Topics in Probabilistic Modeling & Inference (CS698X), Spring 2019
Indian Institute of Technology Kanpur
Homework Assignment Number 4

Student Name: Subham Kumar
Roll Number: 160707
Date: April 22, 2019

QUESTION 1

Introduction to BBVI

Given a dataset we often want to infer the structure in the data. Probablistic models with latent variables is one of the way to draw conclusions from such data. But for most of the real world problems doing inference is often intractable as we need to compute the posterior distribution $p(\mathbf{Z}|\mathbf{X})$. So we need to resort to some approximation of the posterior. Variational Inference is one of the methods to do so. It chooses a variational distribution $q(\mathbf{Z}|\lambda)$ and try to bring it close to $p(\mathbf{Z}|\mathbf{X})$ as much as possible by finding the optimal value of λ such that $KL\left(q(\mathbf{Z}|\lambda)||p(\mathbf{Z}|\mathbf{X})\right)$ is minimized. This amounts to finding the best distribution from a class of distribution parameterized by λ. It can be shown that minimizing KL-divergence is same as maximizing the `Evidence Lower Bound` $L(\lambda)$(ELBO).We then re-write this ELBO in the form of expectation of some function of $\mathbf{X}, \mathbf{Z}$ w.r.t. variational distribution.
Mathematically,

$$L(\lambda) = \mathbb{E}_q\left[\log p(\mathbf{X}, \mathbf{Z}) - \log q(\mathbf{Z}|\lambda)\right]$$

Then it takes the gradient of this elbow of optimize it using standard techniques like gradient descent. But the bottleneck here is calculating the gradient itself is intractable in most pf the real world situations and hence compells us to resort to model specific techniques like Jakkola-Jordan trick proposed for logistic regression case. The BBVI paper tries to overcome this bottleneck as follows:

- It first re-writes the gradient of ELBO as an expectation of an easy to calculate function f.
- Then samples from the variational distribution and computes f.
- Then use Monte-Carlo estimate of gradients to do stochastic optimization.

Mathematically:

$$\nabla_\lambda L = \mathbb{E}_q\left[\nabla_\lambda \log q(\mathbf{Z}|\lambda)\left(\log p(\mathbf{X}, \mathbf{Z}) - \log q(\mathbf{Z}|\lambda)\right)\right]$$

Monte-Carlo estimation of the gradient:

$$\nabla_\lambda L \approx \frac{1}{S}\sum_{s=1}^{S} \nabla_\lambda \log q(\mathbf{Z}_s|\lambda)\left(\log p(\mathbf{X}, \mathbf{Z}_s) - \log q(\mathbf{Z}_s|\lambda)\right)$$

Note that the algorithm only requires that the variational distribution $q(\mathbf{Z}|\lambda)$ should be differentiable and easy to sample from and hence making it independent of the underlying model.This is why it is called Black Box VI. One drawback of this algorithm at this point is that the variance of the monte carlo estimate can be too large making it of no use. So to control the variance the paper discusses two ways:**Rao-Blackwellization** and **Control-Variates** which are discussed below.

Rao-Blackwellization:

This method is based on the idea of conditional expectation of a subset of random variables.The basic zest is,it simply uses the information for a subset of random variable on which the joint distribution is conditioned.In simplest setting,suppose $Q = \mathbb{E}[J(\mathbf{X}, \mathbf{Y})]$ and $\hat{J}(\mathbf{X}) = \mathbb{E}[J(\mathbf{X}, \mathbf{Y})|\mathbf{X}]$ then

$$\mathbb{E}[\hat{J}(\mathbf{X})] = Q \quad \text{using law of iterated expectation}$$

This implies we can use the expectation of $\hat{J}(\mathbf{X})$ in place of expectation of $J(\mathbf{X}, \mathbf{Y})$ in monte-carlo estimate.Note the variance of $\hat{J}(\mathbf{X})$,

$$Var(\hat{J}(\mathbf{X})) = Var(J(\mathbf{X}, \mathbf{Y})) - \mathbb{E}\left[(J(\mathbf{X}, \mathbf{Y}) - \hat{J}(\mathbf{X}))^2\right] \quad \text{using law of iterated variances}$$

Since the variance of the $\hat{J}(\mathbf{X})$ is lesser than that of $J(\mathbf{X}, \mathbf{Y}$, so now we can happily use it in monte-carlo estimate to reduce the variance with the same value of expectation.Next to retain the black-box nature of the algorithm,the paper makes mean-field assumption(although any variational approximation with factorization like structure mean-fied VI would work) and shows the following result:

$$J(\mathbf{X}, \mathbf{Y}|\mathbf{X}) = \mathbb{E}_y[J(x, y)]$$

Using a the above result and little more calculation it can be shown that:

$$\nabla_{\lambda_i} L = \mathbb{E}_{q_i}\left[\nabla_{\lambda_i} \log q_i(Z_i|\lambda_i)\left(\log p_i(\mathbf{X}, Z_i) - \log q_i(Z_i|\lambda_i)\right)\right]$$

Finally the Monte-Carlo estimator for the gradient w.r.t. λ_i is:

$$\nabla_{\lambda_i} L \approx \frac{1}{S}\sum_{s=1}^{S} \nabla_{\lambda_i} \log q_i(\mathbf{Z}_s|\lambda_i)\left(\log p_i(\mathbf{X}, \mathbf{Z}_s) - \log q_i(\mathbf{Z}_s|\lambda_i)\right) \text{ where } \mathbf{Z}_s \sim q_i(\mathbf{Z}|\lambda)$$

Here q_i is the distribution over variables which appear in the Markov-Blanket of Z_i and p_i is the term in the joint that depend on those variables and in optimization step we need to update each of the λ_i separately. Note that this is again model-agnostic.The paper actually shows through experimentation that this method reduces variance by several orders of magnitude.

Control Variates:

Control variates are a family of distribution functions with equivalent expectation.The zest of this way of controlling variance is again to replace the original function f with a proxy function $\hat{f}$ s.t. $\mathbb{E}[\hat{f}] = \mathbb{E}[f]$ and $\hat{f}$ has lower variance than f.Using this method,we write $\hat{f}$ as:

$$\hat{f} \approx f - a(h(z) - \mathbb{E}[h(z)])$$

Note that $\mathbb{E}[\hat{f}] = \mathbb{E}[f]$ as $h(z)$ is from control variates. Also:

$$Var(\hat{f}) = Var(f) + a^2 Var(h) - 2aCov(f, h)$$

This implies to have lower variance,we need a good co-variate which has high covariance with the function f.Note that we want the variance to be minimized hence taking the derivative w.r.t. a and setting it to zero will give: $a^* = \frac{Cov(f,h)}{Var(h)}$.We can use the empirical values of required variance and covariance to get a^*. The paper chooses the score function $\nabla_\lambda \log q(\mathbf{Z}|\lambda)$ as h whose expectation is zero.Then uses this control variate on the Rao-Blackwellized version of noisy gradient(kind of two levels of variance reduction) per component basis leading to the following expression:

$$\hat{\nabla}_{\lambda_i} L \approx \frac{1}{S}\sum_{s=1}^{S} \nabla_{\lambda_i} \log q_i(\mathbf{Z}_s|\lambda_i)\left(\log p_i(\mathbf{X}, \mathbf{Z}_s) - \log q_i(\mathbf{Z}_s|\lambda_i) - \hat{a}_i^*\right) \text{ where } \mathbf{Z}_s \sim q_i(\mathbf{Z}|\lambda)$$

where we calculate $\hat{a}_i^*$ along all the N_i dimensions of λ_i as:

$$\hat{a}_i^* = \frac{\sum_{d=1}^{N_i} Cov(\hat{f}_{id}, \hat{h}_{id})}{\sum_{d=1}^{N_i} Var(\hat{h}_{id})}$$

This leads to the algorithm BBVI-II which the paper shows performs much better and has further reduction in variance as compared to Rao-Blackwellization.
Personal Review: I liked the way the paper build upon the ground step-by-step by first letting the reader know the algorithm and then suggesting the improvements. Moreover,the paper uses most of the basic and intuitive results from the field of statistics to get to the algorithms and improvements which is actually helpful for i guess any reader.

Topics in Probabilistic Modeling & Inference (CS698X), Spring 2019
Indian Institute of Technology Kanpur
Homework Assignment Number 4

Student Name: Subham Kumar
Roll Number: 160707
Date: April 22, 2019

QUESTION 2

The Generative story can be given as:

1. Draw the K topic vectors $\{\boldsymbol{\phi}_k\}_{k=1}^{K}$ from a V dimensional $Dirichlet(\eta, \eta, ..., \eta)$
2. Draw the mixing proportion $\boldsymbol{\theta}_d$ from a K dimensional $Dirichlet(\alpha, \alpha, ..., \alpha)$ $\forall d \, \epsilon\{1, 2, ..., D\}$
3. $\forall d \, \epsilon\{1, 2, ..., D\}$,
$$z_{d,n} \sim multinoulli(\boldsymbol{\theta}_d) \;\; n \, \epsilon\{1, 2, ..., N_d\}$$
$$w_{d,n} \sim multinoulli(\boldsymbol{\phi}_{z_{d,n}}) \;\; n \, \epsilon\{1, 2, ..., N_d\}$$
$$\bar{z}_{d,k} = \frac{1}{N_d}\sum_{n=1}^{N_d} \mathbb{I}[z_{n,d} = k] \;\; \forall \;\; k = 1, 2, ..K$$
4. for each $d, d' \;\; \epsilon \; \{1, 2, ..., D\}$
$$A_{d,d'} = Bern(\sigma(f(\bar{\mathbf{z}}_d, \bar{\mathbf{z}}_{d'})))$$

In step 4 the function $f(\bar{\mathbf{z}}_d, \bar{\mathbf{z}}_{d'})$ can be for example:

- $$f(\bar{\mathbf{z}}_d, \bar{\mathbf{z}}_{d'}) = \bar{\mathbf{z}}_d^T \boldsymbol{\eta} \bar{\mathbf{z}}_{d'}$$
  where η_{kl} denotes how much topic k in documenet d is related to topic l in document d'.
- $$f(\bar{\mathbf{z}}_d, \bar{\mathbf{z}}_{d'}) = \mathbf{W}^T\{\bar{\mathbf{z}}_d \otimes \bar{\mathbf{z}}_{d'}\} + \mathbf{b}$$
  where $\otimes$ denotes the hadamard product.This function only depends on how much topic k in d is related to topic k in d' and also has lesser number of parameters.

Topics in Probabilistic Modeling & Inference (CS698X), Spring 2019
Indian Institute of Technology Kanpur
Homework Assignment Number 4

Student Name: Subham Kumar
Roll Number: 160707
Date: April 22, 2019

QUESTION 3

The joint distribution can be written as:

$$p(\mathbf{A}, \mathbf{z}, \boldsymbol{\pi}, \boldsymbol{\eta}|\alpha, a, b) = p(\mathbf{A}|\boldsymbol{\eta}, \mathbf{z})p(\mathbf{z}|\boldsymbol{\pi})p(\boldsymbol{\pi}|\alpha)p(\boldsymbol{\eta}|a, b)$$

Writing conditiona posterior for $\boldsymbol{\eta}$:

$$p(\boldsymbol{\eta}|\mathbf{A}, \mathbf{z}, a, b) \propto p(\mathbf{A}|\boldsymbol{\eta}, \mathbf{z})p(\boldsymbol{\eta}|a, b)$$

$$= \prod_{k=1}^{K}\prod_{l=1}^{K} \eta_{kl}^{N_{kl}^{\mathbf{A}}}(1-\eta_{kl})^{N_{kl}-N_{kl}^{\mathbf{A}}} \prod_{k=1}^{K}\prod_{l=1}^{K} \eta_{kl}^{a-1}(1-\eta_{kl})^{b-1}$$

Where $N_{kl}^{\mathbf{A}} = \sum_{n=1}^{N}\sum_{m=1}^{N} A_{nm}\mathbb{I}[z_n = k]\mathbb{I}[z_m = l]$ and $N_{kl} = \sum_{n=1}^{N}\sum_{m=1}^{N} \mathbb{I}[z_n = k]\mathbb{I}[z_m = l]$ This implies each η_{kl} are independent to each-other and the form is similar to Beta distribution.Hence the conditional posterior of $\eta_{kl} \sim Beta(N_{kl}^{\mathbf{A}} + a, N_{kl} - N_{kl}^{\mathbf{A}} + b)$.
Also,

$$p(\boldsymbol{\pi}|\mathbf{z}, \alpha) \propto p(\mathbf{z}|\boldsymbol{\pi})p(\boldsymbol{\pi}|\alpha)$$

$$= \prod_{n=1}^{N}\prod_{k=1}^{K} \pi_k^{\mathbb{I}[z_n=k]} \prod_{k=1}^{K} \pi_k^{\alpha-1}$$

$$= \prod_{k=1}^{K} \pi_k^{\sum_{n=1}^{N}\mathbb{I}[z_n=k]+\alpha-1}$$

which is similar to $Dirichlet(n_1 + \alpha, ..., n_K + \alpha)$ where $n_k = \mathbb{I}[z_n = k]$.
Now

$$p(z_n = k|\mathbf{z}_{-n}, \mathbf{A}, \boldsymbol{\eta}, \boldsymbol{\pi}) \propto p(\mathbf{A}|z_n = k, \mathbf{z}_{-n}, \boldsymbol{\eta})p(z_n = k|\boldsymbol{\pi})$$

$$= \pi_k \prod_{m=1}^{M} \eta_{k,z_m}^{A_{n,m}}(1-\eta_{k,z_m})^{(1-A_{n,m})}$$

which can be normalized to get:

$$p(z_n = k|\mathbf{z}_{-n}, \mathbf{A}, \boldsymbol{\eta}, \boldsymbol{\pi}) = \frac{\pi_k \prod_{m=1}^{M} \eta_{k,z_m}^{A_{n,m}}(1-\eta_{k,z_m})^{(1-A_{n,m})}}{\sum_{l=1}^{K} \pi_l \prod_{m=1}^{M} \eta_{l,z_m}^{A_{n,m}}(1-\eta_{l,z_m})^{(1-A_{n,m})}}$$

which is $multinoulli(\beta_1, ..., \beta_K)$ where $\beta_k = \frac{\pi_k \prod_{m=1}^{M} \eta_{k,z_m}^{A_{n,m}}(1-\eta_{k,z_m})^{(1-A_{n,m})}}{\sum_{l=1}^{K} \pi_l \prod_{m=1}^{M} \eta_{l,z_m}^{A_{n,m}}(1-\eta_{l,z_m})^{(1-A_{n,m})}}$

The Gibbs Sampling Algoritm:
1.Initialize $\boldsymbol{\pi}, \mathbf{z}, \boldsymbol{\eta}$ as $\boldsymbol{\pi}^{(0)}, \mathbf{z}^{(0)}, \boldsymbol{\eta}^{(0)}$,set t = 1.
2.

$$\boldsymbol{\pi}^{(t)} \sim Dirichlet(n_1^{(t-1)} + \alpha, ..., n_K^{(t-1)} + \alpha)$$

$$z_n^{(t)} \sim multinoulli(\beta_1^{(t)}, \beta_2^{(t)}, ..., \beta_K^{(t)}) \;\forall\; n \,\epsilon\, \{1, 2, ..., N\}$$

$$\eta_{kl}^{(t)} \sim Beta(N_{kl}^{\mathbf{A}(t)} + a, N_{kl}^{(t)} - N_{kl}^{\mathbf{A}(t)} + b) \;\forall\; k, l \,\epsilon\, \{1, 2, ..., K\}$$

3.If not converged,set t=t+1,go to step 2.

Topics in Probabilistic Modeling & Inference (CS698X), Spring 2019
Indian Institute of Technology Kanpur
Homework Assignment Number 4

Student Name: Subham Kumar
Roll Number: 160707
Date: April 22, 2019

QUESTION 4

Writing finite dimensional posterior of G:

$$p(G|\theta_1,\theta_2,...,\theta_N) \propto p(\theta_1,\theta_2...,\theta_N|G)p(G|G_0)$$

$$= \prod_{n=1}^{N}\prod_{k=1}^{K} p\left(\theta_n \epsilon A_k|G(A_k)\right)^{\delta_{\theta_n}(A_k)} \prod_{k=1}^{K} G(A_k)^{\alpha G_0(A_k)-1}$$

$$= \prod_{k=1}^{K} G(A_k)^{\sum_{n=1}^{N}\delta_{\theta_n}(A_k)+\alpha G_0(A_k)-1}$$

which is similar to a Dirichlet Distribution.
Hence $p(G(A_1),G(A_2),...,G(A_K)|\theta_1,\theta_2,..\theta_N) \sim Dirichlet\left((\alpha+N)G'(A_1),...,(\alpha+N)G'(A_K)\right)$ where $G'(A_k) = \frac{\alpha G_0(A_k)}{\alpha+N} + \frac{\sum_{n=1}^{N}\delta_{\theta_n}(A_k)}{\alpha+N}$.
Now using the fact that if finite dimensional marginal of a distribution G is $Dirichlet(\alpha G_0(A_1),...,\alpha G_0(A_k))$ then $G \sim DP(\alpha, G_0)$,we get the posterior to be $DP(\alpha+N, \frac{\alpha G_0}{\alpha+N} + \frac{\sum_{n=1}^{N}\delta_{\theta_n}}{\alpha+N})$

Topics in Probabilistic Modeling & Inference (CS698X), Spring 2019
Indian Institute of Technology Kanpur
Homework Assignment Number 4

Student Name: Subham Kumar
Roll Number: 160707
Date: April 22, 2019

QUESTION 5

- **A brief intro to HDP**
  Hierarchical Dirichlet Process(HDP) is a way for jointly doing multiple mixture modelling on multiple datasets i.e. if we have m groups of data and we want to do clustering in each group and also want each group to share the clustering information with each-other then we can use this HDP method to achieve it(One ad-hoc solution could have been to merge the groups to one, but this leads to deterioration of individual document structure). Also the number of clusters is not known apriori. The Hierarchical model is as follows:

  - It first defines a global random probability measure G_0 drawn from $DP(\gamma, H)$
  - Then for each group/dataset $j \epsilon 1, 2, ..., m$, it draws random probability measure G_j from $DP(\alpha_0, G_0)$.

  As Sethuraman showed using stick breaking process that draws from a DP are Discrete and hence G_0 can be written as:

$$G_0 = \sum_{k=1}^{\infty} \beta_k \delta_{\phi_k}$$

  This is the basic fact that actually helps in parameter sharing in HDP because if the G_0 were continuous then $p(G_j = G_i) = 0$ and hence the groups won't be able to share any atom.The most appealing part of this model according to me is that it can be extended to any level of hierarchy. The standard Dirichlet Process can be seen as the special case where $m = 1$ and $G_0 = G_1$. Hence no point of information sharing as there is just one document to be clustered.

- **The stick breaking construction**
  Since in HDP, each group's random probability measure G_j is sampled from $DP(\alpha_0, G_0)$, we have the stick breaking weights for each group as $\boldsymbol{\pi}_j$. This can be thought of as breaking of m sticks such that $\sum_{k=1}^{\infty} \pi_{jk} = 1 \; \forall \; j \epsilon \; \{1, 2, ..., m\}$.As shown in paper since each weight $\boldsymbol{\pi}_j \sim DP(\alpha_0, \boldsymbol{\beta})$ leading to each piece of stick $\forall \; j \epsilon \; \{1, 2, ..., m\}$ being dependent on the piece of stick in upper level($\boldsymbol{\beta}$) and hence inherently leading to parameter sharing.
  For HDP the stick breaking process goes as follows:

$$\pi'_{jk} \sim Beta\left(\alpha_0 \beta_k, \alpha_0 \left(1 - \sum_{l=1}^{k} \beta_l\right)\right)$$

$$\pi_{jk} = \pi'_{jk} \prod_{l=1}^{k-1} (1 - \pi'_{jl})$$

  . Note the difference between standard DP and HDP where in the former we use $Beta(1, \alpha_0)$ and if we use this distribution for sampling π'_{jk} then the parameter sharing nature will be lost which is there in HDP.

- **Chinese Restaurant Franchise**
  In this analogy we have m restaurant and a set of dishes $\{\phi_1, \phi_2, ..., \phi_K\}$ in global menu. And customers in each restaurant sitting at an unoccupied table first will order a dish from this menu and people sitting thereafter on that table will share this dish. The process goes as follows:

  1. A customer enters a restaurant j(let be it N_jth customer) and chooses a pre-occupied table t with probability proportional $n_{jt.}$(number of customers already sitting on table t in restaurant j) and an unoccupied table with probability proportion to α_0.In these cases the normalization constant is $1 - \alpha_0 + N_j$.
  2. Now if the customer chooses the unoccupied table,he/she must order some dish.He/she chooses dish from the already existing menu with a probability proportional to $m_{.k}$(number of tables serving dish ϕ_k) and orders a new dish(not in menu) with a probability proportional to γ. The normalization constant in these cases is $\gamma + m_{..}$,where $m_{..}$ is total number of occupied tables in all the restaurants.

  The second step of CRF can be thought of as kind of Chinese Restaurant Process(CRP) where in the later number of tables can be replaced by number of dishes and the number of persons can be replaced as number of tables.As an whole in the former(CRF) which dish is to be served at table t in restaurant j is analogous to which table is chosen by person θ_i in the later.
  Note that this model pools the information of the probability of dish served in restaurant j at table t for a new customer(posterior predictive) by consulting all the previous customers sitting at tables across all the restaurants.Due to such sharing of latent variables,it makes CRF a good prior for joint multiple mixture modelling.