Topics in Probabilistic Modeling & Inference (CS698X), Spring 2019
Indian Institute of Technology Kanpur
Homework Assignment Number 2

QUESTION 1

Student Name: Abhas Kumar
Roll Number: 20111001
Date: April 20, 2021

Part A:

They wanted to chose a batch $D^{'}$ such that updated log posterior i.e $\log p(\boldsymbol{\theta}|D_0 \cup D^{'})$ best approximates the $\log p(\boldsymbol{\theta}|D_0 \cup D_p)$ which is the complete data log posterior.

The key equation 4, $E[\log p(\boldsymbol{\theta}|D_0 \cup (\mathcal{X}_p, \mathcal{Y}_p))] = log\ E_{\mathcal{Y}_p}[log\ p(\boldsymbol{\theta}|D_0) + log\ p(\mathcal{Y}_p|\mathcal{X}_p, \boldsymbol{\theta}) - log\ p(\mathcal{Y}_p|\mathcal{X}_p, D_0)]$
$= log\ p(\boldsymbol{\theta}|D_0) + E_{\mathcal{Y}_p}[log\ p(\mathcal{Y}_p|\mathcal{X}_p, \boldsymbol{\theta})] + H[\mathcal{Y}_p|\mathcal{X}_p, D_0]$
$= log\ p(\boldsymbol{\theta}|D_0) + \sum_{m=1}^{M} L_m(\boldsymbol{\theta})$
where, $L_m(\boldsymbol{\theta}) = E_{\mathbf{y}_m}[log\ p(\mathbf{y}_m|\mathbf{x}_m, \boldsymbol{\theta})] + H[\mathbf{y}_m|\mathbf{x}_m, D_0]$

Since, the term $log\ p(\boldsymbol{\theta}|D_0)$ only depends on D_0, choosing the batch that best approximates $\sum_{m=1}^{M} L_m(\boldsymbol{\theta})$, was enough, they considerd $\mathbf{w} \in \{0,1\}^M$, a weight vector, where a point will be chosen if $w = 1$, $L(\mathbf{w}) = \sum_m w_m L_m$ got converted to a sparse subset approximation problem.

$\mathbf{w}^* = \min_{\mathbf{w}} ||L - L(\mathbf{w})||^2$ such that $w_m \in \{0,1\}\ \forall m$, $\sum_m 1 \leq b$, where $L = \sum_m L_m$

Part B:

Since, the sparse approximation based objective was difficult to optimise, they proposed to construct batches in a Hilbert space induced by inner product $< \mathcal{L}_n, \mathcal{L}_m >$. They tried to relax the binary weight constraint to be non-negative and replaced the cordinality constraint with a polytope constraint, where $\sigma = \sum_m \sigma_m$ and $\sigma_m = ||\mathcal{L}_m||$ and $\mathbf{K} \in \mathcal{R}^{MXM}$. Kernel matrix with $K_{mn} =< \mathcal{L}_m, \mathcal{L}_n >$. So, the optimaisation problem became

$$\min_{\mathbf{w}} \left\{ (1 - \mathbf{w})^T \mathbf{K} (1 - \mathbf{w}) \right\} \text{ such that } w_m \geq 0, \text{ and } \textstyle\sum_m w_m \sigma_m = \sigma,$$

which can be solved by using the Frank-wolfe algorithm. Main computation in that was,

$$< \mathcal{L} - \mathcal{L}(\mathbf{w}), \frac{1}{\sigma_n} \mathcal{L}_n >= \frac{1}{\sigma_n} \sum_{m=1}^{M} (1 - w_m) < \mathcal{L}_m, \mathcal{L}_n >$$

At each iteration, the proposed algorithm greedily selected the $\mathcal{L}_f$ vector, most aligned with residual error $\mathcal{L} - \mathcal{L}(\mathbf{w})$. Since, the algorithm allowed to select indices from previous iteration, the resulting weight vector had $\leq b$ non-zero entries. Finally, they projected weights back to feasible space by setting $w_m = 1$ if $w_m > 0$ otherwise 0. They chosed weighed inner products of the form $< \mathcal{L}_n, \mathcal{L}_m >_{\hat{\pi}} = E_{\hat{\pi}}[< \mathcal{L}_n, \mathcal{L}_m >]$, where $\hat{\pi}$ was the current posterior $p(\boldsymbol{\theta}|D_0)$.

$< \mathcal{L}_n, \mathcal{L}_m >_{\hat{\pi},\mathcal{F}} = E_{\hat{\pi}}[\Delta_{\boldsymbol{\theta}} \mathcal{L}_n(\boldsymbol{\theta})^T \Delta_{m(\boldsymbol{\theta})}]$ and also $< \mathcal{L}_n, \mathcal{L}_m >_{\hat{\pi},2} = E_{\hat{\pi}}[\mathcal{L}_n(\boldsymbol{\theta}) \mathcal{L}_m(\boldsymbol{\theta})]$
The advantage of later inner product is that it only required tractable likelihood computations.

Part C:

For **Bayesian linear Regression** and **Probit Regression**, the acquisition functions proposed in the paper had a closed form expression.
For other types of model where the acquisition function wasn't available, they used random feature projection, to approximate key quantities. They considered models in which expectation of $L_n(\boldsymbol{\theta})$ w.r.t $p(y_n|x_n, D_0)$ was tractable. They considered projections for the weighted Euclidean inner product form $(L_n, L_m)_{\hat{\pi},2} = \mathcal{E}_{\hat{\pi}}[L_n(\boldsymbol{\theta})L_m(\boldsymbol{\theta})]$. This projection was

$$\hat{L_n} = \frac{1}{\sqrt{J}}[L_(\boldsymbol{\theta}_1), L_(\boldsymbol{\theta}_2), ..., L_(\boldsymbol{\theta}_j)]^T, \boldsymbol{\theta}_j \sim \hat{\pi}$$

$\hat{L_n}$ represented the J-dimension projection of L_n in Euclidean space. With this projection they approximated linear products as dot products.$\langle L_n, L_m\rangle_{\hat{\pi},2} \approx \hat{L_n}^T \hat{L_m}$, where $\hat{L_n}^T \hat{L_m}$ was an unbiased sample estimator of $\langle L_n, L_m\rangle_{\hat{\pi},2}$ using J Monte-Carlo simulation from the posterior $\hat{\pi}$.

Topics in Probabilistic Modeling & Inference (CS698X), Spring 2019
Indian Institute of Technology Kanpur
Homework Assignment Number 2

QUESTION 2

Student Name: Abhas Kumar
Roll Number: 20111001
Date: April 20, 2021

Given N scalar observations $x_1, x_2, ..., x_N$ drawn iid from $\mathcal{N}(\mathbf{x}|\mu, \beta^{-1})$ with prior $\mathcal{N}(\mu|\mu_0, s_0)$ where β has a gamma prior $Gamma(\beta|a, b)$, we have ,

$$p(\mu|x, \beta^{-1}) = \frac{p(x|\mu, \beta^{-1}) \times p(\mu)}{\int p(x|\mu, \beta^{-1}) \times p(\mu) d\mu}$$

where $p(\mu) = \mathcal{N}(\mu|\mu_0, s_0)$ and $p(\beta) = Gamma(\beta|a, b)$

Since both $p(x|\mu, \beta)$ and $p(\mu)$ are Gaussian , $p(\mu|x, \beta)$ will also be Gaussian distribution due to conjugacy.

$$P(\mu|x, \beta^{-1}) = \prod_{n=1}^{N} \mathcal{N}(x|\mu, \beta^{-1}) \mathcal{N}(\mu|\mu_0, s_0) = \mathcal{N}(x|\mu_N, \sigma_N^2)$$

where, $\mu_N = \frac{1}{Ns_0 + \beta^{-1}} \left(\beta^{-1}\mu_0 + Ns_0 \right)$

and $\sigma_N^2 = s_0^{-1} + N\beta$

Again we have ,

$$p(\beta|x, \mu) = \frac{p(x|\beta, \mu) \times p(\beta)}{\int p(x|\beta, \mu) \times p(\beta) d\beta}$$

Dince, the Gaussian likelihood and Gamma prior are conjugate to each other,resulting posterior will also be a Gamma distriution.

$$P(\beta|x, \mu) = \prod_{n=1}^{N} \mathcal{N}(x|\mu, \beta^{-1}) Gamma(\beta|a, b) = Gamma(\beta|a_N, b_N)$$

where $a_N = a + \frac{N}{2}$ and $b_N = b + \frac{\sum_{n=1}^{N}(x_n - \mu)^2}{2}$

Using above conditional posteriors in a Gibbs sampling algorithm to approximate the joint posterior μ and β

Gibbs Sampling:

```
1. Initialise $\beta = \beta^0$
2. for s=1, 2, 3,..., S
        3. Draw $\mu_s \sim p(\mu|x, \beta_{s-1})$ i.e $\mathcal{N}(x|u_N, (\sigma_N^2)_{s-1})$
        4 .Draw $\beta_s \sim p(\beta|x, \mu_s)$ i.e $Gamma\left( a + \frac{N}{2}, b + \frac{\sum_{n=1}^{N}(x_n - \mu_s)^2}{2} \right)$
$\{\mu, \sigma^2\}_{s=1}^{S}$ approximates the joint posterior $\mu$ and $\beta$.
```

Topics in Probabilistic Modeling & Inference (CS698X), Spring 2019
Indian Institute of Technology Kanpur
Homework Assignment Number 2

Student Name: Abhas Kumar
Roll Number: 20111001
Date: April 20, 2021

QUESTION 3

What is the effect of assuming the above prior on w?

The effect of prior is to have **sparse learning** for the weights. The prior used for weights is an example of "mixture of Gaussians" which will correspond to L_1 regularization and hence will learn $\mathbf{w}$ as a sparse vector. Also, sparsity is induced one for which the precision is high, and one for which it is lower. The prior classifies the parameters into 2 categories based on their importance and contribution to the regression model.

Deriving an EM algorithm for doing inference for this model

Given likelihood and prior are both Gaussian $p(\mathbf{y}|\mathbf{X},\mathbf{w},\sigma^2) = \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{w},\sigma^2 I_N)$ and $p(\mathbf{w}|\sigma^2,\gamma) = \mathcal{N}(\mathbf{w}|0,\sigma^2\mathbf{K})$ where $\mathbf{K} = diag(\kappa_{\gamma_1},\kappa_{\gamma_2}\dots\kappa_{\gamma_D})$ is a diagonal covariance matrix.

We know, $p(\mathbf{w}|\mathbf{y},\mathbf{X},\sigma^2,\gamma) \propto p(\mathbf{y}|\mathbf{X},\mathbf{w},\sigma^2)p(\mathbf{w}|\sigma^2,\gamma)$ so using Completing the Squares trick to find the posterior, we get,

$$p(\mathbf{w}|\mathbf{y},\mathbf{X},\sigma^2,\gamma) = \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}_w,\boldsymbol{\Sigma}_w)$$

where

$$\boldsymbol{\Sigma}_w = \sigma^2(\mathbf{X}^T\mathbf{X}+\mathbf{K}^{-1})^{-1}$$
$$\boldsymbol{\mu}_w = \frac{1}{\sigma^2}\boldsymbol{\Sigma}_w\mathbf{X}^T\mathbf{y}$$

The Complete Data Log-Likelihood(CLL),

$$\log p(\mathbf{w},\mathbf{y}|\mathbf{X},\sigma^2,\gamma) = \log p(\mathbf{y}|\mathbf{X},\mathbf{w},\sigma^2) + \log p(\mathbf{w}|\sigma^2,\gamma)$$

$$= -\frac{1}{2\sigma^2}(\mathbf{y}-\mathbf{X}\mathbf{w})^T(\mathbf{y}-\mathbf{X}\mathbf{w}) - \frac{N+D}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\mathbf{w}^T\mathbf{K}^{-1}\mathbf{w} - \sum_{d=1}^{D}\frac{1}{2}\log(\kappa_{\gamma_d})$$

Expectation of the above obtained CLL i.e $\mathbf{E}[CLL] = \mathbf{E}_{\mathbf{w}|\mathbf{y}}[\log p(\mathbf{w},\mathbf{y}|\mathbf{X},\sigma^2,\gamma)]$

$$= -\frac{1}{2\sigma^2}\left(\mathbf{y}^T\mathbf{y} - 2\mathbf{y}^T\mathbf{X}E[\mathbf{w}] + trace((\mathbf{X}^T\mathbf{X}+\mathbf{K}^{-1})E[\mathbf{w}\mathbf{w}^T])\right) - \frac{N+D}{2}\log(2\pi\sigma^2) - \frac{1}{2}\sum_{d=1}^{D}\log(\kappa_{\gamma_d})$$

where, $E[\mathbf{w}] = \boldsymbol{\mu}_w = (\mathbf{X}^T\mathbf{X}+\mathbf{K}^{-1})^{-1}\mathbf{X}^T\mathbf{y}$ and $E[\mathbf{w}\mathbf{w}^T] = \boldsymbol{\Sigma}_w + \boldsymbol{\mu}_w{\boldsymbol{\mu}_w}^T = \sigma^2(\mathbf{X}^T\mathbf{X}+\mathbf{K}^{-1})^{-1} + \mathbf{y}^T\mathbf{X}(\mathbf{X}^T\mathbf{X}+\mathbf{K}^{-1})^{-2}\mathbf{X}^T\mathbf{y}$

To estimate MAP, we need to find the posterior with respect to $E[CLL]$. i.e we need to find

$$\arg\max_{\sigma^2,\boldsymbol{\gamma},\theta}\left\{E[CLL]+\log p(\sigma^2,\boldsymbol{\gamma},\theta)\right\}$$

The prior of the parameters is as follows

$$p(\sigma^2,\boldsymbol{\gamma},\theta)=p(\sigma^2)\prod_{d=1}^{D}p(\gamma_d|\theta)p(\theta)$$

$$\log p(\sigma^2,\boldsymbol{\gamma},\theta)=\log p(\sigma^2)+\sum_{d=1}^{D}\log p(\gamma_d|\theta)+\log p(\theta)$$

Given,

$$\log p(\sigma^2)=-\left(\frac{\nu}{2}+1\right)\log\sigma^2-\frac{\nu\lambda}{2\sigma^2}+c$$

$$\log p(\gamma_d|\theta)=\gamma_d\log\theta+(1-\gamma_d)\log(1-\theta)$$

$$\log p(\theta)=(a_0-1)\log\theta+(b_0-1)\log(1-\theta)$$

MAP estimation for σ^2 :

$$\frac{\partial\Big(E[CLL]+\log p(\sigma^2,\boldsymbol{\gamma},\theta)\Big)}{\partial\sigma^2}=0$$

$$\implies \frac{1}{2\sigma^4}\Big(\mathbf{y}^T\mathbf{y}-2\mathbf{y}^T\mathbf{X}E[\mathbf{w}]+trace\big((\mathbf{X}^T\mathbf{X}+\mathbf{K}^{-1})E[\mathbf{w}\mathbf{w}^T]\big)\Big)-\frac{N+D}{2\sigma^2}-\frac{1}{\sigma^2}\big(\frac{\nu}{2}+1\big)+\frac{\nu\lambda}{2\sigma^4}=0$$

$$\therefore \sigma^2=\frac{\mathbf{y}^T\mathbf{y}-2\mathbf{y}^T\mathbf{X}E[\mathbf{w}]+trace\big((\mathbf{X}^T\mathbf{X}+\mathbf{K}^{-1})E[\mathbf{w}\mathbf{w}^T]\big)+\nu\lambda}{N+D+\nu+2}$$

MAP estimation for γ_d:

Since $\gamma_d\in\{0,1\}$, we can write

$$\gamma_d=\arg\max_{\gamma'_d\in(0,1)}\{E[CLL]+\log p(\sigma^2,\boldsymbol{\gamma},\theta)\}$$

$$\arg\max_{\gamma'_d\in(0,1)}\{-\frac{1}{2\sigma^2\kappa_{\gamma'_d}}E[\mathbf{w}\mathbf{w}^T]_{d,d}-\frac{1}{2}\log(\kappa_{\gamma'_d})+\gamma'_d\log\theta+(1-\gamma'_d)\log(1-\theta)\}$$

MAP estimation for θ

$$\frac{\partial\Big(E[CLL]+\log p(\sigma^2,\boldsymbol{\gamma},\theta)\Big)}{\partial\theta}=0$$

$$\implies \frac{1}{\theta}\Big(\sum_{d=1}^{D}\gamma_d+a_0-1\Big)-\frac{1}{1-\theta}\Big(\sum_{d=1}^{D}(1-\gamma_d)+b_0-1\Big)=0$$

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

EM Algorithm:

Step1: Initialize parameters $\big\{\boldsymbol{\gamma}, \sigma^2, \theta\big\} = \big\{\boldsymbol{\gamma}, \sigma^2, \theta\big\}^0$

Step2: For $t = 1, 2, \dots T$

2.a Update the posterior of $\mathbf{w}$ as

$$p(\mathbf{w}^{(t)}|\mathbf{y}, \mathbf{X}, {\sigma^2}^{(t-1)}, \gamma^{(t-1)}) = \mathcal{N}(\mathbf{w}^{(t-1)}|\boldsymbol{\mu}_w^{(t)}, \boldsymbol{\Sigma}_w^{(t)})$$

$$\boldsymbol{\Sigma}_w^{(t)} = {\sigma^2}^{(t-1)}(\mathbf{X}^T\mathbf{X} + {\mathbf{K}^{(t-1)}}^{-1})^{-1}$$

$$\boldsymbol{\mu}_w^{(t)} = \frac{1}{{\sigma^2}^{(t-1)}}\boldsymbol{\Sigma}_w^{(t)}\mathbf{X}^T\mathbf{y}$$

2.b: Update the expectations

$$E[\mathbf{w}]^{(t)} = \boldsymbol{\mu}_w^{(t)}$$

$$E[\mathbf{w}\mathbf{w}^{(t)}] = \boldsymbol{\Sigma}_w^{(t)} + \boldsymbol{\mu}_w^{(t)}{\boldsymbol{\mu}_w^{(t)}}^T$$

2.c Update the parameters as

$${\sigma^2}^{(t)} = \frac{\mathbf{y}^T\mathbf{y} - 2\mathbf{y}^T\mathbf{X}E[\mathbf{w}^{(t)}] + trace\big(\mathbf{X}^T\mathbf{X} + {\mathbf{K}^{(t-1)}}^{-1}E[\mathbf{w}^T\mathbf{w}]^{(t)}\big) + \nu\lambda}{N + D + \nu + 2}$$

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

$$\gamma_d^{(t)} = \underset{\gamma_d \in (0,1)}{\arg\max}\,\big\{\gamma_d \log\theta^{(t)} + (1-\gamma_d)\log(1-\theta^{(t)})\big\} - \frac{1}{2{\sigma^2}^{(t)}\kappa_{\gamma_d}}E[\mathbf{w}{\mathbf{w}}^T]_{d,d}^{(t)} - \frac{1}{2}\log(\kappa_{\gamma_d})$$

Step3: Return posterior over the weight vector $\mathbf{w}$ i.e $p(\mathbf{w}|\mathbf{y}, \mathbf{X}, {\sigma^2}^{(T)}, \boldsymbol{\gamma}^{(T)})$ and MAP for $\big\{\boldsymbol{\gamma}, \sigma^2, \theta\big\} = \big\{\boldsymbol{\gamma}, \sigma^2, \theta\big\}^{(T)}$

Topics in Probabilistic Modeling & Inference (CS698X), Spring 2019
Indian Institute of Technology Kanpur
Homework Assignment Number 2

Student Name: Abhas Kumar
Roll Number: 20111001
Date: April 20, 2021

QUESTION 4

Part 1:
Given a zero mean GP prior $p(\mathbf{f}) = GP(0, \kappa)$ i.e $p(\mathbf{f}) = \mathcal{N}(0, \mathbf{K})$ where $\mathbf{f} = [f(x_1), ..., f(x_N)]^T$ is an Nx1 vector and $\mathbf{K}$ is the NxN kernel matrix with $\mathbf{K}_{nm} = \kappa(x_n, x_m)$. Assuming a likelihood model $p(y_n|x_n, \mathbf{f}) = \mathcal{N}(y_n|f(x_n), \sigma^2)$, where $\mathbf{f} \sim GP(0, \kappa)$.

The expression for the GP posterior, i.e., $p(\mathbf{f}|\mathbf{y})$ can be obtained as

$$p(\mathbf{f}|\mathbf{y}) \propto p(y_n|x_n, \mathbf{f})p(\mathbf{f}) \propto \prod_{n=1}^{N} \mathcal{N}(y_n|f, \sigma^2)\mathcal{N}(0, \mathbf{K})$$

$$p(\mathbf{f}|\mathbf{y}) \propto exp\left(\frac{-||\mathbf{y} - \mathbf{f}||^2}{2\sigma^2}\right) exp\left(\frac{-\mathbf{f}^T\mathbf{K}^{-1}\mathbf{f}}{2}\right)$$

$$\propto exp\left[-\left\{\frac{(\mathbf{y} - \mathbf{f})^T(\mathbf{y} - \mathbf{f})}{2\sigma^2} + \frac{\mathbf{f}^T\mathbf{K}\mathbf{f}}{2}\right\}\right]$$

$$\propto exp\left[-\left\{\frac{\sigma^{-2}\mathbf{y}^T\mathbf{y} - 2\sigma^{-2}\mathbf{y}^T\mathbf{f} + \sigma^{-2}\mathbf{f}^T\mathbf{f} + \mathbf{f}^T\mathbf{K}^{-1}\mathbf{f}}{2}\right\}\right]$$

$$\propto exp\left[-\left\{\frac{\mathbf{f}^T(\mathbf{K}^{-1} + \sigma^{-2}I)\mathbf{f} - 2\sigma^{-2}\mathbf{y}^T\mathbf{f} + \mathbf{y}^{\mathbf{T}}\sigma^{-2}\mathbf{y}}{2}\right\}\right]$$

The term $\mathbf{y}^T\sigma^{-2}\mathbf{y}$ does not depend on $\mathbf{f}$, it can be ignored. So we get,

$$p(\mathbf{f}|\mathbf{y}) \propto exp\left\{-\frac{\mathbf{f}^T(\mathbf{K}^{-1} + \sigma^{-2}I)\mathbf{f} - 2\sigma^{-2}\mathbf{y}^T\mathbf{f}}{2}\right\}$$

Comparing this with $p(\mathbf{f}|\mathbf{y}) \propto \left\{-\frac{(f-\mu)\Sigma^{-1}(f-\mu)}{2}\right\}$ we have $\Sigma^{-1} = (\mathbf{K}^{-1} + \sigma^2 I)$ i.e

$$\Sigma_N = (\mathbf{K}^{-1} + \sigma^{-2}I)^{-1}$$

Also, $(\mathbf{f} - \mu)\Sigma^{-1}(\mathbf{f} - \mu) = \mathbf{f}^T(\mathbf{K}^{-1} + \sigma^{-2}I)\mathbf{f} - 2\sigma^{-2}\mathbf{y}^T\mathbf{f}$ from which we get

$$\mu_N = \mathbf{K}(\mathbf{K}^{-1} + \sigma^{-2}I)^{-1}\mathbf{y}$$

$$\mu_N = \sigma^{-2}\Sigma_N\mathbf{y}$$

Hence, GP posterior, $p(\mathbf{f}|\mathbf{y}) = \mathcal{N}(\mathbf{f}|\mu_N, \Sigma_N)$

Part 2: Visualizing GP Priors and Posteriors for Regression

Higher l values lead to smoother functions and therefore to coarser approximations of the training data. Lower l values make functions more wiggly with wide uncertainty regions between training data points. From the plots below, we can conclude that on small values of l, the plots of prior and posterior are more wiggly(contains more wiggles) and as the l value increases, the wiggles on the posterior mean and prior gets elongated and smoothens.

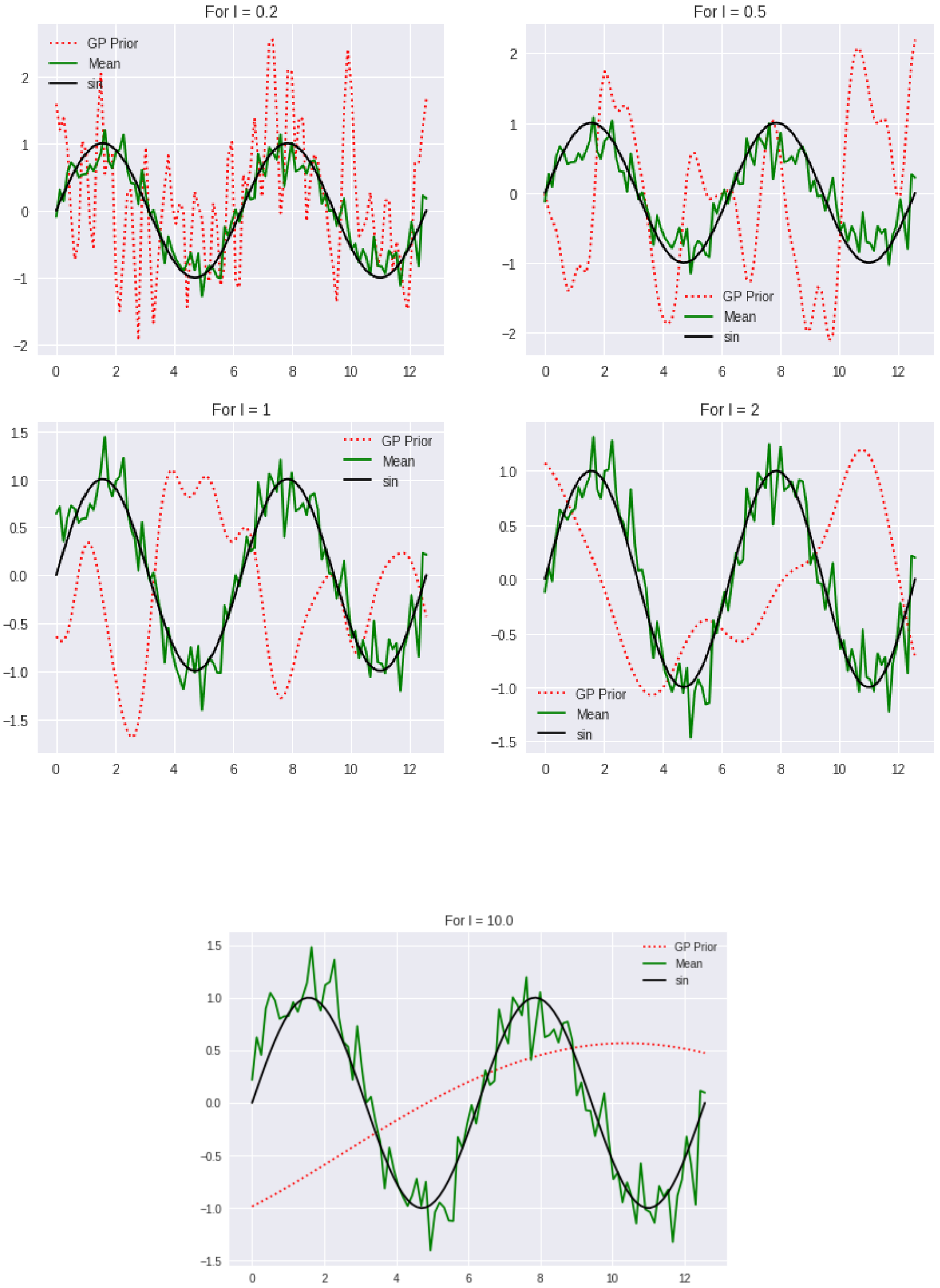

Figure 1: Plots showing random sample from the GP prior $p(\mathbf{f}) = \mathcal{N}(0, \mathbf{K})$, mean of the GP posterior and the true function $\sin(x)$ for $l = 0.2, 0.5, 1, 2, 10$.

Topics in Probabilistic Modeling & Inference (CS698X), Spring 2019
Indian Institute of Technology Kanpur
Homework Assignment Number 2

QUESTION 5

Student Name: Abhas Kumar
Roll Number: 20111001
Date: April 20, 2021

Part 1 : Let $\mathbf{K}$ be the kernel matrix ($N \times N$) for training inputs and $\mathbf{k}_*$ be the $N \times 1$ vector of kernel based similarities of x_* with each of the training inputs. Then, given N training inputs $(\mathbf{X}, \mathbf{f}) = \{x_n, f_n\}_{n=1}^N$, the posterior predictive distribution for a new input $\mathbf{x}_*$ is

$$p(f_*|\mathbf{x}_*, \mathbf{X}, \mathbf{f}) = \mathcal{N}(f_*|\mathbf{k}_*^T\mathbf{K}^{-1}\mathbf{f}, \kappa(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^T\mathbf{K}^{-1}\mathbf{k}_*)$$

It is obvious that this has $\mathcal{O}(n^3)$ complexity as the above expression has a matrix inversion term $\mathbf{K}^{-1}$. To scale this, the problem statement then proposes using pseudo training inputs $\mathbf{Z}$ along with their respective noiseless pseudo output $\mathbf{t}$. With the assumption that the likelihood for each training output f_n to be modeled by a posterior predictive having the same form as the GP regression's posterior predictive but with $(\mathbf{Z}, \mathbf{t})$ acting as "pseudo" training data, we get the following relation :

$$p(f_n|\mathbf{x}_n, \mathbf{Z}, \mathbf{t}) = \mathcal{N}(f_n|\widetilde{\mathbf{k}}_*^T\widetilde{\mathbf{K}}^{-1}\mathbf{f}, \kappa(\mathbf{x}_n, \mathbf{x}_n) - \widetilde{\mathbf{k}}_n^T\widetilde{\mathbf{K}}^{-1}\widetilde{\mathbf{k}}_n).$$

Here, $\widetilde{\mathbf{K}}$ is the $M \times M$ kernel matrix of the pseudo inputs $\mathbf{Z}$ and $\widetilde{\mathbf{k}}_n$ is the $M \times 1$ vector of kernel based similarities of x_n with each of the pseudo inputs. Now, we have,

$$\begin{aligned} p(\mathbf{f}|\mathbf{X}, \mathbf{Z}, \mathbf{t}) &= \prod_{n=1}^N p(f_n|x_n, \mathbf{Z}, \mathbf{t}) \\ &= \mathcal{N}(\mathbf{f}|\mathbf{P}\mathbf{K}_M^{-1}\mathbf{t}, \boldsymbol{\delta}) \end{aligned}$$

In above equation, $(K_M)_{ij} = \kappa(\mathbf{z}_i, \mathbf{z}_j$, $(P)_{ij} = \kappa(\mathbf{x}_i, \mathbf{z}_j)$ and δ is a diagonal matrix with $(\delta)_{ii} = \kappa(\mathbf{x}_i, \mathbf{x}_i) - \mathbf{k}_n^T\mathbf{K}_M^{-1}\mathbf{k}_n$. Moreover, we also have the following relation :

$$p(y_*|\mathbf{x}_*, \mathbf{X}, \mathbf{f}, \mathbf{Z}) = \int p(y_*|\mathbf{x}_*, \mathbf{X}, \mathbf{f}, \mathbf{Z}, \mathbf{t})p(\mathbf{t}|\mathbf{X}, \mathbf{f}, \mathbf{Z})d\mathbf{t}$$

We can now use Baye's rule to obtain an expression for the posterior over $\mathbf{t}$:

$$\begin{aligned} p(\mathbf{t}|\mathbf{X}, \mathbf{f}, \mathbf{Z}) &\propto p(\mathbf{f}|\mathbf{X}, \mathbf{t}, \mathbf{Z})p(\mathbf{t}|\mathbf{Z}) \\ &= \mathcal{N}(\mathbf{t}|\boldsymbol{\mu}_{\mathbf{t}|\mathbf{f}}, \boldsymbol{\Sigma}_{\mathbf{t}|\mathbf{f}}) \end{aligned}$$

This is because the pseudo training points have been modelled by the same GP and hence, $p(\mathbf{t}|\mathbf{Z}) = (\mathbf{t}|0, \mathbf{K}_M)$. Also, $\boldsymbol{\Sigma}_{\mathbf{t}|\mathbf{f}} = (\mathbf{K}_M^{-1}\mathbf{P}^T\boldsymbol{\delta}^{-1}\mathbf{P}\mathbf{K}_M^{-1}$ and $\boldsymbol{\mu}_{\mathbf{t}|\mathbf{f}} = \boldsymbol{\Sigma}_{\mathbf{t}|\mathbf{f}}\mathbf{K}_M^{-1}\mathbf{P}^T\boldsymbol{\delta}^{-1}\mathbf{f}$. As y_* is same as f_*, we can represent $f_* = \mathbf{k}_*^T\mathbf{K}_M^{-1}\mathbf{t} + \mathcal{N}(0, \kappa(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^T\mathbf{K}_M^{-1}\mathbf{k}_*)$. We can now use the property of Linear Gaussian model to get the final expression of posterior predictive as :

$$p(f_*|\mathbf{x}_*, \mathbf{X}, \mathbf{f}, \mathbf{Z}) = \mathcal{N}(f_*|\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_*)$$

Where $\boldsymbol{\mu}_* = \mathbf{k}_*^T\mathbf{K}_M^{-1}\boldsymbol{\Sigma}_{\mathbf{t}|\mathbf{f}}\mathbf{K}_M^{-1}\mathbf{P}^T\boldsymbol{\delta}^{-1}\mathbf{f}$ and $\boldsymbol{\Sigma}_* = \mathbf{k}_*^T\mathbf{K}_M^{-1}\boldsymbol{\Sigma}_{\mathbf{t}|\mathbf{f}}\mathbf{K}_M^{-1}\mathbf{k}_* + \kappa(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^T\mathbf{K}_M^{-1}\mathbf{k}_*$. Calculation of this expression, mainly involves the computation of $\boldsymbol{\Sigma}_{\mathbf{t}|\mathbf{f}}$ term which in turn involves the calculation of $\mathbf{K}_M^{-1}$ term and hence, the overall cost is $\mathcal{O}(NM^2)$ which is a significant improvement over the earlier cost of $\mathcal{O}(N^3)$ as $M << N$.

Part 2 : We know that : $p(\mathbf{f}|\mathbf{X},\mathbf{Z}) = \int p(\mathbf{f}|\mathbf{X},\mathbf{Z},\mathbf{t})p(\mathbf{t}|\mathbf{Z})d\mathbf{t}$, where $\mathbf{f}$ can be written as $\mathbf{P}\mathbf{K}_M^{-1}\mathbf{t} + \mathcal{N}(0,\boldsymbol{\delta})$. Also, $p(\mathbf{f}|\mathbf{X},\mathbf{Z}) = \mathcal{N}(\mathbf{f}|\boldsymbol{\mu},\boldsymbol{\Sigma})$ using linear Gaussian model property. Here, the value of $\boldsymbol{\mu} = 0, \boldsymbol{\Sigma} = \mathbf{P}\mathbf{K}_M^{-1}\mathbf{P}^T + \boldsymbol{\delta}$.

Using above results, we can now write the MLE-II objective function for $\mathbf{Z}$ as follows :

$$\begin{aligned}
\hat{\mathbf{Z}} &= \arg\max_{\mathbf{Z}} p(\mathbf{f}|\mathbf{X},\mathbf{Z}) \\
&= \arg\max_{\mathbf{Z}} \left(-\frac{1}{2} \left(\log|\boldsymbol{\Sigma}| + \mathbf{f}^T\boldsymbol{\Sigma}^{-1}\mathbf{f} \right) \right) \\
&= \arg\min_{\mathbf{Z}} \left(\log|\boldsymbol{\Sigma}| + \mathbf{f}^T\boldsymbol{\Sigma}^{-1}\mathbf{f} \right)
\end{aligned}$$