Combining deep generative and discriminative models for semi-supervised learning

December 10, 2024

Introduction

Supervised learning

Dataset $D = \{x_n, y_n\}_{n=1}^{N}$, neural network models conditional distribution $p(y|x_n)$ with a parameter θ, and optimizes likelihood with respect to θ.
Drawbacks: tend to overfit the data, produce highly-confidence predictions, require massive labelled data for training.

Introduction

Supervised learning

Dataset $D = \{x_n, y_n\}_{n=1}^N$, neural network models conditional distribution $p(y|x_n)$ with a parameter θ, and optimizes likelihood with respect to θ.
Drawbacks: tend to overfit the data, produce highly-confidence predictions, require massive labelled data for training.

DGMs for semi-supervised learning

VAEs introduce latent variables z and use a neural network with parameters θ_g to model $p(x|z)$. Inference networks with ϕ_z are introduced: $q_{\phi_z}(z|x) \sim p(z|x, \theta_g)$.

Inference networks

I

nferences networks are introduced as $q_\phi(y, z|x) = q_{\phi_y}(y|x)q_{\phi_z}(z|x, y)$ with parametes $\phi = \{\phi_y, \phi_z\}$. These are used to integrate out the latent or unobserved variables z and y.

Inference networks

I

nference networks are introduced as $q_\phi(y, z|x) = q_{\phi_y}(y|x)q_{\phi_z}(z|x, y)$ with parametes $\phi = \{\phi_y, \phi_z\}$. These are used to integrate out the latent or unobserved variables z and y.

Loss

$$\mathcal{L}(\theta_g, \phi, x, y) = \sum_{(x_l, y_l)\sim p_l} \mathcal{L}^l(\theta_g, \phi, x_l, y_l) + \sum_{x_u \sim p_u} \mathcal{L}^u(\theta_g, \phi, x_u)$$

Where p_l and p_u stands for labelled and unlabelled data. These losses are calculated with ELBO estimation

Graphical model and M2

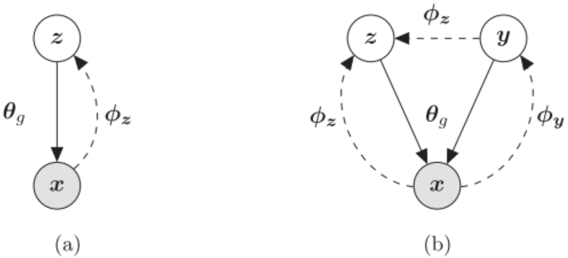


Figure: Continuous edges denote conditional probabilities distributions while discontinuous denotes inference networks.

Proposed model

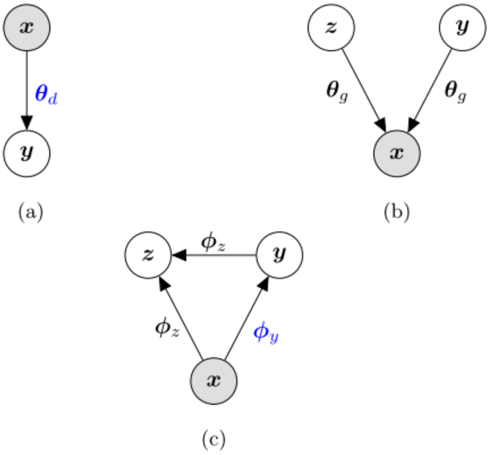


Figure: (a) discriminative component, (b) M2 generative component and (c) inference network. Blue parameters are tied for joint training.

Idea

Likelihood

Our framework seeks to combine deep generative and discriminative models. We jointly train two models

$$\log p(x_l, y_l, x_u, \theta_d, \theta_g) = \log p(\theta_d, \theta_g) + \log p(y_l|x_l, \theta_d) + \log p(x_l|\theta_g) + \log p(x_u|\theta_g)$$

where x_u is independently generated unlabelled point and conditional labelled probabilities are computed with the deep neural network parametrization.

Idea

Likelihood

Our framework seeks to combine deep generative and discriminative models. We jointly train two models

$$\log p(x_l, y_l, x_u, \theta_d, \theta_g) = \log p(\theta_d, \theta_g) + \log p(y_l|x_l, \theta_d) + \log p(x_l|\theta_g) + \log p(x_u|\theta_g)$$

where x_u is independently generated unlabelled point and conditional labelled probabilities are computed with the deep neural network parametrization.

Evidence lower bound

$$\mathcal{L} = \log p(\theta_d, \theta_g) + \log p_{\theta_d}(y_l|x_l) + \mathbb{E}_{q_\phi(y,z|x_l,y_l)}\left[\log \frac{p_{\theta_g}(x_l, y, z)}{q_{\phi_z}(z|x_l, y_l)}\right] + \mathbb{E}_{q_\phi(z,z|x_u)}\left[\log \frac{p_{\theta_g}(x_u, y, z)}{q_{\phi_z}(z, y|x_u)}\right]$$

Expectations are approximated with Monte-Carlo method.

Prior

Prior for $p(\theta_d, \theta_g)$

$$p(\theta_d, \theta_g \mid \phi_y) = \mathcal{N}\left([\phi_y, 0]^T, \begin{bmatrix} \lambda_d^{-1} I & 0 \\ 0 & \lambda_g^{-1} I \end{bmatrix}\right)$$

Using this prior we can further relate λ_d^{-1} to the notions of interpolating between the generative and discriminative cases and give more interpretability to this hyper-parameter.

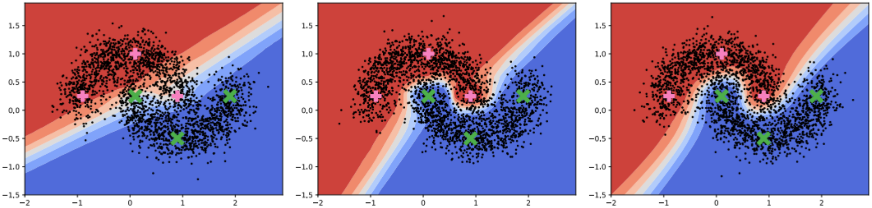
Approximate inference for the discriminative component parameters

Bayesian approach

We can explicitly account for predictive uncertainty via Bayesian inference on θ_d.

$$\begin{aligned}\mathcal{L}_{\text{post}}(\boldsymbol{\epsilon}, \boldsymbol{\varphi}, \boldsymbol{\theta}_{\text{g}}, \boldsymbol{\phi}; \boldsymbol{x}_{\text{l}}, \boldsymbol{x}_{\text{y}}, \boldsymbol{x}_{\text{u}}) = & \, \mathbb{E}_{q_{\varphi}(\boldsymbol{\theta}_d|\boldsymbol{\phi}_y)}\left[\log p_{\boldsymbol{\theta}_d}(\boldsymbol{y}_l|\boldsymbol{x}_l)\right] \\ & + \mathbb{E}_{q_{\phi}(\boldsymbol{y},\boldsymbol{z}|\boldsymbol{x}_l,\boldsymbol{y}_l)}\left[\log \frac{p_{\boldsymbol{\theta}_g}(\boldsymbol{x}_l, \boldsymbol{y}, \boldsymbol{z})}{q_{\boldsymbol{\phi}_z}(\boldsymbol{z}|\boldsymbol{x}_l, \boldsymbol{y})}\right] \\ & + \mathbb{E}_{q_{\phi}(\boldsymbol{z},\boldsymbol{y}|\boldsymbol{x}_u)}\left[\log \frac{p_{\boldsymbol{\theta}_g}(\boldsymbol{x}_u, \boldsymbol{y}, \boldsymbol{z})}{\log q_{\boldsymbol{\phi}}(\boldsymbol{z}, \boldsymbol{y}|\boldsymbol{x}_u)}\right] \\ & - D_{\text{KL}}\left(q_{\varphi}(\boldsymbol{\theta}_d|\boldsymbol{\phi}_y) \,\|\, p(\boldsymbol{\theta}_d|\boldsymbol{\phi}_y)\right). \qquad (12)\end{aligned}$$

Toy data



(a) M2 (b) Blended (c) Bayes-blended

Figure: Two-class moons

Real data

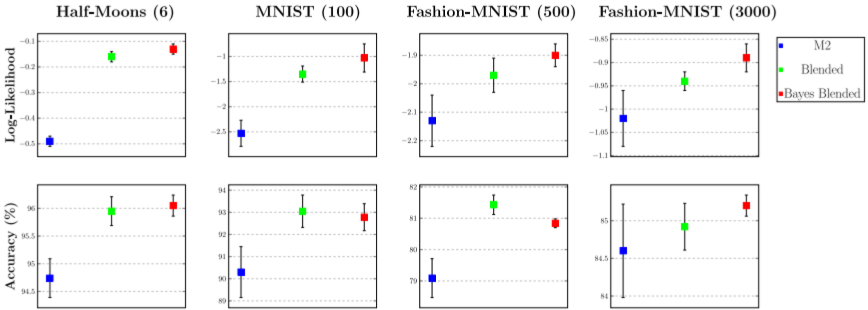


Fig. 6. (Top) Log-likelihood and (bottom) accuracy results for different models and dataset. Number of labelled examples made available to model is in parenthesis.

Figure: Likelihood and accuracy