BAYESIAN MATRIX FACTORIZATION AND RECOMMENDER SYSTEMS

A PREPRINT

Aniket Das (Mentor)
Department of Electrical Engineering
IIT Kanpur
aniketd@iitk.ac.in

Yatin Dandi(Mentor)
Department of Computer Science and Engg.
IIT Kanpur
yatind@iitk.ac.in

Naman Biyani
Department of Computer Science and Engg
IIT Kanpur
namanb@iitk.ac.in

Avinandan Bose
Department of Computer Science and Engg
IIT Kanpur
avibose@iitk.ac.in

Spandan Senapati
Department of Computer Science and Engg.
IIT Kanpur
spandas@iitk.ac.in

July 9, 2019

ABSTRACT 1

Low-rank matrix approximation methods provide one of the simplest and most effective approaches to collaborative filtering.Such models are usually fitted to data by finding a MAP estimate of the model parameters, a procedure that can be performed efficiently even on very large datasets.In this project we implemented a fully Bayesian treatment of the ProbabilisticMatrix Factorization (PMF) model in which model capacity is controlled automatically by integrating over all model parameters and hyperparameter.The model was trained by using Markov Chain Monte Carlo methods on Movielens100k dataset.

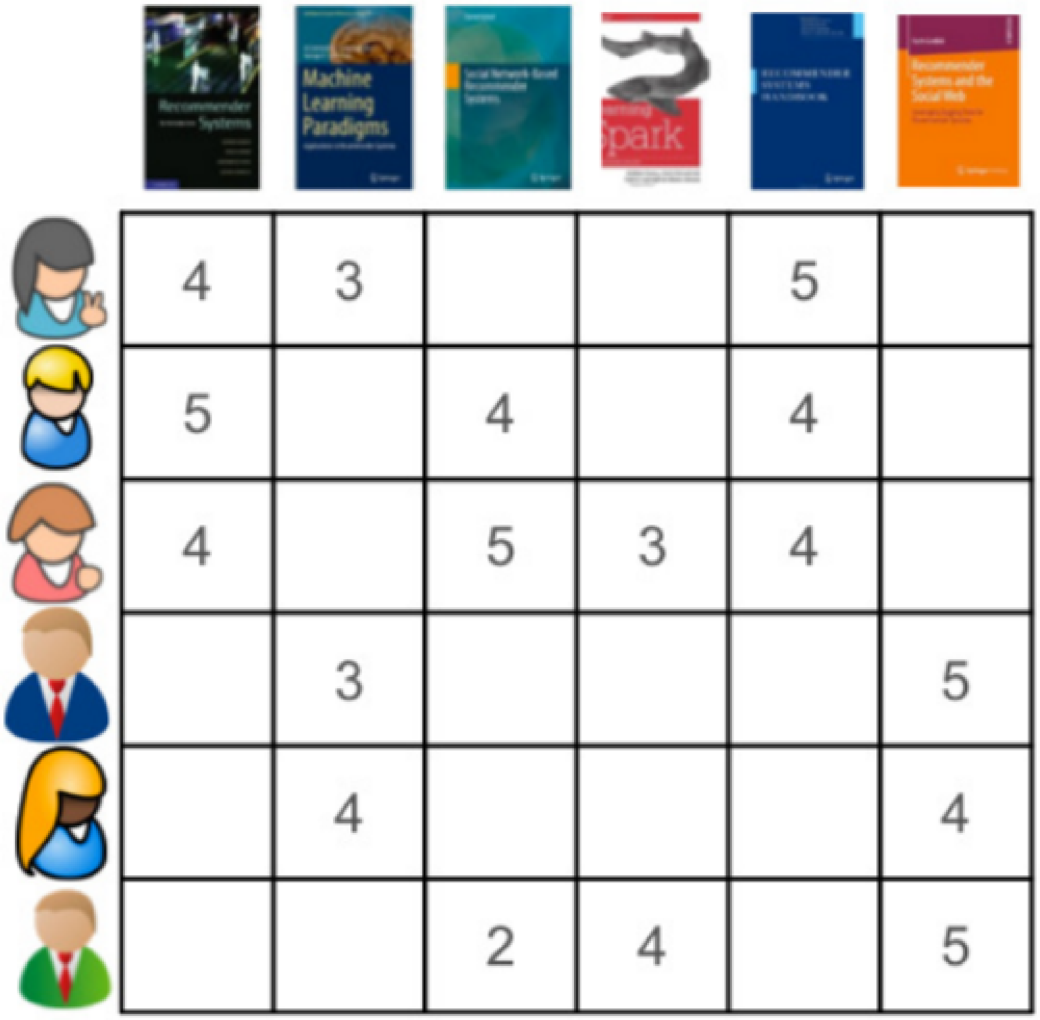
INTRODUCTION

Given data D = r_{ij} of "interactions"(e.g.,ratings) of users i = 1,...,N users on j = 1,...,M items, our goal is to predict the unobserved ratings $r_{ij} \notin D$.
Let **R** be the full rating matrix, we wish to fill.

R = **UV**^{T} + **E**
U = [u$_1$,...,u$_n$]^T is$N \times K$ and consists of the latent factors of the N users.
V = [v$_1$,...,v$_n$]^T is$M \times K$ and consists of the latent factors of the M items.
E consists of the noise in R that the low rank matrix assumption doesn't capture.

GIBBS SAMPLING

It is a general sampling algorithm to simulate samples from multivariate distributions.It samples one component at a time from its conditional, conditioned on all other components ,assuming that the conditional distributions are available in a closed form. The generated samples give a sample-based approximation of the multivariate distribution.


Recommender Systems
Machine Learning Paradigms
Social Network-Based Recommender Systems
Learning Spark
Recommender Systems Handbook
Recommender Systems and the Social Web

4	3			5	
5		4		4	
4		5	3	4	
	3				5
	4				4
		2	4		5

Figure 1: Illustrative scaled-down example of the input.

DESCRIPTION

We implemented two variants of the Gaussian BMF. We assume the noise to be Gaussian for each obervation.

We also assume Gaussian priors on the user and item latent factors. In the first variant we keep the hyperparams β, λ_u ,λ_v to be constant, whereas in the second variant, we allow Gaussian priors on these hyperparameters so that they can be learned.

Our target posterior is intractable, however the conditional posterior on each variable is tractable and thus we can use Gibb's sampling to estimate the posterior.

EXPERIMENTS

We received satisfactory results from our model.We obtained a final RMSE of 1.0252191074035872 for D = 60 and RMSE of 1.0177717637067052 for D = 100 on the second variant on the test R matrix using alpha = 2 and beta =1 as our hyperparameters . Performing hyperparameter tuning and then averaging the weights obtained from different hyperparameter values gave us greater accuracy upto 0.9 which is comparable with the state of the art values. (Where D represents the number of latent variables per user and per item).

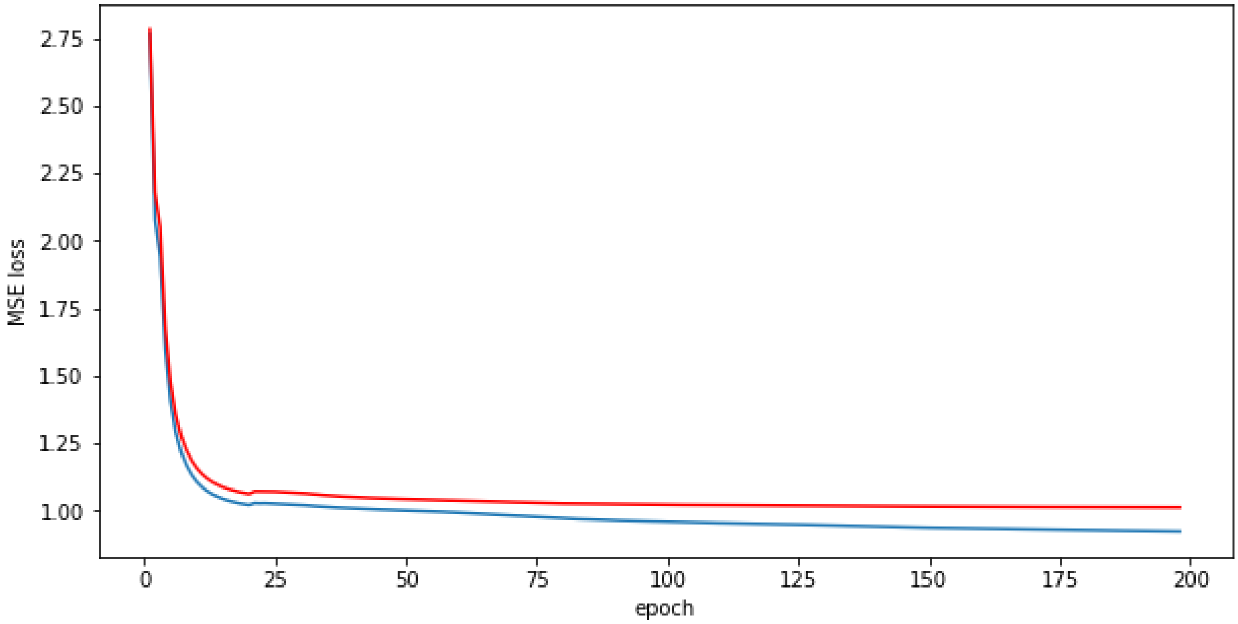


Figure 2: This graph plots the RMSE vs the number of epochs(for 60 latent variables) .The blue line represents the test error, while the red line indicates the train error.

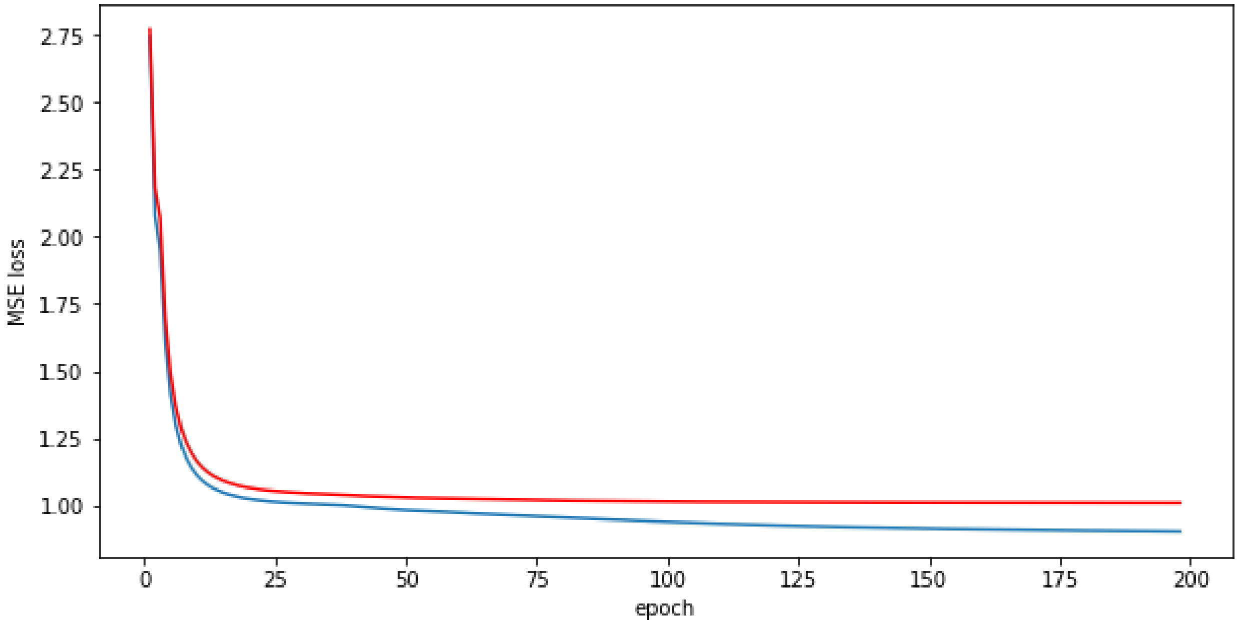


Figure 3: This graph plots the RMSE vs the number of epochs(for 100 latent variables) .The blue line represents the test error, while the red line indicates the train error.

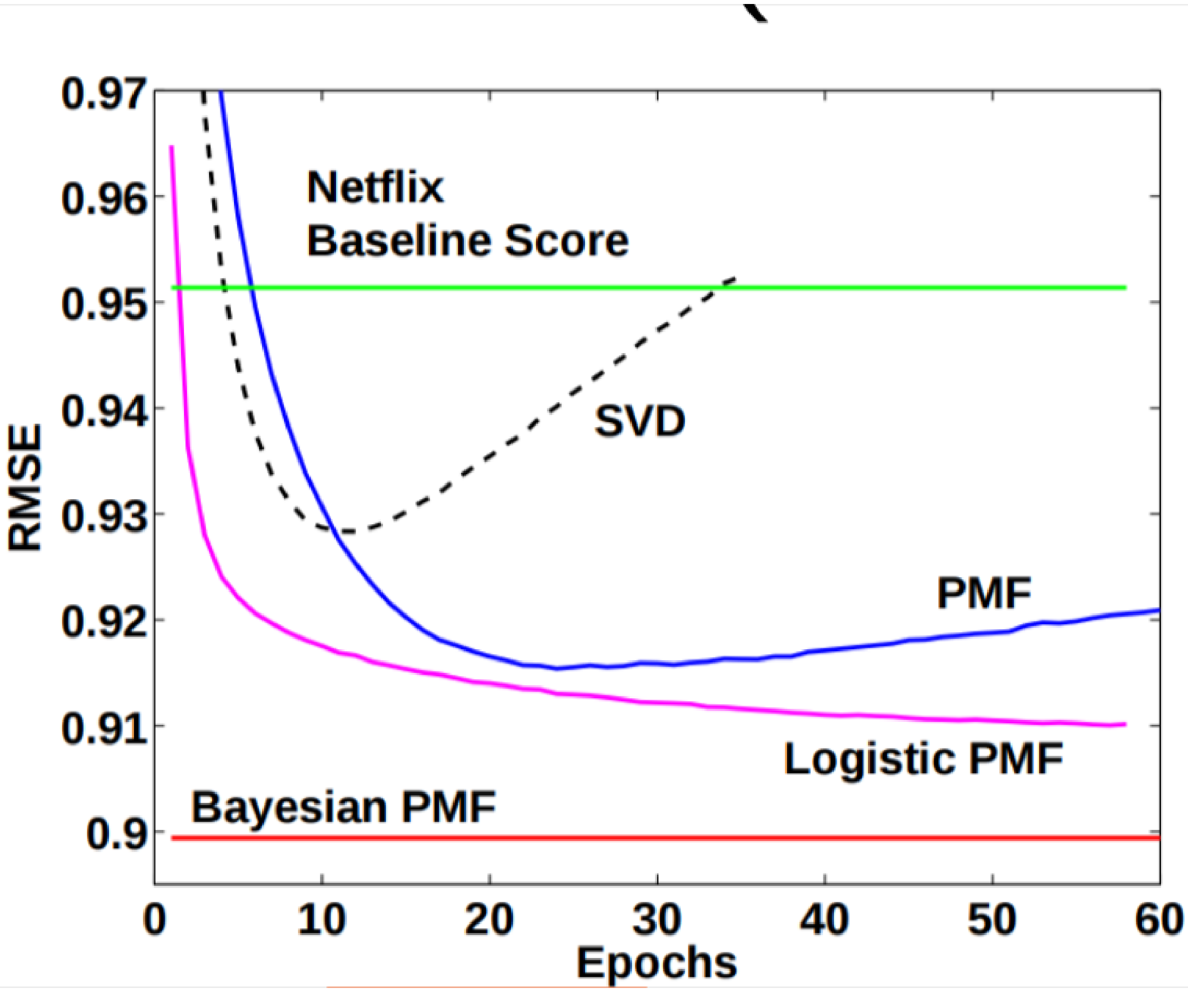


Figure 4: Comparison of results of differnet models for reccomender systems

ADVANCES IN VARIATIONAL INFERENCE

A PREPRINT

July 9, 2019

ABSTRACT

The focus of the project will be on an important aspect of approximate inference in probabilistic machine learning namely Variational Inference and the advances in the feild. The algorithm begins with a trivial assumption similar to what is done in statistical physics by the Mean Feild Theory assumption.Advances in Variational inference include Black Box VI,Stochastic VI,Armotixed VI,Autoencoding VI etc.

1 Introduction

1.1 Variational Inference

Variational Inference and Markov Chain Monte Carlo Stimulation(MCMC)form the two most commonly used means of approximate inference.In MCMC we construct a Markov Chain over the hidden variables whose stationary distribution is the posterior of interest.We run the chain until it has (hopefully) reached equilibrium and collect samples to approximate the poxterior.In Variational Inference, we define a flexible family of distributions over the hidden variables, indexed by free parameters.We then find the setting if the parameters that is closest to the posterior,by minimising the Kullback-Leibler Divergence.

1.2 Mean Field Theory

The simplest variational family of distributions to work with is the *Mean Field Variational Family*,wherein each hidden variable is independent and governed by its own parameter.In mathematical terms

$$\prod_{i=1}^{N} q(z_i) = q(z) \tag{1}$$

. Assuming variables are governed by parameters 1 simplifies to

$$q(z) = \prod_{i=1}^{N} \prod_{j=1}^{J} q(z_{nj}|\phi_{nj}) \tag{2}$$

2 Optimisation

2.1 Evidence Lower Bound(ELBO)

Variational Inference minimises the KL divergence from the variational distribution to the posterior distribution.It thereby maximised the *evidence lower bound*(ELBO),a lower bound on the marginal probability.This is easy to derive using Jensen's inequality $\log \mathbb{E}[f(y)] \geq \mathbb{E}[\log f(y)]$.This gives the bound on the log marginal as

$$\begin{aligned}
\log p(x) &= \log \int p(z,x)dz \\
&= \log \int p(z,x)\frac{q(z)}{q(z)}dz \\
&= \log \left(\mathop{\mathbb{E}}_{q(z)} \left[\frac{p(x.z)}{q(z)} \right] \right) \\
&= \mathop{\mathbb{E}}_{q(z)} \left[\log p(z,x) \right] - \mathop{\mathbb{E}}_{q(z)} \left[\log q(z) \right] \\
&\equiv \mathcal{L}(q)
\end{aligned}$$

Further it is easy to prove by some trivial manipulation and by introducing the KL divergence the relation between the ELBO and KL divergence as,

$$\log p(x) = \mathcal{L}(q) + \mathrm{KL}(q(z)||p(z|x)) \tag{3}$$

In cases where the posterior distribution becomes intracable we introduce the variational family and optimise the ELBO with by using a step wise gradien ascent by updating each member of the family.The update rules can be obtained by considering the form of the ELBO for a particular member j as follows

$$\begin{aligned}
\mathcal{L}(q) &= \int \prod_{i=1}^{N} \left\{ \ln p(\mathrm{X,Z}) - \sum_{i=1}^{N} \ln q_i \right\} d\mathrm{Z} \\
&= \int q_j \left\{ \int \ln p(\mathrm{X,Z}) \prod_{i \neq j} q_i d\mathrm{Z}_i \right\} d\mathrm{Z}_j - \int q_j \ln q_j d\mathrm{Z}_j + \mathrm{const} \\
&= \int q_j \ln \hat{p}(\mathrm{X,Z}_j) d\mathrm{Z}_j - \int q_j \ln q_j d\mathrm{Z}_j + \mathrm{const}
\end{aligned}$$

where we have defined a new distribution

$$\hat{p}(\mathrm{X,Z}_j) = \mathop{\mathbb{E}}_{i \neq j} \left[\ln p(\mathrm{X,Z}) \right]$$

The optimal solution is obtained in the usual way by minimising the KL Divergence.The new distribution of the family hence becomes,

$$q_j^* = \mathop{\mathbb{E}}_{i \neq j} \left[\ln p(\mathrm{X,Z}) \right] \tag{4}$$

This is repeated until the ELBO converges to obatain a distribution closest to the posterior.

3 Black Box Variational Inference

3.1 Principle

Black Box VI can work with small minibatches of data rather than the entire dataset thereby increasing the efficiency of the VI algorithm.It uses a Monte Carlo Stimulation in evaluating the gradients of the ELBO as ,

$$\nabla_\phi \mathcal{L}(q) \approx \frac{1}{S} \sum_{i=1}^{S} \nabla_\phi \log q(\mathrm{Z}_\phi)(\log p(\mathrm{X,Z}_s) - \log q(\mathrm{Z}_s|\phi)) \tag{5}$$

3.2 BBVI Identity

$$\nabla_\phi \mathcal{L}(q) = \mathbb{E}\left[\nabla_\phi \log q(\text{Z}|\phi)(\log p(\text{X,Z}) - \log q(\text{Z}|\phi))\right] \tag{6}$$

Since the required gradients are only a function of $q((Z|\phi))$ and not on the model it is called as Black Box VI.

3.3 Proof of the Identity

Using the Dominated Convergence Theorem,

$$\begin{aligned}
\nabla_\phi \mathcal{L}(q) &= \nabla_\phi \int \big[\log q(\text{Z}|\phi)(\log p(\text{X,Z}) - \log q(\text{Z}|\phi))\big] \\
&= \mathop{\mathbb{E}}_q\big[-\nabla_\phi \log q(\text{Z}|\phi)\big] + \int \nabla_\phi q(\text{Z}|\phi)\big[\log p(\text{X,Z}) - \log q(\text{Z}|\phi)\big] \\
&= \mathbb{E}\left[\nabla_\phi \log q(\text{Z}|\phi)(\log p(\text{X,Z}) - \log q(\text{Z}|\phi))\right] && \left(\because \mathbb{E}_q\left[\nabla_\phi \log q(\text{Z}|\phi)\right] = 0\right) \\
&\approx \frac{1}{S}\sum_{i=1}^{S} \nabla_\phi \log q(\text{Z}_\phi)(\log p(\text{X,Z}_s) - \log q(\text{Z}_s|\phi))
\end{aligned}$$

3.4 Inferences

1. BBVI allows VB inference for a wide variety of probabilistic models
2. Few Requirements
   - Should be able to sample from $q(\text{Z}|\phi)$
   - Should be able to compute the gradient $\nabla_\phi \log q(\text{Z}|\phi)$
   - Should be able to compute $p(\text{X,Z})$ and $\log q(\text{Z}|\phi)$
3. Reparametrization Trick to control Variance in the Monte Carlo estimate of gradient

4 Reparametrization Trick

1. LOTUS(Law of Unconscious Statistician) $\mathbb{E}(g(x)) = \int_{-\infty}^{+\infty} g(x) f_X(x)dx$

4.1 Example

Consider a Normal distribution given by $q \equiv \mathcal{N}(\theta, 1)$ and we wish to evaluate $\zeta = \nabla_\theta \mathbb{E}[x^2]$

$$\begin{aligned}
\zeta &= \int x^2 \nabla_\theta q_\theta(x)dx = \int x^2 q_\theta(x)\nabla_\theta \log q_\theta(x)dx \\
&= \mathbb{E}\left[x^2 \nabla_\theta \log q_\theta(x)\right] \\
&= \mathop{\mathbb{E}}_{q_\theta(x)}\left[x^2(x-\theta)\right] && \text{(Assume } x \approx \theta + \epsilon \text{ where } \epsilon \equiv \mathcal{N}\text{(0,1))} \\
&= \mathop{\mathbb{E}}_{p}\left(2\theta + 2\epsilon\right)
\end{aligned}$$

Now the two distributions differ in Variance.

4.2 Use in BBVI

Numerical estimation of ELBO's gradient.Under a transformation $\text{Z} = g(\epsilon, \phi)$ with $\epsilon = p(\epsilon)$ ELBO's gradient under Monte Carlo Stimulation can be written as

$$\nabla_\phi \mathop{\mathbb{E}}_{q_\phi(z)}(\log p(\text{X,Z}_s) - \log q(\text{Z}_s|\phi)) \approx \frac{1}{S}\sum_{s=1}^{S}\left[\nabla_\phi \log p(\text{X}, g(\epsilon_s, \phi)) - \nabla_\phi \log q_\phi(g(\epsilon_s, \phi))\right]$$

Such a gradient is called *Pathwise Gradient*

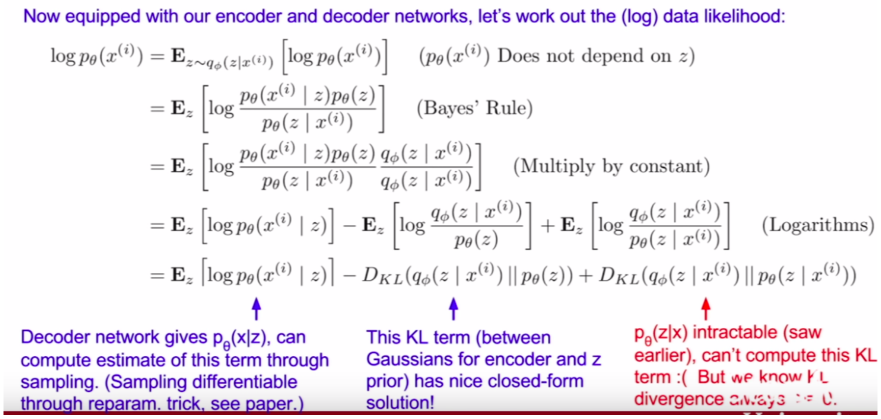
4.3 Inferences

Limitations

- Isn't often applicable e.g when Z is discrete or categorical
- Transformation function g may be difficult to find for general distributions
- Transformation function g needs to be invertible
- Assumption of direct sampling from $p(\epsilon)$

5 Variational Autoencoder

- In this section we'll give an example where we use a neural network for the probabilistic encoder q(z|x) (the approximation to the posterior of the generative model p(x, z)) and where the parameters and are optimized jointly with the AEVB algorithm.
- he unique property in VAEs which makes them so useful in maing generative models is that their latent space is continuous which allows random sampling .This happens because its encoder does not output an encoding vector of size n, rather, outputs two vectors of size n: a vector of means, , and another vector of standard deviations, . and now become the parameters for a vector of size n containing random values having mean and standard deviation as and respectively . This will result in random generation of data having mean and standard deviation close to the input data.
- Hence,the mean vector controls where the encoding of an input should be centered around, while the standard deviation controls the area, how much from the mean the encoding can vary. As encodings are generated at random from anywhere inside the shaded portion, the decoder learns all the nearby points that refer to sample of that class . This allows the decoder to not just decode specific encodings in the latent space but also the ones which vary thus making the latent space continuous . want our encodings to be as close as possible while still being distinct to make smooth interpolation, but sometimes we might end up making our latent space similar to the right picture shown above as there are no limits on what values vectors and can take on, so to avoid this , we introduce KL divergence in the loss function . The following image shows the calculation of the log of the data likelihood
- Although the third term in intractable, by Gibbs inequality , we know that KL divergence of two distributions is always non-negative . Hence, the first two terms can be a tractable lower bound to the log data likelihood which we can easily differentiate (P(X Z) is mostly a simple distribution like Guassian and KL term is also differntiable) and hence optimize .The combination of first two terms is also known as Variational Lower Bound ("ELBO") . The first term here is responsible for reconstructing the original input data and the second term is responsible for making approximate distribution close to input distribution .



Now equipped with our encoder and decoder networks, let's work out the (log) data likelihood:

$$\log p_\theta(x^{(i)}) = \mathbf{E}_{z\sim q_\phi(z|x^{(i)})}\left[\log p_\theta(x^{(i)})\right] \quad (p_\theta(x^{(i)}) \text{ Does not depend on } z)$$

$$= \mathbf{E}_z\left[\log \frac{p_\theta(x^{(i)} \mid z)p_\theta(z)}{p_\theta(z \mid x^{(i)})}\right] \quad \text{(Bayes' Rule)}$$

$$= \mathbf{E}_z\left[\log \frac{p_\theta(x^{(i)} \mid z)p_\theta(z)}{p_\theta(z \mid x^{(i)})}\frac{q_\phi(z \mid x^{(i)})}{q_\phi(z \mid x^{(i)})}\right] \quad \text{(Multiply by constant)}$$

$$= \mathbf{E}_z\left[\log p_\theta(x^{(i)} \mid z)\right] - \mathbf{E}_z\left[\log \frac{q_\phi(z \mid x^{(i)})}{p_\theta(z)}\right] + \mathbf{E}_z\left[\log \frac{q_\phi(z \mid x^{(i)})}{p_\theta(z \mid x^{(i)})}\right] \quad \text{(Logarithms)}$$

$$= \mathbf{E}_z\left[\log p_\theta(x^{(i)} \mid z)\right] - D_{KL}(q_\phi(z \mid x^{(i)}) \,||\, p_\theta(z)) + D_{KL}(q_\phi(z \mid x^{(i)}) \,||\, p_\theta(z \mid x^{(i)}))$$

Decoder network gives $p_\theta(x|z)$, can compute estimate of this term through sampling. (Sampling differentiable through reparam. trick, see paper.)

This KL term (between Gaussians for encoder and z prior) has nice closed-form solution!

$p_\theta(z|x)$ intractable (saw earlier), can't compute this KL term :(But we know KL divergence always >= 0.

Figure 1: Evidence lower bound derivation for a Variational Autoencoder

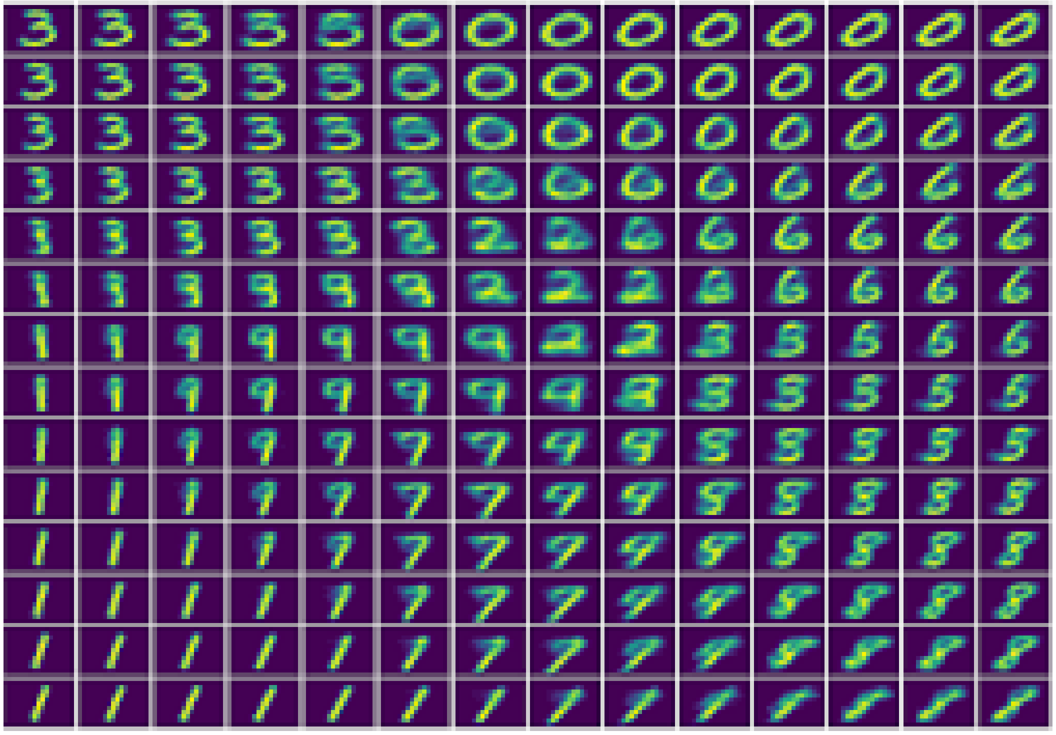


Figure 2: Digits of MNIST dataset generated using a Variational Autoencoder which had being trained using a deep convolution neural network using a 2 dimensional latent space

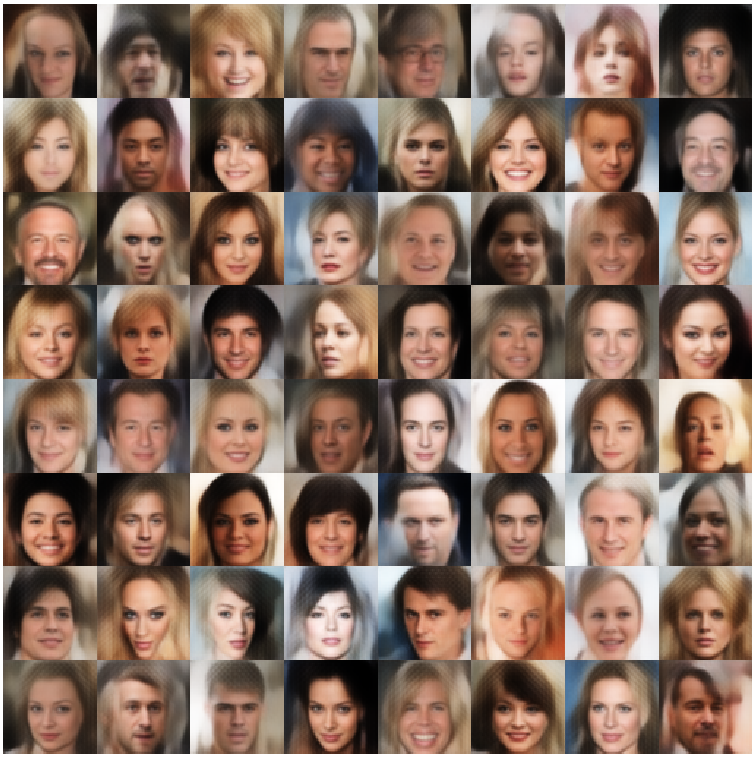


Figure 3: Faces generated on CelebA dataset using a very deep convolutional neural network and a large latent space