

Deep Anomaly Detection with Outlier Exposure

Dan Hendrycks¹ Mantas Mazeika² and Thomas Deitterich³

¹University of California, Berkeley

²University of Chicago

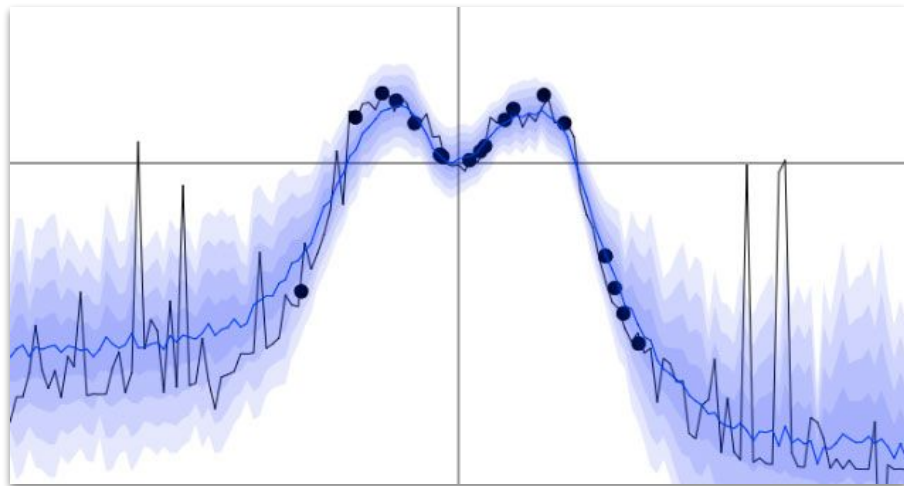
³Oregon State University

Published in ICLR 2019

Presented By:
Anmol Sharma
MIAL

What is Uncertainty?

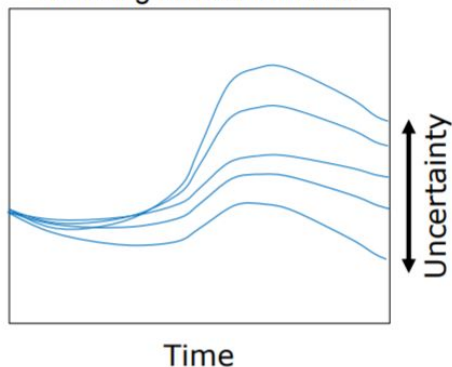
- Ability of a system to quantify the certainty at which the decision it made would be correct.
- Other words: How sure is a system about its own prediction.
- **Question:** Why model uncertainty?
 - Natural phenomenon always has an uncertainty attached to it.
 - Modelling a natural phenomenon from should also take into account related uncertainties.
 - For eg: physics based simulations, quantum tunneling etc.



Type of Uncertainty

Model Uncertainty

Same data (initial conditions),
but using different models



Reducible, as size of
training data increases

Data Uncertainty

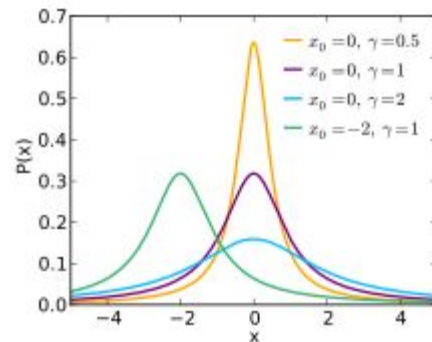
- Also called aleatoric uncertainty
- Arises from class overlap, label noise, homoscedastic¹ and heteroscedastic² noise



Irreducible, as size of
training data increases

Distributional Uncertainty

- Also called dataset shift
- Mismatch between training and test distributions.



Reducible, as more examples
from OOD are added to training.

Modelling Uncertainties Separately

- The three types of uncertainty discussed in the last slide can be modelled as a combination.
- Bayesian Neural Networks model **distributional uncertainty** through **model uncertainty**.
- Ensembling of classifiers also handles **model uncertainty** and **data uncertainty**.
- **Question:** Is there an advantage if we can model each uncertainty separately?
- **Answer:** Yes. Every type of uncertainty requires different steps to be taken in order to reduce it.

Distribution Uncertainty

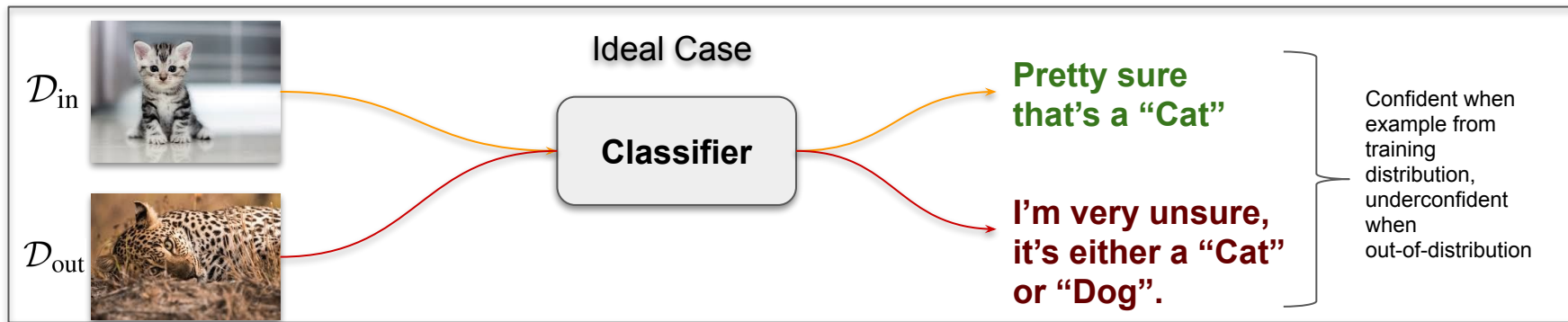
\mathcal{D}_{in}



\mathcal{D}_{out}




Classifier Behaviour in Distribution Uncertainty





Proposed Method: Outlier Exposure (OE)


- Idea is to decentralize each uncertainty, and model them individually.
- Individual modelling will allow uncertainty-specific measures to be taken.
- **Outlier Exposure (OE) models distribution uncertainty**, a.k.a out-of-distribution (OOD) uncertainty.
- OE proposes a small variation in the loss function of a classifier, which when incorporated, provides three benefits:
 - Allows model to learn OOD detection.
 - Calibrates probability estimates to reflect OOD samples (uniform probability across all classes).
 - Improves state-of-art performance by large margin in many tasks.

Outlier Exposure (OE) Terminology

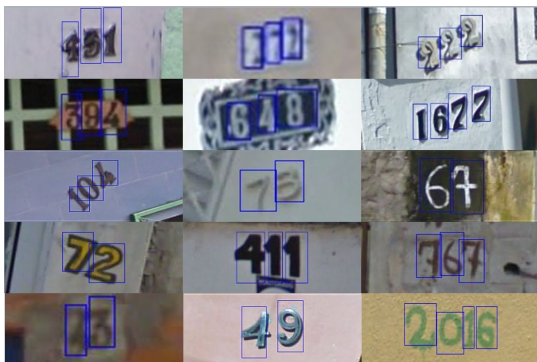
\mathcal{D}_{in}  “In-distribution”, which is the distribution of the training data for classifier.

\mathcal{D}_{out}  “Out-distribution”, any unknown distribution (or combination thereof) which is NOT \mathcal{D}_{in}

$\mathcal{D}_{\text{out}}^{\text{OE}}$  “Out-distribution” from which samples are drawn to train the classifier using proposed OE technique.

$\mathcal{D}_{\text{out}}^{\text{test}}$  “Out-distribution” which is of unknown distribution, unseen by classifier, used as test set.

Outlier Exposure (OE) Terminology

 \mathcal{D}_{in} 

Street View House
Numbers dataset

 \mathcal{D}_{out} 

80 Million Tiny Images
dataset

Outlier Exposure (OE) Terminology

$$\mathcal{D}_{\text{out}}$$

$$\underbrace{\hspace{10em}}_{\mathcal{D}_{\text{out}}^{\text{OE}}} \quad \underbrace{\hspace{10em}}_{\mathcal{D}_{\text{out}}^{\text{test}}}$$

OE Loss Function

$$\mathbb{E}_{(x,y) \sim \mathcal{D}_{\text{in}}} [\underbrace{\mathcal{L}(f(x), y)}_{\text{Sample (x,y) from training distribution}} + \lambda \underbrace{\mathbb{E}_{x' \sim \mathcal{D}_{\text{out}}^{\text{OE}}} [\underbrace{\mathcal{L}_{\text{OE}}(f(x'), f(x), y)]_{\text{OE Loss function, which depends on dataset/task.}}]]$$

Sample (x,y)
from training
distribution

Original learning
objective of the
classifier.

Sample (x')
from OOD

OE Loss function, which
depends on dataset/task.

Minimize the expected value
of this random variable

Minimize the expected value of this
random variable

Experiments and Datasets

- The paper provides extensive experiments on a wide variety of datasets.

\mathcal{D}_{in}



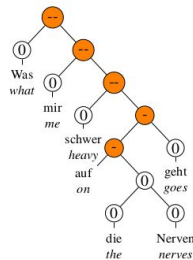
20 Newsgroups Data Set



20 class labels, 20,000 documents
62k unique words

TREC dataset

- ☐ .gov domain web pages in 2002
- ☐ 1,053,110 pages, 11,164,829 hyperlinks
- ☐ 50 queries
- ☐ Binary relevance judgment (relevant or irrelevant)
- ☐ 20 features extracted from each query-document pair (e.g. content features and hyperlink features)



\mathcal{D}_{out}



IMAGENET22K

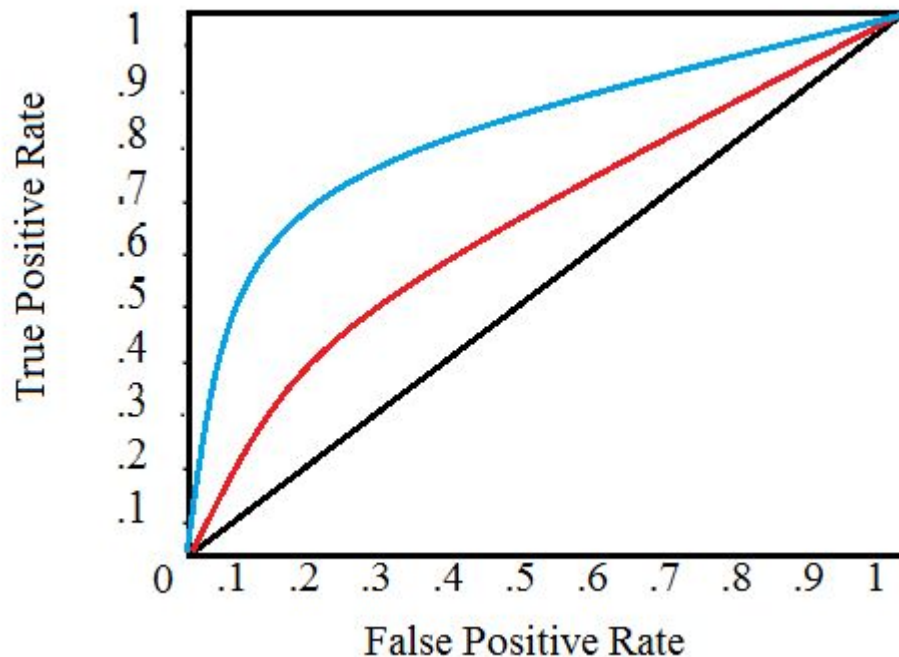
the fed arranged \$ N billion of customer repurchase agreements tuesday the second repurchase agreement in two days the move which <unk> capital into the system is seen as an effort to <unk> the <unk> markets that the u.s. central bank is ready to provide the ample liquidity but other analysts contend that while the fed 's move to loosen credit has n't been aggressive it nevertheless sends a clear signal that at least for now the fed has <unk> its grip on credit they add that the fed has allowed the key federal funds interest rate to dip to about N N N from its levels of just

Metrics

- Performance of vanilla models and models with OE modification are compared using three metrics:
 - False Positive Rate at **N**% True Positive Rate (FPRN) (Lower is better)
 - Area under ROC Curve (AUROC) (Higher is better for test set)
 - Area under Precision-Recall Curve (AUPR) (Higher is better for test set).

Metrics: False Positive Rate at N% True Positive Rate

- Setting True Positive Rate at 95%, what is the False Positive Rate?
- **Lower FPRN is better.**



Metrics: Area under ROC and PRC

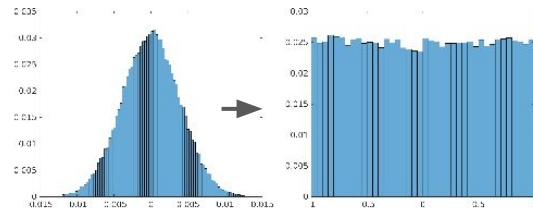
- Area under Receiver Operating Characteristics Curve (ROC) is a standard metric which quantifies the performance of a classifier under different probability thresholds.
 - Higher is better.
- Area under Precision Recall Curve (ROC) is another standard metric which quantifies the performance of a classifier when class distribution is skewed.
 - Higher is better.

Choice of Loss Function for Multiclass Classification

$$\mathbb{E}_{(x,y) \sim \mathcal{D}_{\text{in}}} \underbrace{\left[-\log f_y(x) \right]} + \lambda \mathbb{E}_{x \sim \mathcal{D}_{\text{out}}^{\text{OE}}} \underbrace{\left[H(\mathcal{U}; f(x)) \right]}$$

Negative log probability
of $f(x)$ belonging to real
class y

Cross entropy between a uniform
distribution and posterior distribution
of $f(\cdot)$ for OOD examples



Comparison with Baseline

- OE is compared with a simple baseline detector called the Maximum Softmax Probability (MSP) detector.
- MSP takes softmax distribution from a pre-trained model, and then detects whether the input was an in- or out-of-distribution example.

Results

\mathcal{D}_{in}	FPR95 ↓		AUROC ↑		AUPR ↑	
	MSP	+OE	MSP	+OE	MSP	+OE
SVHN	6.3	0.1	98.0	100.0	91.1	99.9
CIFAR-10	34.9	9.5	89.3	97.8	59.2	90.5
CIFAR-100	62.7	38.5	73.1	87.9	30.1	58.2
Tiny ImageNet	66.3	14.0	64.9	92.2	27.2	79.3
Places365	63.5	28.2	66.5	90.6	33.1	71.0

Computer Vision Datasets

Results

\mathcal{D}_{in}	FPR90 ↓		AUROC ↑		AUPR ↑	
	MSP	+OE	MSP	+OE	MSP	+OE
20 Newsgroups	42.4	4.9	82.7	97.7	49.9	91.9
TREC	43.5	0.8	82.1	99.3	52.2	97.6
SST	74.9	27.3	61.6	89.3	22.9	59.4

NLP Datasets

Comparison with Confidence Branch (Branch)

- OE is compared with another method called Confidence Branch, which adds an auxiliary connection to a pretrained network in order to output calibrated probability values, indirectly detecting OOD examples.

\mathcal{D}_{in}	FPR95 ↓			AUROC ↑			AUPR ↑		
	MSP	Branch	+OE	MSP	Branch	+OE	MSP	Branch	+OE
CIFAR-10	49.3	38.7	20.8	84.4	86.9	93.7	51.9	48.6	66.6
CIFAR-100	55.6	47.9	42.0	77.6	81.2	85.5	36.5	44.4	54.7
Tiny ImageNet	64.3	66.9	20.1	65.3	63.4	90.6	30.3	25.7	75.2

Comparison with GAN Based Training

- Another competing method which uses GAN is used to compare.
- It carefully trains a GAN to generate examples at the decision boundary of classifier. Then, these examples are used to train the network, where it is tasked to output low confidence values for these examples.

\mathcal{D}_{in}	FPR95 ↓			AUROC ↑			AUPR ↑		
	MSP	+GAN	+OE	MSP	+GAN	+OE	MSP	+GAN	+OE
CIFAR-10	32.3	37.3	11.8	88.1	89.6	97.2	51.1	59.0	88.5
CIFAR-100	66.6	66.2	49.0	67.2	69.3	77.9	27.4	33.0	44.7

Density Estimation

- A PixelCNN++ is trained on in-distribution data to detect outliers (or OODs). It outputs a score which correlates with whether the input is in- or -out-of-distribution.
- OE is implemented with PixelCNN++ using a log likelihood difference between in-distribution and anomalous examples.

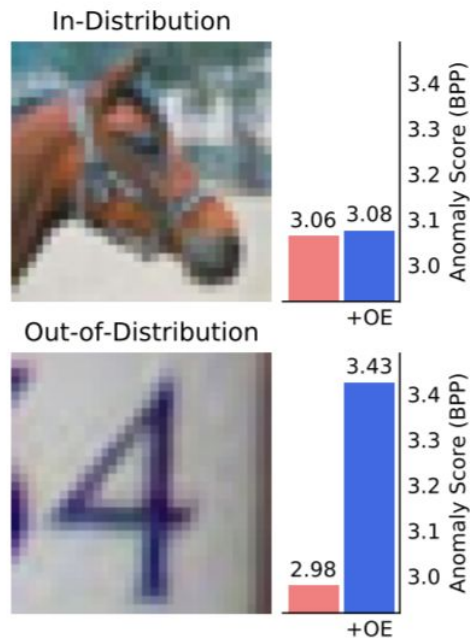


Figure 2: OOD scores from PixelCNN++ on images from CIFAR-10 and SVHN.

Density Estimation

\mathcal{D}_{in}	$\mathcal{D}_{\text{out}}^{\text{test}}$	FPR95 ↓		AUROC ↑		AUPR ↑	
		BPP	+OE	BPP	+OE	BPP	+OE
CIFAR-10	Gaussian	0.0	0.0	100.0	100.0	100.0	99.6
	Rademacher	61.4	50.3	44.2	56.5	14.2	17.3
	Blobs	17.2	1.3	93.2	99.5	60.0	96.2
	Textures	96.8	48.9	69.4	88.8	40.9	70.0
	SVHN	98.8	86.9	15.8	75.8	9.7	60.0
	Places365	86.1	50.3	74.8	89.3	38.6	70.4
	LSUN	76.9	43.2	76.4	90.9	36.5	72.4
	CIFAR-100	96.1	89.8	52.4	68.5	19.0	41.9
Mean		66.6	46.4	65.8	83.7	39.9	66.0

BPP = Bits Per Pixel score, output of the PixelCNN++

Probability Estimate Calibration

- OE can naturally calibrate the probability estimates of the original network to represent the true likelihood of a class prediction.
- OE is compared to the “Temperature Tuning” method.

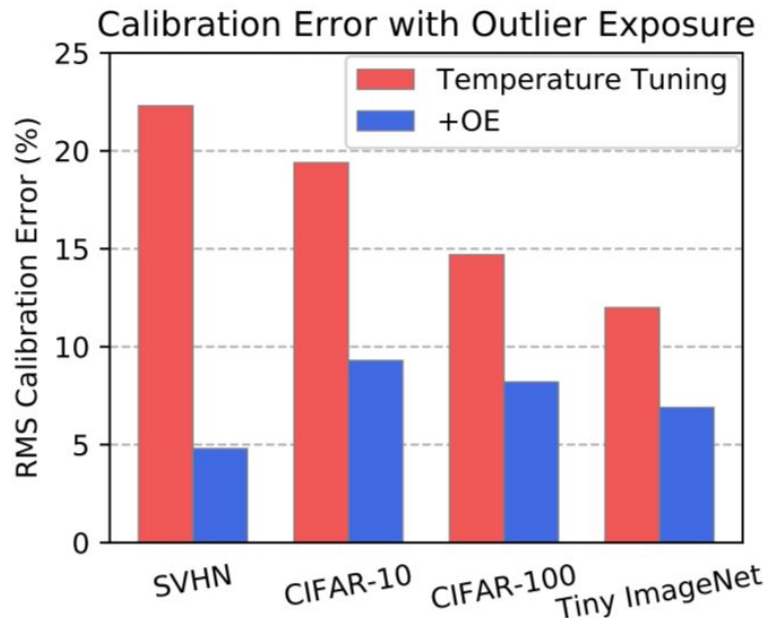


Figure 3: Root Mean Square Calibration Error values with temperature tuning and temperature tuning + OE across various datasets.

Discussion

- Another approach towards OOD detection can be “reject” class built into the original classifier.
 - However, this tends to be uncompetitive to OE.
- The choice of $\mathcal{D}_{\text{out}}^{\text{OE}}$ matters, and is important. Synthetic images, noisy version of \mathcal{D}_{in} data doesn't help much as compared to real images.
- Dataset diversity is important. CIFAR 100 as \mathcal{D}_{in} doesn't work well with CIFAR 10 as $\mathcal{D}_{\text{out}}^{\text{OE}}$.
- The sets $\mathcal{D}_{\text{out}}^{\text{OE}}$ and $\mathcal{D}_{\text{out}}^{\text{test}}$ need not be close to each other at all for optimal performance of OE.
- The sets \mathcal{D}_{in} and $\mathcal{D}_{\text{out}}^{\text{OE}}$ have to close to each other for good performance.

Critique

- Choice of \mathcal{D}_{out} is not clear, and is highly empirical and experimental in nature.
 - Unclear what should be its characteristics.
- Simple baseline of $\mathcal{D}_{\text{in}} + \mathcal{D}_{\text{out}}$ not discussed in paper at all. What happens if we train the model with the out distribution?
 - Discussion Topic
- Only one comparison for the probability estimate calibration (Temperature Tuning), while there exists a vast field that does just this.

Critique

- No investigation \mathcal{D}_{out} on how the performance (in terms of prediction accuracy) would increase if detected OOD examples had their predictions changed to something else.
 - How will the change happen is an open question.
 - Discussion Topic
- The sets \mathcal{D}_{in} and $\mathcal{D}_{\text{out}}^{\text{OE}}$ have to close to each other for good performance. This is a constraint that is too vague in practice. How close?

Questions?

Thank you