IAF AS A DISTRIBUTION OVER TRAJECTORIES

TECHNISCHE UNIVERSITÄT MÜNCHEN

Presented by - Jakob Breuninger, Neha Das, Sumit Dugar

GOAL: MODELLING A DYNAMICAL SYSTEM MODEL

We seek a probabilistic, latent dynamical system model, which given a sequence of controls $u_{0:T}$ and an initial latent state z_0 can generate a sequence of latent states $z_{0:T}$ and corresponding observations $x_{0:T}$.
Such a model should maximize the probability of observing $x_{0:T}$ given $u_{0:T}$ under the assumption of an underlying latent dynamical system. This probability can be formalized with:

$$p(x_{0:T}|u_{0:T}) = \int p(x_{0:T}|z_{0:T}, u_{0:T})p(z_{0:T}|u_{0:T})dz_{0:T}$$

BASELINE DVBF

Our baseline network is a recurrent neural network inspired by the deep variational bayes filter.
In the interest of achieving the goal stated above, we require good long term predictions. This can be achieved, by enforcing a State Space Model (SSM) formulation in latent space. Thus we have:

$$p(x_{0:T}|z_{0:T}) = \prod_{t=0}^{T} p(x_t|z_t) \quad \text{and} \quad p(z_{0:T}|u_{0:T}) = p(z_0)\prod_{t=1}^{T} p(z_t|z_{t-1}, u_{t-1})$$

Now, by applying the Variational Auto Encoder paradigm and replacing the bottleneck in the temporal autoencoder with stochastic units, we can train our model by attempting to maximize the variational lower bound (disregarding $u_{0:T}$ for simplicity):

$$L = \int q(z_{0:T}|x_{0:T}) ln((p(x_{0:T}|z_{0:T})\frac{p(z_{0:T})}{q(z_{0:T}|x_{0:T})})dz_{0:T}$$

The architecture of this model is shown in the adjoining section.

INVERSE AUTOREGRESSIVE FLOW

With **Normalizing Flows** we can transform a computationally cheap posterior q0(z0jx) k-times to be arbitrarily flexible (see Figure below), through invertible mappings zi = f (zi−1; x) [4]:

$$z_k = f_k \circ f_{k-1} \circ f_{k-2} \circ ... f_1(z_0)$$

The **log-likelihood** of the transformed posterior can be calculated through the change of variables theorem by summing up the determinant of the Jacobian of each transformation:

$$\ln q(z_k) = \ln q(z_0) + \sum_{i=1}^{k} \ln\left|\frac{\delta f_i}{\delta z_{i-1}}\right|^{-1}$$

Inverse Autoregressive Flow (IAF) is a flexible type of normalizing flow that scales well to high dimensional latent spaces while remaining computational efficiency [2]. At each IAF step the latent variable is transformed with:

$$z_i = \sigma_{i-1} z_{i-1} + \mu_{i-1}$$

Where $\{\sigma_{i-1}, \mu_{i-1}\}$ are the output of an **autoregressive neural network** with the input $\{z_{i-1}, h\}$
Due to the autoregressive property $\left|\frac{\sigma_i(z_{i-1})}{z_{i-1}}\right|$ and $\left|\frac{\mu_i(z_{i-1})}{z_{i-1}}\right|$ are lower triangular & thus the Jacobian is a trivial sum:

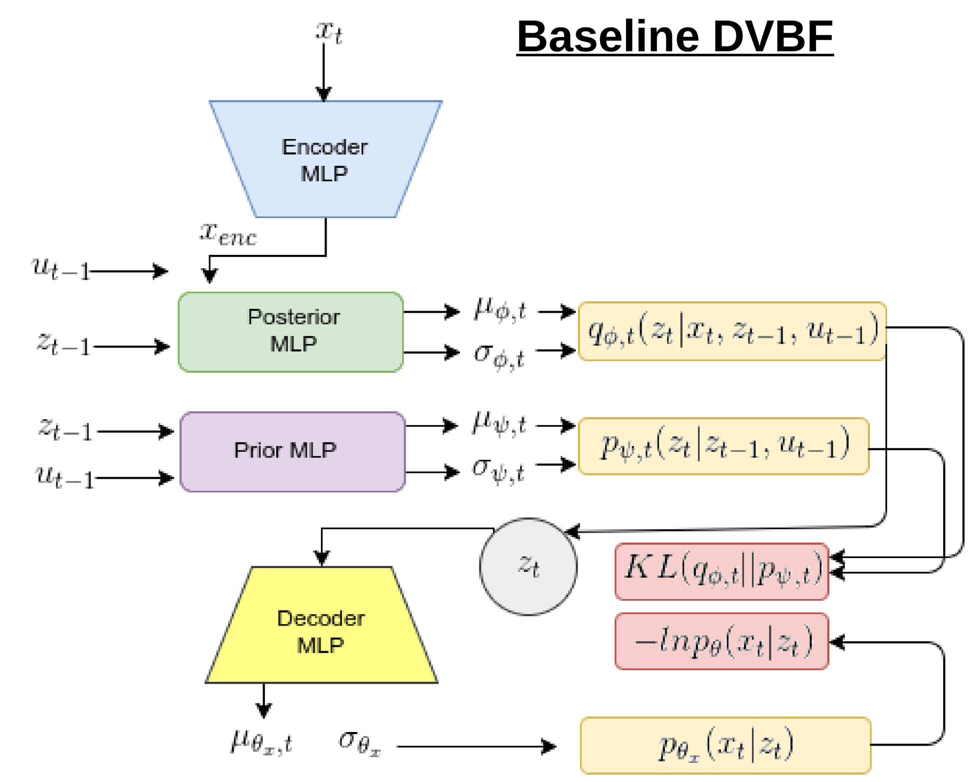
$$\ln\left|\frac{\delta f_i}{\delta z_{i-1}}\right| = \sum_{n=1}^{dim(z)} \ln \sigma_{n,i-1}$$

PROPOSAL

We propose that Inverse Autoregressive Flow be incorporated in the transition of the latent state z through time. This modelling could be used for the posterior $q(z_{0:T}|x_{0:T}, u_{0:T})$. Sampling a trajectory from an IAF transformed posterior would only require only one sample from an auxiliary random variable instead of one per time step.

ARCHITECTURE

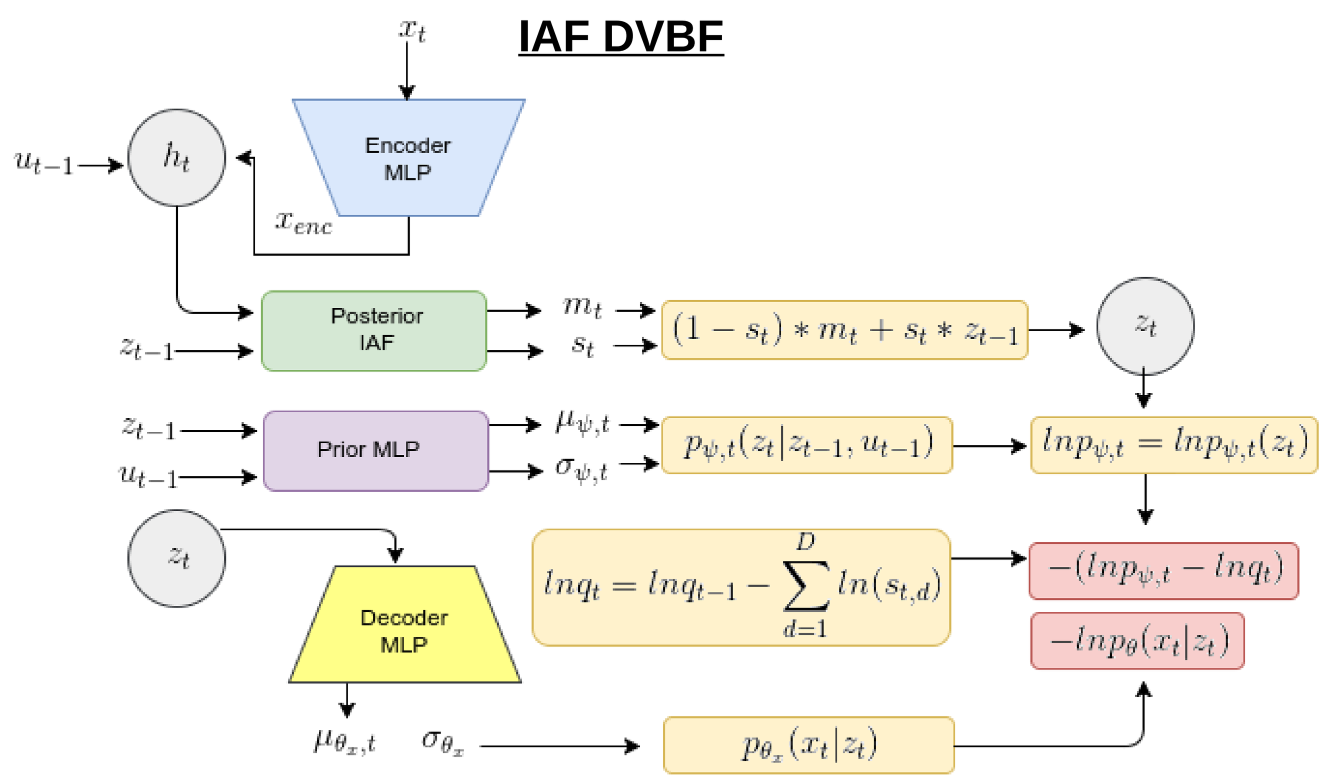
Baseline DVBF



$$L_{ELBO} = E_{z_0 \sim q(z_0|x_0)}[\ln p(x_0|z_0) + \ln p(z_0) - \ln q(z_0|x_0)]$$
$$+ E_{z_{0:t}(z_{0:t}|x_{0:t})}[\sum_{t=1}^{T}[\ln p(x_t|z_t) + \ln p(z_t|z_{t-1}) - \ln q(z_t|x_t, z_{t-1})]]^*$$

* disregarding $u_{0:T}$ for simplicity

IAF DVBF



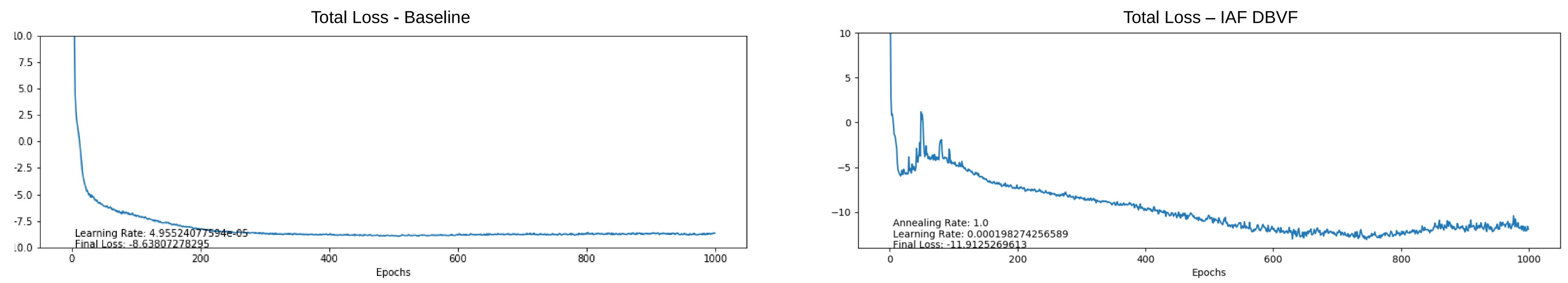
$$L_{ELBO} = E_{z_0 \sim q(z_0|x_0)}[\ln p(x_0|z_0) + \ln p(z_0) - \ln q(z_0|x_0)]$$
$$+ E_{z_{0:t}(z_{0:t}|x_{0:t})}[\sum_{t=1}^{T}[\ln p(x_t|z_t) + \ln p(z_t|z_{t-1}) - \ln q(z_0) + \sum_{i=1}^{t} \ln\left|\frac{\delta z_i}{\delta z_{i-1}}\right|]]^*$$

* disregarding $u_{0:T}$ for simplicity

EXPERIMENTAL RESULTS

Mean absolute difference of the generated trajectory from ground truth : 0.1536

Mean absolute difference of the generated trajectory from ground truth : 0.1915



DATA AND TRAINING

- We train and test our network on data generated from one of the classic conrol environments of OpenAIGym - "Pendulum-v0".
  - Training Set: 500 instances of 50 step trajectories
  - Validation Set: 128 instances of 50 step trajectories
  - Test Set: 128 instances of 50 step trajectories
- Both the networks were trained for 1000 epochs with batch size 16. The initial learning rate for baseline was kept at 0.0005 and for IAF DBVF as 0.001

INSIGHTS

- **Difference in assumptions from baseline** – In order to incorporate IAF model as we do, we had to assume that the distribution q is factorized in a manner different from the baseline.
  - In Baseline:

$$q(z_{0:T}|x_{0:T}, u_{0:T}) = q(z_0|x_0)\prod_{t=1}^{T} q(z_t|x_t, z_{t-1}, u_{t-1})$$

  - In our model:

$$q(z_{0:T}|x_{0:T}, u_{0:T}) = q(z_0|x_0)\prod_{t=1}^{T} q(z_t|x_{0:t}, u_{0:t-1})$$

  However, this doesn't conflict with our enforcement of a state space model in the latent space.
- **Annealing –** In order to stabilize our training and ensure that the loss continues to decrease to negative values, we annealed the KL Loss over 600 epochs.

SETBACKS AND LIMITATIONS

- **Close but not improved results** – As you can see, while the generated trajectory is very close to the baseline, they are still slightly worse and do not improve upon them.
- **Negative KL Loss –** Towards the middle of the training, the KL loss keeps on decreasing and turns negative, which is something that shouldn't happen. This may indicate that there's a flaw in our method.

CONCLUSION

Our method produces comparable (but slightly worse) results and needs only one sampling step per trajectory as opposed to one sampling step per time step, thus reducing the computational cost marginally. However, further investigation is needed for examining what goes wrong with the KL Loss.

REFERENCES

- Maximilian Karl, Maximilian Soelch, Justin Bayer, Patrick van der Smagt, **"Deep Variational Bayes Filters: Unsupervised Learning of State Space Models from Raw Data",** *Proceedings of the International Conference on Learning Representations (ICLR), 2017*
- M. Germain, K. Gregor, I. Murray, and H. Larochelle, **"MADE: masked autoencoder for distribution estimation",** *CoRR, abs/1502.03509, 2015.*
- D. P. Kingma, T. Salimans, and M. Welling, **"Improving variational inference with inverse autoregressive flow",** *CoRR, abs/1606.04934, 2016.*
- D. P. Kingma and M. Welling, **"Auto-encoding variational bayes",** *CoRR, abs/1312.6114, 2013.*
- D. J. Rezende and S. Mohamed **"Variational inference with normalizing flows",** *In D. Blei and F. Bach, editors, Proceedings of the 32nd International Conference on Machine Learning (ICML-15), pages 1530–1538. JMLR Workshop and Conference Proceedings,*