DISRUPTING DEEP UNCERTAINTY ESTIMATION WITHOUT HARMING ACCURACY

Nidhi Patel, Technische Universität Dortmund

Abstract
Deep neural networks (DNNs) are frequently utilized for a variety of applications and have shown to be one of the most effective predictors. However, their implementation in numerous risk-sensitive applications depends on the reliability of the uncertainty estimation of their predictions. The paper introduces an unconventional and straightforward method that, in contrast to adversarial attacks, impairs the network's ability to estimate uncertainty rather than producing inaccurate predictions. As a consequence, the DNN is more confident about its incorrect predictions than about its correct ones after an attack. Here it is important to note that the model's accuracy remains intact throughout. For the newly introduced attack, only small-scaled perturbations are needed. The paper also compares attacks under different settings. At first, the researchers target a black-box system (where they are unaware of the target network), after that, they target a white-box system with full knowledge about the network. In the study, effective attacks against three of the most widely used uncertainty estimating techniques which are the vanilla softmax score, Deep Ensembles, and MC-Dropout are illustrated with respect to different DNN models and architectures .

Keywords
Uncertainty estimation — Attack on confidence estimation (ACE) — Selective prediction — Adversarial attack

Contents

Introduction

This report is intended to summarize the research paper "Disrupting deep uncertainty estimation without harming accuracy" which introduces the concept of attacking deep neural networks (DNNs) without affecting its accuracy as an alternative of standard adversarial attack. [1]. The report is presented as a course work under the seminar "Uncertainty quantification in machine learning " at Technische Universität Dortmund. In a wide range of functional fields, such as computer vision and natural language processing, deep neural networks (DNNs) demonstrate impressive performance and are becoming better. However, the ability to accurately estimate the uncertainty estimations in these models' predictions or the use of some form of selective prediction is crucial for their successful deployment.

There are several popular uncertainty estimation techniques when it comes to classification such as: 1)Softmax score (estimates the embedding distance between a decision boundary and an instance) [2]; 2) MC-Dropout (proposed to substitute Bayesian networks) [3] ;3) Deep Ensembles , which have produced cutting-edge outcomes in a variety of estimate scenarios [4].

The study in this paper demonstrates how all of these well-known strategies for estimating uncertainty are susceptible to new types of attacks that might utterly eliminate their effectiveness. That operates in both black-box(When the attacker has no knowledge of the model itself and can simply query the attacked model for anticipated labels.) as well as White-box (the attacker thoroughly knows the model to be

assaulted) settings. Unlike standard adversarial attacks (aim to reduce model accuracy), the suggested approach is meant to maintain the performance accuracy and avoid altering the original predictions initially classified by the attacked model. The above-mentioned technique of attack is known as ACE: Attack on Confidence Estimation. In order to testify to the findings, the ACE is evaluated on different modern architectures such as MobileNetV2 , EfficientNet-B0, also on some standard baselines for instance ResNet50, DenseNet161 and VGG16.

Figure 1 demonstrates the working of ACE on EfficientNet with softmax as uncertainty estimation method. As we can see, two images on the left-hand side were predicted as tanks before the attack out of which the one in the upper left corner is correctly predicted as a tank with higher confidence of 0.9392 whereas one in the lower left corner is binoculars incorrectly predicted as a tank with a lower confidence score, 0.1. On the right-hand side after adding the perturbations to original inputs the confidence of correct prediction dropped significantly to 0.036 and the confidence score of incorrect prediction highly increased to 0.908. After the attack, the model is more confident about its incorrect prediction than the correct one.

1. Working

In this section, we will have a look into the functioning of the ACE with respect to sofmax function. Let's consider the classifier that classifies cats vs. dogs by using the softmax function as its measure of uncertainty estimation. Before going into details, familiarity with some fundamental concepts and understanding how they get affected by ACE is important.

1.1 Confidence score

In a general scenario, the confidence score ranks correct predictions higher than incorrect ones. However, the ACE causes the confidence score to rank incorrect predictions higher and correct ones lower. The concept is illustrated using figure 2 and figure 3. Figure 2 shows the histogram of EfficientNet confidence scores derived by softmax function for its correct (green) and incorrect predictions (red). Figure 3 shows the histogram of EfficientNet confidence scores after attacking with ACE

1.2 Selective prediction

Majority of Risk-sensitive applications employ uncertainty estimation mechanisms like a selective prediction. It abstains the model from predicting observation for which a confidence score is lower than a certain threshold to achieve higher accuracy[5]. In order to understand its working with respect to ACE, consider the above-mentioned example of the cat vs. dog classifier. As we can see in Figure 4, there are four instances to be classified out of which the first three are correctly classified with higher confidence whereas the last image of a dog (with glasses) is incorrectly predicted as a cat with low confidence of 0.3. As three out of four instances are predicted correctly, the model accuracy would be 75%. If the selective prediction with a 0.6 threshold value is applied to the above-mentioned model as shown in Figure 5, the selective prediction mechanism will abstain to predict the incorrect prediction(dog with glasses) as it has a confidence score (0.3) less than the threshold value 0.6. In this case, the model only predicts three correctly classified instances, therefore, achieving 100% selective accuracy. However, after the attack on the model as demonstrated in Figure 6, ACE has already lowered the confidence score of the correctly classified instance below the threshold and increased the score of incorrect ones above the threshold. That causes the Selective prediction to abstain from predicting the correct observations due to lower confidence. And only predicting incorrect predictions.

2. Comparison: ACE Vs. Adversarial attacks

We can assume a 2D plane where the distance of an instance to its decision boundary can be quantified by the softmax score.[6] That means the farther the instance is from the decision boundary the more confident the model is about the prediction of that instance. As can be seen in Figure 7, the classifier misclassified one instance of a dog as a cat. In general, the model has the lowest confidence of all classifications in predicting that instance which is rather obvious.

2.1 Standard adversarial attack

In the case of a standard adversarial attack as illustrated in Figure 8, the attacker would aim to alter the label of the correctly classified instance. In the context of softmax, that means the attacker would wish to push the correctly predicted instance across the decision boundary by adding the perturbation into the input image. In that case, the added perturbation should be large enough to cross the decision boundary. The larger the perturbation is more likely to aware the victim about the undergoing attack. Figure 9 demonstrates that even if the attacker succeeds to push the instance with higher confidence across the decision boundary, the instance would be assigned with lower confidence after the attack. Therefore there is a higher chance of it getting rejected by the selective prediction and leaving the model unharmed. Additionally, altered labels can cause a sudden and significant drop in the accuracy of the model. Therefore, it is more likely to alert a victim about the attack in case of constant monitoring.

2.2 Attacking Confidence Estimation(ACE)

However, in the case of ACE, the attacker targets the instances in which the model possesses higher confidence. The attacker does so by aiming to decrease the confidence of the correctly predicted instances and pushing them towards the decision boundary and increasing the confidence of the incorrectly classified instance by pushing them away from the decision boundary as demonstrated in Figure 10. For that reason, the required amount of perturbation would be significantly smaller

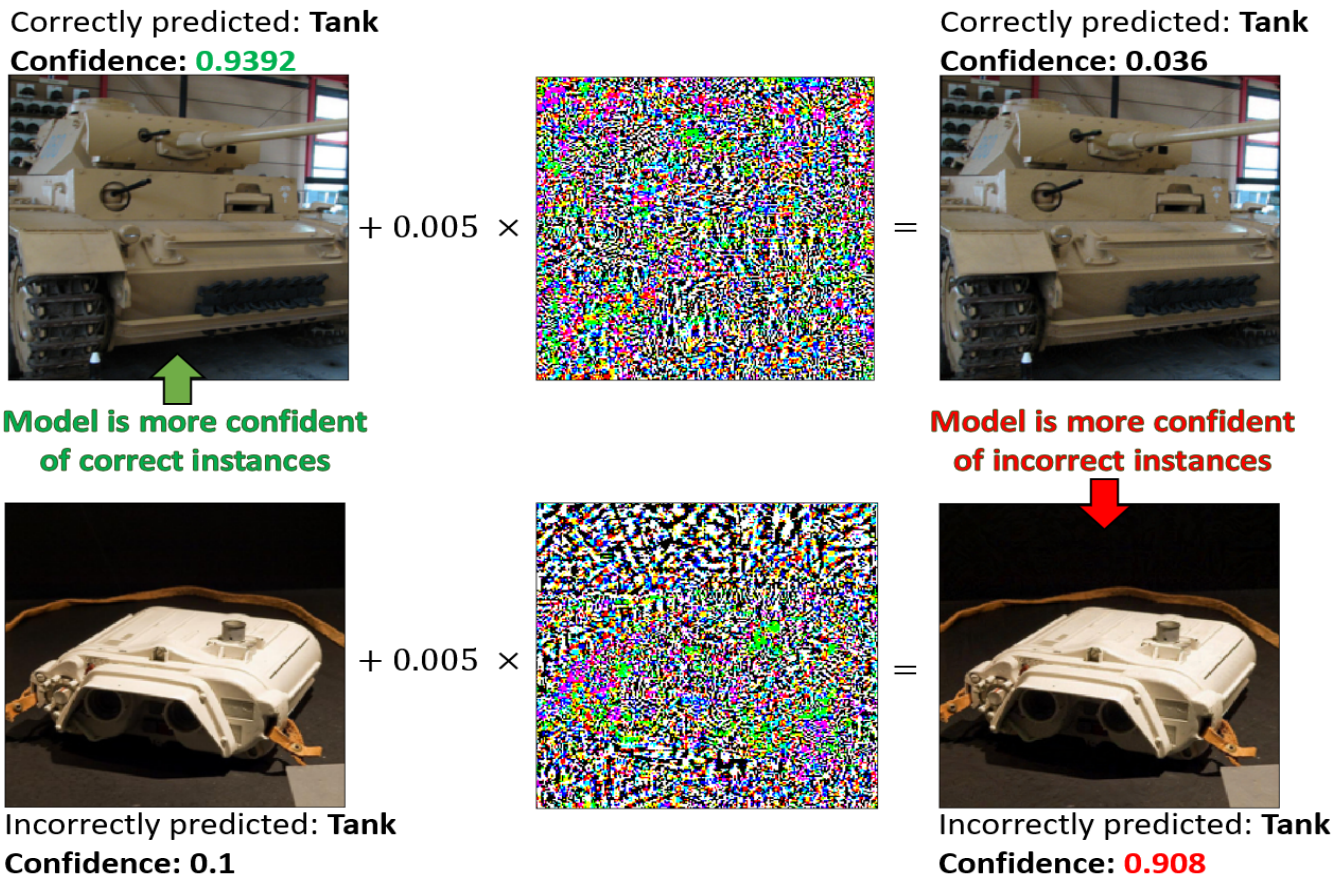

Figure 1. Attacking EfficientNet with the uncertainty estimation technique softmax.

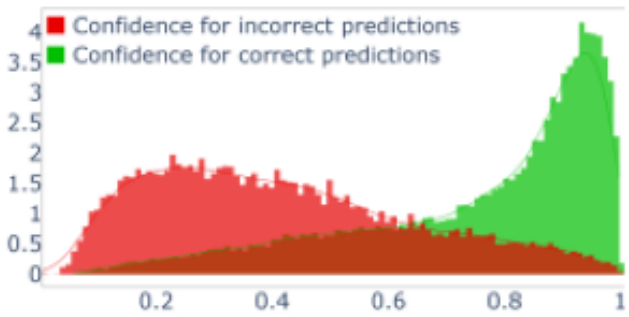

Figure 2. Confidence before attack

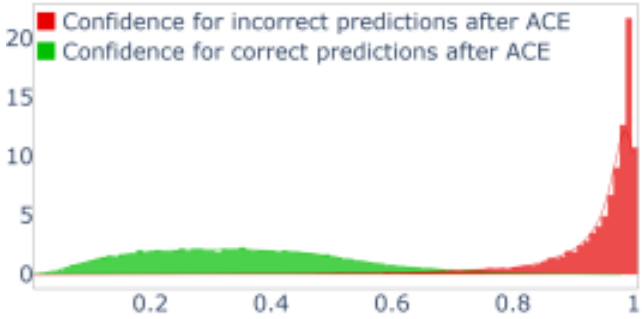

Figure 3. Confidence after ACE.

than an adversarial attack. In other words, as opposed to standard adversarial attacks, ACE successfully causes a significant amount of harm to the model even with a smaller magnitude of perturbation benefitting the attacker with limited resources.

During the process, the attacker makes sure that the instance doesn't cross the decision boundary. Unlike standard adversarial attacks, ACE only targets the model's uncertainty estimation which is less likely to alert the victim about the attack. Here, Figure 11 shows the final result after applying ACE.

2.3 Advantages of ACE over standard adversarial attack

To summarise this section, we can point out a few benefits of ACE that distinguish from other adversarial attacks:

1. Inherently, the perturbations required for this kind of attack are much lower than those required for the majority of adversarial attacks benefiting the attacker with limited resources.
2. Keeps the accuracy intact therefore less likely to alert the victim.
3. Its success isn't double-edged: Pushing an input over

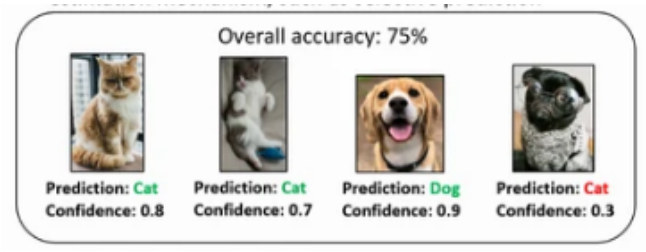

Figure 4. Classification model cat vs. dog

Figure 5. Classification model cat vs. dog with selective prediction

the decision boundary is the primary objective of most adversarial attacks. The assault is considered unsuccessful if the magnitude of perturbation is inadequate to achieve it. However, in the ACE even smaller magnitude of perturbation would succeed to create a considerable amount of harm.

3. Implemetation of ACE

This section will provide detailed insights into the implementation of ACE. In order to attack the model with ACE, the attacker first crafts the perturbation. After that, the attacker pushes the instance towards the boundary in case of a correctly predicted instance or far from the boundary in case of an incorrectly classified instance by iteratively adding the crafted perturbation to the input instance. During the process, if the instance crosses the boundary then the attacker decreases the magnitude of the perturbation by decay factor(ε_{decay}) and tries all over again. Meanwhile, if the algorithm exceeds the maximum number of iterations then the attacker stops attacking and returns the last successfully perturbed instance within the boundary. The algorithm of ACE given in Figure 12 illustrates the above-mentioned process in mathematical form.

Let X be the input space and Y be the response space. And f is the model for a prediction $f : X \rightarrow Y$, and $\hat{y}_f(x)$ is its predicted label for an input image x where $x \in X$ and $y \in Y$. For a given model f , we define a confidence score function $k(x, \hat{y}_f(x)|\hat{f})$. The function k is suppose to measure confidence in the prediction of $\hat{y}$ for the input x, based on signals from the model f. An ensemble of models that do not include the attacked model f itself serves as the proxy ($\hat{f}$) in black-box scenarios. In white-box settings an attacker doesn't need a proxy as he has all the required information about the model to be attacked, $\hat{f} = f$. η is a gradient derived by taking the derivative of k(confidence score) w.r.t. input x. Initial value of perturbation is ε. ε_{decay} is a rate of decay in the value of perturbation ε. *max_iterations* is the limit on the number of iterations.

Finding a minimum perturbation $\varepsilon_{effctive}$ for an input x, $\sim x = x + \varepsilon_{effctive}$ that would cause the confidence score function k to produce a poor partial order on its inputs without changing the model's accuracy enables one to construct an adversarial example that especially attacks uncertainty estimate.

Figure 6. Classification model cat vs. dog with selective prediction after ACE

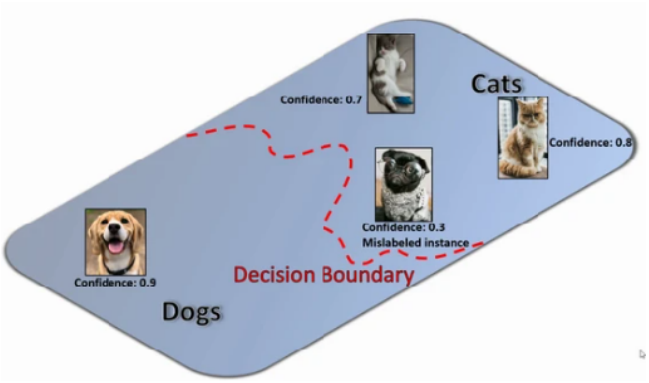

Figure 7. The visualization of a classifier of cats and dogs with the softmax as uncertainty measure

4. Effect of ACE

This section provides a short overview of the impact of ACE on different models with different architectures. It also compares the effect of attacks being held under white-box and black-box settings. A detailed interpretation of results and comparisons has been discussed in the 'Results and Discussion' section. Figure 13 illustrates the EfficientNet RC curves during a white-box assault with different magnitudes of perturbations(ε). The area under the curve (AURC) x 1000 is shown by the colored numbers adjacent to each corresponding colored curve. As we can see, for ε= 0.005, the selective accuracy will be 0% for it's top 20% of most confident predictions which means, an end user querying the model simply for its 20% most confident predictions will receive nearly 100% incorrect results for assaults applied with 0.005 magnitudes of perturbation.

4.1 Effect of ACE under white-box vs. black-box settings

Comparing the results derived from different experiments on the models with different architectures under white-box and black-box settings it can be concluded that for white-box settings, much less perturbation ($\varepsilon_{effctive}$) is required compared

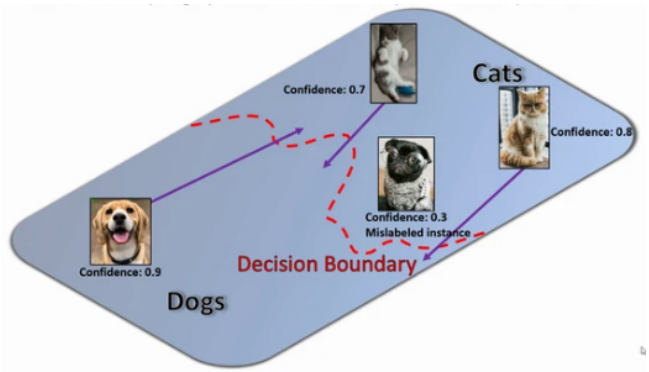

Figure 8. The visualization of standard adversarial attack on a classifier of cats and dogs with the softmax as uncertainty measure

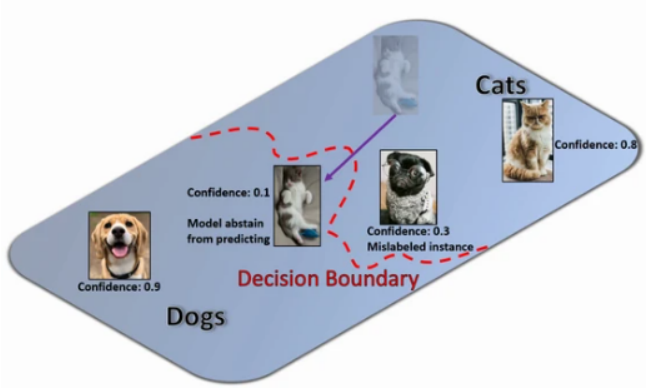

Figure 9. The visualization of standard adversarial attack after pushing correctly classified instance across the boundary.

to black-box to generate a similar amount of harm. Consider table from Figure 14 containing observations for white box and black box attacks for different values of perturbations for ResNet50. Here, $\varepsilon_{effctive}$ value is the smallest value of ε for which maximum harm is caused without instance crossing the boundary. NLL(negative log-likelihood) and brier score represents the loss we can also interpret it as a harm caused to the model in our case. The greater the value of NLL, the more harm has been caused to the model.

Comparing the values of table from Figure 14 we can see that for the same amount of perturbation applied, $\varepsilon_{effctive}$ for White-box is less than that of Black-box attack however, the values of NLL and brier score are considerably higher for White-box as compared to Black-box. Another thing to notice is that as the value of ε increases, the difference between ε and $\varepsilon_{effctive}$ also increases or($\varepsilon_{effctive}$ decreases considerably with an increase in ε) implying that in reality, the attacker requires very less amount of perturbation to effectively harm the model no matter how high the value of applied perturbation is. The losses increases with the increase of applied ε however, the accuracy of the model is intact.

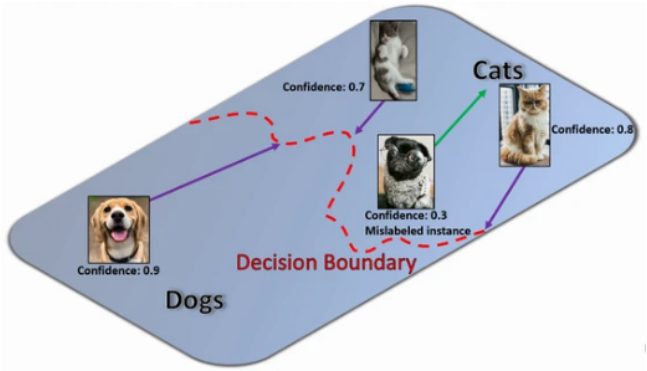

Figure 10. The visualization of ACE on a classifier of cats and dogs with the softmax as uncertainty measure.

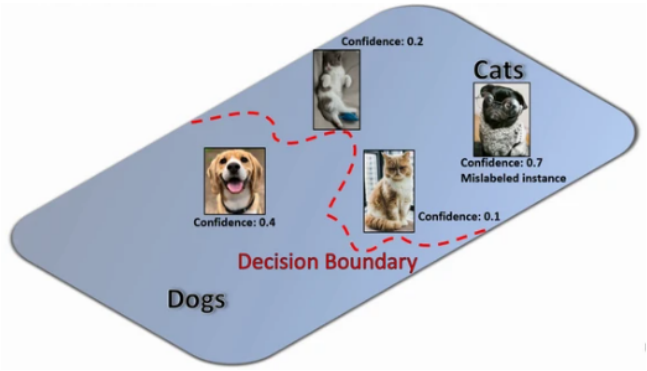

Figure 11. The visualization of final arrangement after ACE on a classifier

5. Limitations and suggestions for improvements

In this section, we would discuss a few limitations concerning the ACE, suggestions for improvements as well as some possible extensions regarding the application domain of the algorithm.

Limitations The fact that knowledge of the instances' ground truths is necessary for carrying out large-scale attacks presents a limitation for ACE because it may be challenging to get. Alternatively, irrespective of the ground truths, the only concern of the attacker is to increase the confidence for certain types of labels and decrease the confidence for other types.

Suggestions for improvement: The efficiency of the ACE algorithm can be improved further by optimizing the perturbation in each iteration rather than when the instance crosses the boundary.

Modifying ACE for the regression task: Although the focus of this paper is classification, subsequent work may alter ACE for regression problems, where the variance of various model outputs is frequently used to quantify the uncertainty. A simple conversion of this algorithm could, for example, define any instance with a loss above some threshold (such as the median loss on some validation set) as an "incorrect

Algorithm 1 Attack on Confidence Estimation

function ACE($f, \hat{f}, \kappa, x, y, \epsilon, \epsilon_{decay}, max\_iterations$)
  $\eta \leftarrow sign(\nabla_x \kappa(x, \hat{y}_f(x)|\hat{f}))$ ▷ κ is calculated with $\hat{f}$ on the label predicted by f
  for $i < max\_iterations$ **do**
    if $\hat{y}_f(x) == y$ **then** ▷ f is correct, decrease confidence for x
      $\tilde{x} \leftarrow x - \epsilon \cdot \eta$
    else ▷ f is incorrect, increase confidence for x
      $\tilde{x} \leftarrow x + \epsilon \cdot \eta$
    if $\hat{y}_f(\tilde{x}) == \hat{y}_f(x)$ **then**
      return $\tilde{x}$ ▷ label is unchanged so accuracy will be unharmed
    else
      $\epsilon \leftarrow \epsilon \cdot \epsilon_{decay}$
  return x ▷ insufficient $max\_iterations$ & ϵ too big

Figure 12. ACE algorithm

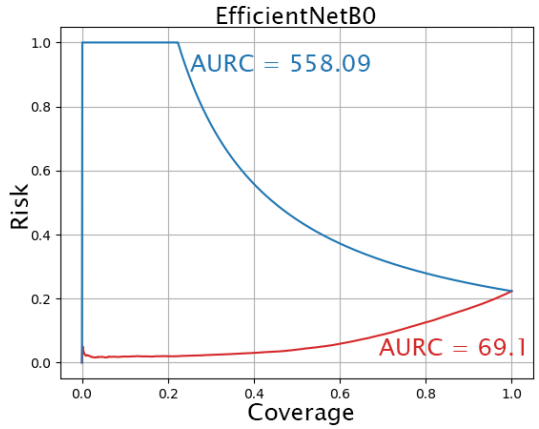

Figure 13. EfficientNet RC curves during a white-box assault with different magnitudes of perturbations(ε).

prediction" and loss values less than the threshold as "correct predictions".The variable that would be under attack would be the variance of the outputs; leading to an increase in variance for the low loss inputs, and a decrease in variance for high loss inputs with respect to the model.

6. Summary

The primary aim of the report was to introduce a new technique of attacking the model called ACE, especially for some risk-sensitive applications. As the name suggests an attack is completely focused to harm the uncertainty estimation performance of the model rather than its accuracy, unlike standard adversarial attacks. To summarize, it basically increases the confidence of the model in its incorrectly predicted instances and decreases that of the correctly predicted ones by adding the perturbation to input images. It keeps the accuracy intact throughout the attack. A brief intuition about the working of the attack with respect to concepts like confidence intervals and selective predictions has also been mentioned. The attack possesses a few benefits over the standard adversarial attack such as a requirement of a much lower amount of perturbation than adversarial attacks, it is less likely to alert the victim and create considerable harm even with a smaller magnitude of the perturbation. The paper also provided an overview of the iterative algorithm to implement the ACE in detail. Additionally, it explains its impact on the model and a comparison of the attacked model under white-box vs. black-box settings. During the comparison, we observed that compared to black-box settings, an attack under white-box settings requires significantly less magnitude of perturbation to cause equivalent destruction. Finally, we concluded by addressing possible limitations and suggesting some changes to make the attack more effective.

	€	€ _effective	NLL	Brier Score	Accuracy
no	0	0	0.963	0.336	76.01
White-box	0.0005	0.000422	1.639	0.562	76.01
Black-box		0.00046	1.308	0.474	76.01
White-box	0.005	0.002245	3.237	0.718	76.01
Black-box		0.003353	2.615	0.652	76.01

Figure 14. Comparision of white-box and black-box settings for different ε for ResNet50

References

[1] Ido Galil and Ran El-Yaniv. Disrupting deep uncertainty estimation without harming accuracy. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.

[2] L.P. Cordella, C. De Stefano, F. Tortorella, and M. Vento. A method for improving classification reliability of multilayer perceptrons. *IEEE Transactions on Neural Networks*, 6(5):1140–1147, 1995.

[3] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning, 2015.

[4] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[5] Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[6] C. De Stefano, C. Sansone, and M. Vento. To reject or not to reject: that is the question-an answer in case of neural classifiers. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 30(1):84–94, 2000.

[7] Yukun Ding, Jinglan Liu, Jinjun Xiong, and Yiyu Shi. Revisiting the evaluation of uncertainty estimation and its

application to explore model complexity-uncertainty trade-off. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 22–31, 2020.

[8] Andrew Howard, Andrey Zhmoginov, Liang-Chieh Chen, Mark Sandler, and Menglong Zhu. Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation. In *CVPR*, 2018.

[9] Cihang Xie, Mingxing Tan, Boqing Gong, Jiang Wang, Alan Yuille, and Quoc V. Le. Adversarial examples improve image recognition, 2019.

7. Results and Discussion

7.1 Evaluation matrices

We assess the effectiveness of ACE using a variety of measures, including the AURC (x 10^3), Negative Log-likelihood (NLL), and Brier score, which are frequently used for DNN uncertainty assessment.

7.1.1 Risk-coverage curve (RC curve):

The risk-coverage curve (RC curve), assessed on a selected test set, is a curve that depicts the selective risk as a function of coverage. Coverage and risk are used to assess a selected model's performance. The coverage measures the proportion of the input that the model processes in absence of human involvement and the risk denotes the level of risk possessed by these model predictions. Where,

$$Coverage = \frac{|X_h|}{|X|}$$

$$Risk = \mathscr{L}\left(\hat{Y}_h\right)$$

Where, $\mathscr{L}$ is a loss function quantifying the quality of prediction. With a confidence score r_i for each input x_i and a threshold t, selective prediction divides the input from dataset X and the prediction $\hat{Y}$ into two parts: $X_h = \{x_i | r_i \geq t\}$, $\hat{Y}_h = \{\hat{Y}_i | r_i \geq t\}$ and $X_l = \{x_i | r_i < t\}$, $\hat{Y}_l = \{\hat{Y}_i | r_i < t\}$ respectivly. [7]

NLL: The Negative Log Likelihood (NLL) is also a loss function of the model defined as

$$\Sigma_{y \in Y} - ln\left(P_y\right)$$

where, Y is the correct labels and P_y is the probability assigned to label y by the model.

Brier score: Can be obtained from a sample of size N for which there are R potential labels can be defined as:

$$\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{R}\left(f_{ij} - o_{ij}\right)^2$$

Where, R is number of potential classes and N is number of onservation. f_{ij} is the predicted probability for i^{th} observation for j^{th} class and o_{ij} is actual observation which takes the value 1 in case of i^{th} observation belongs to j^{th} class.

7.2 Experiments and interpretations

In this section, we will be discussing and interpreting the results obtained by applying the attack on combinations of different architectures with a variety of uncertainty quantification methods under different settings.

7.2.1 Attacking softmax uncertainty

Softmax under white-box settings:

Table in Figure 15 illustrates the outcomes by using ACE for various magnitudes of ε. It is clearly evident from the table, that the $\varepsilon_{effective}$ is approximately half the value or less for greater values of ε, indicating that even fewer resources are required for a highly detrimental attack. The AURC is almost seven times worse under ACE assault with $\varepsilon = 0.005$, whereas the NLL and Brier scores are roughly three times poorer. Figure 16 represnts the RC curves for ACE under white-box settings for MobileNetV2 [8] with softmax score.

White-box	ϵ	Effective ϵ	AURC	NLL	Brier Score	Accuracy
ResNet50	0	0	69.9	0.963	0.336	76.01
	0.00005	0.000049	86.1	1.037	0.371	76.01
	0.0005	0.000422	269	1.639	0.562	76.01
	0.005	0.002245	555.4	3.237	0.718	76.01
EfficientNetB0	0	0	69.1	0.958	0.322	77.67
	0.00005	0.000049	81	1	0.343	77.67
	0.0005	0.000446	291.5	1.416	0.516	77.67
	0.005	0.002563	553.3	2.837	0.815	77.67
Mobilenet V2	0	0	89.7	1.147	0.386	71.85
	0.00005	0.000049	112	1.232	0.428	71.85
	0.0005	0.000397	377.3	1.973	0.671	71.85
	0.005	0.001934	624.8	3.854	0.789	71.85
DenseNet161	0	0	66.8	0.945	0.326	77.15
	0.00005	0.000049	82.6	1.02	0.36	77.15
	0.0005	0.000425	261	1.643	0.54	77.15
	0.005	0.002153	521.6	3.392	0.68	77.15
VGG16	0	0	80.5	1.065	0.366	73.48
	0.00005	0.000049	103.4	1.164	0.413	73.48
	0.0005	0.000387	361.7	2.015	0.646	73.48
	0.005	0.001832	572.3	4	0.7	73.48

Figure 15. Table of for different model architectures with softmax uncertainity quantification for diffrent ε for white-box settings

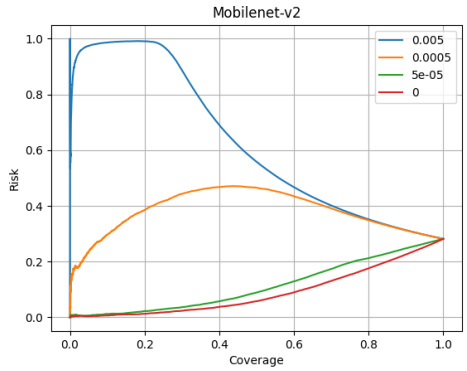

Figure 16. RC curves for ACE under white-box settings for MobileNetV2 with softmax score

Softmax under black-box settings:
Figure 16 represents the RC curves for ACE under white-box settings for MobileNetV2 with softmax score and different values of ε. Be aware that the value of ε for the most effective assault considerably varies on which coverage is targeted. For instance, for MobileNetV2 on coverage 0.6, the selective risk is greater for $\varepsilon = 0.005$ than that for $\varepsilon = 0.05$.

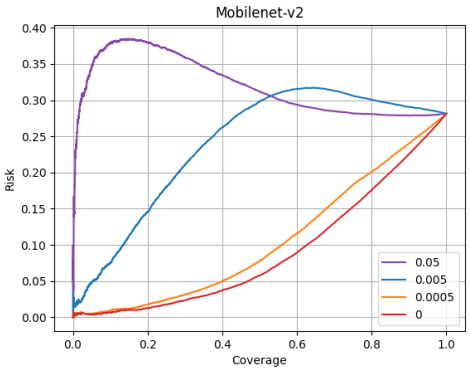

Figure 17. RC curves for ACE under black-box settings for MobileNetV2 with softmax score

7.2.2 Adversarial robustness via adversarial training

The main idea behind the concept of adversarial training is that if employed properly, adversarial examples given as an input during traininng phase can be used to enhance image recognition algorithms[9]. An assessment of ACE performance on an adversarially trained EfficientNetB0 can be found in the table shown in Figure 18. We also provide the outcomes of an EfficientNetB0 that was not trained using adversarial inputs. These findings imply that employing traditional adversarial training does not significantly improve robustness to ACE. We speculate the reason behind that can be the tendency of adversarial training of creating cases with extremely high losses or that can cross the decision boundary. Such cases demand an ε that is relatively large. It may be possible to increase robustness to ACE by adversarial training that employs ACE or uses a variety of smaller values of ε.

7.3 Deep ensembles

Deep ensembles under white-box settings
Table in Figure 19 demonstrates the results of ACE on ensembles of varying sizes consisting of ResNet50 models trained on ImageNet. As can be seen from the table, the AURC deteriorates by a factor of around eight when attacked by ACE with $\varepsilon = 0.005$. It should be emphasized that ACE resilience increases with ensemble size, with even the smallest ensemble being more resilient than a single ResNet50 model.

The RC curve for applying ACE on a size 5 ensemble of ResNet50 models with black-box settings is shown in Figure 20. It should be observed that the selection risk for any coverage above 0.45 for the ResNet50 proxy is somewhat higher for $\varepsilon = 0.005$ than for $\varepsilon = 0.05$, indicating that it is more efficient to employ a smaller ε for these values.

Effects of Adversarial Training	ϵ	Effective ϵ	AURC	NLL	Brier Score	Top1 Accuracy
EfficientNetB0 (Black-box)	0	0.00000	69.1	0.958	0.322	77.67
	0.0005	0.00049	78.8	1	0.342	77.67
	0.005	0.00446	185.1	1.334	0.47	77.67
	0.05	0.02902	300.9	1.775	0.585	77.67
EfficientNetB0 AdvProp (Black-box)	0	0.00000	73.4	1.009	0.337	76.56
	0.0005	0.00050	77.9	1.029	0.346	76.56
	0.005	0.00475	127.3	1.206	0.421	76.56
	0.05	0.03322	295.5	1.736	0.686	76.56
EfficientNetB0 (White-box)	0	0.000000	69.1	0.958	0.322	77.67
	0.00005	0.000049	81	1	0.343	77.67
	0.0005	0.000446	291.5	1.416	0.516	77.67
	0.005	0.002563	553.3	2.837	0.815	77.67
EfficientNetB0 AdvProp (White-box)	0	0.000000	73.4	1.009	0.337	76.56
	0.00005	0.000050	78.5	1.028	0.346	76.56
	0.0005	0.000476	144.7	1.211	0.429	76.56
	0.005	0.003172	569.9	2.537	0.79	76.56

Figure 18. Assessment of an adversarially and non-adverserially trained EfficientNetB0 under black-box and white-box settings

	ϵ	Effective ϵ	AURC	NLL	Brier Score	Accuracy
ResNet50 Ensemble Size 10	0	0	63	0.871	0.314	77.82
	0.00005	4.97E-05	68.8	0.897	0.327	77.82
	0.0005	0.000467573	133.7	1.112	0.425	77.82
	0.005	0.003325908	510.5	2.103	0.678	77.82
ResNet50 Ensemble Size 5	0	0	63.8	0.883	0.317	77.61
	0.00005	4.96E-05	71.2	0.916	0.333	77.61
	0.0005	0.000460713	154.8	1.177	0.449	77.61
	0.005	0.003261595	523.3	2.204	0.696	77.61
ResNet50 Ensemble Size 3	0	0	65.3	0.901	0.321	77.2
	0.00005	4.95E-05	74.4	0.94	0.34	77.2
	0.0005	0.000453732	176.9	1.252	0.474	77.2
	0.005	0.003114596	533.4	2.344	0.713	77.2

Figure 19. ACE under white-box settings on ensembles of different sizes consisting of ResNet50 models trained on ImageNet.

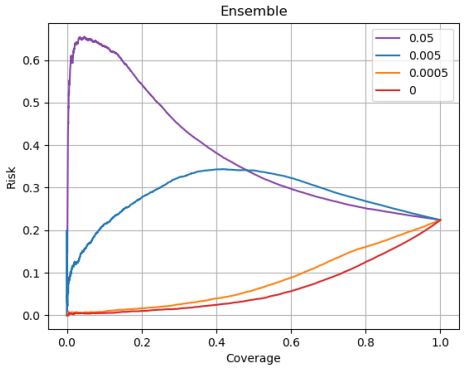

Figure 20. RC curve for applying ACE on a size 5 ensemble of ResNet50 models with black-box settings