A Multiclass Generalization for Uncertainties

In this section we show, how our method can be generalized from binary classification to multiclass problems. Consider data pairs $(X, Y) \sim \mathbb{P}$. Now, $X \in \mathbb{R}^d$ and $Y \in \{1, \ldots, C\}$, where C is the number of classes. We also denote $\eta_c(\mathbf{x}) = \mathbb{P}(Y = c \mid X = \mathbf{x})$.

Let us start with the Bayes risk:

$$\mathbb{P}(Y \neq g^*(X) \mid X = \mathbf{x}) = 1 - \mathbb{P}(Y = g^*(X) \mid X = \mathbf{x}) = 1 - \max_c \eta_c(\mathbf{x}) = \min_c \big\{1 - \eta_c(\mathbf{x})\big\},$$

where $g^*(\mathbf{x}) := \arg\max_c \eta_c(\mathbf{x})$ is the Bayes optimal classifier.

Let us further move to the excess risk and denote by $\hat{\eta}_c(\mathbf{x})$ some estimator of conditional probability. Analogously, $g(\mathbf{x}) := \arg\max_c \hat{\eta}_c(\mathbf{x})$ and we can bound the excess risk in the following way:

$$\begin{aligned}
&\mathbb{P}(Y \neq g(X) \mid X = \mathbf{x}) - \mathbb{P}(Y \neq g^*(X) \mid X = \mathbf{x}) = \eta_{g^*(\mathbf{x})}(\mathbf{x}) - \eta_{g(\mathbf{x})}(\mathbf{x}) \\
&= \eta_{g^*(\mathbf{x})}(\mathbf{x}) - \hat{\eta}_{g^*(\mathbf{x})}(\mathbf{x}) + \hat{\eta}_{g^*(\mathbf{x})}(\mathbf{x}) - \hat{\eta}_{g(\mathbf{x})}(\mathbf{x}) + \hat{\eta}_{g(\mathbf{x})}(\mathbf{x}) - \eta_{g(\mathbf{x})}(\mathbf{x}) \\
&\leq \big|\eta_{g^*(\mathbf{x})}(\mathbf{x}) - \hat{\eta}_{g^*(\mathbf{x})}(\mathbf{x})\big| + \big|\eta_{g(\mathbf{x})}(\mathbf{x}) - \hat{\eta}_{g(\mathbf{x})}(\mathbf{x})\big|,
\end{aligned}$$

where we used the fact that $\hat{\eta}_{g^*(\mathbf{x})}(\mathbf{x}) - \hat{\eta}_{g(\mathbf{x})}(\mathbf{x}) \leq 0$ for any $\mathbf{x}$.

The expectation of the right hand side in the case of kernel density estimator can be upper bounded by $2\sqrt{\frac{2}{\pi}}\tau(\mathbf{x})$, where

$$\tau^2(\mathbf{x}) = \frac{1}{N} \frac{\max_c \big\{\sigma_c^2(\mathbf{x})\big\}}{p(\mathbf{x})} \int \big[K_h(\mathbf{u})\big]^2 d\mathbf{u}$$

and $\sigma_c^2(\mathbf{x}) = \eta_c(\mathbf{x})\big(1 - \eta_c(\mathbf{x})\big)$. Total uncertainty for multiclass problem is thus

$$\mathbf{U}_t(\mathbf{x}) = \min_c \big\{1 - \eta_c(\mathbf{x})\big\} + 2\sqrt{\frac{2}{\pi}}\tau(\mathbf{x}).$$

B Architecture and Training Details for Image Datasets

Base Model. For CIFAR-100 and ImageNet-like datasets, we are using ResNet50 as a base model, with or without spectral normalization [54]. For the spectral normalization, we use 3 iterations of the power method. We use a ResNet50 architecture with implementation from PyTorch [62]. This architecture was implemented for the ImageNet dataset; thus, for the CIFAR-100, we had to adapt it. We changed the first convolutional layer and used kernel size 3x3 with stride 1 and padding 1 (instead of kernel size 7x7 with stride 2 and padding 3 used for ImageNet). For CIFAR-100, we train the model for 200 epochs with an SGD optimizer, starting with a learning rate of 0.1 and decaying it 5 times on 60, 120, and 160 epoch. For ImageNet, we train the model for 90 epochs with an SGD optimizer and learning rate decaying 10 times every 30 epochs.

For MNIST, we train a small convolutional neural network with three convolutional layers with padding of 1 and kernel size of 3. Each of these layers is followed by a batch normalization layer. Finally, it has a linear layer with Softmax activation. This network achieves an accuracy of 0.99 on the holdout set.

We refer readers to our code for more specific details.

Ensemble. For ensemble, we use a combination of 5 base models trained with different random seeds.

Test-Time Augmentation (TTA). For TTA, we use a base model and apply different transformations on the inference stage. Images of CIFAR-100 are randomly cropped with padding 4, randomly horizontally flipped, and randomly rotated up to 15 degrees. ImageNet is randomly cropped from 256 to 224, randomly horizontally flipped, and the color was jittered (0.02).

Spectrally Normalized Models. For SNGP, DDU and NUQ, we need spectral normalized models to extract features. We wrapped each convolutional and linear layer with spectral normalization (PyTorch implementation). We used 3 iterations of the power method in our experiments.

C Hyperparameters Selection Strategies

In this section, we collect all the hyperparameters one should choose to use NUQ, as well as we discuss strategies for choosing them.

C.1 Bandwidth

Probably the most important hyperparameter is bandwidth for kernels used in Nadaraya-Watson estimator. It is extensively studied in literature (see, for example, [45, 68] and references therein). However, there is still no silver bullet, and often the best approach for bandwidth selection is specific to a particular applied problem.

In this work we use cross-validation to estimate the accuracy of resulting Nadaraya-Watson based classifier, and tune the bandwidth to maximize it. We implicitly use an assumption, that the bandwidth, which is good for in-distribution classification, will be still good for out-of-distribution detection. Our experiments show, that this simple idea works well in practice.

C.2 Number of Nearest Neighbors

In this section, we demonstrate, how the number of nearest neighbors affects the results. For the sake of demonstration, we have chosen the most challenging dataset (ImageNet) and its out-of-distribution counterparts. It is important to note, that to make the experiment faster, we used only a subsample (75k) of the ImageNet dataset.

In Figure 5 we can see, that this hyperparameter has a wide interval, within which values do not affect the performance of the overall method. Already starting from 20 nearest neighbours the results are close to the optimal ones (both in terms of accuracy and out-of-distribution ROC AUC). This is due to the fact that typical kernels decay fast with distance, and there is no need to take a lot of nearest neighbors into account.

C.3 Marginal Density Estimation

To compute epistemic uncertainty according to equation (2), we need to estimate $p(\mathbf{x})$. In general, there are multiple possible ways how one can do it (including Variational Autoencoders [34] and Normalizing Flows [61]). However, for computational efficiency something lightweight is required. Specifically, we propose two ways. The first one is to use another KDE with the same kernel and bandwidth used for NW estimator. Another approach is to use training dataset embeddings and corresponding labels to assign a Gaussian distribution per class (GMM). Note, that it is not a Gaussian Mixture Model in the traditional sense, typically fitted with EM-algorithm. From our experiments (see Table 5) we see, that both of them perform on par, with some advantages of GMM for CIFAR-100 dataset, while for ImageNet KDE works better.

C.4 Kernel for KDE

In principle, one could choose different kernels for both, KDE (see Table 5) and NW-regression. But typically, as kernels decay fast with the distance, it does not affect much on the result. Thus, Gaussian (RBF) kernel is a good default choice.

Below, we study the choice of a kernel to plug-in in our approach. First, let us rewrite $K_h(\mathbf{u})$ as follows:

$$K_h(\mathbf{u}) = \prod_{i=1}^{d} K\left(\frac{u_i}{h}\right) = \prod_{i=1}^{d} K(z_i).$$

We consider the different choices of kernels, see Table 3.

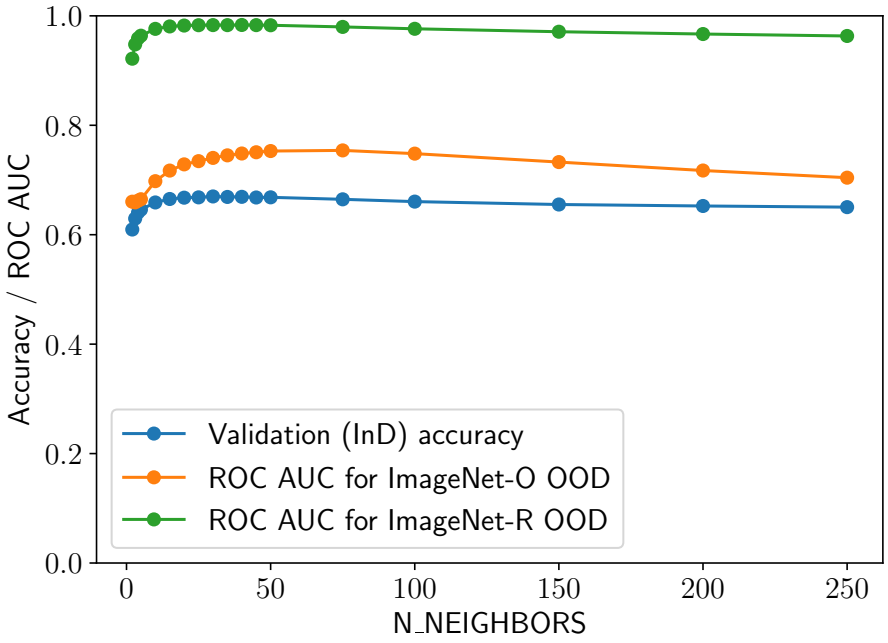


Figure 5: The dependence of ROC AUC for OOD detection and classification accuracy on the cross-validated training dataset as a function of the number of nearest neighbours used for approximation.

Kernel name	**Formula** $K(z)$	**Integral** $\int K_h(\mathbf{u})^2 d\mathbf{u}$
Gaussian (RBF)	$\frac{1}{\sqrt{2\pi}} \exp\left\{-z^2\right\}$	$\frac{h^d}{2\sqrt{\pi}}$
Sigmoid	$\frac{2}{\pi} \frac{1}{\exp\{-z\}+\exp\{z\}}$	$\frac{2h^d}{\pi^2}$
Logistic	$\frac{1}{\exp\{-z\}+2+\exp\{z\}}$	$\frac{h^d}{6}$

Table 3: Different types of kernels $K(z)$ considered and corresponding values of the integral $\int K_h(\mathbf{u})^2 d\mathbf{u}$.

D Consistency of NUQ-based Classification with the Reject Option

In this section, we derive a non-asymptotic upper bound of the excess risk used to obtain the consistency result in Section 4. First, using results of the previous section, notice, that for an arbitrary rejection rule $\alpha(\mathbf{x})$ the excess risk of $\mathcal{R}_\lambda(\mathbf{x})$ is at most

$$\mathbb{E}_{\mathcal{D}}\left\{\mathcal{R}_\lambda(\mathbf{x}) - \mathcal{R}^*_\lambda(\mathbf{x})\right\} \leqslant 2\mathbb{E}_{\mathcal{D}}\left\{\max_{c\in C} |\eta_c(\mathbf{x}) - \hat{\eta}_c(\mathbf{x})| \mathbb{1}\{\alpha(\mathbf{x}) = 0\}\right\} + |\lambda - \mathcal{R}^*(\mathbf{x})| \mathbb{P}_{\mathcal{D}}\left(\alpha(\mathbf{x}) \neq \alpha^*(\mathbf{x})\right).$$

As previously,

$$\frac{\eta_c(\mathbf{x}) - \hat{\eta}_c(\mathbf{x})}{\tau_c(\mathbf{x})} \to \mathcal{N}(0,1), \quad \tau_c(\mathbf{x}) = \|K\|_2 \sqrt{\frac{\eta_c(\mathbf{x})(1-\eta_c(\mathbf{x})}{Nh^d p(\mathbf{x})}}.$$

Thus, asymptotically

$$\mathbb{P}\left(\eta_c(\mathbf{x}) - \hat{\eta}_c(\mathbf{x}) \leqslant z_\beta \|K\|_2 \sqrt{\frac{\eta_c(\mathbf{x})(1-\eta_c(\mathbf{x}))}{Nh^d \hat{p}(\mathbf{x})}}\right) \leqslant \beta.$$

Consequently,

$$\mathbb{P}\left(\min_c\{1-\hat{\eta}_c(\mathbf{x})\} \leqslant \lambda - \max_c z_{1-\beta}\|K\|_2\sqrt{\frac{\hat{\eta}_c(\mathbf{x})(1-\hat{\eta}_c(\mathbf{x}))}{Nh^d\hat{p}(\mathbf{x})}}\right) \leqslant \beta|C|.$$

That leads us to the procedure described as Algorithm 2.

Algorithm 2 Acceptance testing for classification

Input: Samples (X_i, Y_i), bandwidth h, parameters λ, β
Output: Accept or reject the regression result

Calculate $\hat{p}(\mathbf{x}), \hat{\eta}_c(\mathbf{x})$ for $c \in C$;
if *the density estimation* $\hat{p}(\mathbf{x}) > 0$ *and the criterion holds:*

$$\min_c(1-\hat{\eta}_c(\mathbf{x})) \leqslant \lambda - \max_c z_{1-\beta/|\mathcal{C}|}\|K\|_2\sqrt{\frac{\hat{\eta}_c(\mathbf{x})(1-\hat{\eta}_c(\mathbf{x}))}{Nh^d\hat{p}(\mathbf{x})}}$$

then
  Accept results of the regression
else
  Reject
end

We formulate a number of mild assumptions:

Assumption D.1. There exist Hessians of functions $\eta_c(\mathbf{x})$, $c \in \mathcal{C}$, and their spectral norms are bounded by some constant M_η.

Assumption D.2. L_2-norms of ∇p and $\nabla^2 p$ are bounded by constants L_d and M_d respectively, i.e.

$$\|\nabla p(\mathbf{x})\|_2 = \sqrt{\sum_i \left(\nabla_i p(\mathbf{x})\right)^2} \leqslant L_d, \qquad \sup_{\mathbf{x}:\|\mathbf{x}\|_2=1} \|\mathbf{x}^T\nabla^2 p(\mathbf{x})\|_2 \leqslant M_d.$$

Assumption D.3. There exist finite values

$$\max_{\mathbf{t}} K(\mathbf{t}), \quad \int_{\mathbb{R}^d}\|\mathbf{t}\|_2 K^2(\mathbf{t})d\mathbf{t}, \quad \int_{\mathbb{R}^d} K^2(\mathbf{t})d\mathbf{t}, \quad \int_{\mathbb{R}^d}\|\mathbf{t}\|_2^2 K(\mathbf{t})d\mathbf{t},$$

while

$$\int_{\mathbb{R}^d}\mathbf{t}K(\mathbf{t})d\mathbf{t} = 0.$$

Under these assumptions, we state the following theorem:

Theorem D.4. *Suppose that assumptions D.1–D.3 hold,* $p(\mathbf{x}) > 0$*, and* $\beta < 1/2$*. Define* $\Delta = |\lambda - \mathcal{R}^*(\mathbf{x})|$*. Then, if* $\lambda < \mathcal{R}^*(\mathbf{x})$

$$\mathbb{E}_{\mathcal{D}}\left\{\mathcal{R}_\lambda(\mathbf{x}) - \mathcal{R}^*_\lambda(\mathbf{x})\right\} \leqslant |\mathcal{C}|A_\Delta,$$

where

$$A_\Delta = 2\left\{R \wedge \left(\mathbb{1}\{\Delta \leqslant h^2\kappa_\Delta\} \vee q_\Delta\right)\right\} + \Delta(\mathbb{1}\{\Delta \leqslant h^2\kappa_\Delta\} \vee q_\Delta),$$

$$R = 2h^2\kappa_\Delta + 2\sqrt{\frac{\pi\left\{12\left(\|K\|_2^2 + \frac{hL_d}{p(\mathbf{x})}\int_{\mathbb{R}^d}\|\mathbf{t}\|_2 K^2(\mathbf{t})d\mathbf{t}\right) + \max_{\mathbf{t}} K(\mathbf{t})\right\}}{Nh^d p(\mathbf{x})}},$$

$$q_\Delta = \exp\left(-\frac{1}{2}\frac{Nh^d p(\mathbf{x})(\Delta - h^2\kappa_\Delta)^2}{(1+\Delta)^2\left(\|K\|_2^2 + \frac{hL_d}{p(\mathbf{x})}\int_{\mathbb{R}^d}\|\mathbf{t}\|K^2(\mathbf{t})d\mathbf{t}\right) + \frac{1}{3}(\Delta - h^2\kappa_\Delta)\max_{\mathbf{t}} K(\mathbf{t})}\right),$$

and

$$\kappa_\Delta = \frac{1}{p(\mathbf{x})}\left(M_d + 2dM_\eta L_d + 2dL_d\sqrt{M_\eta}\right)\int_{\mathbb{R}^d}\|t\|_2^2 K(\mathbf{t})d\mathbf{t}.$$

If $\mathcal{R}^*(\mathbf{x}) \leqslant \Delta$,

$$\mathbb{E}_{\mathcal{D}}\mathcal{R}(\mathbf{x}) - \mathcal{R}^*(\mathbf{x}) \leqslant |\mathcal{C}|\left(R + \mathbb{1}\left\{\Delta \leqslant h^2\kappa_\Delta + z_{1-\beta/|\mathcal{C}|}\|K\|_2\sqrt{\frac{1}{2Nh^2p(\mathbf{x})}}\right\} \vee \tilde{q}_\Delta + \mathbb{1}\left\{1/2 \leqslant h^2\kappa_p\right\} \vee q_p\right).$$

Here $\tilde{q}_\Delta$ *differs from* q_Δ *by replacing* $h^2\kappa_\Delta$ *with* $h^2\kappa_\Delta + z_{1-\beta/|\mathcal{C}|}\|K\|_2\sqrt{\frac{1}{2Nh^2p(\mathbf{x})}}$, *while*

$$q_p = \exp\left(-\frac{\frac{1}{2}Nh^2p(\mathbf{x})(1/2 - h^2\kappa_p)^2}{\|K\|_2^2 + \frac{hL_d}{p(\mathbf{x})}\int_{\mathbb{R}^d}\|\mathbf{t}\|_2 K^2(\mathbf{t}) + \max_{\mathbf{t}} K(\mathbf{t})(1/2 - h^2\kappa_p)}\right),$$
$$\kappa_p = \frac{M_d}{2p(\mathbf{x})}\int_{\mathbb{R}^d}\|\mathbf{t}\|^2 K(\mathbf{t})d\mathbf{t}.$$

Notice, that Theorem 4.1 follows from Theorem D.4 as all the terms in the upper bound tend to zero when $h \to 0$ and $Nh^d \to \infty$ with $N \to \infty$.

Proof of Theorem D.4. For a reminder, the excess risk is

$$\mathbb{E}_{\mathcal{D}}\left\{\mathcal{R}_\lambda(\mathbf{x}) - \mathcal{R}^*_\lambda(\mathbf{x})\right\} \leqslant 2\mathbb{E}_{\mathcal{D}}\left\{\max_{c\in\mathcal{C}}|\eta_c(\mathbf{x}) - \hat{\eta}_c(\mathbf{x})|\mathbb{1}\{\alpha(\mathbf{x}) = 0\}\right\} + \Delta\cdot\mathbb{P}_{\mathcal{D}}\left(\alpha(\mathbf{x}) \neq \alpha^*(\mathbf{x})\right).$$

First, rewrite the expectation as an integral

$$\begin{aligned}
\mathbb{E}|\hat{\eta}_c(\mathbf{x}) - \eta_c(\mathbf{x})|\mathbb{1}\{\alpha(\mathbf{x}) = 0\} &= \int_0^{+\infty}\mathbb{P}(|\hat{\eta}_c(\mathbf{x}) - \eta_c(\mathbf{x})|\mathbb{1}\{\alpha(\mathbf{x}) = 0\} \geqslant t)dt \\
&= \int_0^{+\infty}\mathbb{P}\left(|\hat{\eta}_c(\mathbf{x}) - \eta_c(\mathbf{x})| \geqslant t \text{ and } \alpha(\mathbf{x}) = 0\right)dt \\
&\leqslant \int_0^{+\infty}\mathbb{P}\left(|\hat{\eta}_c(\mathbf{x}) - \eta_c(\mathbf{x})| \geqslant t \text{ and } \sum_i K_h(X_i - \mathbf{x}) \neq 0\right)dt \\
&\leqslant \int_0^{1}\mathbb{P}\left(|\hat{\eta}_c(\mathbf{x}) - \eta_c(\mathbf{x})| \geqslant t \text{ and } \sum_i K_h(X_i - \mathbf{x}) \neq 0\right)dt,
\end{aligned}$$

since we abstain if $\hat{p}(\mathbf{x}) = 0$. Due to Lemma D.5,

$$\begin{aligned}
&\int_0^{1}\mathbb{P}\left(|\hat{\eta}_c(\mathbf{x}) - \eta_c(\mathbf{x})| \geqslant t \text{ and } \sum_i K_h(X_i - \mathbf{x}) \neq 0\right)dt \leqslant \\
&\leqslant 2h^2\kappa + 2\sqrt{\frac{\pi\left\{\left(\|K\|_2^2 + \frac{hL_d}{p(\mathbf{x})}\int_{\mathbb{R}^d}\|\mathbf{t}\|_2 K^2(\mathbf{t})d\mathbf{t}\right) + \max_{\mathbf{t}} K(\mathbf{t})\right\}}{2Nh^dp(\mathbf{x})}}.
\end{aligned}$$

Here we use the Poisson integral. Denote this upper bound by R.

Now, assume $\lambda < \mathcal{R}^*(\mathbf{x})$**.** Then the excess risk can be estimated in the following way:

$$2\mathbb{E}_{\mathcal{D}}\left\{\max_{c\in C}|\eta_c(\mathbf{x}) - \hat{\eta}_c(\mathbf{x})|\mathbb{1}\{\alpha(\mathbf{x}) = 0\}\right\} \leqslant 2\left[R \wedge \mathbb{P}\left(\alpha(\mathbf{x}) = 0\right)\right],$$
$$\Delta\cdot\mathbb{P}\left(\alpha(\mathbf{x}) \neq \alpha^*(\mathbf{x})\right) = \Delta\mathbb{P}\left(\alpha(\mathbf{x}) = 0\right),$$
$$\mathbb{P}(\alpha(\mathbf{x}) = 0) = \mathbb{P}\left(\sum_i K_h(X_i - \mathbf{x}) > 0 \text{ and } \min_c(1 - \hat{\eta}_c(\mathbf{x})) \leqslant \lambda - z_{1-\beta/|\mathcal{C}|}\|K\|_2\sqrt{\frac{\hat{\eta}(\mathbf{x})\big(1 - \hat{\eta}(\mathbf{x})\big)}{Nh^d\hat{p}(\mathbf{x})}}\right).$$

The event from the RHS implies that there is $c \in C$ such that

$$\hat{\eta}_c(\mathbf{x}) - \eta_c(\mathbf{x}) \geqslant \Delta, \tag{3}$$

and, consequently,

$$\mathbb{P}(\alpha(\mathbf{x}) = 0) \leqslant \sum_{c \in \mathcal{C}} \mathbb{P}\left(\hat{\eta}_c(\mathbf{x}) - \eta_c(\mathbf{x}) \geqslant \Delta \text{ and } \sum_i K_h(X_i - \mathbf{x}) > 0\right).$$

The upper bound was obtained using Lemma D.5.

Finally, consider the case $\mathcal{R}^*(\mathbf{x}) \geqslant \lambda$. Then, we estimate

$$\mathbb{E}_{\mathcal{D}}\{\max_c |\hat{\eta}_c(\mathbf{x}) - \eta_c(\mathbf{x})| \mathbb{1}\{\alpha(\mathbf{x}) = 0\}\} \leqslant R|\mathcal{C}|,$$

and

$$\begin{aligned}
\mathbb{P}(\alpha(\mathbf{x}) = 1) &\leqslant \sum_{c \in \mathcal{C}} \mathbb{P}\left(\eta_c(\mathbf{x}) - \hat{\eta}_c(\mathbf{x}) \geqslant \Delta - z_{1-\beta/|\mathcal{C}|}\|K\|_2 \max_c \sqrt{\frac{\hat{\eta}_c(\mathbf{x})(1-\hat{\eta}_c(\mathbf{x}))}{Nh^d \hat{p}(\mathbf{x})}} \text{ or } \hat{p}(\mathbf{x}) = 0\right) \\
&\leqslant \sum_{c \in \mathcal{C}} \mathbb{P}\left(\eta_c(\mathbf{x}) - \hat{\eta}_c(\mathbf{x}) \geqslant \Delta - z_{1-\beta/|\mathcal{C}|}\|K\|_2 \sqrt{\frac{1}{2Nh^d p(\mathbf{x})}} \text{ and } \hat{p}(\mathbf{x}) > 0\right) \\
&+ |\mathcal{C}| \mathbb{P}\left(\hat{p}(\mathbf{x}) \leqslant \frac{p(\mathbf{x})}{2}\right).
\end{aligned}$$

Similarly to Lemma D.5, we bound the last probability by Bernstein's inequality:

$$\mathbb{P}\left(\hat{p}(\mathbf{x}) \leqslant \frac{p(\mathbf{x})}{2}\right) \leqslant \mathbb{1}\left\{\frac{1}{2} \leqslant h^2 \kappa_p\right\} \vee \exp\left(-\frac{\frac{1}{2}h^d N p(\mathbf{x}) \left(\frac{1}{2} - h_p^\kappa\right)}{\|K\|_2^2 + \frac{hL_d}{p(\mathbf{x})} \int_{\mathbb{R}^d} \|\mathbf{t}\|_2 K^2(\mathbf{t}) + \max_{\mathbf{t}} K(\mathbf{t})(1/2 - h^2\kappa_p)}\right).$$

□

Lemma D.5. *Suppose all conditions of Theorem D.4 holds. Then, for any non-negative r*

$$\mathbb{P}(\hat{\eta}_c(\mathbf{x}) - \eta_c(\mathbf{x}) \geqslant r \text{ and } \sum_i K_h(X_i - \mathbf{x}) > 0) \leqslant$$

$$\leqslant \mathbb{1}\{r \leqslant h^2\kappa\} \vee \exp\left\{-\frac{1}{2}\frac{Nh^d p(\mathbf{x})(r - h^2\kappa)^2}{(1+r)^2\left(\|K\|_2^2 + \frac{hL_d}{p(\mathbf{x})}\int_{\mathbb{R}^d}\|\mathbf{t}\|_2 K^2(\mathbf{t})d\mathbf{t}\right) + (\max_{\mathbf{t}} K(\mathbf{t}) + r)|r - h^2\kappa|}\right\}$$

and

$$\mathbb{P}(\eta_c(\mathbf{x}) - \hat{\eta}_c(\mathbf{x}) \geqslant r \text{ and } \sum_i K_h(X_i - \mathbf{x}) > 0) \leqslant$$

$$\leqslant \mathbb{1}\{r \leqslant h^2\kappa\} \vee \exp\left\{-\frac{1}{2}\frac{Nh^d p(\mathbf{x})(r - h^2\kappa)^2}{(1+r)^2\left(\|K\|_2^2 + \frac{hL_d}{p(\mathbf{x})}\int_{\mathbb{R}^d}\|\mathbf{t}\|_2 K^2(\mathbf{t})d\mathbf{t}\right) + (\max_{\mathbf{t}} K(\mathbf{t}) + r)(r - h^2\kappa)}\right\}.$$

Proof. Let us prove the first inequality, the second one can be proved in the same way. Since $\hat{p}(\mathbf{x}) > 0$, we can multiply both sides of the inequality $\hat{\eta}_c(\mathbf{x}) - \eta_c(\mathbf{x}) \geqslant r$ by $\sum_i K_h(X_i - \mathbf{x})$. Thus, the inner event implies

$$\sum_i \mathbb{1}\{Y_i = c\}K_h(X_i - \mathbf{x}) - \eta_c(\mathbf{x})K_h(X_i - \mathbf{x}) - rK_h(X_i - \mathbf{x}) \geqslant 0.$$

Define

$$\begin{aligned}
e_i &= \mathbb{1}\{Y_i = c\}K_h(X_i - \mathbf{x}) - \eta_c(\mathbf{x})K_h(X_i - \mathbf{x}) - rK_h(X_i - \mathbf{x}), \\
e &= \mathbb{E}e_i = \mathbb{E}\{\eta_c(X_i) - \eta_c(\mathbf{x})\}K_h(X_i - \mathbf{x}) - r \cdot \mathbb{E}K_h(X_i - \mathbf{x}).
\end{aligned}$$

In order to write a concentration we should analyze e. Inequalities

$$\|\nabla\eta_c(\mathbf{x}) - \nabla\eta_c(\mathbf{y})\| \leqslant \sqrt{d}M_\eta\|\mathbf{x}-\mathbf{y}\|,$$
$$\left|\int_0^\lambda \nabla_i\eta_c(\mathbf{x}+se_i)ds\right| \leqslant |\eta_c(\mathbf{x}+\lambda e_i) - \eta_c(\mathbf{x})| \leqslant 1,$$

which hold for each λ, $\mathbf{x}$ and $\mathbf{y}$, guarantee us that the norm of the gradient $\nabla\eta_c(\mathbf{x})$ is bounded by

$$L_\eta = 2d\sqrt{M_\eta}.$$

Moreover, non-negativity of $p(\mathbf{x})$, the L_d-Lipschitz property and its L_1-norm imply that $p(\mathbf{x})$ is bounded by $2L_d d$. Then, Taylor's expansion delivers the following:

$$\begin{aligned}
|\mathbb{E}\{\eta_c(X_1) - \eta_c(\mathbf{x})\}K_h(X_1-\mathbf{x})| &= \left|\int_{\mathbb{R}^d}(\eta(\mathbf{x}') - \eta(\mathbf{x}))K_h(\mathbf{x}'-\mathbf{x})p(\mathbf{x}')d\mathbf{x}'\right| \\
&\leqslant \left|h\int_{\mathbb{R}^d}\langle\nabla\eta(\mathbf{x}), t\rangle K(t)p(\mathbf{x}+ht)dt\right| + h^2 dM_\eta L_d\int_{\mathbb{R}^d}\|\mathbf{t}\|_2^2 K(\mathbf{t})d\mathbf{t} \\
&\leqslant h^2 L_d d\left(\sqrt{M_\eta} + M_\eta\right)\int_{\mathbb{R}^d}\|\mathbf{t}\|_2^2 K(\mathbf{t})d\mathbf{t}.
\end{aligned}$$

Similarly,

$$|\mathbb{E}K_h(X_i-\mathbf{x}) - p(\mathbf{x})| \leqslant \frac{h^2}{2}M_d\int_{\mathbb{R}^d}\|\mathbf{t}\|_2^2 K(\mathbf{t})d\mathbf{t}.$$

Thus,

$$(-e) \geqslant p(\mathbf{x})r - \frac{h^2}{2}\left(M_d + 2dM_\eta L_d + 2dL_d\sqrt{M_\eta}\right)\int_{\mathbb{R}^d}\|\mathbf{t}\|_2^2 K(\mathbf{t})d\mathbf{t} = p(\mathbf{x})(r - h^2\kappa).$$

If $e > 0$ we estimate the probability by 1. Otherwise, we utilize Bernstein's inequality:

$$\begin{aligned}
\mathbb{P}\left(\sum_i e_i - Ne \geqslant N(-e)\right) &\leqslant \exp\left(-\frac{\frac{1}{2}N^2e^2}{\sum_i \operatorname{Var} e_i + \frac{1}{3}h^{-d}\max_t K(t)N(-e)}\right) \\
&\leqslant \exp\left(-\frac{\frac{1}{2}Ne^2}{\mathbb{E}e_1^2 + \frac{1}{3}h^{-d}\max_t K(t)(-e)}\right) \\
&\leqslant \exp\left(-\frac{\frac{1}{2}Ne^2}{(1+r)^2\mathbb{E}K_h^2(X_i-\mathbf{x}) + h^{-d}(\max_{\mathbf{t}} K(\mathbf{t}) + r)(-e)}\right).
\end{aligned} \tag{4}$$

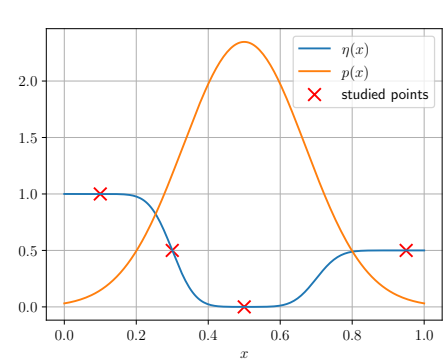
We estimate $\mathbb{E}K_h^2(X_i-\mathbf{x})$ as follows:

$$\mathbb{E}K_h^2(X_i-\mathbf{x}) = h^{-d}\int_{\mathbb{R}^d}K^2(\mathbf{t})p(\mathbf{x}+h\mathbf{t})dt \leqslant h^{-d}\left(p(\mathbf{x})\|K\|_2^2 + hL_d\int_{\mathbb{R}^d}\|\mathbf{t}\|K^2(\mathbf{t})d\mathbf{t}\right).$$

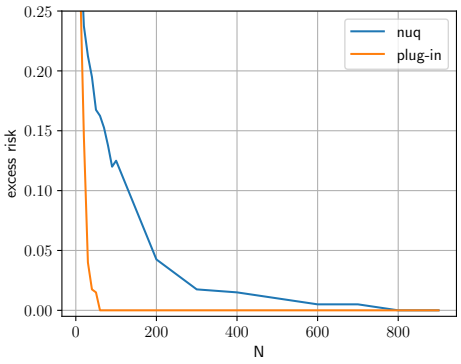
□

D.1 Toy Example of Classification with Reject Option

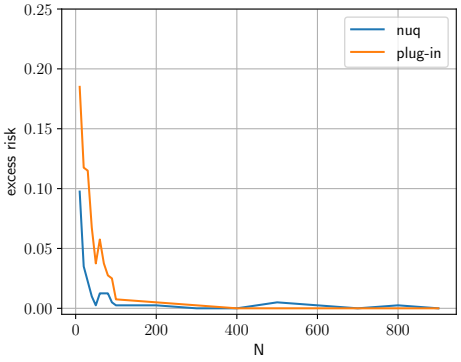
For illustration, we present some experimental results with a toy example, see Figure 6. We consider the smoothed step function as $\eta(\mathbf{x})$, while data X_i is distributed according to the normal distribution with the mean 0.5 and variance 0.04 (see Figure 10b). We study the point-wise excess risk of NUQ and the plug-in rule. For the points with high covariate mass (Figures 6c and 6d) methods show comparable results. NUQ is useful for points lying with low covariate density, see Figure 6e. However, for points without any label noise (Figure 6b) plug-in is still better as it quickly learns the correct class while NUQ is more conservative.



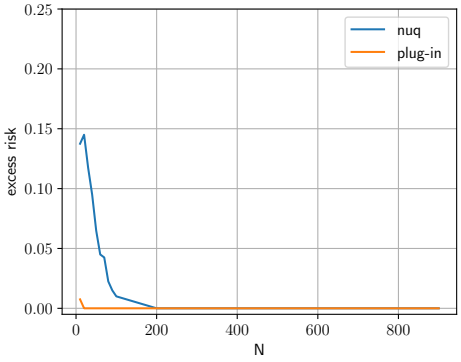
(a) data



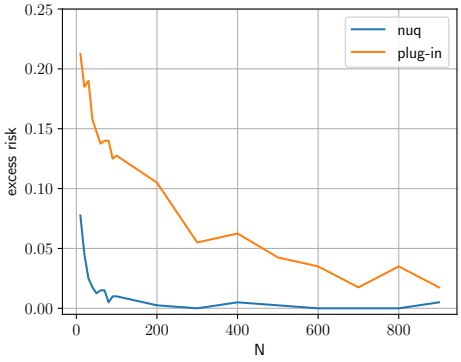
(b) $x = 0.1$



(c) $x = 0.3$



(d) $x = 0.5$



(e) $x = 0.95$

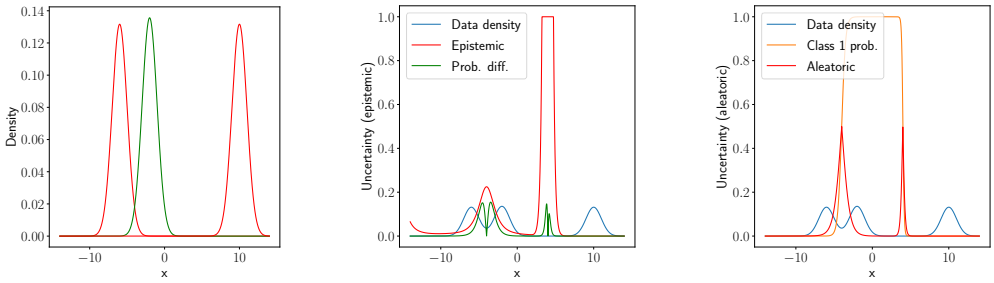
Figure 6: Excess risk at studied points marked on Figure 10b. We choose RBF-kernel and quantile $1 - \beta$ to be equal 0.95.

E Toy Experiment on Detecting Actual Aleatoric and Epistemic Uncertainties

In this section, we conduct a toy experiment, for which we explicitly know what should be the true probability of class one, as well as the true data density.

Let us consider a binary classification problem. Our dataset consists of 5000 samples from three different one-dimensional Gaussians, located to mix classes. Colors denote class label: red – 0; green – 1, see Figure 7a. For this particular data model, we can compute the conditional probability of a data point $\mathbf{x}$ belongs to class 1: $\eta(\mathbf{x}) = p(Y = 1 \mid X = \mathbf{x})$. We build an estimate of this conditional

using our Nadaraya-Watson kernel-based approach $\hat{\eta}(\mathbf{x})$. Further, we generate a uniform grid, and for each point of this grid, using our method, we can upper bound difference between the true conditional and our approximation. This difference, according to our approach, is considered as an epistemic uncertainty (see Figure 7b). The green line in this plot denotes an absolute difference between the true conditional and our approximation. The red line denotes our epistemic uncertainty. From the picture, we can see that our epistemic uncertainty approximates the probabilities difference well. Next, we show how our aleatoric uncertainty relates to the true class 1 conditional probability. In the Figure 7c we show true conditional distribution $\eta(\mathbf{x})$ (orange line) and our approximation of the aleatoric uncertainty $\min\{\hat{\eta}(\mathbf{x}), 1 - \hat{\eta}(\mathbf{x})\}$ (red line). We can see that our approximation is high exactly in the same regions where the true conditional is absolutely unsure about the class label.



(a) (b) (c)

Figure 7: (a): Mixture of one dimensional Gaussians we took samples from. Color denotes class label. (b): Epistemic uncertainty our model assigns to data points. Note that the uncertainty is quite high in the region of 3-5. For the sake of visualization, we clipped the maximum value to be 1. (c): Our approximation of aleatoric uncertainty is built along with the true conditional probability.

F Additional Experiments on Image Datasets

F.1 Rotated MNIST

The first example is classification under covariate shift on MNIST [40]. We train a small convolutional neural network with three convolution layers, see Section B. This is the base model we use to obtain logits for the input objects. We consider a particular instance of distribution shift for evaluation by using a test set of MNIST images rotated at a random angle in the range from 30 to 45 degrees. This set contains 10000 images. The range of angles reassures that the data does not look like the original MNIST data, though many resulting pictures can still remind the ones from training.

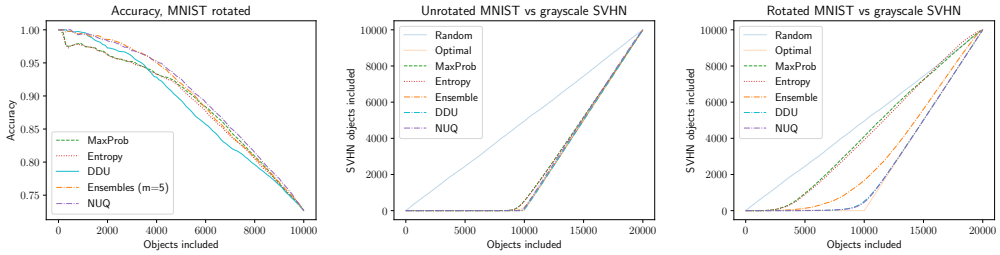
This experiment considers two simple baselines: MaxProb and Entropy-based uncertainty estimates of the base model. We compliment them with two more challenging baselines: Deep ensemble and DDU [55] (as the most successful among deterministic methods). We compare them all with NUQ-based estimate of epistemic uncertainty $\hat{\mathbf{U}}_e(\mathbf{x})$. To evaluate the quality of the uncertainty estimates, we sort the objects from the test dataset in order of ascending uncertainties.

Then we obtain the model's predictions and plot how accuracy changes with the number of objects taken into consideration; see Figure 8a. The valid uncertainty estimation method is expected to produce the plot with accuracy decreasing when more objects are taken into account. Moreover, the higher is the plot, the better is the quality of the corresponding uncertainty estimate.

We see that the plots for all the considered methods show the expected trend, while uncertainties obtained by NUQ are more reliable. NUQ distinctly outperforms DDU and has comparable performance with deep ensembles.

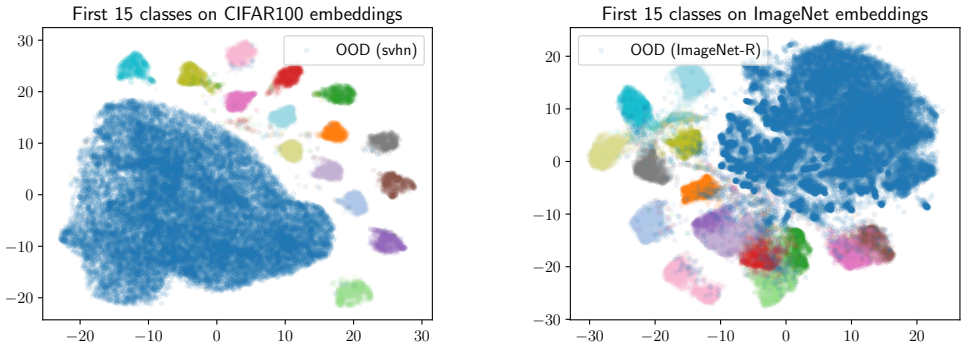
F.2 MNIST vs. SVHN

To make the problem more challenging, we consider the SVHN dataset [58], convert it to grayscale, and resize it to the shape of 28x28. The size of this additional SVHN-based dataset is again 10000.



(a) (b) (c)

Figure 8: (a) Accuracy for images sorted by uncertainty on rotated MNIST. (b) Number of SVHN images included into consideration vs unrotated MNIST. In this simpler version, even the basic entropy manages to achieve a good result. (c) More challenging task – number of SVHN images included into consideration vs rotated MNIST. NUQ still distinguish between datasets with close to an optimal solution.



(a) (b)

Figure 9: Embeddings space visualization for CIFAR (a) and ImageNet (b). We present the embeddings for first 15 classes on test dataset (in various colors) and all the embeddings for out-of-distribution datasets (in blue). The OOD dataset for CIFAR is SVHN and for ImageNet it is ImageNet-R.

We take the base model trained on MNIST from the previous section and consider the problem of OOD detection with SVHN being the OOD dataset. As in-distribution data, we first consider the test set of 10000 MNIST images. We again compute uncertainties for each object on this concatenated dataset (10000 of MNIST and 10000 of SVHN) and sort them by their uncertainties in ascending order. The goal for uncertainty quantification methods is to sort all objects so that all MNIST images have lower uncertainty values than SVHN ones. Note that the optimal decision rule in this case is a ReLU-shaped function, with a break at point 10000.

For NUQ we use epistemic uncertainty $\hat{\mathbf{U}}_e(\mathbf{x})$ in this experiment. In Figure 8b we plot the number of objects included from the SVHN dataset. It is seen that Ensembles, NUQ, and DDU outperform MaxProb and Entropy. All of these three methods perform almost as an optimal decision.

Next, we consider more challenging problem of separation between rotated MNIST (see Section F.1) and SVHN. We expect it is harder to distinguish between them as rotated MNIST images differ from those used to train the network. However, Figure 8c shows that NUQ still does a very good job and allows for almost perfect separation. Interestingly, this example shows that ensembles are worse than DDU and NUQ. The performance of the last two is visually almost identical.

F.3 Performance Difference on CIFAR-100 and ImageNet

One of the things that caught our attention is the superior performance of NUQ on ImageNet, given that it has very similar results with DDU on CIFAR-100. One of our hypotheses was that embeddings have a more complex and multi-modal distribution for a more complex ImageNet dataset compared to simpler CIFAR-100. To check this, we made t-SNE based embeddings of out-of-distribution and test data (see Figure 9). While we understand the limitation of this type of visualization, the ImageNet embeddings appear to be much more irregular compared to well-shaped clusters for CIFAR-100. Because of that, the modeling of the class with a single Gaussian (in DDU) might not work very well for ImageNet. NUQ approach performs the modeling of distributions in a much more flexible way, which is beneficial for approximating complex distributions. We hypothesize that this is the reason for the NUQ's superior performance.

F.4 Ablation Study on CIFAR-100

We need an estimator of marginal density $p(\mathbf{x})$ for our method, and there exist different options. We consider kernel method with RBF kernel and logistic kernel and Gaussian mixtures model (GMM). There is also a question about which embeddings to use – the DDU paper [55] proposes to take the features from the second last layer; while logits from the last layer represent a reasonable choice as well. To validate the options, we conducted some ablation study on out-of-distribution detection for the CIFAR-100 dataset, similar to the main experiment.

First, we compare the DDU and NUQ on embeddings from the second last and last layers (Table 4) on SVHN, LSUN, and Smooth datasets. Secondly, we compare the NUQ method on RBF, logistic kernel, and GMM for both last and penultimate layer embeddings (Table 5). As we can see from the tables, the optimal option is GMM density on the penultimate layer.

	DDU, features	DDU, logits	NUQ, features	NUQ, logits
SVHN	89.6±1.6	88.2±0.6	89.7±1.6	88.2±0.6
LSUN	92.1±0.6	90.9±0.4	92.3±0.6	90.9±0.4
Smooth	97.1±3.1	96.3±4.1	96.8±3.8	96.2±4.1

Table 4: Comparison of DDU and NUQ predictions on different type of embeddings: logits (last layer) and features (second last layer).

	RBF, f	RBF, l	Logistic, f	Logistic, l	GMM, f	GMM, l
SVHN	84.4±3.2	84.7±3.1	84.8±2.9	86.7±2.6	89.7±1.6	88.2±0.6
LSUN	88.2±1.0	88.1±0.8	88.5±4.0	90.3±1.0	92.3±0.6	90.9±0.4
Smooth	85.5±6.8	87.7±9.4	86.2±8.2	90.8±7.8	96.8±3.8	96.2±4.1

Table 5: Probability density methods comparison – radial basis function kernel (RBF), logistic kernel, Gaussian mixture models (GMM). 'f' (features) marks models, built on embeddings from a second last layer and 'l' (logits) is for the ones built on embeddings from a last layer.

Kernel-based methods rely on the "reasonable" geometry of the embedding space, meaning that embeddings of similar images should be not too far and different images should not collapse into a single point. Our motivation to use spectral normalization during training is to make the embedding space smooth with respect to input images. We have conducted an extra ablation study, comparing the result for feature extractors with and without spectral normalization, see Table 6. The results confirm our hypothesis, as the spectral-normalized version performs better, though the NUQ beats the baseline even without applying the modification to the ResNet training. We also show here that entropy performs better than maximum probability as an uncertainty measure.

F.5 Sensitivity to Ensemble Size

In this section, we explore the ensemble model performance regarding the number of models. From Table 7, we can see that 5 models is a reasonable amount number for CIFAR-100. For the ImageNet dataset (Table 8), increasing the number of models gives a steady gain, but still 5 models provide the gain within the error margin.

OOD dataset	DDU	DDU (spectral)	NUQ	NUQ (spectral)
SVHN	88.7±4.3	89.6±1.6	86.8±1.2	89.7±1.6
LSUN	91.3±0.9	92.1±0.6	91.2±1.1	92.3±0.6
Smooth	95.7±1.2	97.1±3.1	95.5±1.3	96.8±3.8

Table 6: Comparing the influence of spectral normalization on the model performance for OOD detection, ROC-AUC.

OOD dataset	2	3	5	7	10
SVHN	82.3 ± 1.3	82.4 ± 0.7	82.9 ± 0.9	82.7 ± 0.7	82.6 ± 0.5
LSUN	85.1 ± 0.6	85.9 ± 0.6	86.5 ± 0.8	87.1 ± 0.6	87.1 ± 0.6
Smooth	83.7 ± 6.5	83.4 ± 3.2	83.7 ± 1.2	83.3 ± 1.5	83.2 ± 1.6

Table 7: Ablation study for ensemble size on CIFAR100 in-distribution dataset for OOD detection. The number of models above 5 give almost no gain (even some loss sometimes) within an error margin.

OOD dataset	2	3	5	7	9
ImageNet-O	49.8	50.8	51.9	52.1	52.6
ImageNet-R	84.7	85.3	85.8	86	86.1

Table 8: Ablation study for ensemble size on ImageNet out-of-distribution detection task.

F.6 Additional Experiments with DUQ

DUQ [69] is one of the baselines in the paper. We chose it as similarly to our method it requires only a single forward pass and uses post-processing for embeddings. In the original article, the method shows a good result on a relatively small dataset CIFAR-10 with a ResNet18 model. We tried to train it on CIFAR-100 and ImageNet with a larger model, ResNet50, but there were difficulties with training process as method failed to converge to the model of reasonable quality. We believe the cause of the problem was gradient penalty as a regularization method, and we switched to spectral normalization instead. DUQ training requires a balance between hyperparameters such as length scale, momentum, and learning rate, so we initially trained the model with a pre-trained feature extractor. With careful selection of parameters, we managed to train end-to-end as well, and we observed improvement in all experiments (see Tables 9 and 10), although methods like DDU and NUQ had more stable training in our experience and a better final result.

OOD dataset	Ensembles	TTA	DDU	NUQ	DUQ Head	DUQ end-to-end
SVHN	82.9 ± 0.9	81.6 ± 1.2	89.6 ± 1.6	**89.7 ± 1.6**	83.6 ± 4.0	88.7 ± 6.3
LSUN	86.5 ± 0.8	85.0 ± 2.7	92.1 ± 0.6	**92.3 ± 0.6**	87.2 ± 2.1	90.8 ± 6.7
Smooth	83.7 ± 1.2	73.2 ± 10.8	**97.1 ± 3.1**	96.8 ± 3.8	83.8 ± 11.4	91.1 ± 8.4

Table 9: Performance on OOD detection for CIFAR-100

OOD dataset	Ensemble	DDU*	NUQ (spectral)*	DUQ Head	DUQ end-to-end
ImageNet-R	84.4	74.2	**99.5**	57.4	73.3
ImageNet-O	51.9	74.1	**82.4**	67.3	71.4

Table 10: Performance on OOD detection on ImageNet

F.7 Computational Costs

We benchmarked the training and inference time overhead for our ImageNet experiments. Base model training time corresponds to training of a ResNet-50 feature extractor on a corresponding dataset. The NUQ training time is measured on a full training dataset and includes the parameter search time. The inference time is measured on a full test dataset. Results are presented in Table 11, where we can see that training overhead is less than 1% and inference overhead is less than 10%, while for ensembles, the overhead would be n-fold.

	Base model	NUQ	Overhead
CIFAR-100 training time	11.5 hours	42.6 seconds	0.10%
CIFAR-100 inference time	81 seconds	1.8 seconds	2.25%
ImageNet training time	49 hours	28 minutes	0.96%
ImageNet inference time	225 seconds	21 seconds	9.30%

Table 11: Execution time overhead for NUQ on CIFAR-100 and ImageNet datasets

	TTA				Ensemble			
	MaxProb	Entropy	Std	MI	MaxProb	Entropy	Std	MI
Imagenet-O	29.2	30.5	32.3	34.8	48.3	51.9	58.5	59.1
Imagenet-R	82.8	85.8	74.1	84.7	81	84.4	70.2	76.1

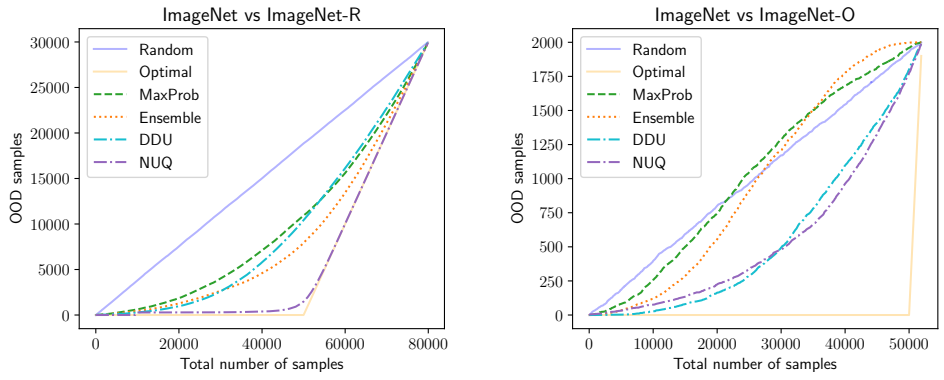
Table 12: Study on ensemble and test time augmentation reduction methods on OOD detection. For in-distribution we used test ImageNet dataset and for out-of-distribution ImageNet-O and ImageNet-R were used, respectively. The metric is ROC-AUC.

F.8 Choice of Uncertainty Measure for Ensembles

In the case of ensembles, we have multiple predictions for a single point. It also applies to other methods like test-time augmentation and Monte-Carlo dropout. To get a single value of uncertainty, we must choose the particular uncertainty measure based on . We have tried a few options. Namely, we used mean maximum probability, a standard deviation of predicted probabilities between different runs, mutual information, and entropy. However, in the manuscript, we decided to focus on only one approach (entropy) to reduce clutter, while in our experiments, it performed better on average. We present the results for test time augmentation and ensembles on ImageNet here, see Table 12.

F.9 Detecting OOD Images for ImageNet

By analogy with Section F.1, we conducted experiments with vanilla ImageNet images vs ImageNet-R and ImageNet-O. The idea is to mix images from two datasets and then sort them by uncertainty. For normal images uncertainty tends to be lower, so we can illustratively see the methods performance (Figure 10). In case of ImageNet-O, basic models like max probability and ensemble have difficulties, while NUQ and DDU performed reasonably well. For ImageNet-R, NUQ has performance close to optimal.



(a) ImageNet-R as out-of-distribution

(b) ImageNet-O as out-of-distribution

Figure 10: Detecting OOD images. We sort images by increasing uncertainty and plot the number of in distribution points (lower is better). Ideal model should select samples from original dataset first.

G Hyperparameter Values and Dataset Statistics for Experiments on Textual Data

For the optimal hyperparameter search in the experiments on textual datasets, we use Bayesian optimization with an early stopping algorithm. We divide the original training data into validation and training subsets in a ratio of 20 to 80 and perform optimization using sets of pre-defined values for each hyperparameter, which are given in the caption of Table 13. As an objective metric, we use the accuracy score. The dataset statistics are presented in Table 14.

Dataset	Objective Score	Spect. Norm.	SNGP	Learning Rate	Num. Epochs	Batch Size	Weight Decay
MRPC	0.867	-	-	5e-05	12	32	1e-1
MRPC	0.858	+	-	3e-05	11	32	1e-1
MRPC	0.873	+	+	1e-4	5	16	0
CoLA	0.88	-	-	1e-05	8	4	1e-1
CoLA	0.876	+	-	3e-05	15	32	1e-1
CoLA	0.884	+	+	7e-06	5	8	0
SST-2	0.936	-	-	1e-05	15	64	1e-1
SST-2	0.939	+	-	5e-05	7	64	1e-2
SST-2	0.921	+	+	2e-05	15	8	1e-1
CLINC	0.979	-	-	3e-05	7	16	1e-1
CLINC	0.98	+	-	7e-05	9	64	1e-1
CLINC	0.978	+	+	7e-05	13	64	1e-2
ROSTD	0.994	-	-	7e-06	6	32	1e-1
ROSTD	0.995	+	-	7e-06	6	32	0
ROSTD	0.994	+	+	3e-05	13	64	0

Table 13: Optimal hyperparameters for the experiments with ELECTRA on textual data. "Objective score" refers to the accuracy score for classification on the validation sample. We select hyperparameter values from the following pre-defined list:
Learning rate: [5e-6, 6e-6, 7e-6, 9e-6, 1e-5, 2e-5, 3e-5, 5e-5, 7e-5, 1e-4];
Num. of epochs: $\{n \in \mathbb{N} | 2 \leq n \leq 15\}$;
Batch size: [4, 8, 16, 32, 64];
Weight decay: [0, 1e-2, 1e-1].

Datasets	Train	Test	# Labels
MRPC	3.7K	0.4K	2
CoLA	8.6K	1.0K	2
SST-2	67.3K	0.9K	2
CLINC	15K	5.5K	150
ROSTD	30.5K	11.7K	12
20 News Groups	11.3K	7.5K	20
IMDB	20K	25K	2
TREC-10	5.5K	0.5K	6
WMT-16	4500K	3K	-
Amazon	207.4K	29.6K	5
MNLI	392.7K	9.8K	3
RTE	2.5K	0.3K	2

Table 14: Dataset statistics. The table presents the number of sequences for the training and test parts of the datasets. For the datasets from the GLUE benchmark (MRPC, CoLA, SST-2), we used the available validation set as the test set. For CLINC and ROSTD we present the size of the training part only for in-domain intents. From the last 7 datasets, we use only the test part as OOD instances.