Computers in Biology and Medicine 143 (2022) 105268

Contents lists available at ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/compbiomed

PairGP: Gaussian process modeling of longitudinal data from paired multi-condition studies

Michele Vantini [a,*], Henrik Mannerström [a], Sini Rautio [a], Helena Ahlfors [b], Brigitta Stockinger [b], Harri Lähdesmäki [a]

[a] Department of Computer Science, Aalto University, Konemiehentie 2, Espoo, 02 150, Finland
[b] The Francis Crick Institute, 1 Midland Road, London, NW1 1AT, United Kingdom

ARTICLE INFO

Keywords:
Gaussian processes
Gene expressions
Time-series
Pairing effect
Differential condition analysis

ABSTRACT

High-throughput technologies produce gene expression time-series data that need fast and specialized algorithms to be processed. While current methods already deal with different aspects, such as the non-stationarity of the process and the temporal correlation, they often fail to take into account the pairing among replicates.

We propose PairGP, a non-stationary Gaussian process method to compare gene expression time-series across several conditions that can account for paired longitudinal study designs and can identify groups of conditions that have different gene expression dynamics. We demonstrate the method on both simulated data and previously unpublished RNA sequencing (RNA-seq) time-series with five conditions. The results show the advantage of modeling the pairing effect to better identify groups of conditions with different dynamics. The pairing effect model displays good capabilities of selecting the most probable grouping of conditions even in the presence of a high number of conditions.

The developed method is of general application and can be applied to any gene expression time series dataset. The model can identify common replicate effects among the samples coming from the same biological replicates and model those as separate components. Learning the pairing effect as a separate component, not only allows us to exclude it from the model to get better estimates of the condition effects, but also to improve the precision of the model selection process. The pairing effect that was accounted before as noise, is now identified as a separate component, resulting in more accurate and explanatory models of the data.

1. Background

Gene expression time-series studies have become popular as they can reveal dynamics of transcriptional processes. These studies typically use longitudinal experimental designs where repeated measurements (over time) of each cell sample are collected. A common study design involves comparisons between treatments, or conditions, and the goal is to identify groups of conditions that have different gene expression dynamics. Further, to reduce variability between conditions and to increase statistical power, biological samples in different conditions are typically matched, resulting in paired longitudinal designs. Thus, it is important to take the paired design into account in the data analysis in order to reveal the true differences between different treatments.

Gene expression microarray and RNA-seq techniques allow quantitative, genome-wide analysis of gene expression levels. A number of software tools are available for statistical analysis of gene expression data measured by microarrays (e.g. LIMMA [18]) and RNA-seq (e.g. DEseq [3] and edgeR [20]). These tools rely on linear and generalized linear models, use empirical Bayes to share information between genes, allow modeling complex experimental designs, and support testing a variety of hypothesis, but are not designed for longitudinal studies that involve repeated measurements of individuals over time. Standard methods for longitudinal data analysis include linear and generalized linear mixed effect (LME) models, as implemented in e.g. lme4 package [6]. Bayesian alternatives for modeling gene expression time-series data have been proposed e.g. in Refs. [4,5] that also support non-Gaussian likelihood models. Methods of gene expression time-series data analysis include also *lmms* [23] and ImpulseDE2 [11]. The former is based on linear mixed models and ANOVA log likelihood ratio tests, while the latter is based on an impulse model as a continuous representation of

* Corresponding author.
E-mail addresses: vantinimichele95@gmail.com (M. Vantini), henrik.mannerstrom@gmail.com (H. Mannerström), rautiosini@gmail.com (S. Rautio), helena.ahlfors@gmail.com (H. Ahlfors), brigitta.stockinger@crick.ac.uk (B. Stockinger), harri.lahdesmaki@aalto.fi (H. Lähdesmäki).

https://doi.org/10.1016/j.compbiomed.2022.105268
Received 10 November 2021; Received in revised form 23 January 2022; Accepted 23 January 2022
Available online 26 January 2022

temporal responses.

A number of non-linear, non-stationary and non-parametric methods for gene expression time series have been proposed using Gaussian processes (GP). Yuan was among the first who used GPs to model gene expression time course data [26]. A number of improved methods have been proposed, such as methods that can account for outliers [22], a method for analyzing multiple conditions [2], methods that identify time intervals of differential expression [13,22], and methods for accounting time delays between replicates and non-Gaussian likelihood models [1]. However, none of these tools can account for paired experimental designs that are commonly used in biological studies. Similar ideas have been proposed in the context of GP-based clustering of time-series data [15], where authors propose a hierarchical GP regression model. Nonetheless, the effects in Ref. [15] are not across replicate pairs but, instead, a different replicate effect is learned for each individual condition. To that end, Spies et al. [21] provide an extensive review of a large selection of methods proposed in the literature for time course data.

Recently, we have developed GP based methods to implement Bayesian non-parametrics for longitudinal studies [8,24] that can also be applied to data from paired longitudinal designs. However, posterior sampling for such models has high computational cost and can become prohibitive when analyzing tens of thousands of genes. Here, we propose a non-stationary GP method for paired, multi-condition longitudinal designs that provides efficient analysis for genome-wide studies.

2. Methods

Each measured gene expression time-series is modeled as a combination of three components; 1) the response model, 2) the pairing model, and 3) uncorrelated random noise fluctuations. The response model is inferred from the data, so that all treatments that produce similar responses share a common response model. The pairing model is shared by all measurements coming from the same biological replicate or batch, and models the deviation from the response model. To enforce that the pairing model does not confound the response model, the sum of all the pairing model components is constrained to zero, as explained below. The model considers each gene separately. The measured gene expression x is transformed as $y = \log(x + 1)$ so that it can be more accurately modeled by a normal distribution. Most gene expression experiments are "hit-and-run", where the changes are rapid in the beginning and then slow down, thus, making it a non-stationary process. To model the non-stationarity, a user is given the choice to transform the wall-clock time $\tilde{t}$ as $t = \omega(\tilde{t}) = \log(1 + \tilde{t})$ and the transformed time is used as an input the kernel function as explained below. This transformation was used in all analyses reported below.

The standardized measurements of treatment (condition) $c \in \{1, \ldots, C\}$ and pairing $p \in \{1, \ldots, P\}$ is modeled as

$$y_{cp}(t) = f_r(t) + f_p(t) + \varepsilon, \tag{1}$$

where $\varepsilon \sim \mathcal{N}(0, \sigma_\varepsilon^2)$. Each response effect f_r is a GP with the standard exponentiated quadratic (EQ) kernel

$$k_r(t, t') = \sigma_r^2 \exp\left[-\frac{1}{2}\ell_r^{-2}(t - t')^2 \right], \tag{2}$$

where σ_r^2 is the variance and ℓ_r the length scale of response effect r. For each pairing p, the pairing effect f_p is modeled with a centered EQ kernel

$$k_\rho((p, t), (p, t')) = \sigma_\rho^2 \exp\left[-\frac{1}{2}\ell_\rho^{-2}(t - t')^2 \right], \tag{3}$$

where σ_ρ^2 is the common variance and ℓ_ρ the common length scale of the pairing effect. The centered EQ kernel has negative covariance between the pairing effects f_p and $f_{p'}(p \neq p')$ to force their sum to zero, i.e,

$$k_\rho((p, t), (p', t')) = -\frac{1}{P - 1}\sigma_\rho^2 \exp\left[-\frac{1}{2}\ell_\rho^{-2}(t - t')^2 \right], \tag{4}$$

when $p \neq p'$ [24]. The centered EQ kernel quarantees that $\sum_{p=1}^{P} f_p(t) = 0$ for all values of t. Note that the response and pairing GPs are non-stationary as the logarithmic time transformation corresponds to input-warped GPs with kernel $k(\tilde{t}, \tilde{t}') = \sigma^2 \exp\left[-\frac{1}{2}\ell^{-2}(\omega(\tilde{t}) - \omega(\tilde{t}'))^2 \right]$ (similarly for the centered EQ kernel). Prior distributions of hyperparameters used to analyze real data are described below.

2.1. Model selection

For each gene, all the partitionings of the treatments are modeled, and the one with the largest marginal likelihood (type-II) is selected as the correct response model. For example, an experiment with three treatments c_1, c_2 and c_3 evaluates five different partitionings (models) for each gene: 1) all the three treatments have a similar temporal response, and there is only one response model: $r_1 = \{c_1, c_2, c_3\}$; 2) treatment c_1 has a different response compared to c_2 and c_3, and the two response models are $r_1 = \{c_1\}$ and $r_2 = \{c_2, c_3\}$; 3) same as (2) but with treatment c_2 singled out, $r_1 = \{c_2\}$ and $r_2 = \{c_1, c_3\}$; 4) same as (2) but with treatment c_3 singled out, $r_1 = \{c_3\}$ and $r_2 = \{c_1, c_2\}$; and 5) all three treatments produce different responses, $r_1 = \{c_1\}$, $r_2 = \{c_2\}$, and $r_3 = \{c_3\}$.

More generally, given that an experiment contains C treatments, they can be partitioned into B_C different partitionings (or models), where

$$B_C = \sum_{k=0}^{C-1} \binom{C-1}{k} B_k \tag{5}$$

is the Bell number. For example, Bell number for 2, 3, 4 and 5 treatments are $B_2 = 2$, $B_3 = 5$, $B_4 = 15$, and $B_5 = 52$, respectively. For each partitioning, we evaluate the marginal likelihood

$$\log p(\boldsymbol{y}|X, \boldsymbol{\theta}) = -\frac{1}{2}\boldsymbol{y}^T\left(\boldsymbol{K}_{X,X} + \sigma_\varepsilon^2 I\right)^{-1}\boldsymbol{y} - \frac{1}{2}\log\left|\left(\boldsymbol{K}_{X,X} + \sigma_\varepsilon^2 I\right)\right| - \frac{n}{2}\log 2\pi, \tag{6}$$

where $\boldsymbol{y} \in \mathbb{R}^{C \cdot P \cdot T}$ contains the standardized gene expression data for a gene from all C treatments, P replicates and T time points, $X = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{C \cdot P \cdot T})$ contains the explanatory covariates (treatment c, replicate p and time point t) for each measurement, $\boldsymbol{\theta}$ is a vector containing all the kernel hyperparameters, $\boldsymbol{K}_{X,X}$ is the sum of the response covariance matrix and the pairing covariance matrix defined by the centered EQ kernels, σ_ε^2 is the Gaussian random noise variance, and $n = CPT$. An example of the covariance matrix $\boldsymbol{K}_{X,X}$ and its components $\boldsymbol{K}_r$ and $\boldsymbol{K}_p$ are shown in Fig. 1.

We call the model presented above the pairing effect model. To assess the performance of this model, we compare it against the base model. The base model is obtained by optimizing one GP regression model for each possible subset of the condition set, and then combining the score of these models to have a score for each partitioning of the condition set. In other words, the log marginal likelihood $\log p(\boldsymbol{y}|X, \boldsymbol{\theta})$ of the models of different subsets is summed up to obtain the score for the partitioning that corresponds to the set of considered subsets. In the base model, we use EQ kernels which model the response functions, but not the pairing effect. The base model corresponds to the standard GP modeling approach used in several previous works [2,16]. In the pairing effect model we standardize the data of all the conditions together, which has the effect of preserving the pairing effect across conditions. On the other hand, in the base model we standardize separately the data of the sets of conditions that corresponds to the different response functions, since in this case we are not attempting to learn the pairing effect.

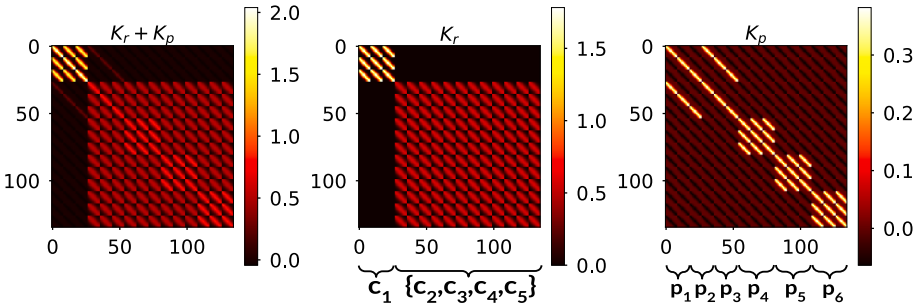

Fig. 1. An example of the covariance matrices. (left) the combination of covariance matrices $\boldsymbol{K}_{X,X} = \boldsymbol{K}_r + \boldsymbol{K}_p$. (middle) the response covariance matrix $\boldsymbol{K}_r$. (right) the pairing covariance matrix $\boldsymbol{K}_p$ without centering. In this example, there are data from 5 different conditions and 6 replicates. The first 3 replicates are paired across the conditions c_1 and c_2, and the last 3 replicates are paired across the conditions c_3, c_4 and c_5. Data are ordered by condition, then replicate and finally by time point. In this example, conditions are assumed to be partitioned into 2 response models $r_1 = \{c_1\}$ (upper left block in $\boldsymbol{K}_r$) and $r_2 = \{c_2, c_3, c_4, c_5\}$ (bottom right block in $\boldsymbol{K}_r$).

2.2. Prior distribution for kernel hyperparameters

Typically, the hyperparameter optimization is done by maximizing the log marginal likelihood of the model. This can be extended in a hierarchical structure by imposing prior distributions on the hyperparameters, also called hyperpriors. This allows us to include any prior information about the hyperparameters in the optimization problem. The kernel choice for the GP regression models is the exponentiated quadratic (EQ) kernel. This means that we can define hyperpriors for the variance σ^2 and the lengthscale ℓ. The information that we want to include are:

- the learned functions have to be smooth. As a result, the lengthscale parameters for the response effect kernels should be relatively high;
- the magnitude of the pairing effect should be small compared to the response effect, with the only exception of silent genes where the variation of the gene expression over time is approximately 0, thus the pairing effect can have higher variation. Thus, we need to constrain the variance parameter for the pairing effect kernel to be small.

We use the log-Gaussian distribution log-Normal(μ, σ^2) with $\mu = 0.5$, $\sigma^2 = 0.5$ as hyperprior distribution for the lengthscale of the response effect, exponential distribution Exp(λ) with $\lambda = 2$ for the pairing effect variance and log-Gaussian distribution log-Normal(μ, σ^2) with $\mu = 0$, $\sigma^2 = 0.5$ for the pairing effect lengthscale. We do not use here any hyperprior distribution on the noise variance σ_ε^2. The log-Normal hyperprior distribution applied on the lengthscale parameters of the kernel has a regularization effect for the model, as it prevents the model to learn too small values for this parameter, which are usually linked to more complex functions. We also note that, should the pairing effect be merely an offset term, such as those implemented in the classical linear mixed models, then the length scale prior can be changed to favour much larger values that will result in nearly constant-valued GPs.

The optimization is done w.r.t. to the following objective function

$$\arg\max_{\boldsymbol{\theta}} \log p(\boldsymbol{y}|X,\boldsymbol{\theta}) + \log p(\boldsymbol{\theta}), \tag{7}$$

where $p(\boldsymbol{\theta})$ corresponds to the hyperpriors. We use the above prior distributions for kernel hyperparameters when analyzing real microarray or RNA-seq data and optimize the above objective function. For simulated data we ignore the hyperpriors and optimize the standard marginal likelihood, i.e., $\log p(\boldsymbol{y}|X,\boldsymbol{\theta})$. We use the gradient-based method L-BFGS-B [7] for the optimization. The optimizer is run for a maximum of 1000 iterations with tolerance for deciding convergence equals to $1e^{-5}$.

2.3. Software

The above method is implemented using the GPy package [12]. The main contribution consists in the implementation of the proposed kernel for the pairing effect model. Also, the implemented software facilitate the use of it for gene expression time series data. Detailed instructions for the installation and the usage are available on the github page https://github.com/michelevantini/PairGP.

3. Data

The methods have been developed to be applied to data set with the following structure: the data contains N genes, C conditions (or treatments), P replicates and T irregularly sampled time points for each gene. That results in $C \times P$ time-series of length T for each gene. We used simulated data and two gene expression time-series data sets.

3.1. Simulated data

To generate simulated data, we simulated one GP for each response and one GP for each replicate pair using a fixed set of hyperparameters. To simulate the data for a condition c and for a replicate pair p we use the following formulation:

$$y_{cp}(t) = f_r(t) + f_p(t) + \varepsilon, \tag{8}$$

where f_r and f_p are *realizations* of GPs $f_r \sim \mathcal{GP}(\mathbf{0}, k_r(x,x'))$ and $f_p \sim \mathcal{GP}(\mathbf{0}, k_p(x,x'))$ and $\varepsilon \sim \mathcal{N}(0,\sigma_\varepsilon^2)$ is a random noise term. Recall that the C treatments (or conditions) result in R different responses, depending on the partitioning (or model; see below), and each treatment c belong to one of the R responses. We used the same $T = 9$ time points 0.5h, 1h, 2h, 4h, 6h, 12h, 24h, 48h, 72h, as with the real data (see below). After fixing the kernel hyperparameters, to simulate the data for a gene with C conditions and P replicates we simulate one realization from f_r for each response effect, and one realization from f_p for each replicate pair and we combine them together with additive noise as shown above. This results in $C \times R \times T$ time points for each simulated gene.

We use the EQ kernels with lengthscale $l_r = 1.0$ and variance $\sigma_r^2 = 1.0$ for the response kernel k_r, and $l_p = 1.0$ for the pairing effect kernel k_p. We decided to simulate data with different levels of pairing effect variance, aiming to replicate different experimental conditions with different levels of pairing effect variance. The set of values used for the pairing effect variance is $\sigma_p^2 \in \{0.001, 0.01, 0.05, 0.1, 0.3, 0.5\}$. The random noise variance σ_ε^2 can also change depending on the experiments, thus we used the set of values {0.1, 0.2, 0.4}. For the simulated data we decided to use 3 replicates, 3 or 4 conditions and a total of 1000 genes.

The generated datasets use all the combinations of parameters values mentioned above. However, when simulating data some partitions of the conditions are *de facto* the same, for example, simulating data as {{c_1}, {c_2, c_3}} or as {{c_1, c_2}, {c_3}} is equivalent. In the simulation we only generate data only for one partitioning in each set of equivalent partitions, even though the model during the inference process can select among all the possible partitions of the condition set.

3.2. Human T-helper cell differentiation microarray data

The first data set contains gene expression time-series data from human CD4$^+$ T cells measured using microarrays originally published in

Ref. [9]. The data set is available in the Gene Expression Omnibus (GEO) repository (GEO:GSE18017). We use data from two treatments measured at time points 0.5h, 1h, 2h, 4h, 6h, 12h, 24h, 48h, 72h. Th0 condition (or treatment) corresponds to activation of naive $CD4^+$ T cells, and Th2 corresponds to activation and differentiation of naive $CD4^+$ cells towards T helper 2 (Th2) lineage. Both conditions (across all timepoints) are measured from three cell cultures that correspond to three biological replicates that are paired across the conditions. Microarray data is RMA preprocessed as in Ref. [9] and further standardized.

3.3. Mouse T-helper cell differentiation RNA-seq data

The second data set is previously unpublished and has been collected from laboratory mice, and it has a total of five treatments and six cell cultures (i.e., biological replicates). The experimental details are as in Ref. [25]. Th0 treatment corresponds to activation of naive T cells. The other four treatments are Th17, Th17+IL1b, Th17+IL21, Th17+IL1b + IL21. Th17 corresponds to activation and differentiation of naive $CD4^+$ cells towards T helper 17 (Th17) lineage. Th17+IL1b, Th17+IL21, Th17+IL1b + IL21 treatments corresponds to simultaneous activation and differentiation of naive $CD4^+$ cells towards Th17 lineage and treatment with interleukin 1 beta (IL-1β), interleukin 21 (IL-21) and combination of IL-1β and IL-21 (with concentration 20 ng/ml) (R&D Systems), respectively. Experimental data for the treatments Th0 and Th17 have been measured from the first three replicates (cell cultures), using a paired design. Experimental data for the other three treatments have been measured from the other three replicates (cell cultures), again using a paired design. Cells are sampled for gene expression analysis at nine time points: 0.5h, 1h, 2h, 4h, 6h, 12h, 24h, 48h, 72h. Sequence reads were mapped with TopHat to mouse mm9 genome as well as to Ensembl transcriptome. After the alignment, the number of reads that mapped to each gene were summarized using HTSEQ-count tool. The raw RNA-seq data used in this manuscript is available in the Gene Expression Omnibus (GEO) repository (GEO:GSE154467).

4. Results

We first tested our method on simulated time-series data with different number of treatments and a varying amount of pairing effect size. Given simulated data from three or four conditions (or treatments) with three biological replicates and the paired experimental design, our goal was to identify the correct model, i.e., the correct partitioning of the conditions. To get a comprehensive view of the model performance, we varied the correct grouping as well as the pairing effect size and the additive noise variance. Comparing our model to an otherwise equal GP model but without the pairing effect, referred here as base model, shows that modeling the pairing component improves the identification of correct partitioning for nearly all scenarios (Fig. 2). For selected partitionings with no or small pairing effect variance, the base model performs better than pairing model. This is expected for the cases of no or

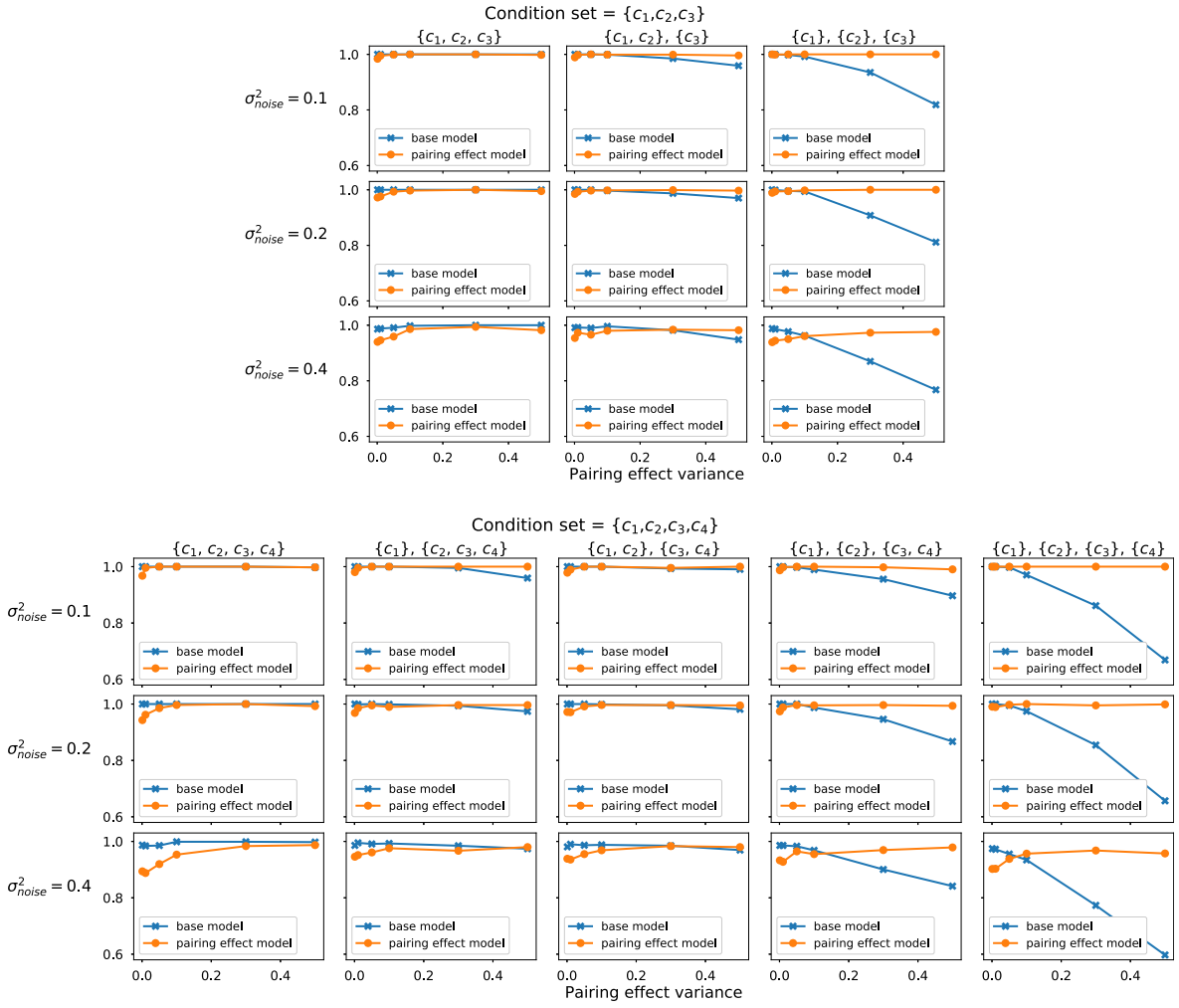

Fig. 2. Inference accuracy of the pairing effect model compared to the base model on simulated data. We consider two scenarios: (top) 3 conditions $\{c_1, c_2, c_3\}$, and (bottom) 4 conditions $\{c_1, c_2, c_3, c_4\}$. For both scenarios we simulate data from all effectively different pairings (i.e., models). We evaluate the inference accuracy for varying amounts of additive noise variance (rows) as well as for varying amounts of pairing effect variance.

small pairing effect variance as the data generation model reduces to the standard model that does not include the pairing effect. However, for all other considered cases, the pairing effect model performs at least equally well as the base model. The performance of the pairing effect model is notably better when the variance of the pairing effect is larger.

Next, we applied our method to microarray-based longitudinal gene expression data measured from activated $CD4^+$ human T cells (Th0) and cells differentiated towards T helper 2 (Th2) cell type with three paired replicates [9]. We identified genes that respond differentially between Th0 and Th2 during the first 72 h of differentiation (Table 1). When taking into account the paired design of the experiment, 30.4% of the genes were found to be differentially expressed between Th2 and Th0 cells, compared to 24.9% when only the response effect was modeled. These results indicate that the pairing effect model is better able to identify differences between conditions and also better able to partition them when the pairing effect is appropriately modeled.

We also applied our method to previously unpublished, longitudinal RNA-seq data measured from $CD4^+$ mouse T cells that were either activated or differentiating towards Th17 lineage. Experiments include six cell cultures and five different treatments: two treatments (Th0, Th17) were applied for the first three cultures and three treatments (Th17+IL1b, Th17+IL21, Th17+IL1b + IL21) for the last three cultures, resulting in two groups of three paired replicates. Our model identifies genes that have different dynamics in different subsets of the five treatments.

Table 2 summarizes how the pairing effect affects the proportion of genes detected for each partition. As reported in Table 2, modeling the pairing effect produces significantly different results compared to the base model. For both the pairing effect model and the base model, the most frequent partition is {{Th0}, {Th17, Th17+IL1b, Th17+IL21, Th17+IL1b + IL21}}. However, the frequency decreases from 47.9% for the base model to 22.8% for the pairing effect model. Another relevant partition to observe is {{Th0, Th17, Th17+IL1b, Th17+IL21, Th17+IL1b + IL21}}, which refers to the case where there is no relevant difference between conditions. For this partition the frequency increases from 9.6% for the base model to 19.4% for the pairing effect model. This indicates that the pairing effect model is not inherently biased to finding more differences between the conditions, but the pairing effect model can also report "no difference" if data supports such a conclusion. We also find significant variation for the frequency of the two partitions {{Th0}, {Th17}, {Th17+IL1b, Th17+IL21, Th17+IL1b + IL21}} and {{Th0, Th17}, {Th17+IL1b, Th17+IL21, Th17+IL1b + IL21}}. In general, the most probable partitioning of conditions for these genes is clearly affected by whether or not we model the variance of each replicate and the pairing information. An analogous situation is reported in Table 1, where the number of differentially expressed genes between the two conditions Th0 and Th2 is 24.9% for the base model compared to 30.4% for the pairing effect model.

We report in Fig. 3 the visualization of the pairing effect model and base model fit on the gene *Fasl* from the $CD4^+$ mouse T cells dataset. The base model identifies two partitions (Fig. 3a), whereas the pairing effect model identifies three partitions (Fig. 3b). Furthermore, the estimated replicates effects are not equal (Fig. 3c) and explain variance that is non-negligible when compared to the variance of the response model in Fig. 3b. The two groups of conditions {Th17} and {Th17+IL1b, Th17+IL21, Th17+IL1b + IL21} are merged together in the pairing effect model and the variation between the two groups is now explained by the pairing effect component. Also, for condition Th0 the variance introduced by replicate 1 is now part of the pairing effect component and can be separated from the response effect, providing a more clean representation of the Th0 condition.

Overall, the results reported here support the introduction of a model component to model the pairing effect, as we do in our pairing effect model. Even though one does not typically have the true classification labels for this type of dataset, it is clear from the results how this model component can affect significantly the outcome of the analysis. At first,

Table 1
T-helper cell gene expression dataset [9] modeling results. Results obtained by fitting the base model and the pairing effect model on the human T-helper cell gene expression dataset [9]. The percentage of the total amount of genes is reported for each partition.

Partition	Base	Pairing
{Th0, Th2}	75.1	69.6
{Th0}, {Th2}	24.9	30.4

Table 2
Mouse T-helper cell RNA-seq dataset modeling results. Results obtained by fitting the base model and the pairing effect model on the mouse T-helper cell RNA-seq dataset. The percentage of the total amount of genes is reported. The values are reported in descending order according to the pairing effect model results.

Partition	Base	Pairing
{Th0}, {Th17, Th17+IL1b, Th17+IL21, Th17+IL1b + IL21}	47.9	22.8
{Th0, Th17, Th17+IL1b, Th17+IL21, Th17+IL1b + IL21}	9.6	19.4
{Th17}, {Th0, Th17+IL1b, Th17+IL21, Th17+IL1b + IL21}	0.5	6.6
{Th0}, {Th17}, {Th17+IL1b, Th17+IL21, Th17+IL1b + IL21}	19.7	5.6
{Th0, Th17}, {Th17+IL1b, Th17+IL21, Th17+IL1b + IL21}	17.4	3.9
{Th0}, {Th17, Th17+IL21}, {Th17+IL1b, Th17+IL1b + IL21}	2.3	3.8
others	2.6	37.9

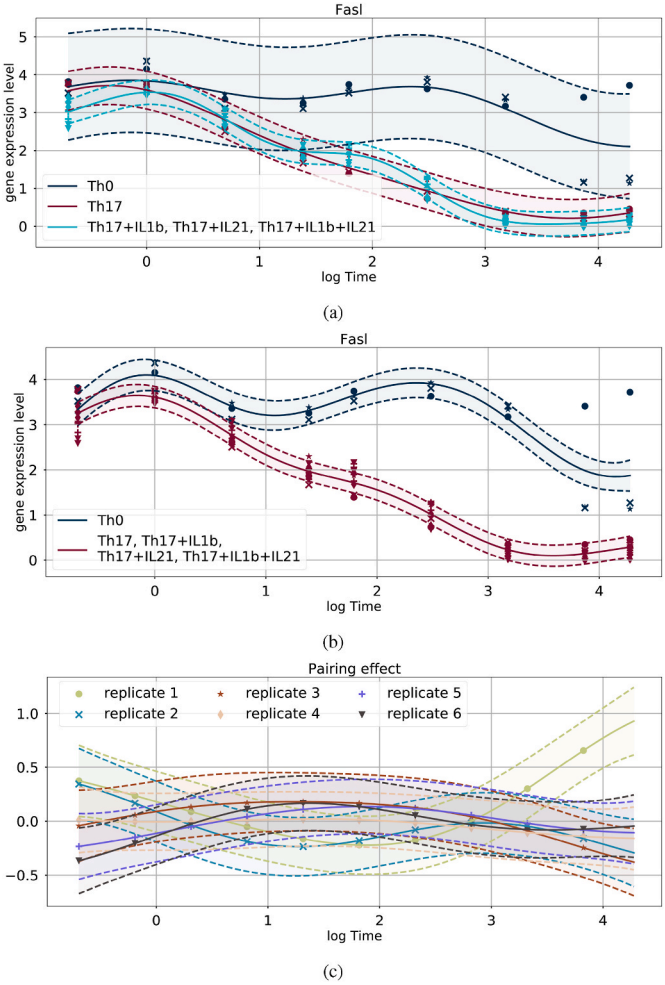

Fig. 3. The result of the pairing effect model on the gene *Fasl*. (a) the base model fit, (b) the response effects from the pairing effect model for the selected groups of conditions and (c) the relative pairing effect.

Table 3
Summary of relevant previous GP methods for gene expression time series analysis.

Authors	Year	Method name	Features of the method
Kalaitzis *at al.* [16]	2011	GP regression	One of first GP methods for differential expression analysis of gene expression time series data.
Stegle *at al.* [22]	2010	Robust GP regression	A robust GP regression method to compare gene expression time-series from two samples and to identify time interval of differential expression.
Äijö *at al.* [2]	2012	LIGAP	This method generalizes GP regression framework to any number of conditions and offers genome-wide grouping and ranking functionalities.
Hensman *at al.* [15]	2013	GP clustering	A hierarchical bayesian clustering of gene expression time series data.
Äijö *at al.* [1]	2014	DyNB	A GP regression method for gene expression time series data that accounts for time delays between biological replicates and non-Gaussian likelihood models.
Quintana *at al.* [17]	2016	Bayesian GP regression	A generalization of the standard mixed models for longitudinal data using GPs.
Cheng *at al.* [8]	2019	LonGP	An additive GP regression model for longitudinal data analysis supporting non-stationary signals as well as Markov chain Monte Carlo (MCMC) sampling and approximate inference using central composite design (CCD).
Timonen *at al.* [24]	2021	lgpr	An additive GP regression model for longitudinal data analysis that additionally accounts for uncertainty in the disease effect time, the disease heterogeneity and arbitrary likelihood models, as well as uses the dynamic Hamiltonian Monte Carlo sampler.
This work	2022	PairGP	A GP model for longitudinal data with paired multi-condition designs that accounts for non-stationarity and pairing of replicas across conditions, and generalizes to any number of conditions and replicas.

we showed how the pairing effect model outperform the base model on simulated data. The results show that the higher is the complexity of the data, the higher is the advantage in using the pairing effect model. Then, we showed that we can obtain more relevant and interpretable results also on two real-world gene expression time series datasets. For data sets generated by paired experimental design, the pairing effect model provides a more realistic model of the data through the use of two components, the response effect and the pairing effect, and produce a better explanation of the data.

5. Discussion

As explained in Ref. [16], existing methods proposed in literature to model time-series data often miss to consider the strong temporal correlation or the non-stationarity of the process. They are also typically not suited for modeling short time series [10], resulting in overfitting. On the other hand, GP regression models naturally include time dependencies, can model non-linear effects through kernels, and explicitly model noise [19]. In addition, GP regression models can deal well with short time series and can be extended to consider the non-stationarity of the process [14]. Overall, GP based approaches have been proven to be solid alternatives to more traditional statistical analysis.

We compared here our new method to the baseline method that fits standard GP regression models but do not consider the pairing effect [2,16]. The results on both simulated and real-world T-helper cell datasets show the benefit of introducing a pairing effect component for both accuracy and explainability of the model. Several different extensions have been proposed to GP modeling for longitudinal data [8,17,24] that also account for the pairing effect considered here. However, all these longitudinal GP extensions are implemented with MCMC sampling techniques that make them less computationally efficient and, in practice, impractical for genome-wide analysis. Whereas, our proposed method implements an efficient analysis that also scales better to genome-wide analysis as well. In Table 3, we report a more complete comparison between this work and several other GP based methods for gene expression time series analysis.

6. Conclusions

We have implemented a GP-based model for analysis of longitudinal gene expression data that accounts for paired multi-condition study designs. Results demonstrate that our model improves the detection of correct partitioning of different conditions. Utilizing Gaussian process regression along with time-warping techniques, to take into account the non-stationarity of the process, and a specific kernel combination, to model the pairing effect as an additional component, has proven to be a successful approach of modeling gene expression time series data. We showed how the pairing effect model produces more accurate results on both simulated data and real T-helper cell datasets with different number of conditions. Even though ground truth labels are not available for the analysed real datasets, the pairing effect model is capable of selecting biologically relevant partitions of the condition set. Compared to the base model that represents that standard applications GP models to gene expression datasets, the pairing effect model takes advantage of the additional model component and produces a better and more interpretable fit of the data. The additional interpretability that the pairing effect model provides benefits the overall data analysis and the understanding of the underlying biological process.

Funding

This work was supported by the Academy of Finland [grant numbers 292 660, 313 271]

Availability of data and materials

The human T-helper cell differentiation data analysed during the current study was originally published in Ref. [9], and are available in the Gene Expression Omnibus (GEO) repository (GEO:GSE18017).

The mouse T-helper cell differentiation data used in this manuscript is available in the Gene Expression Omnibus (GEO) repository (GEO: GSE154467).

Ethics approval

All mice were bred in the animal facility of the Medical Research Council National Institute for Medical Research, currently part of The Francis Crick Institute, London, UK. Mice were kept under specific pathogen–free conditions. All animal experiments were approved by the local Ethical Review panel at the National Institute for Medical Research in accordance with the Institutional Committees on Animal Welfare of the UK Home Office (the Home Office Animals Scientific Procedures Act, 1986).

CRediT authorship contribution statement

Michele Vantini: Design of the algorithm, Implementation of the software package, Writing of the manuscript. **Henrik Mannerström:** Design of the algorithm, Writing of the manuscript. **Sini Rautio:** Design of the algorithm, Writing of the manuscript. **Helena Ahlfors:** Writing of the manuscript, Data collection. **Brigitta Stockinger:** Data collection. **Harri Lähdesmäki:** Design of the algorithm, Writing of the manuscript.

Declaration of competing interest

None Declared.

Abbreviations

GP	Gaussian processes
EQ	Exponentiated quadratic

References

[1] Tarmo Äijö, Butty Vincent, Zhi Chen, Verna Salo, Subhash Tripathi, Christopher B. Burge, Riitta Lahesmaa, Harri Lähdesmäki, Methods for time series analysis of rna-seq data with application to human th17 cell differentiation, Bioinformatics 30 (12) (2014) i113–i120.

[2] Tarmo Äijö, Sanna M. Edelman, Tapio Lönnberg, Antti Larjo, Henna Kallionpää, Soile Tuomela, Emilia Engström, Riitta Lahesmaa, Harri Lähdesmäki, An integrative computational systems biology approach identifies differentially regulated dynamic transcriptome signatures which drive the initiation of human t helper cell differentiation, BMC Genom. 13 (1) (2012) 572.

[3] Simon Anders and Wolfgang Huber, Differential expression analysis for sequence count data, Genome Biol. 11 (10) (2010) R106.

[4] Claudia Angelini, Luisa Cutillo, Daniela De Canditiis, Margherita Mutarelli, Marianna Pensky, Bats: a bayesian user-friendly software for analyzing time series microarray experiments, BMC Bioinf. 9 (1) (2008) 415.

[5] Claudia Angelini, Daniela De Canditiis, Margherita Mutarelli, Marianna Pensky, A bayesian approach to estimation and testing in time-course microarray experiments, Stat. Appl. Genet. Mol. Biol. 6 (1) (2007).

[6] Bates Douglas, Martin Mächler, Bolker Ben, Steve Walker, Fitting linear mixed-effects models using lme4, J. Stat. Software 67 (1) (2015) 1–48.

[7] Richard H. Byrd, Peihuang Lu, Jorge Nocedal, Ciyou Zhu, A limited memory algorithm for bound constrained optimization, SIAM J. Sci. Comput. 16 (5) (1995) 1190–1208.

[8] Lu Cheng, Siddharth Ramchandran, Tommi Vatanen, Niina Lietzen, Riitta Lahesmaa, Vehtari Aki, Harri Lähdesmäki, An additive Gaussian process regression model for interpretable non-parametric analysis of longitudinal data, Nat. Commun. 10 (1798) 2019.

[9] Laura L. Elo, Henna Järvenpää, Soile Tuomela, Sunil Raghav, Helena Ahlfors, Kirsti Laurila, Bhawna Gupta, Riikka J. Lund, Johanna Tahvanainen, R David Hawkins, et al., Genome-wide profiling of interleukin-4 and stat6 transcription factor regulation of human th2 cell programming, Immunity 32 (6) (2010) 852–862.

[10] Jason Ernst, J Nau Gerard, Ziv Bar-Joseph, Clustering short time series gene expression data, Bioinformatics 21 (suppl_1) (2005) i159–i168.

[11] David S. Fischer, Fabian J. Theis, Nir Yosef, Impulse model-based differential expression analysis of time course sequencing data, Nucleic Acids Res. 46 (20) (2018) e119–e119.

[12] GPy GPy, A Gaussian process framework in python, 2012. http://github.com/SheffieldML/GPy. (Accessed 4 November 2021). since.

[13] Markus Heinonen, Olivier Guipaud, Fabien Milliat, Valérie Buard, Béatrice Micheau, Georges Tarlet, Marc Benderitter, Farida Zehraoui, Florence d'Alché Buc, Detecting time periods of differential gene expression using Gaussian processes: an application to endothelial cells exposed to radiotherapy dose fraction, Bioinformatics 31 (5) (2014) 728–735.

[14] Markus Heinonen, Henrik Mannerström, Juho Rousu, Samuel Kaski, Harri Lähdesmäki, Non-stationary Gaussian process regression with Hamiltonian Monte Carlo, in: Artificial Intelligence and Statistics, PMLR, 2016, pp. 732–740.

[15] Hensman James, Neil D. Lawrence, Magnus Rattray, Hierarchical bayesian modelling of gene expression time series across irregularly sampled replicates and clusters, BMC Bioinf. 14 (1) (2013) 252.

[16] Alfredo A. Kalaitzis, Neil D. Lawrence, A simple approach to ranking differentially expressed gene expression time courses through Gaussian process regression, BMC Bioinf. 12 (1) (2011) 1–13.

[17] Fernando A. Quintana, Wesley O. Johnson, L Elaine Waetjen, B. Ellen, Gold. Bayesian nonparametric longitudinal data analysis, J. Am. Stat. Assoc. 111 (515) (2016) 1168–1181.

[18] Matthew E. Ritchie, Belinda Phipson, D.I. Wu, Yifang Hu, Charity W. Law, Wei Shi, Gordon K. Smyth, Limma powers differential expression analyses for rna-sequencing and microarray studies, Nucleic Acids Res. 43 (7) (2015) e47–e47.

[19] Stephen Roberts, Michael Osborne, Mark Ebden, Steven Reece, Neale Gibson, Suzanne Aigrain, Gaussian processes for time-series modelling, Phil. Trans. Math. Phys. Eng. Sci. 371 (1984) (2013) 20110550.

[20] Mark D. Robinson, Davis J. McCarthy, Gordon K. Smyth, edger: a bioconductor package for differential expression analysis of digital gene expression data, Bioinformatics 26 (1) (2010) 139–140.

[21] Daniel Spies, Peter F. Renz, Tobias A. Beyer, Constance Ciaudo, Comparative analysis of differential gene expression tools for rna sequencing time course data, Briefings Bioinf. 20 (1) (2017) 288–298.

[22] Stegle Oliver, J Denby Katherine, Emma J. Cooke, David L. Wild, Zoubin Ghahramani, Karsten M. Borgwardt, A robust bayesian two-sample test for detecting intervals of differential gene expression in microarray time series, J. Comput. Biol. 17 (3) (2010) 355–367.

[23] Jasmin Straube, Alain-Dominique Gorse, Proof Centre of Excellence Team, Bevan Emma Huang, Kim-Anh Lê Cao, A linear mixed model spline framework for analysing time course 'omics' data, PLoS One 10 (8) (2015), e0134540.

[24] Juho Timonen, Henrik Mannerström, Vehtari Aki, Harri Lähdesmäki, An Interpretable Probabilistic Machine Learning Method for Heterogeneous Longitudinal Studies, vol. 37, 2021, pp. 1860–1867, 13.

[25] Soile Tuomela, Sini Rautio, Helena Ahlfors, Viveka Öling, Verna Salo, Ubaid Ullah, Zhi Chen, Saara Hämälistö, Subhash K. Tripathi, Tarmo Äijö, et al., Comparative analysis of human and mouse transcriptomes of th17 cell priming, Oncotarget 7 (12) (2016) 13416.

[26] Ming Yuan, Flexible temporal expression profile modelling using the Gaussian process, Comput. Stat. Data Anal. 51 (3) (2006) 1754–1764.