

Maximum Likelihood Uncertainty Estimation: Robustness to Outliers

Deebul S. Nair,¹ Nico Hochgeschwender,^{1 2} Miguel A. Olivares-Mendez³

¹Dept. of Computer Science, Bonn-Rhein-Sieg University of Applied Sciences, Germany.

² German Aerospace Center (DLR), Institute for Software Technology, Germany.

³ SpaceR Research Group, University of Luxembourg, Luxembourg.

deebul.nair@h-brs.de, nico.hochgeschwender@h-brs.de, miguel.olivaresmendez@uni.lu

Abstract

We benchmark the robustness of maximum likelihood based uncertainty estimation methods to outliers in training data for regression tasks. Outliers or noisy labels in training data results in degraded performances as well as incorrect estimation of uncertainty. We propose the use of a heavy-tailed distribution (Laplace distribution) to improve the robustness to outliers. This property is evaluated using standard regression benchmarks and on a high-dimensional regression task of monocular depth estimation, both containing outliers. In particular, heavy-tailed distribution based maximum likelihood provides better uncertainty estimates, better separation in uncertainty for out-of-distribution data, as well as better detection of adversarial attacks in the presence of outliers.

Introduction

The ability to estimate the uncertainty along with network prediction has become a relevant feature for the adoption of deep neural networks (DNN) in safety-critical and autonomous systems (Borg et al. 2019; Schwalbe and Schels 2020). Accurate and calibrated uncertainties can be used to gain confidence for making decisions in autonomous systems (Jha et al. 2018; Serban, Poll, and Visser 2020). Uncertainty estimation is a challenging problem, especially in high dimensional data because of the lack of ground truth regarding uncertainty. The presence of noisy labels or outliers in the training data elevates the challenge of uncertainty estimation. Robust uncertainty estimation is the capability of learning algorithms to correctly learn uncertainty by ignoring the outliers. In this work we focus on the following research question: *For a regression problem, given a training dataset, where an η -fraction of data are outliers, how can we robustly estimate uncertainty in the predictions?*

Uncertainty estimation approaches can be broadly classified in three categories: 1. Bayesian methods (Kendall and Gal 2017; Dusenberry et al. 2020b); 2. Sampling based methods (Gal and Ghahramani 2016; Lakshminarayanan, Pritzel, and Blundell 2017); 3. Single-model methods (Nix and Weigend 1994; Amini et al. 2020; Sensoy, Kaplan, and Kandemir 2018). Although, Bayesian and sampling based methods provide state-of-the-art uncertainty estimates

Copyright © 2022, for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

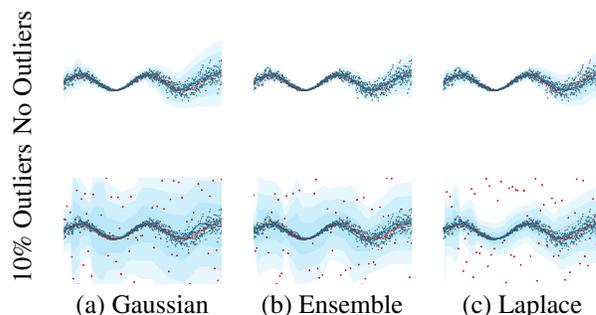


Figure 1: Depiction of the contrast in performance of uncertainty estimation on one dimensional regression data containing outliers. Row 1 shows data with no outliers while row 2 data contains 10% outliers. The blue dots are clean data while red dots are outliers. The shaded blue region is the estimated uncertainty(three standard deviations). (a) Gaussian loss function (Nix and Weigend 1994) (b) Ensemble method (Lakshminarayanan, Pritzel, and Blundell 2017) (c) Laplace loss function (ours)

(Dusenberry et al. 2020a; Wen, Tran, and Ba 2020) they have come at a cost: increased training time, increased prediction time and increased memory requirements. For example, the Bayesian neural learning method by converting weights to distributions (Blundell et al. 2015) requires a doubling in parameters, MC-Dropout (Gal and Ghahramani 2016) with 50 Monte Carlo samples will require 50x forward passes and Deep ensembles (Lakshminarayanan, Pritzel, and Blundell 2017) with an ensemble size of 5 will require 5x the number of weights. These become an operational challenge for their adoption in real-time autonomous systems. On the contrary, single-model based uncertainty estimation methods predict the uncertainty with a single network, no increase in weights and a single forward pass (Nix and Weigend 1994; Malinin and Gales 2019; Sensoy, Kaplan, and Kandemir 2018). In this work, we focus on single-model methods because of their applicability to autonomous systems.

All the uncertainty estimation methods have worked exclusively with clean datasets, however getting clean annotated datasets in real world applications is a difficult task. As a result, real world datasets collected even after human annotation contain large label noise (Algan and Ulu-

soy 2021). Learning in the presence of noisy labels is a well studied problem for the classification task (Arazo et al. 2019; Xia et al. 2019; Li, Soltanolkotabi, and Oymak 2020). Also learning uncertainty in addition to the classification in the presence of label noise is now gaining importance because of safety (Neverova, Novotný, and Vedaldi 2019; Goel and Chen 2021; Northcutt, Jiang, and Chuang 2021). Although all these methods have been evaluated for their prediction and uncertainty quantification for the classification task, their performance for the regression task in the presence of outliers (noisy labels) is not a well studied problem. In this work, we focus on the problem of robustly learning accurate uncertainty in the presence of outliers.

To handle outliers in statistics, heavy-tailed distribution (Student-t distribution, mixture of Gaussians and Laplace) has been introduced (Lange, Little, and Taylor 1989; Tak, Ellis, and Ghosh 2019; Pasi, Jarno, and Aki 2011). They model the noise in the data using a heavy-tailed distribution. As the heavy-tailed distribution assumes some data can be present away from the mean, the outlier data produces comparatively less loss and have minimum impact while training. In this paper, we study the use of heavy-tailed distributions for maximum likelihood based uncertainty estimation and benchmark their robustness to outliers in labels. In particular, we use the Laplace distribution as a heavy-tailed distribution because of its high breakaway point (Bosse, Agammonni, and Gilitschenski 2016), indicating the high proportion of outlier data it can handle.

We formulate the loss function by taking the negative log-likelihood of the Laplace distribution. This loss function provides protection against outliers by reducing their effects while training. This can be demonstrated with the help of a toy example as shown in Fig. 1. The dataset is a one dimensional (1D) regression problem where the ground truth y is a sine-wave. The dataset contains heteroskedastic noise with varying levels of uncertainty across x . In addition to the noisy data we also add 10% outliers to the data. The addition of outliers results in a complete mis-calibration of uncertainty estimated by Gaussian loss function (Nix and Weigend 1994) and Evidential (Amini et al. 2020) however, the proposed Laplace loss function uncertainty estimates are not that degraded.

We benchmark the robustness of uncertainty estimation methods when trained with noisy data. Specifically, this work makes the following contributions:

1. we improve the robustness of uncertainty estimation by modeling the loss function using a heavy-tailed distribution;
2. we evaluate the robustness of the loss function on a standard regression datasets benchmark and complex vision regression task of depth estimation; and
3. we evaluate the use of predicted uncertainty for out-of-distribution (OOD) detection and adversarial attacks detection.

Uncertainty Estimation in Regression

In a regression learning problem we have a training dataset $D = (x, y)$ drawn from a joint distribution $D(X, Y)$, where for each co-variate sample pair, $x \in X$ is the input data and

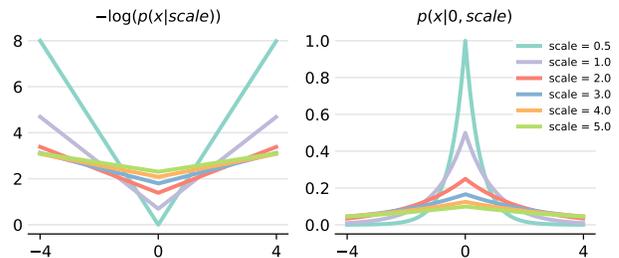


Figure 2: Negative log likelihood (NLL) (left) and probability density function (PDF) (right) of the Laplace loss function

$y \in Y$ is the label data. In addition we define a neural network f_θ parameterized by θ . We train the parameters θ of the neural network by minimizing the empirical loss function ℓ .

$$\hat{\theta} = \underset{\theta}{\text{minimize}} R(\theta); \quad R(\theta) = \underset{\langle x, y \rangle \in D}{E} [\ell(x, y, \theta)]$$

In a typical (i.e. deterministic and non-robust) regression problem, the loss function can be the sum of squared residuals.

$$\ell(x, y, \theta) = \frac{1}{2}(y - f_\theta(x))^2$$

The above problem formulation can only do point estimates and can be considered as estimating the mean of a probability distribution. A simple extension is to estimate the entire conditional probability distribution by predicting variance σ^2 (in case of Gaussian distribution assumption) in addition to the point estimate μ (Nix and Weigend 1994). However, a Gaussian distribution is not robust to outliers in training data.

Improving Robustness with Heavy-Tailed Distribution

We define the problem of robust uncertainty estimation.

Definition 1 (Robustly learning the output of a neural network with bounded second moments under additive outliers). *Let D be the set of n samples so that $D = D_{clean} \cup D_{outlier}$, where D_{clean} is a subset of D and $D_{outlier}$ satisfies $|D_{outlier}| < \eta n$. Given an η -outlier set of samples from $D(x, y)$ and learn the output of a neural network \hat{y} and var such that it minimizes $|y_{clean} - \hat{y}|$ and $|\text{var}(y_{clean}) - \text{var}(\hat{y})|$*

Thus the goal of a robust loss function will be to accurately predict the output and its uncertainty by ignoring outliers in the data. In this paper we have selected a heavy tail distribution, Laplace distribution for modeling the uncertainty in the output.

Regression models using heavy-tailed error distributions to accommodate outliers are studied in statistics by (West 1984). The heavy tail of the distribution reduces the impact of the outlier while finding the maximum likelihood and thus has limited impact on the predicted value and the uncertainty. We choose Laplace as the heavy-tailed distribution because of its well-defined moments and it has a concave loss function, as is required for the uncertainty estimation.

Laplace Maximum Likelihood The probability density function of a Laplace distribution is given by:

$$p(y|\mu, s) = \frac{1}{2s} \exp\left(-\frac{|y - \mu|}{s}\right) \quad (1)$$

where $p(y|\mu, s)$ is only defined for $s > 0$ and $\mu \in R$. We use the Negative Log likelihood (NLL) of the Laplacian distribution $-\log(p(y|\mu, s))$ for training the neural networks

$$\ell_{NLL}(x, y, \theta) = -\log(p(y|f_{\theta}^{\mu}, f_{\theta}^s)) = \log(2f_{\theta}^s) + \frac{|y - f_{\theta}^{\mu}|}{f_{\theta}^s} \quad (2)$$

The probability density function (PDF) and the NLL for different values of f_{θ}^s are plotted in Fig. 2. The loss function has several properties that makes it suitable for gradient-based optimization. The loss function is a convex function. The convexity ensures that the loss is minimum when $|y - \mu|$ is minimum and increases monotonically with respect to $|y - \mu|$.

$$\min(-\log(p(y|f_{\theta}^{\mu}, f_{\theta}^s))) \Rightarrow f_{\theta}^{\mu} = y \quad \frac{\partial \ell}{\partial |y - f_{\theta}^{\mu}|} \geq 0 \quad (3)$$

The loss monotonically increases with respect to f_{θ}^s :

$$\frac{\partial \ell}{\partial f_{\theta}^s}(|y - f_{\theta}^{\mu}|, f_{\theta}^s) \geq 0 \quad (4)$$

This property helps to learn the correct aleatoric uncertainty. The monotonicity guaranteed by the loss function with respect to s forces the optimization algorithm to reduce f_{θ}^s when there is less aleatoric noise in the data. As can be observed in Fig. 2, when f_{θ}^s is reduced the minimum of NLL reduces, thus ensuring that the correct uncertainties are learned.

Related Work

Uncertainty Estimation Nix and Weigend (1994) estimated the uncertainty for regression problems by modeling the output of the neural network as a Gaussian distribution and learning the output using a Gaussian NLL loss function. This was the first demonstration of how the neural network can learn the uncertainty in addition to the predicted value. In the deep learning era, uncertainty estimation in large-scale vision tasks was demonstrated by Blundell et al. (2015) which represented the weights of network with distribution. Gal and Ghahramani (2016) estimated the uncertainty by using dropouts and multiple forward passes. The current state-of-the-art in uncertainty estimation is based on ensemble methods (Lakshminarayanan, Pritzel, and Blundell 2017; Dusenberry et al. 2020a; Wen, Tran, and Ba 2020) which use a set of models under a single one. Even though all these methods provide good uncertainty estimates, they still require multiple forward passes from the model.

Single-Model Approaches to Uncertainty Estimation Uncertainty estimation using a single-model can be achieved by replacing loss function based approaches (Malinin and Gales 2018; Sensoy, Kaplan, and Kandemir 2018; Amini et al. 2020), computing closed-form posterior for the output layer (Riquelme, Tucker, and Snoek 2018; Snoek

et al. 2015), changing the output layer (Calandra et al. 2016; Tagasovska and Lopez-Paz 2019), spectral normalization (Liu et al. 2020), or by two-sided gradient penalty (Van Amersfoort et al. 2020). Laplace maximum likelihood builds on these approaches by replacing the loss function and changing the output layer.

Robust Training in Neural Networks Robust training in the presence of outliers is a classical statistics problem and is dominated by M-estimator methods by Huber (2004). In neural networks, a generalized M-estimator loss function was proposed by Barron (2019), which uses the negative log of the density function to improve robustness while training. Lathuilière et al. (2018) proposed the use of Gaussian-uniform mixture model as a loss function which continuously adapts as per the outliers in data.

Experiments

In this section we benchmark performance of the proposed loss function. We will show that the proposed loss function improves robustness to outliers in data. These benchmarks are not intended to represent the state-of-the-art for any particular task; on the contrary they are intended to demonstrate the capability of our loss function when learning uncertainty with data containing outliers. We compare the following uncertainty estimation methods: 1. **Gaussian** refers to the model introduced in (Nix and Weigend 1994) 2. **Ensemble** corresponds to the ensemble of deep learning method proposed by (Lakshminarayanan, Pritzel, and Blundell 2017) 3. **Evidential** refers to the Normal-inverse-Gamma distribution based uncertainty estimation by (Amini et al. 2020) 4. **Laplace** refers to the method we propose. We benchmark these methods in terms of uncertainty prediction in a 1D outlier regression dataset, real world regression datasets and NyuV2 monocular depth estimation dataset with outliers (Silberman et al. 2012).

We first discuss the need for new metrics for uncertainty comparison and select an appropriate metric to benchmark the predicted uncertainty. In the subsequent section we focus on benchmarking the breakaway point, which is the percentage of outliers that can be present in the data without any significant change in the predicted output. We use a synthetic 1D dataset for adding different percentage of outliers. As we will show, our proposed loss function is particularly effective even with a high percentage of outliers. We further benchmark the methods on real world datasets of regression to demonstrate how the proposed loss function performs compared to other methods. Finally, we focus on vision based learning task of monocular depth estimation. Here we test the robustness of the methods to outliers by using the original depth dataset with outlier sensor noise. The results demonstrate the capability of our proposed loss function to learn correct depth even in the presence of outliers.

Assessing Uncertainty Estimation Techniques

As this work focuses on the robustness capacity of DNNs to correctly estimate the uncertainty, we needed metrics which access the quality of the estimated uncertainty separately from the quality of prediction. *Scoring Rules* are a group

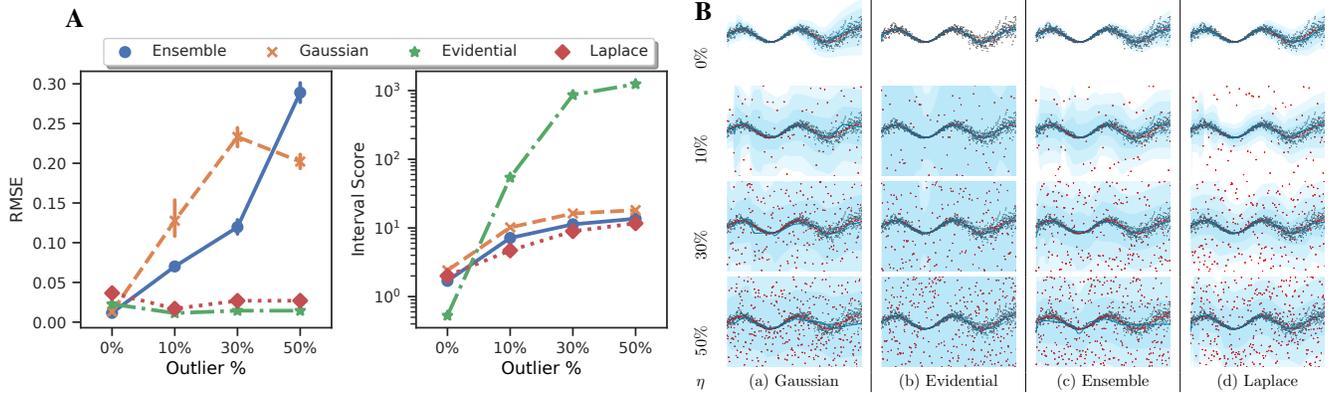


Figure 3: **Breakaway Point estimation:** (A) The change in RMSE and Interval score with respect to outlier percentage in the data for the 1D regression dataset. (B): The uncertainty of the neural network output with different methods learned at different outlier percentages η . The shaded blue region is three standard deviations.

of such metrics which assess the quality of predicted uncertainties, by assigning a numerical score based on both the prediction and uncertainty (Gneiting and Raftery 2007). Most of the literature in uncertainty estimation for regression use root mean square error (RMSE) for comparing performance of model and NLL for comparing performance of uncertainty. Although NLL is a proper Scoring Rule it is not comparable across different distributions. In this work we use *Interval score* Scoring Rule.

Interval score uses the prediction interval, with lower and upper endpoints represented by predictive quantiles at levels $\frac{\alpha}{2}$ and $1 - \frac{\alpha}{2}$. The Interval score is defined as :

$$S_{\alpha}^{int}(l, u; y) = (u-l) + \frac{2}{\alpha}(l-y)\mathbf{1}\{y < l\} + \frac{2}{\alpha}(y-u)\mathbf{1}\{y > u\} \quad (5)$$

The score rewards narrow predictions and penalizes prediction outside the interval. For our experiments we fix the α to 95%. Based on the mean and variance predicted by the neural network, we first calculate the 95% quantile prediction interval (l, u) which is then used to calculate the Interval score. As the Interval score compares the 95% quantile prediction interval it becomes comparable across different distributions (in our case Gaussian and Laplace distribution).

Empirical Breakaway Point Benchmark

In this experiment, we empirically benchmark the breakaway point for all the methods. Breakaway point is the percentage of outliers that can be present in the dataset after which the estimator predicts statistically wrong outputs (Huber 2004). Here we interpret the outputs as both the predicted output and the predicted uncertainty. The regression problem we consider is the 1D regression problem as shown in Fig. 1 which is a sine wave with increasing aleatoric uncertainties along the x-axis. The particular dataset was selected such that the network has to learn the underlying aleatoric uncertainty even in the presence of outliers. The neural network is 4 layer fully connected network with 100 neurons in each layer with rectified linear unit

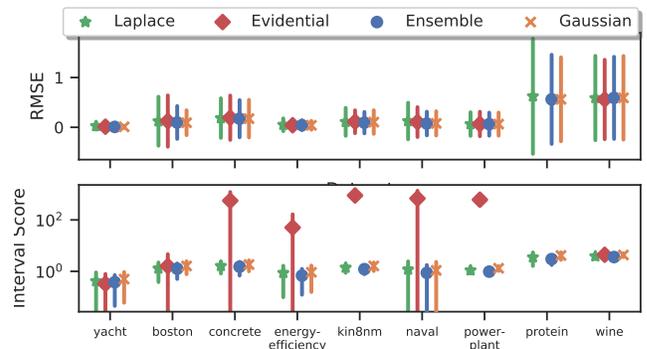


Figure 4: **Benchmark Regression Tasks:** RMSE and Interval score for different methods and datasets. Our method achieves statistical comparable results in eight of the nine datasets in RMSE. While for Interval score it provides comparable in all the nine datasets

(ReLU) activation function. An illustrative scatter plot of the dataset with outliers and the predicted uncertainty is shown in Fig. 3-B. In this benchmark we start with no outliers and increase the outliers upto 50% of the original dataset and record RMSE and Interval score for each dataset. Fig. 3-A plots the RMSE and Interval score for different levels of outliers. Gaussian and Ensemble methods have breakaway points at 10% outliers for the output prediction (Fig. 3-A left plot), while for the uncertainties the breakaway point is 10% for Evidential learning (Fig. 3-A right plot). In this benchmark, we can conclude that the breakaway points are different for the predicted output value and its uncertainty. Some methods are better at output prediction and worse at uncertainty prediction, while Laplace method performs best on both the scenarios in the given example.

Benchmark Regression Tasks

In addition to the toy datasets we validate the uncertainty estimation with respect to real world datasets as used in the

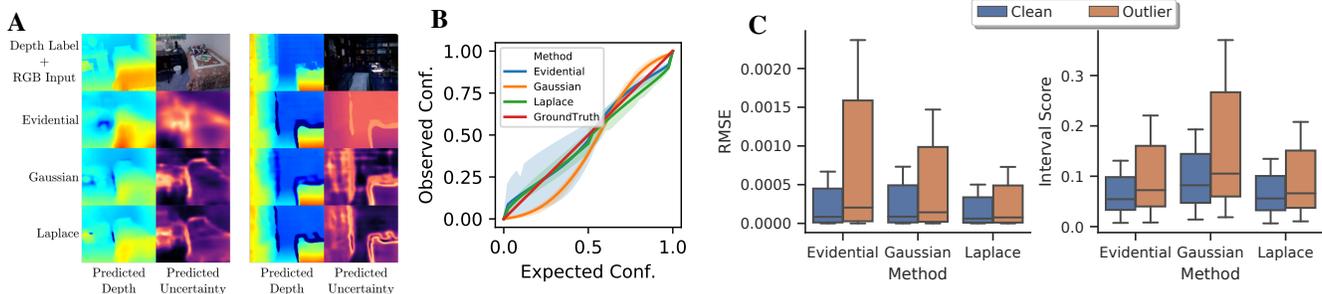


Figure 5: **Monocular Depth Estimation** Trained on the NYU-Depth-v2-dataset containing outliers. (A) Example of estimated depth images and corresponding uncertainty per method. (B) Uncertainty calibration plot (C) Performance comparison between models trained on clean data and outlier data.

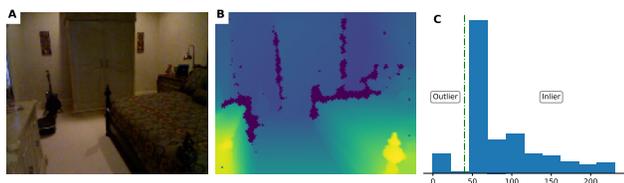


Figure 6: **Outliers in the NYU-Depth-v2-dataset (Silberman et al. 2012)**. (A) Input RGB image (B) Depth image captured from Kinect camera. The dark spots are outliers where no depth information is available because of sensor noise (C) The histogram of the depth where the outliers can be separated

Lakshminarayanan, Pritzel, and Blundell (2017); Gal and Ghahramani (2016); Amini et al. (2020). We replicate the experiment and compare RMSE and the Interval score. The purpose of the experiment is to compare the uncertainty quality obtained from different methods to that of the proposed method on real world datasets. Each experiment is executed 20 times each with random sampling of the train and test dataset. The network used is a fully-connected network with 3 hidden layers containing 100 neurons, and was trained to convergence. All models were trained with learning rate $\eta = 5e^{-3}$ and batch size of 512. The results are shown in Fig. 4. As seen in the results the proposed Laplace method performs comparable to other uncertainty estimation methods both in RMSE and Interval score. The Evidential method shows some deviation in the predicted uncertainty from the other methods in the Interval score plot for all four datasets.

Monocular Depth Estimation with Outliers

Monocular depth estimation is a supervised regression task, where the model is trained to predict depth of each corresponding pixel. The input data is a RGB image of size $H \times W \times 3$ and the label is a depth image with size $H \times W$ whose pixel values represent the depth of the scene in meters. We evaluate monocular depth estimation on the NYU-Depth-v2-

dataset which is collected in an indoor environment using a Kinect v1 camera.

In fields of robotics, the datasets collected are noisy and always contaminated with outliers. For example, in Fig. 6-B we can observe black spots (outliers) which contain no depth information, in the collected depth labels because of sensor noise in the Kinect v1 camera. In the NYU-Depth-v2-dataset the outliers in the images were removed by using the Levin Colorization method. For our work, we have not removed the outliers and used the depth labels along-with the outliers. In order for the results to be comparable to Amini et al. (2020), rather than directly using the outlier labels, we simulate the outliers in the cleaned depth data using simulated noise model of a Kinect v1 camera (Handa et al. 2014; Barron and Malik 2013; Bohg et al. 2014).

The training dataset consists of 27K RGB-depth image pairs. For comparison, we use the architecture similar to Amini et al. (2020) which is a U-Net (Ronneberger, Fischer, and Brox 2015) with spatial dropout. The goal of the experiment is not to prove state-of-the-art depth estimation but comparison of uncertainty estimation, hence we have limited to the popular depth estimation architecture. The U-Net model consists of five convolutional and pooling blocks in both the down-sampling and up-sampling parts. The input image shape is $(160, 128)$. The final layer outputs $H \times W \times n$ layers, where $n = 2$ for Gaussian and Laplace regression while $n = 4$ for Evidential regression. We perform three independent runs for all the methods and report the distribution in all the experiments. The hyper-parameters used are: Adam optimizer with learning rate $5e^{-5}$, batch size of 32 and over 70000 iterations. The evaluation is done using a disjoint dataset with non-outlier depth labels.

Clean Data vs Outlier Data We report the comparison of RMSE and Interval score between models trained on clean depth data and depth data containing outliers in Fig. 5-C. Clean depth data here we refer to the NYU-Depth-v2-dataset in which the outliers are removed by using the Levin Colorization method, while outlier data is the depth data containing outliers. The goal of the experiment is to showcase the degradation in the uncertainty estimation when the depth data contains outliers. In Fig. 5-C we can observe the change

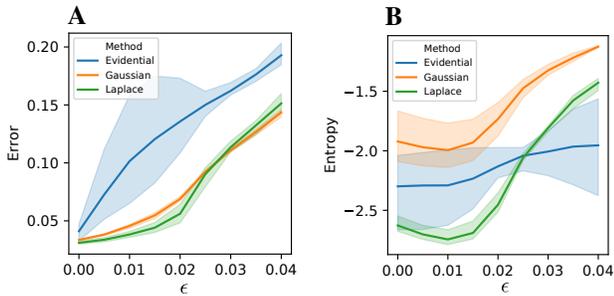


Figure 7: **Adversarial detection** (A) Error, (B) Entropy as a function to adversarial perturbation ϵ on the NYU-Depthv2-dataset.

in performance and uncertainty estimation when trained with outliers in data. First we observe that adding outliers degrades performance of all the three methods which is as expected. For Evidential, we observe there is large shift in RMSE but less in Interval score, while for Gaussian the Interval score shift is highest. The change between both the metrics is minimal for Laplace regression indicating robustness to outliers.

Uncertainty Calibration Here we measure uncertainty calibration of each method using calibration curves, where the expected confidence (inverse of uncertainty) is plotted with respect to observed confidence. The expected confidence is calculated based on the error; for samples with minimum error we expect maximum confidence and vice versa. For a well calibrated method, the curve will follow the line $y = x$. In Fig. 5-B we plot the calibration curves for three methods. The results show calibration curves over three independent trials represented by the shaded region in different color. Here we observe that the Gaussian method underestimates confidence in the low confidence region and overestimates confidence in the high confidence region. Evidential shows calibrated predictions, however, there is large spread between indicating lack of reliability in learning. Overall Laplace has the comparative best calibration over multiple trials.

Adversarial Attack Detection The models which are trained in the above section are evaluated on an adversarial dataset generated using the Fast Gradient Sign Method (FGSM) (Goodfellow, Shlens, and Szegedy 2015), using various values of the adversarial perturbation coefficient ϵ . Adversarial images are artificial generated images with ϵ intentional feature perturbation which causes the model to give wrong predictions. The adversarial attack results for different values of ϵ are shown in Fig. 7. We observe in Fig. 7-A, that the error increases with increase in adversarial attack for all the methods. In Fig. 7-B, we plot the entropy over the range of ϵ . The desirable result is that the entropy should increase for increasing level of ϵ (Amini et al. 2020), indicating higher uncertainty for higher adversarial inputs. We observe Laplace and Gaussian show an increase in entropy as adversarial input increases. For Gaussian, we also observe that for lower ϵ it has higher spread of entropy, thus indicat-

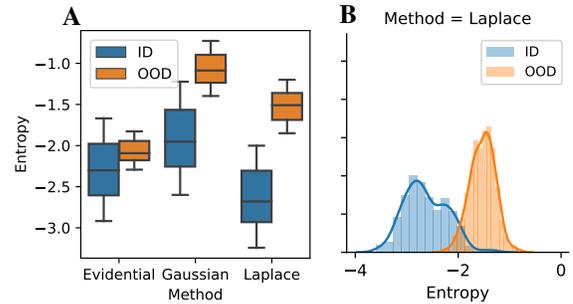


Figure 8: **Out of distribution** Entropy comparison for in-distribution (ID) data and out-of-distribution (OOD) data (A) Comparison Box plot of entropy between ID and OOD data (B) Entropy distribution for the Laplace method

ing high uncertainty for no adversarial input images. The rise of uncertainty is minimum for Evidential. Thus we can conclude that the Laplace loss function learns to estimate appropriate uncertainty while learning from data containing outliers.

Out-of-Distribution Detection We also evaluated our trained models on an OOD dataset. The ApolloScape dataset by Huang et al. 2018 was selected as the OOD dataset. The ApolloScape dataset is also a depth estimation dataset consisting of images from outdoor settings, thus making it appropriate for testing OOD. The desirable result is an increased uncertainty for OOD data predictions as compared to In-distribution (ID) predictions. The predicted uncertainty for OOD data should also be statistically different than the ID predictions. This clear separation helps in defining a threshold value for making informed decisions on trusting the predictions. Fig. 8-A plots the interquartile box plots of entropy which shows that only the Laplace method is able to statistically distinguish between ID and OOD data. The separation is minimum for the Evidential method. This is corroborated in Fig. 8-B where we plot the distribution of entropy for both ID and OOD data from the Laplace method.

Conclusion and Outlook

We have benchmarked maximum likelihood based uncertainty estimation for deep regression. We observed that using Gaussian-based loss functions are not robust to outliers in the training data and consequently provide inaccurate uncertainty estimates. We proposed a heavy-tailed distribution based loss function as an alternative in order to improve robustness to outliers. The heavy-tailed Laplace loss function accurately estimates uncertainty in predictions and has a high breakaway point compared to the other methods. When applied to high dimensional datasets containing outliers, such as depth estimation datasets, the Laplace loss function is able to better estimate the uncertainties. Our proposed loss function improves state-of-the-art in three uncertainty estimating benchmarks: 1. high breakaway point; 2. detection of OOD ; 3. detection of adversarial inputs. In the future

work, we would benchmark performance by modeling the loss function with other heavy-tailed distributions. The proposed robust loss function could benefit in building software which uses uncertainty from the neural network for safe deployment of deep neural networks in autonomous systems.

Acknowledgments

Deebul Nair gratefully acknowledges the ongoing support of the Bonn-Aachen International Center for Information Technology and a PhD scholarship from the Graduate Institute of the Bonn-Rhein-Sieg University. This work was supported by the European Union’s Horizon 2020 project SciRoc (grant agreement No 780086), SESAME (grant agreement No 101017258) and DLR CASSy project. All the code is available at <https://github.com/deebuls/uncertainty-robustness-deep-regression-benchmark>,

References

- Algan, G.; and Ulusoy, I. 2021. Image classification with deep learning in the presence of noisy labels: A survey. *Knowledge Based Systems*, 215: 106771.
- Amini, A.; Schwarting, W.; Soleimany, A.; and Rus, D. 2020. Deep Evidential Regression. In *Advances in Neural Information Processing Systems*, volume 33, 14927–14937.
- Arazo, E.; Ortego, D.; Albert, P.; O’Connor, N.; and McGuinness, K. 2019. Unsupervised label noise modeling and loss correction. In *International Conference on Machine Learning*, 312–321.
- Barron, J. T. 2019. A general and adaptive robust loss function. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4331–4339.
- Barron, J. T.; and Malik, J. 2013. Intrinsic Scene Properties from a Single RGB-D Image. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
- Blundell, C.; Cornebise, J.; Kavukcuoglu, K.; and Wierstra, D. 2015. Weight uncertainty in neural network. In *International Conference on Machine Learning*, 1613–1622.
- Bohg, J.; Romero, J.; Herzog, A.; and Schaal, S. 2014. Robot arm pose estimation through pixel-wise part classification. *International Conference on Robotics and Automation*.
- Borg, M.; Englund, C.; Wnuk, K.; Duran, B.; Levandowski, C.; Gao, S.; Tan, Y.; Kaijser, H.; Lönn, H.; and Törnqvist, J. 2019. Safely Entering the Deep: A Review of Verification and Validation for Machine Learning and a Challenge Elicitation in the Automotive Industry. *Journal of Automotive Software Engineering*, 1(1): 1–19.
- Bosse, M.; Agamennoni, G.; and Gilitschenski, I. 2016. Robust Estimation and Applications in Robotics. *Found. Trends Robotics*, 4(4): 225–269.
- Calandra, R.; Peters, J.; Rasmussen, C. E.; and Deisenroth, M. P. 2016. Manifold Gaussian processes for regression. In *2016 International Joint Conference on Neural Networks*, 3338–3345. IEEE.
- Dusenberry, M.; Jerfel, G.; Wen, Y.; Ma, Y.; Snoek, J.; Heller, K.; Lakshminarayanan, B.; and Tran, D. 2020a. Efficient and scalable bayesian neural nets with rank-1 factors. In *International conference on machine learning*, 2782–2792.
- Dusenberry, M. W.; Tran, D.; Choi, E.; Kemp, J.; Nixon, J.; Jerfel, G.; Heller, K.; and Dai, A. M. 2020b. Analyzing the role of model uncertainty for electronic health records. In *ACM Conference on Health, Inference, and Learning*, 204–213.
- Gal, Y.; and Ghahramani, Z. 2016. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In *International Conference on Machine Learning*, volume 48 of *JMLR Workshop and Conference Proceedings*, 1050–1059.
- Gneiting, T.; and Raftery, A. E. 2007. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477): 359–378.
- Goel, P.; and Chen, L. 2021. On the Robustness of Monte Carlo Dropout Trained with Noisy Labels. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2219–2228.
- Goodfellow, I.; Shlens, J.; and Szegedy, C. 2015. Explaining and Harnessing Adversarial Examples. In *International Conference on Learning Representations*.
- Handa, A.; Whelan, T.; McDonald, J.; and Davison, A. J. 2014. A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM. In *International Conference on Robotics and Automation*, 1524–1531. IEEE.
- Huang, X.; Cheng, X.; Geng, Q.; Cao, B.; Zhou, D.; Wang, P.; Lin, Y.; and Yang, R. 2018. The ApolloScape Dataset for Autonomous Driving. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 1067–10676.
- Huber, P. J. 2004. *Robust statistics*, volume 523. John Wiley & Sons.
- Jha, S.; Raman, V.; Sadigh, D.; and Seshia, S. A. 2018. Safe Autonomy Under Perception Uncertainty Using Chance-Constrained Temporal Logic. *Journal of Automated Reasoning*, 60(1): 43–62.
- Kendall, A.; and Gal, Y. 2017. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? In *Advances in Neural Information Processing Systems*, volume 30.
- Lakshminarayanan, B.; Pritzel, A.; and Blundell, C. 2017. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. In *Advances in Neural Information Processing Systems 30, Long Beach, CA, USA*, 6402–6413.
- Lange, K. L.; Little, R. J. A.; and Taylor, J. M. 1989. Robust Statistical Modeling Using the t Distribution. *Journal of the American Statistical Association*, 84: 881–896.
- Lathuilière, S.; Mesejo, P.; Alameda-Pineda, X.; and Horaud, R. 2018. Deepgum: Learning deep robust regression with a gaussian-uniform mixture model. In *European Conference on Computer Vision*, 202–217.
- Li, M.; Soltanolkotabi, M.; and Oymak, S. 2020. Gradient descent with early stopping is provably robust to label noise for overparameterized neural networks. In *International conference on artificial intelligence and statistics*, 4313–4324.

- Liu, J. Z.; Lin, Z.; Padhy, S.; Tran, D.; Bedrax-Weiss, T.; and Lakshminarayanan, B. 2020. Simple and Principled Uncertainty Estimation with Deterministic Deep Learning via Distance Awareness. In *Advances in Neural Information Processing Systems 33*.
- Malinin, A.; and Gales, M. 2018. Predictive uncertainty estimation via prior networks. In *International Conference on Neural Information Processing Systems*, 7047–7058.
- Malinin, A.; and Gales, M. J. F. 2019. Reverse KL-Divergence Training of Prior Networks: Improved Uncertainty and Adversarial Robustness. In *Advances in Neural Information Processing Systems*.
- Neverova, N.; Novotný, D.; and Vedaldi, A. 2019. Correlated Uncertainty for Learning Dense Correspondences from Noisy Labels. In *Advances in Neural Information Processing Systems 32*, 918–926.
- Nix, D. A.; and Weigend, A. S. 1994. Estimating the mean and variance of the target probability distribution. In *International Conference on Neural Networks*, volume 1, 55–60. IEEE.
- Northcutt, C.; Jiang, L.; and Chuang, I. 2021. Confident learning: Estimating uncertainty in dataset labels. *Journal of Artificial Intelligence Research*, 70: 1373–1411.
- Pasi, J.; Jarno, V.; and Aki, V. 2011. Robust Gaussian Process Regression with a Student-t Likelihood. *Journal of Machine Learning Research*.
- Riquelme, C.; Tucker, G.; and Snoek, J. 2018. Deep Bayesian Bandits Showdown: An Empirical Comparison of Bayesian Deep Networks for Thompson Sampling. In *6th International Conference on Learning Representations*.
- Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, 234–241. Springer.
- Schwalbe, G.; and Schels, M. 2020. A Survey on Methods for the Safety Assurance of Machine Learning Based Systems. In *10th European Congress on Embedded Real Time Software and Systems*.
- Sensoy, M.; Kaplan, L. M.; and Kandemir, M. 2018. Evidential Deep Learning to Quantify Classification Uncertainty. In *Advances in Neural Information Processing Systems 31*, 3183–3193.
- Serban, A.; Poll, E.; and Visser, J. 2020. Towards Using Probabilistic Models to Design Software Systems with Inherent Uncertainty. In *Software Architecture - 14th European Conference*, volume 12292 of *Lecture Notes in Computer Science*, 89–97. Springer.
- Silberman, N.; Hoiem, D.; Kohli, P.; and Fergus, R. 2012. Indoor Segmentation and Support Inference from RGBD Images. In *European Conference on Computer Vision*.
- Snoek, J.; Rippel, O.; Swersky, K.; Kiros, R.; Satish, N.; Sundaram, N.; Patwary, M. M. A.; Prabhat; and Adams, R. P. 2015. Scalable Bayesian Optimization Using Deep Neural Networks. In *International Conference on Machine Learning*, volume 37 of *JMLR Workshop and Conference Proceedings*, 2171–2180.
- Tagasovska, N.; and Lopez-Paz, D. 2019. Single-Model Uncertainties for Deep Learning. *Advances in Neural Information Processing Systems*, 32: 6417–6428.
- Tak, H.; Ellis, J. A.; and Ghosh, S. K. 2019. Robust and Accurate Inference via a Mixture of Gaussian and Student’s t Errors. *Journal of Computational and Graphical Statistics*, 28: 415 – 426.
- Van Amersfoort, J.; Smith, L.; Teh, Y. W.; and Gal, Y. 2020. Uncertainty estimation using a single deep deterministic neural network. In *International Conference on Machine Learning*, 9690–9700.
- Wen, Y.; Tran, D.; and Ba, J. 2020. Batchensemble: an alternative approach to efficient ensemble and lifelong learning. In *International Conference on Learning Representations*.
- West, M. 1984. Outlier models and prior distributions in Bayesian linear regression. *Journal of the Royal Statistical Society: Series B (Methodological)*, 46(3): 431–439.
- Xia, X.; Liu, T.; Wang, N.; Han, B.; Gong, C.; Niu, G.; and Sugiyama, M. 2019. Are anchor points really indispensable in label-noise learning? *Advances in Neural Information Processing Systems*, 32: 6838–6849.