Becs-114.4613 - Special Course in Bayesian Modeling

Sparse Approximate Gaussian Process Regression

Srikanth Gadicherla
8th April 2016

AGENDA

- Recap of Gaussian Process Regression
- A New Unifying View
- The Subset of Data (SoD) Approximations
- The Subset of Regressors (SoR) Approximations
- The Deterministic Training Conditional (DTC) Approximations
- The Fully Independent Training Conditional (FITC) Approximations
- The Partially Independent Training Conditional (PITC) Approximations
- Transduction and Augmentation
- On the Choice of the Inducing Variables

A New Unifying View

- Let us seek to modify the joint prior $p(f_*, f)$ in way which would reduce the computational requirement.
- Let us rewrite the prior by introducing additional set of m latent variables called *inducing variables.*

$$u = [u_1, u_1 \ldots, u_1]^T$$

- These latent variables correspond to a set of input locations X_u, called inducing inputs.
- The additional latent variables are marginalized out in the predictive distribution, the choice of inducing inputs does leave an imprint on the final solution.

A New Unifying View

- Due to consistency, we can recover $p(f_*, f)$ simply integrating out u from the joint prior $p(f_*, f, u)$

$$p(f_*, f) = \int p(f_*, f, u)\, du = \int p(f_*, f|u) p(u)\, du\,,$$

$$p(u) = N(0, K_{u,u})$$

- This is an exact expression, we now introduce the fundamental approximations which give rise to almost all sparse approximations.
- We assume that the joint prior by assuming the f_* and f are conditionally independent given u. This is fundamental for all approximations.

$$p(f_*, f) \approx q(f_*, f) = \int q(f_*|u) q(f|u) p(u)\, du$$

The name inducing variable is motivated by the fact that f and f_* communicate through u and therefore u induces the dependencies between training and test cases.

A New Unifying View

- The different computationally efficient algorithms correspond to different additional assumptions about the two approximate inducing conditionals $q(f_*|u)$ and $q(f|u)$ in the previous integral.
- Just for reference, the exact expression for the above approximations.

training conditional : $\mathrm{p}(f|u) = N(K_{f,u}K_{u,u}^{-1}u, K_{f,f} - Q_{f,f})$

test conditional : $\mathrm{p}(f_*|u) = N(K_{*,u}K_{u,u}^{-1}u, K_{*,*} - Q_{*,*})$

where$Q_{f,f} \triangleq K_{a,u}K_{u,u}^{-1}u, K_{u,b}$

- We can identify this equation as noise-free version of $\mathrm{p}(f_*|y)$ from G.P regression.

A New Unifying View

- The positive-semidefinite covariance matrices in the above exact expression have the form $K - Q$ and can be interpreted as prior covariance K minus a non negative definite matrix Q quantifying how much information u provides about the variables in question (f_* or f)
- The emphasis is that all the sparse methods discussed are simply the different approximations to the exact expression of p($f|u$) and p($f_*|u$) and throughout exact likelihood is used with inducing prior

$$\mathrm{p}(y|f) = N(f, \sigma_{noise}^2 I) \text{ and } p(u) = N(0, K_{u,u})$$

Subset of Data Approximations

- This is considered the baseline method for comparing to other methods.
- In this we only consider a subset of data (SoD)
- The computational complexity is reduced to $O(m^3)$, where $m < n$.
- Though is difficult to select redundant data or to select optimum subset.

Subset of Regressors (SoR) approximation

- It is a linear-in-the-parameters model with a particular prior on the weights.
- For any input x_*, the corresponding function value f_* is given by

$$f_* = K_{*,u} w_u \qquad \text{with } p(w_u) = N(0, K_{u,u}^{-1})$$

where there is one weight associated with each inducing input.

- The covariance matrix for the prior on the weights is the inverse of that of u, so that we recover exact G.P prior on u, which is zero mean and covariance

$$u = K_{u,u} w_u \qquad \Longrightarrow \langle u u^T \rangle = K_{u,u} \langle w_u {w_u}^T \rangle K_{u,u} = K_{u,u}$$

Subset of Regressors (SoR) approximation

- Using the effective prior on u and the fact that $w_u = K_{u,u}^{-1} u$, we can redefine the SoR model

$$f_* = K_{*,u} K_{u,u}^{-1} u \qquad \text{with } u \backsim N(0, K_{u,u})$$

- So, the final model is, given the deterministic relation between any f_* and u

$$q_{SoR}(f|u) = N\left(K_{f,u} K_{u,u}^{-1} u, 0\right),\ q_{SoR}(f_*|u) = N(K_{*,u} K_{u,u}^{-1} u, 0)$$

with zero conditional covariance, compare to equation p($f|u$) and p($f_*|u$).

- The effective prior implied by the SoR approximation is easily obtained from (8),

$$q_{SoR}(f, f_*) = N\left(0, \begin{bmatrix} Q_{f,f} & Q_{f,*} \\ Q_{*,f} & Q_{*,*} \end{bmatrix}\right)$$

where $Q_{a,b} \triangleq K_{a,u} K_{u,u}^{-1} K_{u,b}$.

Subset of Regressors (SoR) approximation

- A more descriptive name would be Deterministic Inducing Conditional (DIC) approximation.
- We see that approximate prior is degenerate. There are only m degrees of freedom in the model, which implies that only m linearly independent functions can be drawn from the prior, the m+1th one is linear combination of the previous.
- This causes unreasonable predictive distribution, and the approximate prior is so restrictive, so given enough data only limited family of functions is plausible under the posterior, leading to overconfident predictive variance .

Subset of Regressors (SoR) approximation

- So, the predictive distribution is

$$q_{SoR}(f_*|y) = N\left(Q_{*,f}\left(Q_{f,f} + \sigma_{noise}^2 I\right)^{-1} y, Q_{*,*} - Q_{*,f}\left(Q_{f,f} + \sigma_{noise}^2 I\right)^{-1} Q_{f,*}\right)$$

$$= N\left(\sigma^{-2} K_{*,u} \sum K_{u,f}\, y, K_{*,u} \sum K_{u,*}\right)$$

where $\sum = \left(\sigma^{-2} K_{u,f} K_{f,u} + K_{u,u}\right)^{-1}$

- Which corresponds to G.P regression prepdiction expression.

Deterministic training conditional (DTC) Approximation

- Doesn't suffer from nonsensical predictive uncertainties of the SoR model, but interestingly leads to same solution as SoR, The method relies on likelihood approximation based on projection $f = K_{f,u}K_{u,u}^{-1}u$

$$\mathrm{p}(y|f) \cong q(y|u) = N(K_{f,u}K_{u,u}^{-1}u, \sigma_{noise}^{2} I) \qquad (17)$$

Also been called as Projected Process Approximation (PPA) and Projected Latent Variables (PLV).

- One way of obtaining equivalent model is to retain the usual likelihood, but to impose a deterministic training conditional and the exact test conditional from equation 9b

$$q_{DTC}(f|u) = N\left(K_{f,u}K_{u,u}^{-1}u, 0\right), q_{DTC}(f_*|u) = p(f_*|u)$$

Deterministic training conditional (DTC) Approximation

- The fundamental difference with SoR is that DTC uses exact test conditional instead of deterministics relation .

$$q_{DTC}(f, f_*) = N\left(0, \begin{bmatrix} Q_{f,f} & Q_{f,*} \\ Q_{*,f} & K_{*,*} \end{bmatrix}\right)$$

- The predictive distribution is now given by

$$q_{DTC}(f_*|y) = N\left(Q_{*,f}\left(Q_{f,f} + \sigma_{noise}^2 I\right)^{-1} y, K_{*,*} - Q_{*,f}\left(Q_{f,f} + \sigma_{noise}^2 I\right)^{-1} Q_{f,*}\right)$$

$$= N\left(\sigma^{-2} K_{*,u} \sum K_{u,f}\, y, K_{*,*} - Q_{*,*} + K_{*,u} \sum K_{*,u}^T\right)$$

where $\sum = \left(\sigma^{-2} K_{u,f} K_{f,u} + K_{u,u}\right)^{-1}$

Deterministic training conditional (DTC) Approximation

- The only difference between the predictive distribution of DTC and SoR is the variance.The predictive variance of DTC is never smaller than that of SoR.
- The DTC approximation does not correspond exactly to a Gaussian process

Approximation on toy data

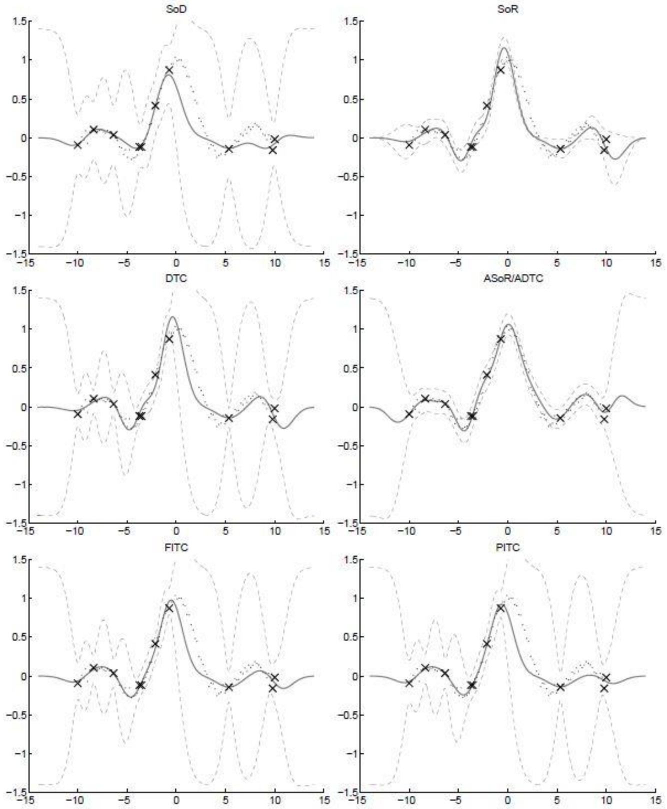


Figure 5: Toy example with identical covariance function and hyperparameters. The squared exponential covariance function is used, and a slightly too short lengthscale is chosen on purpose to emphasize the different behaviour of the predictive uncertainties. The dots are the training points, the crosses are the targets corresponding to the inducing inputs, randomly selected from the training set. The solid line is the mean of the predictive distribution, and the dotted lines show the 95% confidence interval of the predictions. Augmented DTC (ADTC) is equivalent to augmented SoR (ASoR), see Remark 12.

The Fully Independent Training Conditional (FITC) Approximation

- Another likelihood approximation which was called Sparse Gaussian Processes with Pseudo-Inputs (SGPP). While DTC is based on the likelihood approximation given by (17). SGPP proposes a more sophisticated likelihood approximation with a richer covariance

$$\mathrm{p}(y|f) \cong q(y|u) = N\left(K_{f,u}K_{u,u}^{-1}u, diag\left[K_{f,f} - Q_{f,f}\right] + \sigma_{noise}^{2}I\right)$$

- Alternate formulation based on the inducing conditional, similar to DTC:

$$q_{FITC}(f|u) = \prod_{i=1}^{N} p(f_i|u) = N\left(K_{f,u}K_{u,u}^{-1}u, diag\left[K_{f,f} - Q_{f,f}\right]\right),$$
$$q_{FITC}(f_*|u) = p(f_*|u)$$

The Fully Independent Training Conditional (FITC) Approximation

- We see that as opposed to SoR and DTC, FITC doesn’t impose deterministic relation between f and u.
- *Instead of ignoring the variance, FITC proposes an approximation to the training conditional distribution f given u as further independence assumption.*

The Fully Independent Training Conditional (FITC) Approximation

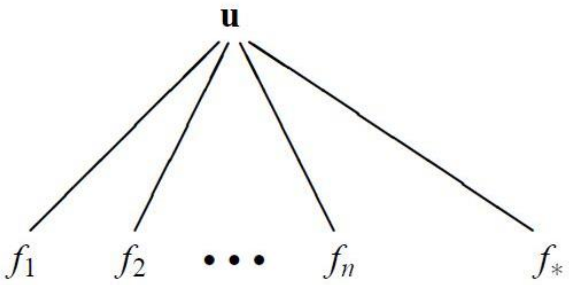


Figure 2: Graphical model for the FITC approximation. Compared to those in Figure 1, all edges between latent function values have been removed: the latent function values are conditionally fully independent given the inducing variables $\mathbf{u}$. Although strictly speaking the SoR and DTC approximations could also be represented by this graph, note that both further assume a deterministic relation between $\mathbf{f}$ and $\mathbf{u}$.

The Fully Independent Training Conditional (FITC) Approximation

- In addition to the exact test conditional, we initially consider only a single test case, f*.
- The effective prior implied by FITC is given by

$$q_{FITC}(f, f_*) = N\left(0, \begin{bmatrix} Q_{f,f} - diag\left[Q_{f,f} - K_{f,f}\right] & Q_{f,*} \\ Q_{*,f} & Q_{*,*} \end{bmatrix}\right)$$

- Sole difference between DTC and FITC is that the top left corner of the implied prior covariance, FITC replaces the approximate covariance of the DTC by the exact ones on the diagonal.

The Fully Independent Training Conditional (FITC) Approximation

- The predictive distribution is

$$
\begin{aligned}
q_{FITC}(f_*|y) &= N\left(Q_{*,f}(Q_{f,f}+\Lambda)^{-1}y, K_{*,*}-Q_{*,f}(Q_{f,f}+\Lambda)^{-1}Q_{f,*}\right) \\
&= N\left(K_{*,u}\Sigma K_{u,f}\,\Lambda^{-1}y, K_{*,*}\ -Q_{*,*}+K_{*,u}\Sigma K_{u,*}\right)
\end{aligned}
$$

where $\Sigma = \left(K_{u,f}\Lambda^{-1}K_{f,u}+K_{u,u}\right)^{-1}$ and $\Lambda = \mathrm{diag}[K_{u,f}\ -Q_{f,f}+\sigma^2_{noise}I]$

- The computational complexity is identical to SoR and DTC.

The Fully Independent Training Conditional (FITC) Approximation

- So far, we've considered only a single test case. There are two options for joint predicitions:
  1. use the exact full test conditional
  2. extend the additional factorizing assumptions to the test conditional
- Above two options not discussed, but the authors would probably intend the second option.
- Additional independence assumption for test cases is not necessary for computational reasons, and it affects the approximation
- Under option (1) the training and test covariances are computed differently, hence not strict GP.

The Fully Independent Conditional (FIC) Approximation

- *Iff the assumption of full independence is extended to the test conditional, the FITC approximation is equivalent to exact inference in a non-degenerate GP with covariance function.*

- $k_{FIC}(x_i, x_j) = k_{SoR}(x_i, x_j) + \delta_{ij}[k(x_i, x_j) - k_{SoR}(x_i, x_j)]$
- It is called Fully Independent Conditional (FIC) approximation. The effective prior implied by FIC is:

- $q_{FIC}(f, f_*) = N\left(0, \begin{bmatrix} Q_{f,f} - diag[Q_{f,f} - K_{f,f}] & Q_{f,*} \\ Q_{*,f} & Q_{*,*} - diag[Q_{*,*} - K_{*,*}] \end{bmatrix}\right)$

The Partially Independent Training Conditional (PITC) Approximation

- Previously, we saw how to improve the DTC approximation by approximating the training conditional with an independent distribution, i.e., one with the diagonal covariance matrix
- PITC further improves the approximation by extending the training conditional to have a block diagonal covariance:

$$q_{PITC}(f|u) = N\left(K_{f,u}K_{u,u}^{-1}u, blockdiag\left[K_{f,f} - Q_{f,f}\right]\right),$$
$$q_{FITC}(f_*|u) = p(f_*|u)$$

where blockdiag[.] is block diagonal matrix, where blocking structure is not explicitly stated.

The Partially Independent Training Conditional (PITC) Approximation

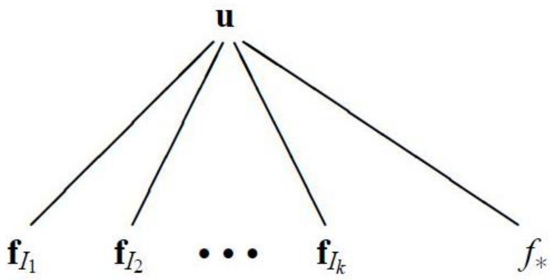


Figure 3: Graphical representation of the PITC approximation. The set of latent function values $\mathbf{f}_{I_i}$ indexed by the the set of indices I_i is fully connected. The PITC differs from FITC (see graph in Fig. 2) in that conditional independence is now between the *k groups* of training latent function values. This corresponds to the block diagonal approximation to the true training conditional given in (26).

The Partially Independent Training Conditional (PITC) Approximation

- Analogously to the FITC approximation, we get the joint prior

$$q_{PITC}(f, f_*) = N\left(0, \begin{bmatrix} Q_{f,f} - \color{red}{blockdiag}[K_{f,f} - Q_{f,f}] & Q_{f,*} \\ Q_{*,f} & Q_{*,*} \end{bmatrix}\right)$$

- Similar to FITC, the predictive distribution is

$$\begin{aligned} q_{PITC}(f_*|y) &= N\left(Q_{*,f}(Q_{f,f} + \Lambda)^{-1}y, K_{*,*} - Q_{*,f}(Q_{f,f} + \Lambda)^{-1}Q_{f,*}\right) \\ &= N\left(K_{*,u} \sum K_{u,f} \Lambda^{-1}y, K_{*,*} - Q_{*,*} + K_{*,u} \sum K_{u,*}\right) \end{aligned}$$

But Λ $\color{red}{= \text{blockdiag}}$$[K_{u,f} - Q_{f,f} + \sigma_{noise}^2 I]$

and $\sum = \left(K_{u,f}\Lambda^{-1}K_{f,u} + K_{u,u}\right)^{-1}$

The Partially Independent Training Conditional (PITC) Approximation

- An identical expression was obtained developing from Bayesian Committee machine (BCM).
- The BCM was originally proposed as a transductive learner(i.e., where test input have to be known before training), and the inducing inputs X_u were chosen to be test points.
- BCM realizes two orthogonal ideas
  1. the block diagonal structure of the partially independent training conditional (discussed next)
  2. the inducing points to be the test points

The Partially Independent Training Conditional (PITC) Approximation

- The computational complexity of the PITC approximation depends on the blocking structure and reasonable choice may be $k = n/m$ blocks, each of size $m \times m$.
- The complexity remains $O(nm^2)$.
- Since, the covariance is computed differently for training and test cases, so PITC doesn't correspond exactly to a GP.

Transduction

- The idea of transduction is to restrict the goal of learning to prediction on a pre-specified set of test cases, rather than trying to learn an entire function (induction) and then evaluate it at the test inputs.

- *“Transduction occurs only if the predictive distribution depends on other test inputs.”*

Transduction

- Motivation:
  1. transduction is somehow easier than induction.
  2. test inputs may reveal important information, which should be used during training(semi-supervised learning).
  3. for approximate algorithm one may be able to limit the discrepancies of the approximations at the test points.
- For exact GP models, first doesn't apply. If you make predictions at the test points that are consistent with a GP, then it is trivial inside the GP framework to extend these any other input points and in effect we've done induction.
- Second reason being, consistency property holds for standard GP setting, therefore, predictions at one test input are independent of location of any other point. So, transduction not applicable with exact GP's.

Transduction

- Transduction can occur in sparse setting though, by making the choice of inducing variables depending on the test inputs.
- Since the inducing variables are connected to all other nodes we would expect the approximation to be good at $u = f_*$, as we care only about predictions. This may not be a sufficient consideration for a good model!
- The model has to also explain the training data as well, which depends on u and wrong choices can distort the posterior.
- Therefore, the choice of u should be governed by the ability to model the conditional of the latent given the inputs, and solely not on test points.

Transduction

- The main drawback of transduction is that it doesn’t provide a predictive model in the way inductive model do.
- Interesting that other methods spend much effort trying to optimize the inducing variables BCM simply uses the test inputs, upon which one usually have no control.

AUGMENTATION

- In contrast to Transduction, it is done for test cases individually.
- No assumptions made on u so far, so an additional inducing variable is added to each test latent fucntion f_*. This is called augmentation.
- Augmentation doesn't make sense for an exact, non-degenerate GP model, since corresponding f_* would already be connected to all other variables.
- But augmentation makes sense for sparse models.

Augmentation

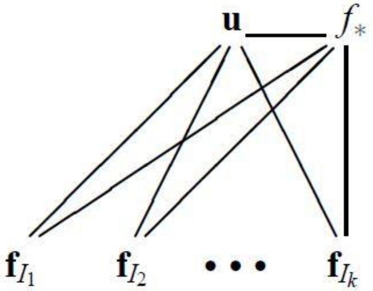


Figure 4: Two views on Augmentation. One view is to see that the test latent function value f_* is now part of the inducing variables $\mathbf{u}$ and therefore has access to the training latent function values. An equivalent view is to consider that we have dropped the assumption of conditional independence between f_* and the training latent function values. Even if f_* has now direct access to each of the training f_i, these still need to go through $\mathbf{u}$ to talk to each other if they fall in conditionally independent blocks. We have in this figure decided to recycle the graph for PITC from Figure 3 to show that all approximations we have presented can be augmented, irrespective of what the approximation for the training conditional is.

AUGMENTATION

- The more general process view on augmentation has several advantages over the basis function view. It is not completely clear from basis function view, which basis function should be used for augmentation.
- Effective prior for augmented SoR

$$q_{ASoR}(f_*, f) = \int q_{SoR}(f|f_*, u)p(f_*|u)\, du$$

- Interesting to see that

$$q_{SoR}(f|f_*, u) = q_{DTC}(f|f_*, u)$$

On the choice of the inducing variables

- We have been assuming that the inducing point have been given.
- Traditionally, the sparse models been built upon a carefully chosen subset of the training inputs. So, the inducing points X_u can be selected among the training inputs.
- Since this involves prohibitive combinatorial optimization, greedy optimization approaches have been suggested using various selection criteria like online learning, greedy posterior maximization, maximum information gain.

ON THE CHOICE OF THE INDUCING VARIABLES

- Selecting the inducing points from the test inputs has also been considered in transductive setting.
- It has been proposed to relax this constraint that the inducing variables must be a subset of training/test cases, turning the discrete selection problem into one of continuous optimization.

- WHAT OPTIMAL CRITERION BE USED?

On the choice of the inducing variables

- Marginal Likelihood approach (ML-II)
  - Departing from the fully Bayesian treatment which would involve defining priors on X_u, one could maximize the marginal likelihood(also called the evidence) with respect to X_u.
  - Each of the approximate methods proposed involves a different effective priors and hence its own particular effective marginal likelihood conditioned on the inducing inputs.

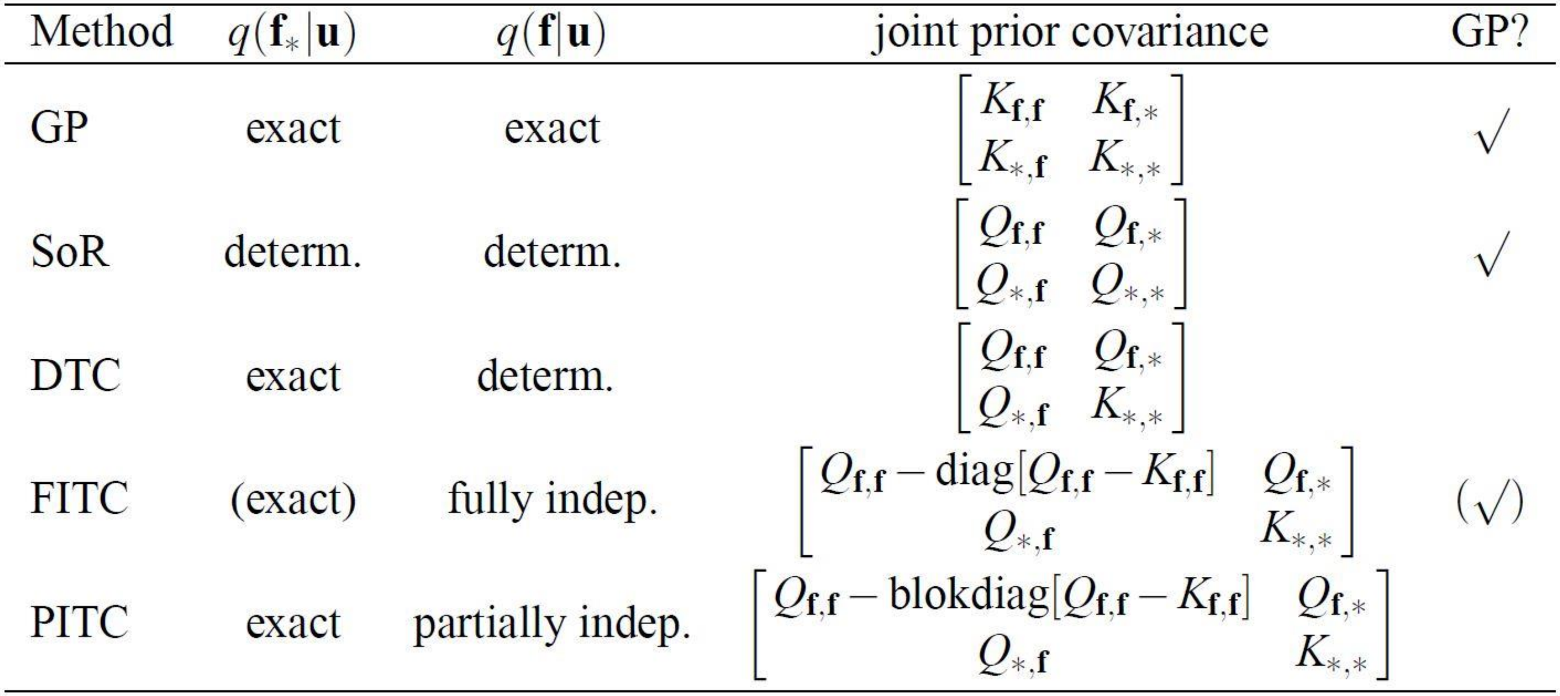
$$\begin{aligned} \mathrm{q}(\mathrm{y}|X_u) &= \int\int p(y|f)q(f|u)p(u|X_u)\,du\,df \\ &= \int p(y|f)q(f|X_u)\,df \end{aligned}$$

which is of course independent of test conditional.

On the choice of the inducing variables

- Maximize the effective posterior
  - It has been proposed to maximize the effective posterior instead of the effective marginal likelihood. However, this is potentially dangerous and can lead to overfitting. Instead maximizing the whole is sound and comes at an identical computational cost.

Summary – insert table 1



Method	$q(\mathbf{f}_* \mid \mathbf{u})$	$q(\mathbf{f} \mid \mathbf{u})$	joint prior covariance	GP?
GP	exact	exact	$\begin{bmatrix} K_{\mathbf{f},\mathbf{f}} & K_{\mathbf{f},*} \\ K_{*,\mathbf{f}} & K_{*,*} \end{bmatrix}$	$\surd$
SoR	determ.	determ.	$\begin{bmatrix} Q_{\mathbf{f},\mathbf{f}} & Q_{\mathbf{f},*} \\ Q_{*,\mathbf{f}} & Q_{*,*} \end{bmatrix}$	$\surd$
DTC	exact	determ.	$\begin{bmatrix} Q_{\mathbf{f},\mathbf{f}} & Q_{\mathbf{f},*} \\ Q_{*,\mathbf{f}} & K_{*,*} \end{bmatrix}$	
FITC	(exact)	fully indep.	$\begin{bmatrix} Q_{\mathbf{f},\mathbf{f}} - \mathrm{diag}[Q_{\mathbf{f},\mathbf{f}} - K_{\mathbf{f},\mathbf{f}}] & Q_{\mathbf{f},*} \\ Q_{*,\mathbf{f}} & K_{*,*} \end{bmatrix}$	$(\surd)$
PITC	exact	partially indep.	$\begin{bmatrix} Q_{\mathbf{f},\mathbf{f}} - \mathrm{blokdiag}[Q_{\mathbf{f},\mathbf{f}} - K_{\mathbf{f},\mathbf{f}}] & Q_{\mathbf{f},*} \\ Q_{*,\mathbf{f}} & K_{*,*} \end{bmatrix}$	

Approximation on toy data

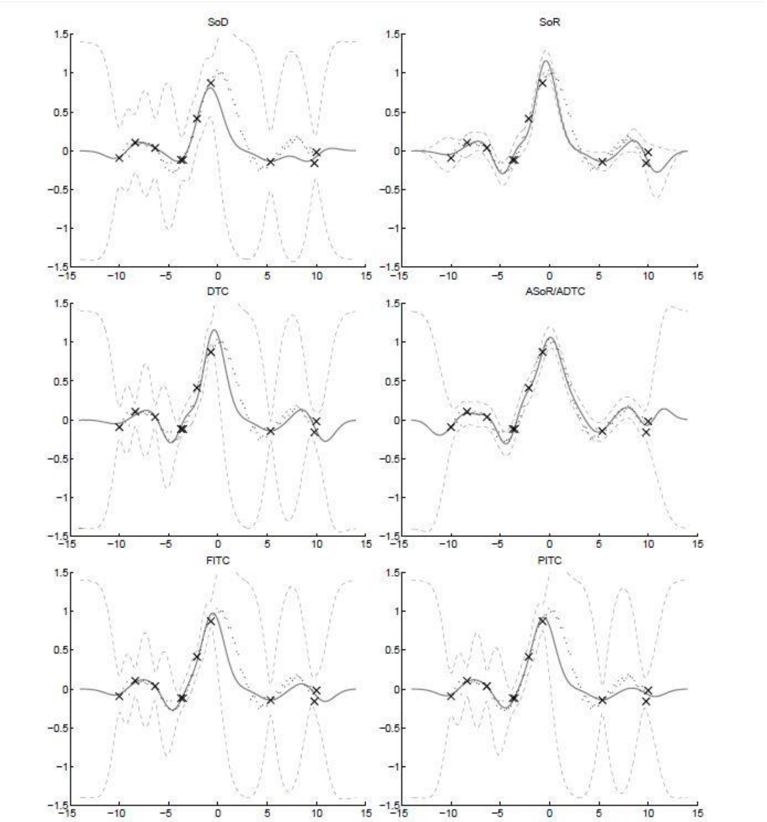


Figure 5: Toy example with identical covariance function and hyperparameters. The squared exponential covariance function is used, and a slightly too short lengthscale is chosen on purpose to emphasize the different behaviour of the predictive uncertainties. The dots are the training points, the crosses are the targets corresponding to the inducing inputs, randomly selected from the training set. The solid line is the mean of the predictive distribution, and the dotted lines show the 95% confidence interval of the predictions. Augmented DTC (ADTC) is equivalent to augmented SoR (ASoR), see Remark 12.