Question 1. Bayes rule and predictive distributions (25 + 25 points)
We have observations $x_1, \ldots, x_n$ with each $x \in \{0, 1, 2, \ldots\}$. We choose to model this as $x_i \overset{iid}{\sim} p(x|\pi, r)$, where Negative Binomial

$$p(x|\pi, r) = \binom{x + r - 1}{x} \pi^x (1 - \pi)^r.$$

We want to learn π, so we place a beta prior on it, $\pi \sim \text{Beta}(a, b)$.

a) Calculate the posterior distribution of π, $p(\pi|x_1, \ldots, x_n)$.

b) What is the predictive distribution of a new x? That is, calculate $p(x_{n+1}|x_1, \ldots, x_n)$ under this modeling assumption.

a) $p(\pi|x_1, \ldots, x_n) \propto \prod_{i=1}^{n} \binom{x_i + r - 1}{x_i} \pi^{x_i}(1-\pi)^r \cdot \Gamma(a+b)/\Gamma(a)\Gamma(b)\, \pi^{a-1}(1-\pi)^{b-1}$

$\propto \prod_{i=1}^{n} \pi^{x_i}(1-\pi)^r \cdot \pi^{a-1}(1-\pi)^{b-1}$

$\propto \pi^{\sum_{i=1}^{n} x_i + a - 1}(1-\pi)^{nr+b-1}$

$= \text{Beta}(a', b')$ where $a' = \sum_{i=1}^{n} x_i + a$, $b' = nr + b$

b) $p(x_{n+1}|x_1, \ldots, x_n) = \int_0^1 p(x_{n+1}|\pi)\, p(\pi|x_1, \ldots, x_n)\, d\pi$

$= \int_0^1 \binom{x_{n+1} + r - 1}{x_{n+1}} \pi^{x_{n+1}}(1-\pi)^r \Gamma(a'+b')/\Gamma(a')\Gamma(b')\, \pi^{\sum_{i=1}^{n} x_i + a - 1}(1-\pi)^{nr+b-1}\, d\pi$

$= \binom{x_{n+1} + r - 1}{x_{n+1}} \Gamma(a'+b')/\Gamma(a')\Gamma(b') \int_0^1 \pi^{\sum_{i=1}^{n+1} x_i + a - 1}(1-\pi)^{(n+1)r+b}\, d\pi$

$= \binom{x_{n+1} + r - 1}{x_{n+1}} \Gamma(a'+b')/\Gamma(a')\Gamma(b') \cdot \Gamma\left(\sum_{i=1}^{n+1} x_i + a\right)\Gamma((n+1)r+b)/\Gamma\left(\sum_{i=1}^{n+1} x_i + a + (n+1)r + b\right)$

Question 2. Expectation-maximization algorithm (50 points)

You are given a data set $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$, where $x \in \mathbb{R}^d$ and $y \in \mathbb{R}$. You model this using Bayesian linear regression with the following prior structure,

$$y_i \overset{ind}{\sim} \text{Normal}(x_i^T w, \alpha^{-1}), \quad w \sim \text{Normal}(0, \lambda^{-1} I), \quad \lambda \sim \text{Gamma}(a, b).$$

Derive an Expectation-Maximization algorithm for optimizing $\ln p(y, \lambda | x)$ over λ, where the vector w functions as the variable being integrated out.

Please note the following about what I am looking for:

- It is not necessary to show all work in deriving $q(w)$ using Bayes rule, but it must be clear that you know how Bayes rule is used here, and also what the solution of $q(w)$ is.
- It must be clear that you understand what constitutes the "E" and the "M" steps. Partial credit will be given for correct algorithms without a clear path to the solution.
- You must give pseudo-code for optimizing λ including the equations that you would implement in a coding language (similar to the algorithms outlined in the notes).
- If any expectations remain in your final algorithm, you should indicate what they are equal to in the pseudo-code.

$p(y, \lambda | x) = \int p(y, w, \lambda | x)\, dw \qquad p(y, w, \lambda | x) = \prod_{i=1}^{N} p(y_i | w, x_i)\, p(w | \lambda)\, p(\lambda)$

Deriving the EM equations

$$p(y, \lambda | x) = p(w, y, \lambda | x) / p(w | y, \lambda, x)$$

$$\ln p(y, \lambda | x) = \ln p(w, y, \lambda | x) - \ln p(w | y, \lambda, x)$$

$$q(w) \ln p(y, \lambda | x) = q(w) \ln p(w, y, \lambda | x) - q(w) \ln p(w | y, \lambda, x)$$

$$\ln p(y, \lambda | x) = \int q(w) \ln p(w, y, \lambda | x)\, dw - \int q(w) \ln q(w)\, dw + \int q(w) \ln q(w)\, dw - \int q(w) \ln p(w | y, \lambda, x)\, dw$$

$$= \int q(w) \ln \left(p(w, y, \lambda | x) / q(w) \right) dw + \int q(w) \ln \left(q(w) / p(w | y, \lambda, x) \right) dw$$

At iteration t:

E-Step $\quad q(w) = q(w | y, \lambda, x) \propto \prod_{i=1}^{N} p(y_i | w, x)\, p(w | \lambda)$

$$\propto \exp\{ -\tfrac{1}{2} (y^T y - 2 w^T x y + w^T x x^T w) - \lambda_{t-1} \tfrac{1}{2} (w^T w) \}$$

$$\propto \exp\{ -\tfrac{1}{2} (w^T (x x^T + \lambda_{t-1} I) w - 2 w^T x y) \}$$

$$\propto \exp\{ -\tfrac{1}{2} (w^T M_t w - 2 w^T M_t M_t^{-1} x y + (M_t^{-1} x y)^T M (M_t^{-1} x y)) \}$$

$$\propto \exp\{ -\tfrac{1}{2} (w^T - M_t^{-1} x y)\, M\, (w^T - M_t^{-1} x y) \}$$

$$= N(M_t^{-1} x y, M_t^{-1})$$

$\mathbb{E}_{q(w)}[w] = M_t^{-1} x y$

$\mathbb{E}_{q(w)}[w^T w] = \text{tr}(M_t^{-1}) + \mathbb{E}_{q(w)}[x]^T \mathbb{E}_{q(w)}[x]$

M-step

$$\ln p(y, w, \lambda | x) = \frac{N}{2}\ln\left(\frac{\alpha}{2\pi}\right) - \frac{\alpha}{2}(y^T y - 2w^T x y + w^T x x^T w) + \frac{d}{2}\ln\left(\frac{\lambda}{2\pi}\right) - \frac{\lambda}{2} w^T w + a\ln(b) - \ln\Gamma(a) + (a-1)\ln\lambda - b\lambda$$

$$\mathbb{E}_{q(w)}[\ln p(y, w, \lambda | x)] = \mathbb{E}_{q(w)}\left[-\frac{\lambda}{2} w^T w\right] + \frac{d}{2}\ln(\lambda) + (a-1)\ln\lambda - b\lambda + \text{constant w.r.t to } \lambda$$

$$\nabla_\lambda \mathbb{E}_{q(w)}[\ln p(y, w, \lambda | x)] = -\frac{1}{2}\mathbb{E}_{q(w)}[w^T w] + \frac{d}{2\lambda} + a - 1/\lambda - b$$

equate to 0 to maximise

$$\frac{1}{2}\mathbb{E}_{q(w)}[w^T w] + b = \frac{1}{\lambda}\left(\frac{d}{2} + a - 1\right)$$

$$\mathbb{E}_{q(w)}[w^T w] + 2b = \frac{1}{\lambda}(d + 2a - 2)$$

$$\lambda = d + 2a - 2 / \mathbb{E}_{q(w)}[w^T w] + 2b$$

Algorithm

1. Initialise λ

2. For iteration $t = 1, \ldots, T$

   a) E-step: Calculate the following where $M_t = \alpha x x^T + \lambda_{t-1} I$

   $$\mathbb{E}_{q(w)}[w] = M_t^{-1} x y$$

   $$\mathbb{E}_{q(w)}[w^T w] = \text{tr}(M_t^{-1}) + \mathbb{E}_{q(w)}[w]^T \mathbb{E}_{q(w)}[w]$$

   b) M-Step: Update λ with $\mathbb{E}_{q(w)}[w^T w]$ as

   $$\lambda_t = d + 2a - 2 / (\mathbb{E}_{q(w)}[w^T w] + 2b)$$

   c) Calculate $\mathbb{E}[\ln p(y, w, \lambda | x)] = \frac{N}{2}\ln\left(\frac{\alpha}{2\pi}\right) - \frac{\alpha}{2}(y^T y - 2\mathbb{E}_{q(w)}[w]^T x y + \text{tr}((M_t^{-1} + \mathbb{E}_{q(w)}[w]\mathbb{E}_{q(w)}[w]^T) x$
   for convergence $\frac{d}{2}\ln\left(\frac{\lambda_t}{2\pi}\right) - \frac{\lambda_t}{2}\mathbb{E}_{q(w)}[w^T w] + a\ln(b) - \ln\Gamma(a) + (a-1)\ln\lambda_t - b\lambda_t$

Question 3. Variational inference (50 points)

You are given a data set $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$, where $x \in \mathbb{R}^d$ and $y \in \mathbb{R}$. You model this using Bayesian linear regression with the following prior structure,

$$y_i \overset{ind}{\sim} \text{Normal}(x_i^T w, \alpha^{-1}), \quad w \sim \text{Normal}(0, \lambda^{-1} I), \quad \alpha \sim \text{Gamma}(a, b), \quad \lambda \sim \text{Gamma}(e, f).$$

Derive a variational inference algorithm for learning $q(w, \alpha, \lambda) \approx p(w, \alpha, \lambda | y, x)$ using the factorization $q(w, \alpha, \lambda) = q(w)q(\alpha)q(\lambda)$. For your q distributions, use

$$q(w) = \text{Normal}(\mu', \Sigma'), \quad q(\alpha) = \text{Gamma}(a', b'), \quad q(\lambda) = \text{Gamma}(e', f').$$

Again, please note the following about what I am looking for:

- You are free to use the "direct method" from the notes, but the "optimal method" is much easier and faster. The q distributions above are the optimal ones.
- Therefore, you do not need to explicitly calculate the variational objective function to receive full credit ("$\mathcal{L}$" in the notes). But again, you are free to take this approach.
- What I am looking for is an equation-based algorithm (not a gradient-based algorithm) for learning the pairs $(a', b'), (e', f'), (\mu', \Sigma')$. Since it's technically possible to calculate $\mathcal{L}$ and optimize with gradient methods, you will be given partial credit if you do this.
- For full credit, your work must show a clear path to your answer.
- Summarize your algorithm for optimizing $(a', b'), (e', f'), (\mu', \Sigma')$ using pseudo-code similar to how is done in the notes. (This will make your final results easier to find and read for the graders.)
- If any expectations remain in your final algorithm, you should indicate what they are equal to in the pseudo-code.

The joint likelihood factorises as $p(y, w, \alpha, \lambda | x) = p(w)p(\alpha)p(\lambda)\prod_{i=1}^N p(y_i | w, x, \lambda, x_i)$

With the factorisation above, write the variational objective as:

$$\mathcal{L} = \int q(w)q(\alpha)q(\lambda) \underbrace{\ln p(y, w, \alpha, \lambda | x)}_{d\lambda\, d\alpha\, dw} - \int q(w)q(\alpha)q(\lambda)\left(\ln q(w) + \ln q(\alpha) + \ln(q\lambda)\right) d\lambda\, d\alpha\, dw$$

$$= \int q(w)q(\alpha)q(\lambda) \ln p(y, w, \alpha, \lambda | x)\, d\lambda\, d\alpha\, dw - \int q(w) \ln q(w) dw - \int q(\alpha) \ln q(\alpha) d\alpha - \int q(\lambda) \ln q(\lambda) d\lambda$$

(not related; not related)

$$q(\lambda) \propto \exp\left\{\mathbb{E}_{-q(\lambda)}\left[\ln p(y, w, \alpha, \lambda, x) + \ln(w) + \ln(\alpha) + \ln(\lambda)\right]\right\}$$

$$\propto \exp\left\{\mathbb{E}_{-q(\lambda)}\left[\ln(w) + \ln(\lambda)\right]\right\}$$

$$\propto \exp\left\{\mathbb{E}_{q(w)}\left[\tfrac{d}{2}\ln(\lambda) - \tfrac{\lambda}{2} w^T w\right] + (e-1)\ln\lambda - f\lambda\right\}$$

$$\propto \lambda^{\frac{d}{2}+e-1} \exp\left(-\lambda\left(f + \tfrac{1}{2}\mathbb{E}_{q(w)}\left[w^T w\right]\right)\right)$$

$\Rightarrow \text{Gamma}(\lambda | e', f')$ where

$$e' = d/2 + e - 1, \quad f' = f + \tfrac{1}{2}\mathbb{E}_{q(w)}\left[w^T w\right]$$

$$q(\alpha) \propto \exp\{\mathbb{E}_{-q(\alpha)}[\ln p(y|w,\alpha,\lambda,x) + \ln(w) + \ln(\alpha) + \ln(\lambda)]\}$$

$$\propto \exp\{\mathbb{E}_{-q(\alpha)}[\ln p(y|w,\alpha,\lambda,x) + \ln(\alpha)]\}$$

$$\propto \exp\{\mathbb{E}_{q(w)}[\sum_{i=1}^{n} \tfrac{1}{2}\ln(\alpha) - \tfrac{\alpha}{2}(y_i - x_i^T w)^2] + (a-1)\ln\alpha - b\alpha\}$$

$$\propto \alpha^{\frac{N}{2}+a-1} \exp\left(-\alpha\left(b + \tfrac{1}{2}\sum_{i=1}^{n}\mathbb{E}_{q(w)}[y_i - x^T w]^2\right)\right)$$

$$= \text{Gamma}(\alpha|a', b') \text{ where } a' = \tfrac{N}{2} + a, \quad b' = b + \tfrac{1}{2}\sum_{i=1}^{n}\mathbb{E}_{q(w)}[(y_i - x^T w)^2]$$

(not related; not related)

$$q(w) \propto \exp\{\mathbb{E}_{-q(w)}[\ln p(y|w,\alpha,\lambda,x) + \ln(w) + \ln(\alpha) + \ln(\lambda)]\}$$

$$\propto \exp\{\mathbb{E}_{-q(w)}[\ln p(y|w,\alpha,\lambda,x) + \ln(w)]\}$$

$$\propto \exp\{\mathbb{E}_{q(\alpha)}[\sum_{i=1}^{n} \tfrac{1}{2}\ln(\alpha) - \tfrac{\alpha}{2}(y_i - x_i^T w)^2] - \tfrac{\lambda}{2} w^T w\}$$

$$\propto \exp\{\tfrac{n}{2}\mathbb{E}_{q(\alpha)}[\ln\alpha] - \tfrac{1}{2}(y^T \mathbb{E}_{q(\alpha)}[\alpha] y - 2w^T X \mathbb{E}_{q(w)}[\alpha] y + w^T(\underbrace{\lambda I + \mathbb{E}_{q(\alpha)}[\alpha] X X^T}_{M}) w^T)\}$$

$$\propto \exp\{-\tfrac{1}{2}(w^T M w - 2w^T M M^{-1}\mathbb{E}_{q(w)}[\alpha] X y + (M^{-1}\mathbb{E}_{q(\alpha)}[\alpha] X y)^T M (M^{-1}\mathbb{E}_{q(\alpha)}[\alpha] X y))\}$$

$$= N(\mu', \Sigma') \text{ where } \Sigma' = M^{-1}, \quad \mu' = M^{-1}\mathbb{E}_{q(\alpha)}[\alpha] X y$$

$$\mathbb{E}_{q(\alpha)}[\alpha] = a'/b'$$

$$\mathbb{E}_{q(w)}[w] = M^{-1}(a'/b') X y \qquad \mathbb{E}_{q(w)}[w^T w] = tr(M^{-1}) + \mathbb{E}_{q(w)}[x^T]\mathbb{E}[x]$$

$$\mathbb{E}_{q(w)}[w w^T] = \mathbb{E}_{q(w)}[w]\mathbb{E}_{q(w)}[w]^T + M^{-1}$$

$$\sum_{i=1}^{n}\mathbb{E}_{q(w)}[(y_i - x_i^T w)^2] = \mathbb{E}_{q(w)}[y^T y - 2y^T x^T w - w^T x x^T w]$$

$$= y^T y - 2y^T x^T \mathbb{E}_{q(w)}[w] - tr(\mathbb{E}[w w^T] x x^T)$$

$$= y^T y - 2y^T x^T \mathbb{E}_{q(w)}[w] - tr((\mathbb{E}_{q(w)}[w]\mathbb{E}_{q(w)}[w]^T + M^{-1}) x x^T)$$

Algorithm:

1. Initialise a_0', b_0', e_0', f_0', μ_0' and Σ_0'.

2. For iterations $t = 1, \ldots, T$

a) Update $q(\lambda)$ by setting $e_t' = d/2 + e_{t-1}' - 1$ and $f_t' = f_{t-1}' + \tfrac{1}{2}tr(M_{t-1}^{-1}) + \mathbb{E}_{q(w)}[X]^T\mathbb{E}[X]$

where $M_{t-1}^{-1} = \Sigma_{t-1}'$ and $\mathbb{E}_{q(w)} = M_{t-1}^{-1}(a_{t-1}'/b_{t-1}') X y$

b) Update $q(\alpha)$ by setting $a_t' = a_{t-1}' + \frac{N}{2}$, $b_t' = b_{t-1}' + y^T y - 2y^T x^T \mathbb{E}_{q(w)}[w] - tr((M_{t-1}^{-1} + \mathbb{E}_{q(w)}[w]\mathbb{E}_{q(w)}[w]^T) x$

c) Update $q(w)$ by setting $\Sigma_t' = (\lambda_t I + a_t'/b_t' X X^T)$ and $\mu_t' = \Sigma_t'(a_t'/b_t') X y$

d) Assess convergence by evaluating $L(a_t', b_t', e_t', f_t', \mu_t', \Sigma_t')$

$$= \mathbb{E}_{q(w,\alpha,\lambda)}[\ln p(y, w, \alpha|\lambda)]$$