Approaching Neural Network Uncertainty Realism

Joachim Sicking[1,2*], **Alexander Kister**[1,2*], **Matthias Fahrland**[3*], **Stefan Eickeler**[1,2], **Fabian Hüger**[4], **Stefan Rüping**[1], **Peter Schlicht**[4], **Tim Wirtz**[1,2]
[1] Fraunhofer IAIS
[2] Fraunhofer Center for Machine Learning
[3] IAV GmbH
[4] Volkswagen Group Innovation
{joachim.sicking, alexander.kister, stefan.eickeler, stefan.rueping, tim.wirtz}@iais.fraunhofer.de, matthias.fahrland@iav.de, {fabian.hueger, peter.schlicht}@volkswagen.de

Abstract

Statistical models are inherently uncertain. Quantifying or at least upper-bounding their uncertainties is vital for safety-critical systems such as autonomous vehicles. While standard neural networks do not report this information, several approaches exist to integrate uncertainty estimates into them. Assessing the quality of these uncertainty estimates is not straightforward, as no direct ground truth labels are available. Instead, implicit statistical assessments are required. For regression, we propose to evaluate *uncertainty realism*—a strict quality criterion—with a Mahalanobis distance-based statistical test. An empirical evaluation reveals the need for uncertainty measures that are appropriate to upper-bound heavy-tailed empirical errors. Alongside, we transfer the variational U-Net classification architecture to standard supervised image-to-image tasks. We adopt it to the automotive domain and show that it significantly improves uncertainty realism compared to a plain encoder-decoder model.

1 Introduction

Having attracted great attention in both academia and digital economy, deep neural networks (DNNs) [1] are about to become vital components of safety-critical mobility applications, particularly as core elements in automated driving systems [2, 3, 4, 5]. These systems range from special applications such as automated car parking and on-premise navigation over highway pilots to fully automated driving and promise more efficient and safer mobility. To actually deliver on this promise, the considerable potential of such cyber-physical systems to harm humans and to cause severe damages has to be minimized. This fact comes with new challenges for the development of DNNs: next to the performance itself, further requirements such as low latency and high robustness gain in importance. Furthermore, safety-critical systems do not tolerate failures and hence have to be monitored and assessed at runtime to ensure safe functioning.

One such mean of understanding the state of a software system is measuring the statistical uncertainty of the system module given the current input. Quantifying such uncertainties helps to make decisions especially in situations of partial availability of the relevant informa-

[*] indicates equal contribution.

Machine Learning for Autonomous Driving Workshop at the 33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

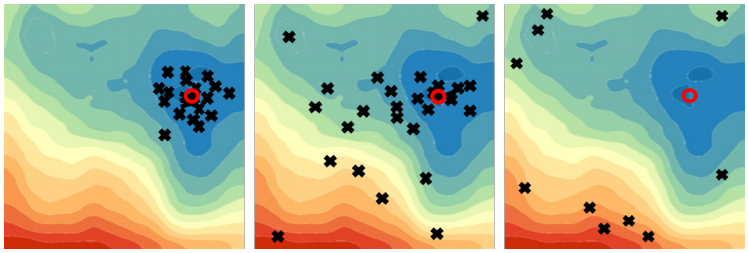

Figure 1: Symbolic image illustrating uncertainty realism. The estimated uncertainties (contour lines) are too broad (left), realistic (middle), too narrow (right) compared to the deviations of model outputs (black crosses) from ground truth (red circle).

tion. In particular in modular systems, subsequent modules should profit from the addition of knowledge about the uncertainty within the processing of the current input.

Amongst others, Monte Carlo (MC) dropout and variational inference are promising approaches to estimate the prediction uncertainty of DNNs (see section 2 for more details). These approaches try to estimate more realistically the statistical uncertainties of DNNs that go beyond computing dispersion metrics on the DNN's softmax output which is known to be rather easily fooled by adversarial perturbations [6].

For uncertainty estimates to be used in safety-critical systems, we require them to be realistic [7] (Figure 1), i.e. we require these estimates to resemble the residuals (the fitting errors) of the neural network outcomes. This poses a conceptual challenge as standard optimization schemes do not allow for direct training of realistic uncertainties. Therefore, high predictive performance and uncertainty realism might be largely unrelated to one another. It is desirable to achieve these two objectives at the same time.

For regression, we put forward an evaluation scheme to test for uncertainty realism. For classification, we argue that existing assessment methods already (partly) satisfy these realism demands. Instead we propose a novel approach to variational inference that provides more realistic uncertainty estimates compared to existing approaches.

In detail, our contribution is as follows:

- We propose a statistical test that allows to evaluate the realism for uncertainty mechanisms in regression tasks. These test outcomes are empirically analyzed for a 4D regression task in the vision domain (object detection with SqueezeDet and MC dropout). Further analyses call for uncertainty measures that are appropriate to upper-bound heavy-tailed empirical errors.
- We introduce a probabilistic U-Net-like FRRN semantic segmentation network and systematically assess the realism of the two uncertainty mechanisms it naturally provides. We find probabilistic FRRN (full-resolution residual network) to significantly improve uncertainty realism compared to (plain) FRRN.

2 Related work

In literature, it is common to distinguish uncertainties by their source: data-inherent uncertainty is called *aleatoric uncertainty* (e.g. rolling a dice) and uncertainty due to an limited amount of training data is called *epistemic uncertainty* [8, 9, 10, 11]. This paper focuses on epistemic uncertainty.

An approach that takes epistemic uncertainty into account are Bayesian neural networks (BNNs) [12] which represent the weights by random variables. Learning a BNN requires to calculate conditional distributions of the weights given the data. Already in early applications, approximate solutions are used [12, 13]. Many recent applications of BNNs to deep architectures are based on variational approximations to Bayesian inference [14] (MC dropout [15], early stopping [16], weight decay [17]). For our experiments, we apply MC

dropout to a regression problem in the detection setting. MC dropout was already used for different regression tasks [8] and also in the detection setting [18].

Bayesian frameworks are also used in other settings than BNNs: the *Probabilistic U-Net* [19], a model for segmentation, uses a neural Network as a feature extractor, the weights of which are not treated as random variables. To obtain a distribution over possible segmentations, it is assumed that there are hidden random variables that generate a distribution over the output space. The probabilistic U-Net was introduced to model an aleatoric uncertainty, namely the label ambiguity caused by different ground truth segmentations. The approximations mentioned above both represent the output by a sample from the approximate posterior distribution. The approximations make it necessary to assess the quality of the uncertainty measures.

One way to assess the quality of an uncertainty estimate is critically studying the algorithm itself. For example [20] points out that MC dropout [15]—unlike a corresponding BNN—does not converge to concentrated distributions in the infinite data limit (see also [21]) or [22] shows that variational approaches tend to underestimate the variance. An alternative way is to treat the uncertainty mechanism as a black box and assess how well the estimated uncertainties fit to the reality. Lacking a ground truth label for the uncertainty, this comparison has to be indirect [23]. The assessment approaches differ between classification and regression tasks.

Classification For the classification task, it is typical to determine an uncertainty score (either directly on the network output [15, 24] or by an additional model [25]) and to raise a flag if this score surpasses a threshold. Given an approximation to a posterior distribution (or a sample from it) it is possible to derive **scores** based on the expected softmax and higher moments. Typical such scores on the expected softmax are the entropy [26] or the maximum [24]. Scores based on the second moment are the variance (of the winning class in sampling or of the expected softmax) and mutual information (MI) [26].

To make evaluation of the above scores independent of a threshold value, areas under the receiver operating characteristic (AUROC), the precision-recall curve (AUPRC) or the negative predictive value versus the recall curve are considered [24, 26]. Alternatively, one can assess the calibration quality of the resulting softmax [23, 27, 21].

Regression For regression tasks, it is suitable to represent the uncertainty of an approximation to the posterior distribution (or of an sample from it) by its (empirical) covariance matrix. A very strict way to assess this matrix is the application of a suitable statistical test, as it is in astronautics (*covariance realism*, see [7] for details). A less strict metric is a multidimensional extension of the standardized mean-squared error (SMSE) [28] that was used in [29] to assess a one-dimensional regression model based on MC dropout (this work also covers other methods than MC dropout). More frequently, models are assessed by the average log-likelihood [30, 31, 15].

The methods described above assess the complete covariance matrix. For further assessments this matrix is reduced to a **score** measuring a certain aspect of the corresponding ellipsoid. Plausible scalar **scores** include the determinant, the maximal eigenvalue or the maximal diagonal entry of the matrix. A completely different method of uncertainty estimate assessment is investigating its behavior. For example [8] checks if the epistemic uncertainty decreases when increasing the training set or [32] studies how the level of uncertainty depends on the distance of the object to a car for some 3D environment regression task.

3 Towards more realistic neural network uncertainties

An uncertainty mechanism should be realistic. To assess its realism we need suitable metrics. The metrics should help us to decide if—or to which degree—the uncertainty reflects the actual statistical weaknesses of the model. Earlier attempts to uncertainty realism can be found in [26]. For regression tasks, we derive a mathematical criterion based on Mahalanobis distances. For classification, existing assessment methods already allow to judge how well uncertainties point out the weaknesses of the model. Here, we propose a novel approach to variational inference that provides more realistic uncertainty estimates compared to existing approaches.

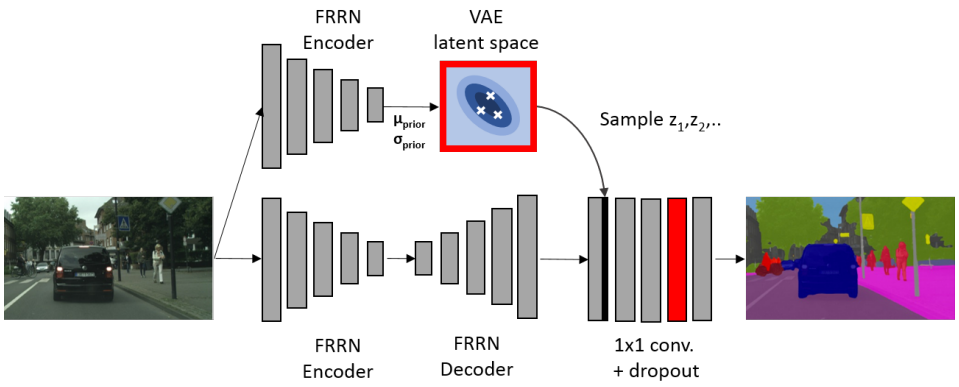

Figure 2: Architecture of the proposed variational FRRN (at inference). The two uncertainty mechanisms *latent space sampling* and *MC dropout* are highlighted in red.

Regression

For regression tasks, uncertainty mechanisms like MC dropout produce an output sample instead of a single output. The sample covariance matrix is a measure for the network uncertainty. In other words we fit a Gaussian to the sample of predictions and call this Gaussian the posterior predictive distribution. There are multiple ways to assess how good this posterior predictive distribution fits to test data. The most common way is to calculate the negative log-likelihood of the test data. However, test log-likelihood is a *performance evaluation metric* and thus not ideally suited for analyzing the quality of the estimated variances as it is biased towards well-performing models. Instead, it is preferable to *assess the variances separately* and without a performance bias (see appendix A.1 for a more detailed discussion). We propose to do this by means of a statistical test:

For conceptual clarity, we consider a simple 1D regression task for the moment. A neural network with MC dropout is trained to predict $y_{i,gt}$ from x_i. At test time, the network generates—for each input x_i—a sample $\{y_{i,k}\}$ that is summarized as (μ_i, σ_i). The network residual is $\xi_i = y_{i,gt} - \mu_i$. Here, uncertainty realism boils down to requiring: $\xi_i \sim \mathcal{N}(0, \sigma_i)$. As only one ξ_i is available for a given x_i, rewriting this criterion as

$$\frac{\xi_i^2}{\sigma_i^2} = \frac{(y_{i,gt} - \mu_i)^2}{\sigma_i^2} \sim \chi^2(d = 1) \tag{1}$$

allows to actually test it—independent of a specific dataset. Note that not only unrealistic uncertainties but also regression bias could lead to violations of this condition.

Generalizing this statistical criterion from 1D to a higher dimensional regression task, the covariance realism can be assessed using the squared Mahalanobis distance [7]:

$$M^2_{\mu_i, \mathbf{\Sigma}_i}(\mathbf{y}) = (\mathbf{y} - \boldsymbol{\mu}_i)^T \, \mathbf{\Sigma}_i^{-1} (\mathbf{y} - \boldsymbol{\mu}_i) \ ,$$

where $\boldsymbol{\mu}_i$ and $\mathbf{\Sigma}_i$ are derived from the uncertainty mechanism for data point $\mathbf{x}_i$. If $\boldsymbol{y}$ is sampled from a Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\mathbf{\Sigma}$, $M^2_{\boldsymbol{\mu}, \mathbf{\Sigma}}(\boldsymbol{y})$ follows a chi-squared distribution $\chi^2(d)$, where d is the dimension of $\boldsymbol{y}$. Note that this statement holds for every choice of $\boldsymbol{\mu}$ and $\mathbf{\Sigma}$ and hence also in our setting, where each observation $\boldsymbol{y}_{i,gt}$ is assumed to be a realization of a Gaussian with different parameters $\{\boldsymbol{\mu}_i, \mathbf{\Sigma}_i\}$. Hence, if the estimates $\boldsymbol{\mu}_i$ and $\mathbf{\Sigma}_i$ are realistic, a strict uncertainty realism criterion is given by requiring the set $\mathcal{M}_{\text{gt}} = \{M^2_{\boldsymbol{\mu}_i, \mathbf{\Sigma}_i}(\boldsymbol{y}_{i,gt})\}$ to follow a $\chi^2(d)$-distribution.

Classification

For classification tasks, not only the study of softmax variances but also the study of the mean softmax is interesting because it provides (inter-class) uncertainty information. This is in contrast to regression tasks. To treat these different uncertainty sources in a uniform way, we derive scalar scores from them, e.g. entropy or variance.

The practical use of such an uncertainty score is to identify inputs for which the model is wrong. The better we are in making this decision, the more realistic is the uncertainty score. This realism is quantified by AUC values.

A standard uncertainty mechanism for classification tasks is MC dropout [15] as an approximation of a BNN. However, it makes use of an uninformed prior over the weights. We propose an alternative mechanism by studying variational inference-based uncertainty inspired by the probabilistic U-Net [19]. In contrast, our approach of using the variational architecture (Figure 2) is to focus less on creating different interpretations of an input image, but to treat the latent space as a more sophisticated and focused mechanism to encode data-inherent uncertainty. Furthermore, instead of using a U-Net as the base structure for the segmentation task and the encoders, we use an FRRN-based [33] architecture (see appendix A for details). Additionally, we allow to combine the latent space sampling with MC dropout by inserting a dropout layer right before the last convolution of the network. To the best of our knowledge, this architecture yields the first DNN that allows for evaluation and fusion of the two mentioned uncertainty mechanisms.

4 Empirical evaluation

Following the outlined assessment scheme for neural network uncertainties, we evaluate their realism for the regression task of object detection. Moreover, we analyze the realism of the two uncertainty mechanisms of the novel variational FRRN for semantic segmentation.

4.1 Regression - object detection

We consider SqueezeDet [34], a lightweight single-stage detector network, that uses a pre-trained SqueezeNet as its backbone. It is trained on KITTI [35] and returns a four-tuple ($d = 4$) of center coordinates, width and height of a 2D bounding box for detected objects. The uncertainty mechanism we use is MC dropout ($p = 0.5$) before the last convolutional layer. The MC sampling of the dropout layer is done before the thresholding and non-maximum suppression stage. The output sample $\{\boldsymbol{y}_{i,k}\}$ for a given input $\boldsymbol{x}_i$ with ground truth $\boldsymbol{y}_{i,gt}$ is characterized by its mean $\boldsymbol{\mu}_{\boldsymbol{y}}$ and its covariance $\boldsymbol{\Sigma}_{\boldsymbol{y}}$. Following section 3, we calculate the squared Mahalanobis distance $M^2_{\boldsymbol{\mu}_i,\boldsymbol{\Sigma}_i}(\boldsymbol{y}_{i,gt})$. $\mathcal{M}_{\text{gt}} = \{M^2_{\boldsymbol{\mu}_i,\boldsymbol{\Sigma}_i}(\boldsymbol{y}_{i,gt})\}$ denotes the set of squared Mahalanobis distances for the entire test set $\{\boldsymbol{x}_i, \boldsymbol{y}_{i,gt}\}$. To apply the strict uncertainty realism criterion, we test if this set is drawn from $\chi^2(d = 4)$. This yields a p-value close to zero and therefore we have to reject the hypothesis and find the uncertainty mechanism to be unrealistic: MC dropout does not provide realistic uncertainty estimates in this empirical setting. In contrast, the set of intra-sample squared Mahalanobis distances $\mathcal{M}_{\text{sample}} = \{M^2_{\boldsymbol{\mu}_i,\boldsymbol{\Sigma}_i}(\boldsymbol{y}_{i,k})\}$ is—in accordance with theory— χ^2-distributed. A comparison of these distributions is shown in Figure 3 (left panel, for per-component visualizations see Figure 8 in the appendix). The width of the MC sample distribution is one order of magnitude smaller than the actual estimation errors - highlighting that variational approaches underestimate variances [22]. Handling this deviation in an ad-hoc fashion by scaling the variance of $\mathcal{M}_{\text{sample}}$ to match the variance of $\mathcal{M}_{\text{gt}}$ is shown in Figure 3 (middle panel). The higher moments of $\mathcal{M}_{\text{gt}}$ are still deviating and cause a decay that is slower than exponential, i.e. a fat tail.

While Mahalanobis distances allow for a combined assessment of the realism of the size and orientations of the covariance ellipsoid, an isolated assessment of the covariance orientation can be done by analyzing the angle $\alpha = \measuredangle(\boldsymbol{v}^{\max}_{\boldsymbol{\Sigma}}, \boldsymbol{\mu}_{\boldsymbol{y}} - \boldsymbol{y}_{gt})$ enclosed by the largest eigenvector of the covariance $\boldsymbol{v}^{\max}_{\boldsymbol{\Sigma}}$ and the estimation error $\boldsymbol{\mu}_{\boldsymbol{y}} - \boldsymbol{y}_{gt}$ (Figure 3, right panel). The high resemblance of the angle distribution with the distribution of the differential solid angle of the 3-sphere emphasizes that no covariance orientation realism is given.

Following section 3, we check if MC dropout is—at least—a good indicator for realism, i.e. if the mean estimation error increases monotonically with the uncertainty score $\boldsymbol{\Sigma}_{\boldsymbol{y}}$. Here, we approximate the covariance with its determinate $\det(\boldsymbol{\Sigma}_{\boldsymbol{y}})$ and its largest component $\Sigma^{\max}$ as rough scalar measures of its size. Figure 4 (left and right panel) suggests that such a monotonicity is given. Due to the discussed fat-tails, however, upper-bounding

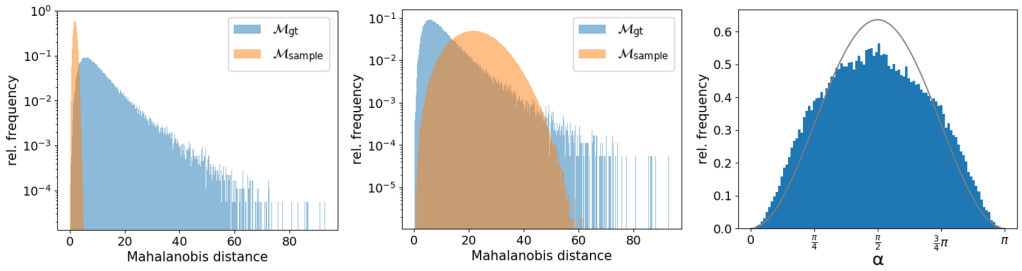

Figure 3: Assessment of uncertainty realism for the regression task. Left: empirical distributions $\mathcal{M}_{\text{sample}}$ and $\mathcal{M}_{\text{gt}}$, middle: rescaled $\mathcal{M}_{\text{sample}}$ to match the variance of $\mathcal{M}_{\text{gt}}$, right: distribution of angles α enclosed by the error direction and the covariance ellipsoid orientation as well as the differential solid angle of the 3-sphere.

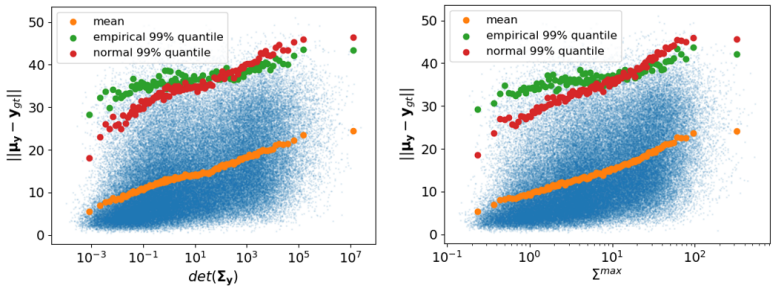

Figure 4: Uncertainty scores and estimation errors (left: covariance determinant, right: largest covariance element).

estimation errors by uncertainty estimates cannot be done using multiples of the standard deviation. The empirical 99% quantiles (Figure 4, green) are underestimated by the—in case of normality—corresponding 2.576σ-interval (red), partly by far. Therefore, residual risks should be quantified using appropriate tail measures such as quantiles or tail mean values.

4.2 Classification - semantic segmentation

The proposed variational FRRN (Figure 2, appendix A) is trained for 3×10^5 steps on the Cityscapes dataset [36] with a batch size of 4 and a drop rate of $p = 0.5$. The experiments are conducted on the images of the city "Münster". In order to compare the three uncertainty mechanisms that the variational FRRN provides, each test set image is processed three times: with only MC dropout switched-on (MC), with only latent space sampling switched-on (CVAE) and with both mechanisms switched-on (CVAE + MC). Regardless of the chosen uncertainty mechanism, the number of samples is fixed to 50. As uncertainty scores we consider the highest probability and the entropy of the mean softmax, the variance of the winning class and the MI (see section 2). To evaluate the realism of the mechanisms not only within the training data distribution but also out-of-distribution, images are vertically flipped and again processed three times as described above. The uncertainty realism is measured by calculating the respective areas under the ROC (AUROC) and the precision-recall curve (AUPRC) where we take the correctly classified pixels as positive. As a baseline we use the plain softmax output without any uncertainty mechanism being switched-on.

Table 1 shows that for in-data samples the plain softmax probability is a reasonably accurate uncertainty estimation and cannot be outperformed by the advanced mechanisms. This may be related to the high accuracy of the network because good predictions require comparably less challenging uncertainty estimations. An example for an in-data prediction and the corresponding estimated uncertainty can be seen in Figure 9 in the appendix.

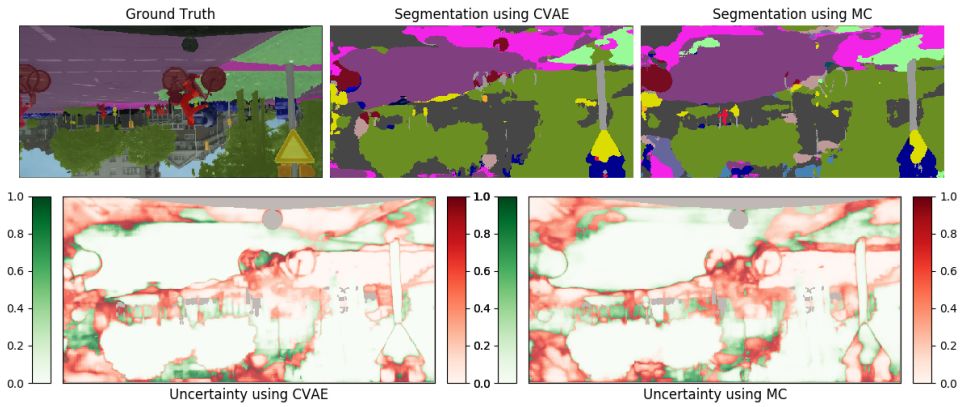

Figure 5: Segmentations and uncertainty maps produced by variational FRRN. Segmentations: ground truth (top left), FRRN with CVAE sampling (top middle), FRRN with MC dropout (top right). Uncertainty maps: FFRN + CVAE sampling (bottom left), FRRN + MC dropout (bottom right). Uncertainties of correctly classified pixels are shown in green, of misclassified pixels in red.

Table 1: Realism assessment for pixel-wise classification (in-sample and out-of-sample). Compared mechanisms are the softmax baseline (none), MC sampling (MC), latent space sampling (CVAE) and a combination of the latter (CVAE + MC).

mechanism	**score**	**assessment**			
		in-sample		out-of-sample	
		AUROC	AUPRC	AUROC	AUPRC
none	max. comp. softmax	**0.945**	**0.997**	0.642	0.643
none	entropy	0.944	0.997	0.639	0.640
	mean-based scores $f(\boldsymbol{\mu}_{\boldsymbol{y}})$				
MC	max. comp. softmax	0.945	0.997	0.651	0.653
MC	entropy	0.943	0.997	0.657	0.655
MC	MI	0.943	0.997	0.654	0.655
CVAE	max. comp. softmax	0.944	0.997	0.718	**0.755**
CVAE	entropy	0.944	0.977	**0.720**	0.751
CVAE	MI	0.926	0.995	0.717	0.747
CVAE + MC	max. comp. softmax	0.946	0.997	0.713	0.742
CVAE + MC	entropy	0.944	0.997	0.714	0.742
CVAE + MC	MI	0.919	0.996	0.706	0.735
	covariance-based scores $f(\boldsymbol{\Sigma}_{\boldsymbol{y}})$				
MC	variance of max. comp. softmax	0.934	0.996	0.657	0.655
CVAE	variance of max. comp. softmax	0.936	0.997	0.718	0.753
CVAE + MC	variance of max. comp. softmax	0.914	0.995	0.702	0.733

For out-of-data images, however, the sampling mechanisms allow a significant improvement. Sampling with the variational FRRN yields the most realistic uncertainties with an AUROC value of 0.72 when using the entropy score and an AUPRC of 0.755 when using the mean softmax probability. The influence of the different scores on the assessment is generally small. Fusing latent space sampling with MC dropout does not lead to a further improvement.

Figure 5 depicts the predictions and uncertainty estimations of the network given an out-of-data input image. As expected, the network fails to correctly classify the majority of the pixels. However, both applied mechanisms allow to detect a reasonable amount of the occurring misclassifications and assign a low uncertainty to regions with many true positives.

Figure 6 shows the ROC and precision-recall curves for the different mechanisms and the entropy score. It can be seen that the general shape is similar for all mechanisms. Further results for using a ROC curve-based thresholding to reject predictions with high chance of being wrong can be found in the appendix (Figure 10). Comparing the mechanisms qualitatively (Figure 5) and empirically (Figure 6, Table 1), we see that MC dropout and

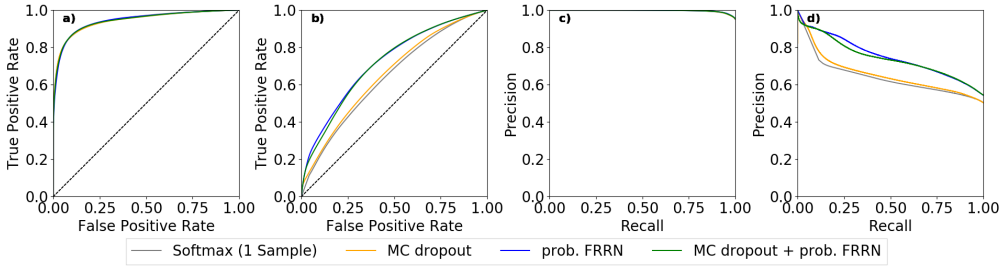

Figure 6: In-sample and out-of-sample comparison of different uncertainty mechanisms. a) and b) show in-sample and out-of-sample ROC curves, c) and d) the equivalent precision-recall curves. Class entropy is used as uncertainty score.

the latent space sampling seem to express the same kind of uncertainty. Both mechanisms generate better probability estimations for out-of-data samples compared to plain softmax.

At a first glance, the similarity between the two mechanisms is surprising because they function in different ways. However, both are based on the principles of variational inference, thus approximate the posterior by sampling from the model parameters. Fusing both mechanisms leads to a different sample of parameters, but from the same distribution, which may be why we see no further improvements in the uncertainty estimate.

5 Discussion and future work

We argue to evaluate uncertainty realism in regression tasks with a Mahalanobis distance-based statistical test. We improve uncertainty realism in a classification task by using the proposed variational FRRN instead of a (plain) FRRN.

Regression For the considered regression task in the vision domain, MC dropout does not provide realistic uncertainty estimates as it does not fit the orientation, width or non-normality of the estimation error distribution. However, it is important to highlight that our experimental setting adds further approximations to the approximations underlying MC dropout (small network, dropout applied only to one layer). This setup is commonly used though and we emphasize to carefully validate MC dropout results for each safety-critical application. While not being realistic, we find it to be a good indicator for realism. The observed non-normality of the error distribution indicates that naive uncertainty upper-bounds on regression errors might fall short. Instead, we recommend the usage of appropriate uncertainty measures such as quantiles or tail mean values.

Classification When comparing the MC dropout with the latent space sampling technique, it seems that both approximate the same form of uncertainty. While small scale differences appear in examplary images, the overall estimation focuses on the same aspects and empirical evaluations show similar tendencies. Both mechanisms allow to reason about epistemic uncertainties. We state that the similarities originate from the mutual base idea of estimating the models uncertainty by approximating the posterior distribution. Compared to MC dropout, the latent space sampling seems to be the more informed uncertainty mechanism.

Future work Uncertainty realism might be improved by using multiple dropouts in the MC setting, adjusting loss functions and other training hyper parameters or using thresholds per class for the score. Studying other out-of-data directions (e.g. lighting conditions, fog and unknown classes) and tasks from other domains could strengthen our results. A shortcoming of probabilistic U-Net is that the latent space sample is only used as an additional channel which the network can choose to ignore. PHiSeg [37] is an extension of the probabilistic U-Net that addresses this. Transferring the PHiSeg approach to our setting and studying the resulting uncertainties seems promising. Important further work is to prove residual risk bounds and to formalize requirements for uncertainty estimates of neural networks. Relating uncertainty estimates to adversarial robustness needs further research.

Acknowledgements

This research has been partly funded (S.R.) by the German Federal Ministry of Education and Research, ML2R - grant no. 01S18038B. We thank the anonymous reviewers for their feedback and Maximilian Pintz and Maram Akila for fruitful discussions.

References

[1] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[2] Dean A Pomerleau. Alvinn: An autonomous land vehicle in a neural network. In *Advances in neural information processing systems*, pages 305–313, 1989.

[3] Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, et al. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016.

[4] Shai Shalev-Shwartz, Shaked Shammah, and Amnon Shashua. Safe, multi-agent, reinforcement learning for autonomous driving. *arXiv preprint arXiv:1610.03295*, 2016.

[5] Wenyuan Zeng, Wenjie Luo, Simon Suo, Abbas Sadat, Bin Yang, Sergio Casas, and Raquel Urtasun. End-to-end interpretable neural motion planner. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8660–8669, 2019.

[6] Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for uncrecognizable images. In *Proceedings of the IEEE converence on computer vision and pattern recognition*, pages 427–436, 2015.

[7] Joshua T Horwood, Jeffrey M Aristoff, Navraj Singh, Aubrey B Poore, and Matthew D Hejduk. Beyond covariance realism: a new metric for uncertainty realism. In *Signal and Data Processing of Small Targets 2014*, volume 9092, page 90920F. International Society for Optics and Photonics, 2014.

[8] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in neural information processing systems*, pages 5574–5584, 2017.

[9] Armen Der Kiureghian and Ove Ditlevsen. Aleatory or epistemic? does it matter? *Structural Safety*, 31(2):105–112, 2009.

[10] Robin Senge, Stefan Bösner, Krzysztof Dembczyński, Jörg Haasenritter, Oliver Hirsch, Norbert Donner-Banzhoff, and Eyke Hüllermeier. Reliable classification: Learning classifiers that distinguish aleatoric and epistemic uncertainty. *Information Sciences*, 255:16–29, 2014.

[11] Stefan Depeweg, José Miguel Hernández-Lobato, Finale Doshi-Velez, and Steffen Udluft. Decomposition of uncertainty in bayesian deep learning for efficient and risk-sensitive learning. *arXiv preprint arXiv:1710.07283*, 2017.

[12] David JC MacKay. A practical bayesian framework for backpropagation networks. *Neural computation*, 4(3):448–472, 1992.

[13] Radford M Neal. Bayesian learning via stochastic dynamics. In *Advances in neural information processing systems*, pages 475–482, 1993.

[14] Hagai Attias. A variational baysian framework for graphical models. In *Advances in neural information processing systems*, pages 209–215, 2000.

[15] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059, 2016.

[16] David Duvenaud, Dougal Maclaurin, and Ryan Adams. Early stopping as nonparametric variational inference. In *Artificial Intelligence and Statistics*, pages 1070–1077, 2016.

[17] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural networks. *arXiv preprint arXiv:1505.05424*, 2015.

[18] Dimity Miller, Lachlan Nicholson, Feras Dayoub, and Niko Sünderhauf. Dropout sampling for robust object detection in open-set conditions. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–7. IEEE, 2018.

[19] Simon Kohl, Bernardino Romera-Paredes, Clemens Meyer, Jeffrey De Fauw, Joseph R Ledsam, Klaus Maier-Hein, SM Ali Eslami, Danilo Jimenez Rezende, and Olaf Ronneberger. A probabilistic u-net for segmentation of ambiguous images. In *Advances in Neural Information Processing Systems*, pages 6965–6975, 2018.

[20] Ian Osband. Risk versus uncertainty in deep learning: Bayes, bootstrap and the dangers of dropout. In *NIPS Workshop on Bayesian Deep Learning*, 2016.

[21] Yarin Gal, Jiri Hron, and Alex Kendall. Concrete Dropout. In *Advances in Neural Information Processing Systems 30 (NIPS)*, 2017.

[22] Christopher M Bishop. *Pattern recognition and machine learning*. Springer, 2006.

[23] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, pages 6402–6413, 2017.

[24] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *CoRR*, abs/1610.02136, 2017.

[25] Ramon Oliveira, Pedro Tabacof, and Eduardo Valle. Known unknowns: Uncertainty quality in bayesian neural networks. *arXiv preprint arXiv:1612.01251*, 2016.

[26] Jishnu Mukhoti and Yarin Gal. Evaluating bayesian deep learning methods for semantic segmentation. *arXiv preprint arXiv:1811.12709*, 2018.

[27] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. *CoRR*, abs/1706.04599, 2017.

[28] Carl Edward Rasmussen. Gaussian processes in machine learning. In *Summer School on Machine Learning*, pages 63–71. Springer, 2003.

[29] Wolfgang Fruehwirt, Adam D Cobb, Martin Mairhofer, Leonard Weydemann, Heinrich Garn, Reinhold Schmidt, Thomas Benke, Peter Dal-Bianco, Gerhard Ransmayr, Markus Waser, et al. Bayesian deep neural networks for low-cost neurophysiological markers of alzheimer's disease severity. *arXiv preprint arXiv:1812.04994*, 2018.

[30] David M Blei, Michael I Jordan, et al. Variational inference for dirichlet process mixtures. *Bayesian analysis*, 1(1):121–143, 2006.

[31] Jacob Walker, Carl Doersch, Abhinav Gupta, and Martial Hebert. An uncertain future: Forecasting from static images using variational autoencoders. In *European Conference on Computer Vision*, pages 835–851. Springer, 2016.

[32] Sascha Wirges, Marcel Reith-Braun, Martin Lauer, and Christoph Stiller. Capturing object detection uncertainty in multi-layer grid maps. *arXiv preprint arXiv:1901.11284*, 2019.

[33] Tobias Pohlen, Alexander Hermans, Markus Mathias, and Bastian Leibe. Full-resolution residual networks for semantic segmentation in street scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4151–4160, 2017.

[34] Bichen Wu, Forrest Iandola, Peter H Jin, and Kurt Keutzer. Squeezedet: Unified, small, low power fully convolutional neural networks for real-time object detection for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 129–137, 2017.

[35] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[36] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[37] Christian F Baumgartner, Kerem C Tezcan, Krishna Chaitanya, Andreas M Hötker, Urs J Muehlematter, Khoschy Schawkat, Anton S Becker, Olivio Donati, and Ender Konukoglu. Phiseg: Capturing uncertainty in medical image segmentation. *arXiv preprint arXiv:1906.04045*, 2019.

[38] Volodymyr Kuleshov, Nathan Fenner, and Stefano Ermon. Accurate uncertainties for deep learning using calibrated regression. In *35th International Conference on Machine Learning, ICML 2018*, 2018.

A Appendices

A.1 Comparison between negative log-likelihood and the uncertainty-realism criterion

Negative log-likelihood (NLL) is a standard *performance measure* that determines the likelihood of a data point y_i being drawn from a Gaussian distribution with parameters μ_i and σ_i. It reads

$$\frac{\log \sigma_i^2}{2} + \frac{(y_i - \mu_i)^2}{2\sigma_i^2} + \text{const.} \tag{2}$$

and has bounded isolines (cp. Figure 7, top) due to the first term. Thus, only a perfect model with $\mu_i = y_i$ and $\sigma_i \to 0$ reaches the minimal NLL.

For regression tasks, we desire not only a good performance but also realistic uncertainties, $\sigma \sim |\mu - y|$, and therefore evaluate statistics of

$$\frac{(y_i - \mu_i)^2}{\sigma_i^2}. \tag{3}$$

These measures allow to determine the *quality of uncertainties* independent of the model performance, i.e. even low-quality predictions can have perfectly realistic uncertainties (see also discussion on sharpness vs. calibration in [38]). Technically, this is reflected by unbounded isolines of (3) (cp. Figure 7, bottom[2]).

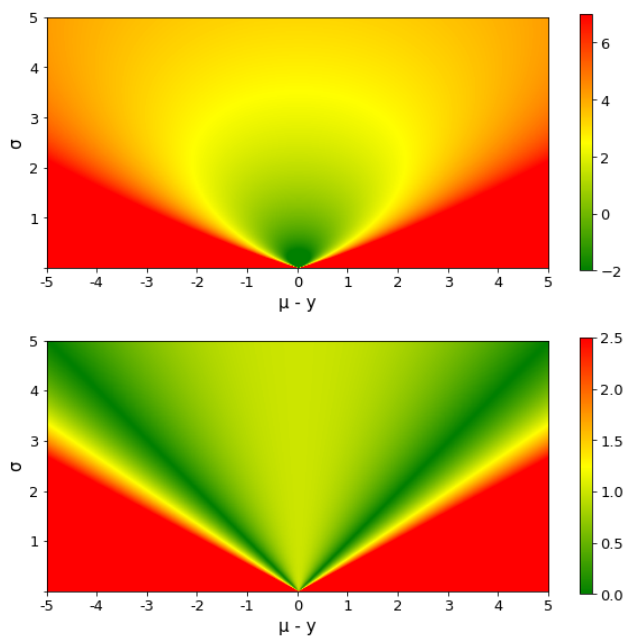

Figure 7: Isolines of negative log-likelihood (top) and of our uncertainty-realism criterion (bottom). Red (green) indicates high (low) values.

[2]In Figure 7, isolines of $|(y_i - \mu_i)^2/\sigma_i^2 - 1|$ are shown to visualize deviations from $\sigma \sim |\mu - y|$.

A.2 Extended evaluation of the uncertainty realism of SqueezeDet using MC dropout

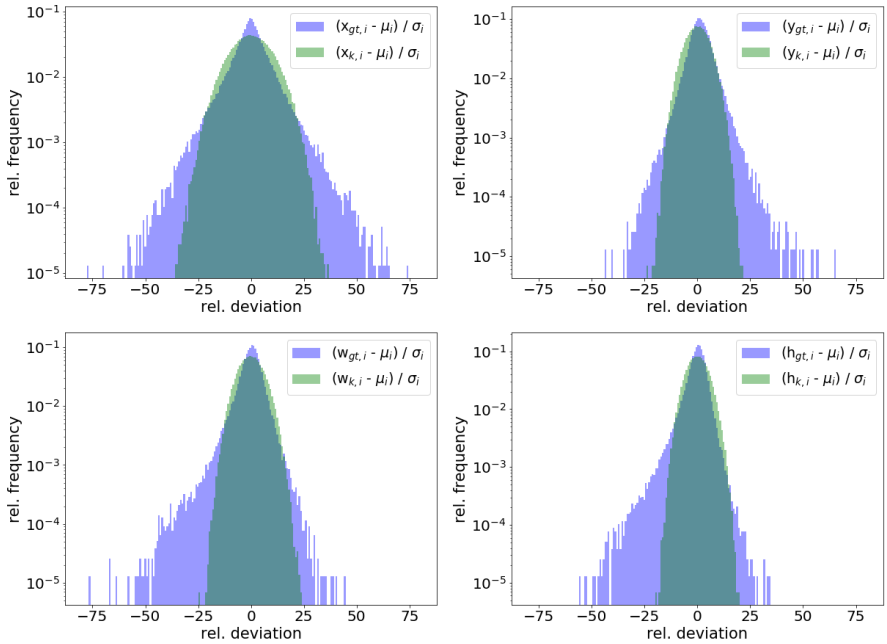

Figure 8: Assessment of uncertainty realism for the regression task. Distributions of scaled MC-dropout samples and estimation errors in x-direction (top left), y-direction (top right), for width (bottom left) and height (bottom right).

A.3 Architectural details of the base FRRN

In this section we describe the FRRN base architecture that we use for the experiments in section 4.2. Initially, the input to the FRRN gets fed into a convolutional layer with a 5 x 5-kernel, followed by three residual units consisting of two convolutions with a skip connection. The data flow then splits into a pooling stream and a residual stream. While the data in the pooling stream gets down- and up-sampled to extract features, the residual stream remains on full resolution, retaining the spatial information. The two streams get combined at each scale in so called full-resolution residual units (FRRUs) by pooling the residual stream to the other streams dimensions and merging it through concatenation and two convolutions. The last upsampling and stream concatenation is followed by another three residual units and a 1 x 1 convolution to classify the pixels. Finally a softmax layer is used to calculate the class probabilities.
We start with a base number of 24 channels and double, respectively half the number with each down- and up-sampling step. For dimension reduction or enlargement we use max pool and bilinear upscale operations. The residual stream is kept at consistently 16 channels, thus after a final concatenation we end with 40 feature maps that are used for the classification. If not stated otherwise all convolutions were run with 3 x 3-kernels and are followed by a ReLU activation.

A.4 Architectural details of the variational FRRN

Compared to the architecture of the probabilistic U-Net, we made various changes: As the base structure we use a full-resolution residual network (FRRN, see appendix A.3) and its feature extraction part and thus the downsampling operations alongside the full-resolution residual units for the variational encoders (Figure 2). To use the features of both the residual and the pooling stream for the probability estimation in the encoders, the final activations of the residual stream are pooled and concatenated with the pooling stream. The resulting final feature maps are global average pooled and a 1×1 convolution is used to predict a mean and variance. As proposed in [19] we use a dimension of six for the latent space and broadcast the drawn sample to the last activation map of the FRRN, followed by three 1×1 convolutions.

A.5 Additional results for the variational FRRN

A.5.1 Qualitative results

Figure 9 shows the predictions and corresponding uncertainty estimates of the variational FRRN. Especially class boundaries and distant objects are assigned a high uncertainty. Note that the selected input image is challenging: The fence on the left side of the image is see-through. While it is labelled as "fence", the network sees the objects behind it. The resulting misclassifications are assigned a high uncertainty.

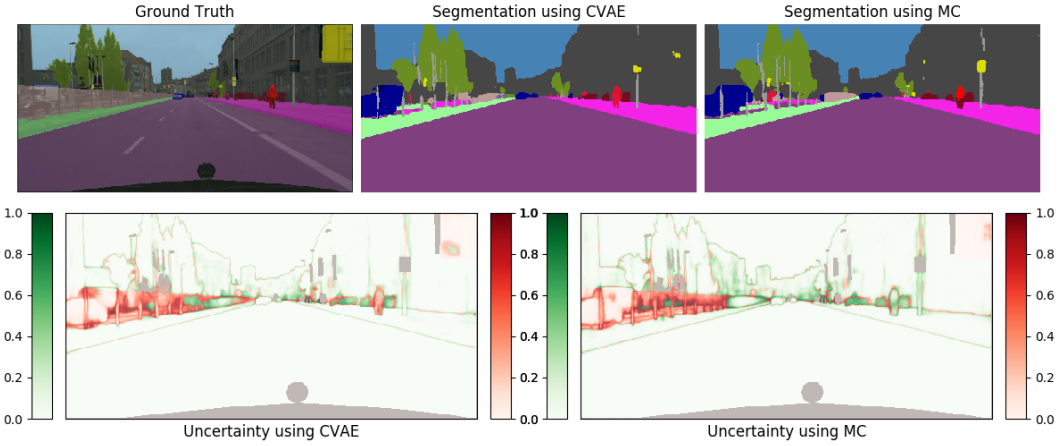

Figure 9: Uncertainty of correct and false classified pixels. Upper row: The flipped input image, overlayed with the Ground truth segmentation as well as the predictions based on the variational FRRN and MC dropout (50 samples each). The lower row shows the normalized, pixelwise uncertainty of correctly classified pixels (green) and misclassified pixels (red). Gray regions are not included in training and evaluation.

A.5.2 Quantitative results

The ROC curves allow to find a threshold that maximises the trade-off between the true positive and false negative rate. Finding this threshold and applying it to the uncertainty estimations allows to reject pixels that have a high chance of being misclassified. Figure 8 shows the amount of false negatives (FN) and true positives (TP) that were detected as being "misclassified".

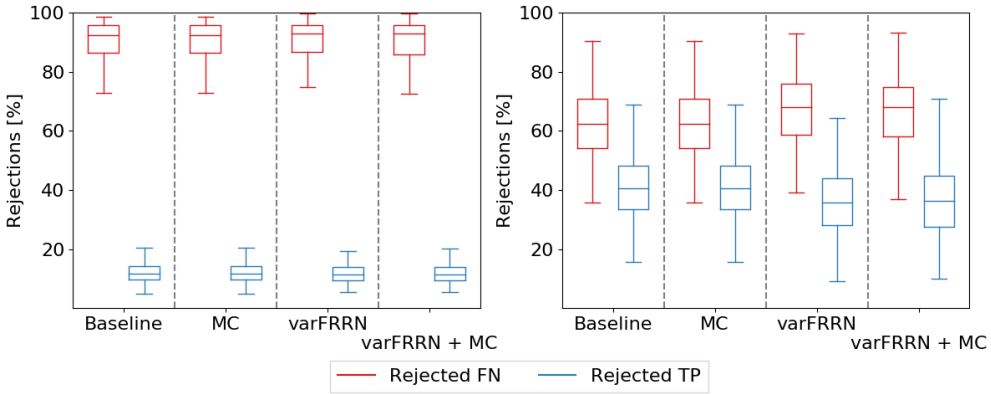

Figure 10: Rejection rates of FN and TP predictions with high uncertainty. The threshold is based on the ROC curve and the entropy score. Left: Rejections on in-data samples, right: on out-of-data samples.