Out-of-distribution Detection Using Kernel Density Polytopes

Jayanta Dey, Will LeVine, Ashwin De Silva, Ali Geisa, Joshua T. Vogelstein

Department of Biomedical Engineering, Johns Hopkins University

Abstract

Any reasonable machine learning model should not only interpolate efficiently in between the training samples provided (in-distribution region), but also approach the extrapolative or out-of-distribution (OOD) region without being overconfident. Many state-of-the-art algorithms [2, 1] have tried to fix the overconfidence problem in the OOD region. However, in doing so, they have often impaired the in-distribution performance of the model. Our key insight is that machine learning models partition the feature space into polytopes and learn constant (random forest) or affine (ReLu nets) functions over those polytopes. This leads to the out-of-distribution overconfidence problem for the polytopes which lie in the training data boundary and extends to infinity. To resolve this issue, we propose kernel density methods that fit Gaussian kernel over the polytopes, which are learned using machine learning models. Specifically, we introduce a variant of kernel density polytopes: Kernel Density Forest (KDF) based on random forests. Studies on various simulation settings show that KDF achieves uniform confidence over the classes in the OOD region while maintaining good in-distribution accuracy compared to that of its parent model—random forests.

Problem Formulation

Consider a supervised learning problem with independent and identically distributed training samples $\{(x_i, y_i)\}_{i=1}^{N}$ such that $(X, Y) \sim \mathcal{P}_{X\times Y}$, where $X \sim \mathcal{P}_{in}$ is a $\mathcal{X} \in \mathbb{R}^d$ valued input and Y is a $\mathcal{Y} = \{1, \cdots, K\}$ valued label. We consider the OOD-distributions $\mathcal{P}_{out}$ with their support spanning the whole feature space excluding the support of $\mathcal{P}_{in}$. In other words, $\mathcal{P}_{out} := \bigcup_j \mathcal{P}_j$, where $\mathcal{P}_j$ is any distribution having little or no overlap with $\mathcal{P}_{in}$. Here the goal is to learn a model $g : \mathbb{R}^d \times \{\mathbb{R}^d \times \{1, \cdots, K\}\}^N \to [0, 1]^k$ such that,

$$g(x) = \begin{cases} P[Y = k|X = x], & \forall k \text{ if } x \sim \mathcal{P}_{in} \\ P[Y = k], & \forall k \text{ if } x \sim \mathcal{P}_{out} \end{cases} \tag{1}$$

where $P[Y = k|X = x]$ is the true posterior probability of class k given by,

$$P[Y = k|X = x] = \frac{f_k(x)P[Y = k]}{\sum_{i=1}^{K} f_i(x)P[Y = i]} \tag{2}$$

Here, $f_i(x)$ is the true training class conditional density function of class i. The class prediction $\hat{y}$ for a test sample x is obtained by,

$$\hat{y} = \arg\max(g(x)) \tag{3}$$

Proposed Model

- Consider p polytopes $\{Q_r\}_{r=1}^{p}$ from a trained random forest.
- Fit a Gaussian to each polytope Q_r.

$$\hat{f}_k(x) = \frac{1}{N_k}\sum_{r=1}^{p} N_{rk}\mathcal{K}_{rk}(x) \tag{4}$$

where N_{rk} is the total number of samples that end up in the current polytope Q_r of class k and $N_k = \sum_{r=1}^{p} N_{rk}$ is the total number of training samples with class k.

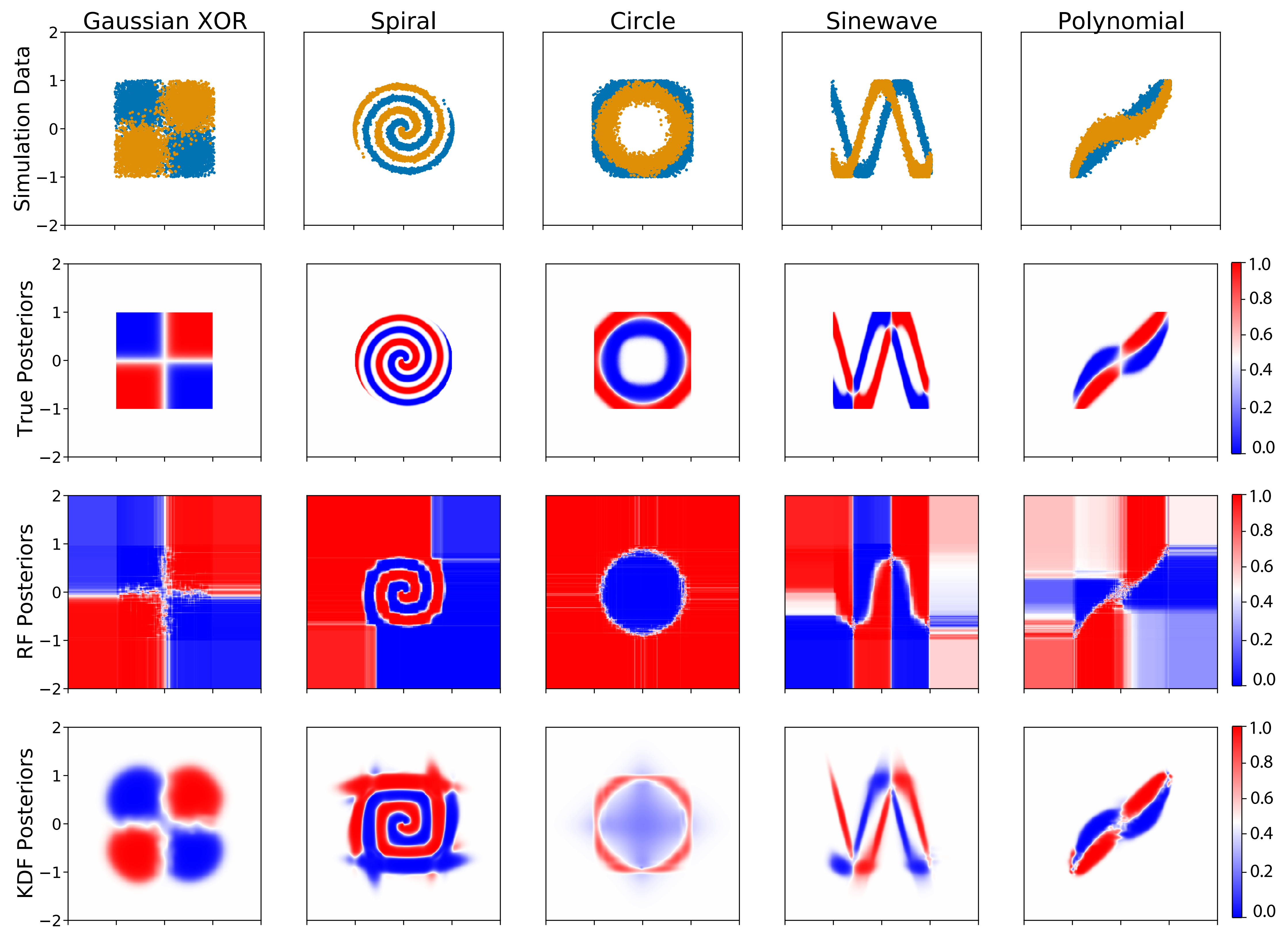
- Add suitable bias the model.

$$\tilde{f}_k(x) = \hat{f}_k(x) + b \tag{5}$$

Bias Intuition

The motivation behind the bias term is to modify our estimator such that its posterior estimates become closer to the class priors as a given inference point x becomes further from the training data X.

Simulation Results



Simulation distributions and posterior estimates by different algorithms. Binary simulation data are generated within the region bounded by $[-1, 1] \times [-1, 1]$, then posteriors are estimated within the $[-2, 2] \times [-2, 2]$ region. Kernel Density Forests (KDFs) yields better estimates when compared to its respective parent models—random forests (RFs). It achieves particularly good posteriors in the out-of-distribution (OOD) region, while RFs yields overconfident posteriors.

Theorem 1: Let d represent euclidean distance. Now assume that $\min_i d(x, X_i) \to \infty$ and that the variances of all $\mathcal{K}_{rk}$ are bounded. Then $\hat{P}[Y = k|X = x] \to \hat{P}[Y = k]$.

More Theoretical Results

- **Theorem 2**: Given the polytope size $h_{rk} \to 0$, the number of samples within each polytope $N_{rk} \to \infty$ and N_{rk} grows slowly compared to the total sample size $\frac{N_{rk}}{N_k} \to 0$, kernel density polytope is an unbiased estimator of the true class conditional in-distribution density function $f_k(x)$.
- **Theorem 3**: Given the conditions in theorem 1, variance of the kernel density polytope estimate var$(\hat{f}_k(x)) \to 0$.

Out-of-distribution Results

	RF		**KDF**	
Simulation	AUROC	FPR@95	AUROC	FPR@95
Gaussian XOR	$0.80(\pm 0.03)$	$0.83(\pm 0.13)$	$\mathbf{0.98}(\pm\mathbf{0.00})$	$\mathbf{0.32}(\pm\mathbf{0.28})$
Spiral	$0.55(\pm 0.02)$	$0.75(\pm 0.16)$	$\mathbf{0.99}(\pm\mathbf{0.00})$	$\mathbf{0.39}(\pm\mathbf{0.36})$
Circle	$0.29(\pm 0.13)$	$0.81(\pm 0.19)$	$\mathbf{0.97}(\pm\mathbf{0.00})$	$\mathbf{0.32}(\pm\mathbf{0.25})$
Sinewave	$0.86(\pm 0.03)$	$0.68(\pm 0.11)$	$\mathbf{0.99}(\pm\mathbf{0.00})$	$\mathbf{0.31}(\pm\mathbf{0.27})$
Polynomial	$0.82(\pm 0.03)$	$0.64(\pm 0.14)$	$\mathbf{0.99}(\pm\mathbf{0.00})$	$\mathbf{0.28}(\pm\mathbf{0.26})$

Table 1: Performance metrics on the out-of-distribution (OOD) region: area under the receiver operating characteristic (AUROC) and false positive rates at 95% recall (FPR@95). For each simulation 10000 samples were used for training the models over 45 Monte Carlo repetitions.

Discussion

kernel density polytopes enable OOD detection without sacrificing any in-distribution performance. Models augmented with such a method could achieve better overall performance than the original and produce posteriors that resemble human decisions. In the meantime, our code, including the package and the experiments in this manuscript, is available from https://github.com/neurodata/kdg

References

[1] Ertunc Erdil, Krishna Chaitanya, and Ender Konukoglu. Unsupervised out-of-distribution detection using kernel density estimation. *arXiv preprint arXiv:2006.10712*, 2020.

[2] Matthias Hein, Maksym Andriushchenko, and Julian Bitterwolf. Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 41–50, 2019.

Contact information: Jayanta Dey, 317 Clark Hall, Department of Biomedical Engineering, Jonhs Hopkins University, Baltimore, MD 21218 – Phone: 202–309–9942 – Email: *jdey4@jhmi.edu*; Web: *https://neurodata.io/*