31

Modelling Gene Expression Dynamics with Gaussian Process Inference

Magnus Rattray,[1] Jing Yang,[1] Sumon Ahmed,[1] and Alexis Boukouvalas[2]

[1] *Division of Informatics, Imaging & Data Sciences, Faculty of Biology, Medicine & Health, University of Manchester, UK*
[2] *PROWLER.io, Cambridge, UK*

Abstract

Gaussian process (GP) inference provides a flexible nonparametric probabilistic modelling framework. We present examples of GP inference applied to time series gene expression data and for single-cell high-dimensional 'snapshot' expression data. We provide a brief overview of GP inference and show how GPs can be used to identify dynamic genes, infer degradation rates, model replicated and clustered time series, model stochastic single-cell dynamics, and model perturbations or branching in time series data. In the case of single-cell expression data we present a scalable implementation of the Gaussian process latent variable model, which can be used for dimensionality reduction and pseudo-time inference from single-cell RNA-sequencing data. We also present a recent approach to inference of branching dynamics in single-cell data. To scale up inference in these applications we use sparse variational Bayesian inference algorithms to deal with large matrix inversions and intractable likelihood functions.

31.1 Introduction

Gaussian process (GP) regression was initially introduced as a nonparametric model of spatial data, but GPs are now widely applied for multivariate regression, classification, reinforcement learning and dimensionality reduction (Rasmussen and Williams, 2006). Their popularity stems from a useful combination of modelling flexibility with inferential tractability. Modelling flexibility is enabled by the covariance function, which can capture rich statistical relationships between data points. Inferential tractability comes from the ability to integrate over Gaussian distributions in high dimensions. GPs provide useful latent variables in non-Gaussian models, with tractable inference for large-scale problems achieved through recent developments in approximate inference. In this chapter we focus on the application of GP methods to model gene expression dynamics both from time series experiments and single-cell snapshot assays. We consider a range of modelling scenarios, from the application of standard GP regression approaches for identifying dynamic or periodic genes, to more problem-specific GP models that incorporate mRNA degradation, clustering or branching dynamics.

Handbook of Statistical Genomics, Fourth Edition, Volume 2. Edited by David J. Balding, Ida Moltke and John Marioni.

31.1.1 Covariance Function

A GP describes a distribution over functions. Functions evaluated at any finite set of points will follow a multivariate Gaussian distribution. Initially we will consider a one-dimensional function $f(x)$ where x represents time,

$$f \sim \mathcal{GP}(\mu, k),$$

in which $\mu = \mu(x)$ is the mean function and $k = k(x, x')$ is the covariance function, often referred to as the kernel function. The mean function is simply the mean of function values at any particular time x,

$$\mu(x) = \mathrm{E}[f(x)],$$

while the covariance function is the covariance of function values at any two times x and x',

$$k(x, x') = \mathrm{E}[f(x)f(x')] - \mathrm{E}[f(x)]\mathrm{E}[f(x')].$$

The covariance function plays a more fundamental role in GP modelling than the mean function, and in many cases the mean function can be set to zero.

The covariance function comes from some parametric family which determines typical properties of the samples $f(x)$. For example, a popular choice for regression is the squared exponential covariance function

$$k(x, x') = \alpha \exp\left(\frac{-(x - x')^2}{2l}\right). \tag{31.1}$$

Figure 31.1(a) shows a function sampled from a GP with this covariance function, which is infinitely differentiable and smooth. This choice is popular in regression over data that is thought to come from a smooth underlying model; for example, bulk gene expression time course data is averaged over millions of cells and may therefore be expected to change smoothly in time. The covariance function has two parameters. The amplitude α determines the scale of the functions (i.e. the marginal variance of the function at a specific value of x). The lengthscale (or in our case time-scale) l determines how frequently the function crosses the zero line on average. As $l \to \infty$ samples approach straight lines while as $l \to 0$ samples approach white noise, which is a completely uncorrelated Gaussian process.

Alternatively, the Ornstein–Uhlenbeck (OU) process covariance function is given by

$$k(x, x') = \alpha \exp\left(\frac{-|x - x'|}{l}\right), \tag{31.2}$$

with an L1 norm replacing the L2 norm in the exponent. Figure 31.1(b) shows a function sampled from a GP with this covariance function. Samples are continuous but they are now rough and non-differentiable. Dynamically this can be thought of as a process with finite velocities but infinite acceleration. We will see later that the OU process can model single-cell gene expression data, where intrinsic fluctuations are not averaged away as they are in bulk gene expression data. The two covariance functions above are limiting cases of a more general Matérn covariance function which can be used to vary the roughness of the samples (Rasmussen and Williams, 2006).

There is much interest in periodic oscillations in biological systems, with circadian rhythms, the cell cycle and various ultradian rhythmic processes the subject of intensive research. GPs provide very natural models for periodic functions. Figure 31.1(c) shows samples from a covariance function that generates smooth periodic functions (MacKay, 1998), while Figure 31.1(d) shows samples from a quasi-periodic OU process (Westermark *et al.*, 2009; Phillips *et al.*, 2017).

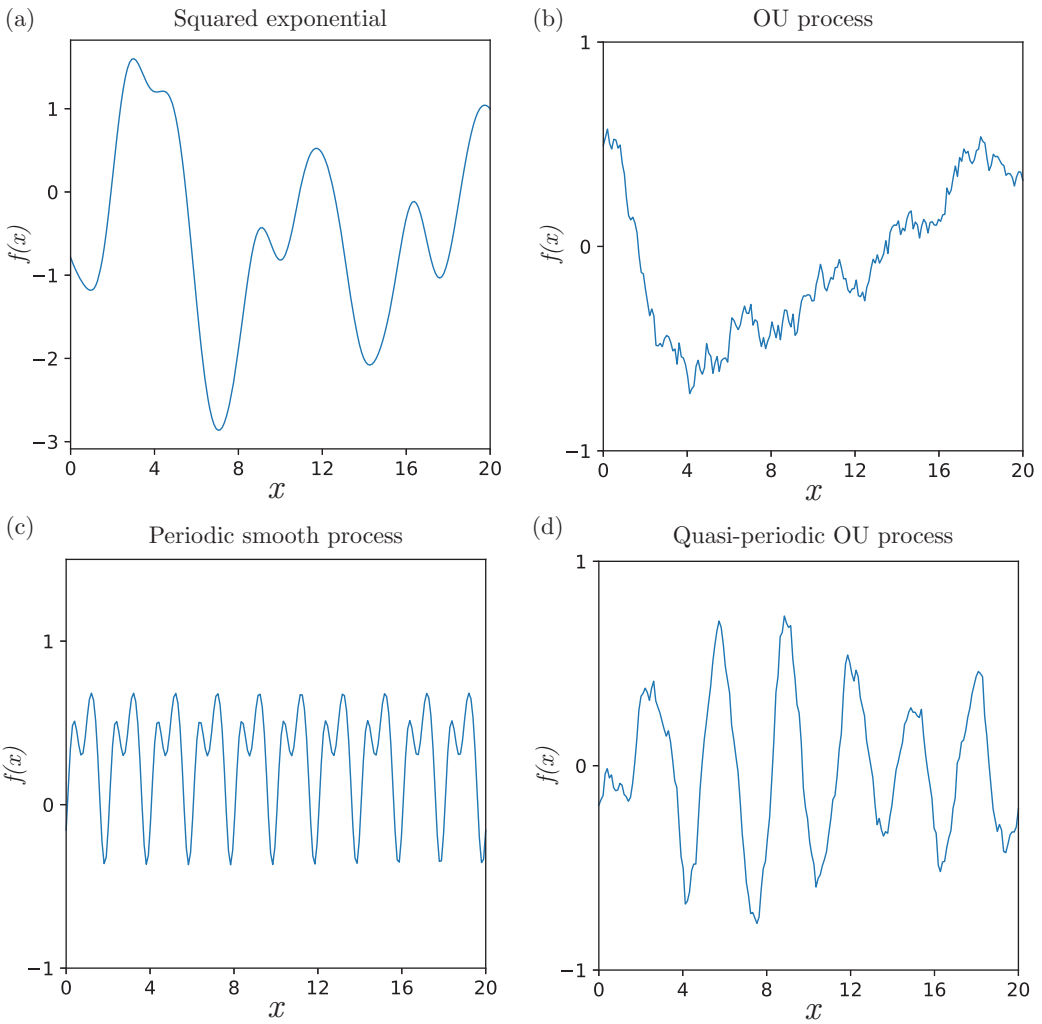

Figure 31.1 Samples from four classes of covariance function: (a) squared exponential; (b) Ornstein–Uhlenbeck process; (c) periodic smooth process; (d) quasi-periodic OU process.

The functions in Figure 31.1 are all stationary, with covariance functions that depend only on the distance between time points $|x - x'|$. This stationarity assumption can break down in certain applications. For example, gene expression time course data may be collected after a perturbation leading to a rapid initial transient phase before settling down to a constant value asymptotically. Non-stationary alternatives have therefore been developed which can better model changes in amplitude or length-scale of gene expression data over time (see, for example, Heinonen *et al.*, 2014).

An appropriate candidate set of covariance functions can be chosen using application domain knowledge, while statistical model selection can be used to select the best one from this candidate set. For example, the roughness or periodicity properties of a system may be suggested from first-principles modelling (as shown in Section 31.3.1) or the experimental design may suggest a hierarchical data structure (as shown in Section 31.2.3). Statistical model selection can then be addressed using standard likelihood-based or Bayesian model selection strategies,

with recent methods to estimate out-of-sample prediction accuracy showing great promise (Vehtari *et al.*, 2017). A nice feature of GP models is that the sample paths can be analytically integrated out to obtain a marginal likelihood that depends on relatively few parameters (see Section 31.1.2). This is an attractive feature of GPs for both maximum likelihood and Bayesian integration approaches, since there are relatively few parameters to optimise or integrate over using numerical methods.

31.1.2 Inference

Given a finite set of noise-corrupted measurements at different times, we are interested in which underlying functions are most likely to have generated the observed data. If we assume that the covariance function is known then this is very easy to do with a GP, because we can condition and marginalise exactly with Gaussian distributions.

In the regression setting, we have a data set D with regressors $\mathbf{X} = \{x_n\}_{n=1}^N$ and corresponding real-valued targets $\mathbf{Y} = \{y_n\}_{n=1}^N$. In the case of time course data the regressors are an ordered vector such that $x_n \geq x_{n-1}$, but there is no restriction on the spacing since GPs operate over a continuous domain. We allow the case $x_n = x_{n-1}$ since that provides a simple way to incorporate replicates. We assume that measurement noise in $\mathbf{Y}$, denoted by ϵ, is independently Gaussian distributed $\epsilon \sim \mathcal{N}(0, \sigma^2\mathbf{I})$ and the underlying model for $\mathbf{Y}$ as a function of $\mathbf{X}$ is $f(\cdot)$, so that

$$\mathbf{Y} = f(\mathbf{X}) + \epsilon, \tag{31.3}$$

where $f(\mathbf{X})$ represents a sample from a GP evaluated at all the times in the vector $\mathbf{X}$. Our prior modelling assumption is that the function f is drawn from a GP prior with zero mean and covariance function $k(x, x')$. The probability of the data $\mathbf{Y}$ under the model is obtained by integrating out the function $f(\mathbf{X})$,

$$\begin{aligned} p(\mathbf{Y}|\mathbf{X}) &= \int \mathcal{N}(\mathbf{Y}|\mathbf{f}, \sigma^2\mathbf{I})\, \mathcal{N}(\mathbf{f}|\mathbf{0}, K(\mathbf{X}, \mathbf{X}))\mathrm{d}\mathbf{f} \\ &= \mathcal{N}(\mathbf{Y}|\mathbf{0}, K(\mathbf{X}, \mathbf{X}) + \sigma^2\mathbf{I}), \end{aligned} \tag{31.4}$$

where we have written $\mathbf{f} = f(\mathbf{X})$ and $K(\mathbf{X}, \mathbf{X})$ is the $N \times N$ covariance matrix with elements $k(x_n, x_m)$ determined by the covariance function.

A typical regression analysis will be focused on a new input x_* and its prediction f_*. Based upon Gaussian properties (Rasmussen and Williams, 2006) the posterior distribution of f_* given data $\mathbf{Y}$ is $f_*|\mathbf{Y} \sim \mathcal{N}(\mu_*, C_*)$, with

$$\begin{aligned} \mu_* &= K(\mathbf{X}, x_*)^T (K(\mathbf{X}, \mathbf{X}) + \sigma^2\mathbf{I})^{-1}\mathbf{Y}, \\ C_* &= K(x_*, x_*) - K(\mathbf{X}, x_*)^T (K(\mathbf{X}, \mathbf{X}) + \sigma^2\mathbf{I})^{-1} K(\mathbf{X}, x_*). \end{aligned}$$

This is the posterior prediction of the function f at a specific time point x_* but is easily generalised to the full functional posterior distribution, showing that the posterior function is another GP (Rasmussen and Williams, 2006). We see above that the mean prediction is a weighted sum over data with weights larger for nearby points in a manner determined by the covariance function. The posterior covariance captures our uncertainty in the inference of f_* which will typically be reduced as we incorporate more data. Figure 31.2 shows an example of regression with a squared exponential covariance function. In Figure 31.2(a) we show some samples from the prior, and in Figure 31.2(b) the posterior distribution is fitted to four observations. In this case, the data are observed without noise ($\sigma^2 = 0$) but we still have uncertainty because many functions are consistent with the data. The posterior shows which functions are likely given the data and our prior belief in the underlying function. The prior expects functions to be smooth and

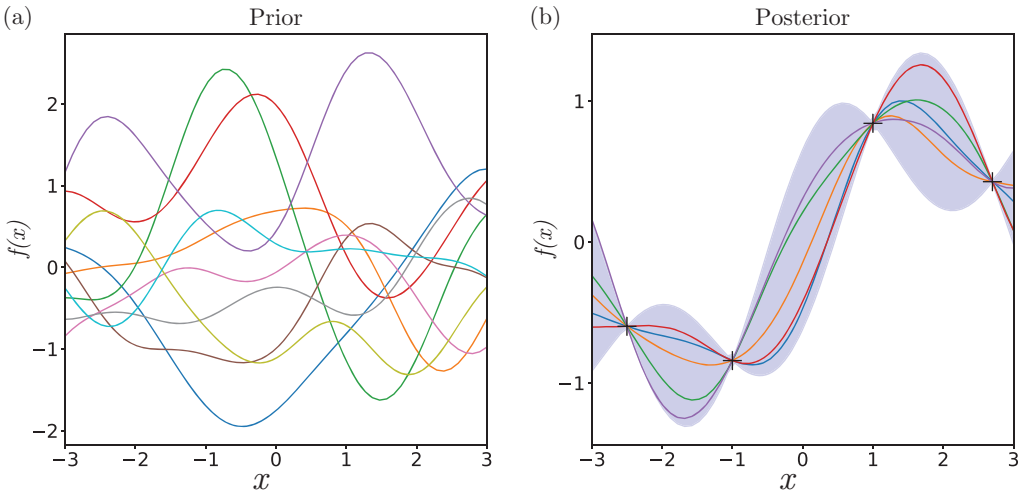

Figure 31.2 (a) Ten samples drawn from a GP with a squared exponential covariance function with hyperparameters $\alpha = 1$ and $l = 1$. (b) Ten samples from the posterior distribution after observing four data points without any observation noise ($\sigma^2 = 0$). The functions are constrained to pass through the data but the posterior distribution captures our uncertainty away from the data. The shading shows two standard deviations of posterior distribution at each time.

not to change very rapidly, and therefore our uncertainty increases gradually as we move away from the data.

We often refer to the parameters of the covariance function (including the noise variance) as hyperparameters, since the function $f(x)$ itself can be considered a functional parameter of the model. The log likelihood of the hyperparameters $L(\theta)$ is the logged probability of the data in equation (31.4),

$$\begin{aligned} L(\theta) &= \log \mathcal{N}(\mathbf{Y}|\mathbf{0}, K(\mathbf{X},\mathbf{X}) + \sigma^2\mathbf{I}) \\ &= -\tfrac{N}{2}\log(2\pi) - \tfrac{1}{2}\log\det(\sigma^2\mathbf{I} + \mathbf{K}) - \tfrac{1}{2}\mathbf{Y}^T(\sigma^2\mathbf{I} + \mathbf{K})^{-1}\mathbf{Y}, \end{aligned} \tag{31.5}$$

where we have written $\mathbf{K} = K(\mathbf{X},\mathbf{X})$. This likelihood function has a complex form and may be multimodal so that hyperparameter inference by either maximum likelihood or Bayesian inference requires numerical optimisation or integration methods. Gradient-based methods for optimisation (e.g. quasi-Newton or conjugate gradient) or Bayesian inference (e.g. Hamiltonian Monte Carlo (HMC)) are implemented in a number of popular GP inference software packages. In this chapter we have used the python packages GPy (https://sheffieldml.github.io/GPy/) and GPflow (https://github.com/GPflow), as well as the DEtime R package (https://github.com/ManchesterBioinference/DEtime).

31.2 Applications to Bulk Time Series Expression Data

Most gene expression experiments involve measuring expression in bulk samples that typically contain many millions of cells, for example microarray or RNA-sequencing data derived from tissue, cell culture or whole organisms. In this case, averaging over many cells will remove the intrinsic stochasticity of gene expression within individual cells and we would expect time course data to follow a smooth trajectory, although measurements will include experimental

sources of variation that have to be taken into account. We typically model these sources as independent and identically distributed (i.i.d.) noise added to the GP function, although in some cases it is advantageous to use more structured models of variation across time series replicates (discussed in Section 31.2.3).

31.2.1 Identifying Differential Expression in Time

Kalaitzis and Lawrence (2011) introduce a simple approach to identify whether genes measured in a time course experiment show significant evidence of changes in time. A GP model is used to determine whether there is evidence for smooth temporal changes. By fitting the covariance function's length-scale parameter, one can determine the likelihood under a GP model and compare with the likelihood under a constant model, in which case all variation is modelled as white noise. Figure 31.3 shows this approach applied to one gene from a time series gene expression data set from Lewis *et al.* (2015). In Figure 31.3(a) we fit a GP model and in Figure 31.3(b) we show the fit of a constant model, with associated credible regions for the fitted model in each case. In this example the likelihood of the GP model is much higher, providing significant evidence for dynamics in this gene. Note that the likelihood here is for the model with the function $f(x)$ marginalised out (sometimes referred to as the marginal likelihood, equation (31.5)) and therefore it can reasonably be used for model selection as complexity of the function $f(x)$ is controlled for by Bayesian model averaging.

The likelihood ratio between the GP and white noise model provides a simple means to rank genes in terms of differential expression (DE) across time. To select a significance threshold one could therefore use a penalised likelihood approach (e.g. the BIC score) to penalise the more complex model or use a likelihood ratio test with an associated false discovery rate (FDR) threshold (see Phillips *et al.*, 2017, for an example of this FDR approach to discovering periodic genes). A fully Bayesian treatment would require a numerical approach to integrate over the hyperparameters.

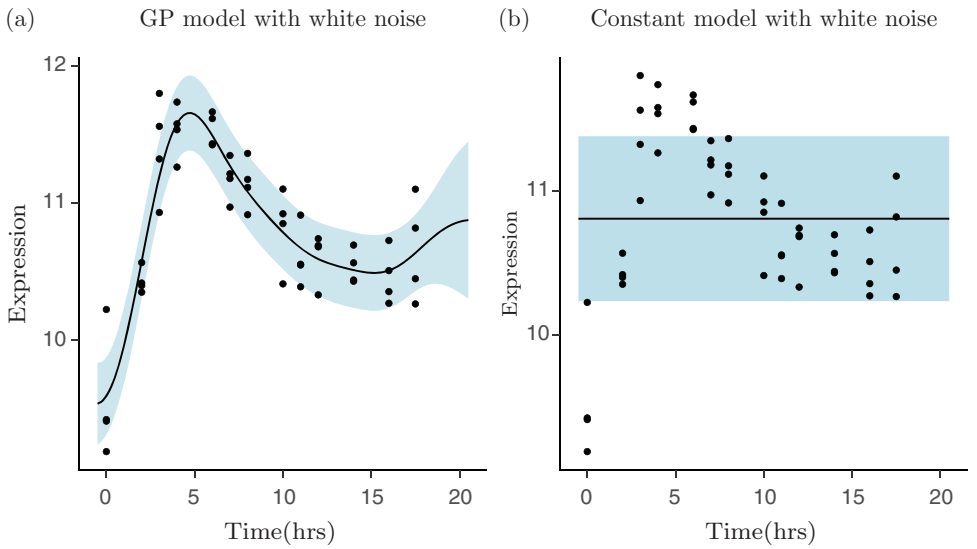

Figure 31.3 In (a) we apply GP regression to expression data from arabidopsis after infection by a plant pathogen (Lewis *et al.*, 2015) for gene *CATMA1a00010* and we compare it to the fit to a constant model shown in (b). For this gene there is strong evidence of differential expression across time.

As well as looking at differential expression, GPs have also been used to investigate changes in splicing over time (Topa and Honkela, 2016) and Huang and Sanguinetti (2016) combine a GP model of temporal change in transcript abundance with a transcript inference algorithm modelling RNA-sequencing read data.

31.2.2 Identifying Changes between Two Time Course Experiments

In some cases a time course experiment is done over two different conditions, for example to compare between a wildtype and mutant strain of an organism. In such a 'two-sample' experiment we may be interested in determining whether two time series are different. The above approach can then easily be extended to compare models in which the two time series data sets come from the same or different underlying GP functions. However, it is often more informative to also identify the time periods where the two samples differ. Stegle *et al.* (2010) introduced a GP-based method for identifying regions of DE between two samples based on a mixture of two GP regression models in which data at each time point can be assigned to the same function or two different functions. A simpler strategy, based on fitting two different GP functions to each time series and developing test statistics to identify regions of DE, was introduced by Heinonen *et al.* (2014) who also improved performance through use of a non-stationary covariance function.

As an alternative approach, Yang *et al.* (2016) develop a method to detect the first point where two time series begin to differ. This is done through a covariance function for a branching process. First we write down the joint covariance function for two GP functions $f(x) \sim \mathcal{GP}(0, k)$ and $g(x) \sim \mathcal{GP}(0, k)$ which are constrained to cross at a specific time $x = x_p$ so that $f(x_p) = g(x_p)$,

$$\begin{Bmatrix} K_{ff} & K_{fg} \\ K_{gf} & K_{gg} \end{Bmatrix} = \begin{Bmatrix} K(\mathbf{X}, \mathbf{X}) & \dfrac{K(\mathbf{X}, x_p)K(\mathbf{X}, x_p)^T}{k(x_p, x_p)} \\ \dfrac{K(\mathbf{X}, x_p)K(\mathbf{X}, x_p)^T}{k(x_p, x_p)} & K(\mathbf{X}, \mathbf{X}) \end{Bmatrix}. \tag{31.6}$$

Consider a control time course data set $\mathbf{Y}^c$ and a perturbed time course data set $\mathbf{Y}^p$. Before x_p we model these two data sets as noise-corrupted versions of the same underlying mean function $f(x) \sim \mathcal{GP}(0, k)$,

$$\begin{aligned} y^c(x_n) &\sim \mathcal{N}(f(x_n), \sigma^2), \\ y^p(x_n) &\sim \mathcal{N}(f(x_n), \sigma^2), \quad \text{for } x_n \leq x_p. \end{aligned}$$

After x_p the mean function for y^c stays intact while the mean function for y^p changes to follow $g(x)$,

$$\begin{aligned} y^c(x_n) &\sim \mathcal{N}(f(x_n), \sigma^2), \\ y^p(x_n) &\sim \mathcal{N}(g(x_n), \sigma^2), \quad \text{for } x_n > x_p, \end{aligned}$$

where f and g are constrained to cross at x_p and therefore follow the GP given by the covariance in equation (31.6).

The model is fitted using the standard regression approach described in Section 31.1.2. The perturbation time x_p is a hyperparameter of the covariance function for this model along with the length-scale and amplitude of the functions f and g. The length-scale and amplitude can be

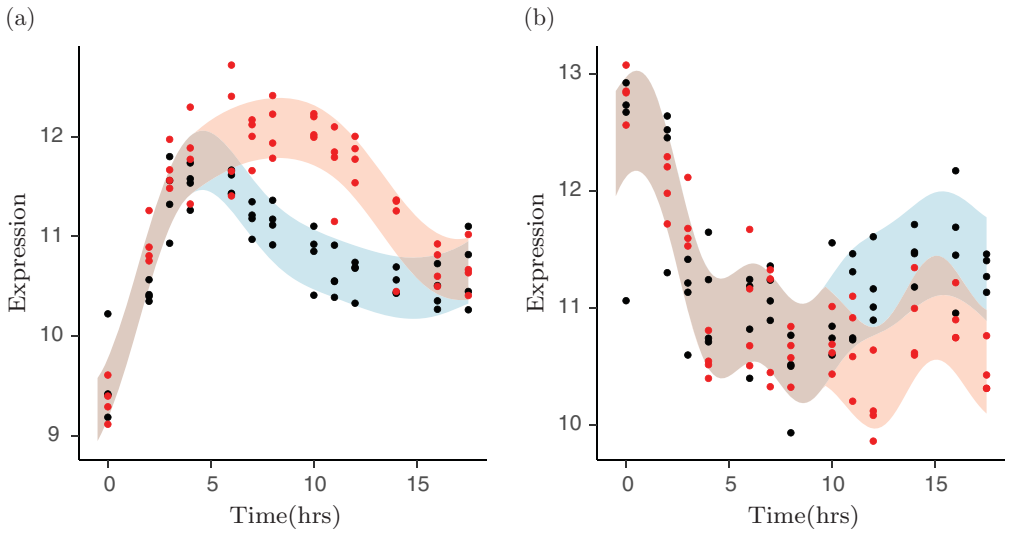

Figure 31.4 Perturbation time inference using the gene expression time series data from Lewis *et al.* (2015) which was analysed using a branching GP model by Yang *et al.* (2016). Arabidopsis gene expression dynamics was studied after infection by a wildtype pathogen (black points) and compared with infection by a mutated pathogen (red points). In (a) we show an example of an early DE gene *CATMA1a00010* from Figure 31.3, while (b) shows a late DE gene *CATMA1a00060*. The GP model fit is shown with x_p set at the mean posterior estimate in each case.

reasonably estimated by fitting independent regression models to the data from the two conditions with shared length-scale and amplitude parameters estimated by maximum likelihood. This then leaves us with the inference problem for x_p. As this is a one-dimensional problem we can estimate the posterior by a simple histogram approach,

$$p(x_p|\mathbf{Y}^c, \mathbf{Y}^p) \simeq \frac{p(\mathbf{Y}^c, \mathbf{Y}^p|x_p)}{\sum_{x_p=x_{\min}}^{x_p=x_{\max}} p(\mathbf{Y}^c, \mathbf{Y}^p|x_p)}, \tag{31.7}$$

which avoids the need to use complex optimisation or integration schemes.

Figure 31.4 shows an example application of this model to a two-sample gene expression data set from an experiment with arabidopsis (Lewis *et al.*, 2015). The experiment involves infecting the plant with a pathogen to produce the control time series and infecting the plant with a mutated strain of the pathogen to produce the perturbation time series. The model identifies the gene in Figure 31.4(a) as early DE (posterior median $x_p = 3.9$ hours) while the gene in Figure 31.4(b) diverges later in the time series (posterior median $x_p = 9.6$ hours). Yang *et al.* (2016) use the model to rank genes in terms of their perturbation time, to help understand the sequence of events underlying the immune response to infection.

As well as returning the posterior over the perturbation time, the above model can also provide evidence for whether the two time series differ (after any time) or are statistically indistinguishable. Figure 31.5 shows examples of data with a perturbation in the middle of the time course (a) and with no perturbation (b). The posterior distribution of the perturbation time is shown on the top in each case. If the perturbation time is inferred closer to the start then it is more likely that the two time course profiles are truly distinct, whereas a perturbation time inferred at the end of the time course indicates the two time profiles are very similar to each other and less likely to differ. We can make a decision over whether or not there is a bifurcation

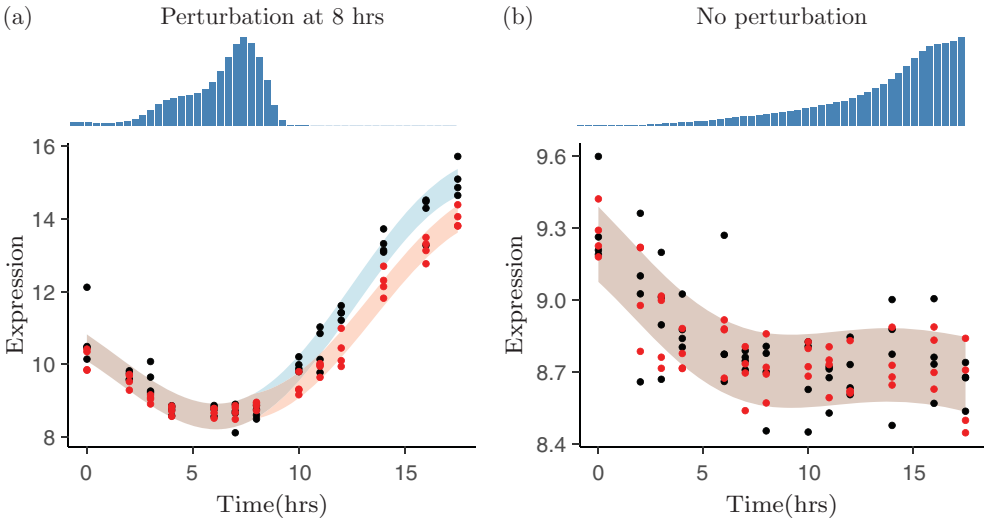

Figure 31.5 Two examples of the posterior distribution of the perturbation time (upper) and GP regression model fit based upon the maximum *a posteriori* estimate of the perturbation time (lower). In (a) we show data (gene CATMA1a00045 from Lewis *et al.* (2015)) with a perturbation introduced halfway along the time range, while in (b) we show data without any perturbation (gene CATMA1a00180). When there is no perturbation then the posterior tends to increase towards the end of the time range. A Bayes factor (equation 31.8) can be used to determine support for whether the data exhibits any bifurcations.

by considering the Bayes factor between a model with or without a perturbation (Boukouvalas *et al.*, 2018). The logged Bayes factor between a model with or without branching is given by

$$\begin{aligned} r_g &= \log \frac{p(0 < x_p < x_{\max} | \mathbf{Y}^c, \mathbf{Y}^p)}{p(x_p = x_{\max} | \mathbf{Y}^c, \mathbf{Y}^p)} \\ &= \log \left[\frac{1}{N_b} \sum_{x_p = x_{\min}}^{x_p = x_{\max}} p(\mathbf{Y}^c, \mathbf{Y}^p | x_p) \right] - \log \left[p(\mathbf{Y}^c, \mathbf{Y}^p | x_{\max}) \right], \end{aligned} \tag{31.8}$$

where N_b is the number of bins in the histogram approximation to the posterior and setting $x_p = x_{\max}$ is equivalent to having no perturbation at all. Here, we are assuming equal prior probability for having a perturbation (at any time before $x_{\max}$ with equal probability) or having no perturbation. We can see from this expression that if the height of the posterior at the final time is greater than the average of the posterior over all other times, as in Figure 31.5(b), then the probability of a bifurcation event under the model is less than 0.5. In this example there is very strong evidence for a perturbation in Figure 31.5(a) ($r_g = 31.73$) and quite strong evidence for no perturbation in Figure 31.5(b) ($r_g = -1.33$).

31.2.3 Hierarchical Models of Replicates and Clusters

In some cases the assumption of biological or technical variation as i.i.d. white noise is not justified and can lead to sub-optimal modelling. For example, biological time course replicates may be collected at different times, leading to large between-replicate variation and an associated batch effect. Similarly, different genes within a cluster should not be treated as white noise around the cluster mean as each gene will have its own different underlying profile within the cluster. Hensman *et al.* (2013) introduced a hierarchical Gaussian process to capture both of

these effects. In the case of time course replicates we model the mean underlying profile of gene n shared by all replicates using a GP. We then model the replicates r as samples from another GP distributed around the shared profile,

$$g_n \sim \mathcal{GP}(0, k_g),$$
$$f_{nr} \sim \mathcal{GP}(g_n, k_f).$$

This simple model allows for variation in the profile across replicates and is also a powerful approach to transfer information between replicates collected at different times. For example, Hensman *et al.* (2013) showed how eight replicated developmental time course data sets could be jointly modelled, with different replicates covering different stages of development (Kalinka *et al.*, 2010). By adding an additional layer in the hierarchy one can also model the shared profile of a cluster c,

$$h_c \sim \mathcal{GP}(0, k_h),$$
$$g_{nc} \sim \mathcal{GP}(h_c, k_g),$$
$$f_{ncr} \sim \mathcal{GP}(g_{nc}, k_f),$$

where there is now a GP for each cluster c, each gene n within the cluster and each replicate r. This hierarchical approach can then be combined with a model-based clustering method to infer the cluster assignments of genes. An efficient Dirichlet process mixture model implementation was developed by Hensman *et al.* (2015) using variational inference techniques and was applied to cluster circadian expression data in Gossan *et al.* (2013) using periodic covariance functions.

31.2.4 Differential Equation Models of Production and Degradation

Any linear transformation of a GP is another GP. This holds true for integral solutions of linear ordinary differential equations (ODEs) which contain GP functions and allows GPs to be used within simple ODE models. For example, consider a model of mRNA $m(x)$ being produced with transcription rate $f(x)$ at time x. We can write

$$\frac{\mathrm{d}m}{\mathrm{d}x} = f(x) - \delta m,$$

where δ is the mRNA degradation rate. The solution to this linear ODE is

$$m(x) = m(0)\mathrm{e}^{-\delta x} + \int_0^x \mathrm{e}^{\delta(u-x)} f(u)\mathrm{d}u.$$

We see that the mRNA concentration is a weighted integral of the production rate. Figure 31.6 shows two scenarios where, for the same transcription rate function, the mRNA time profiles can be very different. In Figure 31.6(a) the degradation rate is relatively high and $m(x)$ and $f(x)$ therefore have similar shapes. In Figure 31.6(b) we see that for low degradation rate, $m(x)$ will integrate $f(x)$ and has a qualitatively different profile with a much later peak in expression.

We can place a GP prior on $f \sim \mathcal{GP}(0, k_f)$, and since $m(x)$ is a linear functional of $f(x)$ they form a two-dimensional GP. If we choose the squared exponential covariance for k_f (recall equation (31.1)) then the covariance for $[f, m]$ can be worked out in closed form (Lawrence *et al.*, 2007). Figure 31.6 shows two examples of samples drawn from this GP, with a different degradation rate hyperparameter used in each case.

This model can be adapted to a number of different scenarios. Barenco *et al.* (2006) showed how the above ODE could be used to model multiple targets of a transcription factor protein, in order to infer its activity and discover other target genes. Lawrence *et al.* (2007) then applied GP inference to the same task which avoided Markov chain Monte Carlo (MCMC) sampling of the

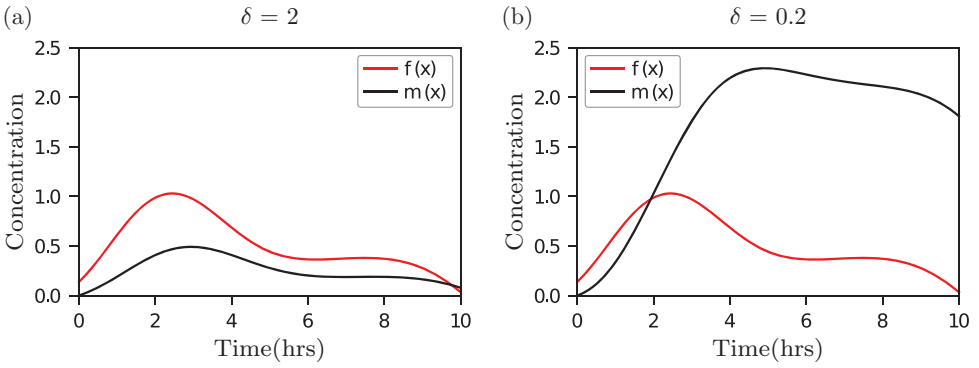

Figure 31.6 Comparison of mRNA profile $m(x)$ generated with the same transcription rate function $f(x)$ but using two different degradation rates. With high degradation ((a), $\delta = 2$) the mRNA and transcription rates have similar shape, but with low degradation ((b), $\delta = 0.2$) the mRNA level peaks much later. Here, the transcription rate function is in fact a sample drawn from a GP and $m(x)$ and $f(x)$ are therefore joint samples from a bivariate GP.

latent function and therefore more computationally efficient inference. Honkela *et al.* (2010) used a similar approach to identify transcription factor targets using data from an embryonic development time course experiment, but they extended the model to an additional layer to account for protein production and degradation. That work was further extended to ODE models with a nonlinear dependence of target gene expression to multiple regulatory transcription factors, requiring the development of MCMC techniques for GP inference and hyperparameter estimation (Titsias *et al.*, 2012). By including a delay parameter in the model, Honkela *et al.* (2015) used a similar ODE model to identify genes with a significant delay between nascent mRNA production and mature mRNA accumulation through joint analysis of RNA-sequencing and Pol-II ChIP-sequencing time course data.

31.3 Modelling Single-Cell Data

Single cells present several new challenges for inference. Gene expression is intrinsically stochastic at the single-cell level and therefore trajectory data at single-cell resolution, available from live cell imaging experiments, has to be modelled with care. One useful approach to modelling this data is to apply a linear noise approximation, in which stochastic oscillations are modelled as a GP. Live cell imaging can only be done for one or two genes simultaneously and is not scalable to genome-wide studies. Single-cell genomics enables genome-wide expression profiling, but because the experiments are destructive it is impossible to profile the same cell at different times. However, inference can be used to try and uncover gene expression dynamics by inferring where each cell lies in some *pseudotemporal* ordering. GPs provide one approach to inferring such a pseudotemporal ordering, as well as being useful in more general dimensionality reduction of single-cell data.

31.3.1 Modelling Single-Cell Trajectory Data

The chemical master equation (CME) provides a description of stochastic biochemical reactions which can be used to model gene expression dynamics in single cells. The CME can be simulated using the Gillespie algorithm (Gillespie, 1977) and provides samples of the stochastic dynamics, but it is not usually possible to compute the likelihood of data sampled

from a CME and therefore inference with such models is difficult. The linear noise approximation (LNA) has been used as an approximation to the CME which is valid for sufficiently large numbers of molecules (Komorowski *et al.*, 2009; Fearnhead *et al.*, 2014). In the LNA the dynamics is approximated by a GP with a mean function determined by an ODE and the covariance described as a Gauss–Markov stochastic process with variance also determined by an ODE. If noise-corrupted data are collected at discrete times then inference can be carried out using a Kalman filter (Kalman, 1960; Jazwinski, 1970). The GP nature of the process also allows inference for data collected through applying any linear function to the process. For example, if microscopy data is collected over long periods in studies using a luciferase reporter then the data can be modelled as the integral of the underlying process and inference with the LNA remains tractable (Folia and Rattray, 2018).

Once the deterministic part of the dynamics has reached a fixed point then the GP follows a multivariate OU process (recall Figure 31.1) which is a stationary Gauss–Markov process. In systems with negative feedback this multivariate Gauss–Markov process can exhibit oscillations, even when the deterministic part has converged to a stable fixed point, and therefore oscillations can be induced by the stochasticity present in a single-cell system (Galla, 2009). For example, consider a linear damped oscillator which can be written as a Langevin equation (Westermark *et al.*, 2009),

$$\frac{dx}{dt} = -\lambda x - \omega y + \xi_x,$$
$$\frac{dy}{dt} = \omega x - \lambda y + \xi_y,$$

where ξ_x and ξ_y are zero-mean white noise with covariance $k_\xi(t,t') = 2D\delta(t-t')$. In this case the variables x and y have the same covariance function,

$$k(t,t') = \frac{D}{\lambda}\exp\left(-\lambda|t-t'|\right)\cos(\omega|t-t'|).$$

Figure 31.1(d) shows a sample from a GP with this covariance function. The samples are rough and approximately periodic, but oscillations gradually shift in phase over time so that they are not precisely periodic. This could represent a regulatory network where gene x is repressed by gene y while gene y is activated by gene x (after normalising expression to have mean zero). Without stochasticity ($D = 0$) the system converges to a fixed point and the covariance is zero – the oscillations are only caused by finite molecule numbers in such a system and are not seen in a large system size limit.

Phillips *et al.* (2017) used GP inference to identify stochastic oscillations in single-cell microscopy data by fitting a GP model with the above covariance function and comparing it with a non-oscillatory OU process with covariance function given by equation (31.2). The Hes1 transcription factor is known to exhibit negative autoregulation with delay which can lead to oscillations (Monk, 2003) and under some conditions these oscillations only persist due to the presence of stochastic fluctuations in single cells (Galla, 2009). Using GP-based inference, Phillips *et al.* (2017) found that single cells were likely to exhibit stochastic oscillations of Hes1 expression while expression data from a constitutive promoter was never classified as an oscillator, showing that the GP-based approach has good specificity in this application.

31.3.2 Dimensionality Reduction and Pseudotime Inference

Single-cell RNA-sequencing experiments have genome-wide coverage and therefore produce high-dimensional data sets with substantial biological and technical variation. GPs can be used

for dimensionality reduction of multivariate data by treating the regressors $\mathbf{X}$ as parameters (or latent variables) to be inferred along with the functions $f(\mathbf{X})$. Recall the GP model in equation (31.3). Consider a multivariate GP regression model for many data dimensions y_i, with $i = 1 \ldots d$, each with their own GP function f_i,

$$y_{in} = f_i(x_n) + \epsilon_{in}.$$

In the case of pseudotime inference $\mathbf{X} = [x_n]$ is a vector, but more generally it will live in some low-dimensional space into which we would like to project our data. We treat $\mathbf{X}$ as a latent variable that has to be inferred along with the functions f_i and associated covariance hyperparameters. This is the Gaussian process latent variable model (GPLVM) which is a popular probabilistic approach for nonlinear dimensionality reduction (Lawrence, 2005; Titsias and Lawrence, 2010).

The log likelihood can be worked out similarly to standard GP regression in equation (31.5), except that $\mathbf{X}$ is now a parameter of the model,

$$L(\theta, \mathbf{X}) = -\frac{ND}{2}\log(2\pi) - \frac{D}{2}\log\det(\sigma^2\mathbf{I} + \mathbf{K}) - \frac{1}{2}\mathrm{tr}\left[(\sigma^2\mathbf{I} + \mathbf{K})^{-1}\mathbf{Y}\mathbf{Y}^T\right],$$

where $\mathbf{K}$ has elements $k(x_n, x_m)$ that depend on $\mathbf{X}$ through the covariance function. In the original formulation of the GPLVM the latent points $\mathbf{X}$ were optimised by maximum likelihood (Lawrence, 2005), but later the Bayesian GPLVM (BGPLVM) was introduced which placed a prior on $\mathbf{X}$ and a variational Bayesian inference algorithm was used to approximate the posterior distribution over the latent space (Titsias and Lawrence, 2010).

The GPLVM has been used to visualise single-cell gene expression in a number of studies (Buettner and Theis, 2012; Buettner *et al.*, 2015; Zwiessele and Lawrence, 2016). It has also been used to sample pseudotime trajectories in order to quantify uncertainty in pseudotime (Campbell and Yau, 2016). In the case of single-cell time series experiments, where cells are collected at multiple capture times, then the prior on the latent variable $\mathbf{X}$ can incorporate this capture time information to improve pseudotime inference (Reid and Wernisch, 2016). In Figure 31.7 we show how including capture time information in the prior of the BGPLVM can be used to align one of the latent dimensions with time in a developmental time course data set (Ahmed *et al.*, 2018). Cells were captured from single embryos at different times in mouse embryonic development (Guo *et al.*, 2010). At the 32-cell stage they begin to differentiate into different cell types. We see that aligning one axis with time makes the latent space more

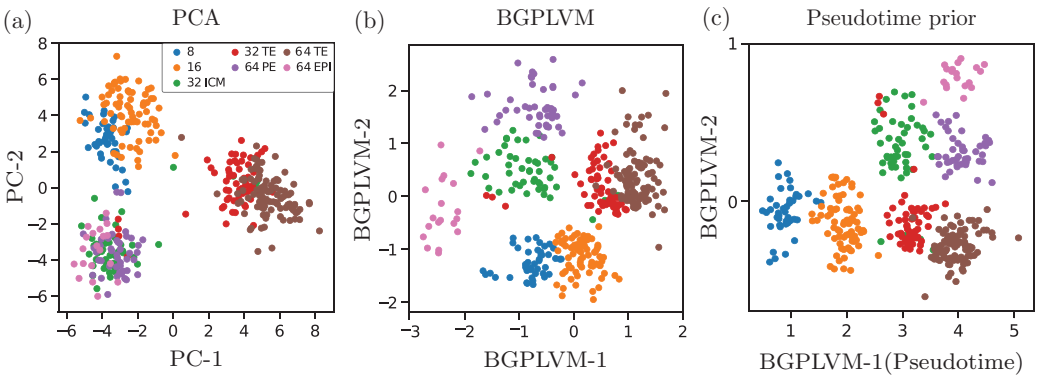

Figure 31.7 We project single-cell gene expression data from Guo *et al.* (2010) onto two latent dimensions using PCA and the BGPLVM with two different choices of prior: (a) PCA; (b) BGPLVM with zero-mean prior for all latent points; (c) BGPLVM with prior mean in one latent dimension based on capture times. For further details, see Ahmed *et al.* (2018).

interpretable, with the other dimension capturing differences in cell type. We can also see that the nonlinear dimensionality reduction of the GPLVM leads to a cleaner separation of cell stages and types than a linear approach such as principal components analysis (PCA). Ahmed *et al.* (2018) show that using a two-dimensional latent space with pseudotime on one dimension leads to a higher correlation between pseudotime and capture time than using a one-dimensional pseudotime latent space with the same capture time informed prior on that dimension.

The BGPLVM approach to pseudotime estimation implemented by Campbell and Yau (2016) and Reid and Wernisch (2016) made use of MCMC or HMC sampling over $\mathbf{X}$. These sampling-based approaches do not scale up to inference over large droplet-based single-cell RNA-sequencing experiments which can profile tens of thousands of cells. The GrandPrix package used in Figure 31.7 (Ahmed *et al.*, 2018) uses a more computationally efficient variational inference scheme (Titsias and Lawrence, 2010) and is implemented using the GPflow package (Matthews *et al.*, 2017) which adapts the TensorFlow package to GP inference, allowing scalability to multiple cores is sufficient cores and optionally also graphics processing units.

31.3.3 Modelling Branching Dynamics with Single-Cell RNA-Sequencing Data

Figure 31.7 shows an example of cells differentiating into different cell types during development. Recent developments in single-cell RNA-sequencing allow gene expression to be profiled in thousands of cells. In some cases, the cells being profiled are undergoing differentiation but there are no time labels in the data. For example, a sample may contain a continuum of stem cells, differentiated cells and intermediates. In that case pseudotime methods can be used to investigate differentiation by fitting models of branching dynamics. A number of algorithms have been proposed to discover branching dynamics in cellular trajectories from single-cell expression data (Haghverdi *et al.*, 2016; Street *et al.*, 2017; Qiu *et al.*, 2017a). Lönnberg *et al.* (2017) used the overlapping mixture of Gaussian processes (OMGP) model (Lázaro-Gredilla *et al.*, 2012) to identify cellular branching dynamics after pseudotime inference, by identifying where in pseudotime the cells are better described as coming from two profiles rather than one.

The methods in Section 31.2.2 can also be extended to model branching dynamics if cell-to-branch assignments are learned in a similar way to the OMGP model. Consider a set of GP functions $F = \{f_1, f_2, \ldots, f_M\}$ which are branches in a tree following a covariance of the type defined in Section 31.2.2. Then define $Z \in \{0,1\}^{N \times M}$ to be binary variables which determine the assignment of N cells to M branches and which have to be inferred along with F. In the simplest case of a single branching event $M = 3$ and x_b is the pseudotime of branching, which we consider to be specific to a particular gene. By applying a global branching and pseudotime algorithm (e.g. Monocle 2; Qiu *et al.*, 2017a) we can gain some prior information about each gene's branching dynamics. Consider that x_g is the global branching point in pseudotime but that genes may branch before or after that point, or possibly not exhibit any branching. If $x_b < x_g$ then the global branching provides no information about which branch the cell belongs to for $x_b < x < x_g$. If $x_b > x_g$ then we can use the inferred global branching to increase the probability of a cell being assigned to a particular branch. This approach was recently used to identify whether individual gene expression shows branching dynamics and whether the branching is early or late in pseudotime (Boukouvalas *et al.*, 2018). Inference is not exactly tractable in this class of models but Boukouvalas *et al.* (2018) developed an efficient sparse variational inference algorithm which generalises the OMGP inference algorithm (Lázaro-Gredilla *et al.*, 2012) to the case where the functions in F are not independent.

The scalability of the model to large data sets is achieved in Boukouvalas *et al.* (2018) by using an inducing point sparse approximation. The key bottleneck of applying GP models to large data sets is that the full covariance inversion, required at each iteration of hyperparameter

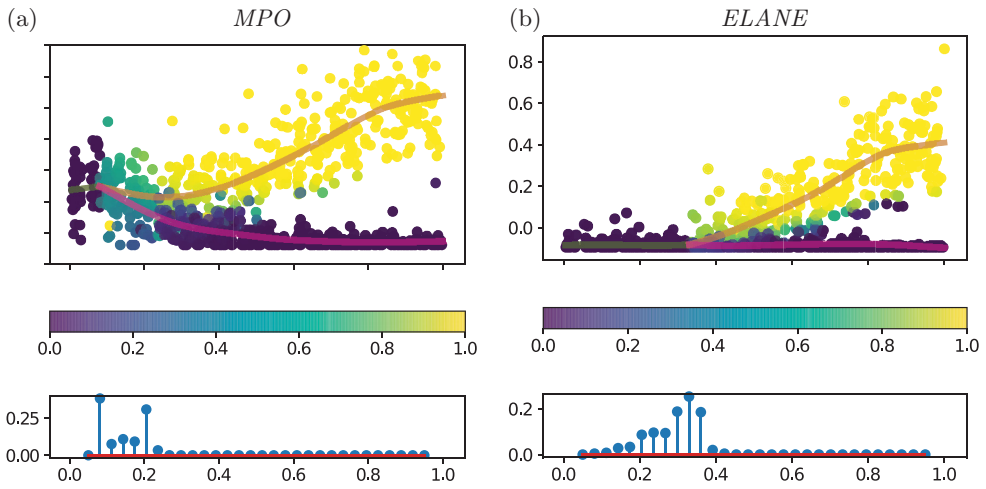

Figure 31.8 Haematopoiesis single-cell gene expression data from Paul *et al.* (2015) is shown for genes (a) *MPO* and (b) *ELANE*. Monocle 2 was used to infer pseudotime for each cell and the BGP model was used to identify the branching time for each gene. For each gene the posterior cell assignment is shown in top subpanel. In the bottom subpanel the posterior branching time is shown. For further details about the BGP model see Boukouvalas *et al.* (2018).

optimisation, scales cubically, $O(N^3)$, where N is the number of data points, or alternatively cells in the case of single-cell data. In contrast, the inducing point approximation defines a number of auxiliary variables, termed inducing points, that trade off fidelity to the full GP and computational speed. Specifically, for k inducing points the inference scales as $O(k^2N)$ rather than $O(N^3)$, where typically $k \ll N$. Where the input dimension is low (e.g. time series data) it is often sufficient to have a relatively small number of inducing points; in the present application we found $k = 30$ to be sufficient and little improvement was obtained when increasing k further in our synthetic data studies (Boukouvalas *et al.*, 2018). Bauer *et al.* (2016) provide more general insights into the performance of alternative inducing point approaches.

In Figure 31.8 we apply the branching GP (BGP; Boukouvalas *et al.*, 2018) model to single-cell RNA-sequencing haematopoietic stem cell data from Paul *et al.* (2015). The data consists of 4423 cells, and Monocle 2 was used to infer the global cellular branching and pseudotime for each cell (Qiu *et al.*, 2017a). We show the inferred gene-specific branching dynamics for two genes, one of which is inferred to branch earlier than the global branching (Figure 31.8(a)) and another which is inferred to branch later (Figure 31.8(b)). Below the genes we show the posterior over the inferred branching time which is computed using the histogram approach in equation (31.7). Boukouvalas *et al.* (2018) show that the GP approach can better deal with cases where branching differs from the global branching than the spline-based approach implemented in the BEAM package (Qiu *et al.*, 2017b) which does not model cell-to-branch assignment probabilistically.

31.4 Conclusion

We have presented a number of ways in which GP inference methods can be used to make inferences about gene expression dynamics. There are many ongoing challenges. In most cases described here we have modelled data as coming from a Gaussian distribution, but this is a poor assumption in many interesting applications. For example, single-cell RNA-sequencing

data can contain large numbers of zero measurements and count-based likelihoods are more suitable for modelling this class of data. Count-based likelihoods are available for standard GP regression and can easily be implemented for other GP models if using MCMC inference – for example, using the popular probabilistic programming language Stan (Carpenter *et al.*, 2016). However, to scale up to large numbers of single cells we have adopted more computationally efficient sparse variational inference algorithms, and count-based likelihoods are not yet available for variational inference in the BGP or GPLVM models described here.

The BGP method is used to infer gene-specific branching after applying another algorithm for inferring pseudotime and the global cellular branching. This approach does not fully take into account errors and uncertainty in this initial cell labelling stage prior to gene-specific modelling. The model does allow branch labels to change through inference since the global labels are treated as a prior. However, we do assume that pseudotimes are known without error, when in reality pseudotime inference is associated with high levels of uncertainty (Campbell and Yau, 2016). An interesting extension would therefore be to combine the BGP and GPLVM models to jointly model branching and infer latent manifolds from single-cell expression data, taking all sources of error into account through use of a unified model.

Acknowledgements

The ideas and results described here are the product of numerous interactions with our close collaborators, including Neil Lawrence, James Hensman and Antti Honkela. MR and JY are supported by MRC award MR/N00017X/1; MR and AB are supported by MRC award MR/M008908/1; MR is supported by a Wellcome Trust Investigator Award; SA was supported by a Commonwealth PhD Scholarship.

References

Ahmed, S., Rattray, M. and Boukouvalas, A. (2018). GrandPrix: Scaling up the Bayesian GPLVM for single-cell data. *Bioinformatics* **35**(1), 47–54.

Barenco, M., Tomescu, D., Brewer, D., Callard, R., Stark, J. and Hubank, M. (2006). Ranked prediction of p53 targets using hidden variable dynamic modeling. *Genome Biology* **7**(3), R25.

Bauer, M., van derWilk, M. and Rasmussen, C.E. (2016). Understanding probabilistic sparse Gaussian process approximations. In *Advances in Neural Information Processing Systems*. MIT Press, Cambridge, MA, pp. 1533–1541.

Boukouvalas, A., Hensman, J. and Rattray, M. (2018). BGP: Identifying gene-specific branching dynamics from single-cell data with a branching Gaussian process. *Genome Biology* **19**:65.

Buettner, F. and Theis, F.J. (2012). A novel approach for resolving differences in single-cell gene expression patterns from zygote to blastocyst. *Bioinformatics* **28**(18), i626–i632.

Buettner, F., Natarajan, K.N., Casale, F.P., Proserpio, V., Scialdone, A., Theis, F.J., Teichmann, S.A., Marioni, J.C. and Stegle, O. (2015). Computational analysis of cell-to-cell heterogeneity in single-cell RNA-sequencing data reveals hidden subpopulations of cells. *Nature Biotechnology* **33**(2), 155–160.

Campbell, K.R. and Yau, C. (2016). Order under uncertainty: Robust differential expression analysis using probabilistic models for pseudotime inference. *PLoS Computational Biology* **12**(11), e1005212.

Carpenter, B., Gelman, A., Hoffman, M., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M.A., Guo, J., Li, P., Riddell, A., *et al.* (2016). Stan: A probabilistic programming language. *Journal of Statistical Software* **20**(2), 1–37.

Fearnhead, P., Giagos, V. and Sherlock, C. (2014). Inference for reaction networks using the linear noise approximation. *Biometrics* **70**(2), 457–466.

Folia, M.M. and Rattray, M. (2018). Trajectory inference and parameter estimation in stochastic models with temporally aggregated data. *Statistics and Computing* **28**(5), 1053–1072.

Galla, T. Intrinsic fluctuations in stochastic delay systems: Theoretical description and application to a simple model of gene regulation. (2009). *Physical Review E* **80**(2), 021909.

Gillespie, D.T. (1977). Exact stochastic simulation of coupled chemical reactions. *Journal of Physical Chemistry* **81**(25), 2340–2361.

Gossan, N., Zeef, L., Hensman, J., Hughes, A., Bateman, J.F., Rowley, L., Little, C.B., Piggins, H.D., Rattray, M., Boot-Handford, R.P., *et al.* (2013). The circadian clock in murine chondrocytes regulates genes controlling key aspects of cartilage homeostasis. *Arthritis & Rheumatology* **65**(9), 2334–2345.

Guo, G., Huss, M., Tong, G.Q., Wang, C., Sun, L.L., Clarke, N.D. and Robson, P. (2010). Resolution of cell fate decisions revealed by single-cell gene expression analysis from zygote to blastocyst. *Developmental Cell* **18**(4), 675–685.

Haghverdi, L., Buettner, M., Alexander Wolf, F., Buettner, F. and Theis, F.J. (2016). Diffusion pseudotime robustly reconstructs lineage branching. *Nature Methods* **13**(10), 845.

Heinonen, M., Guipaud, O., Milliat, F., Buard, Valérie, Micheau, Béatrice, Tarlet, G., Benderitter, M., Zehraoui, F. and d'Alché Buc, F. (2014). Detecting time periods of differential gene expression using Gaussian processes: An application to endothelial cells exposed to radiotherapy dose fraction. *Bioinformatics* **31**(5), 728–735.

Hensman, J., Lawrence, N.D. and Rattray, M. (2013). Hierarchical Bayesian modelling of gene expression time series across irregularly sampled replicates and clusters. *BMC Bioinformatics* **14**(1), 252.

Hensman, J., Rattray, M. and Lawrence, N.D. (2015). Fast nonparametric clustering of structured time-series. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **37**(2), 383–393.

Honkela, A., Girardot, C., Hilary Gustafson, E., Liu, Ya-H., Furlong, E.E., Lawrence, N.D. and Rattray, M. (2010). Model-based method for transcription factor target identification with limited data. *Proceedings of the National Academy of Sciences of the United States of America* **107**(17), 7793–7798.

Honkela, A., Peltonen, J., Topa, H., Charapitsa, I., Matarese, F., Grote, K., Stunnenberg, H.G., Reid, G., Lawrence, N.D. and Rattray, M. (2015). Genome-wide modeling of transcription kinetics reveals patterns of RNA production delays. *Proceedings of the National Academy of Sciences of the United States of America* **112**(42), 13115–13120.

Huang, Y. and Sanguinetti, G. (2016). Statistical modeling of isoform splicing dynamics from RNA-seq time series data. *Bioinformatics* **32**(19), 2965–2972.

Jazwinski, A.H. (1970). *Stochastic Processes and Filtering Theory*. Academic Press, New York.

Kalaitzis, A.A. and Lawrence, N.D. (2011). A simple approach to ranking differentially expressed gene expression time courses through Gaussian process regression. *BMC Bioinformatics* **12**(1), 180.

Kalinka, A.T., Varga, K.M., Gerrard, D.T., Preibisch, S., Corcoran, D.L., Jarrells, J., Ohler, U., Bergman, C.M. and Tomancak, P. (2010). Gene expression divergence recapitulates the developmental hourglass model. *Nature* **468**, 811–814.

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering* **82**, 35–45.

Komorowski, M., Finkenstdt, B., Harper, C.V. and Rand, D.A. (2009). Bayesian inference of biochemical kinetic parameters using the linear noise approximation. *BMC Bioinformatics* **10**, 1–10.

Lawrence, N. (2005). Probabilistic non-linear principal component analysis with Gaussian process latent variable models. *Journal of Machine Learning Research* **6**, 1783–1816.

Lawrence, N.D., Sanguinetti, G. and Rattray, M. (2007). Modelling transcriptional regulation using Gaussian processes. In B. Schölkopf, J.C. Platt, and T. Hoffman (eds), *Advances in Neural Information Processing Systems 19*. MIT Press, Cambridge, MA, pp. 785–792

Lázaro-Gredilla, M., Vaerenbergh, S.V. and Lawrence, N.D. (2012). Overlapping mixtures of Gaussian processes for the data association problem. *Pattern Recognition* **45**(4), 1386–1395.

Lewis, L.A., Polanski, K., de Torres-Zabala, M., Jayaraman, S., Bowden, L., Moore, J., Penfold, C.A., Jenkins, D.J., Hill, C., Baxter, L., Kulasekaran, S., Truman, W., Littlejohn, G., Prusinska, J., Mead, A., Steinbrenner, J., Hickman, R., Rand, D., Wild, D.L., Ott, S., Buchanan-Wollaston, V., Smirnoff, N., Beynon, J., Denby, K. and Grant, M. (2015). Transcriptional dynamics driving mamp-triggered immunity and pathogen effector-mediated immunosuppression in arabidopsis leaves following infection with *Pseudomonas syringae* pv tomato dc3000. *Plant Cell* **27**(11), 3038–3064.

Lönnberg, T., Svensson, V., James, K.R., Fernandez-Ruiz, D., Sebina, I., Montandon, R., Soon, M.S., Fogg, L.G., Nair, A.S., Liligeto, U., *et al.* (2017). Single-cell RNA-seq and computational analysis using temporal mixture modelling resolves Th1/Tfh fate bifurcation in malaria. *Science Immunology* **2**(9).

MacKay, D.J. (1998). Introduction to gaussian processes. In C.M. Bishop (ed.), *Neural Networks and Machine Learning*. Springer. pp. 133–166.

Matthews, A., van der Wilk, M., Nickson, T., Fujii, K., Boukouvalas, A., León-Villagrá, P., Ghahramani, Z. and Hensman, J. (2017). GPflow: A Gaussian process library using Tensorflow. *Journal of Machine Learning Research* **18**(40), 1–6.

Monk, N.A.M. (2003). Oscillatory expression of Hes1, p53, and NF-κB driven by transcriptional time delays. *Current Biology* **13**(16), 1409–1413.

Paul, F., Arkin, Y., Giladi, A., Jaitin, D.A., Kenigsberg, E., Keren-Shaul, H., Winter, D., Lara-Astiaso, D., Gury, M., Weiner, A., *et al.* (2015). Transcriptional heterogeneity and lineage commitment in myeloid progenitors. *Cell* **163**(7), 1663–1677.

Phillips, N.E., Manning, C., Papalopulu, N. and Rattray, M. (2017). Identifying stochastic oscillations in single-cell live imaging time series using Gaussian processes. *PLoS Computational Biology* **13**(5), e1005479.

Qiu, X., Hill, A., Ma, Yi-A. and Trapnell, C. (2017a). Single-cell mRNA quantification and differential analysis with Census. *Nature Methods* **14**, 309–315.

Qiu, X., Mao, Q., Tang, Y., Wang, L., Chawla, R., Pliner, H.A. and Trapnell, C. (2017b). Reversed graph embedding resolves complex single-cell trajectories. *Nature Methods* **14**, 979–982.

Rasmussen, C.E. and Williams, C.K. (2006). *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA.

Reid, J.E. and Wernisch, L. (2016). Pseudotime estimation: Deconfounding single cell time series. *Bioinformatics* **32**(19), 2973–2980.

Stegle, O., Denby, K.J., Cooke, E.J., Wild, D.L., Ghahramani, Z. and Borgwardt, K.M. (2010). A robust bayesian two-sample test for detecting intervals of differential gene expression in microarray time series. *Journal of Computational Biology* **17**(3), 355–367.

Street, K., Risso, D., Fletcher, R.B., Das, D., Ngai, J., Yosef, N., Purdom, E. and Dudoit, S. (2017). Slingshot: Cell lineage and pseudotime inference for single-cell transcriptomics. Preprint, bioRxiv 128843.

Titsias, M. and Lawrence, N.D. (2010). Bayesian Gaussian process latent variable model. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 844–851.

Titsias, M.K., Honkela, A., Lawrence, N.D. and Rattray, M. (2012). Identifying targets of multiple co-regulating transcription factors from expression time-series by Bayesian model comparison. *BMC Systems Biology* **6**(1), **53**.

Topa, H. and Honkela, A. (2016). Analysis of differential splicing suggests different modes of short-term splicing regulation. *Bioinformatics* **32**(12), i147–i155.

Vehtari, A., Gelman, A. and Gabry, J. (2017). Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. *Statistics and Computing* **27**(5), 1413–1432.

Westermark, Pål O., Welsh, D.K., Okamura, H. and Herzel, H. (2009). Quantification of circadian rhythms in single cells. *PLoS Computational Biology* **5**(11), e1000580.

Yang, J., Penfold, C.A., Grant, M.R. and Rattray, M. (2016). Inferring the perturbation time from biological time course data. *Bioinformatics* **32**(19), 2956–2964.

Zwiessele, M. and Lawrence, N.D. (2016). Topslam: Waddington landscape recovery for single cell experiments. Preprint, bioRxiv 057778.