Architecure details of DSA (improved, better stability)

The dynamics network for $p(z_{t+1}|z_t)$:

$$\begin{tikzcd}
c_t \arrow[r] & \mathrm{MLP}_1 \arrow[r] & \hat{c}_t \arrow[dr] & & c_{t+1} & & \\
z_t \arrow[r] \arrow[ur] \arrow[dr] & \mathrm{MLP}_2 \arrow[r] & \phi_t \arrow[r] & \begin{array}{c}\text{LSTM}\\ \text{cell}\end{array} \arrow[ur] \arrow[dr] & & \mathrm{MLP}_4 \arrow[r] & z_{t+1} \\
h_t \arrow[r] & \mathrm{MLP}_3 \arrow[r] & \hat{h}_t \arrow[ur] & & h_{t+1} \arrow[ur] & &
\end{tikzcd}$$

MLP_1: concat(z_t, c_t) $\rightarrow \hat{c}_t$, one-layer network, tanh activation.
MLP_2: $z_t \rightarrow \phi_t$, one-layer network, tanh activation.
MLP_3: concat(z_t, h_t) $\rightarrow \hat{h}_t$, one-layer network, tanh activation.

LSTM cell: apply LSTM equations with ϕ_t as the current input, $\hat{h}_t$ as the previous hidden state, and $\hat{c}_t$ as the previous cell state.

MLP_4: $h_t \rightarrow \mu, \log\sigma$ of $p(z_{t+1}|z_t)$, two-layer network [dimH, dimH, dimZ×2], tanh activation for the first layer and linear activation for the second layer.

$p(f)$ is simply standard normal.

We need to balance the power of $p(f)$ and $p(z_{1:T})$. This means dimZ, dimF and dimH need to be chosen carefully. My empirical choice is dimZ=32, dimF=256 and dimH=256.

$p(x_t|z_t, f)$ is defined by a deconvolution neural network with ReLU activation (except the last layer which uses sigmoid). The architecture is like

- concat(f, z_t) $\rightarrow \Phi_t$ using a two-layer MLP with size [dimZ+dimF, dimHidden, $4 \times 4\times$nChannel], ReLU activation;

- $\Phi_t \rightarrow x_t$ using a deconv net with filter size 3, shape [$4 \times 4\times$nChannel, $8\times8\times$nChannel, $16\times16\times$nChannel, $32\times32\times$nChannel, $64\times64\times$nChannel, $64 \times 64 \times 3$], ReLU activation except for the last deconv layer which uses sigmoid.

The dimensions of the deconv net doesn't matter to much for disentanglement, although using a big value for dimHidden and nChannel would improve the frame quality. I use for example dimHidden=512 and nChannel=256.

I don't think the encoder matters too much in terms of disentanglement, although the image quality can differ. Can simply try the fully factorised one $q(f, z_{1:T}|x_{1:T}) = q(f|x_{1:T})q(z_t|x_t)$ and share a feature extractor for both $q(f)$ and $q(z_t)$.

On mixing deterministic and stochastic dynamics

A more rigorous way to write the prior dynamics would be to define $p(z_{t+1}, h_{t+1}, c_{t+1}|z_t, h_t, c_t)$, where

$$p(z_{t+1}, h_{t+1}, c_{t+1}|z_t, h_t, c_t) = p(z_{t+1}|h_{t+1}, c_{t+1}, z_t, h_t, c_t)p(h_{t+1}, c_{t+1}|z_t, h_t, c_t),$$

$$p(h_{t+1}, c_{t+1}|z_t, h_t, c_t) = \delta([h_{t+1}, c_{t+1}] = g(z_t, h_t, c_t)),$$

$$p(z_{t+1}|h_{t+1}, c_{t+1}, z_t, h_t, c_t) = p(z_{t+1}|h_{t+1}) = \mathcal{N}(z_{t+1}; \mu(h_{t+1}), \sigma^2(h_{t+1})).$$

Here we use dirac measure for the LSTM states h_t and c_t, and we see that this is in fact a mix of deterministic/stochastic dynamics.

For approximate posterior we simply define (for the fully factorised case)

$$q(z_{t+1}, h_{t+1}, c_{t+1}|z_t, h_t, c_t, x_{1:T}) = q(z_{t+1}|x_{t+1})p(h_{t+1}, c_{t+1}|z_t, h_t, c_t).$$

So that it returns the ELBO

$$\mathcal{L} = \sum_{t=1}^{T} E_{q(f, z_{1:T}|x_{1:T}) \prod_{t=1}^{T} p(h_t, c_t|z_{t-1}, h_{t-1}, c_{t-1})} \left[\log \frac{p(f)p(z_t|h_t)p(x_t|f, z_t)}{q(f|x_{1:T})q(z_{t+1}|x_{t+1})} \right],$$

Using LOTUS we can rewrite the lower-bound

$$\mathcal{L} = \sum_{t=1}^{T} E_{q(f, z_{1:T}|x_{1:T})} \left[\log \frac{p(f)p(z_t|h_t = \text{prior-net}(z_{t-1}, h_{t-1}, c_{t-1}))p(x_t|f, z_t)}{q(f|x_{1:T})q(z_{t+1}|x_{t+1})} \right].$$