# Problem 1

Similar to the proof in the LDA paper, we derive the ELBO for smoothed LDA first, then show the update formula for $\lambda, \gamma, \phi$

The ELBO for smoothed LDA is shown below, notice the first 5 terms are the same as LDA paper, and the last two are from the smoothed LDA assumption: $\beta \sim Dirichlet(\eta)$

$$\begin{aligned} L(\lambda, \gamma, \phi; \alpha, \eta) &= \log p(w|\alpha, \eta) - KL(q(\beta, \theta, z|\lambda, \gamma, \phi)||p(\beta, \theta, z|w, \alpha, \eta)) \\ &= E_q \log p(\theta|\alpha) + E_q \log p(z|\theta) + E_q \log p(w|z, \beta) - E_q \log q(\theta) - E_q \log q(z) \\ &+ E_q \log p(\beta|\eta) - E_q \log q(\beta) \end{aligned} \tag{1}$$

As shown in LDA paper appendix A.1(, and shown in class), Dirichlet distribution belongs to exponential family with natural parameter $\alpha - 1$ and sufifcient statistic $\log x$

$$\begin{aligned} f(x|\alpha) &= \frac{1}{B(\alpha)} \prod x_i^{\alpha_i - 1} \\ &= \exp\{\sum_i (\alpha_i - 1) \log x_i + \log \Gamma(\sum_i \alpha_i) - \sum_i \log \Gamma(\alpha_i)\} \end{aligned} \tag{2}$$

, and we have the below formula for exponential family.

$$\frac{d}{d\eta(\alpha)} A(\alpha) = E_{p(x)} T(x) \tag{3}$$

Thus, we show the explicit form for the sixth term

$$\begin{aligned} E_q \log p(\beta|\eta) &= E_q \log \prod_k \frac{\Gamma(\sum_i \eta_i)}{\prod_i \Gamma(\eta_i)} \prod_i \beta_{k,i}^{\eta_i - 1} \\ &= K \log \Gamma(\sum_i \eta_i) - K \sum_i \log \Gamma(\eta_i) + \sum_k E_q[\sum_i (\eta_i - 1) \log \beta_{k,i}] \\ &= K \log \Gamma(\sum_i \eta_i) - K \sum_i \log \Gamma(\eta_i) + \sum_k \sum_i (\eta_i - 1) E_q[\log \beta_{k,i}] \\ &= K \log \Gamma(\sum_i \eta_i) - K \sum_i \log \Gamma(\eta_i) + \sum_k \sum_i (\eta_i - 1) \frac{d}{d\lambda_{k,i}} (\log \Gamma(\lambda_{k,i}) - \log \Gamma(\sum_j \lambda_{k,j})) \\ &= K \log \Gamma(\sum_i \eta_i) - K \sum_i \log \Gamma(\eta_i) + \sum_k \sum_i (\eta_i - 1)(\Psi(\lambda_{k,i}) - \Psi(\sum_j \lambda_{k,j})) \end{aligned} \tag{4}$$

Similarly, we have

$$\begin{aligned} E_q \log p(\theta|\alpha) &= \log \Gamma(\sum_i \alpha_i) - \sum_i \log \Gamma(\alpha_i) + \sum_i (\alpha_i - 1)(\Psi(\gamma_i) - \Psi(\sum_j \gamma_j)) \\ E_q \log p(z|\theta) &= \sum_d \sum_n \sum_i \phi_{d,n,i} (\Psi(\gamma_{d,i}) - \Psi(\sum_j \gamma_{d,j})) \\ E_q \log p(w|z, \beta) &= \sum_d \sum_n \sum_i \sum_j \phi_{d,n,i} w_{d,n,j} (\Psi(\lambda_{k,j}) - \Psi(\sum_k \lambda_{i,k})) \\ E_q \log q(\theta) &= \sum_d (\log \Gamma(\sum_j \gamma_{d,j}) - \sum_i \log \Gamma(\gamma_{d,i}) + \sum_i (\gamma_{d,i} - 1)(\Psi(\gamma_{d,i}) - \Psi(\sum_j \gamma_{d,j}))) \\ E_q \log q(z) &= \sum_d \sum_n \sum_i \phi_{d,n,i} \log \phi_{d,n,i} \\ E_q \log q(\beta) &= \sum_k (\log \Gamma(\sum_i \lambda_{k,i}) - \sum_i \log \Gamma(\lambda_{k,i}) + \sum_i (\lambda_{k,i} - 1)(\Psi(\lambda_{k,i}) - \Psi(\sum_j \lambda_{k,j}))) \end{aligned} \tag{5}$$

Problem 1 continued on next page...

**(1)**
We take the relevant terms w.r.t. $\phi_{d,n,i}$ with a Lagrange multiplier $\lambda(\sum_i \phi_{d,n,i} - 1)$, since $\sum_i \phi_{d,n,i} = 1$

$$\begin{aligned} L = & \phi_{d,n,i}(\Psi(\gamma_{d,i}) - \Psi(\sum_j \gamma_{d,j})) \\ & + \phi_{d,n,i} \sum_j w_{d,n,j}(\Psi(\lambda_{k,j}) - \Psi(\sum_k \lambda_{i,k})) \\ & - \phi_{d,n,i} \log \phi_{d,n,i} \\ & + \lambda(\sum_i \phi_{d,n,i} - 1) \end{aligned} \tag{6}$$

and set it to zero, we have

$$\phi_{d,n,i} \propto \exp(\Psi(\gamma_{d,i}) - \Psi(\sum_j \gamma_{d,j}) + \sum_j w_{d,n,j}(\Psi(\lambda_{k,j}) - \Psi(\sum_k \lambda_{i,k}))), \tag{7}$$

the $\Psi(\sum_j \gamma_{d,j})$ term could be removed, since it is constant for fixed $d$, thus does not matter after normalizing $\phi$. Similarly, we have

$$\begin{aligned} \lambda_i &= \eta + \sum_d \sum_n \phi_{d,n,i} w_{d,n} \\ \gamma_d &= \alpha + \sum_n \phi_{d,n,i} \end{aligned} \tag{8}$$

**(2)**
Already shown in Equation(1, 4, 5)
**(3), (4)**
The vocabulary size is 100.
We show below the negative ELBO as a function of epoch and batch size

- Batched LDA's performance is greatly influenced by initialization, while full-batched LDA is not
- full batched LDA outperforms batched version consistently, almost regradless of initialization

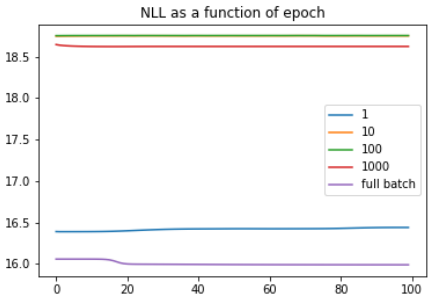


Figure 1: results of ELBO

# Problem 2

**(1)**
The ELBO for logistic regression is

$$
\begin{aligned}
L(\mu,\sigma^2) &= E_q \log p(x,\beta) - \log q(\beta|\mu,\sigma^2) \\
\nabla_{\mu,\sigma^2} L &= \nabla_{\mu,\sigma^2} \int q(\beta,|\mu,\sigma^2)(\log p(x,\beta) - \log q(\beta|\mu,\sigma^2))d\beta \\
&= \int q(\beta|\mu,\sigma^2)\nabla_{\mu,\sigma^2} \log q(\beta,|\mu,\sigma^2)(\log p(x,\beta) - \log q(\beta|\mu,\sigma^2)) \\
&\quad - q(\beta|\mu,\sigma^2)\nabla_{\mu,\sigma^2} \log q(\beta,|\mu,\sigma^2)d\beta \\
&= E_q \nabla_{\mu,\sigma^2} \log q(\beta|\mu,\sigma^2)(\log p(x,\beta) - \log q(\beta|\mu,\sigma^2) - 1) \\
&= E_q \nabla_{\mu,\sigma^2} \log q(\beta|\mu,\sigma^2)(\log p(x,\beta) - \log q(\beta|\mu,\sigma^2)) \\
\log p(x,\beta) &= \sum_i y_i \log \sigma(\beta^T x_i) + (1-y_i)\log(1-\sigma(\beta^T x_i)) + \log N(\beta|0,1) \\
\log q(\beta|\mu,\sigma^2) &= \log N(\beta|\mu,\sigma^2)
\end{aligned} \tag{9}
$$

Now we only need to solve for $\nabla_{\mu,\sigma^2} \log q(\beta|\mu,\sigma^2)$.

$$
\begin{aligned}
\log q(\beta|\mu,\sigma^2) &= \log N(\beta|\mu,\sigma^2) \\
&= -\frac{D\log\sigma^2}{2} - \frac{\|\beta-\mu\|_2}{2\sigma^2} \\
\nabla_{\mu_i} \log q(\beta|\mu,\sigma^2) &= \frac{\beta_i - \mu_i}{\sigma^2} \\
\nabla_{\sigma^2} \log q(\beta|\mu,\sigma^2) &= -\frac{D}{2\sigma^2} + \frac{\|\beta-\mu\|_2}{2(\sigma^2)^2}
\end{aligned} \tag{10}
$$

By substituting the above equation into (9), we derive the score function estimator for the gradient of ELBO w.r.t. $\mu$ and $\sigma$.

**(2)**

We use $\nabla_{\mu,\sigma^2} \log q(\beta,|\mu,\sigma^2)$ to control variation, which is also adopted in BBVI paper.

As for implementation details, I used Adam for optimizing BBVI(vanilla bbvi), BBVI with control variates(bbvi cv) and BBVI with reparameterization trick(bbvi rt). The results in ELBO(log -ELBO) are shown below.

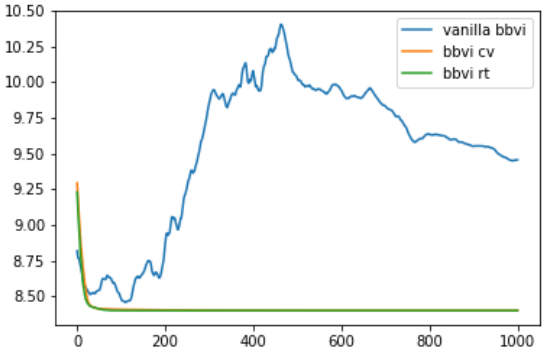


Figure 2: results of various BBVI

**(3)**

We have

$$
\begin{aligned}
\nabla_{\mu,\sigma^2} L &= \nabla_{\mu,\sigma^2} E_q \log p(x,\beta) - \log q_{\mu,\sigma^2}(\beta) \\
&= E_{q(\epsilon)} \nabla_{\mu,\sigma^2} \log p(x, g_{\mu,\sigma^2}(\epsilon)) - \log q_{\mu,\sigma^2}(g_{\mu,\sigma^2}(\epsilon))
\end{aligned} \tag{11}
$$

For logistic regression, we have (note that $\sigma$ is short for $\sigma((\mu+\sigma\epsilon)^T x_i)$ except in $\mu+\sigma\epsilon$)

$$
\begin{aligned}
\log p(x, g_{\mu,\sigma^2}(\epsilon)) &= \sum_i y_i \log\sigma + (1-y_i)\log(1-\sigma) + \log N(\mu+\sigma\epsilon|0,1) \\
\nabla_\mu \log p(x, g_{\mu,\sigma^2}(\epsilon)) &= \sum_i y_i \frac{\sigma(1-\sigma)}{\sigma} x_i + (1-y_i)\frac{-\sigma(1-\sigma)}{1-\sigma} x_i - (\mu+\sigma\epsilon) \\
&= \sum_i y_i(1-\sigma)x_i + (y_i-1)\sigma x_i - (\mu+\sigma\epsilon) \\
\nabla_{\sigma^2} \log p(x, g_{\mu,\sigma^2}(\epsilon)) &= \{\sum_i y_i \frac{\sigma(1-\sigma)}{\sigma}\epsilon x_i + (1-y_i)\frac{-\sigma(1-\sigma)}{1-\sigma}\epsilon x_i - (\mu+\sigma\epsilon)\epsilon\}\frac{d\sigma}{d(\sigma^2)} \\
&= (\sum_i y_i(1-\sigma)x_i + (y_i-1)\sigma x_i - (\mu+\sigma\epsilon))\frac{\epsilon}{2\sqrt{\sigma^2}} \\
\log q_{\mu,\sigma^2}(g_{\mu,\sigma^2}(\epsilon)) &= \log N(\mu+\sigma\epsilon|\mu,\sigma^2) \\
&= -\frac{D}{2}\log\sigma^2 + C \\
\nabla_{\sigma^2} \log q_{\mu,\sigma^2}(g_{\mu,\sigma^2}(\epsilon)) &= -\frac{D}{2\sigma^2}
\end{aligned}
\tag{12}
$$

The results of BBVI with reparameterization trick is shown in **(2)**. Here we show the performance in minibatch senario.

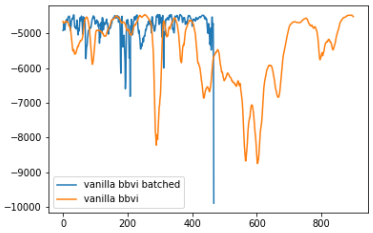


Figure 3: results of vanilla BBVI

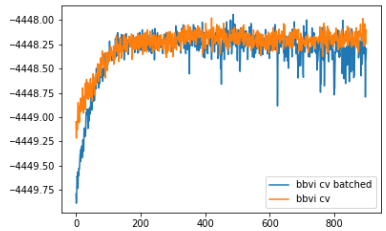


Figure 4: results of BBVI cv

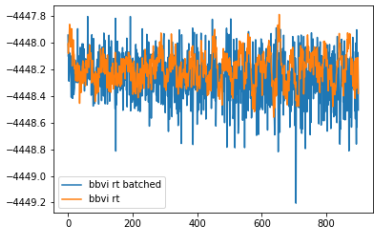


Figure 5: results of BBVI rt

The settings are: batch size being 1000, sample result every 10 iteration *i.e.* one epoch. Notice that vanilla BBVI diverged after 500 epoch even in such a large batch size. For BBVI with control variates and reparameterization trick, we only show their ELBO after 100 epoch to show their performance after convergence.

**(4)**

We show the result w.r.t. $Var(\mu)$ and $Var(\sigma)$ below. Notice that in my implementation, we estimate $\nabla_{\log\sigma^2}$ instead. As batch size increases, BBVI+CV/RT reduces the variances more and the variance becomes more stable.

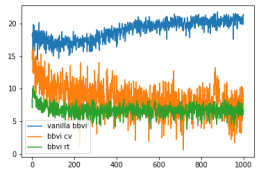


Figure 6: results of $\mu$, 4 samples

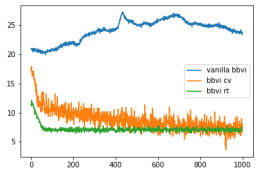


Figure 7: results of $\mu$, 32 samples

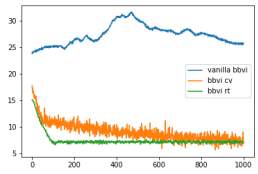


Figure 8: results of $\mu$, 64 samples

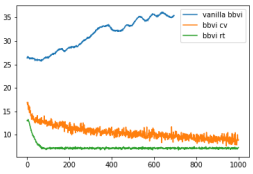


Figure 9: results of $\mu$, 128 samples

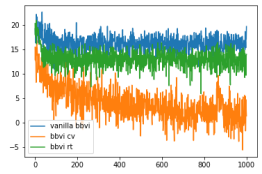


Figure 10: results of $\sigma$, 4 samples

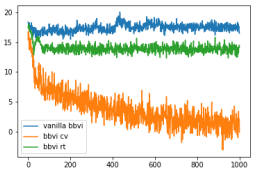


Figure 11: results of $\sigma$, 32 samples

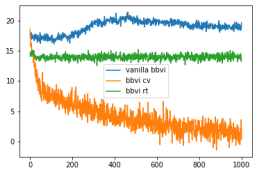


Figure 12: results of $\sigma$, 64 samples

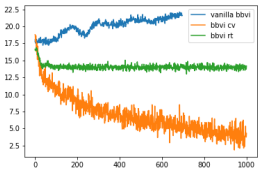


Figure 13: results of $\sigma$, 128 samples

**(6)**

NOTE: no better results made!

- replace $\sigma$ with $\sigma_1, \sigma_2$ theortically should perform no worse than a single $\sigma$, but in practice not(ELBO drops to -1460)
- replace Gaussian prior by t-distribution or Laplace distribution does not work neither(ELBO drops to -1480)