Novelty Recognition with Gaussian Mixture Models

Stephen Scarano and Yi Wei

University of Massachusetts, Amherst

Abstract

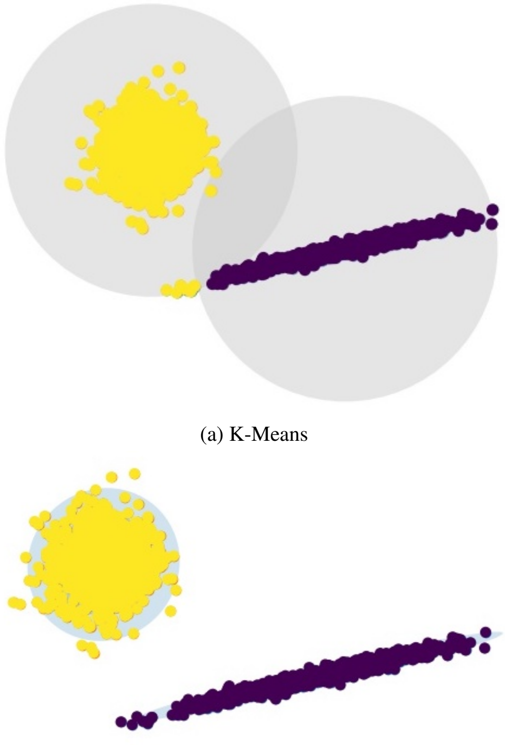
The last decade has witnessed tremendous progress in deep learning classification tasks; however, these models often rely on a notable "closed-set assumption": that classes observed in test-time are a subset of those observed during train-time. We present a new method, called NovelNet*, which integrates Gaussian Mixture Models into an arbitrary deep network architecture to discriminate instances not seen during training time. Our model consistently outperforms conventional techniques such as Nearest-Class Mean and Model Confidence, boosting accuracy by 12% in binary classification tasks.*

1. Introduction

Image classification has made significant progress since the incorporation of Neural Network approaches. However, a common assumption many models make is that at test-time the set of classes encountered would be a subset of all the classes the model observed at train-time [5]. In other words, Neural Networks do not know what they do not know.

Consider the case of an automated diagnostician agent trained on samples of k distinct pathologies. At test-time the agent should not only correctly distinguish between corresponding test samples, but ideally identify *unfamiliar* samples and signal caution. In this example, the agent's ability to identify novel samples could make the difference in discovering a new variant of a highly infectious virus, or simply avoiding a diagnosis with incomplete information. For high-risk deployment, machine learning models must competently acknowledge their limits and drop the assumption that classes seen at test-time are a subset of those observed during train-time

We are not the first to consider the problem of Open Set Recognition (OSR), and benefit from work spanning both traditional ML and deep learning disciplines. In that spirit, we integrate deep learning architectures with Gaussian Mixture Models (GMMs), which fit inputs to a predetermined number of normal distributions. We opt for GMMs since they have an intuitive notion of *distance* which still preserves complex relationships in the data. For clarification, consider the scenario shown in Figure 1.



(a) K-Means

(b) Gaussian Mixture Model

Figure 1. Classification of two distributions using K-Means and Gaussian Mixture Models that demonstrates the relative flexibility of the latter framework.

Gaussian Mixture Models are by no means universal approximators [8], but their descriptive flexibility exceeds that of a strict K-Means method [18]. In addition, their decisions are fundamentally explainable, which may be prioritized in high-impact contexts. The greatest theoretical hurdle—and the crux of our work here—is whether the feature space can be meaningfully approximated by a computationallyacceptable number of Gaussian clusters.

In short, we seek to identify and reject samples from some novel class, N, ideally without compromising model performance. For the sake of this paper, we consider performance in terms of standard accuracy and measure detection of N by novel-class recall.

2. Related Work

Reviews of Open Set Recognition (OSR) and Open World Learning (OWL) typically partition the input space into four quadrants [6, 9, 17]:

1. *Known-known classes* (KKCs): traditional data samples, seen at train time and exemplify one of k classes.
2. *Known-unknown classes* (KUCs): negative data samples defined by a *lack* of positive instances from other classes; i.e, background classes (think object detection [10]).
3. *Unknown-known classes* (UKCs): Classes with no available data samples during training, but with available semantic information.
4. *Unknown-unknown classes* (UUCs): Classes not encountered during training and also lack semantic information. Essentially, these are data instances that are completely unexpected.

Our work largely ignores UKCs which are a focus in Zero-Shot Learning: a process which leverages similarities between KKC and UKC attributes to classify UKC instances [6, 19]. Rather, our approach leverages contrasts between KKC and UUC attributes to discriminate the latter samples; i.e, that known class features and unknown class features will be notably distinct. On first glance, OSR resembles prior work on classification with a reject option [6, 7]; however, these frameworks still operate under the closed-set assumption—that all instance classes are seen during train-time (KKCs).

Since there is substantial overlap between model confidence and OSR, we review common methods here. These techniques can be organized into *ambiguity*-based and *distance*-based frameworks [7]. Ambiguity-methods reject instances whose model probability outputs are within some δ distance from one another [14], as shown in Figure 2.

The problem is formally defined assuming a predetermined reject-penalty, d, and probability function η [7]:

$$f_d(x) := \begin{cases} 1 & \text{if } \eta(x) < d \\ 0 & \text{if } \eta(x) > 1 - d \\ \text{REJECT} & \text{otherwise} \end{cases} .$$

Conversely, *Distance-methods* reject samples by thresholding some notion of *distance* between instances and target classes [6, 14]. As the authors mention in [6], empirically setting a threshold value inherently relies upon information determined from KKCs at train-time, which jeopardizes our generalizability to UUCs at test-time. Current approaches take inspiration from the k-nearest neighbors algorithm: classes are represented as a concatenation of all corresponding positive samples, which while computationally infeasible performs comparable to state-of-the-art methods [6, 15]. The authors of [15] introduce a similarly-inspired Nearest-Class-Mean (NCM) classifier, which instead represents classes by the mean feature-vector (learned at train-time) of its corresponding positive samples. Both methods rely on a metric of "distance" between instances and means, implemented by the Mahalanobis distance.

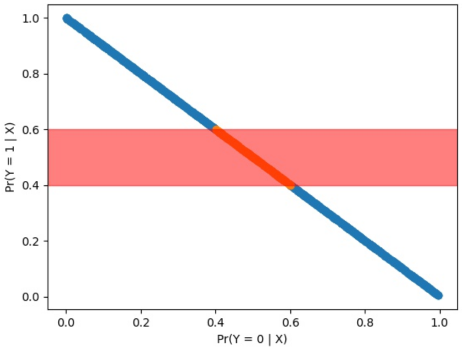


Figure 2. Model probability outputs where outputs of ambiguity within $\delta = 0.20$ are rejected

The Mahalanobis distance differs from Euclidean distance in that it considers the relationship between instances of a distribution. For any two n-dimensional vectors, $x_1, x_2 \in \mathbb{R}^n$ the Mahalanobis distance can be computed as

$$d(x_1, x_2) = \sqrt{(x_1 - x_2)^T C^{-1} (x_2 - x_2)};$$

where C is the covariance matrix of the distribution [3].

Aside from traditional methods, recent years have witnessed novelty detection using a deep-learning approach. The authors of [2] propose a new layer, denoted *OpenMax*, which estimates the probability that a given instance is outside the set of KKCs. Specifically, OpenMax adapts the SoftMax layer for the OSR setting where probabilities do not necessarily sum to 1. Interestingly, OpenMax is largely an ambiguity-based method, rejecting instances whose probability outputs do not exceed a confidence value, ϵ.

Our work consists of unsupervised learning on unlabeled data using a mixture of Gaussians. Each cluster is a multivariate Gaussian with a mean μ_k and covariance matrix C such that the complete model can be described as be-low [16]:

$$p(x_i|\theta) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x_i|\mu_k, C),$$

where $0 \leq \pi_k \leq 1$ are the mixing weights.

The model is fit by expectation maximization, and the proper number of components can be determined apriori using either the Akaike or Bayesian information criterion [1,16]. The former metric estimates the difference in probabilistic density between the true model, $f(x)$, and our model $p(x|\theta)$, penalized by model size. Bayesian information criteria, alternatively, computes the probability that the data-generating process $p(x|\theta)$ is the true model rather than a "good" approximation. Both metrics are defined and behave similarly, so for our work we apply AIC for its comparatively speedier computation-time [13] .

3. Methodology

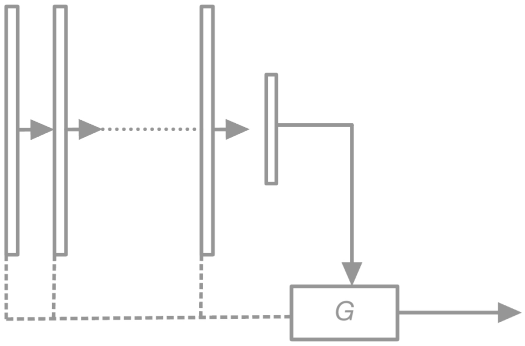


Figure 3. NovelNet "architecture" showing arbitrary an Neural Network and G weed-out component. Dashed lines (- - -) correspond to potential forward feeds into G.

To tackle this problem, we propose *NovelNet* as a new approach. Figure 3 displays a high-level diagram of *NovelNet*, which feeds network features into a G component that identifies and re-labels novel samples. During train-time, the model is trained conventionally (SGD), and only afterwards is G fit on its features. For clarity, G is a conventional gaussian mixture model (GMM) used as a filter on novel points. The GMM models feature relationships seen during train-time so as to discriminate against deviant points at test-time.

As shown in Figure 3, features may be extracted from any layer. In this manner, our approach draws upon model ambiguity methods, since we may choose to identify patterns directly out of the Softmax layer; however, this raises a prominent question: are unfamiliar Softmax probability relationships indicative of unfamiliar inputs? Ambiguity methods (see Section 2) compare output probabilities at face-value, strictly rejecting instances who "self-report" low confidence. In this work, we consider the probability outputs as any other feature representation. Assuming this hypothesis, we suspect that relationships determined by G have more descriptive power than those conservatively estimated by ambiguity methods—that is, since G models relationships dynamically rather than the former rules-based approach, it stands to leverage more information in detection tasks.

3.1. Training

Prior to train-time, we specify a max search-space of Gaussian components available $(n_2, n_3, ..., n_{max})$, since computation scales linearly with the number of clusters. Immediately after the network is trained, we extract features from a sample of the training set (the size of which is an adjustable hyperparameter) and perform a coarse search of the number of components with respect to AIC criterion (see Section 2).

Subsequent to the search, we must identify a maximum distance from the closest cluster, denoted δ, allowable to deem an instance familiar. Over a holdout set including novel samples, we calculate the mahalanobis distance between each sample and its closest cluster as determined by G. Next, we perform a second coarse search over the interval $[\delta_{min}, \delta_{max}]$ where $\delta_{min}, \delta_{max}$ are the minimum and maximum standard deviations encountered. Our experiments in Section 4 partition this range into 1000 equidistant values for search. After performing classification with our model, the threshold $\delta \in [\delta_{min}, \delta_{max}]$ that produces the highest metric of choice (we use standard accuracy) is adopted as the model distance threshold.

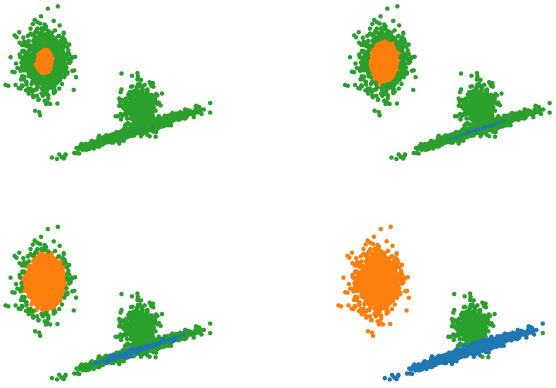


Figure 4. Example of increasing the distance threshold across the domain, where green points are UUC instances

It is a common point of confusion that each component in G maps to a particular known class, but this is not the case. Our Gaussian models are not intended to map to a KKC, rather, they map to *any combination* of KKCs—essentially acting as an "area of familiarity". In this respect we are computing a "familiarity-density" function, which we can approximate using GMMs. Given a sufficient number of components, we can approximate *any* density on $\mathbb{R}^n$ with a GMM of dimension n [16].

3.2. Testing

At test time, we perform the feature extraction as described above, and again calculate the distance between test samples and the nearest corresponding cluster. We then relabel instances that exceed the threshold, δ, determined at train-time. To clarify how accuracy is computed (as mentioned in Section 3.1), we first compute the conventional labels as determined by our base model and then replace entries whose corresponding distance exceeds δ with a novelty label (we use -1). The same feature layers should be used in both training and testing.

4. Experiments

The goal of the experiments in Sections 4.1, 4.2, 4.3, and 4.4 is to compare our method across

- **Feature Depth:** Since our flexible architecture enables feature extraction (or integration) at any layer in the network, it would be to our benefit to examine performance as it relates to depth. We perform this examination in Sections 4.1 and 4.2.

- **Distance Criteria:** Our method relies on a highly-interrelated sense of distance: the Mahalanobis distance explained in Section 2. We examine empirically in Section 4.3 whether modeling features as Gaussians is in practice any more helpful than a K-Means Euclidean distance to the closest cluster.

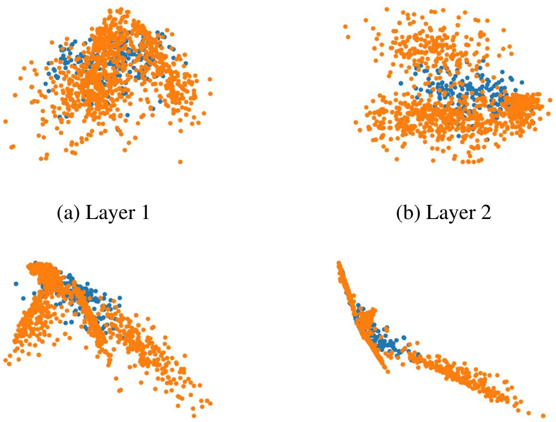
- **Dataset Complexity:** We intuit that modeling feature relationships is simpler for simpler tasks, and methods robust in one setting may not be successful in another. For this reason, we evaluate our method on two datasets of varying complexity in Section 4.4.

- **Existing Methods:** As mentioned earlier, we are not the first to explore the OSR problem. If so, it would be best to determine how our method relates to the model ambiguity and nearest-class mean methods explored in Section 2. We make this comparison across task complexity in 4.5.

We perform classification tasks across two datasets: MNIST and CIFAR-10 [4, 11]. For each, we study performance across feature depth and classification complexity (here defined as the number of KKCs to distinguish between). Unless otherwise stated, all experiments performed use a 4-layer–each of size 8– feed-forward neural network. Similarly the interval from which the number of G components is selected is identical across experiments: $|G| \in [2, 300]$. Additionally, our experiments are limited to one novel class, which is an unrealistic setting.

4.1. Modeling on Feature Layers

Fitting G directly to the intermediate features presents a tension between G computation and layer size: either the network must be "bottlenecked" with smaller layers, or computation and space in the G scales exponentially — since the covariance, C, is a matrix of size kxk (where k is the feature dimension).


(a) Layer 1

(b) Layer 2

(c) Layer 3

(d) Layer 4

Figure 5. 2D cross-sections of Novel and KKC features across depth

Figure 5 shows a comparison between KKC and novel feature maps visualized through 2D cross-sections. These plots present a second—arguably more obvious—hurdle: that the relationship between features may be too complex to model through Gaussian mixtures.

Since performance drops off with the complexity of the classification task (that is, increasing the KKCs), we provide a table with model performance across layers for a simple binary classification task. For evidence that performance gain drops with task complexity, see Sections 4.2, 4.3.

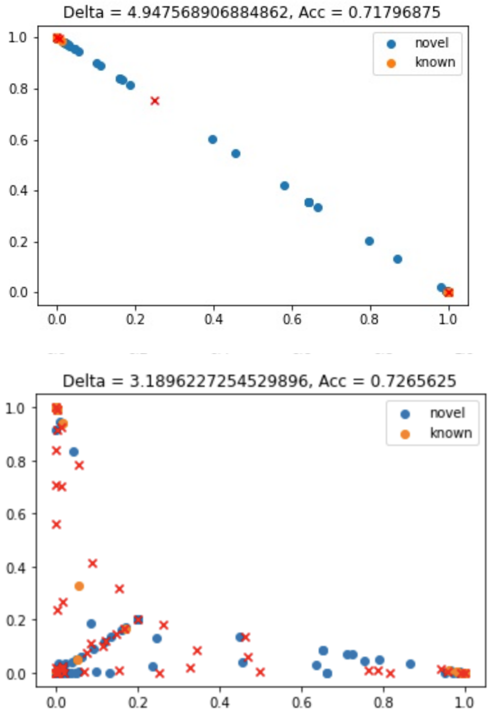


Figure 6. Ground Truth labels for Softmax probability outputs on MNIST data for both binary classification [top] and 5-way classification [down]. The red markers designate component centers determined at train-time.

Evaluation Across Feature Depth		
Layer	Performance Gain	Recall
Layer 1	7%	51%
Layer 2	3%	44%
Layer 3	2%	69%
Layer 4	2%	65%

Table 1. Performance gain (in accuracy) and novelty-class recall across network depth. Each point is the average of 20 trials.

4.2. Modeling on Class Probabilities

In Section 3 we considered the hypothesis that novel relationships between the class Softmax probabilities correspond to novel instances. Figure 6 shows promising qualitative evidence: novel instance probabilities tend to stray from the component locations determined at train-time.

Since modeling on the normalized probability scores resembles model confidence ambiguity-methods (see Section 2), we directly compare both methods across classification complexity. The confidence δ used is found via coarse search over the interval $[0, 1]$ on a holdout set and then applied at test-time.

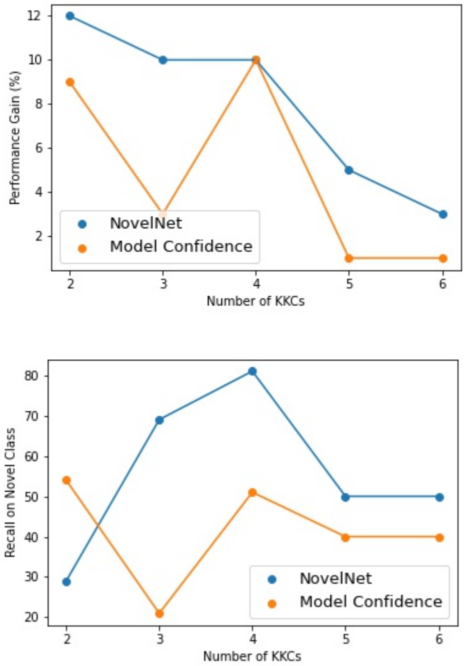


Figure 7. Performance gain comparison between NovelNet and the traditional ambiguity method for accuracy improvement [top] and recall on novel instances [bottom]. Each point is the average of 20 trials over the MNIST dataset.

Figure 7 compares performance gain in accuracy, which demonstrates the comparative strength of our approach. We also include a novelty recall comparison, whose high performance suggests that our method is most constrained by false positives rather than false negatives. Even when our model is achieving 80% recall on the novel class, we gain a mere 10% performance boost, which suggests mislabeled KKC instances. Additionally, the ambiguity method's conservative approach may shield it from this particular pitfall.

4.3. Euclidean vs. Mahalanobis Distance Criterion

Figure 8 compares Euclidean and Mahalanobis distance criteria on the MNIST dataset. Performance gain drops as the number of KKCs increases, but the Mahalanobis distance consistently outperforms Euclidean distance by 3 percentage points. Results appear to affirm the comparative flexibility of the former metric over its competitor.

These results are consistent with our hypothesis and intuition–that the interrelationships between component-instances is relevant to novelty classification. That said, as task complexity increase, the difference between the two models' performances deteriorates.

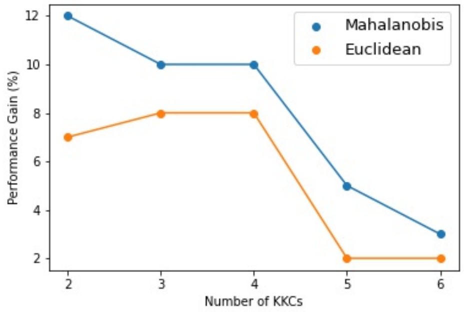


Figure 8. Performance gain comparison between Mahalanobis and Euclidean distance metrics on MNIST. Each point is the average of 20 trials.

4.4. Modeling Across Dataset Complexity

While modeling on the probability outputs of MNIST shows promise, it is unclear whether learning will generalize to more complex, semantic patterns. For this reason, we replicate our experiments in Section 4.2 using the CIFAR-10 dataset, which contains small images of higher-level constructs such as planes, cars etc [11]. We assume that these categories rely on higher-level understanding than MNIST, which is composed of relatively similar handwritten digits in grayscale.

Note that to achieve adequate performance on CIFAR-10, we have modified our network architecture to mimic ALexNET [12]: that is

1. CONV [1]
2. ReLU [1]
3. Average Pool [1]
4. CONV [2]
5. ReLU [2]
6. Average Pool [2]
7. Flatten
8. Linear (Feed-Forward) [1]
9. ReLU [3]
10. Linear (Feed-Forward) [2]
11. ReLU [4]
12. Softmax

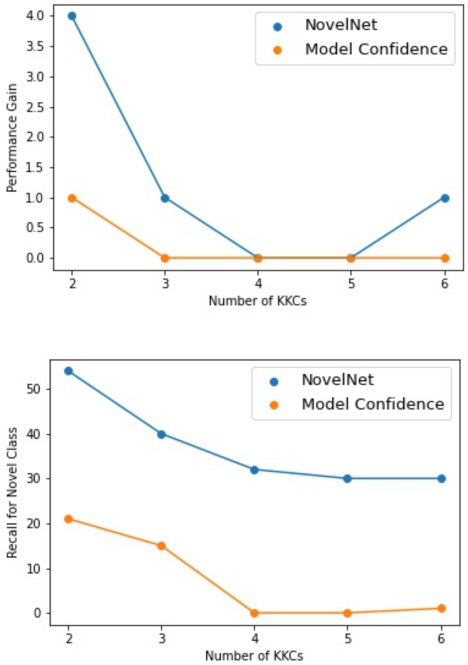


Figure 9. Performance gain in accuracy [top] and novelty-class recall [bottom] for CIFAR dataset. Each point is the average of 20 trials.

The results in Figure 9 seem to confirm our suspicions: performance drop-off in CIFAR for binary classification (2 KKCs) is comparable to MNIST 6-way classification (6 KKCs). These results present problems to the generalizability of our method.

4.5. Comparison with Nearest-Class Mean

Nearest-Class Mean (see Section 2) compares the extracted feature representations of new instances to a mean feature vector for all KKCs. For adequate comparison, we evaluate our method and Nearest-Class Mean using the same extracted feature values: first, we extract the relevant feature values and compute an average over all KKCs at train-time. At test-time, we subtract new instances from all means and map each instance to its lowest distance (of all potential k classes). If its respective distance exceeds a threshold (which is identified over a coarse search on a holdout-set), then we set the point as novel.

Performing classification on the MNIST dataset, results can be observed in Figure 10. Our method is highly competitive and notably less variable across task conditions (that is, the complexity of the classification task). Nearest-Class Mean does not generalize well to a deep network setting by comparison.

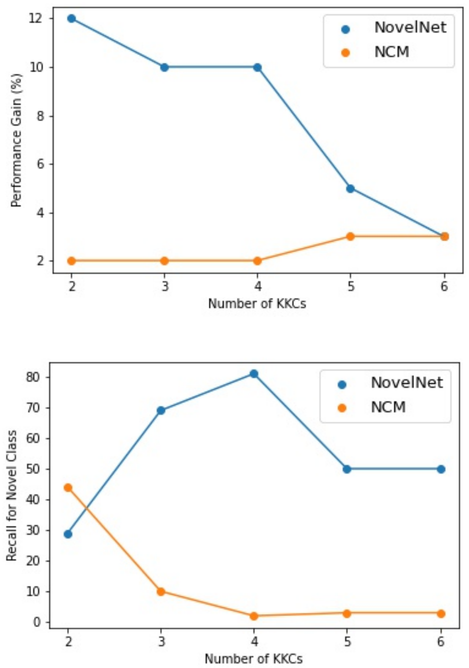


Figure 10. Performance gain and novel-class recall comparison for Nearest-Class Mean (NCM) and NovelNet on the MNIST dataset. Each point is the average of 20 trials.

5. Conclusions

Somewhat intuitively but nevertheless unexpectedly, we found modeling on the output probabilities far more successful and efficient than for intermediate layer features. We suspect that—as proposed in Section 1—the feature patterns are too complex to be modeled by a reasonable number of Gaussians. That said, our G component appears to outperform previous work in ambiguity methods for novelty detection, although our modeling suffers from an increased false-positive rate.

Dataset and task complexity are also open problems for our method: despite relative resilience in comparison to prior techniques, our performance gain drops off as a function of KKCs and qualitative difficulty. Ideally, future work should explore potential dimensionality-reduction methods or general network architectures to either stem or reverse performance drop. Our work accepts an arbitrary network architecture, but this may be too limiting for effective feature modeling; further, we explored feature extraction across depth, but did no experimentation with features *integrated* across depth. One might expand on our modeling of output probabilities by integrating either lower level features or traditional hand-crafted ones; however, deciding which to include—or how to combine them—is itself worthy of another paper.

We note here that the training process of our model is not ideal: There exist a number of unintuitive hyperparameter choices to make which exponentially scale the model selection process, and further work should make haste to identify heuristic or theoretical selection rules or criteria. Alternatively, our framework could benefit from integration into the end-to-end architecture itself, which could make the most of a future investigation.

Ideally, novelty detection methods would not only be successful, but also reasonably explainable. In part, that is also our motivation in goal. Yet, there may exist a tradeoff between explainable weeding-out of unfamiliar data and *accurate* weeding out of unfamiliar data. If so, researchers must consider heuristics for the appropriate application of OSR methods if communities and industry are to trust them. It may be the case that a less accurate diagnostician bot is preferable to a more accurate but uninterpretable one. Only a variety of OSR techniques can meet the highly-contextual needs of the true open world.

References

[1] Donald W.K. Andrews and Biao Lu. Consistent model and moment selection procedures for gmm estimation with application to dynamic panel data models. *Journal of Econometrics*, 101(1):123–164, 2001. 3

[2] Abhijit Bendale and Terrance E. Boult. Towards open set deep networks. *CoRR*, abs/1511.06233, 2015. 2

[3] R. De Maesschalck, D. Jouan-Rimbaud, and D.L. Massart. The mahalanobis distance. *Chemometrics and Intelligent Laboratory Systems*, 50(1):1–18, 2000. 2

[4] Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012. 4

[5] Zhen Fang, Jie Lu, Anjin Liu, Feng Liu, and Guangquan Zhang. Learning bounds for open-set learning, 2021. 1

[6] Chuanxing Geng, Sheng-Jun Huang, and Songcan Chen. Recent advances in open set recognition: A survey. *CoRR*, abs/1811.08581, 2018. 2

[7] Radu Herbei and Marten H. Wegkamp. Classification with reject option. *The Canadian Journal of Statistics / La Revue Canadienne de Statistique*, 34(4):709–721, 2006. 2

[8] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989. 1

[9] Mohsen Jafarzadeh, Akshay Raj Dhamija, Steve Cruz, Chunchun Li, Touqeer Ahmad, and Terrance E. Boult. Open-world learning without labels. *CoRR*, abs/2011.12906, 2020. 2

[10] K J Joseph, Salman Khan, Fahad Shahbaz Khan, and Vineeth N Balasubramanian. Towards open world object detection, 2021. 2

[11] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical Report 0, University of Toronto, Toronto, Ontario, 2009. 4, 6

[12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'12, page 1097–1105, Red Hook, NY, USA, 2012. Curran Associates Inc. 6

[13] Jouni Kuha. Aic and bic: Comparisons of assumptions and performance. *Sociological Methods Research - SOCIOL METHOD RES*, 33:188–229, 11 2004. 3

[14] Thomas C.W. Landgrebe, David M.J. Tax, Pavel Paclík, and Robert P.W. Duin. The interaction between classification and reject performance for distance-based reject-option classifiers. *Pattern Recognition Letters*, 27(8):908–917, 2006. ROC Analysis in Pattern Recognition. 2

[15] Thomas Mensink, Jakob Verbeek, Florent Perronnin, and Gabriela Csurka. Distance-based image classification: Generalizing to new classes at near-zero cost. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2624–2637, 2013. 2

[16] Kevin P. Murphy. *Machine learning : a probabilistic perspective*. MIT Press, Cambridge, Mass. [u.a.], 2013. 3, 4

[17] Jitendra Parmar, Satyendra Singh Chouhan, and Santosh Singh Rathore. Open-world machine learning: Applications, challenges, and opportunities. *CoRR*, abs/2105.13448, 2021. 2

[18] Eva Patel and Dharmender Singh Kushwaha. Clustering cloud workloads: K-means vs gaussian mixture model. *Procedia Computer Science*, 171:158–167, 2020. Third International Conference on Computing and Network Communications (CoCoNet'19). 1

[19] Yongqin Xian, Bernt Schiele, and Zeynep Akata. Zero-shot learning - the good, the bad and the ugly. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2