Name:

Roll No.: **Dept.:**

IIT Kanpur
CS772A (PML)
End-sem Exam
Date: April 29, 2023

Instructions: *Total:* **100 marks**

1. Total duration: **3 hours**. Please write your name, roll number, department on **all pages**.
2. This booklet has 10 pages (9 pages + 1 page for rough work). No part of your answers should be on pages designated for rough work. Additional rough sheets may be provided if needed.
3. Write/mark your answers clearly in the provided space. Please keep your answers precise and concise.
4. Avoid showing very detailed derivations (you may use the rough sheet for that). In some cases, you may directly use the standard results/expressions provided on page 6 of this booklet.

Section 1 (True or False: 15 X 1 = 15 marks). For each of the following simply write **T** or **F** in the box.

1. [T] Gibbs sampling is applicable even if there is no (local) conjugacy. (Reason: Applicable in general; local conjugacy only simplies it further since CPs are available in closed form and thus easier to sample from)
2. [F] Choosing the likelihood and the prior as exponential family distributions results in a closed form posterior distribution. (Reason: The pair must be conjugate to each other. Just being from exp-fam is not enough)
3. [F] Unlike MCMC, variational inference gives point estimates of the unknowns.
4. [T] Denoising diffusion models are slower at generation as compared to GAN or VAE
5. [T] In active learning, inputs for which the current model is least confident in its prediction are likely to be more informative.
6. [F] EM does not provide any uncertainty estimates for any of the unknowns of the model. (Reason: It does, for the unknowns for which EM computes CP)
7. [F] The true joint posterior of all the unknowns of a model is equal to the product of their conditional posteriors.(Reason: No, we may approximate the true joint posterior by a product of distributions of each variable as in mean-field assumption, and then use the CP of each variable to estimate the corresponding term in the product. Or we may use Gibbs sampling which samples from the CPs, but that also is an approximation. In general the true joint posterior isn't a product of the CPs as you can verify using the chain rule)
8. [F] EM algorithm yields local optima of local variables and global optima for global variables. (Reason: EM only yields a local optima of the parameters in the M step. Even in the E step, it computes a CP of the variables that we estimates in the E step but we won't necessarily get a global optima from the CP)
9. [F] A symmetric proposal distribution makes the calculation of acceptance probability significantly faster in MCMC.(Reason: No, the ratio of the unnormalized parts has a more significant computional overhead and usually involves computing the likelihood over the entire dataset)
10. [T] Monte-Carlo dropout based approximation of the posterior can be obtained from the point estimates of the model parameters.(Reason: MC dropout takes a point estimate of the model parameters and applies different masks/noise vectors to yield different purturbations of the model parameters which are akin to samples from the posterior)
11. [F] SWAG and Laplace approximation both yield the same approximation of the posterior.
12. [T] Likelihood is a function, not a probability distribution.
13. [T] It is not possible to do MLE or MAP to estimate the parameters of a generative adversarial network (GAN). (Reason: Because GAN does not have an explicitly defined likelihood model so we can't do MLE/MAP for parameter estimation and have to solve the min-max objective discussed in the class)
14. [T] Monte Carlo sampling can be used to compute both ELBO as well as its derivatives.
15. [F] Generalized linear model (GLM) is a generative model.

Name:

Roll No.: Dept.:

IIT Kanpur
CS772A (PML)
End-sem Exam
Date: April 29, 2023

Section 2 (15 short answer questions: 15 x 3 = 45 marks). .

1. If the target distribution has multiple modes, standard SGLD is prone to generating samples near one of the modes. How can SLGD, or other gradient based sampling methods similar to SGLD, address this issue and generate samples from around the multiple modes of the distribution? Justify your answer briefly.

   Can use SGLD with cyclic learning rate which, when given a big sudden increase in its value can help jumping from an already explored mode to another mode and generate samples from around that mode. Note: If you have answered HMC, we will accept that solution as well since it has somewhat better exploration property (though won't get full marks)

2. Give two reasons as to why Gaussian Process (GP) is a good method to estimate the surrogate model of the function being optimized via Bayesian Optimization.Why won't you use a Bayesian linear regression model for this purpose?

   The two reasons: (1) GP is nonlinear and can model a wide range of function shapes (which is important since the function being optimized can have any shape which we don't know beforehand); (2) GP provides uncertainty estimates. Bayesian linear regression won't be suitable since it can only model linear functions.

3. Briefly explain how entropy of the posterior distribution of model parameters can be used for active learning.

   Can look at the reduction in the entropy of the posterior distribution. The input whose inclusion in the training set results in the maximum reduction in the entropy can be selected.

4. Suppose you have run MCMC to generate (a sufficiently large number of) samples from a distribution $p(\boldsymbol{z})$. How would you use the generated samples to find the maxima (mode) of this distribution (need not be the true mode; an approximate mode is fine)?

   Using the generated samples $\theta_1, \theta_2, \ldots, \theta_S$, and pick the one that has the highest probability/probability-density under the posterior. Mathematically $\hat{\theta} = \arg\max_{\theta_1,\theta_2,\ldots,\theta_S} p(\theta|X)$

5. Can the integral $\int_{-\infty}^{\infty} \exp[-\lambda(x-\mu)^2]dx$ be computed exactly? If yes, write its value. If no, state why.

   Yes, it is the integral of the unnormalized part of PDF $\mathcal{N}(x|\mu,(2\lambda)^{-1}) = \sqrt{\frac{2\lambda}{2\pi}}\exp[-\lambda(x-\mu)^2]dx$. Since the PDF must integrate to 1, it is easy to see that $\int_{-\infty}^{\infty}\exp[-\lambda(x-\mu)^2]dx = \sqrt{\frac{\pi}{\lambda}}$

6. Suppose you have tossed a coin a number of times. Now suppose you want to compute the probability that $\theta \leq 0.4$ where θ is the probability of heads. Briefly suggest a Bayesian way to do this.

   Given the posterior $p(\theta|X)$ based on observations, we can compute the CDF $\int_0^{0.4} p(\theta|X)\theta$

7. Which of these inference algorithms can be used to infer the posterior over the weights $\boldsymbol{w}$ of logistic regression model, assuming no additional variables are introduced for the model: (1) Expectation-Maximization, (2) Gibbs Sampling, (3) Metropolis-Hastings Sampling, (4) Stochastic Gradient Langevin Dynamics? Briefly justify your answers.

   Standard logistic regression is a non-conjugate model. You can't use EM or Gibbs sampling (in order to use them, you will need to introduce auxiliary variables similar to how we used them in probit regression you worked on in HW2). On the other, MH and SGLD don't require any conjugacy and thus either can be used.

8. Can we use the generative approach to learn a regression model? If yes, can it be done in the same way as we learn a classification model using a generative approach, i.e., defining $p(y|\boldsymbol{x}) = \frac{p(y)p(\boldsymbol{x}|y)}{p(\boldsymbol{x})}$? Briefly justify your answer.

   We can use a generative model for regression as well but doing so will require us to estimate $p(x, y)$. However, this distribution can't be estimated by factorizing it as $p(y)p(x|y)$ in the regression case since there is no notion of a class-conditional distribution. We will need to estimate $p(x, y)$ using other ways.

9. Consider a model with data X from a likelihood model $p(X|\theta, \beta)$ and prior $p(\theta|\alpha)$ on the parameters θ. Briefly state how the two hyperparameters β, α of this model (one is part of likelihood and the other is part of the prior) can be estimated using expectation-maximization (EM) if (1) We want their point estimates, and (2) If we want their conditional posterior distributions.

   If we want their point estimates then we will estimate the CP of θ in the E step and compute the point estimate of β, α in the M step. If we want the CP of β, α, then we will need to compute those in the E step and compute the point estimate of θ in the M step.

10. State 3 advantages (should be distinct from each-other) of Gaussian Process based models over standard, kernelized supervised learning models, such as SVM, or methods such as nearest neighbors.

    (1) GPs give uncertainty estimate (both model and predictive uncertainty), (2) GPs allow us to estimate the hyperparameters such as the kernel hyperparameters, (3) GPs can incorporate prior knowledge about the function being learned.

11. What is the difference between variational inference and variational EM? When would you need to use variational EM as opposed to standard EM?

    In VI, we infer the (approximate) posterior of all the unknowns whereas in VEM we have two sets of unknowns - for one we estimate the CP and for the other we do point estimation. VEM can be used if the CP required in the E step of EM is not tractable; we can approximate this intractable CP using VI, and such a version of EM is called VEM.

12. What is the advantage of selecting the "best" hyperparameter values using an MLE-II approach as compared to using cross-validation?

    We only have to run MLE-II once; it's basically an optimization problem whose solution will give us the best value of the hyperparameters. In contrast, cross-validation (CV) must be repeated multiple times, each time with a different candidate value of the hyperparameter(s). Also, MLE-II doesn't waste training data (we can use all the training data to learn the model) whereas in CV, we must also set aside part of the training data as our validation data, so we get less training data to learn from.

13. A zero-mean Gaussian prior is equivalent to using L2 regularization on the weight vector w. Can such a prior be used to impose different amounts of regularization on different components of the weight vector? If yes, how? If no, why not?

    Yes. We can use a Gaussian prior with zero mean and a diagonal covariance matrix. The entires of the diagonal matrix can be used to specify the different amounts of regularization we want on different components of the weight vector (or we can even learn the optimal values of these entries by treating them as additional unknowns and using MLE-II or other methods we studied to estimate these unknowns).

14. Briefly state why the marginal likelihood of a model can also be seen as a special case of the posterior predictive distribution.

    You can think of the marginal likelihood as a prior predictive distribution which is a special case of posterior predictive distribution when the posterior = prior (i.e., we haven't updated it yet).

15. Assume you have K candidate models (assume probabilistic models) that you can possibly try out for a classification problem and don't know which one is the "best". Briefly explain how would a fully Bayesian approach handle this problem.

You can compute the marginal likelihood $p(X|k)$ (or its approximation), or the posterior probabilities $p(k|X)$of each model $k = 1, 2, \ldots, K$, or the ELBO for each model and compare them models based on any of these quantities.

Section 3 (4 not-so-short answer questions: 8+8+8+16 = 40 marks). .

1. Assume N observations $\mathbf{X} = \{x_1, \ldots, x_N\}$ drawn i.i.d. from the exponential distribution,which is defined as $p(x_n|\theta) = \theta \exp(-\theta x_n)$ and the prior on the parameter $\theta > 0$ is $p(\theta) = \text{Gamma}(\theta|a, b) = \frac{b^a}{\Gamma(a)} \theta^{a-1} \exp(-b\theta)$. What is the marginal likelihood $p(\mathbf{X}|a, b)$? Give your answer as a closed-form expression (not an integral). Avoid very detailed derivation; show only the basic steps and write down the final expression.

Note that $p(\mathbf{X}|\theta) = \prod_{n=1}^N p(x_n|\theta) = \theta^N \exp(-\theta \sum_{n=1}^N x_n)$. From this, we can get the marginal likelihood as $p(\mathbf{X}|a, b)$ by integrating out θ over its prior distribution as follows

$$p(\mathbf{X}|a, b) = \int p(\mathbf{X}|\theta) p(\theta|a, b) d\theta = \int \theta^N \exp(-\theta \sum_{n=1}^N x_n) \frac{b^a}{\Gamma(a)} \theta^{a-1} \exp(-b\theta) d\theta$$

The constant $\frac{b^a}{\Gamma(a)}$ comes out and the remaining integral is nothing but the normalization constant of $\text{Gamma}(\theta|a + N, b + \sum_{n=1}^N)$, which is also the posterior of θ. Therefore $p(\mathbf{X}|a, b) = \frac{b^a}{\Gamma(a)} \frac{\Gamma(a+N)}{(b+\sum_{n=1}^N)^{a+N}}$ (Overall, this marginal likelihood can be seen as a ratio of two normalization constants (of the prior and posterior of θ); you may recall our discussion of exponential family distributions and this property holds not just for this example but for the marginal distributions of all exp-fam distributions).

2. Briefly describe what collapsing means in the context of approximate Bayesian inference, and what are the benefits of collapsing? You may use an example, such as a model like Gaussian mixture model or Latent Dirichlet Allocation. You don't need to be excessively detailed (i.e., no derivations etc); it would suffice to explain via a simple example the basic difference between the uncollapsed vs collapsed inference.

Collapsing refers to getting rid of the variables that we don't care about and/or getting rid of which simplifies inference procedure. It reduces the number of variables to be estimated and often results in faster convergence (since our space of solution is now smaller).

In a GMM, if using collapsing, we may consider collapsing the mixing proportions π. If using NIW prior on cluster means/covariances, it is possible to integrate out those as well and only infer the cluster assignment $\mathbf{Z}$. Likewise, for LDA, as we have already seen in HW3, we can integrate out topic vectors ϕ_k's and topic proportion vectors θ_d's and only infer the word-to-topic assignments (from which we can still infer θ_d, ϕ_k).

Name:

Roll No.: **Dept.:**

IIT Kanpur
CS772A (PML)
End-sem Exam
Date: April 29, 2023

3. Suppose you are given a dataset of N labeled images $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^N$, where each image is either a picture of a cat or a dog. You want to train a logistic regression model with weights $\boldsymbol{w}$ to predict whether a new image is of a cat or a dog. However, some of the images in the dataset are mislabeled (i.e., some of the images labeled as "cat" are actually dogs, and vice versa). Let's assume that the mislabeling is random, such that each image is mislabeled with probability p, independently of all other images.

   (a) Write the expression for the likelihood function for this problem. (3 marks)

   (b) Assuming a suitable prior that corresponds to an $L2$ regularizer, write the expression for the posterior distribution of $\boldsymbol{w}$. You only need to write it up to a proportionality constant. (2 marks)

   (c) Given the posterior distribution, how would you compute the probability of correctly classifying a new image? How does this probability depend on the mislabeling probability p? (3 marks)

(a) Let's consider the probability/likelihood w.r.t. the label y_n for a single image n.

If the label is correct (i.e., $y_n = 1$ and the image is actually a cat, or $y_n = 0$ and the image is actually a dog), then the probability of the label is simply the predicted probability of the image being a cat. Let's denote it by $f(\boldsymbol{x}_n; \boldsymbol{w})$ (it's just the sigmoid function in case of logistic regression).

If the label is incorrect, (i.e., $y_n = 1$ and the image is actually a dog, or $y_n = 0$ and the image is actually a cat), the probability of which is p, then the probability of the label is 1 minus the predicted probability of the image being a cat, i.e., $1 - f(\boldsymbol{x}_n; \boldsymbol{w})$.

Therefore, we have the likelihood w.r.t. each observation as:

$$p(y_n|\boldsymbol{x}_n, \boldsymbol{w}, p) = (1-p) \times f(\boldsymbol{x}_n; \boldsymbol{w})^{y_n}(1 - f(\boldsymbol{x}_n; \boldsymbol{w}))^{(1-y_n)} + p \times (1 - f(\boldsymbol{x}_n; \boldsymbol{w}))^{y_n} f(\boldsymbol{x}_n; \boldsymbol{w})^{(1-y_n)}$$

where the first term corresponds to the probability of a correct label, and the second term corresponds to the probability of an incorrect label. The overall likelihood $p(\boldsymbol{y}|\mathbf{X}, \boldsymbol{w}, p)$ will be a product of these individual probabilities.

(b) The suitable prior that corresponds to an $L2$ regularizer will be $p(\boldsymbol{w}) = \mathcal{N}(\boldsymbol{w}|0, \sigma^2\mathbf{I})$. The posterior will be $p(\boldsymbol{w}|\mathbf{X}, \boldsymbol{y}, p) \propto p(\boldsymbol{w})p(\boldsymbol{y}|\mathbf{X}, \boldsymbol{w}, p)$.

(c) The probability of correctly classifying a new image can be computed as the expected value of the correct classification indicator function over the posterior distribution of the model parameters: $p(correct|\boldsymbol{x}_{new}, \boldsymbol{y}, \mathbf{X}, p) = \int p(correct|\boldsymbol{x}_{new}, \boldsymbol{w})p(\boldsymbol{w}|\mathbf{X}, \boldsymbol{y}, p)d\boldsymbol{w}$, where $p(correct|\boldsymbol{x}_{new}, \boldsymbol{w})$ is 1 if the predicted label for $\boldsymbol{x}_{new}$ is correct (i.e., $\text{argmax}_y f(\boldsymbol{x}_{new}; \boldsymbol{w}) = y_{new}$), and 0 otherwise.

Note that $p(correct|\boldsymbol{x}_{new}, \boldsymbol{w})$ depends only on the predicted probability $f(\boldsymbol{x}_{new}; \boldsymbol{w})$ and the true label y_{new} (which is unknown), but not on the mislabeling probability p. Therefore, we can compute this probability for a range of values of p and plot the results to see how the mislabeling probability affects the model's performance.

4. Consider a linear regression model $\boldsymbol{y} = \mathbf{X}\boldsymbol{w} + \boldsymbol{\epsilon}$ with $\boldsymbol{y} = [y_1, \ldots, y_N]^\top$ is the $N \times 1$ response vector, $\mathbf{X}$ is the $N \times D$ feature matrix, and $\boldsymbol{\epsilon} = [\epsilon_1, \ldots, \epsilon_N]^\top$ is the $N \times 1$ vector of i.i.d. Gaussian noise $\mathcal{N}(0, \sigma^2)$. Let us assume the following prior on each entry of the weight vector $\boldsymbol{w} \in \mathbb{R}^D$

$$p(w_d|\sigma, \gamma_d) = \begin{cases} \mathcal{N}(0, \sigma^2 v_0), & \text{if } \gamma_d = 0 \\ \mathcal{N}(0, \sigma^2 v_1), & \text{if } \gamma_d = 1 \end{cases}$$

where $v_1 \gg v_0 > 0$. Further assume the priors $p(\gamma_d) = \text{Bernoulli}(\theta)$, $d = 1, \ldots, D$, $p(\theta) = \text{Beta}(a_0, b_0)$, and $p(\sigma^2) = \text{IG}(\nu/2, \nu\lambda/2)$, where IG denotes the inverse-gamma prior in its shape-scale paramerization. Note that theprior on w_d can also be written as $p(w_d|\sigma, \gamma_d) = \mathcal{N}(0, \sigma^2 \kappa_{\gamma_d})$ with $\kappa_{\gamma_d} = \gamma_d v_1 + (1 - \gamma_d) v_0$.

- What is the effect of assuming the above prior on $\boldsymbol{w}$ (4 marks)?
- Derive an EM algorithm for this model. Your algorithm should give the posterior over the weight vector $\boldsymbol{w}$ and point estimates (MAP) for the remaining unknowns $\boldsymbol{\gamma}, \sigma^2, \theta$ (12 marks).

Effect of assuming the given prior

Since we set different variances on different components of w in practice this would mean having a component wise regularisation. Also when γ_d is 0 the variance is $\sigma^2\nu_0$ and is $\sigma^2\nu_1$ when γ_d is 1. Since $\nu_1 > \nu_0$ this would mean that the variance in the latter case is larger and hence this would promote w to take a wider range of values whereas in the former case w would be forced to take values in a narrower range. Tuning these hyperparams ν_0, ν_1 can promote learning sparse w.

The (conditional) posterior of $\boldsymbol{w}$ can be computed as

$$p(w|y, X, \Theta) = \frac{p(y, w|X, \Theta)}{p(y|X, \Theta)} = \frac{p(y|w, X, \Theta)p(w|\Theta)}{p(y|X, \Theta)} \propto p(y|w, X, \Theta)p(w|\Theta)$$

Also note that$p(w|\Theta) = \prod_{d=1}^{D} \mathcal{N}(w_d|0, \sigma^2 k_{\gamma_d}) = \mathcal{N}(w|0, \Sigma), \Sigma = \text{Diag}(\sigma^2 k_{\gamma_1}, ..., \sigma^2 k_{\gamma_d}) = \sigma^2 K$ We clearly have a standard regression setting, where we need to find the posterior distribution of the weights. Using results directly we have

$$p(w|y, X, \Theta) = \mathcal{N}(w|\mu_N, \Sigma_N), \Sigma_N = \left(\Sigma^{-1} + \frac{X^\intercal X}{\sigma^2}\right)^{-1}, \mu_N = \frac{\Sigma_N X^\intercal y}{\sigma^2}$$

Since $\Sigma = \sigma^2 K$ we have

$$\Sigma_N = \sigma^2 (K^{-1} + X^\intercal X)^{-1}, \mu_N = (K^{-1} + X^\intercal X)^{-1} X^\intercal y$$

The expected CLL (to be maximized in the M step) for the model

$$\begin{aligned}
\Psi &= \mathbb{E}[\log p(y|X, w, \Theta) + \log p(w|\Theta)] \\
&= \frac{-1}{2\sigma^2}\mathbb{E}[(y - Xw)^\intercal (y - Xw)] - \frac{1}{2}\mathbb{E}[w^\intercal \Sigma^{-1} w] - N \log \sigma - \frac{\log|\Sigma|}{2} \\
&= \mathbb{E}\left\{ -w^\intercal \left(\frac{X^\intercal X}{2\sigma^2} + \frac{\Sigma^{-1}}{2}\right) w + \frac{w^\intercal X^\intercal y}{\sigma^2} \right\} - N \log \sigma - \frac{\log|\Sigma|}{2} - \frac{1}{2\sigma^2} y^\intercal y \\
&= -\frac{1}{2}\text{Tr}\left\{ \left(\frac{X^\intercal X}{\sigma^2} + \Sigma^{-1}\right) \Sigma_N \right\} - \frac{1}{2}\mu_N^\intercal \left(\frac{X^\intercal X}{\sigma^2} + \Sigma^{-1}\right) \mu_N + \frac{\mu_N^\intercal X^\intercal y}{\sigma^2} - N \log \sigma - \frac{\log|\Sigma|}{2} - \frac{1}{2\sigma^2} y^\intercal y
\end{aligned}$$

Denoting the parameters as $\Theta = (\gamma, \sigma^2, \theta)$, the M step optimization problem and the solutions (below, t denotes the iteration index, so $t - 1$ means we are using the previous/most recent values of the given parameters)

Since we are looking for MAP updates the M step requires us to solve the optimsiation problem given as -

$$\mathcal{L} = \arg\max_{\Theta} \psi + \log p(\Theta)$$

where we have the prior distribution on the parameters as -

$$p(w, \theta, \gamma) = \prod_{d=1}^{D} \text{Bernoulli}(\gamma_d|\theta)\text{Beta}(\theta|a_0, b_0)\text{IG}(\sigma^2|\nu/2, \nu\lambda/2)$$

Ignoring constants $\log p(\Theta)$ can be expressed as -

$$\log p(\Theta) = \left(\sum_{d=1}^{D} \gamma_d + a_0 - 1\right)\log\theta + \left(D - \sum_{d=1}^{D}\gamma_d + b_0 - 1\right)\log(1-\theta) - (\nu+2)\log\sigma - \frac{\nu\lambda}{2\sigma^2}$$

Solving the optimisation problem w.r.t $\sigma^2 = \phi$ the relevant part of the objective that we wish to optimise is the following

$$\mathcal{L}_{\sigma^2} = -\frac{1}{2}\text{Tr}\left\{\left(\frac{X^\intercal X}{\phi} + \frac{K^{-1}}{\phi}\right)\Sigma_{t-1}^{N}\right\} - \frac{1}{2}\mu_{t-1}^\intercal\left(\frac{X^\intercal X}{\phi} + \frac{K^{-1}}{\phi}\right)\mu_{t-1} + \frac{\mu_{t-1}^\intercal X^\intercal y}{\phi}$$
$$- \frac{\nu+2}{2}\log\phi - \frac{\lambda\nu}{2\phi} - \frac{y^\intercal y}{2\phi} - \frac{N}{2}\log\phi - \frac{D}{2}\log\phi$$

Where we have used $|\Sigma| = (\sigma^2)^D \prod_{d=1}^{D} k_{\gamma_d}$ to express $\log|\Sigma|$. Setting the gradient to be zero we obtain

$$\sigma^2 = \frac{y^\intercal y + \nu\lambda - 2\mu_{t-1}^\intercal X^\intercal y + \mu_{t-1}^\intercal(X^\intercal X + K^{-1})\mu_{t-1} + \text{Tr}\{(X^\intercal X + K^{-1})\Sigma_{t-1}^{N}\}}{\nu + D + N + 2}$$

For θ we need to optimise the following function -

$$\mathcal{L}_\theta = \left(\sum_{d=1}^{D}\gamma_d + a_0 - 1\right)\log\theta + \left(D - \sum_{d=1}^{D}\gamma_d + b_0 - 1\right)\log(1-\theta)$$

Setting the gradient w.r.t θ to be zero we obtain -

$$\theta = \frac{\sum_{d=1}^{D}\gamma_d + a_0 - 1}{D + a_0 + b_0 - 2}$$

For γ_d we need to solve the optimisation problem given by -

$$\gamma_d = \arg\max_{\gamma_d \in \{0,1\}} -\frac{1}{2}\left(\frac{(\mu_{t-1}^{d})^2}{\sigma^2} + \frac{(\Sigma_{t-1}^{N})^{dd}}{\sigma^2}\right)\left(\frac{1}{\gamma_d\nu_1 + (1-\gamma_d)\nu_0}\right) + \gamma_d\log\left(\frac{\theta}{1-\theta}\right) - \frac{1}{2}\log\left(\gamma_d\nu_1 + (1-\gamma_d)\nu_0\right)$$

Some distributions and their properties:

- For $x \in (0,1)$, Beta$(x|a,b) = \frac{x^{a-1}(1-x)^{b-1}}{B(a,b)}$, where $B(a,b) = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}$ and Γ denotes the gamma function s.t. $\Gamma(x) = (x-1)!$ for a positive integer x. Expectation of a Beta r.v.: $\mathbb{E}[x] = \frac{a}{a+b}$.
- For $x \in \{0,1,2,\ldots\}$ (non-negative integers), Poisson$(x|\lambda) = \frac{\lambda^x \exp(-\lambda)}{x!}$ where λ is the rate parameter.
- For $x \in \mathbb{R}_+$, Gamma$(x|a,b) = \frac{b^a}{\Gamma(a)} x^{a-1} \exp(-bx)$ (shape and rate parameterization), and Gamma$(x|a,b) = \frac{1}{\Gamma(a) b^a} x^{a-1} \exp(-\frac{x}{b})$ (shape and scale parameterization)
- For $x \in \mathbb{R}$, Univariate Gaussian: $\mathcal{N}(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\{-\frac{(x-\mu)^2}{2\sigma^2}\}$
- For $x \in \mathbb{R}^D$, D-dimensional Gaussian: $\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^D|\boldsymbol{\Sigma}|}} \exp\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\}$.
  Trace-based representation: $\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^D|\boldsymbol{\Sigma}|}} \exp\Big\{ -\frac{1}{2}\text{trace}\left[\boldsymbol{\Sigma^{-1}}\mathbf{S}\right]\Big\}$, $\mathbf{S} = (\boldsymbol{x}-\boldsymbol{\mu})(\boldsymbol{x}-\boldsymbol{\mu})^\top$.
  Information form: $\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) = (2\pi)^{-D/2}|\boldsymbol{\Lambda}|^{1/2} \exp\left[-\frac{1}{2}\left(\boldsymbol{x}^\top\boldsymbol{\Lambda}\boldsymbol{x} + \boldsymbol{\xi}^\top\boldsymbol{\Lambda}^{-1}\boldsymbol{\xi} - 2\boldsymbol{x}^\top\boldsymbol{\xi}\right)\right]$ where $\boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1}$ and $\boldsymbol{\xi} = \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}$
- For $\boldsymbol{\pi} = [\pi_1,\ldots,\pi_K]$, s.t. $\sum_{k=1}^K \pi_k = 1$, Dirichlet$(\boldsymbol{\pi}|\alpha_1,\ldots,\alpha_K) = \frac{1}{B(\alpha_1,\ldots,\alpha_K)} \prod_{k=1}^K \pi_k^{\alpha_k-1}$
  where $B(\alpha_1,\ldots,\alpha_K) = \frac{\prod_{k=1}^K \Gamma(\alpha_k)}{\Gamma(\sum_{k=1}^K \alpha_k)}$, and $\mathbb{E}[\pi_k] = \frac{\alpha_k}{\sum_{k=1}^K \alpha_k}$
- For $x_k \in \{0, N\}$ and $\sum_{k=1}^K x_k = N$, multinomial$(x_1,\ldots,x_K|N,\boldsymbol{\pi}) = \frac{N!}{\boldsymbol{x}_1!\ldots,\boldsymbol{x}_K!}\pi_1^{x_1}\ldots\pi_K^{x_K}$
  where $\boldsymbol{\pi} = [\pi_1,\ldots,\pi_K]$, s.t. $\sum_{k=1}^K \pi_k = 1$. The multinoulli is the same as multinomial with $N = 1$.

Some other useful results:

- If $\boldsymbol{x} = \mathbf{A}\boldsymbol{z} + \boldsymbol{b} + \epsilon$, $p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Lambda}^{-1})$, $p(\boldsymbol{\epsilon}) = \mathcal{N}(\boldsymbol{\epsilon}|\mathbf{0},\mathbf{L}^{-1})$ then $p(\boldsymbol{x}|\boldsymbol{z}) = \mathcal{N}(\boldsymbol{x}|\mathbf{A}\boldsymbol{z}+\boldsymbol{b},\mathbf{L}^{-1})$,
  $p(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}|\mathbf{A}\boldsymbol{\mu}+\boldsymbol{b},\mathbf{A}\boldsymbol{\Lambda}^{-1}\mathbf{A}^\top+\mathbf{L}^{-1})$, and $p(\boldsymbol{z}|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{z}|\boldsymbol{\Sigma}\left\{\mathbf{A}^\top\mathbf{L}(\boldsymbol{x}-\boldsymbol{b})+\boldsymbol{\Lambda}\boldsymbol{\mu}\right\},\boldsymbol{\Sigma})$,
  where $\boldsymbol{\Sigma} = (\boldsymbol{\Lambda}+\mathbf{A}^\top\mathbf{L}\mathbf{A})^{-1}$.
- Marginal and conditional distributions for Gaussians: $p(\boldsymbol{x}_a) = \mathcal{N}(\boldsymbol{x}_a|\boldsymbol{\mu}_a,\boldsymbol{\Sigma}_{aa})$,
  $p(\boldsymbol{x}_a|\boldsymbol{x}_b) = \mathcal{N}(\boldsymbol{x}_a|\boldsymbol{\mu}_{a|b},\boldsymbol{\Sigma}_{a|b})$ where $\boldsymbol{\Sigma}_{a|b} = \boldsymbol{\Lambda}_{aa}^{-1} = \boldsymbol{\Sigma}_{aa} - \boldsymbol{\Sigma}_{ab}\boldsymbol{\Sigma}_{bb}^{-1}\boldsymbol{\Sigma}_{ba}$, $\boldsymbol{\mu}_{a|b} = \boldsymbol{\Sigma}_{a|b}\left\{\boldsymbol{\Lambda}_{aa}\boldsymbol{\mu}_a - \boldsymbol{\Lambda}_{ab}(\boldsymbol{x}_b-\boldsymbol{\mu}_b)\right\} =$
  $\boldsymbol{\mu}_a - \boldsymbol{\Lambda}_{aa}^{-1}\boldsymbol{\Lambda}_{ab}(\boldsymbol{x}_b-\boldsymbol{\mu}_b) = \boldsymbol{\mu}_a + \boldsymbol{\Sigma}_{ab}\boldsymbol{\Sigma}_{bb}^{-1}(\boldsymbol{x}_b-\boldsymbol{\mu}_b)$, where symbols have their usual meaning. :)
- $\frac{\partial}{\partial\boldsymbol{\mu}}[\boldsymbol{\mu}^\top\mathbf{A}\boldsymbol{\mu}] = [\mathbf{A}+\mathbf{A}^\top]\boldsymbol{\mu}$, $\frac{\partial}{\partial\mathbf{A}}\log|\mathbf{A}| = \mathbf{A}^{-\top}$, $\frac{\partial}{\partial\mathbf{A}}\text{trace}[\mathbf{A}\mathbf{B}] = \mathbf{B}^\top$
- For a random variable vector $\boldsymbol{x}$, $\mathbb{E}[\boldsymbol{x}\boldsymbol{x}^\top] = \mathbb{E}[\boldsymbol{x}]\mathbb{E}[\boldsymbol{x}]^\top + \text{cov}[\boldsymbol{x}]$