

# Truncated Covariance Matrices and Toeplitz Methods in Gaussian Processes

Amos J. Storkey

Institute for Adaptive and Neural Computation  
Division of Informatics, University of Edinburgh \*

## Abstract

Gaussian processes are a limit extension of neural networks. Standard Gaussian process techniques use a squared exponential covariance function. Here, the use of truncated covariances is proposed. Such covariances have compact support. Their use speeds up matrix inversion and increases precision. Furthermore they allow the use of speedy, memory efficient Toeplitz inversion for high dimensional grid based Gaussian process predictors.

## 1 Introduction

Gaussian process methods are a natural extension of Bayesian neural network approaches. However Gaussian processes suffer from the need to invert an  $n \times n$  matrix, where  $n$  is the number of data points. This takes  $o(n^3)$  floating point operations.

For many real life problems, there is some control over how data is collected, and this data often takes a regular form. For example data can be collected at regular time intervals or at points on a grid (e.g. video pictures). Often this structure can be used to ensure covariance matrices have a specific form. If this form happens to be Toeplitz, then the covariance inversion can be performed exactly in  $o(n^2)$  flops. Furthermore the covariance storage requirements are reduced. Here it is shown that using truncated forms of covariance allow Toeplitz methods to be used with dataspace topologies not immediately amenable to this approach.

## 2 Gaussian Processes

Let the set of points  $\{\mathbf{x}_i\}$ , denote the points in input space at which we will later receive data  $\{\mathbf{x}_i\}$   $i = 1, 2, \dots, n$  and the points  $\{\mathbf{x}_i\}$   $i = n + 1, 2, \dots, m$  at which we will be making predictions. A superscript  $D$  (for DATA) denotes an  $m$ -vector truncated to the elements  $i = 1, 2, \dots, n$ , and a superscript  $P$  (for PREDICTION) to denote an  $m$ -vector truncated to the elements  $i = n + 1, \dots, m$ .

An unknown function  $f(\mathbf{x})$  generates datum  $f_i$  at point  $\mathbf{x}_i$ , which is subject to Gaussian measurement noise  $\eta_i$  with variance  $\sigma^2$ . We combine our data (targets) and predictions (output) together into one vector  $y_i$ .

$y_i$  is defined by

$$y_i = \begin{cases} f_i + \eta_i & i = 1, 2, \dots, n \\ f_i & i = n + 1, \dots, m \end{cases}$$

So  $y_i$  combines the possible values of the data to be received (including measurement noise) with the possible values of the predictions (without measurement noise). Now  $\mathbf{y} = (y_1, y_2, \dots, y_m)$  contains all the values of interest, and so we wish to find some prior distribution over  $\mathbf{y}$ .

If we assume that the function  $f$  takes the form of some Gaussian process, then the prior over  $\mathbf{f} = (f_1, f_2, \dots, f_m)$  can be expressed as a multivariate Gaussian with covariance  $C$ . Then  $\mathbf{f}$  and  $\boldsymbol{\eta}$  are independent Gaussian random variables, and so  $\mathbf{y}$  is a sum of independent Gaussian distributed random variables, and therefore has a prior distribution of

$$P(\mathbf{y}|\mathbf{H}) = \frac{1}{Z} \exp\left(-\frac{1}{2}(\mathbf{y} - \boldsymbol{\mu})^T Q^{-1}(\mathbf{y} - \boldsymbol{\mu})\right)$$

where  $Q$  is the sum of the process covariance  $C$  and a noise covariance over the data (but not prediction) points.

---

\*This work was done while the author was at the Neural Systems Group, Imperial College.

For future use, we partition  $Q$  into the form

$$\begin{pmatrix} Q^{DD} & Q^{DP} \\ Q^{PD} & Q^{PP} \end{pmatrix}$$

where  $Q^{DD}$  is  $n \times n$  and  $Q^{PP}$  is  $(m-n) \times (m-n)$ . Note that  $Q^{PD} = (Q^{DP})^T$ .

Suppose we have now received data at points  $\mathbf{x}^D$  given by  $\mathbf{y}^D = \mathbf{y}^*$ . Then we obtain the posterior distribution  $P(\mathbf{y}^P | \mathbf{y}^D = \mathbf{y}^*, H) =$

$$\frac{1}{Z^P} \exp\left(-\frac{1}{2}(\mathbf{y}^P - \hat{\mathbf{y}})^T S^{-1}(\mathbf{y}^P - \hat{\mathbf{y}})\right)$$

where  $S = (Q^{PP} - Q^{PD}(Q^{DD})^{-1}Q^{DP})$  and  $\hat{\mathbf{y}} = Q^{PD}(Q^{DD})^{-1}\mathbf{y}^D + \boldsymbol{\mu}$ .  $Z^P$  is the relevant normalisation constant.

Note that we only need to invert matrices  $Q^{DD}$  which is  $n \times n$ . We tend only to be interested in the diagonal of  $S$ , which gives us the error bars. There is no need to invert any  $m \times m$  matrices such as  $Q$ .

We have said nothing yet of the form of the covariance function  $C$ . In fact  $C$  is used to represent some prior information about the smoothness of the function. It must be positive semidefinite, and  $C_{ij}$  must depend on variables  $\mathbf{x}_i$  and  $\mathbf{x}_j$ , and no other  $\mathbf{x}_k$ . Furthermore the mean  $\mu_i = \mu(\mathbf{x}_i)$  must be specified, and is usually taken to be zero (any known trend can be removed from the data).

Given a set of scaling hyperparameters  $\theta_1, \theta_2, r_l$ , a common choice for  $C$  is

$$C(\mathbf{x}_i, \mathbf{x}_j; H) = \theta_1 \exp\left(-\frac{1}{2}d(\mathbf{x}_i, \mathbf{x}_j)^2\right) + \theta_2 \quad (1)$$

where  $d = \sum_{l=1}^D (x_i^l - x_j^l)^2 / r_l^2$  is a Euclidean distance (in  $D$ -dimensional space). This corresponds to saying that the closer points are in input space, the more correlated their function values will be, and that the function is smooth.

### 3 Problems

One of the biggest difficulties with Gaussian processes is inverting the covariance matrices. These matrices have  $n^2$  elements where  $n$  is the sample size. Furthermore general matrix inversion is an  $o(n^3)$  process. Hence for large sample sizes, the use of Gaussian processes becomes very slow, and the memory storage requirements can be large.

Here fast inversion methods are introduced for a certain class of applications, namely those where there is some control over the data selection procedure, and where regular sampling structures are natural (e.g. time series, images, grid detectors). In these circumstances, the benefits of Toeplitz matrices can be used.

## 4 Toeplitz matrices

A Toeplitz matrix is a matrix  $A_{ij}$  of the form  $A_{ij} = A_{i+1, j+1}$  for all  $i, j$ . In other words it is constant along all diagonals.

Toeplitz matrices can be inverted in  $13n^2/4$  floating point operations using Trench's algorithm [1], [2, p199]. There are other approximate schemes which will invert Toeplitz matrices in  $o(n \ln n)$  flops: see [3, 4]. Furthermore storage requirements are at most  $n^2/4$  for the Toeplitz inverse.

## 5 Suitable topologies

In this section, we examine what topologies and data structures might be amenable for generating Toeplitz covariance matrices, and give examples of practical applications.

It is straightforward to generate Toeplitz covariance matrices on the real line using (1), given regularly spaced readings. Given a metric space and a form of covariance matrix, we call an ordered set of points which generate a Toeplitz covariance matrix a *Toeplitz ordering*. We call the unordered set of such points a *Toeplitz set*.

We call the covariance matrix *metric preserving* if each element is an invertible function of the distance between the relevant points. If the covariance matrix is of this form, then a Toeplitz ordering is independent of the choice of covariance function.

A set  $A$  induces an  $\epsilon$ -cover of a space if the set of  $\epsilon$ -balls around each point of  $A$  is a cover of the space.

We say that a family of Toeplitz sets provides a cover for a space if  $\forall \epsilon > 0$ , there is a Toeplitz set in the family which induces an  $\epsilon$  cover of the space. We use the shorthand 'Toeplitz covering' for such a family.

## 6 2D systems: problems

In  $(R^2, \text{Euclidean})$  with a metric preserving covariance function we cannot generate a family of Toeplitz orderings which provide a cover for the space.

**Proof** We generate a Toeplitz ordering. Choose  $x_0$  and  $x_1$  to be any two points in  $R^2$ . Then  $x_3$  must be such that  $d(x_3, x_2) = d(x_2, x_1)$ . In other words  $x_3$  must lie on a circle of radius  $d(x_2, x_1)$  about  $x_2$ . Choose  $x_3$  accordingly. Now  $x_4$  must be on a circle of radius  $d(x_3, x_2)$  about  $x_3$  and also on a circle of  $d(x_3, x_1)$  about  $x_2$ . There are at most two such points. From  $x_5$  on, each point is fully constrained by the previous choices. The result is either all the points lying on a circle in  $R^2$  (corresponding to one choice of  $x_4$ ) or  $(x_1, x_3, \dots)$  lying on one straight line and  $(x_2, x_4, \dots)$  lying on another (for the other choice of  $x_4$ ). Hence all possible Toeplitz orderings lie on lines or circles of  $R^2$ , implying that there is some finite  $\epsilon$  for which no Toeplitz ordering induces an  $\epsilon$  cover (there is always some point in the space which is further than  $\epsilon$  away from the line or circle on which the Toeplitz ordering lies).  $\square$

In fact covariance matrices on a grid based system take a Block-Toeplitz Toeplitz-Block (BTTB) structure. There is no equivalent of the Trench algorithm for BTTB systems, although some improvements can be made using conjugate gradient methods.

### 6.1 The surface of a cylinder

On the surface of a cylinder,  $(R \times S, \text{Euclidean})$ , things are slightly different. Toeplitz orderings will cover this space.

In order to show this, define a set of points  $x_n$  of a spiral on  $R \times S$  parametrically by  $(h, \theta) = (\alpha n, 2\pi\beta n)$  for  $n = 1, 2, 3, \dots$ . Then it is straightforward to see that this is a Toeplitz ordering in this space. We have  $d^2(x_i, x_j) = \alpha^2(n-m)^2 + 2 - 2\cos(2\pi\beta(n-m))$  which is a function of  $n-m$ , and therefore generates Toeplitz metric-preserving covariance matrices.

If we choose a set of orderings corresponding to different  $\alpha$  and  $\beta$ , then we can see that this set covers the space, because the smaller  $\alpha$  and  $\beta$  get the denser the points get.

Hence we have a useful space within which it is possible to generate Toeplitz covariance matrices.

## 7 Truncated covariances

The problem with the standard types of covariance functions used in Gaussian processes is that they tend to have an infinite range. This means that the value of the process at one point is affected (albeit only a little) by points which are a huge distance away. There is another class of covariance matrices which do not have this property. They are covariance functions with compact support. Here these have been called truncated covariances, because the covariance between two points is zero if they are greater than a certain threshold distance apart.

The first thing to note is that truncated covariance functions cannot simply be produced by cutting the tails of another type of covariance function: this would not generate positive semi-definite covariance matrices. Instead we have to go back to square one.

We can generate covariances functions (which are positive semidefinite) by convolving any symmetric kernel with itself, and so we can choose some more suitable kernels. Mackay [5] mentions the use of a top hat kernel ( $K(x, r) = 1$  if  $|x - r| < 1$ , zero elsewhere) to generate the covariance

$$C(x, y) = \begin{cases} 1 - |x - y| & \text{for } |x - y| < 1 \\ 0 & \text{elsewhere} \end{cases}$$

Clearly this covariance function is of the type we are interested in. It is zero outside the region  $|x - y| < 1$ . But it does not really satisfy all the requirements. It produces Gaussian processes which are far from smooth. This might be useful in some situations, but in general smoothness priors are more common. It would be good to find some covariance function with the same truncation properties, but which is also smooth and usable in higher dimensions.

There are a number of possibilities, but the one which will be introduced here is interesting because it is also generated from bell shaped kernels, and looks very similar to the squared exponential covariance. However it is also a truncated covariance. Consider the kernel

$$K(x) = \begin{cases} 1 + \cos(x) & -\pi < x < \pi \\ 0 & \text{otherwise} \end{cases}$$

Then, when this is convolved with itself, we

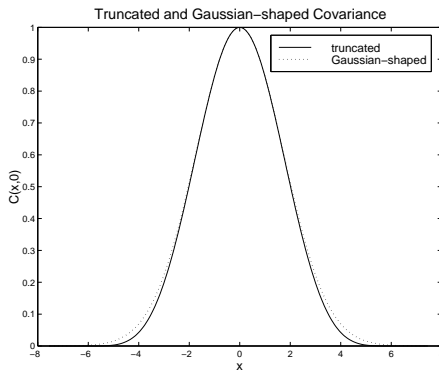


Figure 1: Comparing a Truncated covariance with a squared exponential

get the covariance function  $C(x, y) =$

$$[(2\pi - |d|)(1 + \cos |d|/2) + \frac{3}{2} \sin |d|]/(3\pi) \quad (2)$$

between  $-2\pi < x < 2\pi$  and zero outside, where  $d$  is the Euclidean or Manhattan distance  $d(x, y)$  between  $x$  and  $y$ . This satisfies the required properties. It is zero outside  $\pm 2\pi$ .

Figure 1 illustrates this covariance function, and compares it with the standard squared-exponential (Gaussian shaped) covariance function.

For higher dimensions we can define

$$C(\mathbf{x}, \mathbf{y}) = C(x_1, y_1)C(x_2, y_2) \dots C(x_D, y_D)$$

Of course not all situations are suited to covariance functions which are zero outside a certain region. However some of those with larger distance correlations can be expressed as

$$C_2(x, y) = C(x, y) + \alpha$$

where  $C$  is a truncated covariance. All of the benefits of truncated covariances are still available in these situations. This is because  $\alpha$  generates a rank 1 contribution to the resulting covariance matrix. Then the Bartlett-Sherman-Morrison-Woodbury formula can be used to express the inverse of  $C_2$  in terms of that of  $C$ .

## 7.1 Benefits

Truncated covariances have many benefits. They speed up calculations, and increase accuracy. Because truncated covariances have zeros where other covariances might have small values, many of the multiplications involved in matrix inversion and calculation of

the predictive mean and variances become trivial, error free, multiplications by zero. The end result is a much faster implementation of Gaussian process methods, which is subject to less rounding error. For further details of numerical methods which use the structure of covariances with compact support, see [6] chapter 11 and references therein.

Having dealt with truncated covariances in a more general setting, it would be good to focus more on the topic at hand: how can truncated covariances help generate Toeplitz covariance matrices in situations where they are not normally found?

In most modelling situations, we are dealing with bounded spaces. Consider for example a rectangular region of  $(R^2, \text{Euclidean})$ . This region can be mapped onto the surface of a large cylinder  $(R \times S, \text{Euclidean})$ . We know that in the limit of the cylinder becoming infinitely wide, the metric properties of the rectangular region are preserved. For large but finite cylinders, the metric properties will be slightly but insignificantly distorted.

Now consider what we gain through the use of truncated covariances. The contribution to the value of the Gaussian process at a given point comes only from those points in the immediate neighbourhood. Hence any distortion of the metric at large distances is immaterial to the results. Therefore the region of interest can be mapped onto the whole cylinder, by 'wrapping it around'. Because the covariance is of truncated form, there is little distortion of the covariance between any two points in the space: the larger, more distorted, distances do not contribute to the covariance. The only significant distortion will come from the edge effects, where one side of the region is joined to the other.

Suppose for now that the whole region of interest is mapped around the cylinder. We can then choose a set of measurement points around the cylinder which produces a Toeplitz covariance matrix (see section 6.1). This set of points corresponds to an actual set of measurement points in the original space. Given the (noisy) values of these measurement points, we can calculate the Toeplitz covariance matrix using the truncated covariance function in the cylindrical space. This can be used to infer the values at other

points in the space.

## 8 An example and details

Suppose we are interested in a 2 dimensional region. We map the whole region around the surface of a cylinder. Now we can choose the set of sampling points already mentioned in section 6.1: define a set of points  $x_n$  of a spiral on  $R \times S$  parametrically by  $(h, \theta) = (\alpha n, 2\pi\beta n)$  for  $n = 1, 2, 3, \dots$ . We can map these points back onto the original rectangle to get the required sample points in the original space. We build the covariance matrix and invert it using Toeplitz techniques.

The set of sample points which this method produces is not exactly a grid, but without much loss of accuracy we can use a grid based system. Rather than transforming the rectangle onto the cylinder surface by mapping the vertical component to the vertical component of the cylinder, and the horizontal component to the horizontal component of the cylinder, we can skew the rectangle slightly, so that after mapping onto the cylinder the end of the first line meets the beginning of the second. Then the grid system in the original space becomes a Toeplitz set in the new space. This will amount in a little loss of accuracy.

The following example illustrates this point. The top picture of figure 3 illustrates a two dimensional surface. The next gives a  $(11 \times 11)$  grid based sample of that surface. When measurement noise is added we get the third picture.

Toeplitz Gaussian process methods were used with this noisy data to try to reconstruct the original, using a prior based on the covariance given in (2). The results are given in the figure 3.

All of the work done up to this point has assumed that the hyperparameters are already known. We need to look at how the hyperparameters can be determined.

## 9 Hyperparameters

The method of determining the hyperparameters of the Gaussian process is very much the same as usual [7]. The main problem is that there will be an unwanted contribution to the likelihood from the artificial join of the two edges round the cylinder. This will bias the hyperparameters away from longer

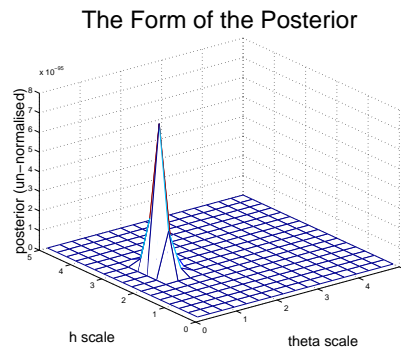


Figure 2: The posterior distribution of the the two x-length scales.

length scales. These edge contributions are likely to be relatively small compared with the overall likelihood. Figure 2 gives a plot of the posterior distribution of our example (figure 3). The peak at the true length scales is unnoticeably affected by the edge effects. The peak in the distribution is found for the values  $h = 1, \theta = 2.5$ , which are in fact the values used to plot figure 3.

However if edge effects are thought to be a problem, we can overcome this by adding a strip of width  $k$  between the two edges to be joined. Then we set  $k$  to be equal to twice the width of the covariance given the current hyperparameter. In this situation the mapping of the rectangular space to the cylinder will change as the hyperparameter width changes. When the hyperparameters correspond to larger scaling factors, then the region will be mapped to a larger cylinder. This is not a problem; In fact the effect will be beneficial in ensuring that there is a similarly low level of distortion from the flat space covariance for all hyperparameter values. The covariance matrix for all the points except those along the strip must be formed from the Toeplitz covariance matrix by using the partitioned inverse equations. This will not be too costly for small strip widths.

## 10 Higher dimensions

All these methods can also be used in higher dimensional spaces in the same ways. Toeplitz orderings can be found on spaces  $R \times S^s$  for all  $s$ . For example in four dimensions we choose the points

$$(h, \theta, \phi, \rho) = (\alpha n, 2\pi\beta n, 2\pi\gamma n, 2\pi\delta n)$$

for some  $\alpha, \beta, \gamma, \delta$ .

In three dimensions this can be visualised as a spiral winding round a torus, while the third component moves steadily along a straight line.

This method relies on the use of regular sampling points. In very high dimensions this can be problematic, as the number of sample points required to cover the space to a given accuracy increases exponentially as the sample size increases.

## 11 Conclusions

Truncated covariances can be used with Gaussian processes to represent smoothness priors, where large distance correlations are considered unlikely. Furthermore they are also the starting point for a number of efficient modelling methods. Not only do they automatically increase speed and reduce inaccuracies by increasing the number of zero multiplications, but they also enable the use of Toeplitz matrix inversion techniques with grid-based systems. The key to this approach is to provide low distortion mappings from spaces in which Toeplitz coverings cannot be found to ones where they can.

## References

- [1] W. F. Trench. An algorithm for the inversion of finite toeplitz matrices. *Journal of SIAM*, 12:515–522, 1964.
- [2] G. H. Golub and C. F. Van Loan. *Matrix Computations*. John Hopkins, third edition, 1996.
- [3] Chan R, J. Nagy, and R. Plemmons. Circulant preconditioned toeplitz least squares iterations. *SIAM Journal of Matrix Analysis and Applications*, 15:80–87, 1994.
- [4] R. Chan and M. Ng. Conjugate gradient methods for solving toeplitz systems. *SIAM Review*, 38:427–482, 1996.
- [5] D. J. C. Mackay. Gaussian processes - a replacement for supervised neural networks? In *NIPS97 Tutorial*, 1997.
- [6] Grace Wahba. *Spline Models for Observational Data*. SIAM, 1990.
- [7] C. K. I. Williams. Prediction with gaussian processes: From linear regression to linear prediction and beyond. Technical report, Neural Computing Research Group, Aston University, 1997.

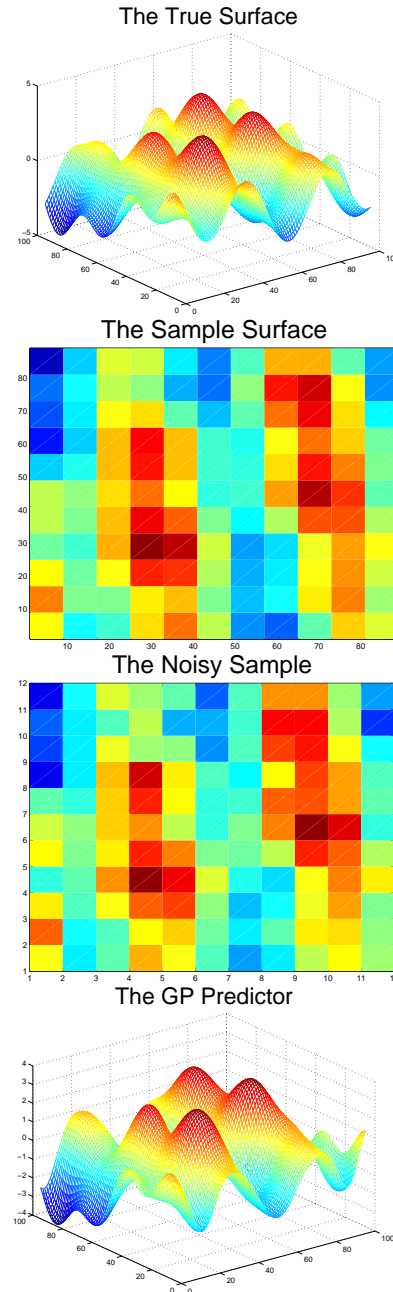


Figure 3: An example of using Toeplitz Gaussian process methods to model two dimensional systems