CaGeM: A Cluster Aware Deep Generative Model

Lars Maaløe, Marco Fraccaro and Ole Winther

Cognitive Systems, Department of Applied Mathematics and Computer Science, Technical University of Denmark

Introduction

We introduce the *Cluster-aware Generative Model (CaGeM)*, an extension of a VAE, that improves the generative performances, by being able to model the natural clustering in the higher feature representations through a discrete variable. The model can be trained fully unsupervised, but its performances can be further improved using labelled class information that helps in constructing well defined clusters.

Cluster-aware Generative Models

In VAEs, the addition of more stochastic layers is often accompanied with a built-in pruning effect so that the higher layers become inactive and therefore not exploited by the model (Sønderby et al., 2016). CaGeM provide the possibility of learning a representation in the higher stochastic layers that can model the natural clustering of the data. This results in a model that is able to disentangle some of the factors of variation in the data and that extracts a hierarchy of features that are beneficial for the generation phase.

We favour the flow of higher-level global information by extending the generative model of a VAE with a discrete variable representing the choice of different clusters in the data (**Figure 1**):

$$p_\theta(x, z_1, z_2) = \sum_y p_\theta(x, y, z_1, z_2)$$
$$= \sum_y p_\theta(x|y, z_1) p_\theta(z_1|y, z_2) p_\theta(y|z_2) p(z_2) \ .$$

The corresponding variational approximation looks as follows:

$$q_\phi(y, z_1, z_2|x) = q_\phi(z_2|x, y, z_1) q_\phi(y|z_1, x) q_\phi(z_1|x) \ .$$

The discrete variable depends solely on the highest latent stochastic layer that needs therefore to stay active for the model to be able to represent clusters in the data. The ELBO in CaGeM is:

$$\mathcal{F}(\theta, \phi) = \mathbb{E}_{q_\phi(z_1|x)} \left[\sum_y q_\phi(y|z_1, x) \cdot \right.$$
$$\left. \cdot \mathbb{E}_{q_\phi(z_2|x,y,z_1)} \left[\log \frac{p_\theta(x, y, z_1, z_2)}{q_\phi(y, z_1, z_2|x)} \right] \right] \ .$$

In some applications we may have class label information for some of the data points in the training set. In the following we will show that CaGeM provides a natural way to exploit additional labelled data to improve the performance of the generative model. This differs from traditional semi-supervised learning (Kingma et al. 2014, Maaløe et al. 2016), since the labeled data support the generative task. CaGeM contains two classifiers:

1. In the inference network we can compute the class probabilities given the data by integrating out the stochastic variables from:

$$q_\phi(y|x) = \int q_\phi(y, z_1|x) dz_1$$
$$= \int q_\phi(y|z_1, x) q_\phi(z_1|x) dz_1$$

2. In the generative model we have another set of class probabilities given the posterior distribution of the top latent stochastic variable:

$$p_\theta(y|x) \approx \int p_\theta(y|z_2) q_\phi(z_2|x) dz_2$$
$$= \int p_\theta(y|z_2) \left(\int \sum_{\widetilde{y}} q_\phi(z_2|x, \widetilde{y}, z_1) \cdot \right.$$
$$\left. \cdot \ q_\phi(\widetilde{y}|z_1, x) q_\phi(z_1|x) dz_1 \right) dz_2 \ .$$

	ELBO
VAE, L=2, IW=50 (Burda et al., 2015)	-106.30
IWAE, L=1, IW=50 (Burda et al., 2015)	-103.38
LVAE, L=5, IW=10 (Sønderby et al., 2016)	-102.11
RBM (Burda et al., 2015)	-100.46
DBN (Burda et al., 2015)	-100.45
DVAE (Rolfe, 2017)	-97.43
CaGeM-500, L=2, IW=1	-100.86

Table 2: Test log-likelihood on permutation invariant OMNIGLOT for L number of stochastic layers and IW importance weighted samples. Directly comparable models are VAE, IWAE and LVAE.

Results

We evaluate the generative performance of CaGeM on the MNIST dataset. We can see from **Table 1** that the more labelled samples we use, the better the generative performance will be. Even though the results are not directly comparable, since CaGeM exploits a small fraction supervised information, we find that by using only 100 labeled samples, CaGeM achieves state-of-the-art performance on permutation invariant MNIST.

It is also interesting to see that the fully unsupervised CaGeM-0 performs substantially better than directly comparable architectures, by embedding information on clusters in the higher latent stochastic layers.

When testing the performance on OMNIGLOT we use the alphabet class as cluster information. From **Table 2** we see an improvement over other comparable VAE architectures (VAE, IWAE and LVAE), however, the performance is far from the once reported from the auto-regressive models. This indicates that the alphabet information is not as strong.

Conclusion

In this work we have shown how to perform semi-supervised generation with CaGeM. We showed that CaGeM improves the generative log-likelihood performance over similar deep generative approaches by creating clusters for the data in its higher latent representation using unlabelled information. CaGeM also provides a natural way to refine the clusters using additional labelled information to further improve its modelling power.

	ELBO
ADGM-100, L=2, IW=1 (Maaløe et al., 2016)	-86.06
IWAE, L=1, IW=1 (Burda et al., 2015)	-85.33
AVAE, L=2, IW=1 (Maaløe et al., 2016)	-82.97
IWAE, L=1, IW=50 (Burda et al., 2015)	-82.90
LVAE, L=5, IW=1 (Sønderby et al., 2016)	-82.12
LVAE, L=5, IW=10 (Sønderby et al., 2016)	-81.74
VAE+VGP, L=2 (Tran et al., 2015)	-81.32
DVAE (Rolfe, 2017)	-80.04
CaGeM-0, L=2, IW=1, K=20	-81.92
CaGeM-0, L=2, IW=1, K=5	-81.86
CaGeM-0, L=2, IW=1, K=10	-81.60
CaGeM-20, L=2, IW=1	-81.47
CaGeM-50, L=2, IW=1	-80.49
CaGeM-100, L=2, IW=1	-79.38

Table 1: Test log-likelihood on permutation invariant MNIST for L number of stochastic layers, IW importance weighted samples and K number of clusters. Directly comparable models are IWAE and LVAE.

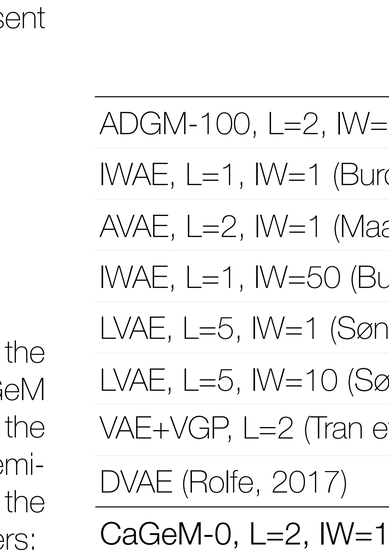

Figure 1: Generative model (left) and inference model (right) of CaGeM with two stochastic layers.

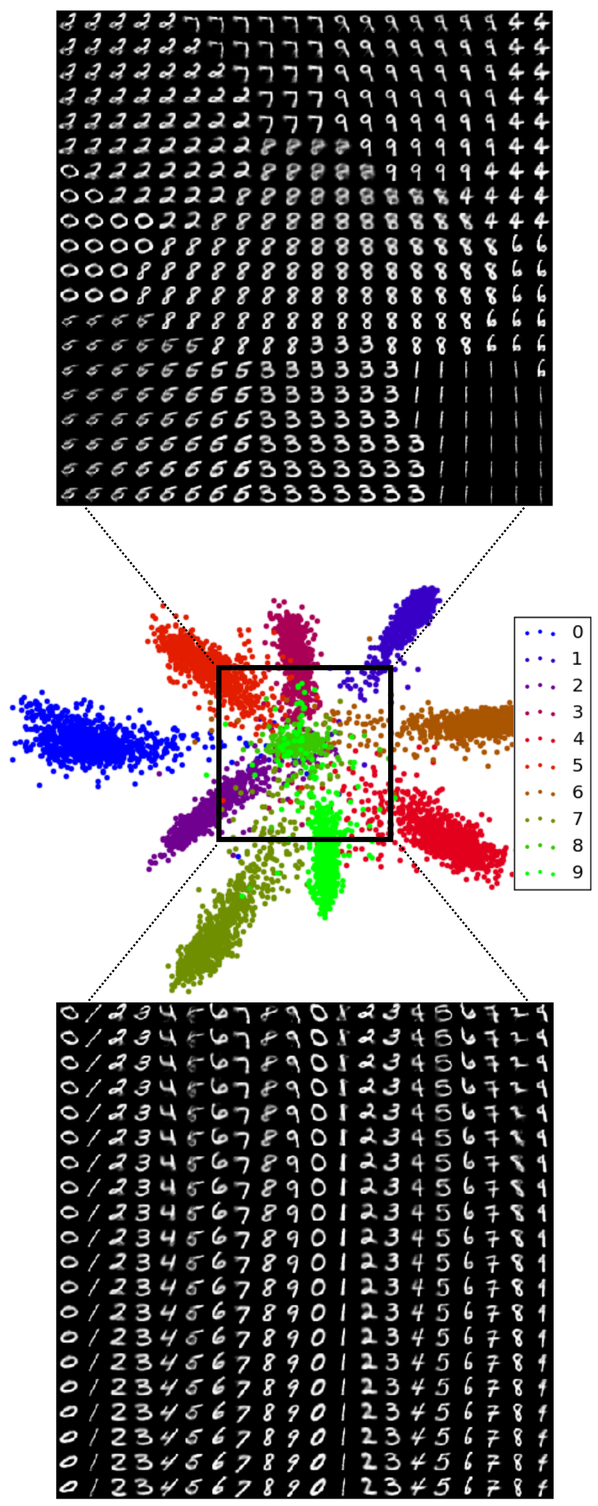

Figure 2: MNIST visualisations from CaGeM-100 with a 2-dimensional top stochastic layer. The middle plot shows the latent space, from which we generate random samples (top) and class conditional random samples (bottom) with a mesh grid (black bounding box). The relative placement of the samples in the scatter plot corresponds to a digit in the mesh grid.

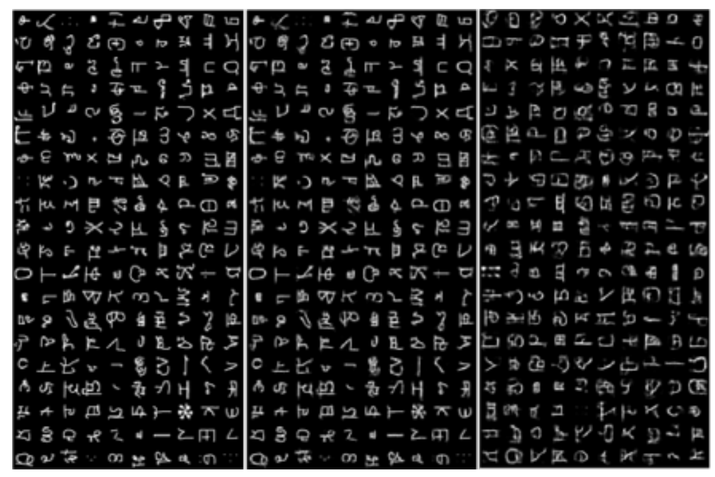

Figure 3: Generations for the OMNIGLOT data set from CaGeM-500. (left) The input images, (middle) the reconstructions, and (right) random samples from the top stochastic layer.

References

Burda, Y., Grosse, R., Salakhutdinov, R.. (2015). Accurate and conservative estimates of mrf log-likelihood using reverse annealing. AISTATS.

Burda, Y., Grosse, R., Salakhutdinov, R.. (2015), Importance Weighted Autoencoders. ICLR.

Kingma, D.P., Rezende, D.J., Mohamed, S., Welling, M.. (2014). Semi-Supervised Learning with Deep Generative Models. ICML.

Sønderby, C.K., Raiko, T., Maaløe, L., Sønderby, S.K., Winther, O.. (2016). Ladder Variational Autoencoders, NIPS.

Tran, D., Raganath, R., Blei, D.M.. (2016). ICLR.

Maaløe, L., Sønderby, C.K., Sønderby, S.K., Winther, O.. (2016). Auxiliary deep generative models. ICML.

Rolfe, J.T.. Discrete variational autoencoders. (2017). ICLR.