Chapter 1

Gaussian conditionals

A standard result shows how to condition on a subset of dimensions $\mathbf{y}_B$ of a vector $\mathbf{y}$ having a multivariate Gaussian distribution. If

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_A \\ \mathbf{y}_B \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}_A \\ \boldsymbol{\mu}_B \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_{AA} & \boldsymbol{\Sigma}_{AB} \\ \boldsymbol{\Sigma}_{BA} & \boldsymbol{\Sigma}_{BB} \end{bmatrix} \right) \tag{1.1}$$

then

$$\mathbf{y}_A | \mathbf{y}_B \sim \mathcal{N}(\boldsymbol{\mu}_A + \boldsymbol{\Sigma}_{AB}\boldsymbol{\Sigma}_{BB}^{-1}\left(\mathbf{x}_B - \boldsymbol{\mu}_B\right), \boldsymbol{\Sigma}_{AA} - \boldsymbol{\Sigma}_{AB}\boldsymbol{\Sigma}_{BB}^{-1}\boldsymbol{\Sigma}_{BA}). \tag{1.2}$$

This result can be used in the context of Gaussian process regression, where $\mathbf{y}_B = [f(\mathbf{x}_1), f(\mathbf{x}_2), \ldots, f(\mathbf{x}_N)]$ represents a set of function values observed at some subset of locations $[\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N]$, while $\mathbf{y}_A = [f(\mathbf{x}_1\star), f(\mathbf{x}_2\star), \ldots, f(\mathbf{x}_N\star)]$ represents test points whose predictive distribution we'd like to know. In this case, the necessary covariance matrices are given by:

$$\boldsymbol{\Sigma}_{AA} = k(\mathbf{X}, \mathbf{X}) \tag{1.3}$$

$$\boldsymbol{\Sigma}_{AB} = k(\mathbf{X}, \mathbf{X}^\star) \tag{1.4}$$

$$\boldsymbol{\Sigma}_{BA} = k(\mathbf{X}^\star, \mathbf{X}) \tag{1.5}$$

$$\boldsymbol{\Sigma}_{BB} = k(\mathbf{X}^\star, \mathbf{X}^\star) \tag{1.6}$$

and similarily for the mean vectors.

Chapter 2

Kernel definitions

Here we give the formulas for all one-dimensional base kernels mentioned in the thesis. Each of these formulas is multiplied by a scale factor σ_f^2, which we omit for clarity.

$$\mathrm{C}(x, x') = 1 \tag{2.1}$$

$$\mathrm{WN}(x, x') = \delta(x - x') \tag{2.2}$$

$$\mathrm{Lin}(x, x') = (x - c)(x' - c) \tag{2.3}$$

$$\mathrm{SE}(x, x') = \exp\left(-\frac{(x - x')^2}{2\ell^2}\right) \tag{2.4}$$

$$\mathrm{RQ}(x, x') = \left(1 + \frac{(x - x')^2}{2\alpha\ell^2}\right)^{-\alpha} \tag{2.5}$$

$$\mathrm{Per}(x, x') = \exp\left(-\frac{2}{\ell^2}\sin^2\left(\pi\frac{x - x'}{p}\right)\right) \tag{2.6}$$

$$\cos(x, x') = \cos\left(\frac{2\pi(x - x')}{p}\right) \tag{2.7}$$

$$\mathrm{CP}(k_1, k_2)(x, x') = \sigma(x)k_1(x, x')\sigma(x') + (1 - \sigma(x))k_2(x, x')(1 - \sigma(x')) \tag{2.8}$$

$$\boldsymbol{\sigma}(x, x') = \sigma(x)\sigma(x') \tag{2.9}$$

$$\bar{\boldsymbol{\sigma}}(x, x') = (1 - \sigma(x))(1 - \sigma(x')) \tag{2.10}$$

where $\delta_{x,x'}$ is the Kronecker delta function, I_0 is the modified Bessel function of the first kind of order zero, and other symbols are kernel parameters. $\sigma(x) = 1/1+\exp(-x)$.

Equations (2.3), (2.4) and (2.11) are plotted in **??**, and equations (2.2), (2.5) and (2.7) are plotted in **??**. Draws from GP priors with changepoint kernels are shown in **??**.

The generalized periodic kernel

Lloyd (2013) showed that the standard periodic kernel due to MacKay (1998) can be decomposed into a periodic and a constant component. He derived the equivalent periodic kernel without any constant component:

$$\mathrm{Per}(x, x') = \sigma_f^2 \frac{\exp\left(\frac{1}{\ell^2}\cos 2\pi\frac{(x-x')}{p}\right) - I_0\left(\frac{1}{\ell^2}\right)}{\exp\left(\frac{1}{\ell^2}\right) - I_0\left(\frac{1}{\ell^2}\right)} \tag{2.11}$$

He further showed that its limit as the lengthscale grows is the cosine kernel:

$$\lim_{\ell\to\infty} \mathrm{Per}(x, x') = \cos\left(\frac{2\pi(x-x')}{p}\right). \tag{2.12}$$

Separating out the constant component allows us to express negative prior covariance, as well as increasing the interpretability of the resulting models.

Chapter 3

Search operators

The model construction phase of ABCD starts with the noise kernel, WN. New kernel expressions are generated by applying search operators to the current kernel, which replace some part of the existing kernel expression with a new kernel expression.

The search used in the multidimensional regression experiments in **????** used only the following search operators:

$$\mathcal{S} \rightarrow \mathcal{S} + \mathcal{B} \tag{3.1}$$

$$\mathcal{S} \rightarrow \mathcal{S} \times \mathcal{B} \tag{3.2}$$

$$\mathcal{B} \rightarrow \mathcal{B}' \tag{3.3}$$

where $\mathcal{S}$ represents any kernel subexpression and $\mathcal{B}$ is any base kernel within a kernel expression. These search operators represent addition, multiplication and replacement. When the multiplication operator is applied to a subexpression which includes a sum of subexpressions, parentheses () are introduced. For instance, if rule (3.2) is applied to the subexpression $k_1 + k_2$, the resulting expression is $(k_1 + k_2) \times \mathcal{B}$.

Afterwards, we added several more search operators in order to speed up the search. These new operators do not change the set of possible models.

To accommodate changepoints and changewindows, we introduced the following additional operators to our search:

$$\mathcal{S} \rightarrow \mathrm{CP}(\mathcal{S}, \mathcal{S}) \tag{3.4}$$

$$\mathcal{S} \rightarrow \mathrm{CW}(\mathcal{S}, \mathcal{S}) \tag{3.5}$$

$$\mathcal{S} \rightarrow \mathrm{CW}(\mathcal{S}, \mathrm{C}) \tag{3.6}$$

$$\mathcal{S} \rightarrow \mathrm{CW}(\mathrm{C}, \mathcal{S}) \tag{3.7}$$

where C is the constant kernel. The last two operators result in a kernel only applying outside, or within, a certain region.

To allow the search to simplify existing expressions, we introduced the following operators:

$$\mathcal{S} \rightarrow \mathcal{B} \tag{3.8}$$

$$\mathcal{S} + \mathcal{S}' \rightarrow \mathcal{S} \tag{3.9}$$

$$\mathcal{S} \times \mathcal{S}' \rightarrow \mathcal{S} \tag{3.10}$$

where $\mathcal{S}'$ represents any other kernel expression. We also introduced the operator

$$\mathcal{S} \rightarrow \mathcal{S} \times (\mathcal{B} + \mathrm{C}) \tag{3.11}$$

Which allows a new base kernel to be added along with the constant kernel, for cases when multiplying by a base kernel by itself would restrict the model too much.

Chapter 4

Example automatically generated report

The following pages of this appendix contain an entire automatically-generated report, run on a dataset measuring annual solar irradiation data from 1610 to 2011. This dataset was previously analyzed by Lean et al. (1995).

The structure search was run using the ABCD-interpretable variant, with base kernels SE, Lin, C, Per, $\boldsymbol{\sigma}$, and WN.

Other example reports can be found at `mlg.eng.cam.ac.uk/Lloyd/abcdoutput/`, including anlyses of wheat prices, temperature records, call centre volumes, radio interference, gas production, unemployment, number of births, and wages over time.

1 Executive summary

The raw data and full model posterior with extrapolations are shown in figure 1.

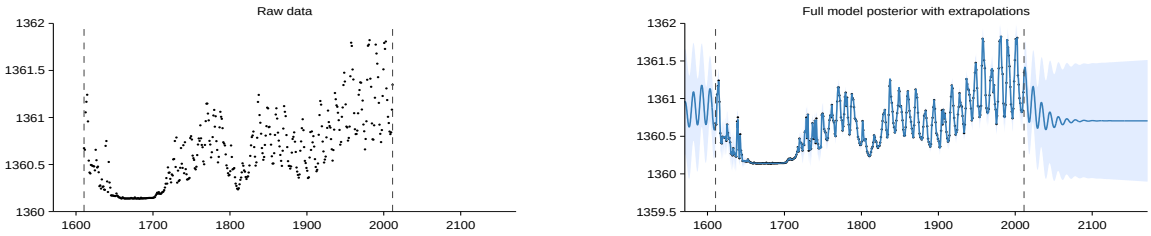

Figure 1: Raw data (left) and model posterior with extrapolation (right)

The structure search algorithm has identified nine additive components in the data. The first 4 additive components explain 92.3% of the variation in the data as shown by the coefficient of determination (R^2) values in table 1. The first 8 additive components explain 99.2% of the variation in the data. After the first 5 components the cross validated mean absolute error (MAE) does not decrease by more than 0.1%. This suggests that subsequent terms are modelling very short term trends, uncorrelated noise or are artefacts of the model or search procedure. Short summaries of the additive components are as follows:

- A constant.
- A constant. This function applies from 1644 until 1713.
- A smooth function. This function applies until 1644 and from 1719 onwards.
- An approximately periodic function with a period of 10.8 years. This function applies until 1644 and from 1719 onwards.
- A rapidly varying smooth function. This function applies until 1644 and from 1719 onwards.
- Uncorrelated noise.
- A rapidly varying smooth function with marginal standard deviation increasing linearly away from 1843. This function applies from 1751 onwards.
- A rapidly varying smooth function. This function applies until 1644 and from 1719 until 1751.
- A constant. This function applies from 1713 until 1719.

#	R^2 (%)	ΔR^2 (%)	Residual R^2 (%)	Cross validated MAE	Reduction in MAE (%)
-	-	-	-	1360.65	-
1	0.0	0.0	0.0	0.33	100.0
2	35.3	35.3	35.3	0.23	29.4
3	72.5	37.2	57.5	0.18	20.7
4	92.3	19.9	72.2	0.15	16.4
5	97.8	5.5	71.4	0.15	0.4
6	97.8	0.0	0.2	0.15	0.0
7	98.4	0.5	24.8	0.15	-0.0
8	99.2	0.8	50.7	0.15	-0.0
9	100.0	0.8	100.0	0.15	-0.0

Table 1: Summary statistics for cumulative additive fits to the data. The residual coefficient of determination (R^2) values are computed using the residuals from the previous fit as the target values; this measures how much of the residual variance is explained by each new component. The mean absolute error (MAE) is calculated using 10 fold cross validation with a contiguous block design; this measures the ability of the model to interpolate and extrapolate over moderate distances. The model is fit using the full data so the MAE values cannot be used reliably as an estimate of out-of-sample predictive performance.

2 Detailed discussion of additive components

2.1 Component 1 : A constant

This component is constant.

This component explains 0.0% of the total variance. The addition of this component reduces the cross validated MAE by 100.0% from 1360.6 to 0.3.

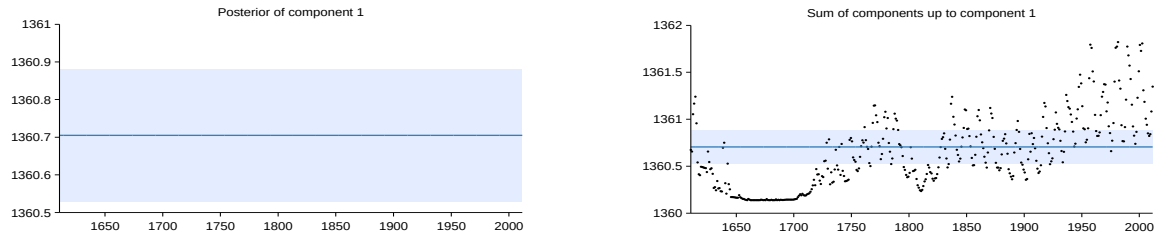

Figure 2: Posterior of component 1 (left) and the posterior of the cumulative sum of components with data (right)

2.2 Component 2 : A constant. This function applies from 1644 until 1713

This component is constant. This component applies from 1644 until 1713.

This component explains 35.3% of the residual variance; this increases the total variance explained from 0.0% to 35.3%. The addition of this component reduces the cross validated MAE by 29.42% from 0.33 to 0.23.

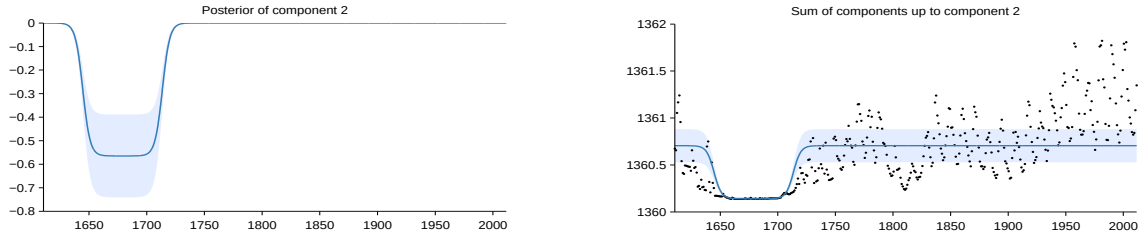

Figure 3: Posterior of component 2 (left) and the posterior of the cumulative sum of components with data (right)

2.3 Component 3 : A smooth function. This function applies until 1644 and from 1719 onwards

This component is a smooth function with a typical lengthscale of 21.9 years. This component applies until 1644 and from 1719 onwards.

This component explains 57.5% of the residual variance; this increases the total variance explained from 35.3% to 72.5%. The addition of this component reduces the cross validated MAE by 20.66% from 0.23 to 0.18.

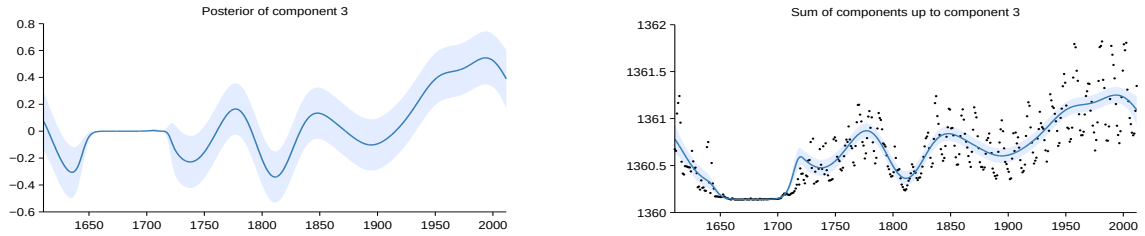

Figure 4: Posterior of component 3 (left) and the posterior of the cumulative sum of components with data (right)

2.4 Component 4 : An approximately periodic function with a period of 10.8 years. This function applies until 1644 and from 1719 onwards

This component is approximately periodic with a period of 10.8 years. Across periods the shape of the function varies smoothly with a typical lengthscale of 33.2 years. The shape of the function within each period has a typical lengthscale of 12.6 years. This component applies until 1644 and from 1719 onwards.

This component explains 72.2% of the residual variance; this increases the total variance explained from 72.5% to 92.3%. The addition of this component reduces the cross validated MAE by 16.42% from 0.18 to 0.15.

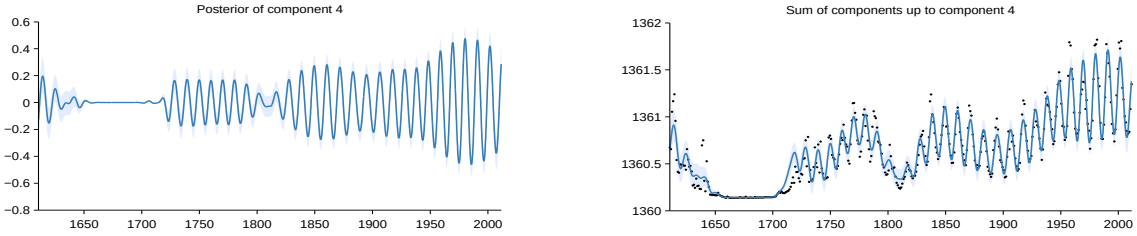

Figure 5: Posterior of component 4 (left) and the posterior of the cumulative sum of components with data (right)

2.5 Component 5 : A rapidly varying smooth function. This function applies until 1644 and from 1719 onwards

This function is a rapidly varying but smooth function with a typical lengthscale of 1.2 years. This component applies until 1644 and from 1719 onwards.

This component explains 71.4% of the residual variance; this increases the total variance explained from 92.3% to 97.8%. The addition of this component reduces the cross validated MAE by 0.41% from 0.15 to 0.15.

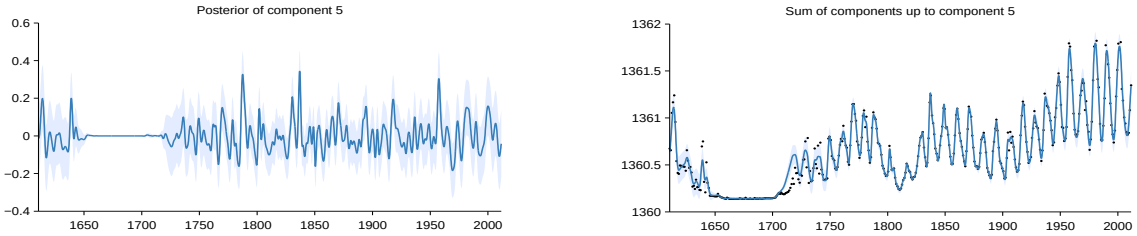

Figure 6: Posterior of component 5 (left) and the posterior of the cumulative sum of components with data (right)

2.6 Component 6 : Uncorrelated noise

This component models uncorrelated noise.

This component explains 0.2% of the residual variance; this increases the total variance explained from 97.8% to 97.8%. The addition of this component reduces the cross validated MAE by 0.00% from 0.15 to 0.15. This component explains residual variance but does not improve MAE which suggests that this component describes very short term patterns, uncorrelated noise or is an artefact of the model or search procedure.

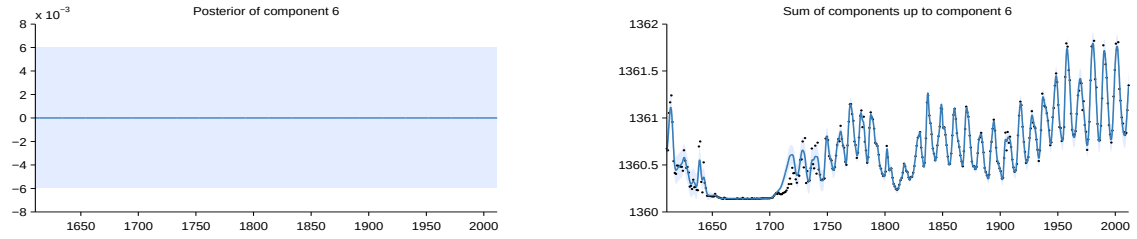

Figure 7: Posterior of component 6 (left) and the posterior of the cumulative sum of components with data (right)

2.7 Component 7 : A rapidly varying smooth function with marginal standard deviation increasing linearly away from 1843. This function applies from 1751 onwards

This function is a rapidly varying but smooth function with a typical lengthscale of 3.1 months. The marginal standard deviation of the function increases linearly away from 1843. This component applies from 1751 onwards.

This component explains 24.8% of the residual variance; this increases the total variance explained from 97.8% to 98.4%. The addition of this component increases the cross validated MAE by 0.00% from 0.15 to 0.15. This component explains residual variance but does not improve MAE which suggests that this component describes very short term patterns, uncorrelated noise or is an artefact of the model or search procedure.

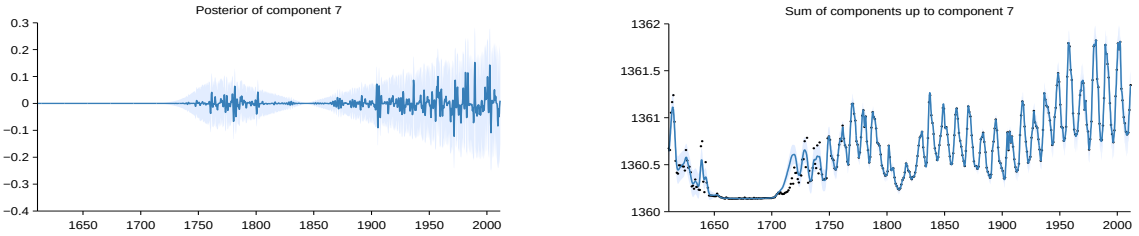

Figure 8: Posterior of component 7 (left) and the posterior of the cumulative sum of components with data (right)

2.8 Component 8 : A rapidly varying smooth function. This function applies until 1644 and from 1719 until 1751

This function is a rapidly varying but smooth function with a typical lengthscale of 3.1 months. This component applies until 1644 and from 1719 until 1751.

This component explains 50.7% of the residual variance; this increases the total variance explained from 98.4% to 99.2%. The addition of this component increases the cross validated MAE by 0.00% from 0.15 to 0.15. This component explains residual variance but does not improve MAE which suggests that this component describes very short term patterns, uncorrelated noise or is an artefact of the model or search procedure.

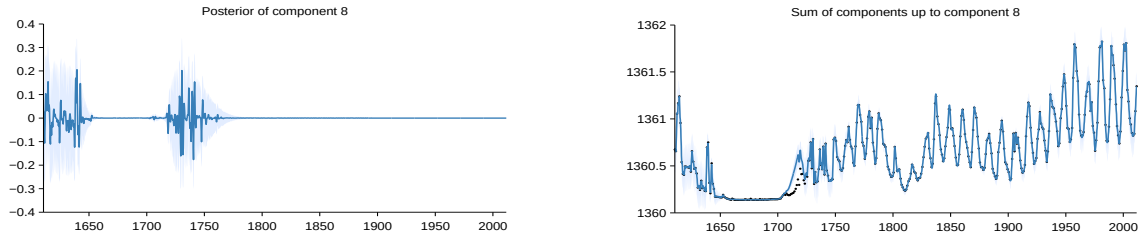

Figure 9: Posterior of component 8 (left) and the posterior of the cumulative sum of components with data (right)

2.9 Component 9 : A constant. This function applies from 1713 until 1719

This component is constant. This component applies from 1713 until 1719.

This component explains 100.0% of the residual variance; this increases the total variance explained from 99.2% to 100.0%. The addition of this component increases the cross validated MAE by 0.01% from 0.15 to 0.15. This component explains residual variance but does not improve MAE which suggests that this component describes very short term patterns, uncorrelated noise or is an artefact of the model or search procedure.

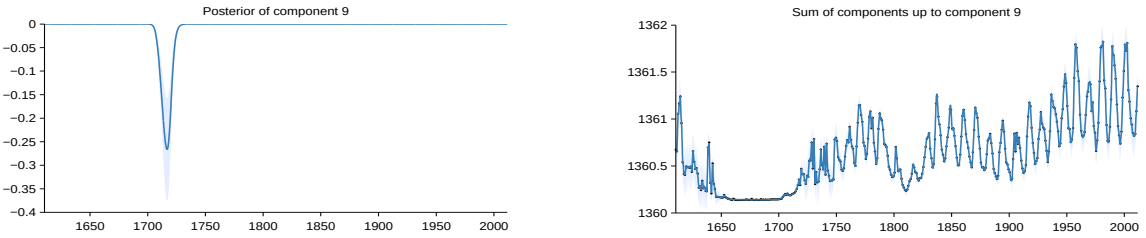

Figure 10: Posterior of component 9 (left) and the posterior of the cumulative sum of components with data (right)

3 Extrapolation

Summaries of the posterior distribution of the full model are shown in figure 11. The plot on the left displays the mean of the posterior together with pointwise variance. The plot on the right displays three random samples from the posterior.

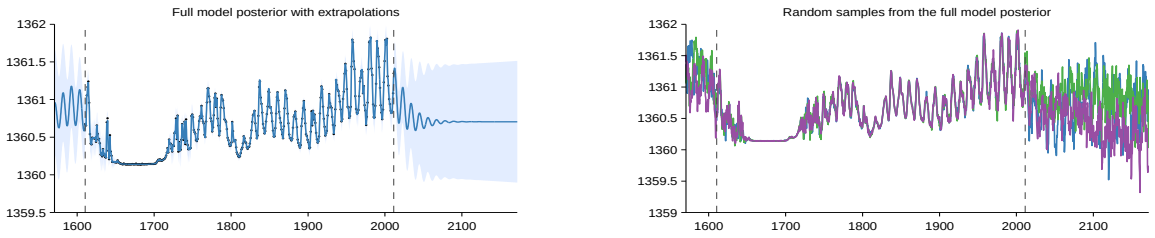

Figure 11: Full model posterior. Mean and pointwise variance (left) and three random samples (right)

Chapter 5

Inference in the warped mixture model

Detailed definition of model

The iWMM assumes that the latent density is an infinite mixture of Gaussians:

$$p(\mathbf{x}) = \sum_{c=1}^{\infty} \lambda_c \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_c, \mathbf{R}_c^{-1}) \tag{5.1}$$

where λ_c, $\boldsymbol{\mu}_c$ and $\mathbf{R}_c$ is the mixture weight, mean, and precision matrix of the c^{th} mixture component. We place a conjugate Gaussian-Wishart priors on the Gaussian parameters $\{\boldsymbol{\mu}_c, \mathbf{R}_c\}$:

$$p(\boldsymbol{\mu}_c, \mathbf{R}_c) = \mathcal{N}(\boldsymbol{\mu}_c|\mathbf{u}, (r\mathbf{R}_c)^{-1})\mathcal{W}(\mathbf{R}_c|\mathbf{S}^{-1}, \nu), \tag{5.2}$$

where $\mathbf{u}$ is the mean of $\boldsymbol{\mu}_c$, r is the relative precision of $\boldsymbol{\mu}_c$, $\mathbf{S}^{-1}$ is the scale matrix for $\mathbf{R}_c$, and ν is the number of degrees of freedom for $\mathbf{R}_c$. The Wishart distribution is defined as:

$$\mathcal{W}(\mathbf{R}|\mathbf{S}^{-1}, \nu) = \frac{1}{G}|\mathbf{R}|^{\frac{\nu-Q-1}{2}} \exp\left(-\frac{1}{2}\text{tr}(\mathbf{SR})\right), \tag{5.3}$$

where G is the normalizing constant.

Because we use conjugate Gaussian-Wishart priors for the parameters of the Gaussian mixture components, we can analytically integrate out those parameters given the assignments of points to components. Let z_n be the assignment of the n^{th} point. The prior probability of latent coordinates $\mathbf{X}$ given latent cluster assignments $\mathbf{Z} = (\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_N)$

factorizes over clusters, and can be obtained in closed-form by integrating out the Gaussian parameters $\{\boldsymbol{\mu}_c, \mathbf{R}_c\}$ to give:

$$p(\mathbf{X}|\mathbf{Z}, \mathbf{S}, \nu, r) = \prod_{c=1}^{\infty} \pi^{-\frac{N_c Q}{2}} \frac{r^{Q/2}}{r_c^{Q/2}} \frac{|\mathbf{S}|^{\nu/2}}{|\mathbf{S}_c|^{\nu_c/2}} \times \prod_{q=1}^{Q} \frac{\Gamma\left(\frac{\nu_c+1-q}{2}\right)}{\Gamma\left(\frac{\nu+1-q}{2}\right)}, \tag{5.4}$$

where N_c is the number of data points assigned to the c^{th} component, $\Gamma(\cdot)$ is the Gamma function, and

$$r_c = r + N_c, \qquad \nu_c = \nu + N_c, \quad \mathbf{u}_c = \frac{r\mathbf{u} + \sum_{n:z_n=c} \mathbf{x}_n}{r + N_c}, \tag{5.5}$$

$$\text{and} \qquad \mathbf{S}_c = \mathbf{S} + \sum_{n:z_n=c} \mathbf{x}_n \mathbf{x}_n^{\mathsf{T}} + r\mathbf{u}\mathbf{u}^{\mathsf{T}} - r_c \mathbf{u}_c \mathbf{u}_c^{\mathsf{T}}, \tag{5.6}$$

are the posterior Gaussian-Wishart parameters of the c^{th} component (Murphy, 2007).

To model the cluster assignments, we use a Dirichlet process (MacEachern and Müller, 1998) with concentration parameter η. Under a Dirichlet process prior, the probability of observing a particular cluster assignment matrix $\mathbf{Z}$ depends only on the partition induced, and is given by the Chinese restaurant process:

$$p(\mathbf{Z}|\eta) = \frac{\Gamma(\eta)\eta^C}{\Gamma(\eta + N)} \prod_{c=1}^{C} \Gamma(N_c) \tag{5.7}$$

where C is the number of components for which $N_c > 0$, and N is the total number of datapoints.

The joint distribution of observed coordinates, latent coordinates, and cluster assignments is given by

$$p(\mathbf{Y}, \mathbf{X}, \mathbf{Z}|\boldsymbol{\theta}, \boldsymbol{S}, \nu, \mathbf{u}, r, \eta) = p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\theta}) p(\mathbf{X}|\mathbf{Z}, \boldsymbol{S}, \nu, \mathbf{u}, r) p(\mathbf{Z}|\eta), \tag{5.8}$$

where the factors in the right hand side can be calculated by equations (**??**), (5.4) and (5.7), respectively.

Details of inference

After analytically integrating out the parameters of the Gaussian mixture components, the only remaining variables to infer are the latent points $\mathbf{X}$, the cluster assignments $\mathbf{Z}$, and the kernel parameters $\boldsymbol{\theta}$. We'll estimate the posterior over these parameters using Markov chain Monte Carlo. In particular, we'll alternate between collapsed Gibbs

sampling of each row of $\mathbf{Z}$, and Hamiltonian Monte Carlo sampling of $\mathbf{X}$ and $\boldsymbol{\theta}$.

First, we explain collapsed Gibbs sampling for the cluster assignments $\mathbf{Z}$. Given a sample of $\mathbf{X}$, $p(\mathbf{Z}|\mathbf{X}, \mathbf{S}, \nu, \mathbf{u}, r, \eta)$ does not depend on $\mathbf{Y}$. This lets us resample cluster assignments, integrating out the iGMM likelihood in closed form. Given the current state of all but one latent component z_n, a new value for z_n is sampled with the following probability:

$$p(z_n = c|\mathbf{X}, \mathbf{Z}_{\backslash n}, \boldsymbol{S}, \nu, \mathbf{u}, r, \eta) \propto \begin{cases} N_{c\backslash n} \cdot p(\mathbf{x}_n|\mathbf{X}_{c\backslash n}, \boldsymbol{S}, \nu, \mathbf{u}, r) & \text{existing components} \\ \eta \cdot p(\mathbf{x}_n|\boldsymbol{S}, \nu, \mathbf{u}, r) & \text{a new component} \end{cases} \tag{5.9}$$

where $\mathbf{X}_c = \{\mathbf{x}_n|z_n = c\}$ is the set of latent coordinates assigned to the c^{th} component, and $\backslash n$ represents the value or set when excluding the n^{th} data point. We can analytically calculate $p(\mathbf{x}_n|\mathbf{X}_{c\backslash n}, \boldsymbol{S}, \nu, \mathbf{u}, r)$ as follows:

$$p(\mathbf{x}_n|\mathbf{X}_{c\backslash n}, \boldsymbol{S}, \nu, \mathbf{u}, r) = \pi^{-\frac{N_{c\backslash n}Q}{2}} \frac{r_{c\backslash n}^{Q/2} \left|\mathbf{S}_{c\backslash n}\right|^{\nu_{c\backslash n}/2}}{r_{c\backslash n}'^{Q/2} \left|\mathbf{S}'_{c\backslash n}\right|^{\nu'_{c\backslash n}/2}} \times \prod_{d=1}^{Q} \frac{\Gamma\left(\frac{\nu'_{c\backslash n}+1-d}{2}\right)}{\Gamma\left(\frac{\nu_{c\backslash n}+1-d}{2}\right)}, \tag{5.10}$$

where r'_c, ν'_c, $\mathbf{u}'_c$ and $\mathbf{S}'_c$ represent the posterior on Gaussian-Wishart parameters of the c^{th} component when the n^{th} data point has been assigned to it. We efficiently calculate the determinant by using the rank-one Cholesky update. A special case of equation (5.10) gives the likelihood for a new component: $p(\mathbf{x}_n|\boldsymbol{S}, \nu, \mathbf{u}, r)$.

Gradients for Hamiltonian Monte Carlo

Hamiltonian Monte Carlo (HMC) sampling of $\mathbf{X}$ from posterior $p(\mathbf{X}|\mathbf{Z}, \mathbf{Y}, \boldsymbol{\theta}, \boldsymbol{S}, \nu, \mathbf{u}, r)$, requires computing the gradient of the log-unnormalized-posterior with respect to $\mathbf{X}$:

$$\frac{\partial}{\partial \mathbf{X}}\Big[\log p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\theta}) + \log p(\mathbf{X}|\mathbf{Z}, \boldsymbol{S}, \nu, \mathbf{u}, r)\Big] \tag{5.11}$$

The first term of gradient (5.11) can be calculated by

$$\frac{\partial \log p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\theta})}{\partial \mathbf{X}} = \frac{\partial \log p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\theta})}{\partial \mathbf{K}} \frac{\partial \mathbf{K}}{\partial \mathbf{X}} = \left[-\frac{1}{2} D\mathbf{K}^{-1} + \frac{1}{2}\mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^T\mathbf{K}^{-1}\right]\left[\frac{\partial \mathbf{K}}{\partial \mathbf{X}}\right], \tag{5.12}$$

where for an SE + WN kernel with the same lengthscale ℓ on all dimensions,

$$\frac{\partial k(\mathbf{x}_n, \mathbf{x}_m)}{\partial \mathbf{x}_n} = -\frac{\sigma_f^2}{\ell^2} \exp\left(-\frac{1}{2\ell^2}(\mathbf{x}_n - \mathbf{x}_m)^\mathsf{T}(\mathbf{x}_n - \mathbf{x}_m)\right)(\mathbf{x}_n - \mathbf{x}_m). \tag{5.13}$$

The second term of (5.11) is

$$\frac{\partial \log p(\mathbf{X}|\mathbf{Z}, \boldsymbol{S}, \nu, \mathbf{u}, r)}{\partial \mathbf{x}_n} = -\nu_{z_n} \boldsymbol{S}_{z_n}^{-1}(\mathbf{x}_n - \mathbf{u}_{z_n}). \tag{5.14}$$

We also infer kernel parameters $\boldsymbol{\theta}$ via HMC, using the gradient of the log unnormalized posterior with respect to the kernel parameters and an improper uniform prior.

Posterior predictive density

In the GP-LVM, the predictive density of at test point $\mathbf{y}_\star$ is usually computed by finding the point $\mathbf{x}_\star$ which has the highest probability of being mapped to $\mathbf{y}_\star$, then using the density of $p(\mathbf{x}_\star)$ and the Jacobian of the warping at that point to approximate the density at $\mathbf{y}_\star$. When inference is done this way, approximating the predictive density only requires solving a single optimization for each $\mathbf{y}_\star$.

For our model, we use approximate integration to estimate $p(\mathbf{y}_\star)$. This is done for two reasons: First, multiple latent points (possibly from different clusters) can map to the same observed point, meaning the standard method can underestimate $p(\mathbf{y}_\star)$. Second, because we do not optimize the latent coordinates of training points, but instead sample them, we would need to optimize each $p(\mathbf{x}_\star)$ seperately for each sample in the Markov chain. One advantage of our method is that it gives estimates for all $p(\mathbf{y}_\star)$ at once, but it may not be accurate in very high dimensions.

The posterior density in the observed space given the training data is

$$\begin{aligned} p(\mathbf{y}_\star|\mathbf{Y}) &= \iint p(\mathbf{y}_\star, \mathbf{x}_\star, \mathbf{X}|\mathbf{Y}) d\mathbf{x}_\star d\mathbf{X} \\ &= \iint p(\mathbf{y}_\star|\mathbf{x}_\star, \mathbf{X}, \mathbf{Y}) p(\mathbf{x}_\star|\mathbf{X}, \mathbf{Y}) p(\mathbf{X}|\mathbf{Y}) d\mathbf{x}_\star d\mathbf{X}. \end{aligned} \tag{5.15}$$

We first approximate $p(\mathbf{X}|\mathbf{Y})$ using samples from the Gibbs and Hamiltonian Monte Carlo chains. We then approximate $p(\mathbf{x}_\star|\mathbf{X}, \mathbf{Y})$ by sampling points from the sampled latent mixtures and warping them, using the following procedure:

1. Draw latent cluster assignment $z_\star \sim \text{Mult}(\frac{N_1}{N+\eta}, \cdots, \frac{N_C}{N+\eta}, \frac{\eta}{N+\eta})$
2. Draw latent cluster precision matrix $\mathbf{R}_\star \sim \mathcal{W}(\mathbf{S}_{z_\star}^{-1}, \nu_{z_\star})$

3. Draw latent cluster mean $\boldsymbol{\mu}_\star \sim \mathcal{N}(\mathbf{u}_{z_\star}, (r_{z_\star}\mathbf{R}_\star)^{-1})$
4. Draw latent coordinates $\mathbf{x}_\star \sim \mathcal{N}(\boldsymbol{\mu}_\star, \mathbf{R}_\star^{-1})$
5. Draw observed coordinates $\mathbf{y}_\star \sim \mathcal{N}(\mathbf{k}_\star^\mathsf{T}\mathbf{K}^{-1}\mathbf{Y}, k(\mathbf{x}_\star, \mathbf{x}_\star) - \mathbf{k}_\star^\mathsf{T}\mathbf{K}^{-1}\mathbf{k}_\star)$

If $z_\star$ is assigned to a new component in step 1, the prior Gaussian-Wishart distribution (5.2) is used for sampling in steps 2 and 3. The density drawn from in step 5 is simply the predictive distribution of a GP, where $\mathbf{k}_\star = (k(\mathbf{x}_\star, \mathbf{x}_1), \cdots, k(\mathbf{x}_\star, \mathbf{x}_N))^\mathsf{T}$.

Each step of this sampling procedure draws from the exact conditional distribution, so the Monte Carlo estimate of the predictive density $p(\mathbf{y}_\star|\mathbf{X}, \mathbf{Y})$ will converge to the true marginal distribution. Since the observations $\mathbf{y}_\star$ are conditionally normally distributed, each one adds a smooth contribution to the empirical Monte Carlo estimate of the posterior density, as opposed to a collection of point masses.

Source code

A reference implementation of the above algorithms is available at `github.com/duvenaud/warped-mixtures`.

References

J. Lean, J. Beer, and R. Bradley. Reconstruction of solar irradiance since 1610: Implications for climate change. *Geophysical Research Letters*, 22(23):3195–3198, 1995. (page 6)

James Robert Lloyd. personal communication, 2013. (page 3)

S.N. MacEachern and P. Müller. Estimating mixture of Dirichlet process models. *Journal of Computational and Graphical Statistics*, pages 223–238, 1998. (page 15)

David J.C. MacKay. Introduction to Gaussian processes. *NATO ASI Series F Computer and Systems Sciences*, 168:133–166, 1998. (page 3)

Kevin P. Murphy. Conjugate Bayesian analysis of the Gaussian distribution. Technical report, Computer Science Department, University of British Columbia, 2007. (page 15)

Jayaram Sethuraman. A constructive definition of Dirichlet priors. *Statistica Sinica*, 4: 639–650, 1994.