COMPACT BINARY REPRESENTATIONS AND GENERATIVE MODELS

Jakub Zadrożny
University of Wrocław
Advisor: Rafał Nowak
Contact: jakub.r.zadrozny@gmail.com

While you were reading this sentence, several yottabytes of new data has been registered all over the world. It is extremely hard, to build computer systems that scale to the amount of information we gather nowadays. That's why it's crucial to find good, efficient representation of our data. Here, we give an example of a highly compacted representation for a complex dataset of 3D point clouds.

Learning representations

- We are given a dataset $\mathscr{X}$, containing some d-dimensional samples $\hat{X}$.
- We seek a compact representation of this data that generalizes to unseen examples.
- We would also like to generate artificial samples.

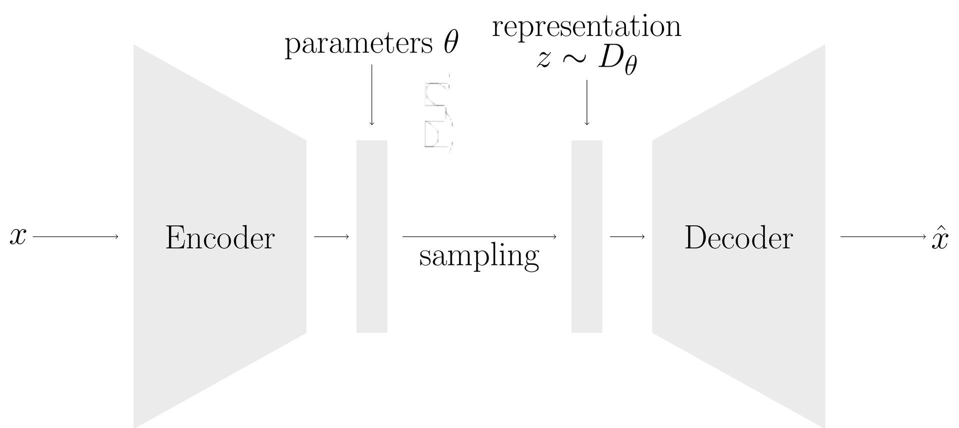

Fig. 1: Schematic representation of a VAE.

- Suppose each sample $\hat{X}$ is an observation of a random variable X whose distribution depends on a latent variable Z.
- We assume Z has a parametrized distribution with known true parameters.
- Also $X \mid Z$ is normally distributed with mean being a complicated, parametrized transformation of Z with unknown parameters.
- We use a variational autoencoder (VAE) to capture dependencies between X and Z.
- This allows us to construct a representation of data in a regularized space, which allows us to sample new data points.

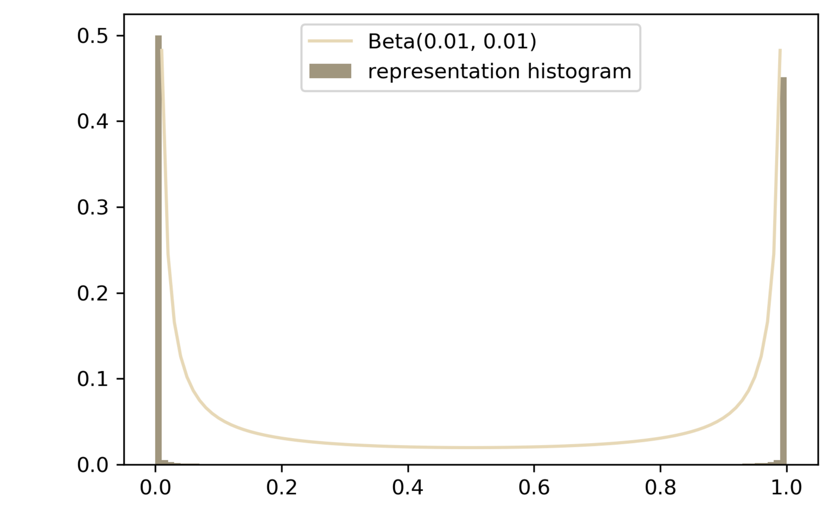

Fig. 2: Density of the Beta(0.01, 0.01) distribution and the histogram of representation values produced by the Beta-regularized model.

Methodology

- Our goal is to find as compact representation as possible.
- We train two benchmark models: a baseline autoencoder (discriminative) and a standard VAE (generative), i.e. with $Z \sim \mathcal{N}(0, I_k)$.
- This gives us two continuous representations.
- We continue by proposing a method to obtain a representation of the same dimension, but with binary components.
- Namely we assume that Z has k independent components, each of which is distributed with Beta(0.01, 0.01).
- The Beta distribution is used here because (1) its support is the (0, 1) interval, (2) it's highly and equally condensed near 0 and 1.

- We train the model like a standard VAE, in a continuous fashion.
- If we manage to get a well regularized model, the sampled representations should contain values very close to 0 and 1.
- We obtain the binary representation by thresholding the continuous one at 0.5.

Working with 3D point clouds

- We display the potential of VAE's to find compact representations by applying them to datasets of 3D point clouds.
- We learn representations of objects belonging to a single class from the ShapeNet and ModelNet40 datasets.
- We base the reconstruction loss on the *Chamfer distance (CD)* to obtain point permutation invariance.

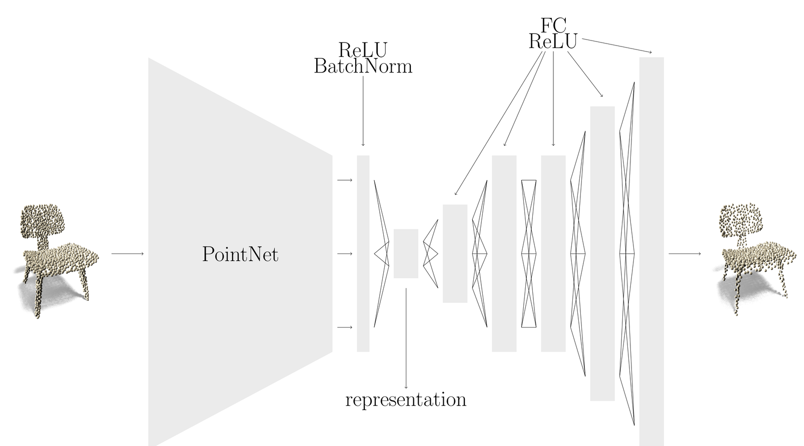

Fig. 3: Our PointNet-based model architecutre.

- To facilitate working with spatial data we make use of the PointNet architecture.
- We use a simple MLP decoder to encourage a more meaningful representation.

In partnership and following the work of M. Zamorski, M. Zięba, R. Nowak, W. Stokowiec, and T. Trzciński: "Adversarial autoencoders for generating 3d point clouds," 2018.

Results

- The binarization (thresholding) of Beta-regularized representation has a minor effect on reconstruction quality.
- Our continuous and discrete models reconstruct train and test data well, perform natural interpolations, allow for predictable arithmetics on objects and artificial data generation.
- The binary representation allows for reconstructions of quality visually comparable to continuous, yet it occupies 32x less space (with single-precision floats).
- An explicit data format requires almost 200kb of memory, while our binary encoding fits in just 128 bits, the same space as just 4 floats. This results in a massive **1500x compression rate** without significant quality loss.

	train	test
AE	0.853	1.247
VAE-N	0.851	1.287
VAE-B	1.024	1.464
VAE-bin	1.027	1.464

Fig. 4: Average *Chamfer distance* between original objects and reconstructions generated by different models.

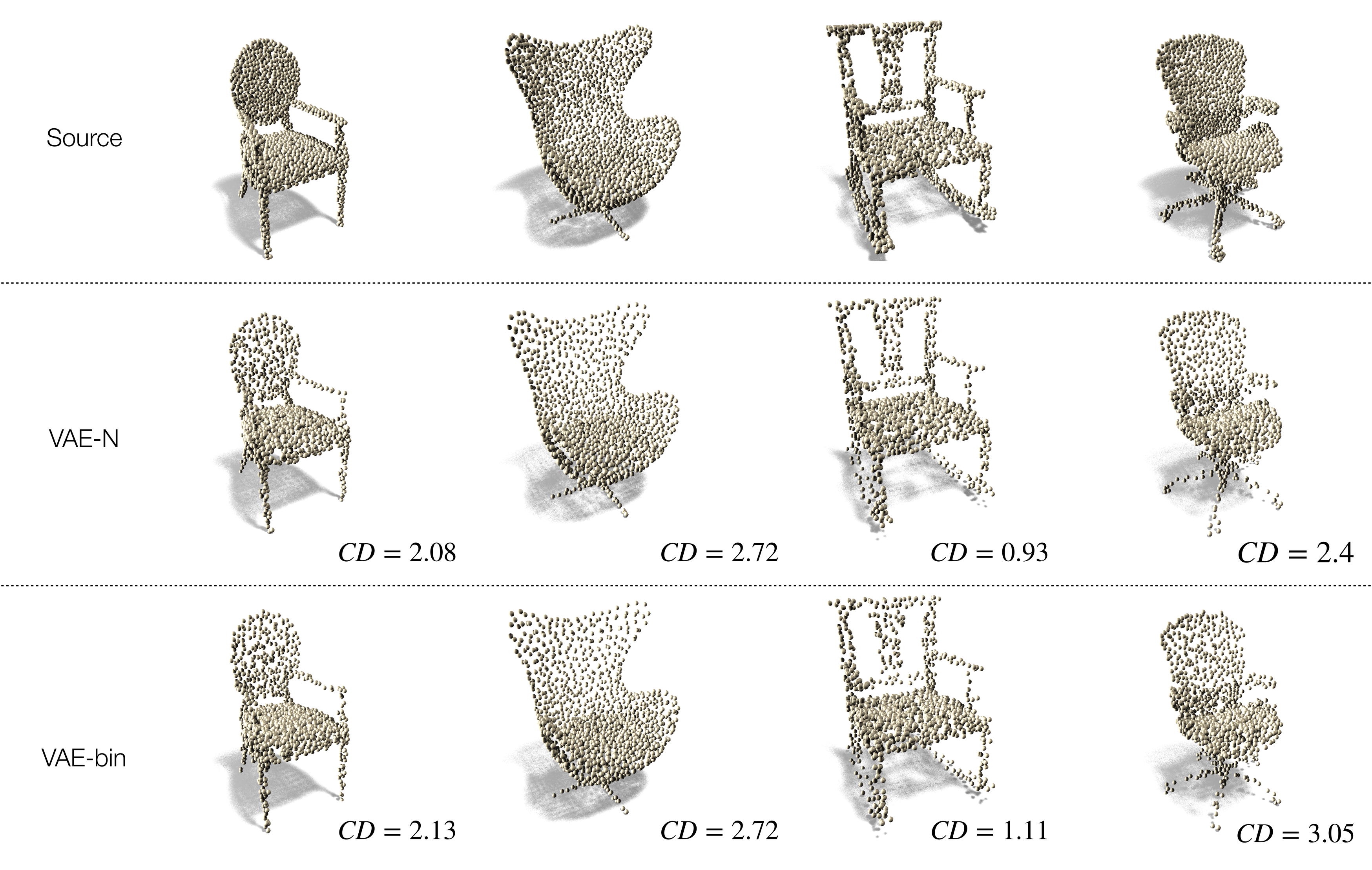

Fig. 5: Reconstructions of selected objects from the test dataset. The quantity below images is the *Chamfer distance* from the original. The average CD is 1.29 for VAE-N and 1.46 for VAE-bin.

Fig. 6: Artificial objects generated from binary representations sampled from a multidimensional Bernoulli distribution with $p = 0.5$.

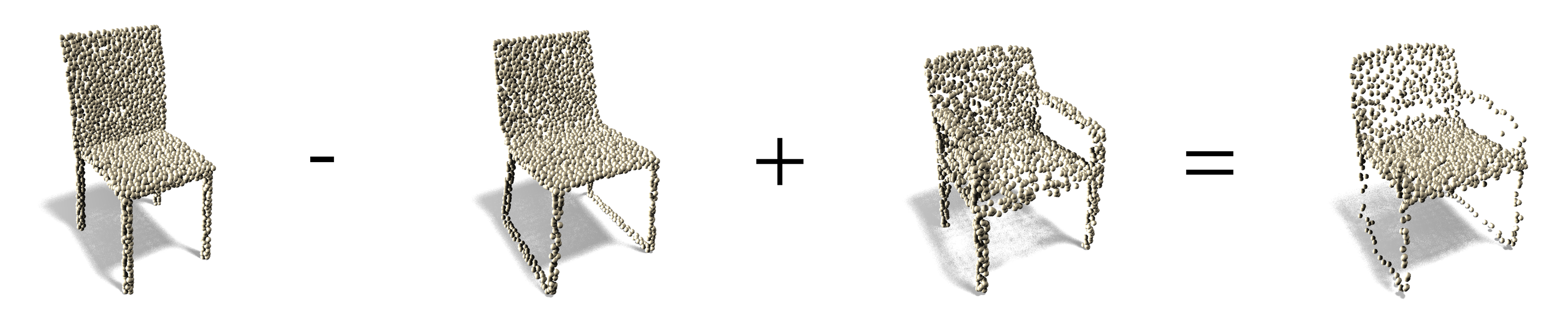

Fig. 7: Adding armrests to a chair by performing arithmetic on binary representations.

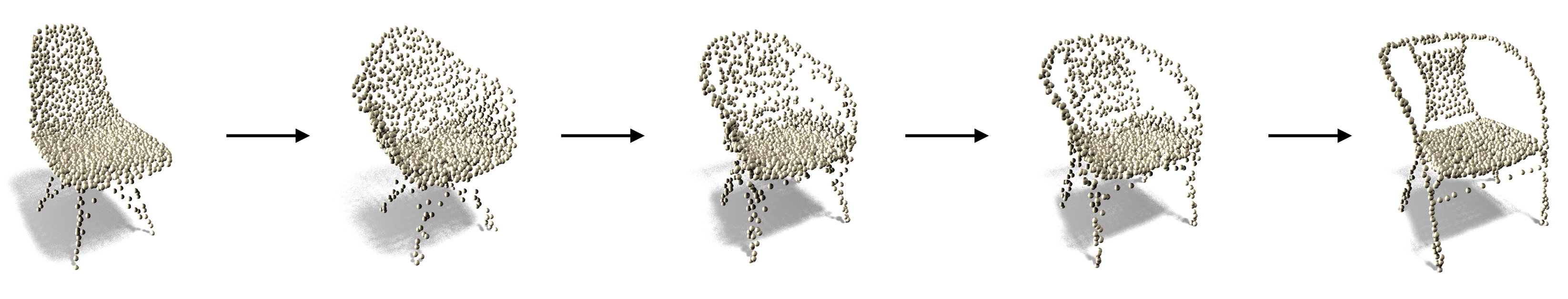

Fig. 8: Performing smooth interpolation on binary representations. The intermediate representations are obtained by flipping an increasing number of bits in the encoding, according to a random permutation.

Side project: clustering with VAE

- We suspect our data can be naturally divided into several subcategories.
- Such data could be well fitted with a mixture distribution.
- We introduce a new, discrete variable Y, which determines the distribution of Z. Then Z has a mixture distribution.
- The distribution of X is still determined by a transformation of Z.
- The encoder now produces mixture weights and component parameters.
- Each sample is assigned to a cluster corresponding to the heaviest component.

Fig. 9: Randomly selected objects from 3 chosen clusters (out of 10).

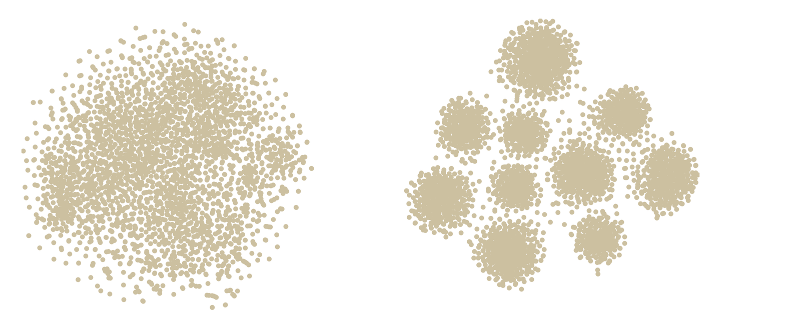

Fig. 10: A t-SNE visualiztion of the representation space of the standard VAE (left) and the mixture VAE (right).

Following the work of D. P. Kingma, S. Mohamed, D. J. Rezende, and M. Welling, "Semi-supervised learning with deep generative models", and R. Shu, "Gaussian mixture vae: Lessons in variational inference, generative models and deep nets."