Inducing Point Methods

A major limiting factor of Gaussian processes (GPs) is the cubic cost associated with inference. Specifically, given n training data points, computing the relevant posterior exactly involves inverting an $n \times n$ matrix (along with other costly matrix operations), which is a $O(n^3)$ operation; this effectively limits GPs to datasets of $\approx 10,000$ data points or fewer. Beyond just the time costs, even storing this inverse requires $O(n^2)$, which can itself be prohibitive for sufficiently large datasets.

There have been many methods proposed to address these computational costs. In these notes, we will focus on one family of such methods, known as *inducing point methods.* Intuitively, these methods seek to "replace" the intractably large dataset of n data points with a smaller, representative set of $m \ll n$ data points. We will introduce these approximations at a high-level; this will also serve to motivate a discussion of *variational inference* in this setting.

The Nyström Approximation

A key question in this setting is how can we define a "good" set of inducing points, $\mathbf{X}_m = \{\tilde{\mathbf{x}}_1, \ldots, \tilde{\mathbf{x}}_m\}$? A reasonable desiderata for these inducing points might be that they contain all the "information" necessary to compute the kernel between arbitrary pairs of points in our domain e.g.,

$$k(\mathbf{x}, \mathbf{x}') \approx \tilde{k}(\mathbf{x}, \mathbf{x}') := k(\mathbf{x}, \mathbf{X}_m) k(\mathbf{X}_m, \mathbf{X}_m)^{-1} k(\mathbf{X}_m, \mathbf{x}').$$

This approximation, known as the *Nyström approximation*, has some interesting theoretical justifications related to minimizing the distance between the true kernel and a linear combination of kernel evaluations against $\mathbf{X}_m$ in "function space"; that discussion is beyond the scope of this course but those who are interested may refer to Section 8.1 of Rasmussen and Williams (2006).

Qualitatively, this approximation asserts that the relationship between $\mathbf{x}$ and $\mathbf{x}'$ is entirely determined by their relationship with the inducing points. Given a zero-mean GP prior, $\mathcal{GP}(0, k)$, and a dataset, $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n = (\mathbf{X}, \mathbf{y})$, we can express the approximate (noisy) posterior as

$$\begin{aligned}
\mu_{\mathcal{D}}(\mathbf{x}^*) &= k(\mathbf{x}^*, \mathbf{X}) \left(k(\mathbf{X}, \mathbf{X}) + \sigma^2 I_n\right)^{-1} \mathbf{y} \approx \tilde{k}(\mathbf{x}^*, \mathbf{X}) \left(\tilde{k}(\mathbf{X}, \mathbf{X}) + \sigma^2 I_n\right)^{-1} \mathbf{y} \\
k_{\mathcal{D}}(\mathbf{x}^*, \mathbf{x}^*) &= k(\mathbf{x}^*, \mathbf{x}^*) - k(\mathbf{x}^*, \mathbf{X}) \left(k(\mathbf{X}, \mathbf{X}) + \sigma^2 I_n\right)^{-1} k(\mathbf{X}, \mathbf{x}^*) \\
&\approx k(\mathbf{x}^*, \mathbf{x}^*) - \tilde{k}(\mathbf{x}^*, \mathbf{X}) \left(\tilde{k}(\mathbf{X}, \mathbf{X}) + \sigma^2 I_n\right)^{-1} \tilde{k}(\mathbf{X}, \mathbf{x}^*)
\end{aligned}$$

Note that this approximation becomes exact when $\mathbf{X}_m = \mathbf{X}$ as

$$\begin{aligned}
\tilde{k}(\mathbf{x}^*, \mathbf{X}) &= k(\mathbf{x}, \mathbf{X}) k(\mathbf{X}, \mathbf{X})^{-1} k(\mathbf{X}, \mathbf{X}) = k(\mathbf{x}, \mathbf{X}) \text{ and} \\
\tilde{k}(\mathbf{X}, \mathbf{X}) &= k(\mathbf{X}, \mathbf{X}) k(\mathbf{X}, \mathbf{X})^{-1} k(\mathbf{X}, \mathbf{X}) = k(\mathbf{X}, \mathbf{X}).
\end{aligned}$$

The astute reader might observe that this approximation, when written out as above, has not actually saved us any computation: computing these approximated posterior terms still requires inverting $\tilde{k}(\mathbf{X}, \mathbf{X}) + \sigma^2 I_n$, an $n \times n$ matrix! Fortunately, we can apply the Woodbury identity which states

$$(ABA^T + C)^{-1} = C^{-1} - C^{-1} A \left(B^{-1} + A^T C^{-1} A\right)^{-1} A^T C^{-1}.$$

This means that in our expression

$$\left(\tilde{k}(\mathbf{X},\mathbf{X})+\sigma^2 I_n\right)^{-1}=\left(k(\mathbf{X},\mathbf{X}_m)k(\mathbf{X}_m,\mathbf{X}_m)^{-1}k(\mathbf{X}_m,\mathbf{X})+\sigma^2 I_n\right)^{-1}$$

the matrix C in the Woodbury identity corresponds to $\sigma^2 I_n$, which is trivial to invert! Thus, we get the final, much more computationally efficient result:

$$\left(\tilde{k}(\mathbf{X},\mathbf{X})+\sigma^2 I_n\right)^{-1}=\frac{1}{\sigma^2}I_n-\frac{1}{\sigma^4}k(\mathbf{X},\mathbf{X}_m)\left(k(\mathbf{X}_m,\mathbf{X}_m)+\frac{1}{\sigma^2}k(\mathbf{X}_m,\mathbf{X})k(\mathbf{X},\mathbf{X}_m)\right)^{-1}k(\mathbf{X}_m,\mathbf{X})$$

The required matrix inversion is now just of an $m \times m$ matrix, which by assumption is much cheaper than the original formulation. Indeed, the computational cost of computing this object is now dominated by the matrix multiplications, which require $O(nm^2)$ time, a potentially massive reduction from the previous $O(n^3)$ cost, depending on the relationship between n and m.

The key unanswered question is thus how to find a "good" set of inducing points i.e., a set $\mathbf{X}_m$ that results in a good approximation of the form $k(\mathbf{x},\mathbf{x}') \approx k(\mathbf{x},\mathbf{X}_m)k(\mathbf{X}_m,\mathbf{X}_m)^{-1}k(\mathbf{X}_m,\mathbf{x}')$. Most inducing point methods rely on access to an existing dataset of points $\mathcal{D}=(\mathbf{X},\mathbf{y})$; a few techniques proposed in the literature include:

1. simply sampling m points uniformly at random from $\mathcal{D}$, a surprisingly effective strategy for sufficiently large m and a well-distributed training dataset;
2. clustering $\mathcal{D}$ into m clusters and using the cluster centers as the inducing points;
3. greedily building the set $\mathbf{X}_m$ from data points in $\mathcal{D}$ according to some selection criterion or
4. treating the set $\mathbf{X}_m$ as *latent* variables to be inferred given the "observed" training dataset.

We will expand upon this last method but in order to do so, we first need to introduce a powerful tool for approximate Bayesian inference known as *variational inference*.

Variational Inference

The central task in Bayesian inference is to compute the posterior using Bayes' theorem. However, an added wrinkle is that for some probabilistic models, we might assume the existence of variables that are not measured by our observations but still affect the distribution. These models are typically referred to as *latent variable models* and are defined by an observed dataset,

$$\mathcal{D}=\mathbf{X}=\begin{bmatrix}\mathbf{x}_1^\top\\ \vdots\\ \mathbf{x}_n^\top\end{bmatrix},$$

and the corresponding latent variables

$$\mathbf{Z}=\begin{bmatrix}\mathbf{z}_1^\top\\ \vdots\\ \mathbf{z}_m^\top\end{bmatrix}$$

(note the highly suggestive subscripts on the observed and latent variables).

In order to reason about the latent variables, we need to compute the posterior

$$p(\mathbf{Z}|\mathbf{X}) = \frac{p(\mathbf{X}|\mathbf{Z})p(\mathbf{Z})}{p(\mathbf{X})} = \frac{p(\mathbf{X}|\mathbf{Z})p(\mathbf{Z})}{\int p(\mathbf{X}|\mathbf{Z})p(\mathbf{Z})\,\mathrm{d}\mathbf{Z}}.$$

The problem, however, is that it is difficult to exactly compute the posterior for many interesting models. Specifically, the marginal likelihood $p(\mathbf{X})$ in the denominator is generally some high-dimensional, intractable integral.

The goal of variational inference is to *approximate* the conditional distribution of the latent variables given the observed variables. Variational inference restricts its attention to a specific family of distributions, say $\mathcal{Q}$, then frames the problem of finding the "best" conditional distribution, $q \in \mathcal{Q}$, as an optimization problem. We use the KL–divergence between $p(\mathbf{Z}|\mathbf{X})$ and $q(\mathbf{Z})$ as the criteria for finding the "best" approximation. In general, the conditional density chosen from $\mathcal{Q}$ is parameterized by some number of *variational parameters*, which will be the means by which we optimize the KL–divergence.

Forward vs. Reverse KL–divergence

The KL–divergence is asymmetric in its arguments, i.e., given two probability densities p and q, we know that $D(p\,||\,q) \neq D(q\,||\,p)$. If p is the true posterior density, $D(p\,||\,q)$ is called the *forward KL–divergence* and $D(q\,||\,p)$ is called the *reverse KL–divergence*. One may ask: which one should we minimize, $D(p\,||\,q)$ or $D(q\,||\,p)$?

For the distributions in question, the forward KL can be written as

$$D(p\,||\,q) = \int_{\mathcal{Z}} p(\mathbf{Z} \mid \mathbf{X}) \log \frac{p(\mathbf{Z} \mid \mathbf{X})}{q(\mathbf{Z})}\,\mathrm{d}\mathbf{Z}$$

and is low if $q(\mathbf{Z})$ is generally large everywhere (at least relative to $p(\mathbf{Z} \mid \mathbf{X})$), so minimizing the forward KL will tend to give rise to very broad approximations that 'cover" all of the probability mass of $p(\theta)$.

Conversely, the reverse KL is given by

$$D(q\,||\,p) = \int_{\mathcal{Z}} q(\mathbf{Z}) \log \frac{q(\mathbf{Z})}{p(\mathbf{Z} \mid \mathbf{X})}\,\mathrm{d}\mathbf{Z}$$

and is low if $q(\mathbf{Z})$ is small everywhere relative to $p(\mathbf{Z} \mid \mathbf{X})$, so minimizing this quantity will lead to very dense, highly peaked approximations (because $q(\mathbf{Z})$ still needs to integrate to 1 so the probability mass needs to be put somewhere). This distinction is represented in Figure 1.

One way of interpreting Figure 1 is that minimizing the forward KL encourages "mean–seeking" behavior while minimizing the reverse KL encourages "mode–seeking" behavior. In practice, we generally prefer the mode–seeking behavior as the mean of a multi–modal distribution is not guaranteed to be a good description of the entire distribution.

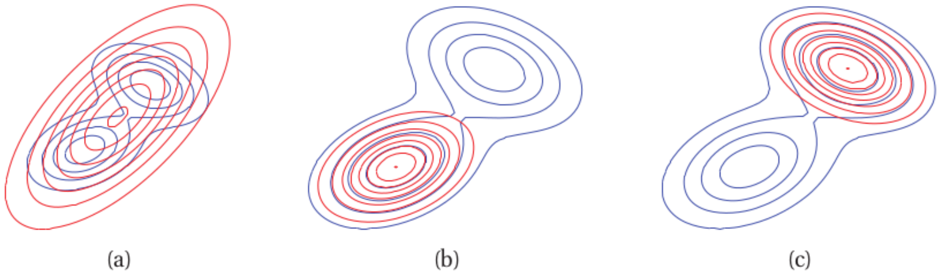

Figure 1: Forward vs. reverse KL on a bimodal posterior distribution. The blue curves are the contours of the true posterior density p and the red curves are the contours of the unimodal approximation q that (a) minimize the forward KL and (b,c) minimize the reverse KL.

Another reason we do not minimize the forward KL–divergence is that doing so involves working with the intractable posterior density $p(\mathbf{Z}|\mathbf{X})$. Remarkably, the reverse KL can be optimized without ever needing to directly compute this distribution! To see this, consider the following derivation:

$$\begin{aligned}
D(q \,||\, p) &= \int_{\mathcal{Z}} q(\mathbf{Z}) \log \frac{q(\mathbf{Z})}{p(\mathbf{Z} \mid \mathbf{X})} \,\mathrm{d}\mathbf{Z} \\
&= \int_{\mathcal{Z}} q(\mathbf{Z}) \log \frac{q(\mathbf{Z})\, p(\mathbf{X})}{p(\mathbf{Z}, \mathbf{X})} \,\mathrm{d}\mathbf{Z} \\
&= \int_{\mathcal{Z}} q(\mathbf{Z}) \log \frac{q(\mathbf{Z})}{p(\mathbf{Z}, \mathbf{X})} \,\mathrm{d}\mathbf{Z} + \int_{\mathcal{Z}} q(\mathbf{Z}) \log p(\mathbf{X}) \,\mathrm{d}\mathbf{Z} \\
&= \int_{\mathcal{Z}} q(\mathbf{Z}) \log \frac{q(\mathbf{Z})}{p(\mathbf{Z}, \mathbf{X})} \,\mathrm{d}\mathbf{Z} + \log p(\mathbf{X}) \\
\rightarrow \log p(\mathbf{X}) &= D(q \,||\, p) - \int_{\mathcal{Z}} q(\mathbf{Z}) \log \frac{q(\mathbf{Z})}{p(\mathbf{Z}, \mathbf{X})} \,\mathrm{d}\mathbf{Z} \\
&\coloneqq D(q \,||\, p) + \mathcal{L}(q)
\end{aligned}$$

where we have defined

$$\mathcal{L}(q) = \int_{\mathcal{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{Z}, \mathbf{X})}{q(\mathbf{Z})} \,\mathrm{d}\mathbf{Z}.$$

The term $\mathcal{L}(q)$ is known as the *evidence lower bound* (ELBO). Since $\log p(\mathbf{X})$ is some fixed value, we can see from the derivation above that maximizing the ELBO is equivalent to minimizing the reverse KL. Therefore,

$$\begin{aligned}
q^*(\mathbf{Z}) &= \arg\min_{q \in \mathcal{Q}} D(q \,||\, p) = \arg\max_{q \in \mathcal{Q}} \mathcal{L}(q) \\
&= \arg\max_{q \in \mathcal{Q}} \mathbb{E}_q \left[\log \frac{p(\mathbf{Z}, \mathbf{X})}{q(\mathbf{Z})} \right]
\end{aligned}$$

Variational Inference for $\mathbf{X}_m$

We are now equipped to formulate the problem of learning a sparse GP as a latent variable model. The observed variables are the values in the training dataset: $\mathcal{D} = (\mathbf{X}, \mathbf{y})$. The latent variables are $\mathbf{f}$ and $\mathbf{f}_m$ i.e., the *noiseless* latent function's values at $\mathbf{X}$ and $\mathbf{X}_m$ respectively. Given the GP belief on f, this allows us to treat $\mathbf{X}_m$ as the variational parameters that govern $q(\mathbf{f}_m)$: changing the locations in $\mathbf{X}_m$ will affect the distribution over $\mathbf{f}$ and $\mathbf{f}_m$ and the quality of the subsequent approximation.

Formally, the goal is to approximate the true intractable posterior $p(\mathbf{f}, \mathbf{f}_m \mid \mathbf{X}, \mathbf{y})$ using some distribution $q(\mathbf{f}, \mathbf{f}_m)$. We will actually go a step further and note that a reasonable desiderata for the optimal set of inducing points $\mathbf{X}_m$ is that their latent function values contain all of the relevant information about $\mathbf{f}$ i.e., $\mathbf{f}$ and $\mathcal{D}$ are conditionally independent given $\mathbf{f}_m$; this can be thought of as a reinterpretation of our original assertion about the Nyström approximation: the distribution over the function values in $\mathbf{f}$ is fully determined by their relationship with $\mathbf{f}_m$. Thus, we will assume that both p and q factor as

$$\begin{aligned} p(\mathbf{f}, \mathbf{f}_m \mid \mathbf{X}, \mathbf{y}) &= p(\mathbf{f} \mid \mathbf{f}_m)\, p(\mathbf{f}_m \mid \mathbf{X}, \mathbf{y}) \\ q(\mathbf{f}, \mathbf{f}_m) &= p(\mathbf{f} \mid \mathbf{f}_m)\, \phi(\mathbf{f}_m) \end{aligned}$$

where ϕ is now the variational distribution i.e., instead of optimizing q directly, we will minimize the reverse KL–divergence by finding $\phi^* \in \mathcal{Q}$. We will assume that ϕ is a multivariate Gaussian i.e. $\phi(\mathbf{f}_m) = \mathcal{N}(\mathbf{f}_m; \mathbf{a}, \mathbf{S})$.

Approximate Posterior Inference

Let us first stop and appreciate what we are working towards: once we solve for the optimal inducing points $\mathbf{X}_m^*$ and the optimal variational distribution parameters, $\mathbf{a}^*$ and $\mathbf{S}^*$, what do we them? By the assumptions of our latent variable model, it turns out that knowing these variational parameters is sufficient to perform efficient, approximate GP inference! Specifically, given the optimal variational parameters, we can compute the posterior belief for any set of latent function values $\hat{\mathbf{f}}$ as

$$\begin{aligned} p(\hat{\mathbf{f}} \mid \mathbf{X}, \mathbf{y}) &= \iint p(\hat{\mathbf{f}} \mid \mathbf{f}, \mathbf{f}_m, \mathbf{X}, \mathbf{y})\, p(\mathbf{f}, \mathbf{f}_m \mid \mathbf{X}, \mathbf{y})\, \mathrm{d}\mathbf{f}\, \mathrm{d}\mathbf{f}_m \\ &= \iint p(\hat{\mathbf{f}} \mid \mathbf{f}_m)\, p(\mathbf{f} \mid \mathbf{f}_m)\, p(\mathbf{f}_m \mid \mathbf{X}, \mathbf{y})\, \mathrm{d}\mathbf{f}\, \mathrm{d}\mathbf{f}_m \\ &= \int p(\hat{\mathbf{f}} \mid \mathbf{f}_m)\, p(\mathbf{f}_m \mid \mathbf{X}, \mathbf{y}) \left(\int p(\mathbf{f} \mid \mathbf{f}_m)\, \mathrm{d}\mathbf{f} \right) \mathrm{d}\mathbf{f}_m \\ &= \int p(\hat{\mathbf{f}} \mid \mathbf{f}_m)\, p(\mathbf{f}_m \mid \mathbf{X}, \mathbf{y})\, \mathrm{d}\mathbf{f}_m \approx \int p(\hat{\mathbf{f}} \mid \mathbf{f}_m)\, \phi(\mathbf{f}_m)\, \mathrm{d}\mathbf{f}_m. \end{aligned}$$

This final term is a convolution of two Gaussians as

$$\begin{aligned} \phi(\mathbf{f}_m) &= \mathcal{N}(\mathbf{f}_m; \mathbf{a}^*, \mathbf{S}^*) \text{ and} \\ p(\hat{\mathbf{f}} \mid \mathbf{f}_m) &= \mathcal{N}\left(\hat{\mathbf{f}}; k(\hat{\mathbf{X}}, \mathbf{X}_m^*) k(\mathbf{X}_m^*, \mathbf{X}_m^*)^{-1} \mathbf{f}_m, k(\hat{\mathbf{X}}, \hat{\mathbf{X}}) - k(\hat{\mathbf{X}}, \mathbf{X}_m^*) k(\mathbf{X}_m^*, \mathbf{X}_m^*)^{-1} k(\mathbf{X}_m^*, \hat{\mathbf{X}})\right), \end{aligned}$$

again using a zero-mean GP belief on f with covariance function k. This implies the (approximate) posterior belief on $\hat{\mathbf{f}}$ is Gaussian by closure under convolutions with

$$\begin{aligned}
\mu_{X_m^*}(\hat{\mathbf{X}}) &= \mathbb{E}_{\phi(\mathbf{f}_m)}[k(\hat{\mathbf{X}}, \mathbf{X}_m^*)k(\mathbf{X}_m^*, \mathbf{X}_m^*)^{-1}\mathbf{f}_m] = k(\hat{\mathbf{X}}, \mathbf{X}_m^*)k(\mathbf{X}_m^*, \mathbf{X}_m^*)^{-1}\mathbf{a}^* \\
k_{X_m^*}(\hat{\mathbf{X}}, \hat{\mathbf{X}}) &= \mathbb{E}_{\phi(\mathbf{f}_m)}\big[\operatorname{cov}(\hat{\mathbf{f}}, \hat{\mathbf{f}} \mid \mathbf{f}_m)\big] + \operatorname{cov}_{\phi(\mathbf{f}_m)}\left(\mathbb{E}[\hat{\mathbf{f}} \mid \mathbf{f}_m], \mathbb{E}[\hat{\mathbf{f}} \mid \mathbf{f}_m]\right) \\
&= \mathbb{E}_{\phi(\mathbf{f}_m)}\left[k(\hat{\mathbf{X}}, \hat{\mathbf{X}}) - k(\hat{\mathbf{X}}, \mathbf{X}_m^*)k(\mathbf{X}_m^*, \mathbf{X}_m^*)^{-1}k(\mathbf{X}_m^*, \hat{\mathbf{X}})\right] \\
&+ \operatorname{cov}_{\phi(\mathbf{f}_m)}\left(k(\hat{\mathbf{X}}, \mathbf{X}_m^*)k(\mathbf{X}_m^*, \mathbf{X}_m^*)^{-1}\mathbf{f}_m, k(\hat{\mathbf{X}}, \mathbf{X}_m^*)k(\mathbf{X}_m^*, \mathbf{X}_m^*)^{-1}\mathbf{f}_m\right) \\
&= k(\hat{\mathbf{X}}, \hat{\mathbf{X}}) - k(\hat{\mathbf{X}}, \mathbf{X}_m^*)k(\mathbf{X}_m^*, \mathbf{X}_m^*)^{-1}k(\mathbf{X}_m^*, \hat{\mathbf{X}}) \\
&+ k(\hat{\mathbf{X}}, \mathbf{X}_m^*)k(\mathbf{X}_m^*, \mathbf{X}_m^*)^{-1}\mathbf{S}^*k(\mathbf{X}_m^*, \mathbf{X}_m^*)^{-1}k(\mathbf{X}_m^*, \hat{\mathbf{X}})
\end{aligned}$$

where the derivation of the approximate posterior covariance uses the law of total covariance and the linearity of covariance. Despite their ungodly appearance, the key thing to note about these expressions is that they can be computed efficiently as all matrix inverses only involve $m \times m$ matrices. If $\hat{\mathbf{X}}$ consists of $N \gg m$ points, then the computational cost of computing these quantities is $O(Nm^2)$.

Deriving the Optimal Variational Parameters

Returning again the key question of optimizing the variational parameters, we begin by writing out the ELBO for this setting:

$$\begin{aligned}
\mathcal{L}(q) = \int_{\mathcal{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{Z}, \mathbf{X})}{q(\mathbf{Z})}\,\mathrm{d}\mathbf{Z} &= \iint p(\mathbf{f} \mid \mathbf{f}_m)\,\phi(\mathbf{f}_m) \log \frac{p(\mathbf{y}, \mathbf{f}, \mathbf{f}_m \mid \mathbf{X})}{p(\mathbf{f} \mid \mathbf{f}_m)\,\phi(\mathbf{f}_m)}\,\mathbf{df}\,\mathbf{df}_m \\
&= \iint p(\mathbf{f} \mid \mathbf{f}_m)\,\phi(\mathbf{f}_m) \log \frac{p(\mathbf{y} \mid \mathbf{f})\,p(\mathbf{f} \mid \mathbf{f}_m)\,p(\mathbf{f}_m \mid \mathbf{X})}{p(\mathbf{f} \mid \mathbf{f}_m)\,\phi(\mathbf{f}_m)}\,\mathbf{df}\,\mathbf{df}_m \\
&= \iint p(\mathbf{f} \mid \mathbf{f}_m)\,\phi(\mathbf{f}_m) \log \frac{p(\mathbf{y} \mid \mathbf{f})\,p(\mathbf{f}_m \mid \mathbf{X})}{\phi(\mathbf{f}_m)}\,\mathbf{df}\,\mathbf{df}_m \\
&= \int \phi(\mathbf{f}_m)\left[\int p(\mathbf{f} \mid \mathbf{f}_m) \log p(\mathbf{y} \mid \mathbf{f})\,\mathbf{df} + \log \frac{p(\mathbf{f}_m \mid \mathbf{X})}{\phi(\mathbf{f}_m)}\right]\mathbf{df}_m
\end{aligned}$$

To proceed, we will first simplify the integral with respect to $\mathbf{f}$:

$$\begin{aligned}
\int p(\mathbf{f} \mid \mathbf{f}_m) \log p(\mathbf{y} \mid \mathbf{f})\,\mathbf{df} &= \int p(\mathbf{f} \mid \mathbf{f}_m)\left[-\frac{n}{2}\log(2\pi) - \frac{1}{2}\log\det(\sigma^2 I_n) - \frac{1}{2}(\mathbf{y} - \mathbf{f})^T(\sigma^2 I_n)^{-1}(\mathbf{y} - \mathbf{f})\right]\mathbf{df} \\
&= \int p(\mathbf{f} \mid \mathbf{f}_m)\left[-\frac{n}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\left(\mathbf{y}^T\mathbf{y} - 2\mathbf{y}^T\mathbf{f} + \mathbf{f}^T\mathbf{f}\right)\right]\mathbf{df} \\
&= \mathbb{E}_{p(\mathbf{f} \mid \mathbf{f}_m)}\left[-\frac{n}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\left(\mathbf{y}^T\mathbf{y} - 2\mathbf{y}^T\mathbf{f} + \mathbf{f}^T\mathbf{f}\right)\right] \\
&= -\frac{n}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\left(\mathbf{y}^T\mathbf{y} - \mathbb{E}_{p(\mathbf{f} \mid \mathbf{f}_m)}\left[2\mathbf{y}^T\mathbf{f}\right] + \mathbb{E}_{p(\mathbf{f} \mid \mathbf{f}_m)}\left[\mathbf{f}^T\mathbf{f}\right]\right).
\end{aligned}$$

To continue the derivation, let

$$\begin{aligned}
p(\mathbf{f} \mid \mathbf{f}_m) &= \mathcal{N}(\mathbf{f}; \boldsymbol{\mu}_{\mathbf{f} \mid \mathbf{f}_m}, k_{\mathbf{f} \mid \mathbf{f}_m}) \\
&= \mathcal{N}\left(\mathbf{f}; k(\mathbf{X}, \mathbf{X}_m)k(\mathbf{X}_m, \mathbf{X}_m)^{-1}\mathbf{f}_m, k(\mathbf{X}, \mathbf{X}) - k(\mathbf{X}, \mathbf{X}_m)k(\mathbf{X}_m, \mathbf{X}_m)^{-1}k(\mathbf{X}_m, \mathbf{X})\right)
\end{aligned}$$

To compute the last term, $\mathbb{E}_{p(\mathbf{f}|\mathbf{f}_m)}\left[\mathbf{f}^T\mathbf{f}\right]$, we need to derive the expectation of the quadratic form of a Gaussian random variable, which we can do as follows:

$$\begin{aligned}\text{for } \mathbf{x} \sim \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \Sigma),\ \mathbb{E}[\mathbf{x}^T\mathbf{x}] = \mathbb{E}[\mathrm{Tr}(\mathbf{x}\mathbf{x}^T)] &= \mathrm{Tr}\left(\mathbb{E}[\mathbf{x}\mathbf{x}^T]\right) = \mathrm{Tr}\left(\mathrm{cov}(\mathbf{x},\mathbf{x}) + \mathbb{E}[\mathbf{x}]\mathbb{E}[\mathbf{x}^T]\right)\\ &= \mathrm{Tr}\left(\Sigma + \boldsymbol{\mu}\boldsymbol{\mu}^T\right) = \boldsymbol{\mu}^T\boldsymbol{\mu} + \mathrm{Tr}\left(\Sigma\right)\end{aligned}$$

Thus, continuing the derivation above gives

$$\begin{aligned}\int p(\mathbf{f} \mid \mathbf{f}_m)\log p(\mathbf{y} \mid \mathbf{f})\,\mathrm{d}\mathbf{f} &= -\frac{n}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\Big(\mathbf{y}^T\mathbf{y} - \mathbb{E}_{p(\mathbf{f}|\mathbf{f}_m)}\left[2\mathbf{y}^T\mathbf{f}\right] + \mathbb{E}_{p(\mathbf{f}|\mathbf{f}_m)}\left[\mathbf{f}^T\mathbf{f}\right]\Big)\\ &= -\frac{n}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\Big(\mathbf{y}^T\mathbf{y} - 2\mathbf{y}^T\boldsymbol{\mu}_{\mathbf{f}|\mathbf{f}_m} + \boldsymbol{\mu}_{\mathbf{f}|\mathbf{f}_m}^T\boldsymbol{\mu}_{\mathbf{f}|\mathbf{f}_m} + \mathrm{Tr}\left(k_{\mathbf{f}|\mathbf{f}_m}\right)\Big)\\ &= \log\mathcal{N}(\mathbf{y}; \boldsymbol{\mu}_{\mathbf{f}|\mathbf{f}_m}, \sigma^2 I_n) - \frac{1}{2\sigma^2}\mathrm{Tr}\left(k_{\mathbf{f}|\mathbf{f}_m}\right)\end{aligned}$$

Plugging this back into the ELBO gives

$$\begin{aligned}\mathcal{L}(\phi, \mathbf{X}_m) &= \int \phi(\mathbf{f}_m)\left[\log\mathcal{N}(\mathbf{y}; \boldsymbol{\mu}_{\mathbf{f}|\mathbf{f}_m}, \sigma^2 I_n) - \frac{1}{2\sigma^2}\mathrm{Tr}\left(k_{\mathbf{f}|\mathbf{f}_m}\right) + \log\frac{p(\mathbf{f}_m \mid \mathbf{X})}{\phi(\mathbf{f}_m)}\right]\mathrm{d}\mathbf{f}_m\\ &= \int \phi(\mathbf{f}_m)\left[\log\frac{\mathcal{N}(\mathbf{y}; \boldsymbol{\mu}_{\mathbf{f}|\mathbf{f}_m}, \sigma^2 I_n)\,p(\mathbf{f}_m \mid \mathbf{X})}{\phi(\mathbf{f}_m)}\right]\mathrm{d}\mathbf{f}_m - \frac{1}{2\sigma^2}\mathrm{Tr}\left(k_{\mathbf{f}|\mathbf{f}_m}\right)\end{aligned}$$

The first term can be thought of as the *negative* "KL-divergence" between $\phi(\mathbf{f}_m)$, which we assumed to be a Gaussian, and $\mathcal{N}(\mathbf{y}; \boldsymbol{\mu}_{\mathbf{f}|\mathbf{f}_m}, \sigma^2 I_n)\,p(\mathbf{f}_m \mid \mathbf{X})$, the product of two Gaussians, which is proportional to a Gaussian by closure under pointwise multiplication; the use of scare quotes is because $\mathcal{N}(\mathbf{y}; \boldsymbol{\mu}_{\mathbf{f}|\mathbf{f}_m}, \sigma^2 I_n)\,p(\mathbf{f}_m \mid \mathbf{X})$ is not a proper distribution so we cannot call this a true KL-divergence.

Thus, we can conclude that this quantity is maximized when $\phi(\mathbf{f}_m) \propto \mathcal{N}(\mathbf{y}; \boldsymbol{\mu}_{\mathbf{f}|\mathbf{f}_m}, \sigma^2 I_n)\,p(\mathbf{f}_m \mid \mathbf{X})$, which allows us to compute the optimal variational distribution as

$$\begin{aligned}&\phi^*(\mathbf{f}_m) = \mathcal{N}(\mathbf{f}_m; \mathbf{a}^*, \mathbf{S}^*) \text{ where}\\ &\quad\mathbf{a}^* = \frac{1}{\sigma^2}k(\mathbf{X}_m, \mathbf{X}_m)\left(k(\mathbf{X}_m, \mathbf{X}_m) + \frac{1}{\sigma^2}k(\mathbf{X}_m, \mathbf{X})k(\mathbf{X}, \mathbf{X}_m)\right)^{-1}k(\mathbf{X}_m, \mathbf{X})\mathbf{y}\\ &\quad\mathbf{S}^* = k(\mathbf{X}_m, \mathbf{X}_m)\left(k(\mathbf{X}_m, \mathbf{X}_m) + \frac{1}{\sigma^2}k(\mathbf{X}_m, \mathbf{X})k(\mathbf{X}, \mathbf{X}_m)\right)^{-1}k(\mathbf{X}_m, \mathbf{X}_m)\end{aligned}$$

Finally, in order to solve for $\mathbf{X}_m^*$, the optimal set of inducing points, we observe that *if* $\phi^*(\mathbf{f}_m)$ were exactly equal to $\mathcal{N}(\mathbf{y}; \boldsymbol{\mu}_{\mathbf{f}|\mathbf{f}_m}, \sigma^2 I_n)\,p(\mathbf{f}_m \mid \mathbf{X})$, then the first term of $\mathcal{L}(\phi, \mathbf{X}_m)$ at optimality would be zero. However, because these two are only proportional to each other, if we plug $\phi^*(\mathbf{f}_m)$ back in to $\mathcal{L}(\phi, \mathbf{X}_m)$, the first term becomes the log of the normalizing constant for the numerator i.e.,

$$\begin{aligned}\mathcal{L}(\mathbf{X}_m) = \log\mathcal{N}\big(&\mathbf{y}; \mathbf{0}, \sigma^2 I_n + k(\mathbf{X}, \mathbf{X}_m)k(\mathbf{X}_m, \mathbf{X}_m)^{-1}k(\mathbf{X}_m, \mathbf{X})\big)\\ &- \frac{1}{2\sigma^2}\mathrm{Tr}\left(k(\mathbf{X}, \mathbf{X}) - k(\mathbf{X}, \mathbf{X}_m)k(\mathbf{X}_m, \mathbf{X}_m)^{-1}k(\mathbf{X}_m, \mathbf{X})\right)\end{aligned}$$

where we have substituted back in $k_{\mathbf{f}|\mathbf{f}_m} = k(\mathbf{X}, \mathbf{X}) - k(\mathbf{X}, \mathbf{X}_m)k(\mathbf{X}_m, \mathbf{X}_m)^{-1}k(\mathbf{X}_m, \mathbf{X})$ from earlier. This can be optimized directly using gradient-based methods to solve for $\mathbf{X}_m^*$ provided we choose a differentiable kernel (as we typically do).