Sparse Sequential Gaussian Processes

Remi Barillec and Dan Cornford

MUCM meeting, 8-9 July 2009

Outline

Dealing with large data sets

- GPs are a very flexible and rich model class, but computationally scale as $O(n^3)$, due to the matrix inversion.
- This might not seem too big a problem since for prediction the inverse needs only be computed once (and we can use a variety of linear algebra tricks).
- However to estimate hyper-parameters in the covariance function will require many inversions.
- In really big data sets just storing the covariance matrix becomes problematic.
- In this talk, we describe a method to address these problems, developed by Manfred Opper and Lehel Csato [1]

The representer theorem

The posterior GP given some observations X is:

$$p(f|X,\boldsymbol{y}) = \frac{p(\boldsymbol{y}|X,f)p(f|X)}{p(\boldsymbol{y}|X)},$$

- It can be shown that the posterior process has a mean and kernel function given by:

$$m_{post}(\boldsymbol{x}) = m(\boldsymbol{x}) + \sum_i K(\boldsymbol{x},\boldsymbol{x}_i)\, q(i) \tag{1}$$

$$K_{post}(\boldsymbol{x},\boldsymbol{x}') = K(\boldsymbol{x},\boldsymbol{x}') + \sum_{i,j} K(\boldsymbol{x},\boldsymbol{x}_i) R(i,j) K(\boldsymbol{x}_j,\boldsymbol{x}') \tag{2}$$

- However, the $q(i)$ and $R(i,j)$ involve non-trivial integrals and are often intractable
- Furthermore, the posterior process can be non-Gaussian
- Csato and Opper introduce a sequential online learning algorithm to overcome these difficulties

Sequential inference in GPs

- Assume the likelihood factorises, so that $p(\boldsymbol{y}|X,f) = \prod_i p(y_i|X_i,f)$.
- We can update the posterior sequentially by considering one observation at a time:

$$p(f|X_t,\boldsymbol{y}_t) = \frac{p(y_t|\boldsymbol{x}_t,f)p(f|X_{t-1},\boldsymbol{y}_{t-1})}{p(\boldsymbol{y}_t|X_t)}\,.$$

- $p(f|X_t,\boldsymbol{y}_t)$ is generally no longer a GP, so the best approximating GP, $\hat{p}(f|X_t,\boldsymbol{y}_t)$, is found by minimising $\mathrm{KL}[p(f|X_k,\boldsymbol{y}_k)\|\hat{p}(f|X_k,\boldsymbol{y}_k)]$.
- At each step we include a new observation, which updates our GP posterior, computed as the projection onto the optimal GP posterior using the KL divergence.

The Kullback-Leibler divergence

The KL divergence is a distance measure between pdf's:

$$\mathrm{KL}[p(\theta)\|q(\theta)] = \int \ln\left(\frac{p(\theta)}{q(\theta)}\right) p(\theta) d\theta \,.$$

The KL distance is very widely used in variational treatments of machine learning problems.

- The approach is to minimise:

$$\mathrm{KL}[p(f|X_t, \boldsymbol{y}_t)\|\hat{p}(f|X_t, \boldsymbol{y}_t)] = \int p(f|X_t, \boldsymbol{y}_t) \ln\left(p(f|X_t, \boldsymbol{y}_t)\right) df - \int p(f|X_t, \boldsymbol{y}_t) \ln\left(\hat{p}(f|X_t, \boldsymbol{y}_t)\right) df \,,$$

  with respect to the parameters in the parametrisation shown previously, sequentially.
- For a Gaussian $\hat{p}$, minimising $\mathrm{KL}[p\|\hat{p}]$ is equivalent to matching moments

Sequential inference in GPs

Single observation update

$$m_t(\boldsymbol{x}) = m_{t-1}(\boldsymbol{x}) + K_{t-1}(\boldsymbol{x}, \boldsymbol{x}_t)\, q_t \tag{3}$$

$$K_t(\boldsymbol{x}, \boldsymbol{x}') = K_{t-1}(\boldsymbol{x}, \boldsymbol{x}') + r_t\, K_{t-1}(\boldsymbol{x}, \boldsymbol{x}_t) K_{t-1}(\boldsymbol{x}_t, \boldsymbol{x}') \tag{4}$$

- q_t and r_t are defined as the first and second derivatives of the logarithm of the expected likelihood: $\int p(y_t|\boldsymbol{x}_t, f)p(f|X_{t-1}, \boldsymbol{y}_{t-1})df$ wrt the posterior GP at $t-1$ at the new point $\boldsymbol{x}_t$.
- Since a single observation is considered at a time, the q_t and r_t are scalars which only involve 1-D integrals
- These integrals can be computed analytically for many likelihoods, otherwise numerical methods can be used to solve them efficiently

Recursive formulation

- The previous parametrisation can be rewritten using a recursive formulation:

$$f(\boldsymbol{x}) \sim N\left[\overset{1\times n}{\alpha_t^T}\overset{n\times 1}{k(X,\boldsymbol{x})}\,,\quad \overset{1\times 1}{k(\boldsymbol{x},\boldsymbol{x}')}+\overset{1\times n}{k(\boldsymbol{x},X)}\overset{n\times n}{C_t}\overset{n\times 1}{k(X,\boldsymbol{x}')}\right]$$

The α_i's and C_{ij}'s do not depend on the new points $\boldsymbol{x}$, $\boldsymbol{x}'$ – this is the essence of the representer theorem.

where the α_t's and C_t's are updated as follows:

$$\begin{aligned}
\overset{t\times 1}{\boldsymbol{s}_t} &= \overset{t-1\times t-1}{C_{t-1}}\ \overset{t-1\times 1}{k(X_{t-1},\boldsymbol{x}_t)}+\overset{t\times 1}{\boldsymbol{e}_t}\ ,\\
\overset{t\times 1}{\alpha_t} &= \overset{t-1\times 1}{\alpha_{t-1}}+\overset{1\times 1 t\times 1}{q_t\ \boldsymbol{s}_t}\ ,\\
\overset{t\times t}{C_t} &= \overset{t-1\times t-1}{C_{t-1}}+\overset{1\times 1 t\times 1}{r_t\ \boldsymbol{s}_t}\overset{1\times t}{\boldsymbol{s}_t^T}\ .
\end{aligned}$$

Sparse sequential GPs

- This method still scales as $O(n^3)$.
- However it does provide a bound on the evidence for hyper-parameter optimisation.
- Next we seek an $O(nm^2)$ scaling by retaining m points in an active set (subset of observations) which is a sparse parametrisation of the posterior process.

Sparse sequential inference in GPs

- To address the growth in complexity of the algorithm several approaches have been suggested, including:
  - choosing a subset of the observations – e.g. Informative Vector Machine;
  - reduced rank approximations of $\tilde{K}(\cdot,\cdot) = K_{nm}K_{mm}^{-1}K_{mn}$ – e.g. Nystrom methods;
  - partitioning the input space – e.g. Bayesian Committee Machine;
  - projecting the GP to a simpler representation – e.g. sparse, sequential GP (ssGP) method.
- The essence of the ssGP method is to project the GP at each time an observation is included, to a best GP without increasing the size of the active set.

Sparse update of parameters

- Aim: include the effect of $\boldsymbol{y}_{t+1}$ while keeping only t active points
- Update α_t and C_t so that GP_t is as close as possible to a GP which would include $\boldsymbol{y}_{t+1}$ in its set of active points.
- This is done by minimising $\text{KL}[\text{GP}_t, \text{GP}_{t+1}]$

Data recycling

- As currently explained the ssGP algorithm has a weakness; the algorithm permits only one pass through the observations.
- For non-Gaussian likelihoods this is likely to be highly suboptimal.
- The solution is the Expectation Propagation (EP) within the ssGP framework.
- EP stores an effective likelihood for each observation added.
- The observations can then be re-used by removing the effective likelihood, then re-using the observations.
- The algorithm can be shown to converge to a fixed point, and again provides a bound on the model evidence.

Sparse, sequential GPs

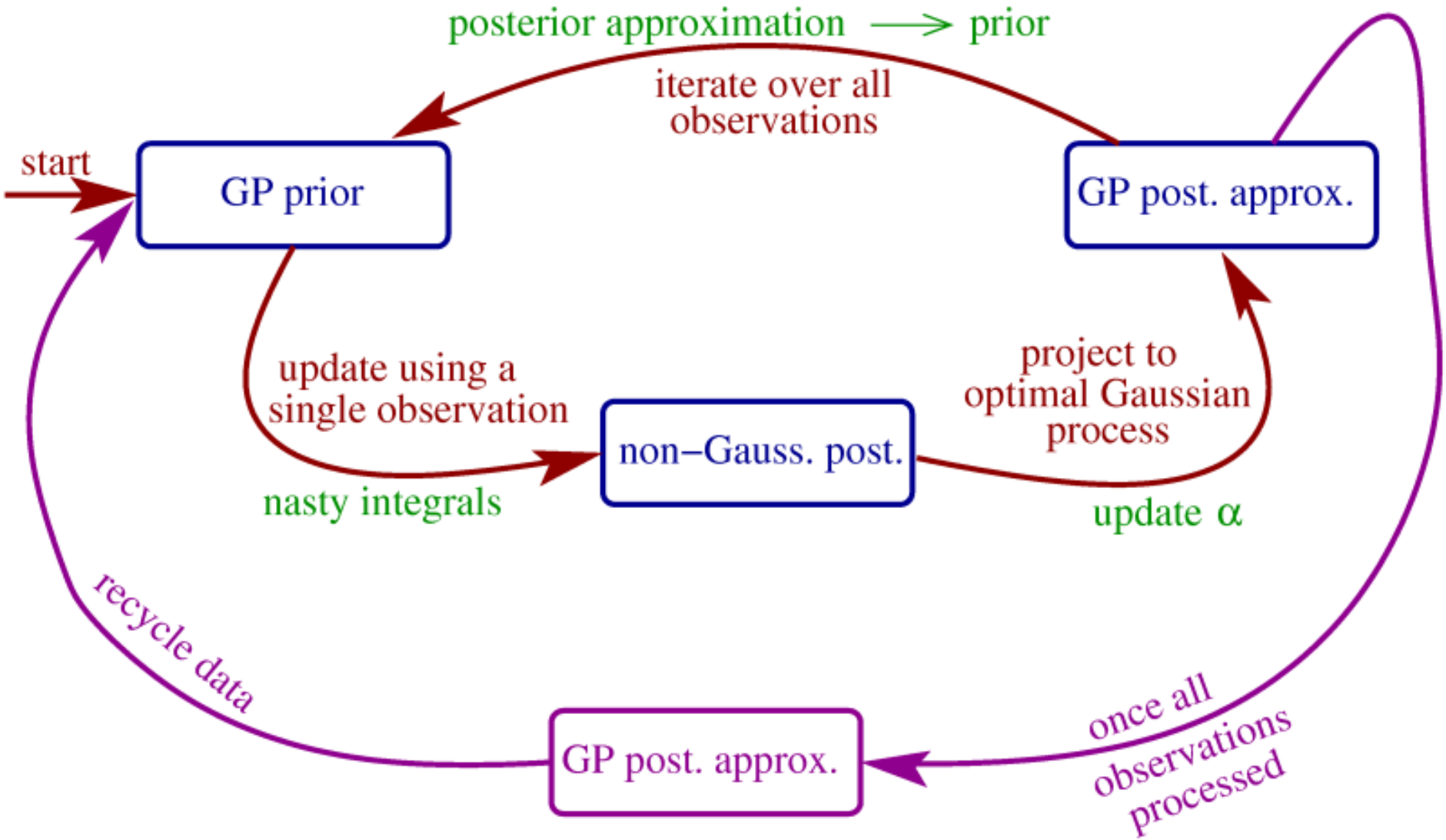


Illustration - 5 active points

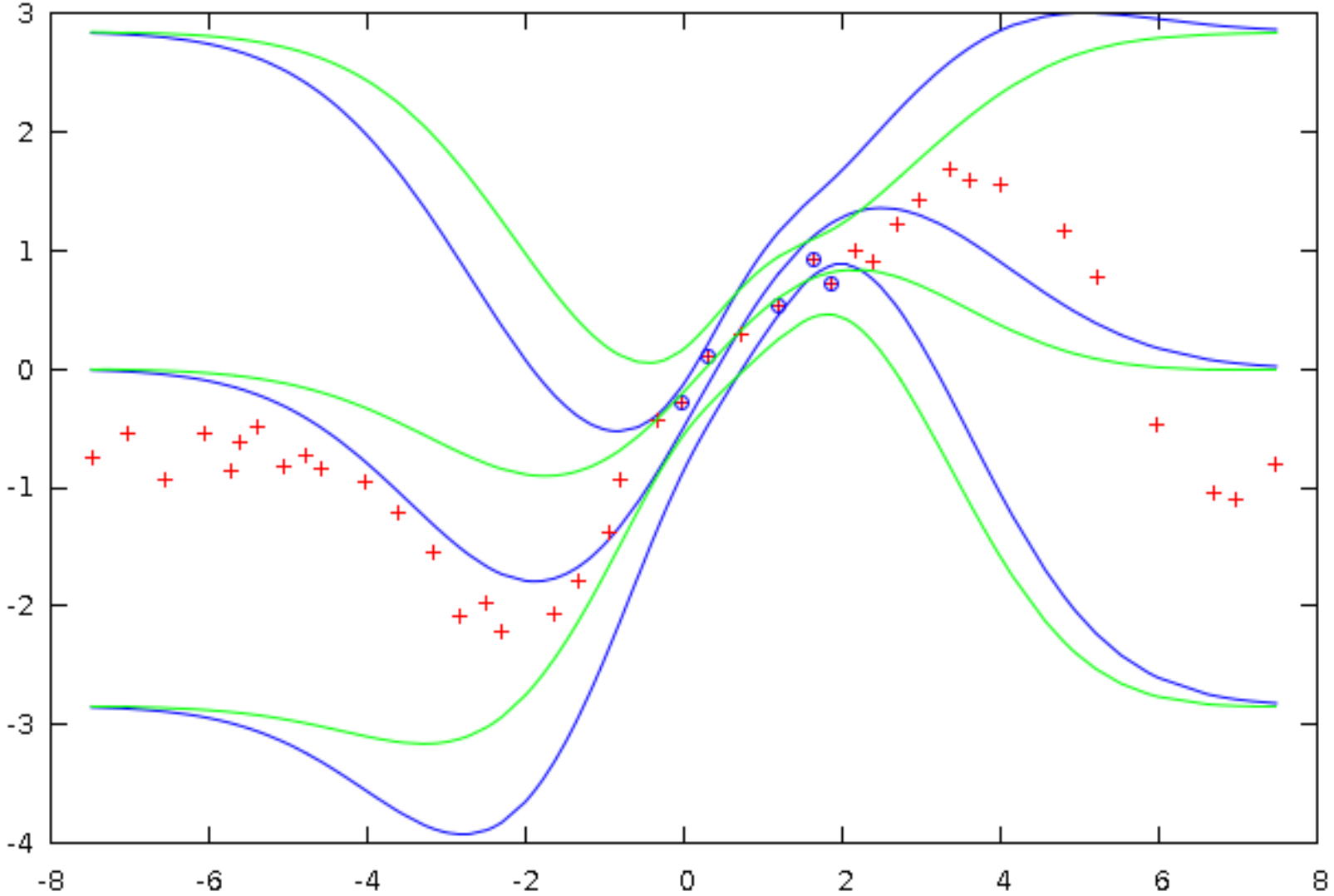


Illustration - 10 active points

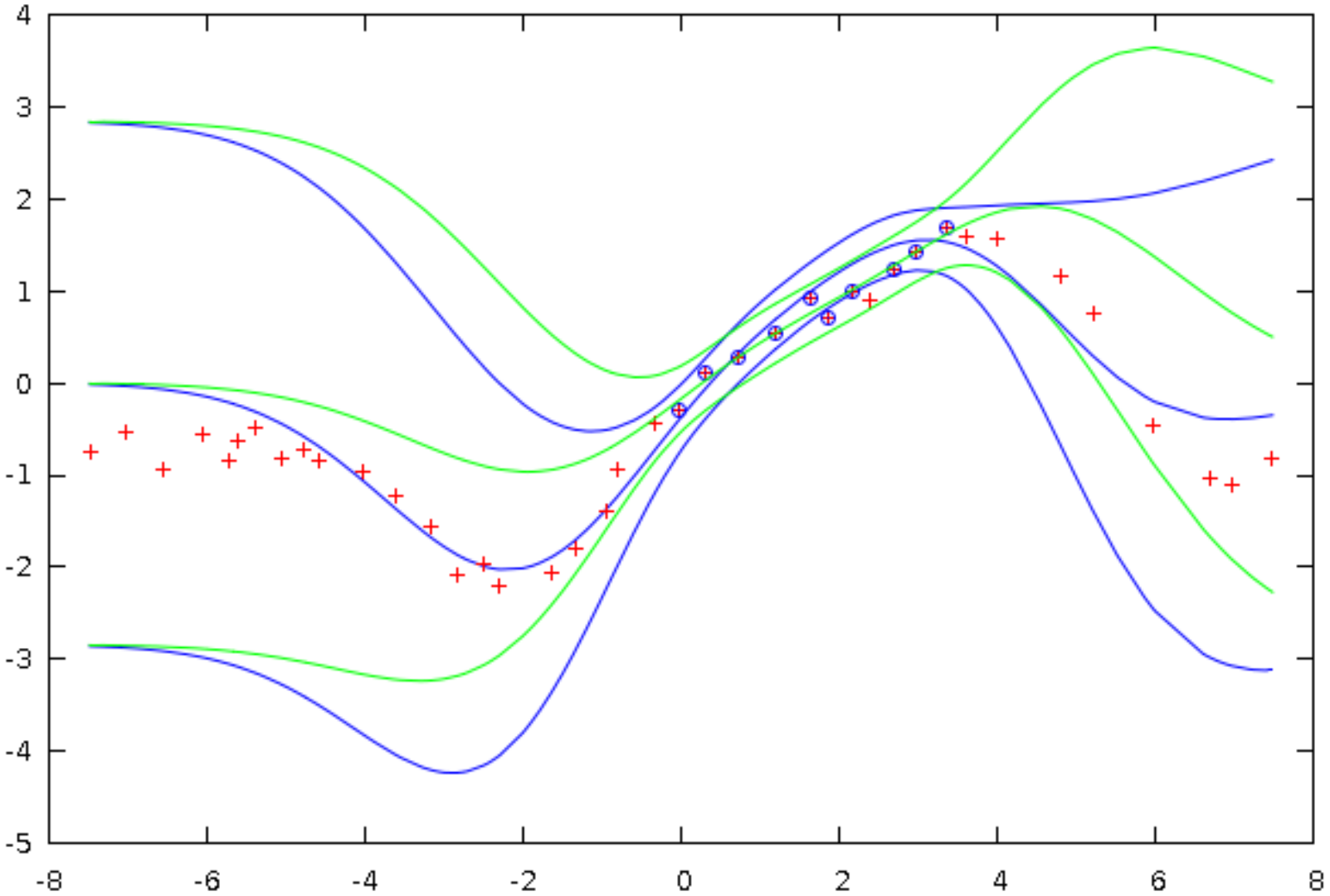


Illustration - 15 active points

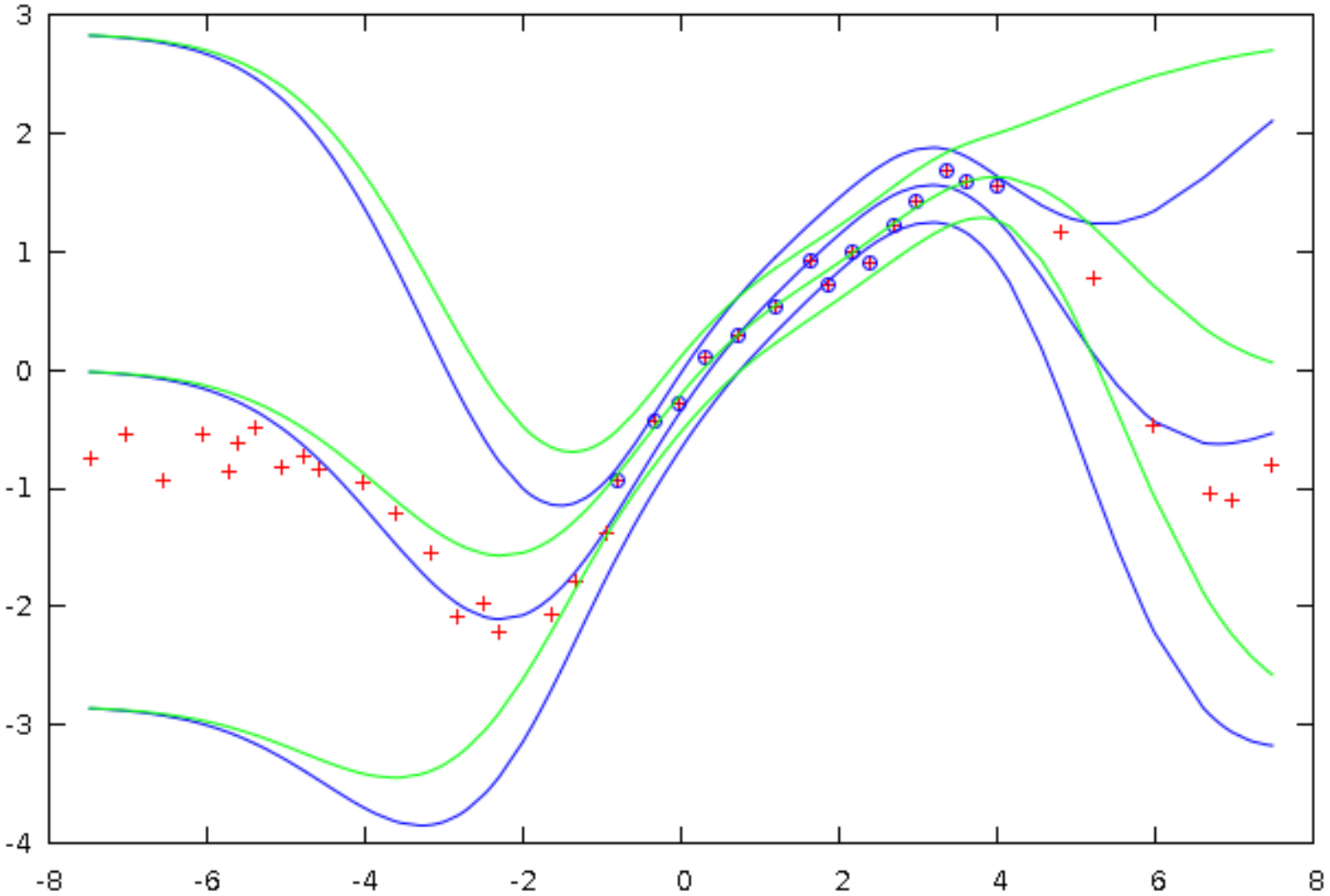


Illustration - 20 active points

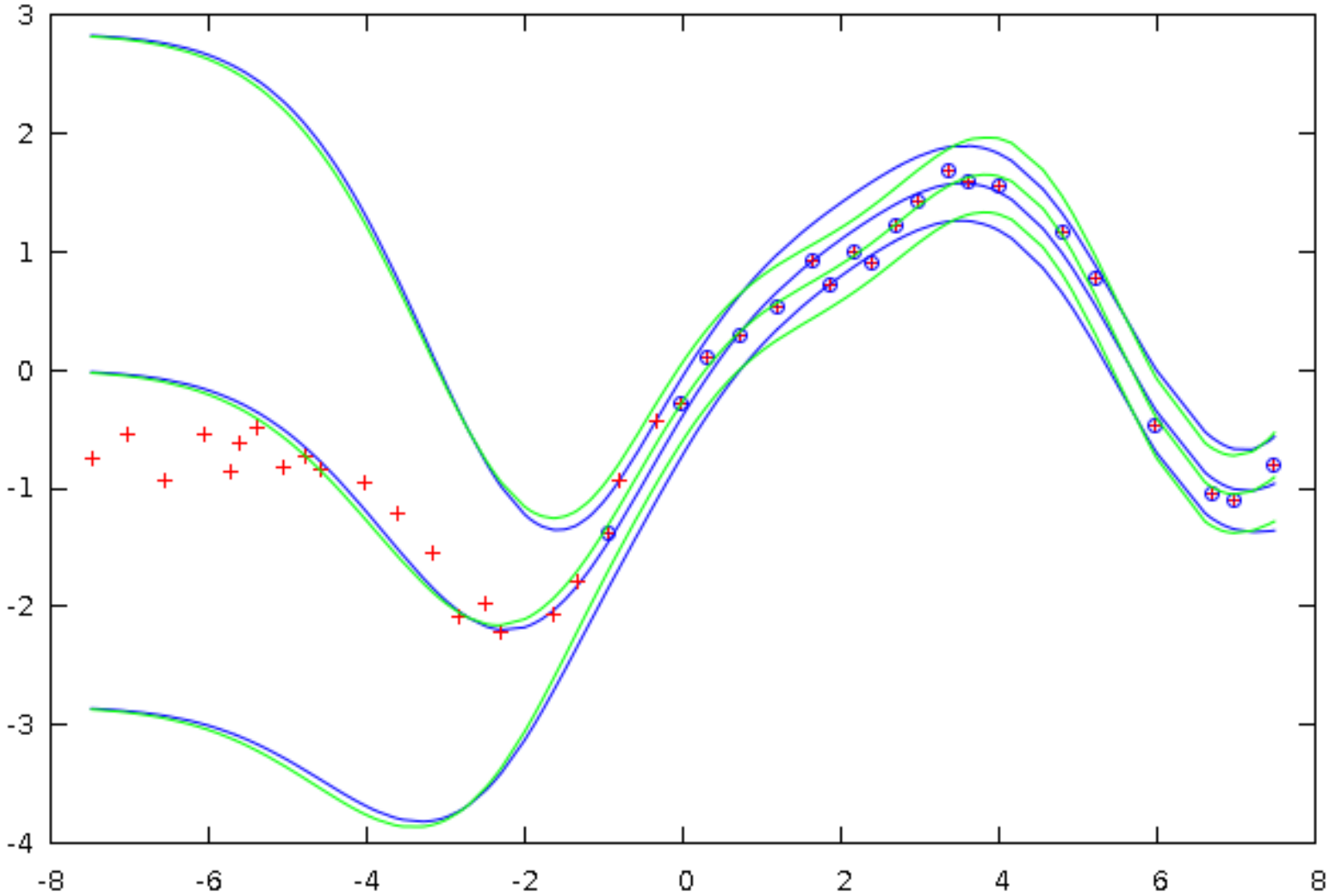


Illustration - 25 active points

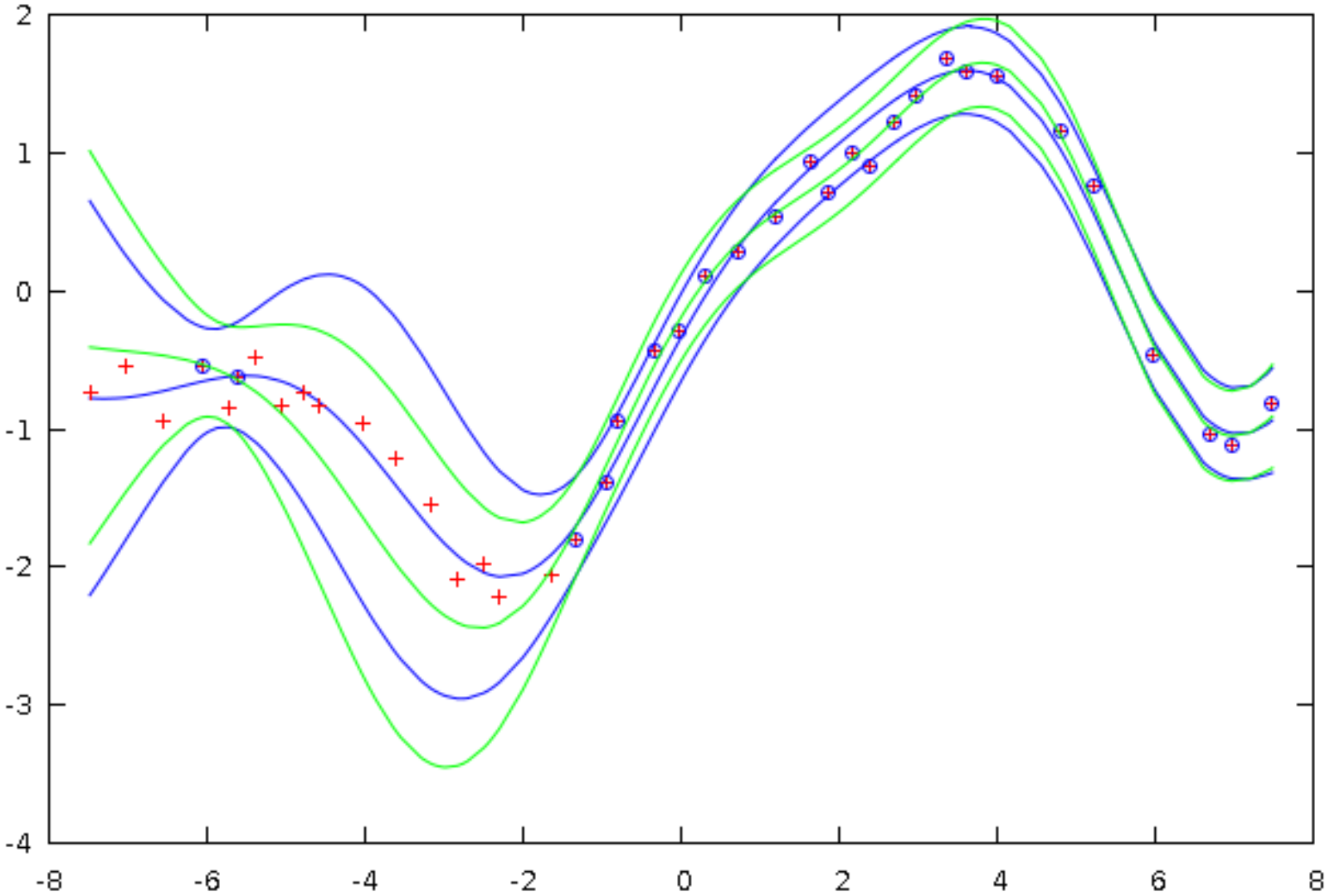


Illustration - 30 active points

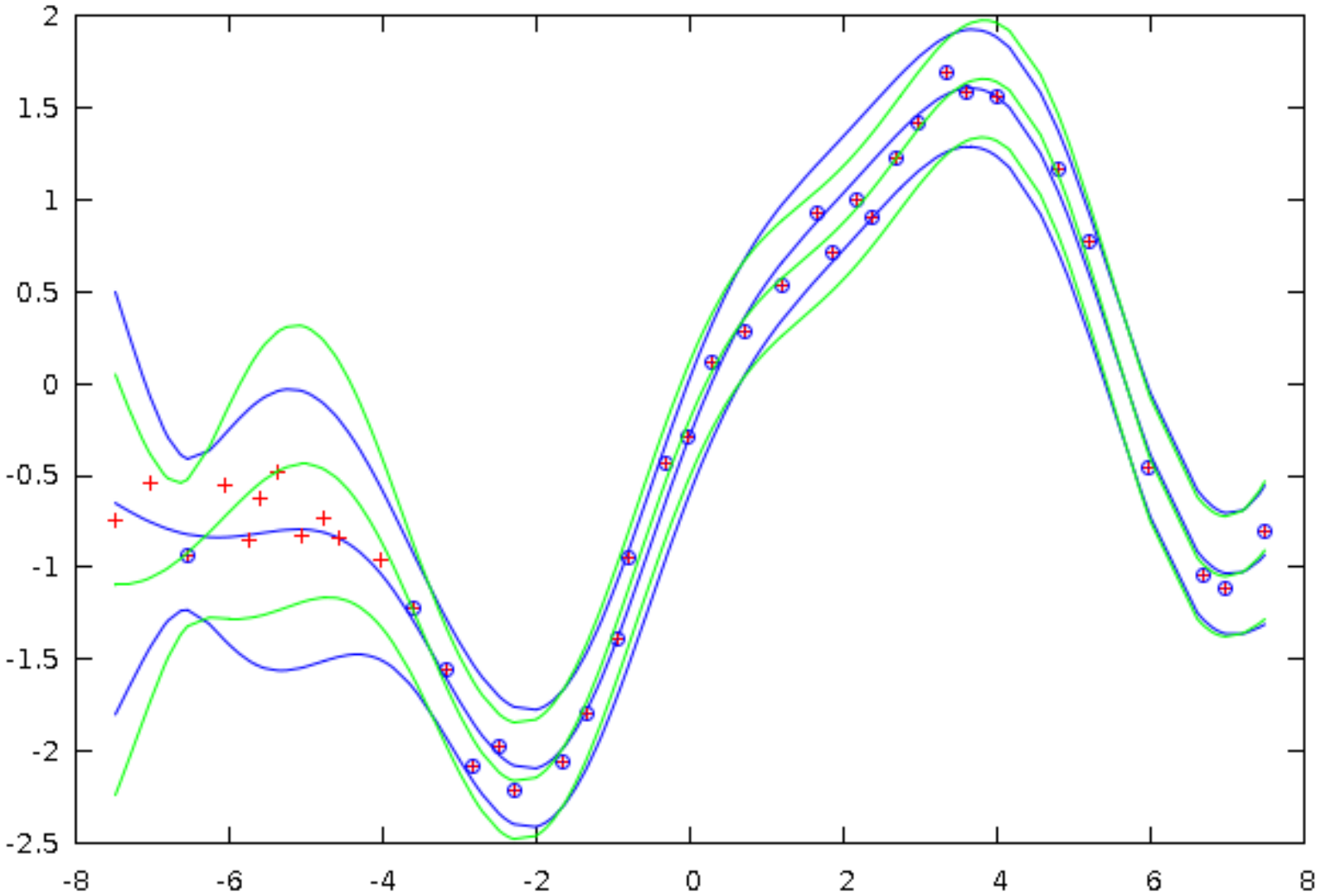


Illustration - 35 active points

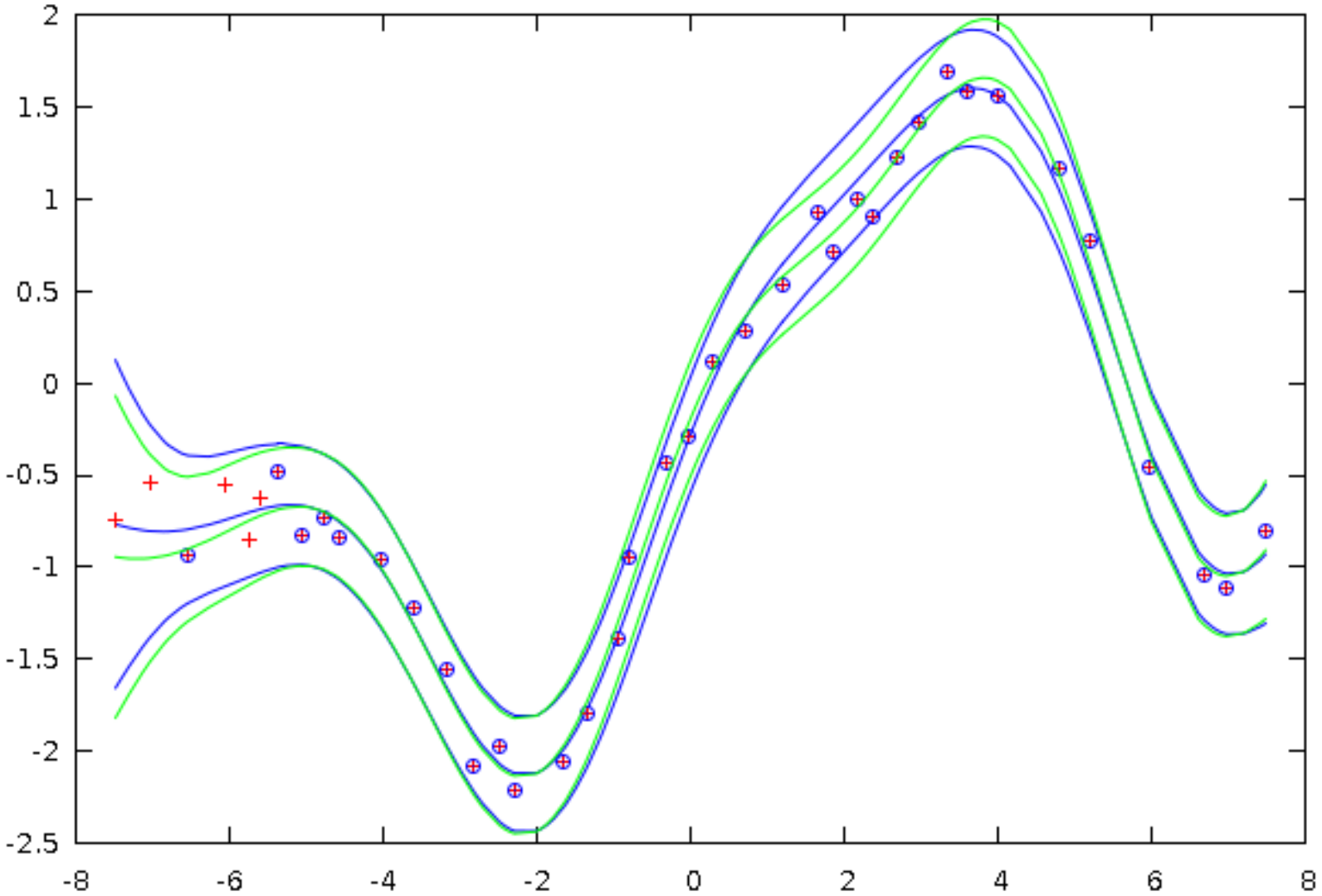
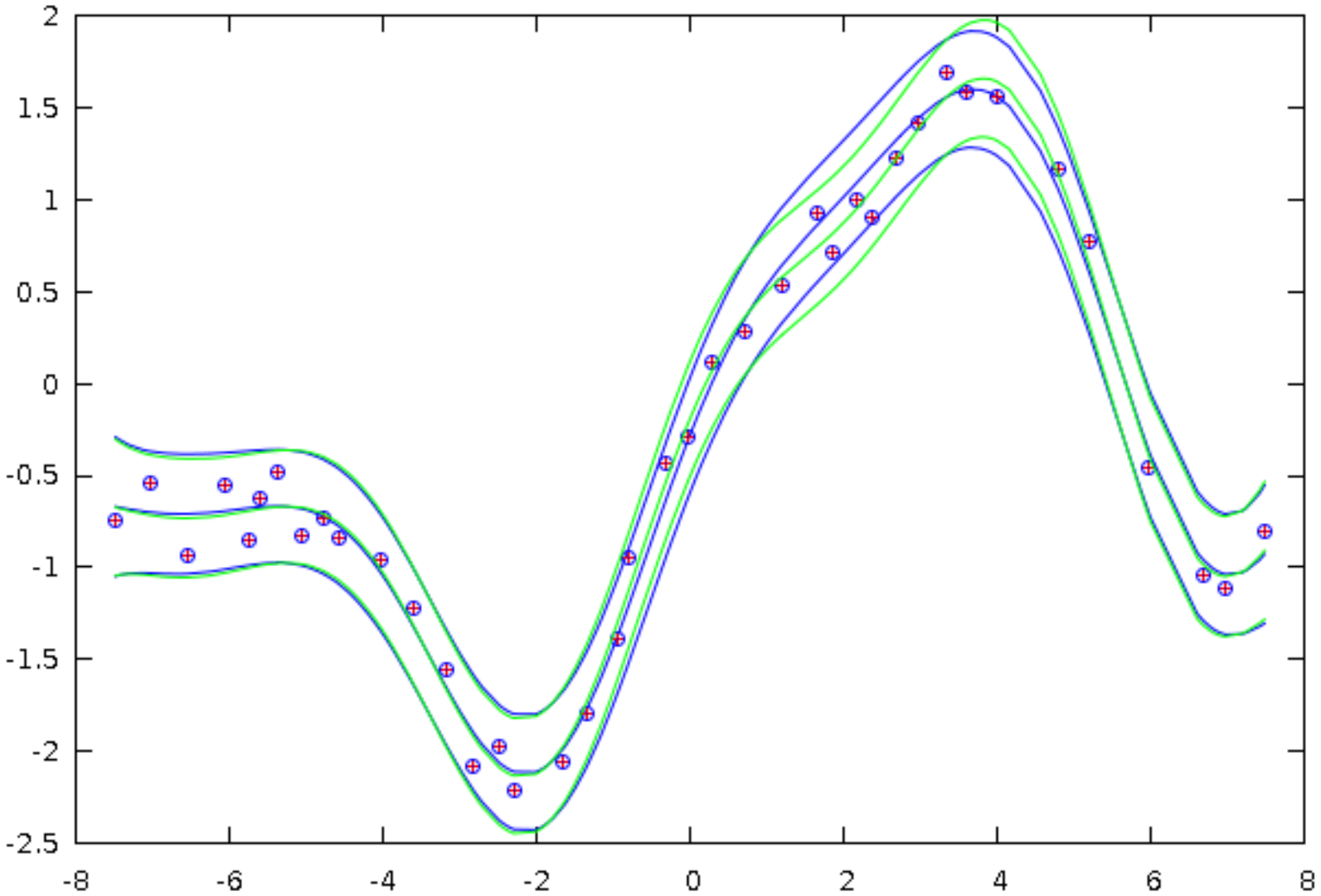


Illustration - 40 active points



Future work

- Hyperparameter estimation can be difficult, due to the interdependence between the active set and the evidence (upper bound) used in the estimation process $\Rightarrow$ importance of a good "first guess"
- So far, the framework only looked at zero-mean, single output problems $\Rightarrow$ extension to mean function and multivariate output
- Reference

[1] L. Csato and M. Opper. Sparse on-line Gaussian processes. *Neural Computation*, 14(3):641–668, 2002.