Gaussian Processes in Cosmology

Robert Crittenden
Institute of Cosmology and Gravitation
University of Portsmouth

ICG Portsmouth

Gaussian processes papers

Holsclaw et al. (2010, 2011) MCMC + GP
arXiv:1009.5443v1, 1011.3079v1, 1104.2041

Shafieloo, Kim & Linder
arXiv:1204.2272v2

Seikel, Clarkson & Smith
arXiv:1204.2832v2

Other References

D. MacKay, *Information Theory, Inference and Learning Algorithms*, Cambridge University Press (2003), chapter 45

C. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*, The MIT Press (2006)

The reconstruction problem

How do we reconstruct an unknown function given some noisy and possibly contradicting data?

- Assume some parameterization and find the best fitting (highest likelihood) parameters. (Parametric)
- Find the best spline between chosen points, implicitly applying a high frequency cutoff.
- Find the best fitting curve in a finite family of functions. (E.g. Genetic Algorithms given a grammar?)

- Allow any function, but impose a prior metric on the space of possible curves, so that some are more probable than others; then find the most probable function given the data.

Gaussian processes

The simplest metric to impose is a multi-variate Gaussian (or sometimes called a multi-variate normal, MVN.)

This can be defined by a mean function and a covariance between points:

$$\langle f(x) \rangle = m(x)$$

$$\langle (f(x) - m(x))(f(x') - m(x')) \rangle = K(x, x')$$

Given some representation of a curve (a finite number of points, or a binning) we can evaluate its prior probability:

$$\mathcal{P}(\mathbf{f}(\mathbf{x})) = \frac{1}{(2\pi)^{n/2} \det |K|} e^{-\frac{1}{2}(\mathbf{f}(\mathbf{x})-\mathbf{m}(\mathbf{x}))^T K^{-1} (\mathbf{f}(\mathbf{x})-\mathbf{m}(\mathbf{x}))} = \mathcal{N}(\mathbf{m}(\mathbf{x}), K)$$

The data model

The data inherits the correlations of the function and the correlations of the noise.

$$\mathbf{y}(\mathbf{x}) = \mathbf{f}(\mathbf{x}) + \mathbf{n}(\mathbf{x}) \qquad \langle \mathbf{n}(\mathbf{x}) \rangle = 0$$

$$\langle \mathbf{y}(\mathbf{x}) \rangle = \mathbf{m}(\mathbf{x}) \qquad \langle \mathbf{n}(\mathbf{x}) \mathbf{n}(\mathbf{x}) \rangle = C(\mathbf{x}, \mathbf{x})$$

$$\langle (\mathbf{y}(\mathbf{x}) - \mathbf{m}(\mathbf{x}))(\mathbf{y}(\mathbf{x}) - \mathbf{m}(\mathbf{x})) \rangle = K + C$$

Similarly, we can cross correlate the data with the underlying function:

$$\langle (\mathbf{y}(\mathbf{x}) - \mathbf{m}(\mathbf{x}))(\mathbf{f}(\mathbf{x}) - \mathbf{m}(\mathbf{x})) \rangle = K(\mathbf{x}, \mathbf{x})$$

Note the noise covariance is not relevant here.

The joint probability

Suppose we have data at some set of *n* points **x** and we want to correlate it with the function at some other set of *m* points **x***.

We can calculate the probability of the data vector and the function vector both being true:

$$\mathcal{P}\begin{pmatrix}\mathbf{y}(\mathbf{x})\\ \mathbf{f}(\mathbf{x}^*)\end{pmatrix} = \mathcal{N}\begin{pmatrix}\mathbf{m}(\mathbf{x})\\ \mathbf{m}(\mathbf{x}^*)\end{pmatrix}, \begin{pmatrix}K(\mathbf{x},\mathbf{x}) + C(\mathbf{x},\mathbf{x}) & K(\mathbf{x}^*,\mathbf{x})\\ K(\mathbf{x},\mathbf{x}^*) & K(\mathbf{x}^*,\mathbf{x}^*)\end{pmatrix}$$

Note that these matrices have different sizes *n x n*, *n x m*, *m x n* and *m x m*.

The conditional probability

What we really want to know is the most likely function vector given a data vector.

$$\mathcal{P}(\mathbf{f}(\mathbf{x}^*)|\mathbf{y}(\mathbf{x})) = \frac{\mathcal{P}(\mathbf{y}(\mathbf{x}), \mathbf{f}(\mathbf{x}^*))}{\mathcal{P}(\mathbf{y}(\mathbf{x}))}$$

Through a miracle of inverting a partitioned matrices, this has an analytic solution:

$$\mathcal{P}(\mathbf{f}(\mathbf{x}^*)|\mathbf{y}(\mathbf{x})) = \frac{1}{(2\pi)^{m/2}\det|A|^{1/2}} e^{-\frac{1}{2}(\mathbf{f}(\mathbf{x}^*)-\mathbf{b}(\mathbf{x}^*))^T A^{-1}(\mathbf{f}(\mathbf{x}^*)-\mathbf{b}(\mathbf{x}^*))}$$

$$\mathbf{b}(\mathbf{x}^*) = \mathbf{m}(\mathbf{x}^*) + K(\mathbf{x}^*, \mathbf{x})(K(\mathbf{x}, \mathbf{x}) + C(\mathbf{x}, \mathbf{x}))^{-1}(\mathbf{y}(\mathbf{x}) - \mathbf{m}(\mathbf{x}))$$

$$A(\mathbf{x}^*, \mathbf{x}^*) = K(\mathbf{x}^*, \mathbf{x}^*) - K(\mathbf{x}^*, \mathbf{x})(K(\mathbf{x}, \mathbf{x}) + C(\mathbf{x}, \mathbf{x}))^{-1} K(\mathbf{x}, \mathbf{x}^*)$$

The peak of conditional probability

The conditional distribution is just a normal distribution with peak given by

$$\mathbf{b}(\mathbf{x}^*) = \mathbf{m}(\mathbf{x}^*) + K(\mathbf{x}^*, \mathbf{x})(K(\mathbf{x}, \mathbf{x}) + C(\mathbf{x}, \mathbf{x}))^{-1}(\mathbf{y}(\mathbf{x}) - \mathbf{m}(\mathbf{x}))$$

and with a known covariance, A.

We can similarly calculate derivatives or integrals of the function. As linear functions of *f(x)*, they are also Normal distributed with easily calculable means and covariances.

In this way, given data of supernovae luminousities, we can reconstruct *w(z)* and its covariance. (Shafieloo, et al., Seikel et al. 2012.)

What prior to take?

The big question is what prior *(m(x), K(x,x'))* to take for the function.

The standard approach is to assume the covariance is ***stationary*** in some variable, which means that it is taken to be translation invariant:

$$K(x, x') = K(|x - x'|)$$

It is also common to take a two parameter form describing its amplitude and correlation length:

$$K(|x - x'|) = \sigma^2 e^{-(x-x')^2/\ell_c^2}$$

These hyper-parameters are found by fitting to the data, given some priors on the hyper-parameters themselves.

The prior choice

The priors are Gaussian and dependent on the data sets one chooses to use.

The hyper-parameters are chosen to fit mean properties of the data, e.g.

$$\sigma^2 \sim \langle (y-m)^2 \rangle \qquad \sigma/\ell_c \sim \langle (y-m)' \rangle$$

These then determine the higher order smoothing imposed by the prior.

This makes the choice of the mean function very important, but it can be difficult to choose this in a model independent way.

What people have done

The answers vary depending on what and how GP is applied, and the range of data one has.

Seikel et al. (2012) apply GP to the dimensionless luminousity distance *D(z)*, and assume *m(z) = 0* to infer a distribution for *w(z)*.

Shafieloo et al. (2012) apply GP to 1/H(z) and assume a mean defined by a local averaging to infer *q(z)*.

Even applied to the same data set, they could get much different results because their effective priors translate quite differently.

What people have done

Earlier work by Holsclaw et al. (2010) did the reverse of the classical Gaussian process procedure.

Instead of setting a prior on the observational space, they used MCMC to generate simulated data in *w(z)*, and then use Gaussian Processes to translate this into an observed variable. This they use to calculate a likelihood, which they feedback into the MCMC chain.

The Gaussian prior is in *w(z)* space, and its hyper-parameters are trained via MCMC's as well; these will change for every new MCMC step. However, new *w(z)* guesses are taken using a flat measure in *w(z)* space.

What we have done

In our work, (Crittenden et al. 2009, 2012, Zhao et al 2012) we use a Gaussian prior on *w(a)* but it is not formally a Gaussian Process method.

Our prior is determined by what we might expect from theoretical grounds, and tuned so those models are reconstructed with little bias and minimal variance.
We do assume a phenomenological form with translation invariance in scale factor.

Our implementation is simply to add a theoretical term to the total chi-squared, which is then implemented in a standard MCMC approach.

Comments on Gaussian Processes

Gaussian processes can yield quite different results depending on how it is implemented.

The choice of what function to fit makes a big difference.

- One could interpret data in any number of bases: SN-magnitude, *D(z), H(z), 1/H(z)*. Each of these translates into a different effective prior in *w(z)* space.

The choice of the mean function also can make a big difference, and can radically alter the derived prior covariance.

Usually applied to functions which are positive definite and monotonically increasing, so not particularly Gaussian!

Warning for consistency checks

Often people look at tests of consistency of reconstructions of different types of data to some broader framework, e.g. $D_A(z)$ versus $D_L(z)$.

If these are reconstructed with different effective priors, which don't themselves satisfy the consistency relations, then one shouldn't be surprised if the reconstructions don't satisfy the consistency relations either!

This could also relate to Genetic Algorithms; depending on the grammars chosen, there may simply not be the functional freedom to find two functions which satisfy the consistency relations.

Prior principal components

Perhaps surprisingly, the different correlation functions all have virtually the same eigenvectors, ordered in the same way. These are effectively the Fourier modes.

All that changes are the eigenvalues, but the highest frequency modes are always the most strongly constrained by the prior. This is precisely the opposite of the data constraints.