Out-of-distribution Detection with Deep Nearest Neighbors

Yiyou Sun
UW-Madison

Yifei Ming
UW-Madison

Xiaojin Zhu
UW-Madison

Yixuan Li
UW-Madison
Problem Introduction
**Closed-world:**
Training and testing distributions **match**

**Open-world:**
Training and testing distributions **differ**

[Diagram showing the comparison between closed-world and open-world training and testing distributions.]
Training examples:
traffic signs

\[ p_i(x) = \frac{\exp(f_i(x))}{\sum_{j=1}^{N} \exp(f_j(x))}, \]

confidence: \( \max_i p_i \)

High confidence in classifying traffic signs.

Why is OOD Detection hard?
Test time: **out-of-distribution** example

Ideally: *Low confidence in predicting as traffic sign*
Neural Networks Can Be Over-confident to Out-of-distribution Examples

In-Distribution (ID) Samples

Out-of-distribution (OOD) Samples

[Nguyen et al. 2015]
Prior literature on distance-based OOD detection

Mahalanobis
[Lee et al. NeurIPS 2018]

CSI
[Tack et al. NeurIPS 2020]

SSD
[Sehwag et al. ICLR 2021]

In-Distribution (ID) Embeddings

Out-of-distribution (OOD) Embeddings

Far away
Can we leverage the non-parametric nearest neighbor approach for OOD detection?
Methodology
Problem Statement

• Supervised learning setting
  • input space: $X = \mathbb{R}^d$
  • label space: $Y = \{1, 2, \ldots, K\}$
  • a training dataset drawn i.i.d. from a joint distribution: $P = X \times Y$
  • training a neural network: $f(x; \theta)$

• In test time, test data can come from a different distribution
  • whose label set has not interaction with $Y$
  • should not be predicted by model

• OOD detection can be can be made via a level set estimation

\[ G_\lambda(x; f) = \begin{cases} 
\text{ID} & S(x; f) \geq \lambda \\
\text{OOD} & S(x; f) < \lambda \end{cases} \]
Set up

For all $x_i$ in the training data, collect embeddings: $Z_n = (z_1, z_2, \ldots, z_n)$
OOD Detection with k-NN Distance

1. Given a test sample \( \mathbf{x}^* \), we calculate feature vector \( \mathbf{z}^* = \phi(\mathbf{x}^*) / \| \phi(\mathbf{x}^*) \|_2 \)

2. Calculate the Euclidean distances \( \| \mathbf{z}_i - \mathbf{z}^* \|_2 \) with respect to embedding vectors \( \mathbf{z}_i \in \mathbb{Z}_n \)

3. Reorder \( \mathbb{Z}_n \) according to the increasing value of \( \| \mathbf{z}_i - \mathbf{z}^* \|_2 \), Denote the reordered data sequence as \( \mathbb{Z}'_n = (\mathbf{z}(1), \mathbf{z}(2), \ldots, \mathbf{z}(n)) \)

4. Calculate distance to the k-th nearest neighbor (k-NN): \( r_k(\mathbf{z}^*) = \| \mathbf{z}^* - \mathbf{z}(k) \|_2 \)

5. The decision function is given by \( G_\lambda(\mathbf{x}^*; \lambda) = \begin{cases} 
\text{ID} & -r_k(\mathbf{z}^*) \geq \lambda \\
\text{OOD} & -r_k(\mathbf{z}^*) < \lambda 
\end{cases} \)

(The threshold \( \lambda \) is typically chosen so that a high fraction of ID data (e.g., 95%) is correctly classified.)
Experiments
**Experiments Setting: Training Loss**

(1) **KNN:**
Cross-entropy Loss

\[ L_{ce} = - \sum_{i=1}^{C} y_i \log(p_i), \text{ for } C \text{ classes} \]

(2) **KNN+:**
Supervised Contrastive Loss

[Khosla, et al. 2020]

\[ L_{sup} = \sum_{i=1}^{2b} \frac{-1}{|P(i)|} \sum_{j \in P(i)} \log \frac{\exp(z_i \cdot z_j / \tau)}{\sum_{t=1, t \neq i}^{2b} \exp(z_i \cdot z_t / \tau)} \]
Experiments Setting: Backbone and Datasets

Model

ID
- CIFAR-10
- ImageNet-1k

OOD
- SVHN
- SUN
- Places
- Textures

ID
- iNaturalist
- SUN
- Places
- Textures
# K-NN Distance Achieves Superior Performance (CIFAR-10)

Table 1. Results on CIFAR-10. Comparison with competitive OOD detection methods. All methods are based on a discriminative model trained on ID data only, without using outlier data. ↑ indicates larger values are better and vice versa.

| Method       | SVHN | LSUN | Texture | Places365 | Average | ID ACC |
|--------------|------|------|---------|-----------|---------|--------|
|              | FPR↓ | FPR↓ | FPR↓    | FPR↓      |         |        |
|              | AUROC↑| AUROC↑| AUROC↑  | AUROC↑    |         |        |
| --------------|------|------|---------|-----------|---------|--------|
| Without Self-supervised Learning | | | | | | |
| MSP           | 59.66| 91.25| 45.21   | 93.80     | 54.57   | 92.12  | 66.45  | 88.50  | 62.46  | 88.64  | 57.67 | 90.86 | 94.21 |
| ODIN          | 20.93| 95.55| 7.26    | 98.53     | 33.17   | 94.65  | 56.40  | 86.21  | 63.04  | 86.57  | 36.16 | 92.30 | 94.21 |
| Energy        | 54.41| 91.22| 10.19   | 98.05     | 27.52   | 95.59  | 55.23  | 89.37  | 42.77  | 91.02  | 38.02 | 93.05 | 94.21 |
| GODIN         | 15.51| 96.60| 4.90    | 99.07     | 34.03   | 94.94  | 46.91  | 89.69  | 62.63  | 87.31  | 32.80 | 93.52 | 93.96 |
| Mahalanobis   | 9.24 | 97.80| 67.73   | 73.61     | 6.02    | 98.63  | 23.21  | 92.91  | 83.50  | 69.56  | 37.94 | 86.50 | 94.21 |
| KNN (ours)    | 24.53| 95.96| 25.29   | 95.69     | 25.55   | 95.26  | 27.57  | 94.71  | 50.90  | 89.14  | 30.77 | 94.15 | 94.21 |
| --------------|------|------|---------|-----------|---------|--------|
| With Self-supervised Learning | | | | | | |
| Rotation      | 53.31| 81.69| 24.80   | 93.49     | 20.53   | 93.26  | 18.78  | 94.65  | 56.47  | 78.47  | 34.78 | 88.31 | 93.65 |
| CSI           | 37.38| 94.69| 5.88    | 98.86     | 10.36   | 98.01  | 28.85  | 94.87  | 38.31  | 93.04  | 24.16 | 95.89 | 94.38 |
| SSD+          | 1.51 | 99.68| 6.09    | 98.48     | 33.60   | 95.16  | 12.98  | 97.70  | 28.41  | 94.72  | 16.52 | 97.15 | 95.07 |
| KNN+ (ours)   | 2.42 | 99.52| 1.78    | 99.48     | 20.06   | 96.74  | 8.09   | 98.56  | 23.02  | 95.36  | **11.07** | **97.93** | **95.07** |
Contrastively Learned Representation Helps More Distinguishable
K-NN Distance Achieves Superior Performance (ImageNet)

Table 4. Results on ImageNet. All methods are based on a model trained on ID data only (ImageNet-1k (Deng et al., 2009)). We report the OOD detection performance, along with the per-image inference time.

| Methods   | Inference time (ms) | iNaturalist | SUN | Places | Textures | Average | ID ACC |
|-----------|---------------------|-------------|-----|--------|----------|---------|--------|
|           |                     | FPR95 | AUROC | FPR95 | AUROC | FPR95 | AUROC | FPR95 | AUROC |          |
|           |                     | ↓     | ↑     | ↓     | ↑     | ↓     | ↑     | ↓     | ↑     |          |
| MSP       | 7.04                | 54.99 | 87.74 | 70.83 | 80.86 | 73.99 | 79.76 | 68.00 | 79.61 | 66.95     | 81.99  | 76.65 |
| ODIN      | 7.05                | 47.66 | 89.66 | 60.15 | 84.59 | 67.89 | 81.78 | 50.23 | 85.62 | 56.48     | 85.41  | 76.65 |
| Energy    | 7.04                | 55.72 | 89.95 | 59.26 | 85.89 | 64.92 | 82.86 | 53.72 | 85.99 | 58.41     | 86.17  | 76.65 |
| GODIN     | 7.04                | 61.91 | 85.40 | 60.83 | 85.60 | 63.70 | 83.81 | 77.85 | 73.27 | 66.07     | 82.02  | 70.43 |
| Mahalanobis| 35.83               | 97.00 | 52.65 | 98.50 | 42.41 | 98.40 | 41.79 | 55.80 | 85.01 | 87.43     | 55.47  | 76.65 |
| KNN (α = 100%) | 10.31             | 59.00 | 86.47 | 68.82 | 80.72 | 76.28 | 75.76 | 11.77 | 97.07 | 53.97     | 85.01  | 76.65 |
| KNN (α = 1%)  | 7.04               | 59.08 | 86.20 | 69.53 | 80.10 | 77.09 | 74.87 | 11.56 | 97.18 | 54.32     | 84.59  | 76.65 |
| Rotation  | 7.04                | 76.65 | 82.63 | 88.54 | 54.01 | 88.14 | 53.63 | 70.71 | 82.29 | 81.01     | 68.14  | 73.10 |
| SSD+      | 28.31               | 57.16 | 87.77 | 78.23 | 73.10 | 81.19 | 70.97 | 36.37 | 88.52 | 63.24     | 80.09  | 79.10 |
| KNN+ (α = 100%) | 10.47             | 30.18 | 94.89 | 48.99 | 88.63 | 59.15 | 84.71 | 15.55 | 95.40 | **38.47** | **90.91** | 79.10 |
| KNN+ (α = 1%)  | 7.04               | 30.83 | 94.72 | 48.91 | 88.40 | 60.02 | 84.62 | 16.97 | 94.45 | 39.18     | 90.55  | 79.10 |

CLNN outperforms the competitive baseline by **24.77%** in false positive rate (FPR95)
## K-NN Distance Has Similar Inference Speed

**Table 4. Results on ImageNet.** All methods are based on a model trained on ID data only (ImageNet-1k ([Deng et al., 2009])). We report the OOD detection performance, along with the per-image inference time.

| Methods          | Inference time (ms) | iNaturalist | SUN | Places | Textures | Average | ID ACC |
|------------------|---------------------|------------|-----|--------|----------|---------|--------|
|                  |                     |            | FPR95 | AUROC | FPR95 | AUROC | FPR95 | AUROC | FPR95 | AUROC | FPR95 | AUROC |        |
|                  | ↓   ↑                | ↓           | ↓    | ↑      | ↓      | ↑      | ↓    | ↑      | ↓    | ↑      | ↓    | ↑      | ↓    |
| MSP              | 7.04                | 54.99      | 87.74| 70.83  | 80.86  | 73.99  | 79.76| 68.00  | 79.61| 66.95  | 81.99| 76.65  |
| ODIN             | 7.05                | 47.66      | 89.66| 60.15  | 84.59  | 67.89  | 81.78| 50.23  | 85.62| 56.48  | 85.41| 76.65  |
| Energy           | 7.04                | 55.72      | 89.95| 59.26  | 85.89  | 64.92  | 82.86| 53.72  | 85.99| 58.41  | 86.17| 76.65  |
| GODIN            | 7.04                | 61.91      | 85.40| 60.83  | 85.60  | 63.70  | 83.81| 77.85  | 73.27| 66.07  | 82.02| 70.43  |
| Mahalanobis      | 35.83               | 97.00      | 52.65| 98.50  | 42.41  | 98.40  | 41.79| 55.80  | 85.01| 87.43  | 55.47| 76.65  |
| KNN (α = 100%)   | 10.31               | 59.00      | 86.47| 68.82  | 80.72  | 76.28  | 75.76| 11.77  | 97.07| 53.97  | 85.01| 76.65  |
| KNN (α = 1%)     | 7.04                | 59.08      | 86.20| 69.53  | 80.10  | 77.09  | 74.87| 11.56  | 97.18| 54.32  | 84.59| 76.65  |

|                 |                     |            | ↓   ↑ |    | ↓   ↑ |    | ↓   ↑ |    | ↓   ↑ |    | ↓   ↑ |    | ↓   ↑ |    | ↓   ↑ |
|------------------|---------------------|------------|------|-----|------|-----|------|-----|------|-----|------|-----|------|-----|------|
|                  |                      |            |      |    |      |    |      |    |      |    |      |    |      |    |      |
| **Without Self-supervised Learning** |                      |            |      |    |      |    |      |    |      |    |      |    |      |    |      |
| **With Self-supervised Learning** |                      |            |      |    |      |    |      |    |      |    |      |    |      |    |      |
| Rotation         | 7.04                | 76.65      | 82.63| 88.54| 54.01| 88.14| 53.63| 70.71| 82.29| 81.01| 68.14| 73.10|
| SSD+             | 28.31               | 57.16      | 87.77| 78.23| 73.10| 81.19| 70.97| 36.37| 88.52| 63.24| 80.09| 79.10|
| KNN+ (α = 100%)  | 10.47               | 30.18      | 94.89| 48.99| 88.63| 59.15| 84.71| 15.55| 95.40| **38.47** | 90.91| **79.10** |
| KNN+ (α = 1%)    | 7.04                | 30.83      | 94.72| 48.91| 88.40| 60.02| 84.62| 16.97| 94.45| 39.18| 90.55| **79.10** |
Other Experiments

1. Ablation on k and random sampling ratio
2. Nearest Neighbor Approach is competitive on hard OOD datasets
3. Comparison with other non-parametric methods
4. Nearest Neighbor Approach is competitive on ViT
5. KNN can be further boosted by activation rectification
6. Ablation on the penultimate layer and the projection layer
7. Ablation on the k-th and the averaged k-NN distance
8. Results on different architectures

(See more details in the paper)
Discussion and Theoretical Insight
Feature normalization is critical to KNN’s success.
Theoretical Insight

• **Set Up:** We consider OOD detection task as a special *binary classification* task, where the negative samples (OOD) are only available in the testing stage.

• **Main Result:** KNN-based OOD detector can reject inputs equivalent to the estimated Bayesian binary decision function.

\[
\text{Theorem 6.1.} \quad \text{If } \hat{p}_{\text{out}}(z_i) = \tilde{c}_0 \mathbf{1} \left\{ \hat{p}_{\text{in}}(z_i; k, n) < \frac{\beta \varepsilon \tilde{c}_0}{(1-\beta)(1-\varepsilon)} \right\}, \text{ and } \lambda = -\frac{m-1}{\beta \varepsilon c_b n \tilde{c}_0} \sqrt{(1-\beta)(1-\varepsilon)k}, \text{ we have} \\
\mathbf{1} \left\{ -r_k(z_i) \geq \lambda \right\} = \mathbf{1} \left\{ \hat{p}(g_i = 1 \mid z_i) \geq \beta \right\}
\]
Summary

1. We present the first study exploring and demonstrating the efficacy of non-parametric density estimation with nearest neighbors for OOD detection.

2. We demonstrate the superior performance of the KNN-based method on several OOD detection benchmarks, different model architectures.

3. We offer new insights on the key components to make KNN effective in practice, including feature normalization and a compact representation space.

4. We provide theoretical analysis, showing that KNN-based OOD detection can reject inputs equivalent to the Bayes optimal estimator.
Thank you!

https://github.com/deeplearning-wisc/knn-ood