Berlin Mathematics Research Center MATH+

Approximate Convex Hulls With Bounded Complexity

Michael Joswig (TUB), Marek Kaluba (TUB), Klaus-Robert Müller (TUB), and Lukas Ruff (TUB)

For a given finite point set T in $\mathbb{R}^d$ we investigate methods to describe the convex hull conv(T) in geometricaly approximate sense. As an additional constraint we wish to bound the combinatorial complexity of the convex approximation, e.g., given in terms of the total number of faces. This problem is motivated by investigation of latent spaces of autoencoder neural networks.

The **convex hull problem**:

Given a set X of finitely many points in some Euclidean space $\mathbb{R}^d$ we wish to list:

1. linear inequalities (hyperplanes) for the finitely many facets of the convex polytope which is the convex hull of the input,
2. the set of vertices $V \subset X$ of the polytope.

We are interested in the situation where the dimension d is large. From known theoretical results it is clear, however, that this can only work in some approximate sense: even a small number of points may lead to exponentially many facets.

Theorem (Buchta-Müller-Tichy)

The expected number of facets of the polytope defined by a set of n *points in* $\mathbb{R}^d$ *is*

$$(2e)^{d+1} O(n \log^{d-1} n).$$

The precise complexity of the convex hull problem (measured by the sizes of both input and output, combined) is **unknown**.

Approximate convex hulls

We use the stochastic approach of [2] of random polytope. There, a random model of polytopes is defined as follows: we say that

$$P \sim P(d, n) \iff P = \mathrm{conv}(X)$$

where X is chosen uniformly over $S^{d-1} \subset \mathbb{R}^d$.

Definition (Dual bounding body)

Let $X \subset \mathbb{R}^d$ be a set of n points, ans $Y \subset S^{d-1}$, a set of m unit vectors. The **dual bounding body** is the polytope defined by the set of subspaces

$$D_Y(X) = \left\{ v \in \mathbb{R}^d : \langle v, y_i \rangle \leqslant \max_{i,j} \langle x_i, y_j \rangle \right\}_{y_j \in Y}.$$

Intuitively, dual bounding body can be understood as the result of following procedure.

Dual Bounding Body procedure

1. Start with the set $X \in \mathbb{R}^d$
2. For each direction $y_j \in Y$ slide (from infinity) a hyperplane perpendicular to y_j until it hits a point x_i. Let $k = \langle x_i, y_j \rangle$.
3. Add hyperplane $\langle v, y_j \rangle \leqslant k$ to the description of $D_Y(X)$.

It turns out that the dual bounding body $D_Y(X)$ is provably geometrically close to the convex hull conv(X), and it is a much better approximation from the practical point of view:

+ deciding whether $p \in \mathbb{R}^n$ is contained in $D_Y(X)$ is straightforward;
+ there is no need to compute all of its vertces as we can work with a fraction of them at a time.

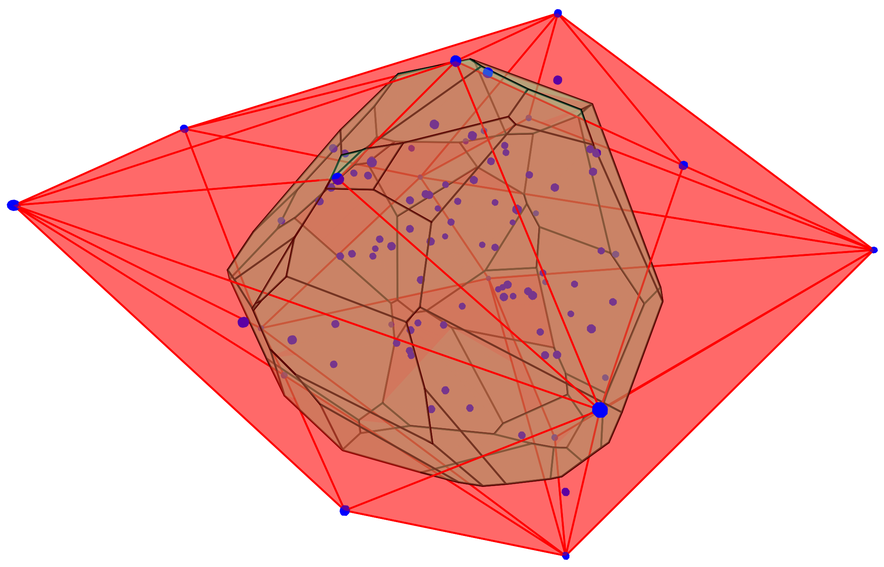

Figure 1. Random polytope data descriptor (in green, inside) defined by 50 randomly chosen directions in $\mathbb{R}^3$. Observe that the despite the presence of outliers, the classifier closely approximates the convex hull of the normal class (the cluster of blue points), as compared to the convex hull (in red).

Random Polytope Descriptors

Definition (Random Polytope Descriptor)

Random Polytope Descriptor $\mathrm{RPD}_{m,\ell}(X)$ defined by a dataset $X \subset \mathbb{R}^d$ is the dual bounding body

$$\mathrm{RPD}_{m,\ell}(X) = \left\{ v \in \mathbb{R}^d : \langle v, y_j \rangle \leqslant \ell\text{-max}_{i,j} \langle x_i, y_j \rangle \right\}_{y_j \in Y},$$

where $Y \sim P(d, m)$ is a set of m directions, chosen, uniformly at random from S^{d-1}.

In the presence of outliers in the data the parameter ℓ parametrizes the robustness of RPD, while m controls the computational complexity and its approximation characteristics.

Autoencoders and a measure of geometric (dis)entanglement

In machine learning it is a key challenge to solve the *classification problem*, i.e. to seek a mechanism to decide whether or not some input shares a given trait. A common solution is to train a neural network to discover the best features of the data for the classification purposes. This task can be viewed as a non-linear, low-dimensional embedding of the data. A particular case of such embeddings are presented by autoencoder networks. We analyzed vanilla autoencoder (AE) and the variational autoencoder (VAE) with Gaussian geometric prior:

$$\phi_{\mathrm{AE}} : \mathbb{R}^N \to \mathbb{R}^d,$$
$$\phi_{\mathrm{VAE}} : \mathbb{R}^N \to \mathbb{R}^{2d}.$$

Here $\mathbb{R}^d$ is known as **latent space** and the the image of the dataset under such map is referred to as **learned representation**.

+ Are the representations of classes learned from the dataset **entangled**?
+ Are the networks susceptible to **out-of-distribution attacks**?

While the entanglement does not have a universally accepted definition, a shared and very often implicit expectation is that “in good representations, the factors are related through simple, typically linear dependencies” [1].

Measuring entanglement We use the RPDs to analyze geometric properties of the learned representation $\phi(X) \subset \mathbb{R}^d$ for different d (i.e. different number of features).

1. Autoencoder network is trained for fixed embedding dimension d on the training part of the dataset.
2. Training data for RPD is prepared as the training data for a given class poisoned by 2% of points selected at random from all remaining classes;
3. RPD is trained on the prepared data and evaluated on the whole of the test data.

By evaluating the so called **scaling distance** of a sample to the data descriptor $\mathrm{RPD}_{m,\ell}$ we can label a sample as “normal” (if it comes from the training distribution) or “anomalous” (otherwise). Such classifier retains 95% accuracy of the original network, as assesed by the AUC score.

RPDs provide a **measure of the stability** of interpolations in the latent space.

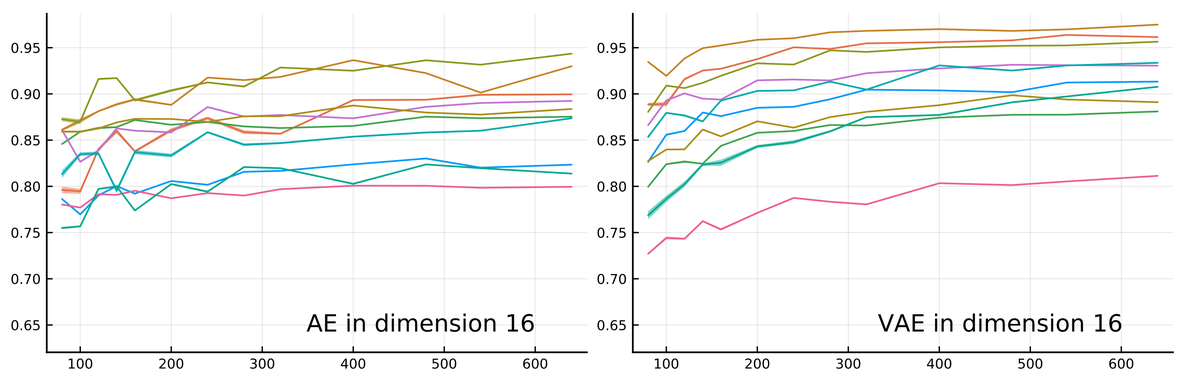

Figure 2. $\mathrm{RPD}_{m,2}$ AUC (vertical axis) scores measured on FMNIST dataset per class. The number m of hyperplanes on horizontal axis. Note: RPDs retain 95% of accuracy of the original network.

Out-of-distribution attacks We can use the computed RPDs to evaluate networks resilence to out-of-distribution attacks. These attacks test how susceptible the network is to assign with high confidence learned labels to samples which do not come from the training distribution.

RPDs can be used to **quantify out-of-distribution susceptibility**.

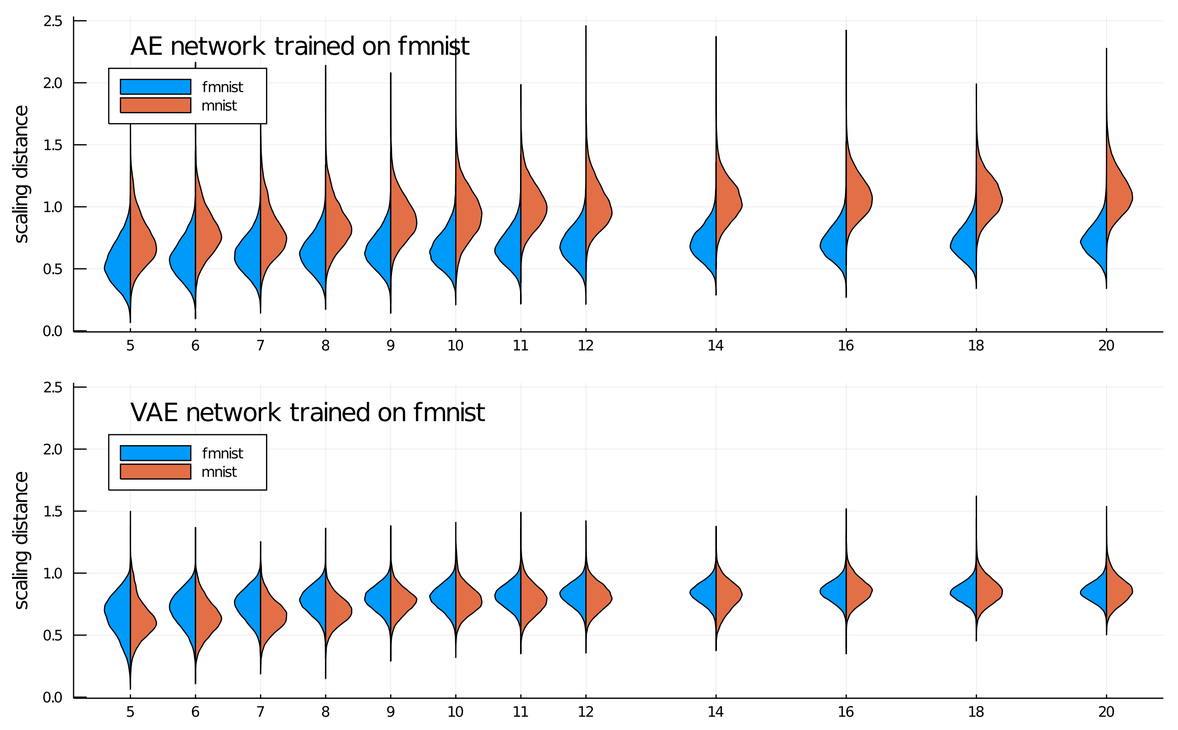

Figure 3. Results of the out-of-distribution detection experiment. AE and VAE networks were trained on FMNIST data and used to embed MNIST test sets. The plot depicts the distribution of minimal scaling distance to one of the ten FMNIST random polytope descriptors in various dimensions up to 20 for fixed $(m, \ell) = (320, 1)$.

Want to learn more?

M. Kaluba, L. Ruff and M. Joswig *Geometric Disentanglement by Random Convex Polytopes arXiv:2009.13987*

References

[1] Y. Bengio, A. Courville, and P. Vincent.
Representation learning: A review and new perspectives.
IEEE Transactions on Pattern Analysis and Machine Intelligence, 35(8):1798–1828, 2013.

[2] K.-H. Borgwardt.
The simplex method, volume 1 of *Algorithms and Combinatorics*.
Springer-Verlag, Berlin, 1987.
A probabilistic analysis.
doi:10.1007/978-3-642-61578-8.