Open-Set Recognition with Gaussian Mixture Variational Autoencoders

Alexander Cao
Department of Industrial Engineering and Management Sciences
Northwestern University
Evanston, IL 60208
a-cao@u.northwestern.edu

Yuan Luo
Department of Preventive Medicine
Northwestern University
Chicago, IL 60611
yuan.luo@northwestern.edu

Diego Klabjan
Department of Industrial Engineering and Management Sciences
Northwestern University
Evanston, IL 60208
d-klabjan@northwestern.edu

Abstract

In inference, open-set classification is to either classify a sample into a known class from training or reject it as an unknown class. Existing deep open-set classifiers train explicit closed-set classifiers, in some cases disjointly utilizing reconstruction, which we find dilutes the latent representation's ability to distinguish unknown classes. In contrast, we train our model to cooperatively learn reconstruction and perform class-based clustering in the latent space. With this, our Gaussian mixture variational autoencoder (GMVAE) achieves more accurate and robust open-set classification results, with an average F1 improvement of 29.5%, through extensive experiments aided by analytical results.

1 Introduction

Until recently, nearly all classification algorithms have been designed for closed-set evaluation. This means that all testing classes are seen in training. However, real-world applications necessitate open-set evaluation where unknown classes, not seen in training, appear during testing. For instance, computer vision systems in self-driving cars must classify and navigate around many different objects. Given the countless number of such possible objects, it is infeasible for all classes to be seen in training [1]. Open-set recognition addresses this generalization of the classification task.

While there are several facets of open-set learning, in this paper we focus on training from C known classes for $(C+1)$-class classification. This $(C+1)$-th class catches all unknown test samples not belonging to any of the known classes. The training data has no unseen classes from class $C+1$. To this end, we present a novel supervised, Gaussian mixture variational autoencoder (GMVAE). The bottleneck latent layer simultaneously learns reconstruction and performs class-based clustering. This allows the latent representation to capture complementary structure and classifier information.

Preprint. Under review.

Furthermore, the latent layer has the explicit capability to form multiple subclusters per class. This challenges the implicit assumption made by many classification methods that a class's embedding is a convex set and thus is best represented by a single centroid [2, 3, 4]. This provides further flexibility in capturing complementary structure and classifier information

Our contributions are as follows. In §3, we derive GMVAE to learn the embedding and amend its objective function to make open-set recognition more amenable. We also present a new and simple open-set classification algorithm that utilizes an "uncertainty" threshold on the learned embedding. Following in §4, we present analytical results regarding the number of subclusters and the resulting heuristic procedure for identifying the appropriate number of subclusters in each class. Finally in §5, we conduct open-set classification experiments on three standard datasets. Our findings from experiments are two-fold. First, GMVAE outperforms the state-of-the-art deep open-set classifier both in terms of accuracy and robustness to an increasing number of unknown classes. Second, the use of extreme value theory (EVT) to infer class-belongingness [2, 4] may be ill-suited as we find that ours and another simple algorithm consistently beat it.

2 Related work

While closed-set classification has been well-studied, open-set recognition has been somewhat dormant with the majority of its work appearing in the last decade. Outlier or novelty detection is a precursor but is not generally concerned with distinguishing between the known classes [5, 6]. Earlier works that study $(C+1)$-class classification utilize, for example, SVM scores [7, 8] or sparse representation [9] to fit EVT-based densities to predict classes. The use of deep networks in open-set recognition appears even more recently in studies such as [2, 4]. Both use similar procedures of fitting EVT-based densities to the distances between a class's embedding and its centroid to approximate probability of class inclusion.

In this paper, our experimental results are benchmarked against the Classification-Reconstruction learning for Open-Set Recognition (CROSR) method [4]. We chose this particular benchmark as it achieves state-of-the-art open-set classification accuracies and it relies on a similar framework of dual reconstruction-classification learning.

We next summarize CROSR. The latent representation is a concatenation $[y, z]$ where y is the activation vector of a closed-set, softmax classifier and z is the reconstructive latent representation. To learn an effective y and z concurrently, [4] introduced Deep Hierarchical Reconstruction Nets (DHRNets). Conceptually, the DHRNet architecture is a deep classifier f with autoencoder networks $h_l, \widetilde{h}_l$ appended at the internal layers x_l. Thus, bottleneck representations can be extracted from multi-stage features of the classifier. The autoencoders' reconstructions then form a reverse network to reconstruct the original input. Mathematically, the main-body network $f(x) = (y, z)$ is comprised of

$$x_{l+1} = f_l(x_l) \quad l\text{-th layer of the DHRNet classifer,} \;\; z_l = h_l(x_l) \quad \text{encoder network for } l\text{-th layer}$$
$$\widetilde{x}_l = g_l(\widetilde{x}_{l+1} + \widetilde{h}_l(z_l)) \quad \text{decoder network } \widetilde{h}_l \text{ and reconstruction network } g_l \text{ for } l\text{-th layer}$$

where networks are a series of convolutions and up or down-sampling layers. For training, [4] minimizes the sum of the cross-entropy classification error and the L_2 reconstruction errors.

With latent representation $[y, z]$ in hand, CROSR applies EVT by fitting a Weibull distribution to the hypersphere defined by $d(x, C_i) = |[y, z] - \mu_i|_2$ where μ_i is the respective mean within class C_i. A proxy for probability of class inclusion is then given by $\mathbb{P}(x \in C_i) = 1 - \text{WeibullCDF}(d(x, C_i); \rho_i) = \exp\left\{-\left(\frac{d(x,C_i)}{\eta_i}\right)^{m_i}\right\}$ and thresholding is then used to classify a sample as "unknown." Here m_i and η_i are parameters of the distribution fitted from class C_i's training data.

In contrast to DHRNets, Gaussian mixture variational autoencoders from [10] are deep generative models which estimate the density of training data under assumptions on its latent prior. This could lead to more complex latent structures than in classification-based models. However, inference in this unsupervised setting is challenging, especially with open-set recognition. We address this by extending this deep generative model to supervised learning including capturing subclusters within classes.

3 Gaussian mixture variational autoencoders

In this section we present our complete, novel procedure for open-set recognition. It follows the same two phases as previous works: first, learn a latent representation to (sub)cluster known classes, and second, apply an open-set classification algorithm on that embedding. Our GMVAE model is an extension of the Gaussian mixture variational autoencoder presented in [10] and explained next.

Variational autoencoders (VAEs) assume data is generated from a uni-modal Gaussian prior. In [10], the authors instead choose a mixture of Gaussians as an intuitive extension. In order to maintain standard backpropagation via the reparametrisation trick, the standard VAE architecture was altered. The generative model, factorizing as $p_{\beta,\theta}(x,z,w,v) = p(w)p(v)p_\beta(z|w,v)p_\theta(x|z)$, generates a sample x from the latent variables z, w, and v with the following process

$$\begin{aligned} w \sim \mathcal{N}(0,I), \quad v \sim \text{Mult}(\pi) \\ (z|w,v) \sim \prod_{k=1}^{K} \mathcal{N}\left(\mu_k(w;\beta), \text{diag}\left(\sigma_k^2(w;\beta)\right)\right)^{v_k} \\ (x|z) \sim \mathcal{N}\left(\mu(z;\theta), \text{diag}\left(\sigma^2(z;\theta)\right)\right) \quad \text{or} \quad \mathcal{B}\left(\mu(z;\theta)\right) \end{aligned}$$

where K is the user-defined number of mixture components and $\mu_k(\cdot;\beta)$, $\sigma_k^2(\cdot;\beta)$, $\mu(\cdot;\theta)$, and $\sigma^2(\cdot;\theta)$ are neural networks parametrized by β and θ, respectively. The recognition model is then factorized as $q(z,w,v|x) = q_{\phi_z}(z|x)q_{\phi_w}(w|x)p_\beta(v|z,w)$ where ϕ_z and ϕ_w parametrize neural networks that output means and diagonal covariances of the Gaussian posterior variational distributions. Using Bayes' rule, the v-posterior term $p_\beta(v|z,w)$ can be written in terms of factors of the generative model. To train, the log-evidence lower bound (ELBO) $\mathbb{E}_{q(z,w,v|x)}\left[p_{\beta,\theta}(x,z,w,v)/q(z,w,v|x)\right]$ is maximized. In §3.1 and 3.2, we present the derivation and differences of our GMVAE. Finally we introduce our new open-set classification algorithm that utilizes an "uncertainty" threshold in §3.3.

3.1 Gaussian mixture variational autoencoders with multiple subclusters per class

Our GMVAE model nontrivially extends the unsupervised learning framework of [10] to essentially a Gaussian mixture prior for each class. For notation, there are C known classes with each class composed of K_c subclusters where $c = 1, 2, ..., C$. The samples $x \in \mathbb{R}^d$ and labels $y \in \mathbb{R}^C$ as one-hot vectors comprise the labeled, known data set $(x,y) \in \mathcal{X}$. The GMVAE's generative process $p_{\beta,\theta}(x,v,w,z|y) = p_\theta(x|z)p_\beta(z|w,y,v)p(w)p(v|y)$ is conditioned on class and given by

$$\begin{aligned} w \sim \mathcal{N}(0,I), \quad (v|y) \in \mathbb{R}^{K_c} \sim \text{Mult}(\pi(y)) \\ (z|w,y,v) \sim \prod_{c=1}^{C}\prod_{k=1}^{K_c} \mathcal{N}\left(\mu_{ck}(w;\beta), \text{diag}\left(\sigma_{ck}^2(w;\beta)\right)\right)^{y_c \cdot v_k} \\ (x|z) \sim \mathcal{B}\left(\mu(z;\theta)\right). \end{aligned}$$

It is common to take $\pi(y)$ to simply be uniform for each class. The recognition model is factorized as $q_\phi(v,w,z|x,y) = p_\beta(v|z,w,y)q_{\phi_w}(w|x,y)q_{\phi_z}(z|x)$ where $\phi = (\phi_x, \phi_w)$. We parametrize variational factors with networks ϕ that output mean and diagonal covariance of variational distributions and specify their form to be Gaussian posteriors:

$$\begin{aligned} (z|x) \sim \mathcal{N}\left(\mu(x;\phi_z), \text{diag}\left(\sigma^2(x;\phi_z)\right)\right) \\ (w|x,y) \sim \mathcal{N}\left(\mu(x,y;\phi_w), \text{diag}\left(\sigma^2(x,y;\phi_w)\right)\right). \end{aligned}$$

There is a p_β factor in the q_ϕ factorization because the p_β factor can be written in terms of generative factors, lowering the number of trainable parameters. Using Bayes', we can rewrite $p_\beta(v|z,w,y)$ as

$$p_\beta(v|z,w,y) = \frac{p_\beta(z|w,y,v)p(v|y)}{\sum_{v'} p_\beta(z|w,y,v')p(v'|y)}. \tag{1}$$

The details are provided in the supplementary material. Another benefit is that $p_\beta(v|z,w,y)$ can be computed for all v with simply one forward pass. The GMVAE's ELBO is then given by

$$\mathcal{L}(K) = \mathbb{E}_{q_\phi(v,w,z|x,y)}\left[\log \frac{p_{\beta,\theta}(x,v,w,z|y)}{q_\phi(v,w,z|x,y)}\right]$$

$= \mathbb{E}_{q_{\phi_z}(z|x)} \left[\log p_\theta(x|z)\right]$ (reconstruction)

$- \mathbb{E}_{q_{\phi_w}(w|x,y)q_{\phi_z}(z|x)} \left[\log q_{\phi_z}(z|x) - \sum_{j=1}^{K_c} p_\beta(v=j|z,w,y) \log p_\beta(z|w,y,v=j)\right]$ $\left(\begin{smallmatrix}\text{latent}\\ \text{covering}\end{smallmatrix}\right)$

$- KL(q_{\phi_w}(w|x,y)||p(w))$ (w-prior)

$- \mathbb{E}_{q_{\phi_w}(w|x,y)q_{\phi_z}(z|x)} \left[KL(p_\beta(v|z,w,y)||p(v|y))\right]$ (subcluster v-prior).

Since $K = (K_1, K_2, ..., K_C)$ is user-defined, the ELBO dependence on K is made explicit and used later in the analyses. The reconstruction term promotes a latent representation meaningful to reconstruct the samples. The latent covering term attempts to subcluster the latent representation based on classes. The w-prior and subcluster v-prior terms drive those posteriors closer to their respective priors.

3.2 Modification of the ELBO: removing v-prior

In this subsection, we propose removing the v-prior term from the original ELBO to make GM-VAE more amenable to open-set recognition for two reasons. First, minimizing the v-prior term $\mathbb{E}_{q_{\phi_w}(w|x,y)q_{\phi_z}(z|x)} \left[KL(p_\beta(v|z,w,y)||p(v|y))\right]$ is in direct conflict with the goal of distinct subclustering within a class. Our goal is to create disjoint subclusters in a class's latent representation so as to further provide reconstruction more flexibility and alleviate the assumption that a class's embedding is a convex set. However, notice that the v-prior term is minimized when $p_\beta(v|z,w,y) = p(v|y)$ for every z, w, and y. Combined with (1) and a uniform $p(v|y)$, this in turn implies that $p_\beta(z|w,y,v=i) = p_\beta(z|w,y,v=j)$ for every w, y, i, and j. Equivalent generative model distributions leads to mode collapse in the latent subclusters due to the maximization of the latent covering term. Put differently, the v-prior term discourages one-hot subcluster v posteriors. However, this is exactly what is needed to robustly identify subclusters.

Second, as proven in Proposition 2 in §4, without the v-prior term the optimal GMVAE loss for $C = 1$ is non-increasing with respect to K. This is an analytical result which provides a heuristic procedure for identifying the appropriate number of subclusters K_c to use for each class. Given these two reasons, for all the experiments in §5, we used the following modified ELBO:

$$\mathcal{L}_{\text{no } v\text{-prior}}(K) = \mathbb{E}_{q_{\phi_z}(z|x)} \left[\log p_\theta(x|z)\right] - KL(q_{\phi_w}(w|x,y)||p(w)) - \mathbb{E}_{q_{\phi_w}(w|x,y)q_{\phi_z}(z|x)} \left[\log q_{\phi_z}(z|x) - \sum_{j=1}^{K_c} p_\beta(v=j|z,w,y) \log p_\beta(z|w,y,v=j)\right].$$

In a sense, it is as if we do not impose a prior on the subcluster distributions. While we could have also negated the v-prior term, simply removing it actually yields the best experimental results.

3.3 Open-set classification algorithms

With recent literature in open-set recognition, it has nearly become universal to model class-belongingness by fitting a Weibull distribution to the inlier distances between a class's latent representations and its centroid [2, 3, 4]. Indeed, the benchmark method CROSR [4] achieves state-of-the-art accuracies through this EVT framework. However our experiments demonstrate that two much simpler algorithms can significantly outperform CROSR's EVT-based classification algorithm. While fitting a distribution to the inlier distances may be an effective way to conform a hypersphere-density around a class's centroid, we believe it is much too sensitive to these inliers. If a class has samples whose latent representations are "misclassified" and far away from its centroid, then the resulting distribution fit will be extremely skewed and render inaccurate predictions. This possible negative effect is severely magnified for embeddings that do not optimize for low intra-spread within each class. For instance, CROSR's embedding is composed of the closed-set, softmax classifier's activation vector; this encourages elements of that vector to tend towards positive and negative infinity.

Next we present the two simple open-set classification algorithms we implemented. While GMVAE outputs a Gaussian distribution in latent space, we simply choose the mean $\mu(x;\phi_z)$ as the effective

latent representation. Algorithm 1 is derived from the so-called outlier score from [3] but is most aptly described as nearest centroid thresholding on distance to the nearest centroid. This algorithm is modified to incorporate multiple subclusters per class.

Algorithm 1: Nearest centroid thresholding on distance to the nearest centroid

Input: Training samples $\mathcal{X}_c$ for each known class $c = 1, 2, ..., C$ and test sample $\widehat{x}$
1. For each class c, compute K_c centroids of $\mu(\mathcal{X}_c; \phi_z)$ using k-means clustering. Denote centroid $\overline{z}_{ck}$ as k-th centroid of class c.
2. Let $(c^*, k^*) = \arg\min_{c,k} ||\mu(\widehat{x}; \phi_z) - \overline{z}_{ck}||_2$ and $d = \min_{c,k} ||\mu(\widehat{x}; \phi_z) - \overline{z}_{ck}||_2$
3. If $d < \tau$, predict class as c^*; else, predict class as unknown $C + 1$

Experimental results show that thresholding on distance to the nearest centroid more robustly fits a hypersphere around the respective centroid. However, a similar shortcoming shared with CROSR's EVT method is that distance is a rotationally symmetric measure. It does not include any sense of orientation. We stand to reason that in any nearest centroid-based algorithm, the open space between centroids poses the most risk from an open-set classification standpoint. This leads into the second algorithm which utilizes a novel threshold on an "uncertainty" quantity U. We define U as the ratio between the distance to the nearest centroid to the average distance to all other centroids. So if $U = 1$, the test sample's latent representation is equidistant from all centroids which can be interpreted as unclassifiable. If $U = 0$, the test sample's latent representation is exactly a centroid meaning there is no ambiguity in classification. In this way, Algorithm 2 includes a notion of orientation between centroids as U penalizes the open space directly between centroids more heavily. This is reminiscent of the nearest neighbors distance ratio of [11].

Algorithm 2: Nearest centroid thresholding on uncertainty U

Input: Training samples $\mathcal{X}_c$ for each known class $c = 1, 2, ..., C$ and test sample $\widehat{x}$
1. For each class c, compute K_c centroids of $\mu(\mathcal{X}_c; \phi_z)$ using k-means clustering. Denote centroid $\overline{z}_{ck}$ as k-th centroid of class c.
2. Let $(c^*, k^*) = \arg\min_{c,k} ||\mu(\widehat{x}; \phi_z) - \overline{z}_{ck}||_2$, $N = \sum_{c=1}^{C} K_c$, and

$$U = \frac{\min_{c,k} ||\mu(\widehat{x}; \phi_z) - \overline{z}_{ck}||_2}{\frac{1}{N-1} \sum_{(c,k) \neq (c^*, k^*)} ||\mu(\widehat{x}; \phi_z) - \overline{z}_{ck}||_2}$$

3. If $U < \tau$, predict class as c^*; else, predict class as unknown $C + 1$

4 Identifying the number of subclusters in each class

Since the number of subclusters in each class is user-defined, identifying the appropriate number is critical for model usage. A natural procedure that immediately arises is to iteratively apply GMVAE to each class's data alone for an increasing number of subclusters K_c. Given the reconstruction and clustering objectives, the empirical model losses should naturally inform us of the optimal number of subclusters. This is akin to increasing k in k-means clustering and studying the resulting inertia plot. To this end, in this section we first present analytical results regarding the effect of $K = K_1$ on the optimal $C = 1$ (single class) GMVAE loss. This then leads to our heuristic procedure for identifying the ideal number of subclusters in each class.

With two unrestrictive neural network assumptions, we are able to prove two propositions regarding the effect of K on the optimal GMVAE loss. The assumptions and proofs can be found in the supplementary material. The first proposition demonstrates that when there truly is only one subcluster within a class, and we know its distribution, then the optimal loss is constant with respect to K. Since $C = 1$, we write x instead of (x, y).

Proposition 1. *Let us assume that $x \in \mathcal{X}$ is distributed as $x \sim p_{data} = \mathcal{B}(\mu_x)$, $C = 1$, and Assumption 1 holds. Then the optimal GMVAE loss is constant with respect to K. In fact, we have that* $\min -\mathbb{E}_{\mathcal{X}}[\mathcal{L}(K)] = -\mathbb{E}_{\mathcal{X}}[\log p_{data}]$ *for every $K \geq 1$ and a globally optimal solution reads*

$$\mu(x; \phi_z^*) = \mu_{c=1,k}(w; \beta^*) = \mu_z, \quad \sigma^2(x; \phi_z^*) = \sigma^2_{c=1,k}(w; \beta^*) = \sigma_z^2$$

$$\mu(x, y; \phi_w^*) = \vec{0}, \quad \sigma^2(x, y; \phi_w^*) = \vec{1}, \quad \mu(z; \theta^*) = \mu_x$$

for any constant vectors μ_z, σ_z.

The second proposition makes no data assumptions but shows that the optimal loss is quasi-non-increasing with respect to K. However, it does demonstrates the uniform lower-boundedness of sequential differences.

Proposition 2. *Let us assume* $C = 1$*, Assumptions 1 and 2 hold, and that* $p(v|y = 1)$ *is uniform in the appropriate dimension. We have* $\min\{-\mathbb{E}_{\mathcal{X}}[\mathcal{L}(K; \phi_z, \phi_w, \beta, \theta)]\} - \min\{-\mathbb{E}_{\mathcal{X}}[\mathcal{L}(K+1; \phi_z, \phi_w, \beta, \theta)]\} \geq \epsilon_K$ *where* $-\log 2 \leq \log(K/(K+1)) \leq \epsilon_K$ *for all* K.

These proofs do not inform us on the transient dynamics of training nor even reaching the global optimum. As such, in the following experimental results section, we apply these propositions in practice by comparing the latent covering loss given reconstruction loss for each $K \geq 1$. This answers: How well does K subclusters "cover" the embedding for a given reconstruction level? When the latent covering loss's decreases begins to diminish, then it is an indication that additional subclusters were only marginally beneficial and perhaps should not be included.

5 Experimental results

The experimental results demonstrate several findings. First, EVT-based open-set classification may not be appropriate as simple nearest centroid procedures consistently beat it. Second, even without the added benefit of subclustering, GMVAE for $K = \vec{1}$ often leads to a latent representation more amenable for open-set recognition compared to CROSR. Finally, subclustering within classes represents a means of bolstering dual supervised-reconstruction embeddings.

Each dataset has the following composition. The training data has only labeled samples from the C classes. The validation set has samples from the same C classes and samples from additional Q classes which are all treated as class $C + 1$. The validation set is used to determine the threshold τ. Finally, the test set has the same distribution as the validation set.

For each of the experiments below, we perform an ablation study. Four combinations of model and classification algorithms were applied: (i) CROSR with CROSR's EVT (CROSR+EVT), (ii) CROSR with Algorithm 1 (CROSR+NC-D), (iii) GMVAE with Algorithm 1 (GMVAE+NC-D), and (iv) GMVAE with Algorithm 2 (GMVAE+NC-U). CROSR+NC-D and GMVAE+NC-D are meant to directly compare latent representations' amenability to open-set recognition. We did not study CROSR with Algorithm 2 because our "uncertainty" measure is really a proxy for confidence and it has been shown that it is erroneous to equate softmax classifiers with confidence [12]. Correctly adapting "uncertainty" to CROSR is out of this paper's scope. For each combination, we calculate the optimal macro-averaged F1 scores (optimizing threshold τ on the validation set) for an increasing number Q of unknown classes (and samples). However, for actual inference in model serving, we would not have any prior knowledge on the known and unknown composition of the test set. Accordingly, we also compute the F1 scores using the mean of those optimal thresholds. The first two experiments are for $K = \vec{1}$ and in the last two, we manufacture classes with multiple subclusters to apply $K = (2, 2)$.

We optimize over the training set using Adam until the loss, evaluated on the known validation set, plateaus. For the MNIST and Fashion MNIST datasets (grayscale images), the reconstruction distribution used was the unnormalized, continuous Bernoulli distribution. For the CIFAR-10 dataset (RGB images), a truncated $[0, 1]$ Gaussian models the reconstruction. The latent space dimension of z equals 10, 50, 5, and 20 for the four experiments. We will publish our code upon acceptance of this paper.

5.1 Fashion MNIST withholding 4 classes

The six known classes are t-shirts/tops, trousers, pullovers, dresses, coats, and shirts, while the four unknown classes are sandals, sneakers, ankle boots, and bags. Fashion MNIST's standard training set is randomly split into the validation set (6,000 samples of known classes and 4,000 samples of unknown classes) and training set (30,000 samples). Fashion MNIST's standard testing set (10,000 samples) is kept the same. We use the same CROSR network architecture as [4] for their MNIST

experiment. The $K = \vec{1}$ GMVAE network architectures are given in Table 1. The θ network is always the mirrored ϕ_z network. Convolution parameters are denoted by "⟨number of layers⟩ conv⟨kernel size⟩-⟨number of channels⟩" and max pooling strides are denoted by "maxpool-⟨stride⟩." ReLU activations follow each weight layer.

Table 1: Network architectures for $K = \vec{1}$ GMVAE used in Fashion MNIST experiment.

ϕ_z	ϕ_w	β
Input: x	Input: x, y	Input: w
2 conv3-20	1 conv3-10 on x only	FC-50
maxpool-2	maxpool-4	FC-50
2 conv3-50	Concatenate with y	FC-120 ($6 \times 2 \times \dim(z)$)
maxpool-2	FC-20 ($2 \times \dim(w)$)	
FC-500		
FC-100		
FC-20 ($2 \times \dim(z)$)		

F1 scores are plotted in Figure 1. On the left are the optimal threshold F1 scores versus the number of unknown classes Q. While GMVAE is not as accurate in the closed-set regime, it outperforms CROSR as Q increases. CROSR's open-set accuracies, in turn, quickly diminish as Q increases, CROSR+EVT in particular. On the right is the mean threshold F1 scores where GMVAE's F1 scores are still more robust to increasing Q. For $Q \geq 1$ and using the mean thresholds, GMVAE+NC-U F1 scores are on average 3.9% greater than those of CROSR+NC-D.

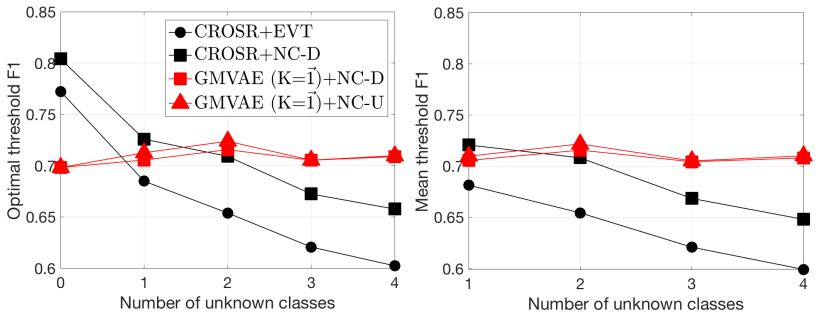

Figure 1: Fashion MNIST open-set test F1 scores.

5.2 CIFAR-10 withholding 4 classes

The six known classes are airplanes, automobiles, birds, cats, deer, and dogs. The four unknown classes are frogs, horses, ships, and trucks. CIFAR-10's standard training set is randomly split into the validation set (6,000 samples of known classes and 4,000 samples of unknown classes) and training set (24,000 samples). CIFAR-10's standard testing set (10,000 samples) is kept the same. For both CIFAR-10 experiments, we use the same CROSR architecture as [4] for their CIFAR-10 experiment. The $K = \vec{1}$ GMVAE network architectures are given in Table 2. For both CIFAR-10 experiments, the ϕ_z network is pretrained on the known classes and those weights are then frozen.

F1 scores are plotted in Figure 2. GMVAE consistently beats CROSR and again CROSR+EVT performs worst. Algorithm 2 augments GMVAE and we deduce this is because unknown CIFAR-10 samples are more difficult to distinguish and thus more likely to be embedded to the interior of known latent clusters where "uncertainty" has more influence. For $Q \geq 1$ and using the mean thresholds, GMVAE+NC-U F1 scores are on average 61.3% greater than those of CROSR+NC-D.

We present a t-SNE plot of the CROSR latent representation components in Figure 3 to bring into question the explicit use of classifier activation vectors in an open-set recognition embedding. We see that the reconstruction latent variable z does little to cluster the known classes and so open-set classification is dominated by the known classifier's activation vector y. We believe this to be the underlying reason why CROSR's F1 scores in Figures 2 and 8 are so poor.

Table 2: Network architectures for $K = \vec{1}$ GMVAE used in the first CIFAR-10 experiment.

ϕ_z	ϕ_w	β
Input: x	Input: x, y	Input: w
2 conv3-64	1 conv3-10 on x only	FC-100
maxpool-2	maxpool-4	FC-100
2 conv3-128	Concatenate with y	FC-600 ($6 \times 2 \times \dim(z)$)
maxpool-2	FC-100 ($2 \times \dim(w)$)	
4 conv3-256		
maxpool-2		
FC-1000		
FC-500		
FC-100 ($2 \times \dim(z)$)		

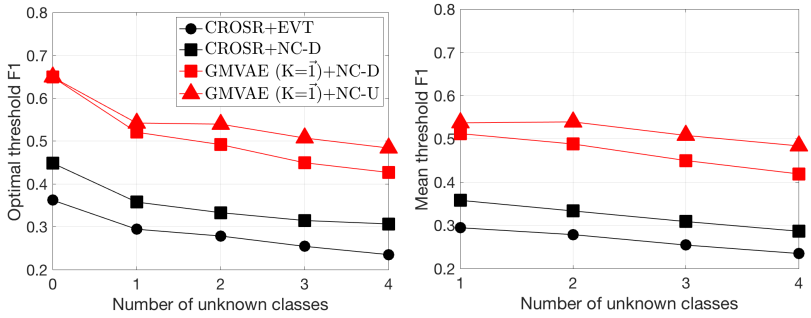

Figure 2: $K = \vec{1}$ CIFAR-10 open-set test F1 scores.

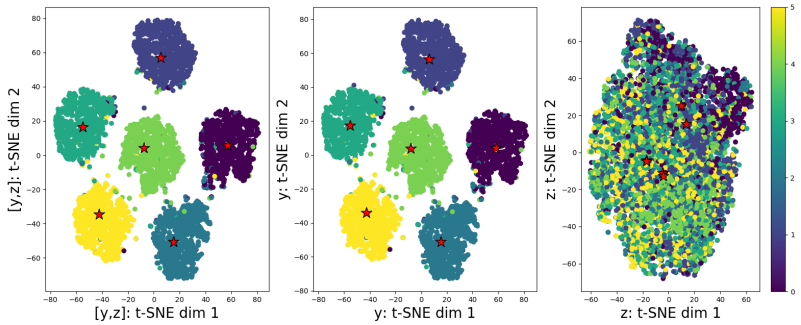

Figure 3: t-SNE plot of (left) both components $[y, z]$, (center) only y, and (right) only z of CROSR's training latent representations for the first CIFAR-10 experiment. Stars are the respective component's class centroids.

In contrast to CROSR, GMVAE's latent representation $\mu(x; \phi_z)$ in Figure 4 separates classes better (in comparison to the right figure in Figure 3). GMVAE's embedding is able to effectively capture both class and reconstruction information simultaneously, leading to more amenable open-set recognition. As CIFAR-10 images are highly hetergeneous within classes, we expect class overlap from reconstruction.

5.3 MNIST with "even" and "odd" classes

The two known classes are "even," comprised of digits 0 and 2, and "odd," comprised of digits 1 and 3. The six unknown classes are digits 4 and greater. MNIST's standard training set is randomly split into the validation set (4,000 samples of known classes and 6,000 samples of unknown classes) and training set (about 18,000 samples). MNIST's standard testing set (10,000 samples) is kept the same. We use the same CROSR architecture as [4] for their MNIST experiment. The GMVAE network architectures are given in Table 3.

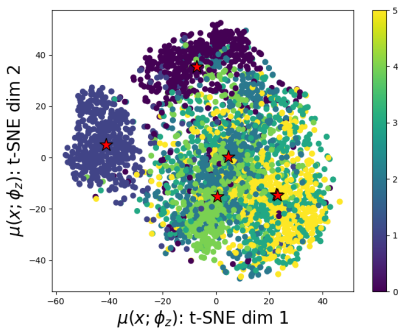

Figure 4: t-SNE plot of $\mu(x;\phi_z)$ of GMVAE's training latent representations for the first CIFAR-10 experiment. Stars are the respective component's class centroids.

Table 3: Network architectures for GMVAE used in the "even" and "odd" MNIST experiment.

ϕ_z	ϕ_w	β
Input: x	Input: x, y	Input: w
1 conv3-20	1 conv3-10 on x only	FC-20
maxpool-2	maxpool-4	FC-20
1 conv3-50	Concatenate with y	FC-$(2 \times \sum_c K_c \times \dim(z))$
maxpool-2	FC-10 $(2 \times \dim(w))$	
FC-500		
FC-100		
FC-10 $(2 \times \dim(z))$		

This is a clearcut example where each class has two subclusters. To determine that $K = (2, 2)$ is indeed the optimal GMVAE selection, we implement the procedure in §4 in Figure 5. On the left, the mean difference between the $K = 1$ and $K = 2$ latent covering loss is 0.86 while the mean difference between $K = 2$ and $K = 3$ is 0.22. This is indicative of two true subclusters within "even." Similarly on the right, the mean difference between $K = 1$ and $K = 2$ latent covering loss is 1.23 while the mean difference between $K = 2$ and $K = 3$ is -0.09. This is again indicative of two true subclusters within "odd." For these plots, the early epochs are truncated.

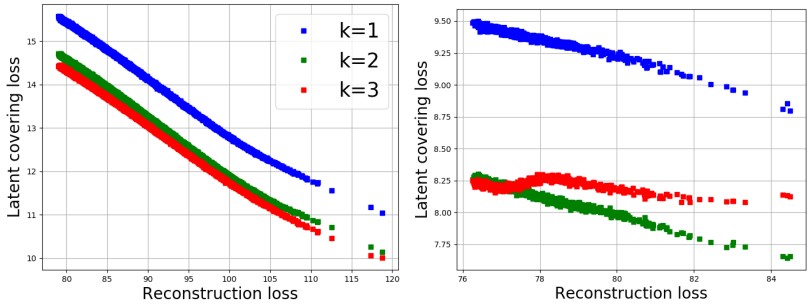

Figure 5: The latent covering loss plotted against reconstruction loss for increasing K for the (left) "even" and (right) "odd" classes of MNIST.

F1 scores are plotted in Figure 6. Similar to the Fashion MNIST experiment, GMVAE+NC-D begins to outperform CROSR+NC-D when the number of unknown classes $Q \geq 3$. However slightly, CROSR+EVT again performs worst. There is a significant increase in GMVAE open-set accuracy and robustness to increasing Q from utilizing the "uncertainty" threshold. This algorithm complements the use of class subclusters as unknown classes' latent representations are strategically more likely embedded in the open space between centroids where U is larger. For $Q \geq 1$ and using the mean thresholds, GMVAE+NC-U F1 scores are on average 19.9% greater than those of CROSR+NC-D.

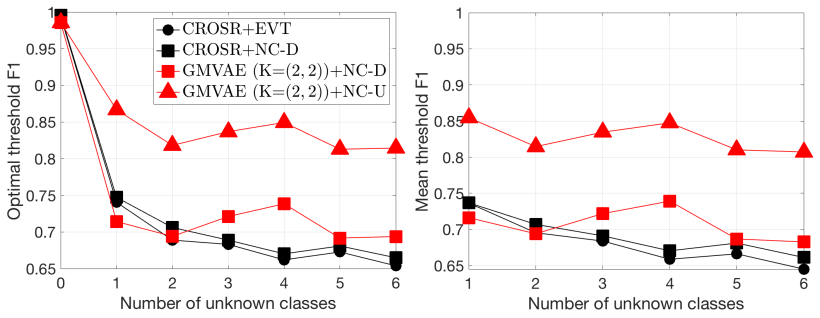

Figure 6: "Even" and "odd" MNIST open-set test F1 scores.

5.4 CIFAR-10 with "animals" and "vehicles" classes

The two known classes are "animals," comprised of cats and dogs, and "vehicles," comprised of cars and trucks. The unknown classes are the other 6 classes. CIFAR-10's standard training set is randomly split into the validation set (4,000 samples of known classes and 6,000 samples of unknown classes) and training set (16,000 samples). CIFAR-10's standard testing set (10,000 samples) is kept the same. The GMVAE network architectures are given in Table 4.

Table 4: Network architectures for GMVAE used in the second CIFAR-10 experiment with "animals" and "vehicles".

ϕ_z	ϕ_w	β
Input: x	Input: x, y	Input: w
2 conv3-64	1 conv3-10 on x only	FC-50
maxpool-2	maxpool-4	FC-50
2 conv3-128	Concatenate with y	FC-$(2 \times \sum_c K_c \times \dim(z))$
maxpool-2	FC-40 $(2 \times \dim(w))$	
4 conv3-256		
maxpool-2		
FC-1000		
FC-500		
FC-40 $(2 \times \dim(z))$		

This is another clearcut example where each class has two subclusters. We again implement the procedure in §4 to determine that $K = (2, 2)$ is indeed the optimal GMVAE selection and show the results in Figure 7. On the left, the mean difference between $K = 1$ and $K = 2$ latent covering loss is 1.31 while the mean difference between $K = 2$ and $K = 3$ is -1.47. This is indicative of two true subclusters within "animals." On the right, the mean difference between $K = 1$ and $K = 2$ latent covering loss is 0.82 while the mean difference between $K = 2$ and $K = 3$ is 0.5. This is moderately indicative of two true subclusters within "vehicles."

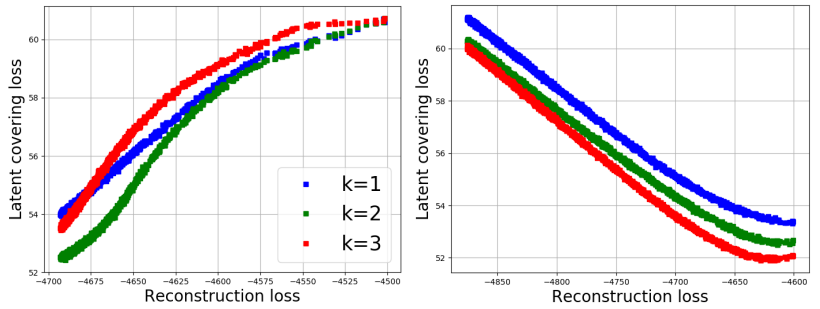

Figure 7: The latent covering loss plotted against reconstruction loss for increasing K for the (left) "animals" and (right) "vehicles" classes of CIFAR-10.

F1 scores are plotted in Figure 8. Discussed in §3.3, as a result of CROSR's softmax classifier, the centroids are not representative and thus its closed-set classification suffers. While CROSR+EVT eventually recovers to near similar performance as GMVAE+NC-D for $Q \geq 4$, the difference is stark for a low number of unknown classes. Again, because of the class subclusters, the "uncertainty" threshold provides a significant increase in open-set recognition capability. For $Q \geq 1$ and using the mean thresholds, GMVAE+NC-U F1 scores are on average 33% greater than those of CROSR+EVT.

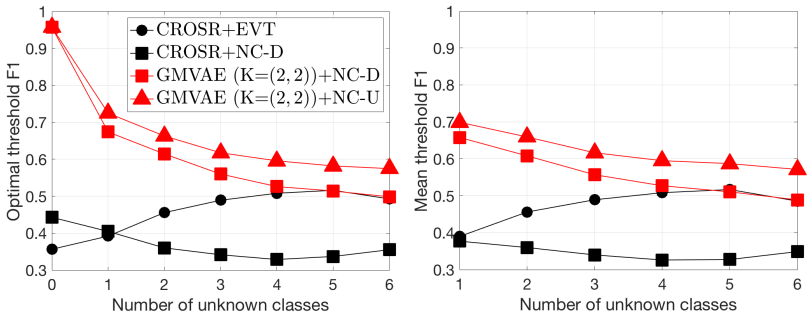

Figure 8: $K = (2, 2)$ CIFAR-10 open-set test F1 scores.

Acknowledgments and Disclosure of Funding

The work of the first author is supported by the Predoctoral Training Program in Biomedical Data Driven Discovery (BD3) at Northwestern University (National Library of Medicine Grant 5T32LM012203). The work of the second author is supported in part by NIH Grant R21LM012618.

References

[1] Niko Sünderhauf, Oliver Brock, Walter Scheirer, Raia Hadsell, Dieter Fox, Jürgen Leitner, Ben Upcroft, Pieter Abbeel, Wolfram Burgard, Michael Milford, et al. The limits and potentials of deep learning for robotics. *The International Journal of Robotics Research*, 37(4-5):405–420, 2018.

[2] A. Bendale and T. E. Boult. Towards open set deep networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1563–1572, 2016.

[3] Mehadi Hassen and Philip K Chan. Learning a neural-network-based representation for open set recognition. In *Proceedings of the 2020 SIAM International Conference on Data Mining*, pages 154–162. SIAM, 2020.

[4] Ryota Yoshihashi, Wen Shao, Rei Kawakami, Shaodi You, Makoto Iida, and Takeshi Naemura. Classification-reconstruction learning for open-set recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition*, June 2019.

[5] Chuanxing Geng, Sheng-jun Huang, and Songcan Chen. Recent advances in open set recognition: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[6] Chong Zhou and Randy C Paffenroth. Anomaly detection with robust deep autoencoders. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 665–674, 2017.

[7] Walter J. Scheirer, Anderson Rocha, Archana Sapkota, and Terrance E. Boult. Towards open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35, July 2013.

[8] Lalit P. Jain, Walter J. Scheirer, and Terrance E. Boult. Multi-class open set recognition using probability of inclusion. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 393–409, Cham, 2014. Springer International Publishing.

[9] H. Zhang and V. M. Patel. Sparse representation-based open set recognition. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 39(08):1690–1696, 2017.

[10] Nat Dilokthanakul, Pedro A. M. Mediano, Marta Garnelo, Matthew C. H. Lee, Hugh Salimbeni, Kai Arulkumaran, and Murray Shanahan. Deep unsupervised clustering with gaussian mixture variational autoencoders. *CoRR*, abs/1611.02648, 2016.

[11] Pedro R. Mendes Júnior, Roberto M. de Souza, Rafael de O. Werneck, Bernardo V. Stein, Daniel V. Pazinato, Waldir R. de Almeida, Otávio A. B. Penatti, Ricardo da S. Torres, and Anderson Rocha. Nearest neighbors distance ratio open-set classifier. *Machine Learning*, 106 (3):359–386, 2017.

[12] A. Nguyen, J. Yosinski, and J. Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *2015 IEEE Conference on Computer Vision and Pattern Recognition*, pages 427–436, 2015.