Semi-supervised Learning with Deep Generative Models

March 20, 2018

1. Introduction

2. Latent-feature discriminative model (M1)

3. Generative semi-supervised model (M2)

4. Stacked generative semi-supervised model (M1+M2)

5. Results

Background

- Paper: Semi-supervised Learning with Deep Generative Models
- **Semi-supervised** learning considers the problem of classification when only a small subset of the observations have corresponding class labels.
- Main contributions:
  1. New framework for semi-supervised learning with generative models
  2. First time bring variational inference to bear upon the problem of semi-supervised classification
  3. Better performance on several benchmark problems
  4. Generative semi-supervised models learn to separate the data classes (content types) from the intra-class variabilities (styles)

Latent-feature discriminative model (M1)

- We construct a **deep generative model** that provides a **robust** set of **latent features** representation of the data.
- Generative Model:

$$p(z) = N(z|0, I) \tag{1}$$

$$p_\theta(x|z) = f(x; z, \theta) \tag{2}$$

- Here, $f(x; z, \theta)$ is a suitable likelihood function (e.g. a Gaussian or Bernoulli distribution), whose probabilities are formed by a non-linear transformation (deep neural networks), with parameters θ, of a set of latent variables z.

Training

- We construct the approximate posterior distribution q_ϕ as an inference model:

$$q_\phi(z|x) = N(z|\mu_\phi(x), diag(\sigma^2_\phi(x)))$$

  Here, μ_ϕ and σ_ϕ are MLPs.

- Then, we optimize the ELBO:

$$logp_\theta(x) \geq \mathbb{E}_{q_\phi(z|x)}[logp_\theta(x|z)] - KL[q_\phi(z|x)||p_\theta(z)] = -J(x) \tag{3}$$

- Optimization Recap: For the first expectation term, we will use reparametriation + monte carlo samples. For KL, since they are both gaussian distributions, it's analytic. Now, we use the estimated gradients in conjunction with standard stochastic gradient-based optimization methods such as SGD or AdaGrad.

Application

- We will use $q_\phi(z|x)$ to extract features used for training a classifier such as (transductive) SVM or multinomial regression.
- Using this approach, we can now perform classification in a lower dimensional space since we typically use latent variables whose dimensionality is much less than that of the observations.
- This simple approach results in improved performace for SVMs.

Generative semi-supervised model (M2)

- We propose a probabilistic model that describes the data as being geneearted by a latent class variable y in addition to a continuous latent variable z.
- The data is explained by the generative process:

$$
\begin{aligned}
p(y) &= Cat(y|\pi) \\
p(z) &= N(z|0, I) \\
p_\theta(x|y, z) &= f(x; y, z, \theta)
\end{aligned}
$$

Here, $Cat(y, \pi)$ is the multinomial distribution. The class labels y are treated as latent variables if no class label is available and z are additional latent variables.

Generative semi-supervised model (M2) - continued

- The data is explained by the generative process:

$$p(y) = Cat(y|\pi)$$
$$p(z) = N(z|0, I)$$
$$p_\theta(x|y, z) = f(x; y, z, \theta)$$

- Notably, these latent variables are marginally independent, and allow us, in case of digit generation for example, to separate the class specification from the writing style of the digit.
- $f(x; y, z, \theta)$, like before, is a suitable likelihood function.

Training

- We approximate posteriors using inference networks:

$$q_\phi(z, y|x) = q_\phi(z|y, x)q_\phi(y|x) \tag{4}$$
$$q_\phi(z|y, x) = N(z|\mu_\phi(y, x), diag(\sigma^2_\phi(x))) \tag{5}$$
$$q_\phi(y|x) = Cat(y|\pi_\phi(x)) \tag{6}$$
$$\tag{7}$$

- For labeled data, we optimize the ELBO for $p_\theta(x, y)$:

$$logp_\theta(x, y) \geq \mathbb{E}_{q_\phi(z|x,y)}[logp_\theta(x|y, z) \tag{8}$$
$$+ logp_\theta(y) + logp(z) - logq_\phi(z|x, y)] \tag{9}$$
$$= -L(x, y) \tag{10}$$

Training

- For unlabeled data, we treat label as a latent variable, and optimize the ELBO for $p_\theta(x)$:

$$
\begin{aligned}
logp_\theta(x) &\geq \mathbb{E}_{q_\phi(y,z|x)}[logp_\theta(x|y,z) && (11)\\
&+ logp_\theta(y) + logp(z) - logq_\phi(y,z|x)] && (12)\\
&= \sum_y q_\phi(y|x)(-L(x,y)) + H(q_\phi(y|x)) && (13)\\
&= -U(x) && (14)
\end{aligned}
$$

- Thus, the bound on the marginal likelihood for the entire dataset is now:

$$
J = \sum_{(x,y)\sim\tilde{p_l}} L(x,y) + \sum_{x\sim\tilde{p_u}} U(x) \qquad (15)
$$

Application

- We use $q_\phi(y|x)$ at test time for predictions (classifier).
- In the objective function, the label predictive distribution $q_\phi(y|x)$ contributes only to the second term relating to the unlabelled data, which is an undesirable proprety if we wish to use this distribution as a classifier.
- To remedy this, we add a classification loss, such that the distribution $q_\phi(y|x)$ also learns from labelled data. The extended objective funciton is:

$$J^\alpha = J + \alpha \cdot \mathbb{E}_{\tilde{p}_l(x,y)}[-log q_\phi(y|x)] \tag{16}$$

Here, the hyper-parameter α controls the relative weight between generative and purely discriminative learning.

Stacked generative semi-supervised model (M1+M2)

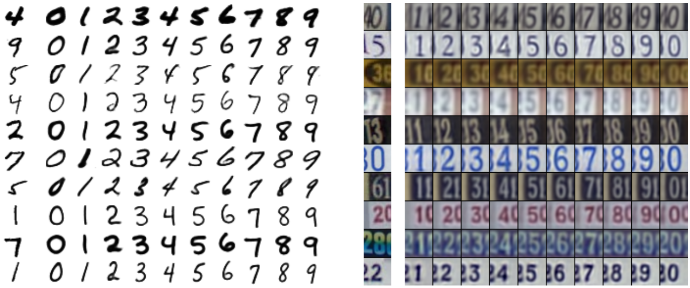
- We combine previous two models: we first learn a new latent representation z_1 with latent variables z_2 using the generative model from M1, and subsequently learn a generative semi-supervised model M2, using embeddings from z_1 instead of the raw data x.
- We can represent it as:

$$p_\theta(x, y, z_1, z_2) = p(y)p(z_2)p_\theta(z_1|y, z_2)p_\theta(x|z_1)$$

Table 1: Benchmark results of semi-supervised classification on MNIST with few labels.

N	NN	CNN	TSVM	CAE	MTC	AtlasRBF	M1+TSVM	M2	M1+M2
100	25.81	22.98	16.81	13.47	12.03	8.10 ($\pm$ 0.95)	11.82 ($\pm$ 0.25)	11.97 ($\pm$ 1.71)	**3.33** ($\pm$ 0.14)
600	11.44	7.68	6.16	6.3	5.13	–	5.72 ($\pm$ 0.049)	4.94 ($\pm$ 0.13)	**2.59** ($\pm$ 0.05)
1000	10.7	6.45	5.38	4.77	3.64	3.68 ($\pm$ 0.12)	4.24 ($\pm$ 0.07)	3.60 ($\pm$ 0.56)	**2.40** ($\pm$ 0.02)
3000	6.04	3.35	3.45	3.22	2.57	–	3.49 ($\pm$ 0.04)	3.92 ($\pm$ 0.63)	**2.18** ($\pm$ 0.04)

(a) Handwriting styles for MNIST obtained by fixing the class label and varying the 2D latent variable $\mathbf{z}$



(b) MNIST analogies (c) SVHN analogies

Figure 1: **(a)** Visualisation of handwriting styles learned by the model with 2D $\mathbf{z}$-space. **(b,c)** Analogical reasoning with generative semi-supervised models using a high-dimensional $\mathbf{z}$-space. The leftmost columns show images from the test set. The other columns show analogical fantasies of $\mathbf{x}$ by the generative model, where the latent variable $\mathbf{z}$ of each row is set to the value inferred from the test-set image on the left by the inference network. Each column corresponds to a class label $\mathbf{y}$.