presented above, it is also readily approximated using the Bayesian interpretation of dropout. The first term we will refer to as the 'predictive entropy'; this is just the entropy of the predictive distribution, which we have already discussed. The second term is the mean of the entropy of the predictions given the parameters over the posterior distribution $p(\omega \mid \mathcal{D})$, and we thus refer to it as the expected entropy.

These quantities are not tractable analytically for deep nets, but using dropout inference and equation (2), the predictive distribution, entropy and the MI are readily approximated; (Gal, 2016):

$$\begin{aligned} p(y \mid \mathcal{D}, \mathbf{x}) &\simeq \frac{1}{T}\sum_{i=1}^{T} p(y \mid \omega_i, \mathbf{x}) \qquad (5) \\ &:= p_{MC}(y \mid \mathbf{x}) \\ H[p(y \mid \mathcal{D}, \mathbf{x})] &\simeq H[p_{MC}(y \mid \mathcal{D}, \mathbf{x})] \qquad (6) \\ I(\omega, y \mid \mathcal{D}, x) &\simeq H[p_{MC}(y \mid \mathcal{D}, \mathbf{x})] \qquad (7) \\ &\quad - \frac{1}{T}\sum_{i=1}^{T} H[p(y \mid \omega_i, \mathbf{x})] \qquad (8) \end{aligned}$$

where $\omega_i \sim q(\omega \mid \mathcal{D})$ are samples from the dropout distribution.

Other, measures of uncertainty include the empirical variance of the softmax probabilities $p(y = c \mid \omega_i, \mathbf{x})$ (with the variance calculated over i), and variation ratios (Gal, 2016), with the former commonly used in previous research on adversarial examples.

3 UNCERTAINTY FOR ADVERSARIAL EXAMPLE DETECTION

We start by explaining the type of uncertainty relevant for adversarial example detection under the hypothesis that adversarial images lie off the manifold of natural images, occupying regions where the model makes unconstrained extrapolations. We continue by relating the *softmax variance* measure of uncertainty to mutual information.

3.1 WHAT KIND OF UNCERTAINTY?

Both the MI and predictive entropy should increase on inputs which lie far from the image manifold. Under our hypothesis, we expect both to be effective in highlighting such inputs. However, predictive entropy could also be high *near* the image manifold, on inputs which have inherent ambiguity. Such inputs could be ambiguous images, such as an image that contains both a cat and a dog, or more generally interpolations between classes, such as a digit that could be either a 1 or a 7. Such inputs would have high predictive probability for more than one class even in the limit of infinite data, yielding high predictive entropy (but low MI). Such inputs are clearly not adversarial, but would falsely trigger a hypothetical automatic detection system[2]. We demonstrate this experimentally in the next section.

Algorithms to find adversarial examples seek to create an example image with a different class to the original, typically by either minimising the predicted probability of

[2] We speculate that previous research using predictive entropy has not encountered this phenomenon due to insufficient coverage of the test cases.

the current class for an untargeted attack, or maximising the predicted probability of a target class. This has the side-effect of minimising the entropy of the predictions, a simple consequence of the normalisation of the probability. It is interesting to highlight that this also affects the uncertainty as measured by MI; since both the mutual information and entropy are strictly positive, the mutual information is bounded above by the predictive entropy (see equation 4). Therefore, the model giving low entropy predictions at a point is a sufficient condition for the mutual information to be low as well. Equally, the mutual information bounds the entropy from below; it is not possible for a model to give low entropy predictions when the MI is high. It is important to realise that this means that adversarial example algorithms implicitly seek low uncertainty examples: detecting adversarial examples, at least via model uncertainty, is *not* independent of being able to fool the model without explicit detection methods.

3.2 MI AND SOFTMAX VARIANCE

Some works in the literature estimate the epistemic uncertainty of a dropout model using the estimated variance of the MC samples, rather than the mutual information (Leibig et al., 2017; Feinman et al., 2017; Carlini & Wagner, 2017a). This is somewhat arbitrary for classification, but seems to work fairly well in practice. We suggest a possible explanation of the effectiveness of this measure, arguing that the softmax variance can be seen as a proxy to the mutual information.

One way to see the relation between the two measures of uncertainty is to observe that the variance is the leading term in the series expansion of the mutual information. For brevity, we denote the sampled distributions $p(y \mid \omega_i, \mathbf{x})$ as p_i and the mean predictive distribution $p_{MC}(y \mid \mathbf{x})$ as $\hat{p}$. These are in general distribution over C classes, and we denote the probability of the j^{th} class as $\hat{p}_j$ and p_{ij} for the mean and i^{th} sampled distribution respectively. The variance score is the mean variance across the classes

$$\begin{aligned}\hat{\sigma}^2 &= \frac{1}{C}\sum_{j=1}^{C}\frac{1}{T}\sum_{i=1}^{T}(p_{ij}-\hat{p}_j)^2 \qquad (9)\\ &= \frac{1}{C}\left(\sum_{j=1}^{C}\left(\frac{1}{T}\sum_{i=1}^{T}p_{ij}^2\right)-\hat{p}_j^2\right)\end{aligned}$$

And the mutual information score is

$$\begin{aligned}\hat{I} &= H(\hat{p}) - \frac{1}{T}\sum_i H(p_i)\\ &= \sum_j\left(\frac{1}{T}\sum_i p_{ij}\log p_{ij}\right) - \hat{p}_j\log\hat{p}_j\end{aligned}$$

Using a Taylor expansion of the logarithm,

$$\begin{aligned}\hat{I} &= \sum_j\left(\frac{1}{T}\sum_i p_{ij}(p_{ij}-1)\right) - \hat{p}_j(\hat{p}_j-1) + \ldots\\ &= \sum_j\left(\frac{1}{T}\sum_i p_{ij}^2\right) - \hat{p}_j^2 - \left(\frac{1}{T}\sum_i p_{ij}\right) + \hat{p}_j + \ldots\\ &= \sum_j^C\left(\frac{1}{T}\sum_i^T p_{ij}^2\right) - \hat{p}_j^2 + \ldots \qquad (10)\end{aligned}$$

we see that the first term in the series is identical up to a multiplicative constant to the mean variance of the samples.

This relation between the softmax variance and the mutual information measure could explain the effectiveness of the variance in detecting adversarial examples encountered by (Feinman et al., 2017). MI increases on images far from the image manifold and not on image interpolations (on which the predictive variance increases as well), with the variance following similar trends.

4 EMPIRICAL EVALUATION

In the next section we demonstrate the effectiveness of various measures of uncertainty as proxies to distance from the image manifold. We demonstrate the difference in behaviour between the predictive entropy and mutual information on image interpolations, for interpolations in the latent space as well as interpolations in image space. We continue by visualising the various measures of uncertainty, highlighting the differences discussed above. This is further developed by highlighting shortcomings with current approaches for uncertainty estimation, to which we suggest initial ideas on how to overcome and suggest new ideas for attacks (to be explored further in future research). We finish by assessing the ideas discussed in this paper on a real world dataset of cats vs dogs image classification.

4.1 UNCERTAINTY ON INTERPOLATIONS

We start by assessing the behaviour of the measures of uncertainty on image interpolations, comparing interpolations via convex combination ($\lambda x_1 + (1-\lambda)x_2, \lambda \in [0,1], x_i \in \mathcal{D}$) in latent space to those in image space. A convex combination in image space will clearly produce off manifold images, while we assume that moving in latent space approximates the manifold of the data fairly closely. That model uncertainty can capture what we want in practice is demonstrated in Figures 2 and 3. We see that the MI distinguishes between these on-manifold and off-manifold images, whereas the entropy fails to do so. This is necessary for the hypothesis proposed in the introduction; if we are able to accurately capture the MI,

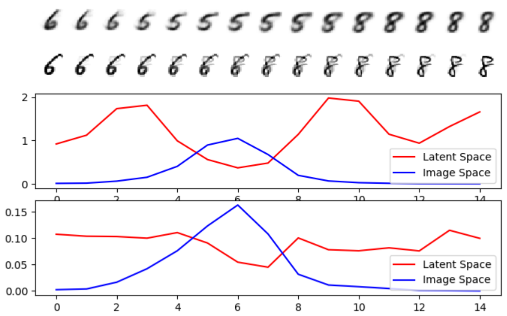

Figure 2: Entropy (middle) and the MI (bottom) vary along a convex interpolation between two images in latent space and image space (top). The entropy is high for regions along both interpolations, wherever the class of the image is ambiguous. In contrast, the MI is roughly constant along the interpolation in latent space, since these images have aleatoric uncertainty (they are ambiguous), but the model has seen data that resembles them. On the other hand, the MI has a clear peak as the pixel space interpolation produces out-of-sample images between the classes

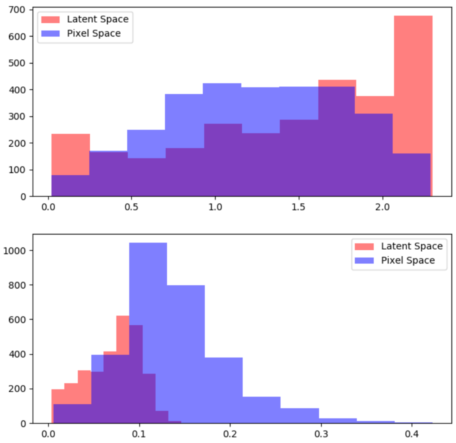

Figure 3: The entropy (top) and mutual information (bottom) of the interpolation halfway between 3000 random points of different classes in the MNIST test set, showing that the two modes of interpolation have very different statistical properties with respect to the model uncertainty, as shown for a single example in figure 2.

it would serve well as a proxy for whether an images belongs to the learned manifold or not.

4.2 VISUALIZATION IN LATENT SPACE

We wish to gain intuition into how the different measures of uncertainty behave. In order to do so, we use a variational autoencoder (Kingma & Welling, 2013) to compress the MNIST latent space. We choose a latent space of two dimensions so we can use this to visualise the dataset. By decoding the image that corresponds to a point in latent space, we can classify the decoded image and evaluate the network uncertainty, thus providing a two dimensional map of the input space. Figure 1 shows the latent space with the uncertainty measured using the MI, calculated using dropout. Similarly, Figure 4 shows the predictive entropy. Note the differences in uncertainty near the class cluster boundaries (corresponding to image interpolations) – the MI has low uncertainty in these regions, whereas the predictive entropy is high.

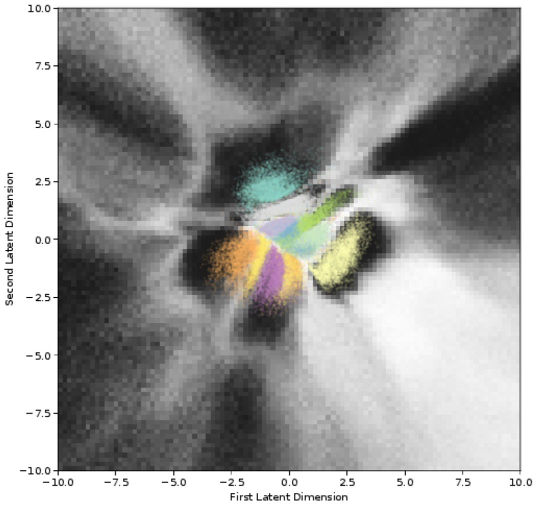

Figure 4: The predictive entropy of the same network as in figure 1. Note the differences with the MI, which is low everywhere close to the data in the centre of the plot, but the entropy is high between the classes here. These points correspond to images which resemble digits, but which are inherently ambiguous. Note however that there are large regions of latent space where the predictive entropy is high and the MI low, despite being far from any training data.

Another question of interest in this context is how well the dropout approximation captures uncertainty. The approximating distribution is fairly crude, and variational inference schemes are known to underestimate the uncertainty of the posterior, tending to fit an approximation to a local mode rather than capturing the full posterior[3].

As seen from the figures, the network does a reasonable job of capturing uncertainty close to the data. However, the network's uncertainty has 'holes'– regions where the predictions of the model are very confident, despite the images generated by the decoder here being essentially nonsense (see Figure 5). This suggests that, while the uncertainty estimates generated by MC dropout are useful,

[3] There are two reasons for this behaviour: firstly, that the approximating distribution q may not have sufficient capacity to represent the full posterior, and secondly, the asymmetry of the KL divergence, which penalizes q placing probability mass where the support of p is small far more heavily than the reverse.

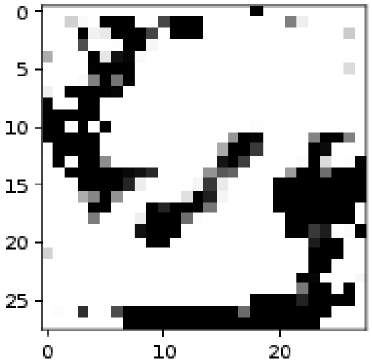

Figure 5: A typical garbage class example from the 'holes' in latent space. This is classified as a 2 with high confidence.

they do not capture the full posterior, instead capturing local behaviour near one of its modes, since we would expect the uncertainty to be high for a neural network everywhere where it is not constrained by the training data due to the high capacity of the model.

This may offer an explanation as to why MC dropout nets are still vulnerable to adversarial attack; despite their treatment of uncertainty, there are still large regions where they are mistakenly overconfident due to the approximations used, which adversarial attack algorithms can exploit. It may be possible to deliberately find and exploit these 'holes' to create adversarial examples. This intuition suggests a simple fix; since a single dropout model averages over a single mode of the posterior, we can capture the posterior using an ensemble of dropout models using different initializations, assuming that these will converge to different local modes. We find that even a small ensemble can qualitatively improve this behaviour (Figure 6).

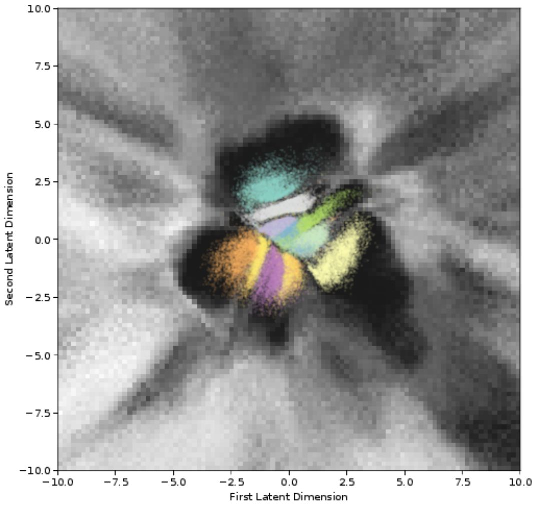

Figure 6: The MI calculated using an ensemble of dropout models, treating all of their predictions as Monte Carlo samples from the posterior. This mitigates some of the spuriously confident regions in latent space

It should be noted, though, that there is no guarantee that an ensemble of dropout models is a better approximation to the true posterior. It will approximate it well only if the posterior is concentrated in many local modes, all of roughly equal likelihood (since all the models in the ensemble are weighted equally), and a randomly initialized variational dropout net trained with some variant of gradient descent will converge to all of these modes with roughly equal probability[4]. Investigating possible theoretical justification for this ensembling procedure for variational models is a possible direction for future research.

4.3 EVALUATION ON CATS AND DOGS DATASET

It has been observed by (Carlini & Wagner, 2017a) that many proposed defences against adversarial examples fail to generalize from MNIST. Therefore, we also evaluate the various uncertainty measures on a more realistic dataset; the ASSIRA cats and dogs dataset (see Figure 7 for example images). The task is to distinguish pic-

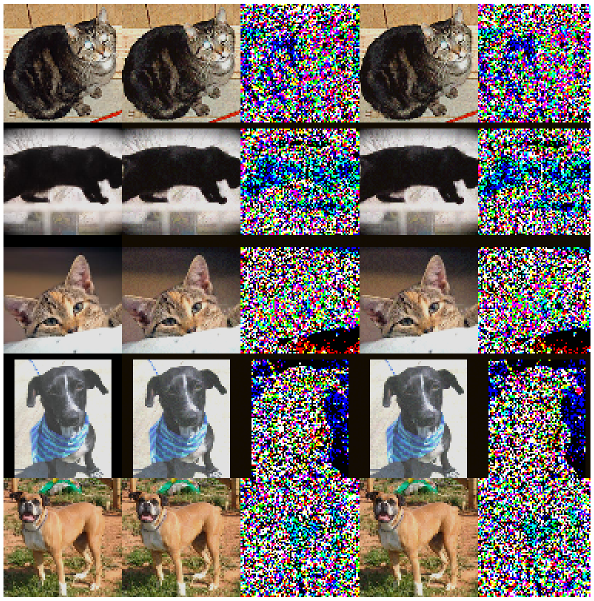

Figure 7: Example adversarial images generated by the Momentum iterative method at $\epsilon = 10$, with original images on the left, adversarial images on the deterministic model in the second column, and those for the MC dropout model in the fourth column. The difference between the adversarial image and the original is shown on the right of each image.

[4] This description does coincide with common beliefs about neural network loss surfaces, for which there is some justification in the literature; see, for example, Choromanska et al. (2015)

Table 1: The AUC for the adversarial discrimination task described in the experiments section. Fields marked with (S) denote this quantity evaluated on a version of the dataset with unsuccessful adversarial examples (that do not change the label) removed. The success rate of each attack in changing the label is given as a measure of each attacks effectiveness on this dataset.

	entropy	MI	entropy (S)	MI (S)	Success Rate
bim $\epsilon = 5$					
Deterministic	0.322	N.A	0.293	N.A	0.757
MC	0.0712	**0.728**	0.0617	**0.733**	0.900
fgm $\epsilon = 5$					
Deterministic Model	0.439	N.A	**0.490**	N.A	0.517
MC Model	0.426	**0.557**	0.465	**0.497**	0.563
mim $\epsilon = 5$					
Deterministic Model	0.347	N.A	0.319	N.A	0.743
MC Model	0.0476	**0.657**	0.0410	**0.669**	0.917
bim $\epsilon = 10$					
Deterministic Model	0.302	N.A	0.285	N.A	0.753
MC Model	0.0686	**0.708**	0.0719	**0.723**	0.917
fgm $\epsilon = 10$					
Deterministic Model	0.502	N.A	**0.550**	N.A	0.487
MC Model	0.480	**0.529**	0.514	0.491	0.547
mim $\epsilon = 10$					
Deterministic Model	0.350	N.A	0.319	N.A	0.763
MC Model	0.0527	**0.661**	0.0442	**0.665**	0.907

tures of cats and dogs. While this is not a state of the art problem, these are realistic, high resolution images. We finetune a ResNet model (He et al., 2015), pre-trained on Imagenet, replacing the final layer with a dropout layer followed by a new fully connected layer. We use 20 forward passes for the Monte Carlo dropout estimates. We use dropout only on the layers we retrain, treating the pre-trained convolutions as deterministic.

We compare the receiver operating characteristic (ROC) of the predictive entropy of the deterministic network, the predictive entropy of the dropout network (equation 7), and the MI of the dropout network (the MI is always zero if the model is deterministic; this corresponds to the approximating distribution q being a delta function). Note that we compare with the *same set of weights* (trained with dropout) – the only difference is whether we use dropout at test time. For each measure of uncertainty we generate the ROC plot by thresholding the uncertainty at different values, using the threshold to decide whether an input is adversarial or not.

The receiver operating characteristic is evaluated on a synthetic dataset consisting of images drawn at random from the test set and images from the test set corrupted by Gaussian noise, which comprise the negative examples, as well as adversarial examples generated with the Basic Iterative Method (Kurakin et al., 2016), Fast Gradient method (Goodfellow et al., 2014), and Momentum Iterative Method (Dong et al., 2017). We test with the final attack because it is notably strong, winning the recent NIPS adversarial attack competition, and is simpler to adapt to stochastic models than the other strong attacks in the literature, such as that of Carlini and Wagner (Carlini & Wagner, 2017b).

We find that only the mutual information gets a useful AUC on adversarial examples. In fact, most other measures of uncertainty seem to be worse than random guessing; this suggests that this dataset has a lot of examples the model considers to be ambiguous (high aleatoric uncertainty), which mean that the entropy has a high false positive rate. The fact the AUC of the entropy is low suggests that the model is actually *more* confident about adversarial examples than natural ones under this measure.

An interesting quirk of this particular model is that the accuracy of using Monte Carlo estimation is lower than the point estimates, even though the uncertainty estimates are sensible. Possibly this is because the dropout probability is quite high; only a subset of the features in the later layers of a convnet are relevant to cat and dog discrimination, so this may be a relic of our transfer learning procedure; dropout does not normally have an adverse effect on the accuracy of fully trained models (Gal, 2016).

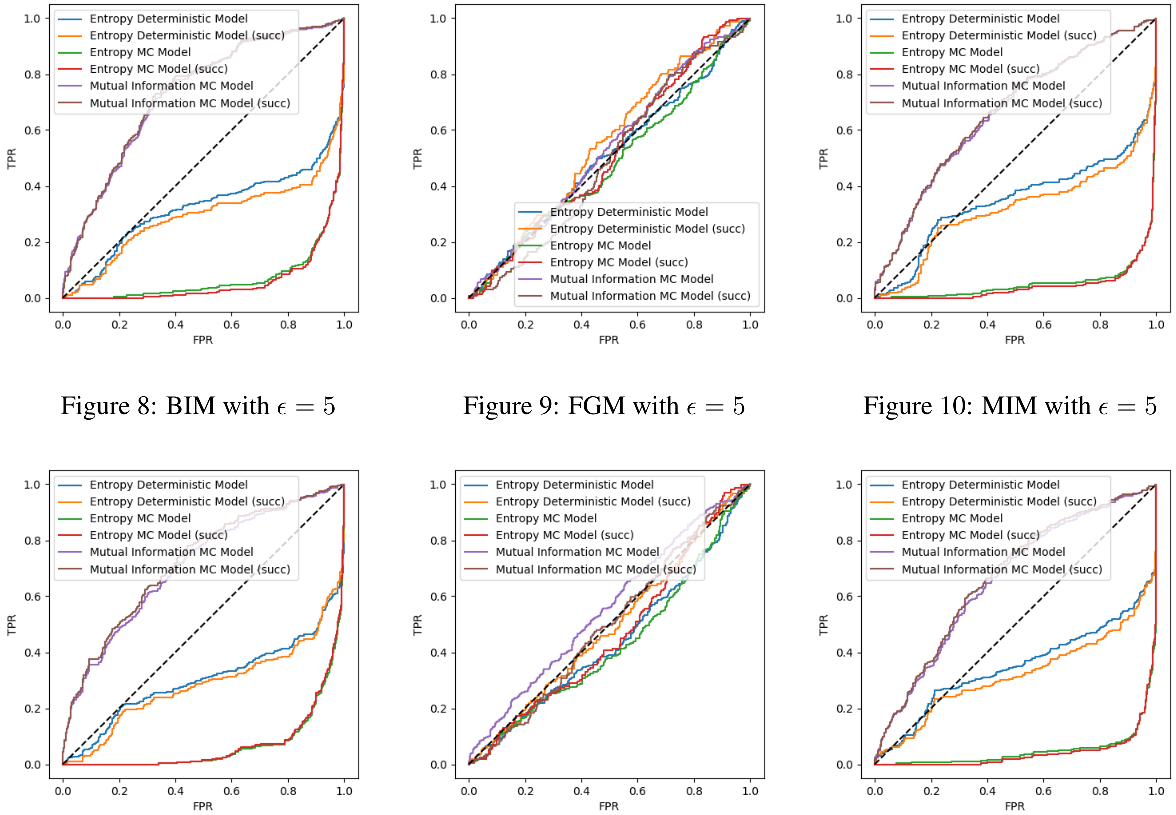

Figure 8: BIM with $\epsilon = 5$

Figure 9: FGM with $\epsilon = 5$

Figure 10: MIM with $\epsilon = 5$

Figure 11: BIM with $\epsilon = 10$

Figure 12: FGM with $\epsilon = 10$

Figure 13: MIM with $\epsilon = 10$

Figure 14: ROC plots for adversarial example detection with different measures of uncertainty and different attacks. From left to right: basic iterative method (BIM), fast gradient method (FGM), and momentum iterative method (MIM). Top row uses ϵ of 5, bottom row uses ϵ of 10. All use infinity norm. (succ) denotes the quantity evaluated only for successful adversarial examples. We suspect that the low FGM attack success rate is related to the difficulty we observe in identifying these using model uncertainty, however further investigation is required.

5 DISCUSSION & CONCLUSION

We have examined various measures of uncertainty for detecting adversarial examples, and provided both theoretical and experimental evidence that measuring the epistemic uncertainty with the mutual information is the most appropriate and effective for this task.

We do not claim, however, that using dropout provides a very convincing defence against adversarial attack. Our results (in agreement with previous literature on the subject) show that dropout networks are *more difficult* to attack than their deterministic counterparts, but attacks against them can still succeed while remaining imperceptible to the human eye, at least in the white-box setting we investigated.

It is worth noting, however, that these techniques for quantifying uncertainty can be derived without any explicit reference to the adversarial setting, and no assumptions are made about the distribution of adversarial examples. By improving model robustness and dealing with uncertainty more rigorously, models become harder to fool as a side effect; model robustness and good uncertainty estimates are not independent, as discussed in section 3. We think the fact that dropout models can still be defeated by adversarial attack is at least partly because dropout is a fairly crude approximation that underestimates the uncertainty significantly, as we have demonstrated here. Looking for scalable ways to improve on the uncertainty quality captured by dropout is an important avenue for future research.

ACKNOWLEDGEMENTS

LS is supported by EPSRC. This work was supported by the Alan Turing Institute's Defence and Security programme.

References

Akhtar, N., & Mian, A. (2018). Threat of adversarial attacks on deep learning in computer vision: A survey. *arXiv preprint arXiv:1801.00553*.

Carlini, N., & Wagner, D. (2017a). Adversarial examples are not easily detected: Bypassing ten detection methods. *arXiv preprint arXiv:1705.07263*.

Carlini, N., & Wagner, D. (2017b). Towards evaluating the robustness of neural networks. In *Security and privacy (sp), 2017 ieee symposium on* (pp. 39–57).

Choromanska, A., Henaff, M., Mathieu, M., Arous, G. B., & LeCun, Y. (2015). The loss surfaces of multilayer networks. In *Artificial intelligence and statistics* (pp. 192–204).

Dong, Y., Liao, F., Pang, T., Su, H., Hu, X., Li, J., & Zhu, J. (2017). Boosting adversarial attacks with momentum. *arXiv preprint arXiv:1710.06081*.

Elson, J., Douceur, J. J., Howell, J., & Saul, J. (2007, October). Asirra: A captcha that exploits interest-aligned manual image categorization. In *Proceedings of 14th acm conference on computer and communications security (ccs).* Association for Computing Machinery, Inc.

Feinman, R., Curtin, R. R., Shintre, S., & Gardner, A. B. (2017). Detecting adversarial samples from artifacts. *arXiv preprint arXiv:1703.00410*.

Gal, Y. (2016). *Uncertainty in deep learning* (Unpublished doctoral dissertation). PhD thesis, University of Cambridge.

Gal, Y., & Ghahramani, Z. (2016). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning* (pp. 1050–1059).

Gal, Y., Hron, J., & Kendall, A. (2017). Concrete dropout. *arXiv preprint arXiv:1705.07832*.

Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014). Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). *Deep residual learning for image recognition. corr abs/1512.03385 (2015).*

Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Kurakin, A., Goodfellow, I., & Bengio, S. (2016). Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*.

Leibig, C., Allken, V., Ayhan, M. S., Berens, P., & Wahl, S. (2017). Leveraging uncertainty information from deep neural networks for disease detection. *Scientific Reports*, *7*(1), 17816. Retrieved from `https://doi.org/10.1038/s41598-017-17876-z` doi: 10.1038/s41598-017-17876-z

Li, Y., & Gal, Y. (2017). Dropout inference in bayesian neural networks with alpha-divergences. *arXiv preprint arXiv:1703.02914*.

Nguyen, A., Yosinski, J., & Clune, J. (2015). Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 427–436).

Rawat, A., Wistuba, M., & Nicolae, M.-I. (2017). Adversarial phenomenon in the eyes of bayesian deep learning. *arXiv preprint arXiv:1711.08244*.

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *Journal of machine learning research*, *15*(1), 1929–1958.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R. (2013). Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Tanay, T., & Griffin, L. (2016). A boundary tilting persepective on the phenomenon of adversarial examples. *arXiv preprint arXiv:1608.07690*.