

# Branch-recombinant Gaussian processes for analysis of perturbations in biological time series

Christopher A. Penfold<sup>1,2,\*†</sup>, Anastasiya Sybirna<sup>1,3,4,†</sup>, John E. Reid<sup>5,6</sup>, Yun Huang<sup>1,4</sup>, Lorenz Wernisch<sup>5</sup>, Zoubin Ghahramani<sup>7</sup>, Murray Grant<sup>8</sup> and M. Azim Surani<sup>1,2,3</sup>

<sup>1</sup>Wellcome Trust/Cancer Research UK Gurdon Institute, Henry Wellcome Building of Cancer and Developmental Biology, Cambridge CB2 1QN, UK, <sup>2</sup>Department of Statistics, University of Warwick, Coventry, CV4 7AL, UK, <sup>3</sup>Wellcome/MRC Stem Cell Institute, University of Cambridge, CB2 1QR, UK, <sup>4</sup>Department of Physiology, Development and Neuroscience, University of Cambridge, Cambridge, <sup>5</sup>MRC Biostatistics Unit, University of Cambridge, Cambridge Institute of Public Health, Cambridge Biomedical Campus, Cambridge, CB2 0SR, UK, <sup>6</sup>The Alan Turing Institute, London, NW1 2DB, UK, <sup>7</sup>Department of Engineering, University of Cambridge, Cambridge, CB2 1PZ, UK and <sup>8</sup>School of Life Sciences, Gibbet Hill Campus, The University of Warwick, Coventry, CV4 7AL, UK

\*To whom correspondence should be addressed.

†The authors wish it to be known that, in their opinion, the first two authors should be regarded as Joint First Authors.

## Abstract

**Motivation:** A common class of behaviour encountered in the biological sciences involves branching and recombination. During branching, a statistical process bifurcates resulting in two or more potentially correlated processes that may undergo further branching; the contrary is true during recombination, where two or more statistical processes converge. A key objective is to identify the time of this bifurcation (branch or recombination time) from time series measurements, e.g. by comparing a control time series with perturbed time series. Gaussian processes (GPs) represent an ideal framework for such analysis, allowing for nonlinear regression that includes a rigorous treatment of uncertainty. Currently, however, GP models only exist for two-branch systems. Here, we highlight how arbitrarily complex branching processes can be built using the correct composition of covariance functions within a GP framework, thus outlining a general framework for the treatment of branching and recombination in the form of branch-recombinant Gaussian processes (B-RGPs).

**Results:** We first benchmark the performance of B-RGPs compared to a variety of existing regression approaches, and demonstrate robustness to model misspecification. B-RGPs are then used to investigate the branching patterns of *Arabidopsis thaliana* gene expression following inoculation with the hemibiotrophic bacteria, *Pseudomonas syringae* DC3000, and a disarmed mutant strain, *hrpA*. By grouping genes according to the number of branches, we could naturally separate out genes involved in basal immune response from those subverted by the virulent strain, and show enrichment for targets of pathogen protein effectors. Finally, we identify two early branching genes WRKY11 and WRKY17, and show that genes that branched at similar times to WRKY11/17 were enriched for W-box binding motifs, and overrepresented for genes differentially expressed in WRKY11/17 knockouts, suggesting that branch time could be used for identifying direct and indirect binding targets of key transcription factors.

**Availability and implementation:** <https://github.com/cap76/BranchingGPs>

**Contact:** [c.penfold@gurdon.cam.ac.uk](mailto:c.penfold@gurdon.cam.ac.uk)

**Supplementary information:** [Supplementary data](#) are available at *Bioinformatics* online.

## 1 Introduction

A common class of behaviour encountered in the biological sciences involves branching. In a branching process, often driven by a biological perturbation, a statistical process bifurcates at a specific time, leading to two potentially correlated processes that may, themselves, undergo further branching (Poincaré, 1885). Reciprocal behaviour is encountered in recombination processes, where two or more statistical processes converge.

Such branching and recombination are frequently encountered in transcriptional time series data involving host-pathogen interactions. The initial response to infection is the activation of innate immunity, a highly conserved response based upon perception of non-self. Subsequently, pathogens can deliver protein effectors which collectively suppress immunity, and later collaborate to reconfigure plant metabolism for pathogen nutrition. Thus, initially, the expression dynamics of key infection marker genes will be identically distributed in both infected and uninfected host cells. Expression patterns will begin to diverge as the host mounts immunity; in many cases, this innate immune response is suppressed by the pathogen, potentially driving expression levels of certain genes back to uninfected levels. Indeed nearly 50% of the transcriptome is observed to be differentially expressed during some plant infections (Lewis et al., 2015; Windram et al., 2012). More complex patterns of branching and recombination may exist in such datasets due to the ongoing evolutionary arms race between pathogens and their hosts (Boller and He, 2009; Jones and Dangl, 2006).

The ability to infer the timing of bifurcations in individual genes should reveal important information about the onset and development of infection. The inference of branching and recombination processes from systems level measurements, such as collections of microarray or RNA-sequencing data, remains a difficult challenge, partially due to datasets being noisy in nature, with (potentially) missing observations or uneven temporal sampling. The dynamic nature of different biological systems may also vary significantly, frustrating efforts to find a robust, broadly applicable approach to the inference of branching and recombination. Nonparametric Bayesian approaches to inference would therefore be advantageous, addressing these key issues. Gaussian processes represent a flexible Bayesian nonparametric approach to nonlinear regression able to gracefully cope with uncertainty, uneven sampling and a diverse range of dynamic behaviour (Rasmussen and Williams, 2006). However, currently, Gaussian processes treatments for branching processes have only been developed for the two-branch case (Stegle et al., 2010; Yang et al., 2016). Here, we develop an approach to inference of arbitrarily complex branching and recombination processes, in the form of branch-recombinant Gaussian processes (B-RGPs). In Section 2 we first introduce B-RGPs, highlighting their key limiting behaviour. In Section 3, we demonstrate the advantages of B-RGPs over GPs on a variety of simulated datasets, and in Section 4, we demonstrate the utility of our approach on genome-scale time-course microarray data, by identifying transcriptional branching and recombination in *Arabidopsis thaliana* infected with the bacterial pathogen *Pseudomonas syringae*. Finally, in Section 5, we discuss a variety of possible applications for B-RGPs and future avenues for research.

## 2 Materials and methods

Within a Bayesian setting, Gaussian processes (GPs) can be used to represent prior distributions over smooth functions, providing a flexible framework for regression and classification with robust treatment of uncertainty (Rasmussen and Williams, 2006). This

makes GP-based approaches ideal frameworks for quantifying the dynamics of gene expression from biological observations (Breeze et al., 2011; Hensman et al., 2013; Kalaitzis and Lawrence, 2011; Stegle et al., 2010). For regression, we typically have a set of observations,  $\mathbf{y}$ , assumed to be noisy instances of a continuous underlying function at input locations  $\mathbf{t}$ :  $\mathbf{y} = f(\mathbf{t}) + \epsilon$ , where  $\epsilon$  represents Gaussian additive noise. In our applications,  $\mathbf{y}$  will typically be used to denote a vector of the observed expression levels for a given gene at times,  $\mathbf{t}$ . We can assign the unknown function a GP prior, denoted  $f(\mathbf{t}) \sim \mathcal{GP}(\mu(\mathbf{t}), k(\mathbf{t}, \mathbf{t}'))$ , and analytically evaluate the posterior distribution at a set of new input locations,  $\mathbf{t}^*$ . The marginal likelihood, too, may be analytically evaluated, making GPs a flexible and efficient framework for both prediction and model comparison. Previous GP-based approaches to branching have been outlined for the two-dataset case, i.e. where there exists two biological processes following branching. These include the studies by Stegle et al. (2010), who developed a GP two-sample approach, based on mixtures of GPs, and the more recent work of Yang et al. (2016), who demonstrate explicitly how a two-branch process can be encoded within a joint GP model. To our knowledge, the generalisation of GPs to  $>2$  branches has not been addressed, whilst no explicit closed-form solution to recombination has been outlined.

A useful extension to the GP framework is the multiple output hierarchical Gaussian process (HGP; Hensman et al., 2013), in which a basal process is defined by a zero-mean GP with covariance function  $k_1(\mathbf{t}, \mathbf{t}')$ , with a subsequent process having mean  $f_1(\mathbf{t})$  and covariance function,  $k_2(\mathbf{t}, \mathbf{t}')$ :

$$\begin{aligned} f_1(\mathbf{t}) &\sim \mathcal{GP}\left(0, k_1(\mathbf{t}, \mathbf{t}')\right), \\ f_2(\mathbf{t}) &\sim \mathcal{GP}\left(f_1(\mathbf{t}), k_2(\mathbf{t}, \mathbf{t}')\right). \end{aligned}$$

Within this framework, we assume noisy observations of the functions,  $y_1 = f_1(\mathbf{t}) + \epsilon$ , and  $y_2 = f_2(\mathbf{t}) + \epsilon$ , and may analytically evaluate the posterior distribution at a new set of input locations for prediction, or the marginal likelihood for model comparison. A class of branching behaviour can naturally be encoded within this HGP framework, assuming the basal (main branch) process is defined by a zero-mean GP with covariance function  $k_{b_1}(\mathbf{t}, \mathbf{t}')$ , with a subsequent process having mean  $f_{b_1}(\mathbf{t})$  and an appropriate covariance function that ensures the two processes are identically distributed prior to an arbitrarily chosen time point,  $t_b$ :

$$\begin{aligned} f_1(\mathbf{t}) &\sim \mathcal{GP}\left(0, k_1(\mathbf{t}, \mathbf{t}')\right), \\ f_2(\mathbf{t}) &\sim \mathcal{GP}\left(f_1(\mathbf{t}), CP_{t_b}(K_0, k_2(\mathbf{t}, \mathbf{t}'))\right), \end{aligned}$$

where  $K_0 = K_0(\mathbf{t}, \mathbf{t}')$  denotes a zero-kernel, and  $CP_{t_b}(k_1, k_2)$  denotes a change-point kernel (Lloyd et al., 2014), defined as:

$$\begin{aligned} CP_{t_b}\left(k_1(\mathbf{t}, \mathbf{t}'), k_2(\mathbf{t}, \mathbf{t}')\right) &= \sigma(\mathbf{t})k_1(\mathbf{t}, \mathbf{t}')\sigma(\mathbf{t}') \\ &\quad + (1 - \sigma(\mathbf{t}))k_2(\mathbf{t}, \mathbf{t}')\left(1 - \sigma(\mathbf{t}')\right), \end{aligned}$$

where  $(1 - \sigma(\mathbf{t})) = 1 + \tanh\left(\frac{t_b - t}{s}\right)/2$ . Here, we introduce two hyperparameters:  $t_b$ , which represents the branch time, and  $s$ , which controls how fast the second branch diverges from the basal process to the potentially correlated branch process. Note that each data point must be assigned a branch label,  $z \in [1, 2]$ , according to which branch it belongs to, e.g.  $z = 1$  will be used to denote data belonging to the control or wildtype branch, with  $z = 2$  referring to the perturbed dataset. For a two branch case observations are *a priori* Gaussian distributed,  $y_1, y_2 | \mathbf{t}, z \sim N(0, K(\mathbf{t}, \mathbf{t}', z, z'))$ , where:

$$K(t, t', z, z') = k_1(t, t') + CP_{t_b} \left( K_0, k_2(t, t') \right) \delta_{z,2} \delta_{z',2} + \beta \delta_{t,t'} \delta_{z,z'},$$

and the delta function  $\delta_{z,2} \delta_{z',2}$  ensures the change-point kernel only operates over the second branch, i.e. where the branch label  $z = 2$  and  $z' = 2$ . Within this framework, we may again make a prediction  $y^*$  at a new set of input locations,  $(t^*, z^*)$ , and analytically evaluate the marginal likelihood, allowing us to compare the goodness of fit between different branching processes.

We can allow further branches that independently diverge from the main branch, with each data point assigned a branch label. For a  $n$ -component system  $z \in [1, \dots, n]$  and we have the following covariance function:

$$K(t, t', z, z') = K_{b_1}(t, t') + \sum_{i=2}^n CP_{t_i} \left( K_0, k_{b_i}(t, t') \right) \delta_{z,i} \delta_{z',i} + \beta \delta_{t,t'} \delta_{z,z'}.$$

Alternatively, rather than each branch diverging from the main process, each branch could itself give rise to further branches in a recurrent manner, e.g. a basal (main branch) from which a secondary branch diverges, with a third branching from the second and so forth. For an  $n$ -component recurrent branching system we have:

$$K(t, t', z, z') = \begin{cases} k_1(t, t') + \beta \delta_{t,t'} \delta_{z,z'}, & \min(z, z') = 1, \\ k_1(t, t') + CP_{t_2} \left( K_0, k_2(t, t') \right) + \beta \delta_{t,t'} \delta_{z,z'}, & \min(z, z') = 2, \\ \vdots \\ k_1(t, t') + \sum_{j=2}^{n-1} CP_{t_j} \left( K_0, k_j(t, t') \right) + \beta \delta_{t,t'} \delta_{z,z'}, & \min(z, z') = n-1, \\ k_1(t, t') + \sum_{j=2}^n CP_{t_j} \left( K_0, k_j(t, t') \right) + \beta \delta_{t,t'} \delta_{z,z'}, & z = n, z' = n. \end{cases}$$

When observation data for all branches are specified over identical time points, the covariance matrix can be expressed in a more compact notation:

$$K(t, t', z, z') = \sum_{j=1}^n A^{(j)} \otimes \bar{k}_j(t, t') + A^{(1)} \otimes \beta \mathbb{I},$$

where  $\otimes$  denotes the Kronecker product and:

$$\bar{k}_j(t, t') = \begin{cases} k_1(t, t'), & j = 1 \\ CP_{t_j} \left( K_0, k_j(t, t') \right), & \text{otherwise,} \end{cases}$$

where  $m$  represents the number of unique time points,  $\mathbb{I}$  represents an  $(m \times m)$  identity matrix and  $A^{(j)} = \mathbf{u} \mathbf{u}^\top$ , with  $\mathbf{u}$  representing a column vector of length  $n$ , with ones in elements  $j$  through  $n$  and zeros everywhere else. Far more complex branching patterns can easily be built via the correct composition of independent and recurrent branching covariance functions.

As well as building branching structures of arbitrary complexity, we further note that the dynamic behaviour of the individual branches themselves may themselves be arbitrarily complex, comprised of any linear combination of positive semi-definite kernels. In [Supplementary Figure S1a](#), we indicate example behaviour of simple branching GPs.

## 2.1 Recombinant Gaussian processes

Recombinant processes can be defined in a reciprocal fashion to branching processes. Notable examples might include the reprogramming of

different terminally differentiated cell lineages to iPSCs ([Gurdon, 1962](#); [Takahashi and Yamanaka, 2006](#)). We can describe a two-component system via the following composition of covariance functions:

$$K(t, t', z, z') = k_1(t, t') + CP_{r_b} \left( k_2(t, t'), K_0 \right) \delta_{z,2} \delta_{z',2} + \beta \delta_{t,t'} \delta_{z,z'},$$

which encodes the main branch process, with a second (potentially correlated) process that recombines after time  $r_b$ . Multiple processes can again independently recombine with the main branch, or recurrently recombine via a series of parental branches, analogously to branching GPs. Example recombinant GPs are shown in [Supplementary Figure S1b](#).

## 2.2 Branch-recombinant Gaussian processes

Another important process exists where a statistical process transiently branches into two or more processes, before recombining back into a single process. Such combinations of branching and recombination may be encountered during development when there exists  $>1$  route to a terminal cell fate, as may be the case in certain neuronal lineages ([Zawadzka et al., 2010](#)), as well as in certain diseases, such as during dedifferentiation of cancer cells ([Friedmann-Morvinski and Verma, 2014](#)). An example two-component system can be encoded by the following covariance function:

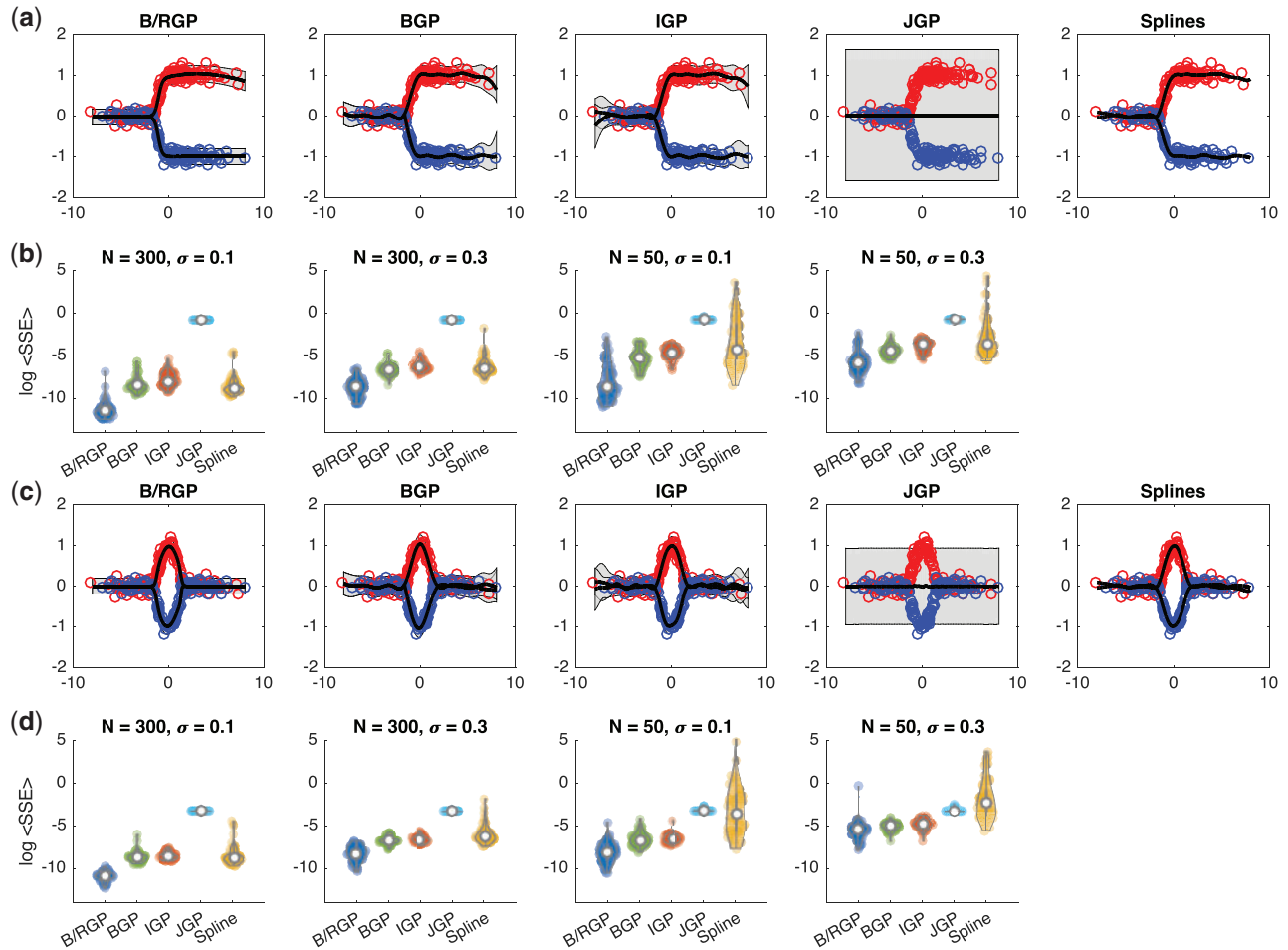
$$K(t, t', z, z') = k_1(t, t') + CP_{r_2} \left( CP_{b_2} \left( K_0, k_2(t, t') \right), K_0 \right) \delta_{z,2} \delta_{z',2} + \beta \delta_{t,t'} \delta_{z,z'}.$$

Again, more complex patterns, with arbitrary numbers of branches and recombination, can readily be built with GPs via the correct composition of covariance functions, with more complex examples shown in [Supplementary Figure S1c](#).

## 2.3 Optimisation, run time and limiting behaviour

A key advantage of the B-RGP framework outlined here over existing approaches ([Yang et al., 2016](#)) is the ability to fit arbitrarily complex branch-recombinant structures, i.e.  $>2$  branches. Furthermore, unlike the earlier work of [Yang et al. \(2016\)](#), all hyperparameters including those relating to branch and recombination times can be directly optimized via gradient based approaches, e.g. type II ML estimators. Example B-RGPs have been implemented in MATLAB using the gpml package ([Rasmussen and Williams, 2006](#)). In general, we note that inference with B-RGPs scales as any other GP, with complexity  $O(n^3)$ , where  $n$  is the number of observations; for larger datasets full GP inference becomes unfeasible, but sparse approximations are possible ([Quinonero-Candela and Rasmussen, 2005](#)), with existing support in gpml. The time required for optimisation of hyperparameters via type II ML estimates varied: for a dataset with 300 observations, 1000 steps of the gpml minimize function took approximately 30 s on a Desktop computer (2.5 GHz Intel Core i7), although it should be noted that, in many cases, full convergence could require  $>1000$  steps. This makes B-RGPs slightly slower than the time taken for DEtime ([Yang et al., 2016](#)), which, for the same dataset and default parameters, ran in around 10 s.

Depending upon the branch time hyperparameters and other hyperparameters in the change-point kernel, the behaviour of B-RGPs can naturally tend towards either an independent GP or a HGP. Specifically, for a branching GP, when  $\sigma(t) \rightarrow 1$ , as may be the case when  $(t_b - t)/s$  is very large, such as when a branch occurs much later than the last data point, then a BGP will behave as single joint GP with behaviour defined by the main branch kernel only. When  $\sigma(t) \rightarrow 0$ , as may be the case when  $(t_b - t)/s$  has increasingly low values, such as when branching occurs much earlier than the



**Fig. 1.** (a) Fits to a two-component branching process using a branch GP outlined here, the branching GP outlined in Yang *et al.* (2016), independent GPs, a joint GP and independent splines. (b) We indicate the log mean sum squared error for each of the methods for different number of training points and for different noise levels. (c) Fits to a two component branch-recombinant process using branch-recombinant GPs, branch GPs of Yang *et al.* (2016), independent GPs, joint GPs and splines. (d) Log mean sum squared error for the different approaches for different number of data points and noise levels

first observations, then the BGP will behave as a HGP. Similar limiting behaviour applies for recombination processes.

### 3 Results

As a preliminary test of the B-RGP framework we fitted five simulated labelled time series datasets, and evaluated the predictive accuracy over a range of test locations, comparing the accuracy to that achieved using DTime (Yang *et al.*, 2016), independent Gaussian process regression (IGP) over the individual branches, joint Gaussian process regression (JGP) over the union of data, and splines. We first evaluated the ability to fit the following branching process:

$$f(t, z) = \begin{cases} 0, & \text{if } t \leq -\pi/2, \\ \cos(t), & \text{if } -\pi/2 < t \leq 0, z = 1, \\ 1, & \text{if } t > 0, z = 1, \\ -\cos(t), & \text{if } -\pi/2 < t \leq 0, z = 2, \\ -1, & \text{if } t > 0, z = 2. \end{cases}$$

where  $z$  indicates the branch label. Random input locations were sampled,  $t \sim \mathcal{N}(0, 3\mathbb{I})$ , with branch labels assigned with equal probability,  $z_i \in [1, 2]$ . Observations were generated as noisy instances,  $y|t, z \sim \mathcal{N}(f(t, z), \sigma_n^2\mathbb{I})$ , where  $\sigma_n \in [0.1, 0.3]$ . A three-

component branching process, comprised of a (latent) main process from which two observed branches diverge, was fitted to the simulated data, with hyperparameters optimized using type II maximum likelihood (ML). The base kernel and all kernels were set a squared-exponentials. Branch time hyperparameters were tied, i.e.  $t_{b_1} = t_{b_2}$ , with initial values set as  $t_{b_1} = 4$ ,  $\log s_{b_1, b_2} = 0.5$ ,  $\sigma_n = 0.2$ , and all other hyperparameters  $\theta = [l_{b_0}, \sigma_{b_0}, l_{b_1}, \sigma_{b_1}, l_{b_2}, \sigma_{b_2}]$  initiated as i.i.d. random variables  $\theta_i \sim U(0.1, 1)$ . In Figure 1a, we indicate an example posterior fits to the data using a BGP, IGPs, JGPs and splines, respectively. In Figure 1b, we indicate the log mean sum squared error (SSE) over 50 randomly initiated runs using  $N=50$  and  $N=300$  training points and for different noise levels. The B-RGPs shows superior fits (reduced SSE) and decreased negative log marginal likelihood compared to other approaches. The fits obtained using DTime also appeared to perform well in all cases, outperforming independent GPs, and demonstrating the usefulness of using more accurate generative models for inference of branching data. Next we evaluated the ability of branching GPs to estimate the branch time. In Supplementary Figure S2b, we plot the branch time versus inferred branch time for 50 instances and compare to that achieved using DTime (Yang *et al.*, 2016). We note that the correlation for our approach ( $R=0.9999$ ) indicates good ability to infer branch times, and was greater than that the correlation when using DTime with default settings ( $R=9007$ ). Here, the increased

accuracy partly comes from the ability to directly optimise the branch time hyperparameters via type II ML estimates, rather than relying on a grid search of inferred branch times. To further explore the ability to infer branch times for datasets with missing observations, we repeated this experiment, but excluded observations close to the true branch point, specifically removing any data points where  $|t_{\hat{b}} - t_o| < 2$ , where,  $t_{\hat{b}}$  represents the true branch time, and  $t_o$  is the time of the data point. Even with missing observations centred at the true branch time, the inferred branch times were found to be highly correlated with the true branch time ( $R=0.9649$ ; [Supplementary Fig. S2c](#)).

In dataset 2, we assumed the following branch-recombinant process:

$$f(t, z) = \begin{cases} 0 & \text{if } |t| > \pi/2 \\ \cos(t) & \text{if } |t| \leq \pi/2, z = 1 \\ -\cos(t) & \text{if } |t| \leq \pi/2, z = 2 \end{cases}$$

where  $z$  indicates the branch label. Again, randomly determined input locations were sampled as before. A three-component branch-recombinant GP comprised of a (latent) main process from which two branches diverge and recombine, was fitted to the simulated data, with hyperparameters optimized using type II ML estimates. Example fits are shown in [Figure 1c](#), with the log mean sum square error shown in [Figure 1d](#). Again, branch-recombinant GPs outperformed all other methods, with branching GPs DEtime performing next best.

To test for robustness to model mismatch, we used B-RGPs on two other datasets. In dataset 3, a non-branching, three-component process was used to generate data, corresponding to a HGP,  $f_0(t) \sim GP(0, K_0(t, t'))$ ,  $f_1(t) \sim GP(f_0(t), K_1(t, t'))$ ,  $f_2(t) \sim GP(f_0(t), K_2(t, t'))$ , with squared-exponential covariance functions used throughout. A B-RGP was fitted to the data with hyperparameters initialized as  $t_{b_1} = -4, t_{r_1} = 4, \log s_{b_1, b_2, r_1, r_2} = 0.5$ , and  $\log \sigma_n = 0.2$ , and all other hyperparameters initiated as  $\theta_i \sim U(0.1, 1)$ . In this case, there exists a model mismatch between the data, which has no explicit branching or recombination, and the branch-recombinant covariance function used for inference. Nevertheless, we note that informally, if the branch point occurs much earlier than the first data point and the recombinant point occurs much later than the last data point, the behaviour over the range of observations is identical to that of a HGP,  $f_1(t) \sim GP(0, K_{b_1}(t, t') + K_{b_2}(t, t'))$ ,  $f_2(t) \sim GP(0, K_{b_1}(t, t') + K_{b_2}(t, t'))$ . Tuning of the branch/recombination time hyperparameters should therefore allow a good fit over the regions of observation despite the model mismatch. In [Supplementary Figure S3a](#), we plot example fits to the function using a B-RGP, BGPs, IGPs, a JGP and splines. In [Supplementary Figure S3b and c](#), we indicate the sum of squared errors and negative log marginal likelihoods. As expected, B-RGPs and IGPs were more accurate than other approaches, due to the increased flexibility to fit the two processes, rather than fitting the general underlying trend. In most cases B-RGPs performed comparably to IGPs, although in a few instances the B-RGP appeared to suffer from numerical instability and failed to converge, with the resulting mean SSE and negative log marginal likelihood distributions heavy tailed and not as favourable as for IGPs. These results suggest that B-RGPs offer comparable performance to IGPs, although performance depends on sensible initialization of hyperparameters.

To further evaluate the effect of model mismatch, we fit to data from a single, noisy process. Specifically, we used the same three-component HGP as in dataset 3, with noisy observation

data generated from the first process only, i.e. representing two replicates  $y_1 \sim \mathcal{N}(f_1(t), \sigma_n^2 \mathbb{I})$ ,  $y_2 \sim \mathcal{N}(f_2(t), \sigma_n^2 \mathbb{I})$ . As before, we fitted the data using a three-component branch-recombinant process, with squared-exponential covariance function assumed for all branches, and hyperparameters initiated  $t_{b_1} = -4, t_{r_1} = 4$ . Informally, we note that, despite the model mismatch, when branching and recombination both occur much earlier than the first observation, or much later than the last observation, the fit over the range of observations should correspond to that of a JGP with covariance function corresponding to that of the main branch process,  $f_{1,2}(t) \sim GP(0, K_0(t, t'))$ . In [Supplementary Figure S4a](#), we indicate example fits to the function using a B-RGP, IGPs and a JGP, whilst in [Supplementary Figure S4b and c](#), we indicate the SSE and negative log marginal likelihood. In general, both the B-RGP and JGP outperform the other approaches.

Finally, we performed inference on a four-branch system, in which we have one latent basal branch, from which two intermediate latent branches emerge. For comparison, we evaluate the sum squared error for the B-RGP, IGPs a JGP and splines, with the results indicating B-RGPs provide better overall performance ([Supplementary Fig. S4](#)).

Together, analysis of datasets 1 – 4 indicate B-RGPs offer superior performance compared to other approaches when the underlying data is branch-recombinant, with good ability to estimate the timing of bifurcations. Crucially, all hyperparameters can be optimized directly using type II ML. Therefore, branch and recombination time hyperparameters can be tuned, which, due to their limiting behaviour, means that they can gracefully cope with datasets where no branching structure exists, provided hyperparameters are sensibly initialized.

### 3.1 Inference for partially labelled datasets

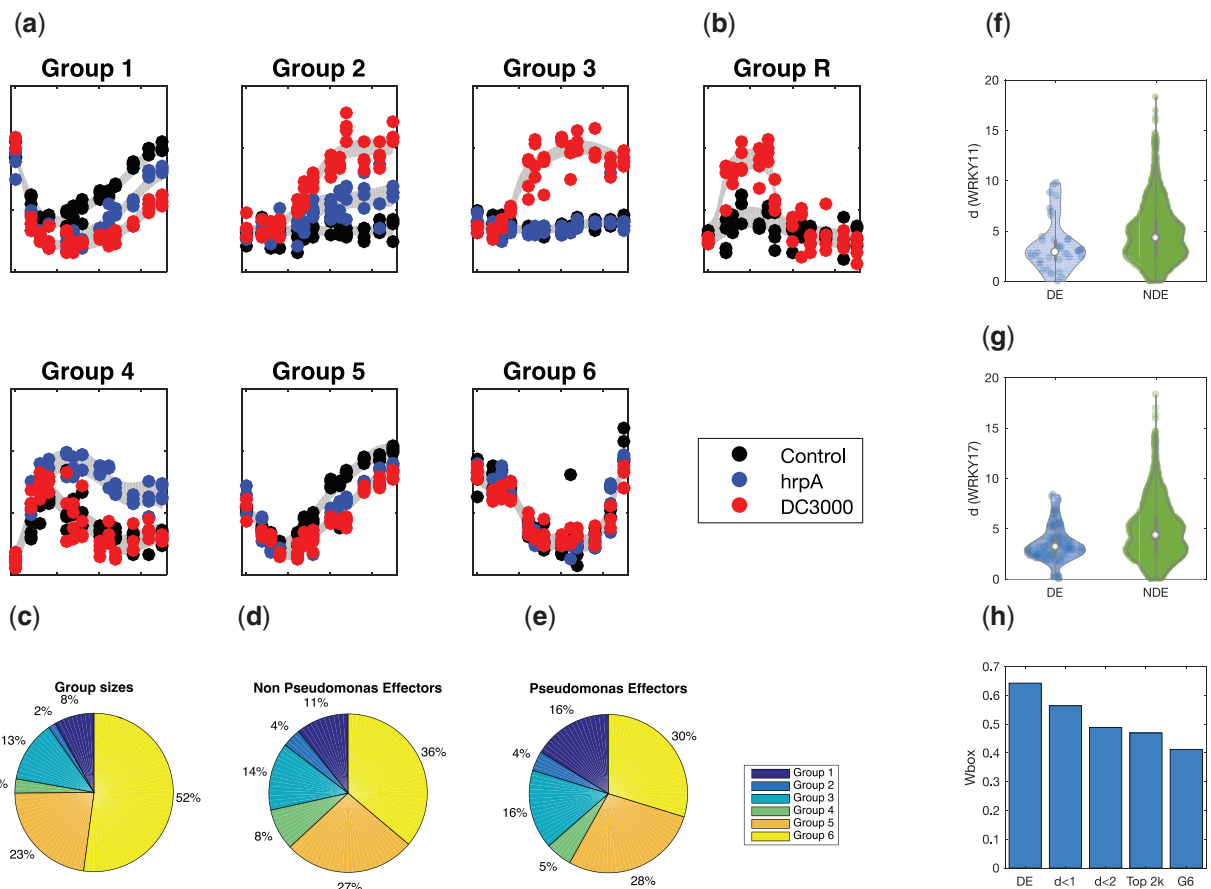
In our previous examples, inference relied on the existence of explicit branch labels. In some cases, however, branch labels may be incomplete or missing entirely. For example, in a collection of single cell transcriptomics data there may be various cell types, including some that cannot be unambiguously assigned to a particular branch *a priori*. We can attempt to infer the branch labels,  $z$ , using Markov chain Monte Carlo (MCMC). Here, we assume partially labelled data, with a subset of branch labels known, and the remainder unknown, denoted  $\mathbf{z} = [\mathbf{z}^{(labelled)}, \mathbf{z}^{(unlabelled)}]$ . When branch labels are known, they can be fixed, whilst unknown branch labels are initialized stochastically, and updated via a Gibbs sampler, similar to the usage in [Stegle et al. \(2010\)](#). For an  $n$ -component branching process, the unknown label for cell  $i$ , is Gibbs sampled conditional on the observation data and branch assignment of all other cells:

$$P(z_i = I | \mathbf{t}, \mathbf{z} \setminus z_i, \mathbf{y}, \theta) = \frac{P(\mathbf{y} | z_i = I, \mathbf{z} \setminus z_i, \mathbf{t}, \theta)}{\sum_{k=1}^n P(\mathbf{y} | z_i = k, \mathbf{z} \setminus z_i, \mathbf{t}, \theta)},$$

with hyperparameters updated conditional on all branch labels using hybrid Monte Carlo (HMC):

$$\theta \sim P(\theta | \mathbf{t}, \mathbf{z}, \mathbf{y})$$

To test the accuracy of our B-RGPs on partially labelled data we generated observations from the simple branching process outlined in [Supplementary Section S2](#). We first generated a set of test input locations,  $t \sim \mathcal{N}(0, 5\mathbb{I})$ , with observation data generated as noisy instances of the process. We then attempted to infer branch labels and hyperparameters within an MCMC scheme, with labels updated via Gibbs sampling, and hyperparameters sampled using Hybrid Monte Carlo. A subset of data points,  $n$ , were assigned the correct branch label, where  $n/N \in [0, 0.1, 0.25, 0.5]$  and  $N$  indicates the



**Fig. 2.** Branching processes were fitted to the three *Arabidopsis* time series, with hyperparameters optimized to MAP values, and the BIC used to select optimal branching structure. (a) Example expression profile plots for each of the different classes of branching. (b) Example expression profile of a branch-recombinant structure within the dataset. (c) The prevalence of each of the six groups within the dataset, compared to the breakdown of non-*Pseudomonas* effector targets (d), and *Pseudomonas*-effector targets show a clear enrichment of effector genes (e). (f, g) The Euclidean distance of branching times of genes from that of WRKY11/17 is statistically lower in genes that are DE in WRKY11/17 knockouts versus those that are NDE, indicating that perturbation times are predictive of direct and indirect targets of WRKY11/17. (h) The prevalence of Wbox motifs decreases amongst sets of genes whose branch times are increasingly distant from WRKY11

total number of observations, with the remaining data points randomly assigned to either branch with equal probability and updated within the MCMC. Five randomly initiated runs were used, with 20 000 steps in the MCMC chain, and the first 5000 discarded for burn-in. An example of the initial branch assignment is shown in Supplementary Figure S6a, with red indicating data points initially assigned to branch 1, and blue assigned to branch 2. An example fit (and updated branch labels) is shown for step 20 000 in Supplementary Figure S6b. The accuracy of classification is summarized using receiver operating characteristic (ROC) curves in Supplementary Figure S6c and d. We note good overall ability to infer branch labels even for the unlabeled case.

#### 4 *Arabidopsis thaliana* transcriptional branching in response to *Pseudomonas syringae*

To evaluate the utility of B-RGPs on a genome scale applications, we used our framework to investigate transcriptional branching in model plant organism *A. thaliana* in response to infection with hemibiotrophic bacterial pathogen *Pseudomonas syringae*. Recent studies by Lewis et al. (2015) (GEO GSE56094) have provided highly temporally resolved transcriptional datasets for *Arabidopsis* following inoculation with disease-causing *Pseudomonas syringae* pv.

tomato DC3000, and a disarmed mutant strain *hrpA* using bulk microarray measurements. The DC3000 variant delivers 28 effector proteins that subvert the plant's immune response; the disarmed *hrpA* mutant lacks the apparatus for effector delivery and thus elicits a classical immune response. Yang et al. (2016) developed a two-component branching GP to investigate transcriptional bifurcations between time series of *hrpA*- and DC3000-inoculated cells. Here, we extend this analysis by simultaneously deciphering the branching structure that exists between all 3 time series [mock/control, virulent (DC3000) and innate immune (*hrpA*) responses].

For each gene in the three datasets, we consider a number of possible branching structures: *hrpA* branches from the control, with DC3000 branching from *hrpA* at a later point (Group 1), or *hrpA* and DC3000 independently branch from the control (Group 2), which collectively represent immune response genes that are targeted by effectors; DC3000, but not *hrpA*, branches from control (Group 3), representing host susceptibility genes that have been targeted by effectors; *hrpA*, but not DC3000, branches from the control (Group 4), likely representing immune genes that have been targeted by effectors prior to their natural immune response times; both DC3000 and *hrpA* jointly branch from the control, but not from one another (Group 5), representing core immune response genes not targeted by effectors; no branching exists (Group 6),

representing genes unaffected by plant immunity or pathogen virulence strategies. Example expression patterns of individual genes from each of the six groups are shown in Figure 2a.

For Group 1, we assume that the hrpA-infected time series branches from the mock-infected time series, with the DC3000-infected time series branching from hrpA-infected:

$$\begin{aligned} f_{\text{mock}}(t) &\sim \mathcal{GP}(c, k_{\text{mock}}(t, t')), \\ f_{\text{hrpA}}(t) &\sim \mathcal{GP}(f_{\text{mock}}(t), CP_{t_{b_1}}(K_0, k_{\text{hrpA}}(t, t'))), \\ f_{\text{DC3000}}(t) &\sim \mathcal{GP}(f_{\text{hrpA}}(t), CP_{t_{b_2}}(K_0, k_{\text{DC3000}}(t, t'))), \end{aligned}$$

where observation data was assumed to be a noisy instances of these functions, e.g.  $y_{\text{mock}}(t) = f_{\text{mock}}(t) + \varepsilon$ . For Group 2, we have hrpA-infected and the DC3000-infected time series independently branching from the mock-infected time series:

$$\begin{aligned} f_{\text{mock}}(t) &\sim \mathcal{GP}(c, k_{\text{mock}}(t, t')), \\ f_{\text{hrpA}}(t) &\sim \mathcal{GP}(f_{\text{mock}}(t), CP_{t_{b_1}}(K_0, k_{\text{hrpA}}(t, t'))), \\ f_{\text{DC3000}}(t) &\sim \mathcal{GP}(f_{\text{mock}}(t), CP_{t_{b_2}}(K_0, k_{\text{DC3000}}(t, t'))). \end{aligned}$$

Collectively Groups 1 and 2 should represent immune response genes targeted by effectors, and therefore associated with the onset of disease. For Group 3, we have mock-infected and hrpA-infected datasets drawn from an identical process, with the DC3000-infected branching from this:

$$\begin{aligned} f_{\text{mock,hrpA}}(t) &\sim \mathcal{GP}(c, k_{\text{mock,hrpA}}(t, t')), \\ f_{\text{DC3000}}(t) &\sim \mathcal{GP}(f_{\text{mock,hrpA}}(t), CP_{t_{b_1}}(K_0, k_{\text{DC3000}}(t, t'))). \end{aligned}$$

This group represents genes not associated with the immune response that are nonetheless targeted by effectors, and may therefore represent those functioning in metabolism. For Group 4, we have mock-infected and DC3000-infected datasets drawn from an identical process, with the hrpA-infected branching from this:

$$\begin{aligned} f_{\text{mock,DC3000}}(t) &\sim \mathcal{GP}(c, k_{\text{mock,DC3000}}(t, t')), \\ f_{\text{hrpA}}(t) &\sim \mathcal{GP}(f_{\text{mock,DC3000}}(t), CP_{t_{b_1}}(K_0, k_{\text{hrpA}}(t, t'))). \end{aligned}$$

These genes likely reflect downstream immune response genes that are targeted very early by effectors. For Group 5, we have hrpA-infected and DC3000-infected datasets drawn from an identical process that branches from mock-infected:

$$\begin{aligned} f_{\text{mock}}(t) &\sim \mathcal{GP}(c, k_{\text{mock}}(t, t')), \\ f_{\text{hrpA,DC3000}}(t) &\sim \mathcal{GP}(f_{\text{mock}}(t), CP_{t_{b_1}}(K_0, k_{\text{hrpA,DC3000}}(t, t'))). \end{aligned}$$

These genes represent immune response genes not targeted by effectors. Finally, for Group 6, we have all datasets drawn from an identical process:

$$f_{\text{mock,hrpA,DC3000}}(t) \sim \mathcal{GP}(c, k_{\text{mock}}(t, t')),$$

representing genes that are unbranched, i.e. not differentially expressed. Because these datasets correspond to bulk observations from microarrays with well-defined measurement times we assumed smooth functions throughout, and therefore, in all cases, the covariance functions were taken to be squared-exponentials, e.g.  $k_{\text{mock}}(t, t') = SE_{\theta_{\text{mock}}}(t, t') = \sigma_{\text{mock}}^2 \exp((t - t')/2l_{\text{mock}})^2$ , where  $\theta_{\text{mock}}$

$= [l_{\text{mock}}^2, \sigma_{\text{mock}}^2]$  denotes a set of mock dataset-specific hyperparameters, and hyperparameters were optimized to their ML or MAP values. We assumed the following prior distributions: the first branch time was Gamma distributed,  $t_{b_1} \sim \Gamma(2, 2)$ , with the second branch also Gamma distributed,  $t_{b_2} \sim \Gamma(4, 2)$ , and the change-point transition rate was Gaussian distributed,  $s \sim \mathcal{N}(0, 0.5)$ . All other hyperparameters were optimized to their ML values. Finally, we selected the optimal group based on the Bayesian information criterion (BIC).

In Supplementary Figure S7a, we indicate the branch time between control and hrpA time series using B-RGPs versus that obtained using the Gaussian process two-sample (GP2S; Stegle *et al.*, 2010; Supplementary Fig. S7b). Here, the GP2S approach incorrectly identified a peak perturbation time at  $t = 0$ , before Arabidopsis could mount an immune response. This peak was notably absent in our B-RGP approach. To further gauge the accuracy of our approach, we compared the estimated branch times between hrpA and DC3000 using B-RGPs (Supplementary Fig. S6c) to that obtained using the perturbation times previously estimated in Yang *et al.* (2016) (Supplementary Fig. S6d). The analysis in Yang *et al.* (2016) provide 90% confidence intervals for branch time estimates, and we note that our MAP estimation falls within these bounds in 67% of cases. Of the remaining genes, 27% of our MAP estimates lie to the right of the confidence bounds, and 5% to the left of the confidence bounds, suggesting that our approach has a tendency to estimate later branch times than that of Yang *et al.* (2016). This is likely due to differences in the prior distributions over branch times. Indeed, if we plot the estimated branch time using our method versus the PT approach for the 27% of genes noted above, we see a strong correlation ( $R = 0.8507$ ). Together, these results suggest that, although there is good agreement between the methods for a large fraction of cases, inference of branch time in a subset of the observations may be unidentifiable.

Our results indicate that approximately 50% of genes were unperturbed by either the hrpA or DC3000 strains, in agreement with previous studies based on pairwise comparison of the time series using mixtures of Gaussian Processes (Lewis *et al.*, 2015), where 52% of genes were identified as being differentially expressed in control versus hrpA or control versus DC3000.

Since DC3000 is known to subvert the basal immune response of Arabidopsis, we hypothesized that the expression of a subset of genes in the DC3000-infected dataset might converge fully back to control levels later in the infection process. To identify such genes, we also fitted a two-component branch-recombinant GP using the control and DC3000-infected time series only, again using the BIC to distinguish between genes undergoing branching and recombination from those undergoing branching alone.

$$\begin{aligned} f_{\text{mock}}(t) &\sim \mathcal{GP}(c, k_{\text{mock}}(t, t')), \\ f_{\text{DC3000}}(t) &\sim \mathcal{GP}(f_{\text{DC3000}}(t), CP_{t_{b_1}}(CP_{t_{b_1}}(K_0, k_{\text{mock}}(t, t')), K_0)). \end{aligned}$$

An example branch-recombinant expression profile is shown in Figure 2b. We note that relatively few genes were identified as having their expression levels fully converge back to control levels. Of those that did, none were identified as being targets of effectors or previously implicated in the response to *P. syringae*, suggesting that the full suppression of early immune response genes to control levels is not required for infection to advance.

Gene Ontology analysis identified several highly enriched terms across the first five groups (see Supplementary Table S1), suggesting distinct biological functions relating to pathogen response and

metabolic reprogramming. Prior to 3 h, the ontologies represent some of the earliest transcriptional processes targeted by effectors. Consequently, there is a diverse array of GOs represented. Notable are the combination of proteolytic, ribosome, vitamin and amino acid metabolic and transport processes. This is indicative of assembly of the processing machinery to enable effector mediated reprogramming of core cellular processes. Between 3 and 5 h post infection (hpi) the impact of effectors was evident by the number of GOs identified, with processes associated with nuclear processes, in particular chromatin remodelling, nuclear transport and transcription, most highly enriched. Other GO processes, such as hormone responses and primary metabolism, were, unexpectedly, less abundant. While Lewis et al. (2015) also reported evidence for chromatin remodelling in this dataset, B-RGPs provided much better temporal resolution. As the effector-driven virulence programme proceeds, but prior to bacterial multiplication (5–8 hpi), there is a strong enrichment of terms related to adenylyl ribonucleotide binding, reflecting the high energy demands at this phase of the infection process, when *Pseudomonas* effectors have suppressed immune responses and are reconfiguring the metabolism to facilitate pathogen growth.

To further investigate the nature of these groups, we looked for enrichment of known targets of effectors of various pathogens (Mukhtar et al., 2011). We first checked for enrichment of targets of non-*Pseudomonas* effectors, hypothesizing that the Arabidopsis immune response to different pathogens might be conserved (Mukhtar et al., 2011). Figure 2c and d, shows that these groups were indeed enriched. Next, we checked for enrichment of Avr and Hop effectors that are present in several strains of *Pseudomonas syringae*, and were again enriched in DC3000-responsive groups (Fig. 2c and e).

We next looked for enrichment of genes with known pathogen-response phenotypes, using TAIR (Huala et al., 2001) to query for genes using the terms ‘*Pseudomonas*’, ‘*Botrytis*’ and ‘*Peronospora*’. In Supplementary Figure S8, we indicate the frequency of pathogen-response genes within various groups, and our results show a distinct enrichment for *Pseudomonas*-related and *Botrytis*-related genes amongst the various immune responsive and disease-responsive groups.

Finally, we investigated whether inferred branch times of key regulators were predictive of branching of the direct and indirect targets of key regulators. Here, we focused on WRKY11 and WRKY17, known to be amongst the earliest branching transcription factors (TFs) implicated in the *Arabidopsis* response to *P. syringae* (Journot-Catalino et al., 2006). Both genes showed branching between control and hrpA, and between hrpA and DC3000 consistent with (i) their immune-responsive expression and (ii) their suppression by DC3000 effectors. Genes that branched between control and hrpA and between hrpA and DC3000 were assigned a Euclidean distance ( $d$ ) based on the position of their branch times with respect to that of WRKY11 or WRKY17. We then compared the distributions of these Euclidean distances for the subset of genes identified as being differentially expressed (DE) in knockout mutants of WRKY11/17 (Journot-Catalino et al., 2006) versus the distribution of the subset of genes that were not differentially expressed (NDE) in those mutants. Our results show that DE genes had significantly smaller Euclidean distances than NDE genes ( $p < 0.05$  for WRKY11 and  $P < 0.005$  for WRKY17 using two-sided Student’s  $t$ -test; Fig. 2f and g), suggesting that genes that branched at similar times to WRKY11/17 were likely to represent a core set of genes targeted by the pathogen’s virulence strategy. WRKY11/17 are TFs and could exert direct regulation of their targets by binding to their regulatory elements. To check this, we searched for the presence of WRKY motifs within a 1 kb promoter region using FIMO (Grant et al.,

2011); specifically, the stringent WRKY binding site (Wbox) motif, TWGTTGACYWWWW, identified by Ciolkowski et al. (2008). Here, we looked at the frequency of Wbox motifs ( $P < 0.0001$ ) in sets of genes whose branch times were increasingly distal from WRKY11. These groups were based on: (i) genes whose Euclidean distance  $d < 1$ , representing the closest 156 genes (see Supplementary Fig. S9); (ii) genes whose Euclidean distance  $d < 2$ , representing the closest 454 genes; and (iii) the closest 2000 genes. As positive and negative controls, we also included the 157 genes that were identified as DE in the WRKY11 knockout line compared to control, and 2000 genes randomly selected from Group 6 (genes with no branching). Our results showed a clear trend of increasing frequency of Wbox motifs in sets of genes whose branch times were closest to that of WRKY11 (Fig. 2h; see also Supplementary Table S2). Altogether, these results suggest that estimation of branch times may be useful for identifying direct and indirect targets of perturbed genes, and more generally demonstrate the efficacy of B-RGPs for extracting temporally resolved information from complex biological datasets.

## 5 Discussion

The ability to identify and quantify branching and recombination processes from systems-level measurements has a variety of important applications in the biological sciences. Here, we have outlined a general framework for the composition of covariance functions that allow for the prior specification of branch-recombination processes of arbitrary complexity, both in terms of the number of branches and richness of dynamics, via simple compositional of covariance functions within a HGP framework. As well as specifying arbitrarily complex processes, all hyperparameters could be optimized via gradient based approaches, resulting in more accurate inference of branch times compared to existing approaches, although inference took slightly longer.

Here, we applied B-RGPs to a time-series microarray data of *Arabidopsis thaliana* infected with a bacterial pathogen *Pseudomonas syringae*. By explicitly enumerating over all possible branch structures, i.e. all 1, 2 and 3 branch structures, and using the AIC as a selection criterion, we were able to infer the branch structure for each gene. Whilst exhaustive iteration will not necessarily be possible for more complex datasets with  $>3$  time series, we note that greedy approaches based on merging of time series could instead be used.

More generally, B-RGPs represents a flexible approach for the analysis of branching and recombination in time series datasets. This approach can be thought of as a natural extension to two-sample based approaches, allowing analysis of arbitrary numbers of time series.

Whilst here we focused on branching as a function of time, our approach is equally amenable to branching as a function of any other variable, such as expression level of a specific regulator. An intriguing possibility is therefore to incorporate B-RGPs into existing GP-based approaches for the inference of nonlinear dynamical systems (Penfold and Wild, 2011; Penfold et al., 2012; Penfold et al., 2015a, b; Äijö and Lähdesmäki, 2009), which would naturally allow inference of nonstationary nonlinear dynamical systems, such as temporally or spatially varying networks.

In addition, we envisage that B-RGPs could also be used to capture transcriptional dynamics underpinning cell fate decisions from single cell transcriptomics data. For this, cells are first pseudotemporally ordered along a developmental axis using a combination of dimensionality reduction techniques and curve-fitting or graph-theoretic approaches (Bendall et al., 2014; Ji and Ji, 2016; Marco



*et al.*, 2014; Setty *et al.*, 2016; Trapnell *et al.*, 2014). Once ordered along pseudotime, B-RGPs could capture the branching dynamics of individual genes, thus identifying the earliest molecular events controlling cell fate decisions. Alternatively, B-RGPs could be used to directly model cell fate decisions. Recent studies by (Reid and Wernisch, 2016) have shown how Gaussian process latent variable modes (GPLVMs), can be used to pseudotemporally order genes along a developmental axis, with a key advantage over other pseudotime approaches: the incorporation of capture time into the inference procedure. However, due to a previous lack of treatment for branching in GP models, the approach of Reid and Wernisch (2016) did not explicitly allow for pseudotemporal ordering of datasets with branching behavior. The incorporation of B-RGPs into a GPLVM model naturally allows for pseudotemporal ordering over branching process, whilst retaining the ability to leverage highly informative data, such as capture time (Penfold *et al.*, 2017).

## Acknowledgements

We thank Ufuk Günesdogan and Naoko Irie for critical reading of the manuscript, and the Surani lab for useful insights. We thank Sabine Dietmann for early discussions about branching processes. We also thank Charles Bradshaw for help with high performance computing, and the Gurdon Institute core facilities for continued support. Finally, we thank Magnus Ratray and Jing Yang for useful insights and discussion of statistical branching models.

## Funding

C.A.P. is supported by Wellcome Trust Grant (083089/Z/07/Z). A.S. is supported by a 4-year Wellcome Trust PhD Scholarship and Cambridge International Trust Scholarship. C.A.P. and A.S. also acknowledge funding from the BBSRC-EPSC funded OpenPlant Synthetic Biology Research Centre (BB/L014130/1). M.A.S. is supported by HFSP and a Wellcome Trust Senior Investigator Award. J.R. and L.W. are funded by the UK Medical Research Council (Grant Ref MC\_U105260799). Y.H. is supported by a studentship from the James Baird Fund, University of Cambridge. M.G. acknowledges funding from BBSRC grants BB/F005903/1 and BB/P002560/1. Z.G. acknowledges funding from the Alan Turing Institute, Google, Microsoft Research and EPSRC Grant EP/N014162/1.

*Conflict of Interest:* none declared.

## References

Äijö, T. and Lähdesmäki, H. (2009) Learning gene regulatory networks from gene expression measurements using non-parametric molecular kinetics. *Bioinformatics*, **25**, 2937–2944.

Bendall, S.C. *et al.* (2014) Single-cell trajectory detection uncovers progression and regulatory coordination in human B cell development. *Cell*, **157**, 714–725.

Boller, T. and He, S.Y. (2009) Innate immunity in plants: an arms race between pattern recognition receptors in plants and effectors in microbial pathogens. *Science*, **324**, 742–744.

Breeze, E. *et al.* (2011) High-resolution temporal profiling of transcripts during Arabidopsis leaf senescence reveals a distinct chronology of processes and regulation. *Plant Cell*, **23**, 873–894.

Ciolkowski, I. *et al.* (2008) Studies on DNA-binding selectivity of WRKY transcription factors lend structural clues into WRKY-domain function. *Plant Mol. Biol.*, **68**, 81–92.

Friedmann-Morvinski, D. and Verma, I.M. (2014) Dedifferentiation and reprogramming: origins of cancer stem cells. *EMBO Rep.*, **15**, 244–253.

Grant, C.E. *et al.* (2011) FIMO: scanning for occurrences of a given motif. *Bioinformatics*, **27**, 1017–1018.

Gurdon, J.B. (1962) The developmental capacity of nuclei taken from intestinal epithelium cells of feeding tadpoles. *Development*, **10**, 622–640.

Hensman, J. *et al.* (2013) Hierarchical Bayesian modelling of gene expression time series across irregularly sampled replicates and clusters. *BMC Bioinformatics*, **14**, 252.

Huala, E. *et al.* (2001) The Arabidopsis information resource (TAIR): a comprehensive database and web-based information retrieval, analysis, and visualization system for a model plant. *Nucleic Acids Res.*, **29**, 102.

Ji, Z. and Ji, H. (2016) TSCAN: pseudo-time reconstruction and evaluation in single-cell RNA-seq analysis. *Nucleic Acids Res.*, **44**, e117.

Jones, J.D. and Dangl, J.L. (2006) The plant immune system. *Nature*, **444**, 323–329.

Journot-Catalino, N. *et al.* (2006) The transcription factors WRKY11 and WRKY17 act as negative regulators of basal resistance in *Arabidopsis thaliana*. *Plant Cell*, **18**, 3289–3302.

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**, 180–193.

Lewis, L.A. *et al.* (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**, 3038–3064.

Lloyd, J.R. *et al.* (2014) Automatic construction and natural-language description of nonparametric regression models. *arXiv*, **1402**, 4304.

Marco, E. *et al.* (2014) Bifurcation analysis of single-cell gene expression data reveals epigenetic landscape. *Proc. Natl. Acad. Sci. USA*, **111**, E5643–E5650.

Mukhtar, M.S. *et al.* (2011) Independently evolved virulence effectors converge onto hubs in a plant immune system network. *Science*, **333**, 596–601.

Penfold, C.A. and Wild, D.L. (2011) How to infer gene networks from expression profiles, revisited. *Interface Focus*, **1**, 857–870.

Penfold, C.A. *et al.* (2012) Nonparametric Bayesian inference for perturbed and orthologous gene regulatory networks. *Bioinformatics*, **28**, i233–i241.

Penfold, C.A. *et al.* (2015a) Inferring orthologous gene regulatory networks using interspecies data fusion. *Bioinformatics*, **31**, i97–105.

Penfold, C.A. *et al.* (2015b) CSI: a nonparametric Bayesian approach to network inference from multiple perturbed time series gene expression data. *Stat. Appl. Genet. Mol. Biol.*, **14**, 307.

Penfold, C.A. *et al.* (2017) Nonparametric Bayesian inference of transcriptional branching and recombination identifies regulators of early human germ cell development. *bioRxiv*, doi: 10.1101/167684.

Poincaré, H. (1885) Sur l'équilibre d'une masse fluide animée d'un mouvement de rotation. *Acta Math.*, **7**, 259–380.

Quinero-Candela, J. and Rasmussen, C.E. (2005) A unifying view of sparse approximate Gaussian process regression. *J. Mach. Learn. Res.*, **6**, 1939–1959.

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**, 2973–2980.

Setty, M. *et al.* (2016) Wishbone identifies bifurcating developmental trajectories from single-cell data. *Nat. Biotechnol.*, **34**, 637–645.

Stegle, O. *et al.* (2010) A robust Bayesian two-sample test for detecting intervals of differential gene expression in microarray time series. *J. Comput. Biol.*, **17**, 355–367.

Takahashi, K. and Yamanaka, S. (2006) Induction of pluripotent stem cells from mouse embryonic and adult fibroblast cultures by defined factors. *Cell*, **126**, 663–676.

Trapnell, C. *et al.* (2014) The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nat. Biotechnol.*, **32**, 381–386.

Windram, O. *et al.* (2012) Arabidopsis defense against Botrytis cinerea: chronology and regulation deciphered by high-resolution temporal transcriptomic analysis. *Plant Cell*, **24**, 3530–3557.

Yang, J. *et al.* (2016) Inferring the perturbation time from biological time course data. *Bioinformatics*, **32**, 2956.

Zawadzka, M. *et al.* (2010) CNS-resident glial progenitor/stem cells produce Schwann cells as well as oligodendrocytes during repair of CNS demyelination. *Cell Stem Cell*, **6**, 578–590.