Stochastic Neural Networks with Variational Inference

Kyeong Ryeol, Go
M.S. Candidate of OSI Lab

Deep Latent Gaussian Models (DLGMs)

- Each layer's variables are drawn from MLP of previous layers with gaussian noise
- Generative Process

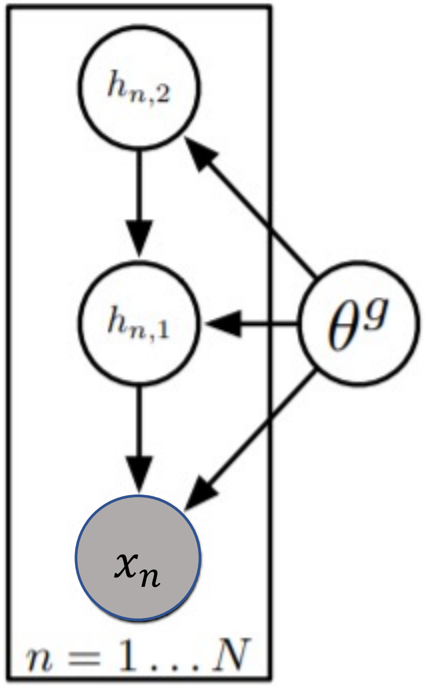
  (Generative model : $p(x, h) = p(x|h_1, \theta^g)p(h_L|\theta^g)p(\theta^g)\prod_{l=1}^{L-1} p(h_l|h_{l+1}, \theta^g)$)

  - Prior : $\theta^g \sim N(0, \kappa I)$)
  - Gaussian noise : $\xi_l \sim N(0, I) \quad l = 1, \dots, L$
  - Hidden layer : $h_l = \begin{cases} T_l(h_{l+1}) + G_l\xi_l & l = 1, \dots, L-1 \\ G_L\xi_L & l = L \end{cases}$ where $G_l : matrix$ and $T_l : MLP$
  - Observation : $x \sim \pi\left(T_0(h_1)\right)$

- Stochastic backpropagation

  (f is a loss function that is smooth and integrable)

  1. Gaussian backpropagation
     - $\nabla_{\mu_i} E_{\xi \sim N(\mu,C)}[f(\xi)] = E_{\xi \sim N(\mu,C)}\left[\nabla_{\xi_i} f(\xi)\right] \coloneqq E_{\xi \sim N(\mu,C)}[g_i]$
     - $\nabla_{C_{ij}} E_{\xi \sim N(\mu,C)}[f(\xi)] = \frac{1}{2} E_{\xi \sim N(\mu,C)}\left[\nabla^2_{\xi_i,\xi_j} f(\xi)\right] \coloneqq \frac{1}{2} E_{\xi \sim N(\mu,C)}[H_{ij}]$
     - $\nabla_\theta E_{\xi \sim N(\mu(\theta),C(\theta))}[f(\xi)] = E_{\xi \sim N(\mu(\theta),C(\theta))}\left[g^T \frac{\partial \mu(\theta)}{\partial \theta} + \frac{1}{2} Tr\left(H \frac{\partial C(\theta)}{\partial \theta}\right)\right]$
  2. Co-ordinate transformation
     - $\xi \sim N(\mu, C) = N\left(\mu, RR^T\right)$ and $\epsilon \sim N(0, I) \quad \longrightarrow \quad \xi = \mu + R\epsilon$
     - $\nabla_R E_{N(\mu,C)}[f(\xi)] = \nabla_R E_{N(0,I)}[f(\mu + R\epsilon)] = E_{N(0,1)}[\epsilon g^T]$

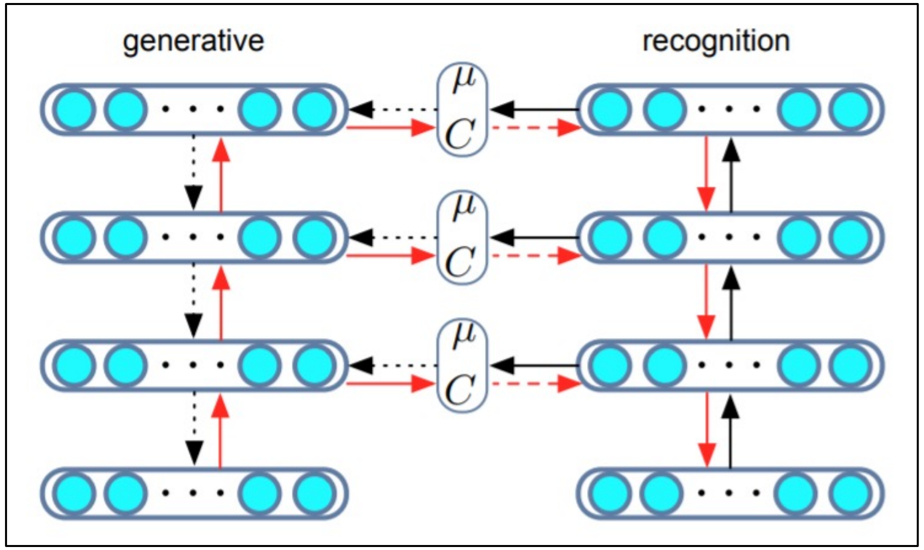


(continued)

- Free energy objective

  (Recognition model : $q(\xi|X,\theta^r) = \prod_{n=1}^{N}\prod_{l=1}^{L} N\big(\mu_l(x_n), C_l(x_n)\big)$)

  - $F(X) = KL(q(\xi|X,\theta^r)\|p(\xi)) - E_{\xi\sim q(\xi|X,\theta^r)}[\log p(X|\xi,\theta^g)p(\theta^g)]$
  - $\nabla_{\theta_j^g} F(X) = -E_q[\nabla_{\theta_j^g}\log p(X|h)] + \frac{1}{\kappa}\theta_j^g$
  - $\nabla_{\theta^r} F(x) = \nabla_{\mu} F(x)^T \frac{\partial\mu}{\partial\theta^r} + Tr(\nabla_R F(x)\frac{\partial R}{\partial\theta^r})$

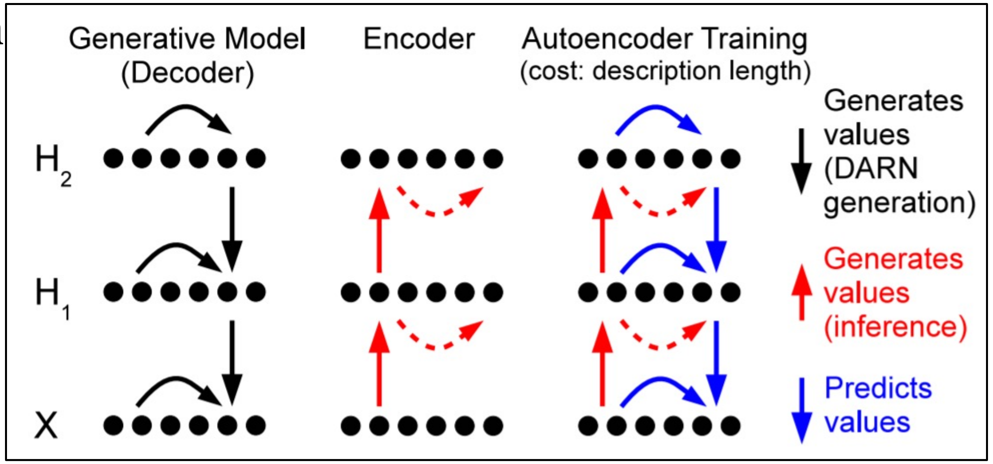


- Covariance parameterization
  1. $C = diag(d)$ where d is a $k-$dimensional vector
  2. $C^{-1} = D + uu^T = RR^T$ where $D = diag(d)$
     - $C = D^{-1} - \eta D^{-1}uu^TD^{-1}$ where $\eta = \frac{1}{u^TD^{-1}u+1}$ and $\log|C| = \log\eta - \log|D|$
     - $R = D^{-1/2} - \left[\frac{1-\sqrt{\eta}}{u^TD^{-1}u}\right]D^{-1}uu^TD^{-1/2}$

Deep Auto-Regressive Networks (DARNs)

- Each layer's variables are computed by previous layers and the units from the current layers in auto-regressive manner.
- A single stochastic hidden layer

  ($h_i \in \{0,1\}$ and every conditional probability can
  - Prior : $p(h) = \prod_{j=1}^{n_h} p(h_j | h_{1:j-1})$
  - Encoder : $q(h|x) = \prod_{j=1}^{n_h} q(h_j | h_{1:j-1}, x)$
  - Decoder : $p(x|h) = \prod_{j=1}^{n_x} p(x_j | x_{1:j-1}, h)$



- Deeper model architecture
  1. Adding stochastic hidden layers

     ($H^0 = X,\ H^{n_{layers}+1} = \Phi$)
     - $p(h^l | h^{l+1}) = \prod_{j=1}^{n_h^l} p(h_j^l | h_{1:j-1}^l, h^{l+1})$ *where* $l = 0, \ldots, n_{layers}$
     - $q(h^k | h^{k-1}) = \prod_{j=1}^{n_h^k} q(h_j^k | h_{1:j-1}^k, h^{k-1})$ *where* $k = 1, \ldots, n_{layers}$
  2. Adding deterministic hidden layers
     - $p(H_j^l = 1 | h_{1:j-1}^l, h^{l+1}) = \sigma(W_j^l \cdot (h_{1:j-1}^l, \tanh(U h^{l+1})) + b_j^l)$
  3. Using alternate kinds of auto-regressive structure
     - NADE or EoNADE

(continued)

- Minimum Description Length (MDL) principle
  - Find parameter that maximally compress the training data x
  - Description length : number of bits needed to communicate the particular value
    1. Sample a representation of h to communicate
       - Bits back coding : $L(h) = -\log_2 p(h) + \log_2 q(h|x)$
    2. Send the residual of x relative to h
       - Shannon's source coding theorem : $L(x|h) = -\log_2 p(x|h)$
  - Expected description length($\approx ELBO$)
    - $L(x) = \sum_h q(h|x)\big(L(h) + L(x|h)\big) = \sum_h q(h|x)(\log_2 q(h|x) - \log_2 p(x,h))$
    - $\nabla_\theta L(x) = \sum_h q(h|x)\nabla_\theta \log q(h|x)\,(\log_2 q(h|x) - \log_2 p(x,h))$

      $\coloneqq \sum_h q(h|x)\nabla_\theta \log q(h|x)\, f(h)$

      $\approx \sum_h q(h|x)\nabla_\theta \log q(h|x)\,(f(h) - b(h)) = \sum_h q(h|x)\nabla_\theta \log q(h|x)\frac{df(h)}{dh}\left(h - \frac{1}{2}\right)$

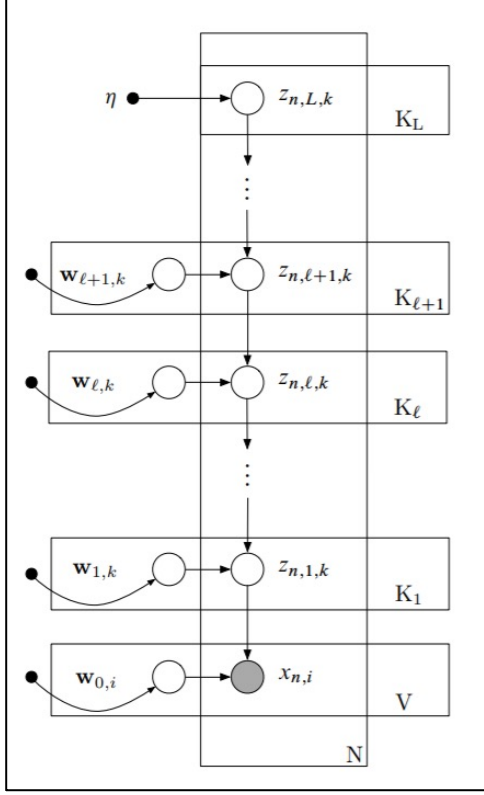
      $= \sum_h q(h|x)\frac{\nabla_\theta q(H=1)}{2q(h)}\frac{df(h)}{dh}$
    - $b(h) = f(h) + \frac{df(h)}{dh}(h' - h)\ \ s.t.\ \sum_h q(h|x)\,\nabla_\theta \log q(h|x)\, b(h) = 0$

      (1^{st} order Taylor approximation of f around h evaluated at h'=1/2)

Deep Exponential Families (DEFs)

- Deep Exponential families
  - One layer controls the natural parameters of the next
    - Top most layer : $p\left(z_{L,k}\right) = EXPFAM_L\left(z_{L,k}, \eta\right) \;\rightarrow\; \eta\ is\ the\ hyperparameter$
    - Following layers : $p\left(z_{l,k} \middle| z_{l+1}, W_l\right) = EXPFAM_l\left(z_{l,k}, g_l\left(z_{l+1}^T w_{l,k}\right)\right)$ where $l = 1, \dots, L-1$
    - Lowest layer : $p(x_i | z_1, W_0) = Poisson\left(z_1^T w_{0,i}\right) \;\rightarrow\; enrty\ of\ W_0\ is\ gamma\ distributed$
    - $E\left[T\left(z_{l,k}\right)\right] = \nabla_\eta a\left(g_l\left(z_{l+1}^T w_{l,k}\right)\right)$

1. Sparse gamma DEF : $z_{l+1} = Gamma\ R.V.$
   - Control the expected activation of the next layer while the shape is fixed to be less than 1
   - $p\left(z_{l,k} | z_{l+1}, W_l\right) = z_{l,k}^{-1} \exp\left(\alpha_l \log z_{l,k} - \beta_l z_{l,k} - \log\Gamma\left(\alpha_{l,k}\right) - \alpha_{l,k} \log \beta_{l,k}\right)$
   - $\alpha_l = g_{\alpha_l}\left(z_{l+1}^T w_{l,k}\right) = \alpha_{l+1}, \quad \beta_l = g_{\beta_l}\left(z_{l+1}^T w_{l,k}\right) = \frac{\alpha_l}{z_{l+1}^T w_{l,k}} \;\rightarrow\; E\left[z_{l,k}\right] = \frac{\alpha_l}{\beta_l} = z_{l+1}^T w_{l,k}$
   - $entry\ of\ W_l\ is\ gamma\ distributed\ in\ a\ factorized\ manner$
2. Sigmoid belief network : $z_{l+1} = Bernoulli\ R.V.$
   - $p\left(z_{l,k} \middle| z_{l+1}, W_l\right) = \exp\left(\eta_l z_{l,k} - \log\left(1 + \exp\left(z_{l+1}^T w_{l,k}\right)\right)\right)$
   - $\eta_l = g_l\left(z_{l+1}^T w_{l,k}\right) = z_{l+1}^T w_{l,k} \;\rightarrow\; E\left[z_{l,k}\right] = \nabla_\eta \log\left(1 + \exp\left(z_{l+1}^T w_{l,k}\right)\right) = 1/\left(1 + \exp\left(-z_{l+1}^T w_{l,k}\right)\right)$
   - $entry\ of\ W_l\ is\ normally\ distributed\ in\ a\ factorized\ manner$
3. Poisson DEF : $z_{l+1} = Poisson\ R.V.$
   - $p\left(z_{l,k} \middle| z_{l+1}, W_l\right) = \left(z_{l,k}!\right)^{-1} \exp\left(\eta_l z_{l,k} - z_{l+1}^T w_{l,k}\right)$
   - $\eta_l = g_l\left(z_{l+1}^T w_{l,k}\right) = \log\left(z_{l+1}^T w_{l,k}\right) \;\rightarrow\; E\left[z_{l,k}\right] = \nabla_\eta \log z_{l+1}^T w_{l,k} = z_{l+1}^T w_{l,k}$
   - $entry\ of\ W_l\ is\ gamma\ distributed\ in\ a\ factorized\ manner$
   - $entry\ of\ W_l\ is\ normally\ distributed\ in\ a\ factorized\ manner\ when\ using\ log - softmax\ link\ function$



(continued)

- Inference

  (z : all latent variables associated with the observations)

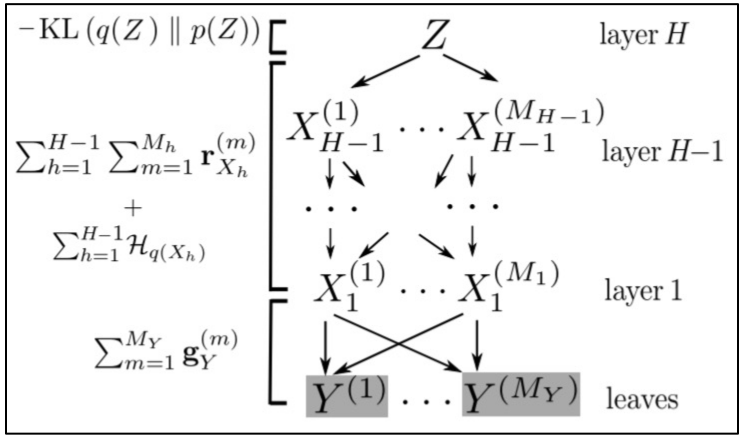
  (W : all latent variables shared across observations)

  - $L(x) = E_{q(z,W)}[\log p(x,z,W) - \log q(z,W)]$
    - $q(z,W) = q(W_0)\prod_{l=1}^{L} q(W_l;\xi_l)\prod_{n=1}^{N}\prod_k q\left(z_{n,l,k};\lambda_{n,l,k}\right)$
    - $q(W_l;\xi_l), q\left(z_{n,l,k};\lambda_{n,l,k}\right)$ $follow\ the\ same\ distribution\ as$ $p(W_l), p(z_{n,l,k}|z_{n,l+1}, W_l)$

    1. $\nabla_{\lambda_{n,l,k}} L(x) = E_{q(z_{n,l,k};\lambda_{n,l,k})}\left[\nabla_{\lambda_{n,l,k}} \log q\left(z_{n,l,k};\lambda_{n,l,k}\right)\left(\log p_{n,l,k}(x,z,W) - \log q\left(z_{n,l,k};\lambda_{n,l,k}\right)\right)\right]$
    2. $\nabla_{\xi_l} L(x) = E_{q(W_l;\xi_l)}\left[\nabla_{\xi_l} \log q(W_l;\xi_l)\left(\log p_{n,l,k}(x,z,W) - \log q(W_l;\xi_l)\right)\right]$
       - $\log p_{n,1,k}(x,z,W) = \log p(z_{n,1,k}|z_{n,2}, w_{1,k}) + \log p(x_n|z_{n,1}, W_0)$
       - $\log p_{n,l,k}(x,z,W) = \log p(z_{n,l,k}|z_{n,l+1}, w_{l,k}) + \log p(z_{n,l-1}|z_{n,l}, W_{l-1})$
       - $\log p_{n,L,k}(x,z,W) = \log p(z_{n,L,k}) + \log p(z_{n,L-1}|z_{n,L}, W_{L-1})$

- Double DEFs for pairwise data
  - Use two DEFs one for the latent representation of user and the other for items
  - Replace W_0 with another DEFs
    - $p\left(x_{i,j}\middle|z_{i,1}^{c}, z_{j,1}^{r}\right) = Poisson\left({z_{i,1}^{c}}^{T} z_{j,1}^{r}\right)$

Deep Gaussian Processes



- $Y: N\times D,\ X_h: N\times Q_h\ (h=1,\ldots,H-1),\ Z: N\times Q_H,\ \tilde{X}: K\times Q,\ \tilde{Z}: K\times H$
  - $F^Y=\{f_d^Y\}_{d=1}^{D},\ F^X=\{f_q^X\}_{q=1}^{Q},\ \mathrm{U^Y}=\{\mathrm{u_d^Y}\}_{\mathrm{d=1}}^{\mathrm{D}},\ \mathrm{U^X}=\{\mathrm{u_q^X}\}_{\mathrm{q=1}}^{\mathrm{Q}}$
  - $y_{nd}=f_d^Y(x_n)+\epsilon_{nd}^Y,\ u_{nd}^Y=f_d^Y(\tilde{x}_n)+\epsilon_{nd}^Y\ \ where\ \ f_d^Y\sim GP\big(0,k^Y(\cdot)\big)\ \ for\ all\ p$
  - $x_{nq}=f_q^X(z_n)+\epsilon_{nq}^X,\ u_{nq}^X=f_q^X(\tilde{z}_n)+\epsilon_{nq}^X\ \ where\ \ f_q^X\sim GP\big(0,k^X(\cdot)\big)\ \ for\ all\ q$
- Variational parameters with sparse approximations
  - $q(X)=\prod_{q=1}^{Q}N(\mu_q^X,S_q^X),\ q(Z)=\prod_{h=1}^{H}N(\mu_h^Z,S_h^Z)$
  - $G(Y,F^Y,U^Y,X)=p(F^Y|U^Y,X)q(U^Y)q(X),\ R(X,F^X,U^X,Z)=p(F^X|U^X,Z)q(U^X)q(Z)$
  - $q(F^Y,U^Y,X,F^X,U^X,Z)=G(Y,F^Y,U^Y,X)R(X,F^X,U^X,Z)$
  - $\log p(Y)$
    $$\geq\int q(F^Y,U^Y,X,F^X,U^X,Z)\log\frac{p(Y,F^Y,U^Y,X,F^X,U^X,Z)}{q(F^Y,U^Y,X,F^X,U^X,Z)}dF^Y dU^Y dX\,dF^X\,dU^X\,dZ$$
    $$=\int q(F^Y,U^Y,X,F^X,U^X,Z)\log\frac{p(Y|F^Y)p\big(U^Y|\tilde{X}\big)p(X|F^X)p(U^X|\tilde{Z})p(Z)}{q(U^Y)q(U^X)q(Z)}dF^Y dU^Y dX\,dF^X\,dU^X\,dZ$$
    $$=g_Y+r_X+\mathcal{H}\big(q(X)\big)-KL\big(q(Z)\|p(Z)\big)$$
    - $g_Y=E_{G(Y,F^Y,U^Y,X)}\left[\log p(Y|F^Y)+\log\frac{p(U^Y)}{q(U^Y)}\right],\ r_X=E_{R(X,F^X,U^X,Z)q(X)}\left[\log p(X|F^X)+\log\frac{p(U^X)}{q(U^X)}\right]$
- Extending hierarchy
  - $\log p(Y)\geq\sum_{m=1}^{M_Y}g_Y^m+\sum_{h=1}^{H-1}\sum_{m=1}^{M_{X_h}}r_{X_h}^m+\sum_{h=1}^{H-1}\mathcal{H}\big(q(X_h)\big)-KL(q(Z)\|p(Z))$

Hierarchical Variational Models (HVMs)

- Capture both posterior dependencies between the latent variables and more complex marginal distributions thus better inferring the posterior
  - $q_{HVM}(z;\theta) = \int q(\lambda;\theta) \prod_i q(z_i|\lambda_i)\, d\lambda$

  1. Draws variational parameters from a variational prior $q(\lambda;\theta)$
     - Mixture of gaussians : $q(\lambda;\theta) = \sum_{i=1}^{k} \pi_k N(\mu_k, \Sigma_k)$
       - Impractical and not scalable to high dimensions
     - Normalizing flows : $q(\lambda;\theta) = q(\lambda_0) \prod_{k=1}^{K} \left|\det\left(\frac{\partial f_k}{\partial \lambda_k}\right)\right|^{-1}$ $where\ \lambda_k = f_k \circ \cdots \circ f_1(\lambda_0)$
  2. Draw latent variables from the corresponding likelihood $q_{MF}(z|\lambda)$

- Hierarchical ELBO
  - $$\begin{aligned} L(\theta) &= E_{q_{HVM}(z;\theta)}[\log p(x,z) - \log q_{HVM}(z;\theta)] \\ &\geq E_{q(z,\lambda;\theta)}[\log p(x,z) - \log q(\lambda;\theta) - \log q_{MF}(z|\lambda) + \log r(\lambda|z;\phi)] \\ &= E_{q(z,\lambda;\theta)}\left[\log p(x,z) - \textstyle\sum_{i=1}^{d} \log q(z_i|\lambda_i) + \log r(\lambda|z;\phi) - \log q(\lambda;\theta)\right] \\ &= E_{q(\lambda;\theta)}[L_{MF}(\lambda)] + E_{q(z,\lambda;\theta)}[\log r(\lambda|z;\phi) - \log q(\lambda;\theta)] \coloneqq \tilde{L}(\theta,\phi) \end{aligned}$$

  - $\nabla_\theta \tilde{L}(\theta,\phi)$

    $$\begin{aligned} &= E_\epsilon\left[\nabla_\theta \lambda(\epsilon;\theta)\left[\nabla_\lambda L_{MF}(\lambda) + \nabla_\lambda E_{q_{MF}(z|\lambda)}[\log r(\lambda|z;\phi)] - \nabla_\lambda \log q(\lambda;\theta)\right] + \nabla_\theta \log q(\lambda;\theta)\right] \\ &= E_\epsilon\left[\nabla_\theta \lambda(\epsilon;\theta)\left[\nabla_\lambda L_{MF}(\lambda) + E_{q_{MF}(z|\lambda)}[\nabla_\lambda \log q_{MF}(z|\lambda) \log r(\lambda|z;\phi) + \nabla_\lambda \log r(\lambda|z;\phi)] - \nabla_\lambda \log q(\lambda;\theta)\right]\right] \end{aligned}$$

    ($\because$ $E_\epsilon[\nabla_\theta \log q(\lambda;\theta)] = E_{q(\lambda;\theta)}[\nabla_\theta \log q(\lambda;\theta)] = 0$)

  - $\nabla_\phi \tilde{L}(\theta,\phi) = E_{q(z,\lambda;\theta)}\left[\nabla_\phi r(\lambda|z;\phi)\right]$

(continued)

- Reducing variance
  - Rao-Blackwellizing (Localizing)

$\rightarrow E_{q_{MF}(z|\lambda)}[\nabla_\lambda \log q(z|\lambda) \log r(\lambda|z;\phi) + \nabla_\lambda \log r(\lambda|z;\phi)]$

$= \sum_{i=1}^{d} E_{q_{MF}(z|\lambda)}[\nabla_\lambda \log q(z_i|\lambda_i) \log r(\lambda|z;\phi)]$

$= \sum_{i=1}^{d} E_{q_{MF}(z|\lambda)}[\nabla_\lambda \log q(z_i|\lambda_i) (\log r_i(\lambda|z;\phi) + \log r_{-i}(\lambda|z;\phi))]$

$= \sum_{i=1}^{d} E_{q(z_i|\lambda)} \left[\nabla_\lambda \log q(z_i|\lambda_i) E_{q(z_{-i}|\lambda)}[\log r_i(\lambda|z;\phi) + \log r_{-i}(\lambda|z;\phi)] \right]$

$= \sum_{i=1}^{d} E_{q(z_i|\lambda)}[\nabla_\lambda \log q(z_i|\lambda_i) \log r_i(\lambda|z;\phi)]$

$= \sum_{i=1}^{d} E_{q_{MF}(z|\lambda)}[\nabla_\lambda \log q(z_i|\lambda_i) \log r_i(\lambda|z;\phi)]$

($\because E_{q(z_{-i}|\lambda)}[\log r_{-i}(\lambda|z;\phi)]$ *is a function of* z_{-i} and an expectation of the score function of a distribution is zero)

$\rightarrow \log r(\lambda|z) = \log r(\lambda_0|z) + \sum_{k=1}^{K} \log \left|\det\left(\frac{\partial g_k^{-1}}{\partial \lambda_k}\right)\right| \quad where \ \ \lambda_k = g_k \circ \cdots \circ g_1(\lambda_0)$

($inverse\ functions\ g^{-1} have\ a\ known\ parametric\ form$)

1. $r(\lambda|z)$ is differentiable with respect to λ
2. $r(\lambda|z)$ is flexible enough to model the variational posterior $q(\lambda|z)$
3. $r(\lambda|z)$ factorize with respect to its dependence on each z_i : $r(\lambda_0|z) = \prod_{i=1}^{d} r(\lambda_{0,i}|z_i)$

Ladder Variation Autoencoders (LVAE)

- Inference model recursively corrects the generative model with a data dependent approximate likelihood term
  - Generative model : $p_\theta(x|z) = p_\theta(z_L) \prod_{i=1}^{L-1} p_\theta(z_i|z_{i+1})\, p_\theta(x|z_1)$
    - Stochastic upward pass
      - $p_\theta(z_L) = N(0, I)$
      - $p_\theta(z_i|z_{i+1}) = N\left(\mu_{p,i}, \sigma_{p,i}^2\right)$ $for\ \ i = 0, \ldots, L-1$ where $z_0 = x$
        - $d_{p,i} = MLP(z_{i+1})$
        - $\mu_{p,i} = Linear\left(d_{p,i}\right),\ \ \sigma_{p,i}^2 = Softplus\left(Linear(d_i)\right)$
  - Inference model : $q_\phi(z|x) = q_\phi(z_L|x) \prod_{i=1}^{L-1} q_\phi(z_i|z_{i+1})$

    1. Deterministic upward pass
      - $q_\phi(z_i|z_{i-1}) = N\left(\hat{\mu}_{q,i}, \hat{\sigma}_{q,i}^2\right)\ \ for\ \ i = 1, \ldots, L\ \ where\ z_0 = x$
        - $\hat{d}_{q,i} = MLP\left(\hat{d}_{q,i-1}\right)\ \ where\ \ \hat{d}_{q,0} = x$
        - $\hat{\mu}_{q,i} = Linear\left(\hat{d}_{q,i}\right),\ \ \hat{\sigma}_{q,i}^2 = Softplus\left(Linear(d_i)\right)$

    2. Stochastic downward pass
      - $q_\phi(z_L|x) = N\left(\mu_{q,L}, \sigma_{q,L}^2\right) = N(\hat{\mu}_{q,L}\ \hat{\sigma}_{q,L}^2)$
      - $q_\phi(z_i|z_{i+1}) = N\left(\mu_{q,i}, \sigma_{q,i}^2\right)\ \ for\ \ i = 1, \ldots, L-1$
        - $\mu_{q,i} = \frac{\hat{\mu}_{q,i}\hat{\sigma}_{q,i}^{-2} + \mu_{p,i}\sigma_{p,i}^{-2}}{\hat{\sigma}_{q,i}^{-2} + \sigma_{p,i}^{-2}},\ \ \sigma_{q,i}^2 = \left(\frac{1}{\hat{\sigma}_{q,i}^{-2} + \sigma_{p,i}^{-2}}\right)^2$
- $L(x) = E_{q_\phi(z|x)}[\log p_\theta(x|z)] - \beta KL\left(q_\phi(z|x) \| p_\theta(z)\right)$
  - Need warm-up for β that increases linearly from 0 to 1 during the first N_t epochs of training
  - Batch normalization was critical for the improved performance

Importance Weighted Auto-Encoder (IWAE)

- Tighter lower bound derived from importance weighting which leads to richer representation
  1. Generative model : $p(x|\theta) = \sum_{h^1,\ldots,h^L} p(h^L|\theta)p(\mathrm{h}^{L-1}|h^L,\theta)\cdots p(x|h^1,\theta)$
  2. Recognition model : $q(h|x,\theta) = q(h^1|x,\theta)q(h^2|h^1,\theta)\cdots q(h^L|h^{L-1},\theta)$
  - $\log p(x) = \log E_{h\sim q(h|x,\theta)}\left[\frac{p(x,h|\theta)}{q(h|x,\theta)}\right] = \log E_{h\sim q(h|x,\theta)}\left[\frac{1}{k}\sum_i \frac{p(x,h_i|\theta)}{q(h_i|x)}\right]$

    $\geq E_{h\sim q(h|x,\theta)}\left[\log\frac{1}{k}\sum_i \frac{p(x,h_i|\theta)}{q(h_i|x,\theta)}\right] \coloneqq L_k$
  - $L_k = E_{h_1,\ldots,h_k\sim q(h|x,\theta)}\left[\log\frac{1}{k}\sum_i \frac{p(x,h_i|\theta)}{q(h_i|x,\theta)}\right] = E_{h_1,\ldots,h_k\sim q(h|x,\theta)}\left[\log E_{I=\{i_1,\ldots,i_m\}}\left[\frac{1}{m}\sum_j \frac{p(x,h_{i_j}|\theta)}{q(h_{i_j}|x,\theta)}\right]\right]$

    $\geq E_{h_1,\ldots,h_k\sim q(h|x,\theta)}\left[E_{I=\{i_1,\ldots,i_m\}}\left[\log\frac{1}{m}\sum_j \frac{p\left(x,h_{i_j}|\theta\right)}{q\left(h_{i_j}|x,\theta\right)}\right]\right] = E_{h_1,\ldots,h_m\sim q(h|x,\theta)}\left[\log\frac{1}{m}\sum_j \frac{p\left(x,h_{i_j}|\theta\right)}{q\left(h_{i_j}|x,\theta\right)}\right]$

    $= L_m\ (= L(x)\ \ when\ \ m = 1)$

    $(\therefore \log p(x) \approx \lim_{k\to\infty} L_k \geq L_k \geq L_m\ \ when\ \ k \geq m)$

  - $\nabla_\theta L(x) = \nabla_\theta E_{h\sim q(h|x,\theta)}\left[\log\frac{p(x,h|\theta)}{q(h|x,\theta)}\right] = \nabla_\theta E_{\epsilon\sim N(0,I)}\left[\log\frac{p(x,h(\epsilon,x,\theta)|\theta)}{q(h(\epsilon,x,\theta)|x,\theta)}\right]$

    $= E_{\epsilon\sim N(0,I)}\left[\nabla_\theta \log\frac{p(x,h(\epsilon,x,\theta)|\theta)}{q(h(\epsilon,x,\theta)|x,\theta)}\right] \approx \frac{1}{k}\sum_i \nabla_\theta \log\frac{p(x,h(\epsilon_i,x,\theta)|\theta)}{q(h(\epsilon_i,x,\theta)|x,\theta)}$
  - $\nabla_\theta L_k = \nabla_\theta E_{h_1,\ldots,h_k\sim q(h|x,\theta)}\left[\log\frac{1}{k}\sum_i \frac{p(x,h_i|\theta)}{q(h_i|x,\theta)}\right] = \nabla_\theta E_{\epsilon_1,\ldots,\epsilon_k\sim N(0,I)}\left[\log\frac{1}{k}\sum_i \frac{p(x,h(x,\epsilon_i,\theta)|\theta)}{q(h(x,\epsilon_i,\theta)|x,\theta)}\right]$

    $= E_{\epsilon_1,\ldots,\epsilon_k\sim N(0,I)}\left[\nabla_\theta \log\frac{1}{k}\sum_i \frac{p(x,h(x,\epsilon_i,\theta)|\theta)}{q(h(x,\epsilon_i,\theta)|x,\theta)}\right]$

    $= E_{\epsilon_1,\ldots,\epsilon_k\sim N(0,I)}\left[\sum_i \frac{w_i}{\sum_{i'} w_{i'}}\nabla_\theta \log\frac{p(x,h(x,\epsilon_i,\theta)|\theta)}{q(h(x,\epsilon_i,\theta)|x,\theta)}\right]\ where\ \ w_i = \frac{p(x,h(x,\epsilon_i,\theta)|\theta)}{q(h(x,\epsilon_i,\theta)|x,\theta)} : importance\ weight$

    $\approx \sum_i \frac{w_i}{\sum_{i'} w_{i'}}\nabla_\theta \log\frac{p(x,h(x,\epsilon_i,\theta)|\theta)}{q(h(x,\epsilon_i,\theta)|x,\theta)} (= \nabla_\theta L(x)\ \ when\ k = 1)$

Variational Canonical Component Analysis

- Capture common sources of variation
- CCA : project X, Y in low-dimensional subspace to maximize correlation
- DCCA : non-linear extension of CCA

$$\max_{\mathrm{W_f, W_g, U, V}} \; tr(U^T f(X) g(Y)^T V)$$
$$s.t. \;\; U^T(f(X)f(X)^T)U = V^T(g(Y)g(Y)^T)V = NI$$

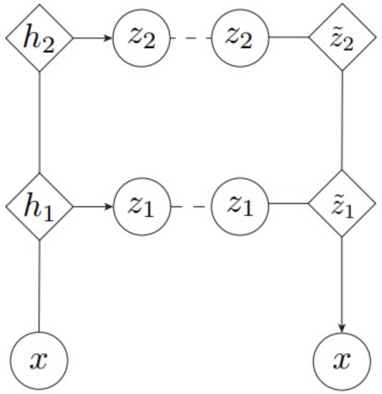
- VCCA : variational extension of CCA
  1. $p(x,y,z) = p(z)p(x|z)p(y|z)$
  - $\log p_\theta(x,y) \geq E_{q_\phi(z|x(,y))}[\log p_\theta(x|z) + \log p_\theta(y|z)] - KL(q_\phi(z|x(,y))\|p(z))$
  2. $p(x,y,z,h_x,h_y) = p(z)p(h_x)p(h_y)p_\theta(x|z,h_x)p_\theta(y|z,h_y)$
  - $\log p_\theta(x,y) \geq E_{q_\phi(z|x),q_\phi(h_x|x)}[\log p_\theta(x|z,h_x)] + E_{q_\phi(z|x),q_\phi(h_y|y)}[\log p_\theta(y|z,h_y)]$
    $-KL(q_\phi(z|x)\|p(z)) - KL(q_\phi(h_x|x)\|p(h_x)) - KL(q_\phi(h_y|y)\|p(h_y))$

ELBO surgery

- Rewrite ELBO by decomposing KL term to highlight the role of the encoded data distribution
  - $q(x, z) = q(x)q(z|x) = \frac{1}{N} q(z|x) \ \rightarrow q(z) = \frac{1}{N} \sum_{n=1}^{N} q(z|x_n)$
  - $p(x, z) = p(x)p(z|x) = \frac{1}{N} p(z)$
  - $$\begin{aligned} E_{p(x)}[KL(q(z|x) \| p(z))] &= KL(q(z) \| p(z)) + E_{q(z)}[KL(q(x|z) \| p(x)] \\ &= KL(q(z) \| p(z)) + I_{q(x,z)}(x, z) \\ &= KL(q(z) \| p(z)) + \log N - E_{q(z)}[H(q(x|z)] \end{aligned}$$
- Mutual information term is near its maximum value
  - No significant overlap between the individual encoding distribution $q(z|x_n)$
- Small marginal KL term was observed in small ELBO
  - Rigid prior might be used where encoder and decoder are unable to match
  - Multimodal prior is suggested

Variational Ladder Autoencoder

- HVAE : $p(x,z) = p(x|z_1)\prod_{l=1}^{L-1} p(z_l|z_{l+1})p(z_L)$
- Provide a deeper understanding of the design and performance of hierarchical LVM
  - Limitations
    - $x \sim p(x|z_1)\, where\, z_1 \sim q(z_1|x)$ is enough to converge to $\mathrm{p_{data}}(x)$
      (Redundancy of $p(z_l|z_{l+1})\, for\, 1 \le l < L$)
    - $q(z_l|z_{l+1})\, and\, p(z_l|z_{l+1})$ is encouraged to match to be parameterized gaussians
      (Limit the hierarchical relationship between features)
- Use neural network of different level of expressiveness to generate each feature
- More abstract features are constructed by deeper network
  - $z_l \sim N\big(\mu_l(h_l),\ \sigma_l(h_l)\big)\ where\ h_l = g_l(h_{l-1})\ for\ l = 1, \ldots, L\ and\ h_0 = x$
  - $x \sim r\big(x; f_0(\tilde{z}_1)\big)\ where\ \tilde{z}_l = f_l(\tilde{z}_{l+1}, z_l)\ for\ l = 1, \ldots, L-1\ and\ \tilde{z}_L = f_L(z_L)$
  - $L(x) = E_{q(z|x)}[\log p(x|z)] - KL(q(z|x)\|p(z))$



Deep Variational Information Bottleneck

- Learn z that is maximally compressive and expressive about x and y, respectively
  - minimal sufficient statistics of x for predicting y
  - $\max_{\theta} I(z, y; \theta) \;\; s.t. \;\; I(x, z; \theta) \leq I_c \;\; \Leftrightarrow \;\; I(z, y; \theta) - \beta \cdot I(z, x; \theta)$
- Construct the lower bound on the information bottleneck objective
  - $p(x, y, z) = p(x)p(y|x)p(z|x) = \frac{1}{N}\sum_{n=1}^{N} \delta_{x_n}(x)\delta_{y_n}(y)\, N\left(z \middle| f_e^{u}(x), f_e^{\Sigma}(x)\right)$
  - $q(y|z) = S(y|f_d(z)) \;\; where \;\; S(a) = \left[\frac{\exp(a_c)}{\sum_{c'} \exp(a_{c'})}\right]$
  - $r(z) = N(z|0, I)$
  - $L(x, y) = \int p(x)p(y|x)p(z|x)\left(\log q(y|z) - \beta \log \frac{p(z|x)}{r(z)}\right) dxdydz$

    $= \mathrm{E}_{p(x)p(y|x)}[E_{p(z|x)}[\log q(y|z)] - \beta \cdot KL(p(z|x) \| r(z))]$

InfoVAE

- Point out the problems in VAE objective that degrades the inference quality

$$-KL(q(x,z)\|p(x,z)) = -KL(p_{data}(x)\|p_\theta(x)) - E_{p_{data}(x)}\left[KL\left(q_\phi(z|x)\|p_\theta(z|x)\right)\right]$$
$$= -KL\left(q_\phi(z)\|p(z)\right) - E_{q_\phi(z)}[KL(q_\phi(x|z)\|p_\theta(x|z)]$$

  - Amortized Inference failures
    - ELBO can be maximized even with inaccurate variational posterior
    - Error in X is more critical than in Z due to high dimensionality → overfitting
  - Information preference property
    - Complex decoder improves sample quality while neglecting the latent variable
- Introduce a new training objective to weight the preference b/t inference quality and likelihood maximization

$$-\lambda \cdot KL\left(q_\phi(z)\|p(z)\right) - E_{q_\phi(z)}\left[KL\left(q_\phi(x|z)\|p_\theta(x|z)\right]\right] + \alpha \cdot I_q(x,z)$$
$$\approx -(\alpha+\lambda-1)\cdot D\left(q_\phi(z)\|p(z)\right) + E_{p_{data}(x)q_\phi(z|x)}[\log p_\theta(x|z)]$$
$$-(1-\alpha)\cdot E_{p_{data}(x)}\left[D\left(q_\phi(z|x)\|p(z)\right)\right]$$

  - Set λ so that loss from x equals loss from z
  - Set $\alpha = 0$ for simple decoder and $\alpha = 1$ for complex decoder
  - Any strict divergence is okay s.t. $D\left(q_\phi(z)\|p(z)\right) = 0 \ iff \ \ q_\phi(z) = p(z)$

    ex) MMD or Jenson Shannon divergence

Fixing a broken ELBO

- Derive the variational bounds on the mutual information b/t x and z
  - $H - D \leq I_q(x,z) \leq R$
  - Data entropy : $H = -\int p_{data}(x) \log p_{data}(x)\, dx$
  - Distortion : $D = -\int p_{data}(x) \int q_\phi(z|x) \log p_\theta(x|z)\, dz\, dx$
  - Rate : $R = \int p_{data}(x) \int q_\phi(z|x) \log \frac{\mathrm{q}_\phi(\mathrm{Z|X})}{\mathrm{r}_\psi(z)}\, dz\, dx$
- Derive (convex) RD curve explaining the trade-off b/t compression and reconstruction
  1. Auto-encoding limit : $R = H, D = 0$ → extreme reconstruction
  2. Auto-decoding limit : $R = 0, D = H$ → extreme compression
  - When $\mathrm{R} = H - D$, $r_\psi(z) = \int q_\phi(z|x) p_{data}(x)\, dx = q_\phi(z)$ and $p_\theta(x|z) = \frac{q_\phi(z|x) p_{data}(x)}{q_\phi(z)}$

    (but, with only finite parametric families, the bound would not be tight)
- Constrained optimization : minimize D while fixing R
  - $\min_{\theta,\phi,\psi} D + |\sigma - R|\ where\ \sigma\ is\ the\ target\ rate$
  - all current approaches are having hard time to achieve low D at high R
  - Need to develop better approximation $r_\phi(z)$ on marginal posterior $q_\phi(z)$

Mutual autoencoder

- Forces information flow by achieving the user specified mutual information
  1. $\max_{\theta} E_{p_{data}(x)}[\log \int p_\theta(x|z)p(z)\, dz] \;\; s.t. \;\; I_{p_\theta}(x,z) = M$
  2. $I_{p_\theta}(x,z) \geq \hat{I}_{p_\theta}(x,z) = H_{p_\theta}(z) + \max_{w} E_{p_\theta}[\log r_w(z|x)]$
  - $E_{p_{data}(x)q_\phi(z|x)}\left[\log p_\phi(x|z)\right] - E_{p_{data}(x)}\left[KL\left(q_\phi(z|x) \| p(z)\right)\right]$
    $-C \left| H_{p_\theta}(z) + \max_{w} E_{p_\theta(x|z)p(z)}[\log r_w(z|x)] - M \right|$

Algorithm 1 Mutual Autoencoder Training

```
procedure TRAINMAE(θ, ω, B, C, M, N)
    for i = 1, ..., N do
        UPDATEMODEL(θ, ω, B, C, M)            // We simultaneously optimize the model...
        UPDATEMIESTIMATE(ω, θ, B)             // ...and the mutual information estimate.
    end for
end procedure

procedure UPDATEMIESTIMATE(ω, θ, B)
    Sample (z_i, x_i) ~ p_θ for i = 1, ..., B
    g ← (1/B) Σ_{i=1}^{B} ∇_ω log r_ω(z_i|x_i)   // Gradient estimate of the infomax bound.
    ω ← Update(ω, g)
end procedure

procedure UPDATEMODEL(θ, ω, B, C, M)
    g_ELBO ← EstimateElboGradient(θ)
    g_MI ← Estimate of ∇_θ E_{(x,z)~p_θ}[log r_ω(z|x)]   // Using reparametrization trick or REINFORCE.
    Sample (z_i, x_i) ~ p_θ for i = 1, ..., B
    m ← H_p(z) + (1/B) Σ_{i=1}^{B} log r_ω(z_i|x_i)     // Mutual information estimate.
    θ ← Update(θ, g_ELBO − C · sign(m − M) · g_MI)
end procedure
```

Auto-encoding total correlation explanation

- Derive variational lower bound to total Cor-relaton Ex-planation (CorEx)
  - Total correlation captures the dependence across all the dimensions
    - $TC(x) = \sum_{i=1}^{d} H(x_i) - H(x) = KL\left(p(x) \middle\| \prod_{i=1}^{d} p(x_i)\right)$
    - $TC_\theta(x|z) = \sum_{i=1}^{d} H_\theta(x_i|z) - H_\theta(x|z) = KL\left(p_\theta(x|z) \middle\| \prod_{i=1}^{d} p_\theta(x_i|z)\right)$
    - $TC_\theta(x, z) = TC(x) - TC_\theta(x|z)$: amount of correlation explained by z
  - CorEx
    - $$\begin{aligned} TC_\theta(x, z) - TC_\theta(z) &= TC(x) - TC_\theta(x|z) - TC_\theta(z) \\ &= \sum_{i=1}^{d} I_{p_\theta}(x_i, z) - I_{p_\theta}(x, z) - \sum_{i=1}^{m} H_\theta(z_i) + H_\theta(z) \\ &= \sum_{i=1}^{d} I_{p_\theta}(x_i, z) - \sum_{i=1}^{m} H_\theta(z_i) + H_\theta(z|x) \\ &\approx \sum_{i=1}^{d} I_{p_\theta}(x_i, z) - \sum_{i=1}^{m} I_{p_\theta}(z_i, x) \end{aligned}$$
    - AnchorVAE : $TC_\theta(x, z) - TC_\theta(z) + \lambda \cdot I_\theta(z_k, x)$ → concentrate the explanatory power to particular latent variable
    - Maximizes when $p_\theta(x|z) = \prod_{i=1}^{d} p_\theta(x_i|z)$ s.t. $x_i's$ are factorized conditioned on z
    - Maximizes when $p_\theta(z) = \prod_{i=1}^{m} p_\theta(z_i)$ s.t. $z_i's$ are independent
    - Last equality holds when $p_\theta(z|x) = \prod_{i=1}^{m} p_\theta(z_i|x)$ (as usual)
  - Variational Lower bound
    - $L(x) = E_{p(x)p_\theta(z|x)}\left[\sum_{i=1}^{d} \log q_\phi(x_i|z)\right] - E_{p(x)}\left[\sum_{i=1}^{m} KL\left(p_\theta(z_i|x) \| r_\psi(z_i)\right)\right]$

    (factorized encoder and decoder of VAE)
  - Stacking layers for hierarchical structure
    - $TC(x) - \sum_{l=1}^{L} TC_\theta\left(z^{(l-1)} \middle| z^{(l)}\right) - TC_\theta\left(z^{(L)}\right)\ where\ \ z^{(0)} = x$
    - $$\begin{aligned} L(X) = {} & E_{p(x)p_\theta(z|x)}\left[\sum_{l=1}^{L} \sum_{i=1}^{m^{(l-1)}} \log q_\phi\left(z^{(l-1)} \middle| z^{(l)}\right)\right] - E_{p(x)}\left[\sum_{l=1}^{L-1} \sum_{i=1}^{m^{(l)}} KL(\, p_\theta\left(z_i^{(l)} \middle| z_i^{(l-1)}\right) \middle\| q_\phi\left(z_i^{(l)} \middle| z_i^{(l+1)}\right)\right] \\ & - E_{p(x)}\left[\sum_{i=1}^{m^{(L)}} KL(\, p_\theta\left(z_i^{(L)} \middle| z_i^{(L-1)}\right) \middle\| r_\psi(z_i^{(L)})\right] \end{aligned}$$