Project report
Improving Explorability in Variational Inference with Annealed Variational Objectives.

Mikhail Kurenkov, Timur Chikichev, Aleksei Pronkin

* https://github.com/alexey-pronkin/annealed

Abstract

We represent the results of paper [1]. The main work of research is a special procedure, applied into the training of hierarchical variational methods. The method called Annealed Variational Objectives (AVO) has to solve the problem of limited posterior distribution density. The method facilitates learning by integrating energy tempering into the optimization objective.

The paper presents contains experiments on the proposed method. These experiments represent the drawbacks of biasing the true posterior to be unimodal, and show how proposed method solve this problem. We repeat experiments from [1] and compare performance of AVO with normalizing flows (NF) and variational auto-encoders (VAE). Additionally we make experiments with deterministic warm up (analogously to AVO), which when applied to NF and VAE may benefits in better space exploration.

1 Introduction

With variational inference, we have some variational distribution we may use to generate samples. The resulting variance can be lowered by two ways, increasing number of samples and increasing approximation accuracy. If variational inference has bad uncertainty approximation (Turner and Sahani (2011)), we will receive bias in statistics in terms of overconfidence and inaccuracy. The statistics we check in models are marginal likelihood of data and the predictive posterior. The same in the amortized VI setups, the representation of data will require better exploration from approximation model during training.

To express the bias induced by a non-rich and non-expressive variational family, the objective can be written as KL-divergence between proposal and target distributions.

Variational inference objective:

$$F(q) = E_q[\log q(z) - \log f(z)] = D_{KL}(q\|f)$$

Due to KL-divergence, the resulting approximation will have low probability mass in regions with low density. The variational approximation may escape points with sufficient statistics in true target, but with small local density. For multi-modal target distributions, not all target space will be covered and the model will loose some sufficient statistics.

Annealing techniques may increase exploration of the target density.

Alpha-annealing (expressiveness):

$$E_q[\log q(z) - \alpha \log f(z)]$$

where $\alpha \sim \frac{1}{T}$, and T is temperature which defines the speed of approximate model changes, e.g. learning rate. When α goes from zero to 1, we obtain the usual objective, but with full energy landscape covered.

The problem, with low penalty on the energy term, the whole procedure is time consuming. This is because multiple inferences are required on each maximization step (deep neural networks, hierarchical models, etc.).

Beta-annealing (optimization):

Deterministic warm-up (Raiko et al., 2007) is applied to improve training of a generative model.

$$p(x, z) = p(x|z)p(z)$$

The joint likelihood is equal to the un-normalized true posterior $f(z) = p(z|x)$.

The annealed objective is (negative ELBO):

$$E_q[\beta(\log q(z) - \log p(z)) - \log p(x|z)]$$

In annealed objective, the β is annealed from 0 to 1. This disables the regularisation of posterior to be like a prior. First training the negative log-likelihood, we train the decoder independently. With this the model is trained to fit the data, so we have more deterministic auto-encoder. With this approach we additionally lose in latent space exploration.

The model: Latent variable model with a joint probability $p_\theta(x, z) = p(x|z)p(z)$.

x and z are observed and latent variables, θ - model parameters to be learned.

Training procedure, given expected complete data log likelihood over q:

$$\max_\theta E_{q(z)}[\log p_\theta(x, z)]$$

Conditional $q(z|x)$

1. tractable: Expectation-Maximization(EM) algorithm.
2. non-tractable: approximate the true posterior (MCMC, VI)

Variational distribution subfamilies with expressive parametric form

1. Hierarchical Variational Inference(HVI)
2. Auxiliary variable methods
3. Normalizing flows

In HVI, we use a latent variable model $q(z_T) = \int q(z_T, z_{t<T})dz_{t<T}$, where $t < T$ denoting latent variables.

We use reverse network $r(z_{t<T})$ to lower bound intractable $q(z_T)$.

$$\begin{aligned}
&- E_{q(z_T)}[\log q(z_T)] \geq -E_{q(z_T)}[\log q(z_T) + D_{KL}(q(z_{t<T}|z_T)||r(z_{t<T}|z_T))] = \\
&= -E_{q(z_T, z_{t<T})}[\log q(z_T|z_{t<T})q(z_{t<T}) - \log r(z_{t<T}|z_T)]
\end{aligned}$$

The variational lower bound is:

$$L(x) = E_{q(z_T, z_{t<T})}[\log \frac{p(x, z_T)r(z_{t<T}|z_T)}{q(z_T|z_{t<T})q(z_{t<T})}]$$

As the $q(z)$ is one from chosen distribution subfamilies, we have the capability to represent any posterior distribution. If possible to invert $q(z_T|z_{t<T})$, we choose r to be its invert transformation. This is the so called inverse auto-regressive flow (IAF). The KL term is zero, the variance is lower, the entropy is computed via change of the volume formula.

$$q(z_T) = q(z_{T-1})|\frac{\partial z_T}{\partial z_{T-1}}|^{-1} \tag{1}$$

Loss function tempering: annealed importance sampling

Annealed importance sampling(AIS) is an MCMC method with same concept as alpha annealing, it let the variational distribution be more exploratory early on during training.

We have an extended state space with $z_0, .., .z_T$ latent variables. z_0 is sampled from simple distribution (Gaussian normal prior distribution $p(z)$). Particles are sequentially sampled from the transition operators $q_t(z_t\|z_{t-1})$.

To define transition operators, we design a set of intermediate target densities as $\tilde{f}_t = f_T^{\tilde{\alpha}_t} f_T^{1\tilde{-}\alpha_t}$. This is the set of targets defined as a mixture of initial (normal) and target (complex multi-modal) distributions.

For intermediate targets to be invariant, the transitions are constructed as Markov chain with the following weights:

$$w_j = \frac{\tilde{f}_1(z_1)\tilde{f}_2(z_2)}{\tilde{f}_0(z_1)\tilde{f}_1(z_2)} ... \frac{\tilde{f}_T(z_T)}{\tilde{f_{T-1}}(z_T)}$$

For the estimate to be accurate we need a long sequence of transitions, computationally difficult.

Annealed Variational Objectives(AVO)

Similar to AIS and alpha-annealing, authors of [1] propose to integrate energy tempering into the optimization objective of the variational distribution.

As in AIS, we consider an extended state space with same transitional targets. The marginal $q_T(z_T)$ is an approximate posterior. To define incremental steps, we construct T forward transition operators and T backward operators. We construct intermediate targets as an interpolation between the true (unnormalized) posterior and the initial (normal) distribution: $\tilde{f}_t = f_T^{\tilde{\alpha}_t} f_T^{1\tilde{-}\alpha_t}$, where $\alpha \in [0, 1]$.

Different from AIS, we learn the parametric transition operators which are assigned to each transition pair. We have a sequence of one layer networks as a result.

Annealed Variational Objectives(AVO):

$$\max_{q_t(z_t|z_{t-1})r_t(z_{t-1}|z_t)} E_{q_t(z_t|z_{t-1})q_{t-1}(z_{t-1})}[\log \frac{\tilde{f}_t(z_T)r_t(z_{t-1}|z_t)}{q_t(z_t|z_{t-1})q_{t-1}(z_{t-1})}]$$

In implementation we use detach method of the Pytorch for optimization of the AVO loss respect to parameters of $q(z_t|z_{t-1})$ and $r(z_t|z_{t-1})$ with fixed z_{t-1} and $q_{t-1}(z_{t-1})$.

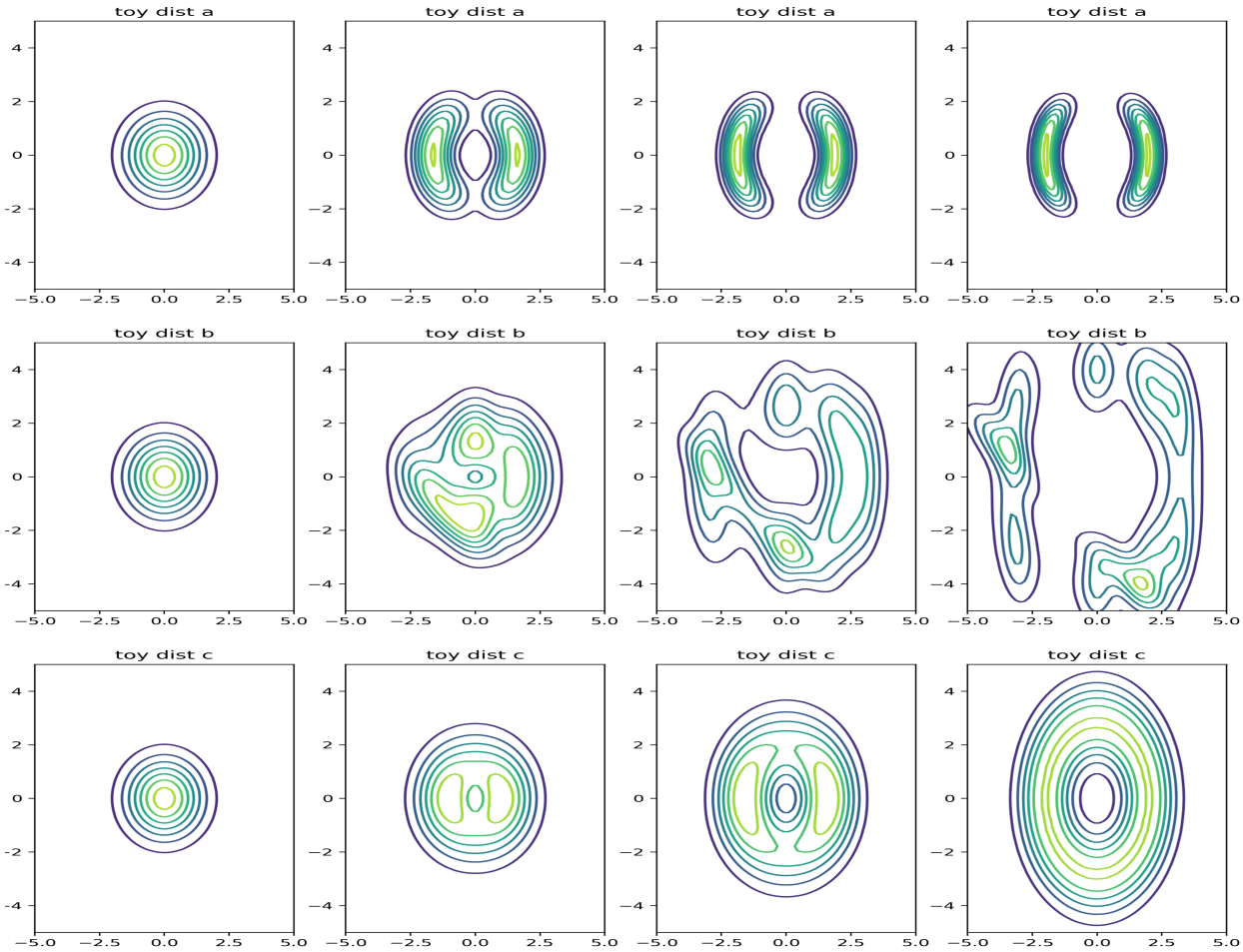
Stochastic refinement transition operator:

$$\begin{aligned}
&q_t(z_t|z_{t-1}) = N(z_t|\mu_t^q(z_{t-1}), \sigma_t^q(z_{t-1}));\\
&r_t(z_{t-1}|z_t) = N(z_{t-1}|\mu_t^r(z_t), \sigma_t^r(z_t));\\
&\mu(z) = g(z) \cdot m(z) + (1 - g(z)) \cdot z; \sigma(z) = act_\sigma(W_\sigma \cdot h(z) + b_\sigma)\\
&m(z) = W_m \cdot h(z) + b_m; g(z) = act_g(W_g \cdot h(z) + b_g); h(z) = act_h(W_h \cdot z + b_h)
\end{aligned}$$

In implementation we use reparametrization technique for performing forward and reverse transitions.

On figure 1, we present a transitional targets for a first three toy energy functions. The list of all functions used in paper will be presented in section 2.1.

Figure 1: Annealed Variational Objectives. Transitional targets presented on a first three toy energy functions.



2 Experiments

2.1 Toy energy fitting

Specification of the toy energy functions:

$$(a) -\frac{((\|z\|_2 - 2)/0.4)^2}{2} + \log(\exp(-\frac{((z_1 - 2)/0.6)^2}{2}) + \exp(-\frac{((z_1 + 2)/0.6)^2}{2}))$$

$$(b) -\frac{((\|0.5z\|_2 - 2)/0.5)^2}{2} + \log(\exp(-\frac{((z_1 - 2)/0.6)^2}{2}) + \exp(-\frac{(2\sin z_1)^2}{2}) + \exp(-\frac{((z_1 z_2 + 2.5)/0.6)^2}{2}))$$

$$(c) -(2 - \sqrt{z_1^2 + \frac{z_2^2}{2}})^2$$

$$(d) \log(0.1N([-2, 0]^T, 0.2I) + 0.3N([2, 0]^T, 0.2I) + 0.4N([0, 2]^T, 0.2I) + 0.2N([0, -2]^T, 0.2I))$$

$$(e) -\frac{((z_2 - w_1)/0.4)^2}{2} + 0.1(z_1^2)$$

$$(f) \log(\exp(-\frac{((z_2 - w_1)/0.35)^2}{2}) + \exp(-\frac{((z_2 - w_1 + w_2)/0.35)^2}{2})) - 0.05z_1^2$$

$$w_1 = \sin(\frac{\pi}{2} z_1), w_1 = 3\exp(\frac{(z_1 - 2)^2}{2})$$

Figure 2: HVI - AVO, toy energy fitting: target is four-mode mixture of Gaussian

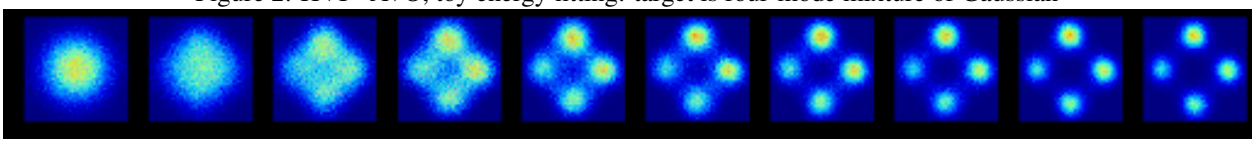
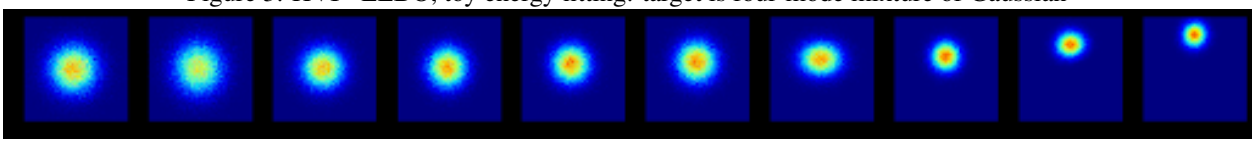


Figure 3: HVI - ELBO, toy energy fitting: target is four-mode mixture of Gaussian



2.2 Compare HVI - ELBO and HVI - AVO

We use HVI with 4 layers (hidden dimension 32, activation - ReLU) 2k learning parameters with Adam optimizer (learning rate = 1e-3, beta1=0.999, beta2=0.999) and batch size 128. We use the Inverse Auto-regressive Flow (IAF) from the Pyro with [32, 32] hidden dimensions in IAF (activation ReLU, depth=4, 3k learning parameters) the batch size=128 and the optimizer - Adam (lr=1e-3, beta1=0.999, beta2=0.999)

On figures 2 and 3 we see results of the experiment run for same input data and target, only the model is different. The multi-modal target is mixture of four Gaussian defined as toy D model. We see that HVI-ELBO results in only one mode covered, and HVI-AVO explores the energy space qualitatively better.

On figure 4 we show methods performance on a different toy models. The methods are:

1. HVI ELBO - Hierarchical Variational Inference evidence lower bound optimization [2]
2. IAF ELBO - Inverse auto-regressive flow lower bound optimization
3. HVI AVO - Hierarchical Variational Inference with Annealed Variational Objectives
4. IAF AVO - Inverse auto-regressive flow with Annealed Variational Objectives

Notes:

- In the first row(HVI ELBO), the model space is not covered well.
- In the last row(IAF AVO), the posterior distribution is unimodal, it has a large probability density in regions with low target energy
- The third column, we see that HVI methods show better performance and the result distribution is not unimodal.

2.3 Amortized inference on MNIST dataset

Method	Our result	Original result
VAE	94.16	87.50
VAE-HVI	95.90	87.62
VAE-HVI-AVO	108.67	86.06

Table 1: Negative log likelihood for VAE, VAE-HVI and VAE-HVI-AVO

Also during this research project we conducted experiments with amortized inference. For these experiments we took MNIST dataset. For amortized inference we tested three setups: VAE, VAE-HVI, VAE-HVI-AVO. In the first setup Gaussian proposal distribution $q(z|\mu(x), \sigma(x))$ is used. In the second setup we applied HVI with 5 stochastic transition operators as variational proposal distribution. In this setup we optimizer ELBO loss. In the last setup we also use HVI, but for optimization we utilize annealed objective. In VAE-HVI-AVO setup we optimize AVO loss with probability α which is equal to 0.2 and in another case we optimize only ELBO loss. This method is taken from [1].

Figure 4: Toy energy fitting with inverse auto-regressive flows. Columns left to right: HVI ELBO, IAF ELBO, HVI AVO, IAF AVO. Rows top to bottom: toy models A to F.

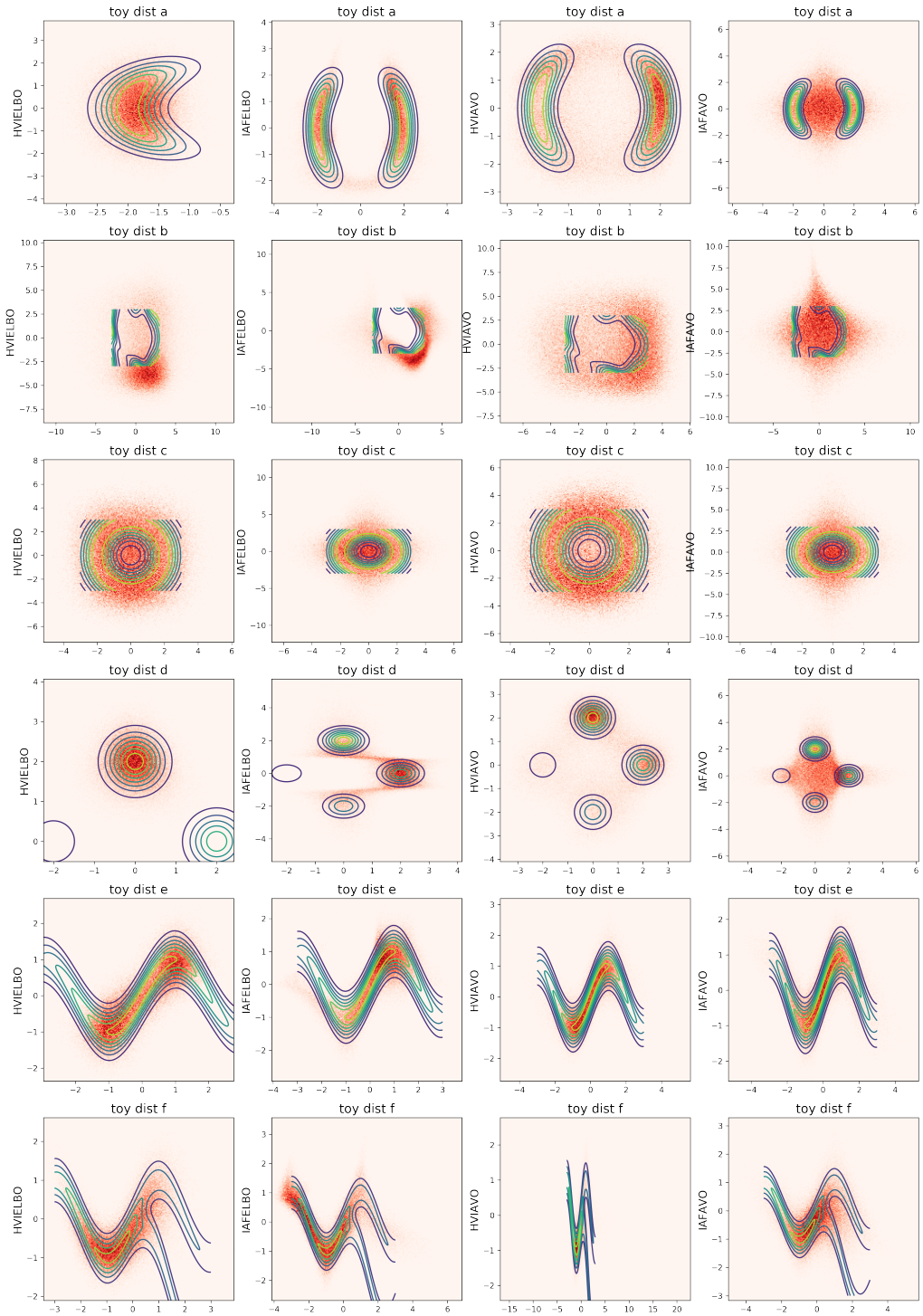


Figure 5: KL part of ELBO during training. Blue line is VAE, green VAE-HVI, orange line VAE-HVI-AVO

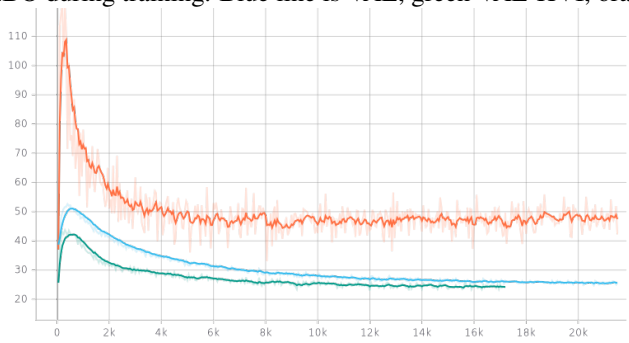


Figure 6: NLL part of ELBO during training. Blue line is VAE, green VAE-HVI, orange line VAE-HVI-AVO

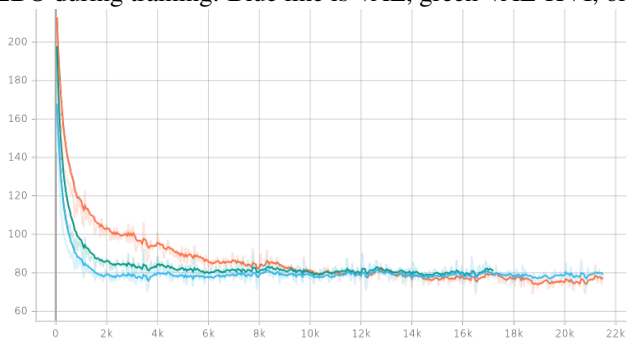


Figure 7: Reconstruction of VAE. Left is ground truth, right is reconstructed

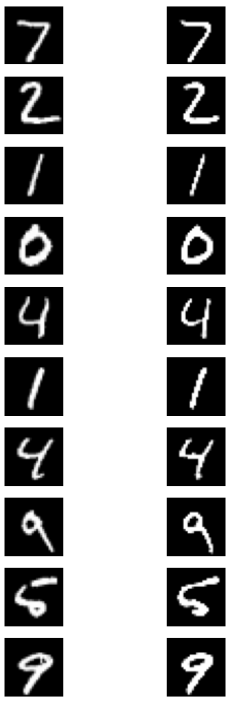
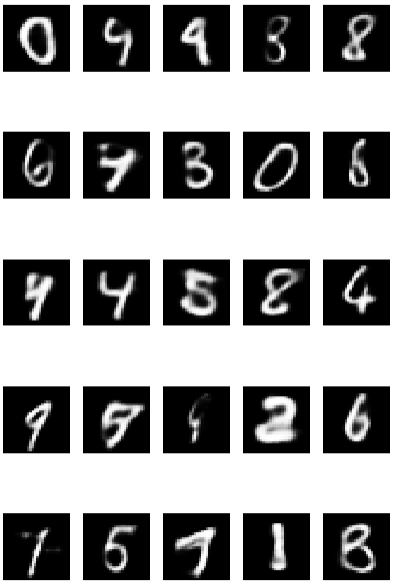


Figure 8: Generation of images of VAE



For decoder and encoder of VAE we choose parameters from [3] article. Decoder and encoder are two layers neural networks. The dimension of each hidden layer is 300. Latent space size equals 40. Also we apply batch normalization and leaky ReLu activation function. Batch size for MNIST dataset is 256. We also take beta annealing scheduler (warm-up) from [4]. We increases beta from 0 to 1 during 200 starting epochs. Optimizer is Adam with standard parameters (learning rate is $1e-3$). For each setup we optimize during 400 epochs.

The result of these experiments is shown in table 1. On the table negative log likelihoods on the test dataset of each method are presented. This likelihoods are calculated by the importance sampling technique [5] with sample count which equals 100. For comparison result from [1] also presented on the table. Also on figures 6 and 5 reconstruction loss and KL divergence during the training are plotted. From the table 1 you can observe that our method produces worse result than presented in the original paper. It is because we use smaller number of epochs. The VAE in the original paper is trained during 5000 epochs. Also each method VAE, VAE-HVI and VAE-HVI-AVO have the same NLL metric. Reconstruction for VAE method is shown on the figure 7 and generation is shown on 8 for the VAE-HVI model.

3 Discussion of results

During this project we managed to implement hierarchical variational inference with stochastic transition operator and normalizing flows. We also implemented annealed variational objective instead of standard ELBO loss. We demonstrated that for multi-model distributions methods with ELBO can not sample from whole distribution and collapses to one mode. However the AVO loss allows to overcome this drawback and fits to the whole distribution. Also we compared HVI with stochastic and deterministic transition operators. The result is that deterministic operator has more narrow variability if we compare it with stochastic transitions. For example, auto regressive normalizing flows can not fetch multi-Gaussian distribution correctly. We also conducted experiments with amortized variational inference. We tested VAE with Gaussian proposal, VAE-HVI and VAE-HVI-AVO. We received metrics which is worse than in the original paper. One of possible reason is that we train on the small number of epochs. But this question requires further investigations.

4 Members contribution

- Mikhail Kurenkov - HVI, HVI-AVO for toy energy fitting and VAE-HVI, VAE-HVI-AVO models implementation, experiments with VAE
- Aleksei Pronkin - Normalizing flows, VAE, IWAE implementation, github merging and refactoring
- Timur Chikichev - Toy models, experiments with model fitting, final report

List of Figures

1 Annealed Variational Objectives. Transitional targets presented on a first three toy energy functions. . 4
2 HVI - AVO, toy energy fitting: target is four-mode mixture of Gaussian 5
3 HVI - ELBO, toy energy fitting: target is four-mode mixture of Gaussian 5
4 Toy energy fitting with inverse auto-regressive flows. Columns left to right: HVI ELBO, IAF ELBO, HVI AVO, IAF AVO. Rows top to bottom: toy models A to F. 6
5 KL part of ELBO during training. Blue line is VAE, green VAE-HVI, orange line VAE-HVI-AVO . . 7
6 NLL part of ELBO during training. Blue line is VAE, green VAE-HVI, orange line VAE-HVI-AVO . 7
7 Reconstruction of VAE. Left is ground truth, right is reconstructed 7
8 Generation of images of VAE 8

5 Resources

Github repository: `https://github.com/alexey-pronkin/annealed`

Presentation: https://docs.google.com/presentation

Acknowledgements

The project represents the paper [1].

We utilize some code from:

- `https://github.com/joelouismarino/iterative_inference/`,
- `https://github.com/jmtomczak/vae_householder_flow`,
- `https://github.com/AntixK/PyTorch-VAE`,
- `https://github.com/haofuml/cyclical_annealing` and
- `https://github.com/ajayjain/lmconv`.

References

[1] C. Huang, Shawn Tan, Alexandre Lacoste, and Aaron C. Courville. Improving explorability in variational inference with annealed variational objectives. *ArXiv*, abs/1809.01818, 2018.

[2] Rajesh Ranganath, Dustin Tran, and David M. Blei. Hierarchical variational models, 2015.

[3] Jakub M Tomczak and Max Welling. Improving variational auto-encoders using householder flow. *arXiv preprint arXiv:1611.09630*, 2016.

[4] Casper Kaae Sønderby, Tapani Raiko, Lars Maaløe, Søren Kaae Sønderby, and Ole Winther. Ladder variational autoencoders, 2016.

[5] Yuri Burda, Roger Grosse, and Ruslan Salakhutdinov. Importance weighted autoencoders, 2015.