Analysing Gene Expression Data Using Gaussian Processes

Lorenz Wernisch

School of Crystallography

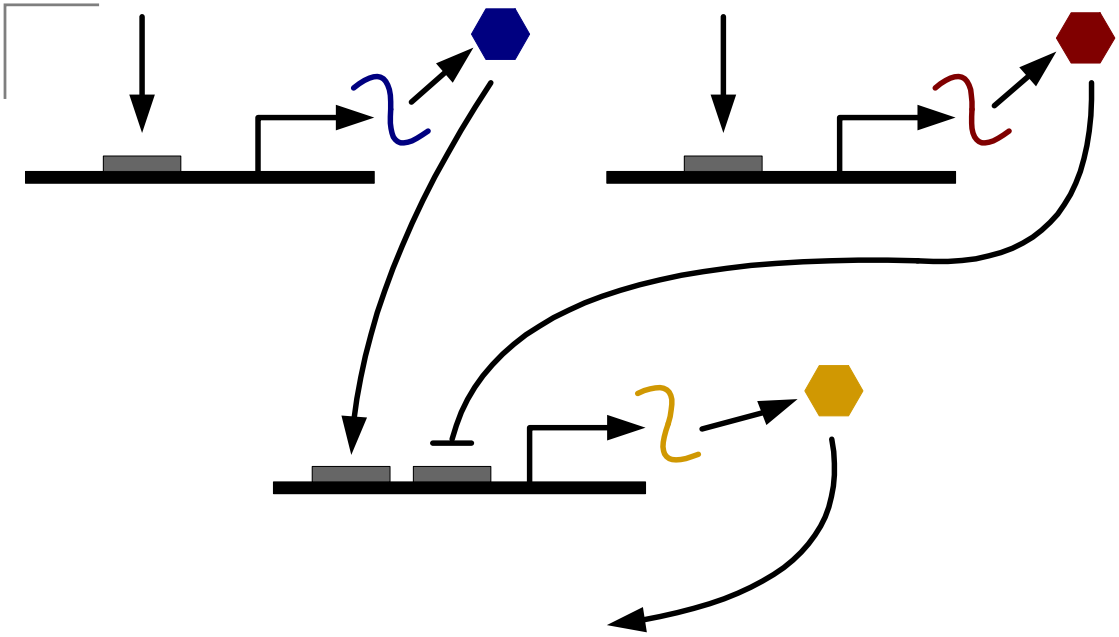
Birkbeck College

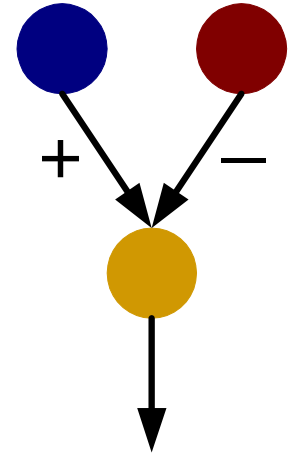
London

Overview

- Gene regulatory networks, microarrays
- Time-series analysis by linear regression
- Bayesian inference, Occam's razor
- Extension to nonlinear models
- Gaussian processes
- Applications
- Filtering with Gaussian processes

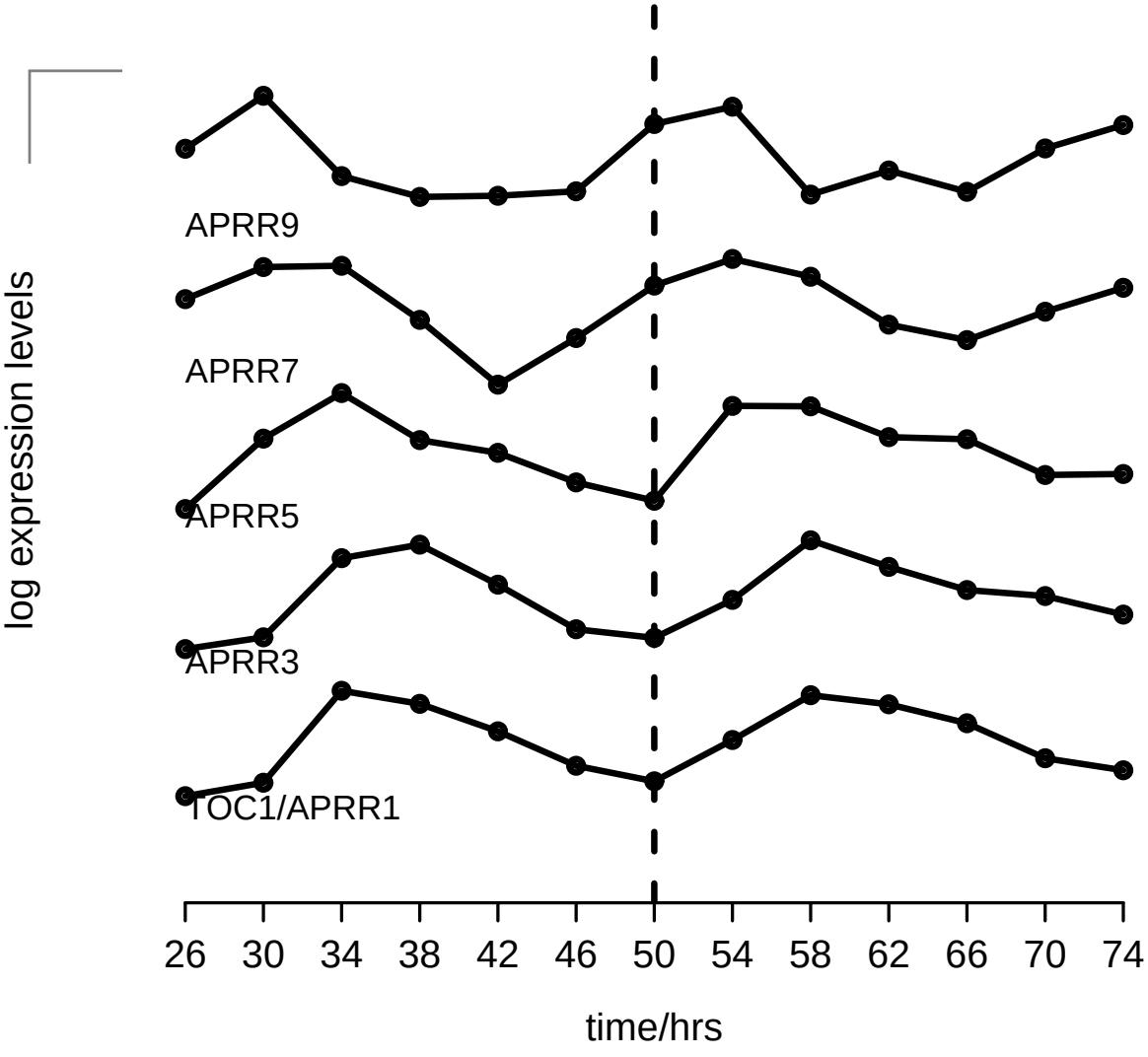
Gene Regulatory Networks





- Gene expression levels depend on external stimuli and activity of genes (transcription factors)
- **Microarrays** measure the mRNA levels of genes
- Construction of gene networks from microarray data

A. thaliana: APRR family



Time-series of *A. thaliana*

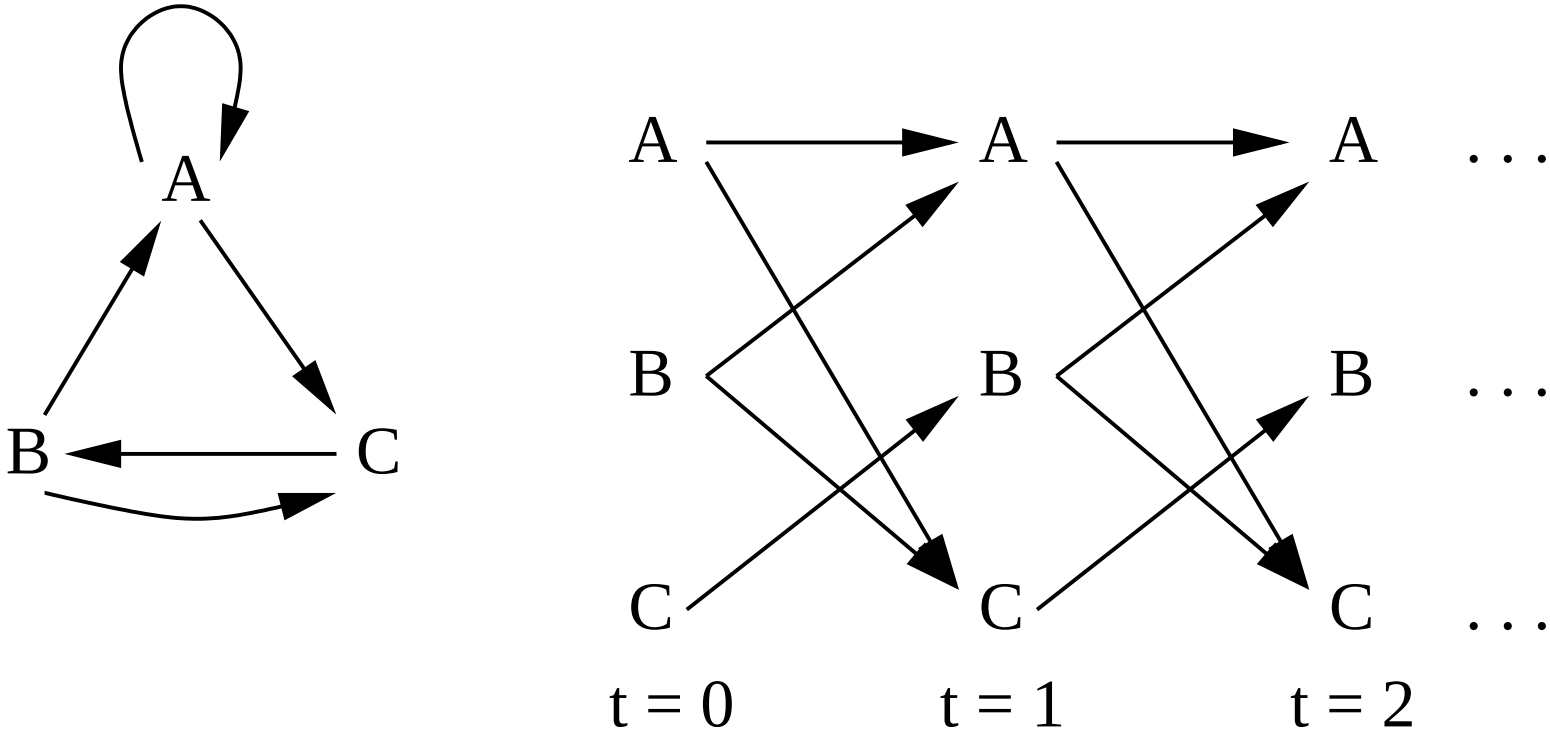
Constant light

13 time points every 4 hours from 26 to 74 hrs

Data by **Kieron Edwards** and **Andrew Millar**

APRR family, possible modulators for light sensitivity of main circadian clock series

Networks from time-series data



Static graph representing dependencies between genes has cycles

Cycles unrolled in time: **acyclic** graph

Network topology repeated over time slices

Linear time-series model

$$x_t = \Phi x_{t-1} + \mu + w_t$$

x_t is N-vector of RNA levels at time t (of N genes)

w_t is N-vector of **biological** noise added at t

μ is N-vector of constant trend, ie constitutive expression

If there is **no constant trend**, $\mu = 0$, Φ can be estimated by standard regression:

$$\Phi' = (X_{t-1}X'_{t-1})^{-1}X_{t-1}X'_t$$

where X_t and X_{t-1} are $N \times (T-1)$ matrices with time vectors $x_2, \ldots, x_T$ and $x_1, \ldots, x_{T-1}$ as **columns**

Estimating matrix for *APPR* family

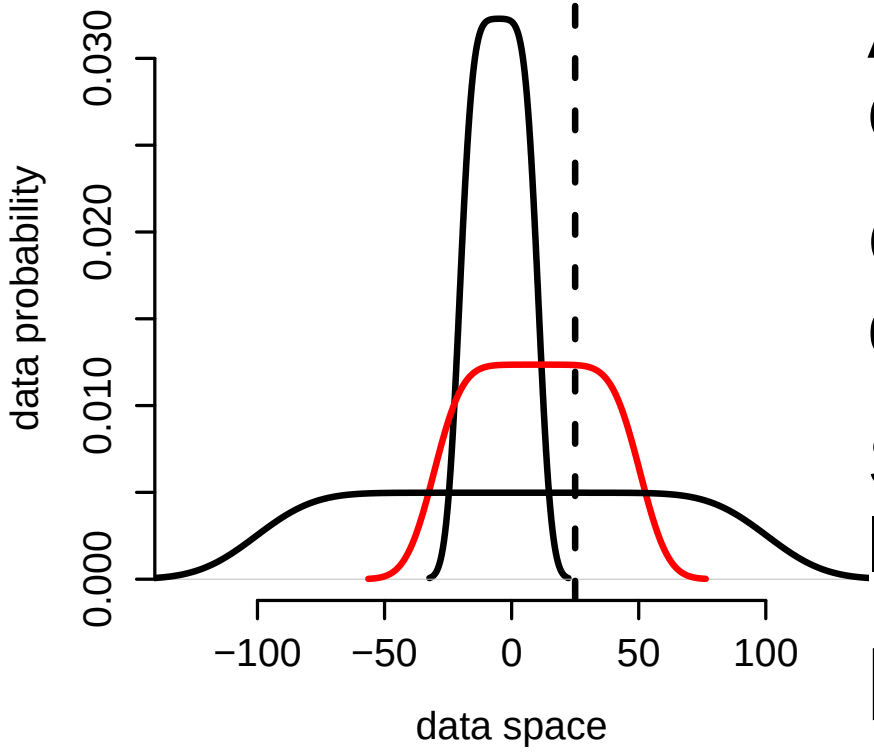
Estimation by standard (least squares) regression:

	APRR9	APRR7	APRR5	APRR3	TOC1
APRR9	-0.59	-0.06	0.78	0.39	0.48
APRR7	0.56	0.35	0.34	0.29	0.21
APRR5	-0.80	0.15	-0.26	0.46	0.43
APRR3	-0.34	-0.94	-0.12	-0.13	0.05
TOC1	-0.11	-0.05	0.66	0.46	0.30

Problem: each gene connected to each other

One could test for significance of nonzero parameters: problems of significance tests, significance levels, multiple testing, . . .

Bayesian models are simple



Automatic complexity control, **Occam's razor**:

Complex model covers many data sets: small probability each

Simple model few data sets: large probability each

[MacKay, Neal]

Automatic relevance determination: assume Gaussian distribution for each matrix entry a_{ij} with variances σ_{ij}^2 as free parameters, **integrate out** a_{ij} and **maximize** $P(D \mid \text{model}, \{\sigma_{ij}^2\})$ [RVMs Tipping]

Linear regression framework

$$t = \Phi w + \epsilon$$

Probability of data, given parameters (**likelihood**):

$$p(t \mid w, \sigma^2) = \frac{1}{(2\pi)^{N/2}\sigma^N} \exp(-\frac{|t - \Phi w|^2}{2\sigma^2})$$

Gaussian **prior** on coefficients (weights) w:

$$p(w \mid \alpha) = \frac{1}{(2\pi)^{-M/2}} \prod_{m=1}^{M} \alpha_m^{1/2} \exp(-\frac{\alpha_m w_m^2}{2})$$

α_m is the **precision** (the inverse variance $1/\sigma_m^2$)

Maximum likelihood type II

Integrating out w:

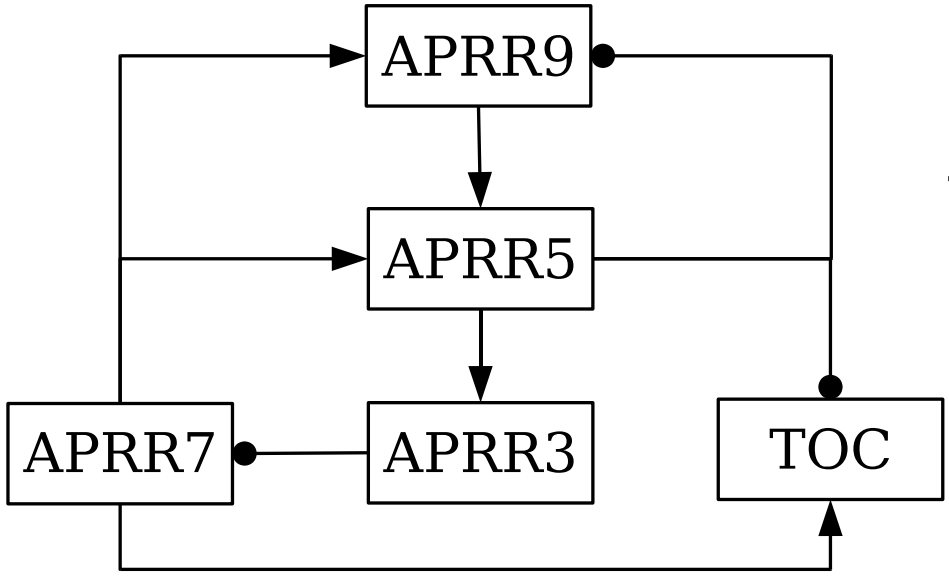
$$p(t \mid \alpha, \sigma^2) = \frac{1}{(2\pi)^{N/2}|C|^{1/2}} \exp(-\frac{1}{2} t' C^{-1} t)$$

$$C = \sigma^2 I + \Phi A^{-1} \Phi'$$

- Maximum likelihood estimation of hyperparameters α by maximizing $p(t \mid \alpha, \sigma^2)$ (type II ML) brings Occam's razor to bear
- Tipping et al. suggest analytical solutions for iterative optimization, optimizing for α_i in turn
- Maximization, eg, by conjugate gradients seems to be at least as efficient

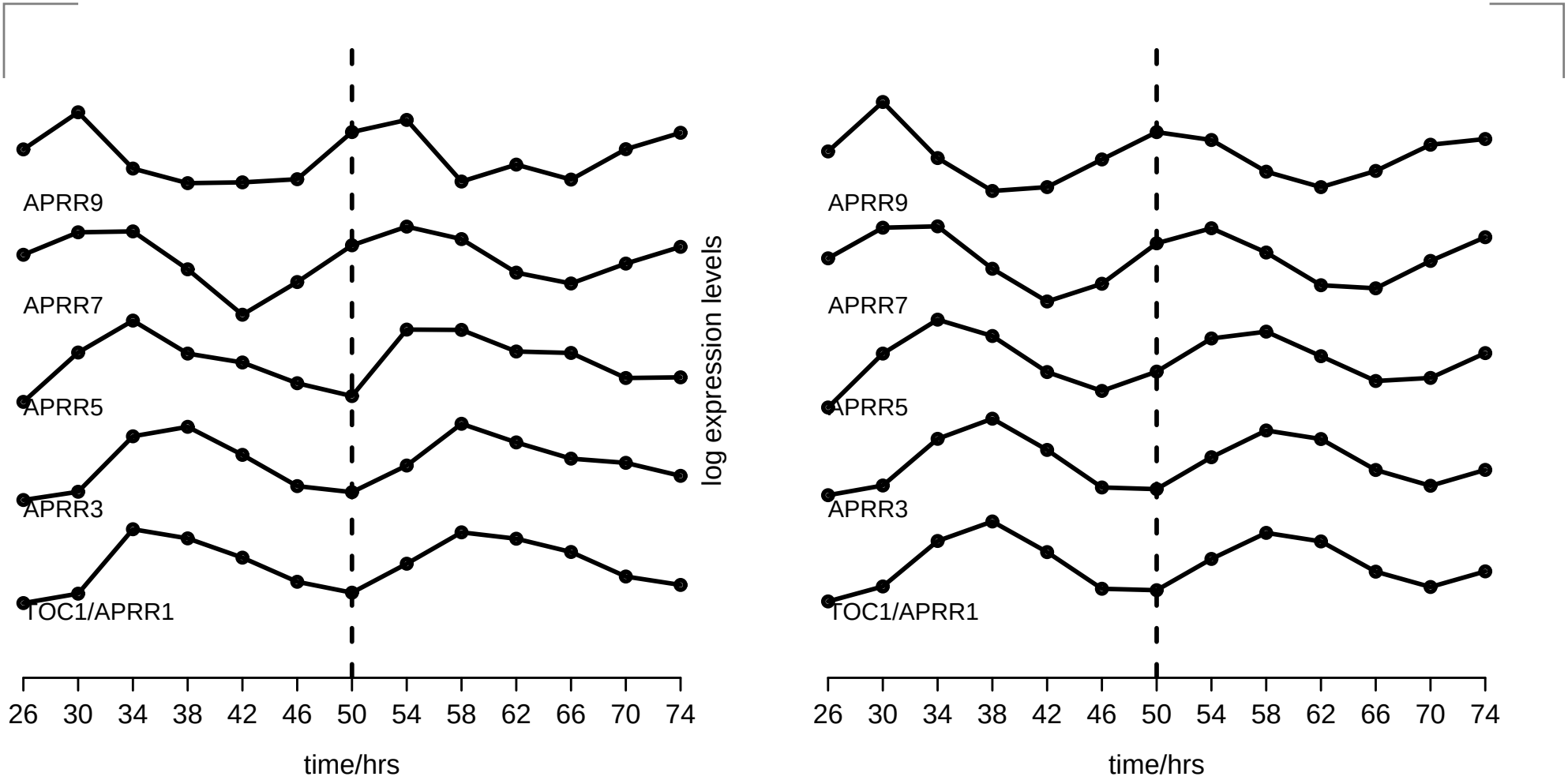
Sparse Bayesian estimates for APRR net

	APRR9	APRR7	APRR5	APRR3	TOC1
APRR9	-0.11	0.27	-0.90	-0.01	0
APRR7	0.00	0.28	0.00	-0.80	0
APRR5	0.28	0.39	0.00	0.00	0
APRR3	0.00	0.41	0.59	0.00	0
TOC1	0.00	0.37	0.52	0.00	0



Far fewer nonzero entries than in standard regression!

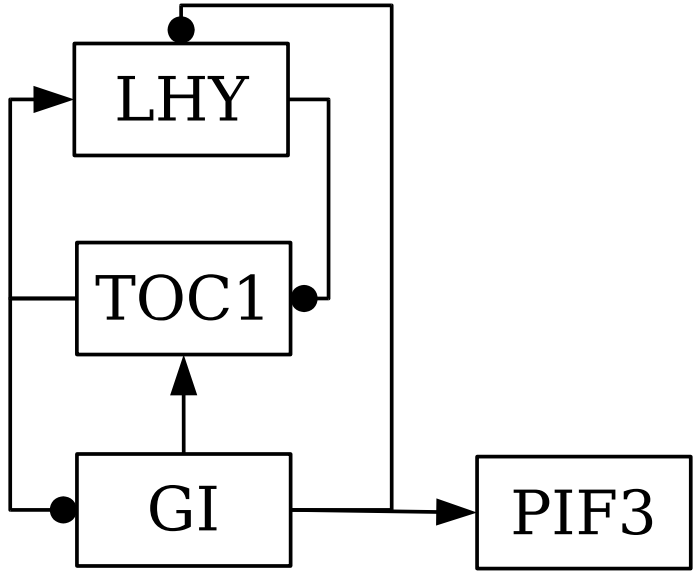
Reconstruction of APRR traces



Start estimated dynamics on initial conditions with 0 process noise: good agreement

Sparse Bayesian estimates for LHY/TOC1 net

	LHY	TOC1	GI	PIF3
LHY	0.66	0.80	-0.78	0.00
TOC1	-0.34	-0.19	0.58	-0.10
GI	0.00	-0.87	0.65	0.00
PIF3	0.00	0.00	0.22	-0.14

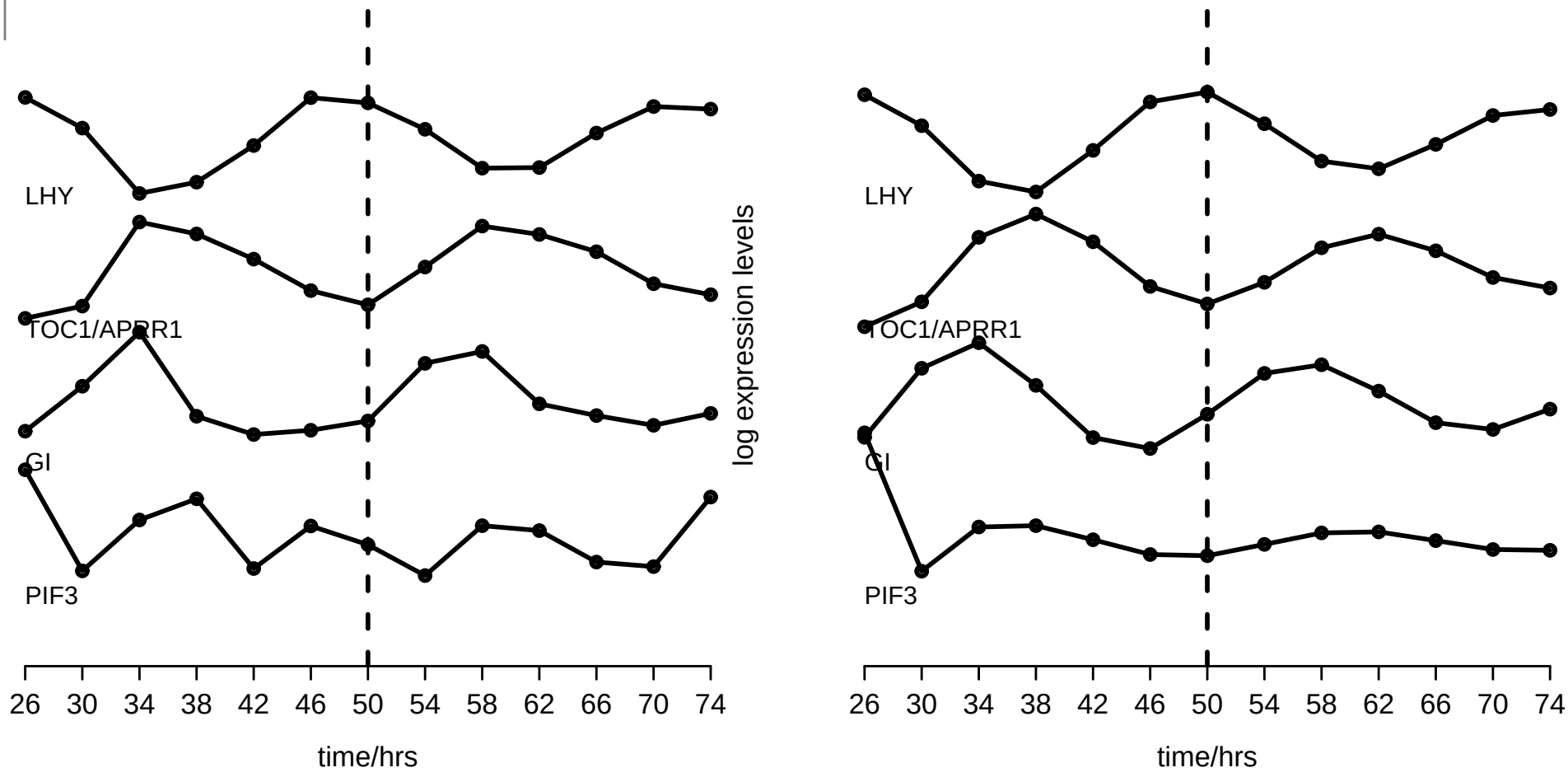


LHY in negative feedback with TOC1

Second negative feedback loop involving GI

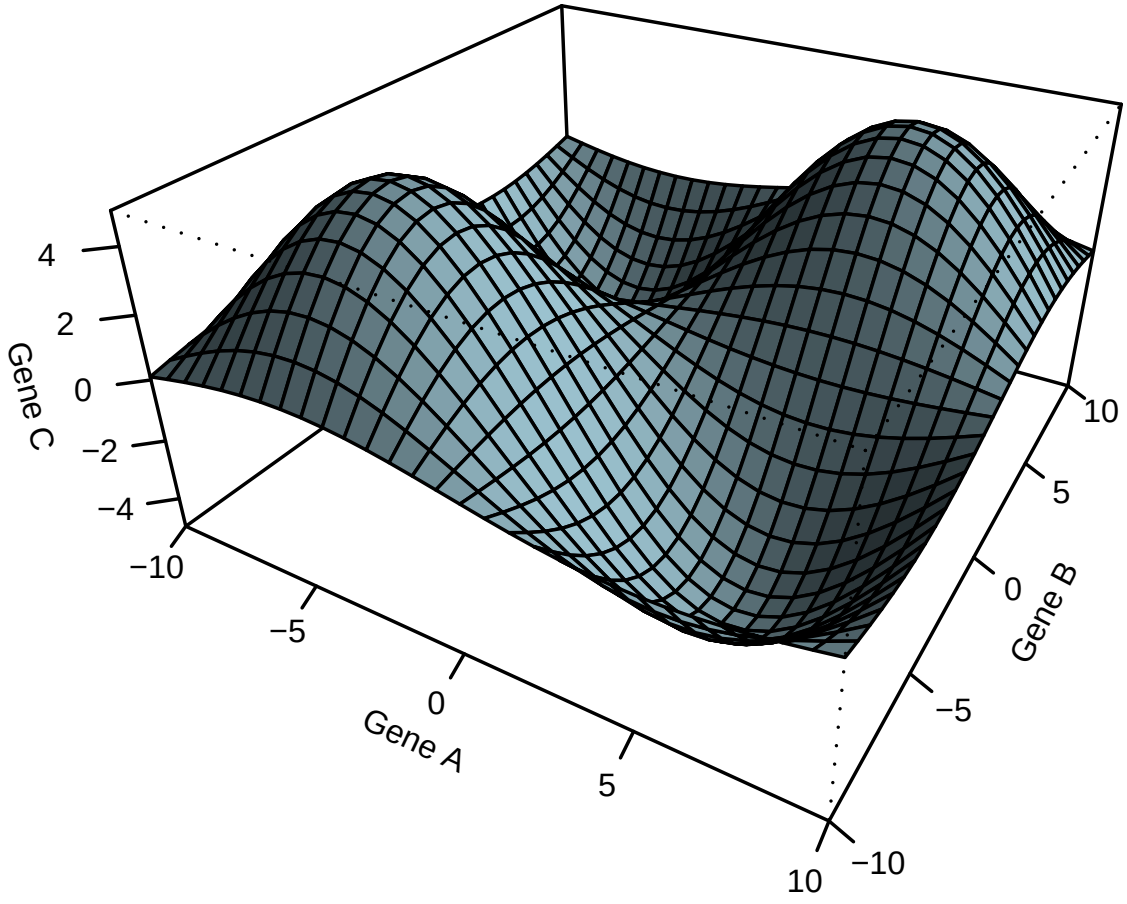
PIF3 just added for good measure

Reconstruction of LHY/TOC1 traces



Start estimated dynamics on initial conditions with 0 process noise

Nonlinear dependencies



Assumed **linear** depencies of level of gene A on other gene levels

Genes often operate as switches and complex gates with nonlinear interactions (eg exclusive or)

Need to go beyond linear models:
Gaussian processes (GP)

Gaussian process

- Input values d-dimensional $x = (x_1, \ldots, x_N)$, $x_i \in \mathbb{R}^d$
- Target values $t = (t_1, \ldots, t_N)$, $t_i \in \mathbb{R}$
- Joint distribution of the output t is multivariate Gaussian $N(0, K)$
- Covariance matrix K

$$K_{pq} = \beta_0 + C_L(x_p, x_q) + C_G(x_p, x_q) + \sigma_\epsilon^2 I(p = q)$$

β_0 overall constant
σ_ϵ^2 noise term along diagonal of K
$I()$ indicator function

Covariance components

Linear covariance part

$$C_L(x_p, x_q) = x_p' B^{-1} x_q$$

with **linear relevance parameters**
$B = \text{diag}(\beta_1, \ldots, \beta_d)$

Squared exponential (Gaussian) covariance part

$$C_G(x_p, x_q) = \alpha_0 \exp(-\frac{1}{2}(x_p - x_q)' A^{-1}(x_p - x_q))$$

with **nonlinear relevance parameters**
$A = \text{diag}(\alpha_1, \ldots, \alpha_d)$ and scale parameter α_0

Compare with linear regression

Compare linear covariance part with noise:

$$C_L(x_p, x_q) = x_p' B^{-1} x_q + \sigma_\epsilon^2 I$$

with the covariance matrix of a linear regression with weights integrated out (see above):

$$C = \Phi A^{-1} \Phi' + \sigma_\epsilon^2 I$$

This is the same if

$$B = \mathrm{diag}(\alpha_1, \ldots, \alpha_p) = \mathrm{diag}(1/\sigma_1^2, \ldots, 1/\sigma_p^2)$$

and the rows of Φ are the input vectors x_i

Training of GP

Covariance parameters θ_{MAP} maximizing posterior probability:

$$P(\theta \mid t, x) \propto P(t \mid x, \theta) P(\theta)$$

with

$$\log P(t \mid x, \theta) = -\frac{1}{2}\big(t' K(x,\theta) t - \log |K(x,\theta)| - n \log 2\pi\big)$$

Lognormal prior $P(\theta)$ with fixed a and b

$$\log P(\theta) = N(\theta \mid a, b)$$

Optimization with conjugate gradients (using derivatives)

Conditional mean and variance

New input point x^*:

$$\tilde{K} = \begin{pmatrix} K & k(x^*) \\ k(x^*)' & k(x^*, x^*) \end{pmatrix}$$

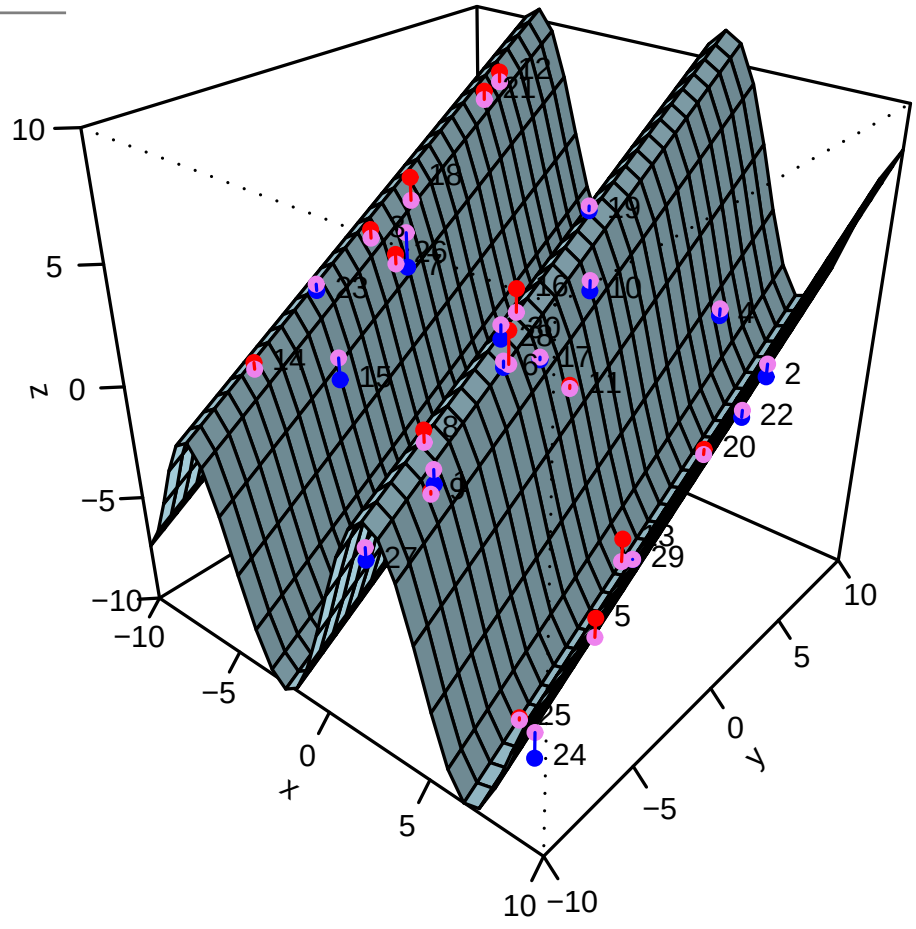
where

$$k(x^*) = (\beta_0 + C_L(x^*, x_q) + C_G(x^*, x_q))_{q=1}^N$$
$$k(x^*, x^*) = \beta_0 + x^{*\prime} B^{-1} x^* + \alpha_0 + \sigma_\epsilon^2$$

$f(x^*)$ is Gaussian $N(\mu(x^*), \sigma^2(x^*))$

$$\mu(x^*) = k(x^*)' K^{-1} t$$
$$\sigma^2(x^*) = k(x^*, x^*) - k(x^*)' K^{-1} k(x^*)$$

GP on simulated static data



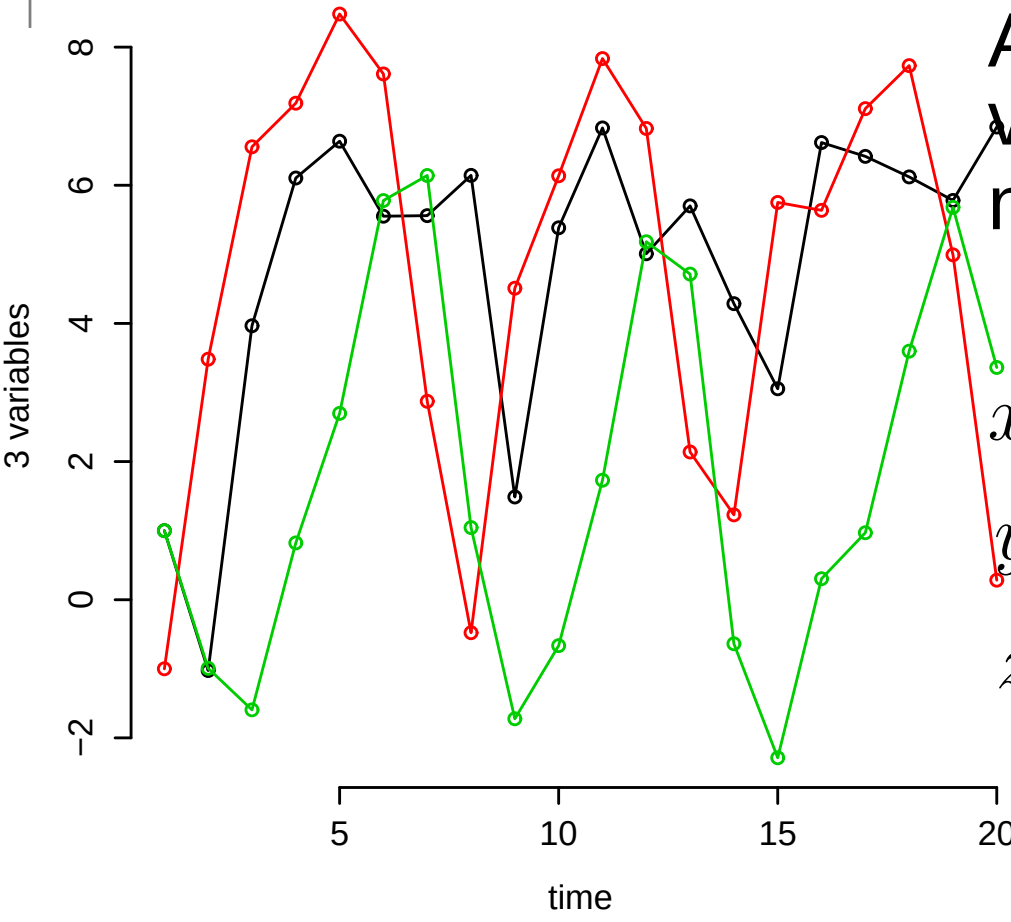
Relevance parameters:

	x_1	x_2	x_3
nonlinear	0.21	0	0
linear	0	0.35	0

estimated sd 0.92

30 data points with $f(x_1, x_2, x_3) = 5\sin(0.7x_1) + 0.5x_2 + \epsilon$ where $\epsilon \sim N(0, 1)$

GP on simulated time-series data



Artificial network of 3 variables connected by nonlinear relationships

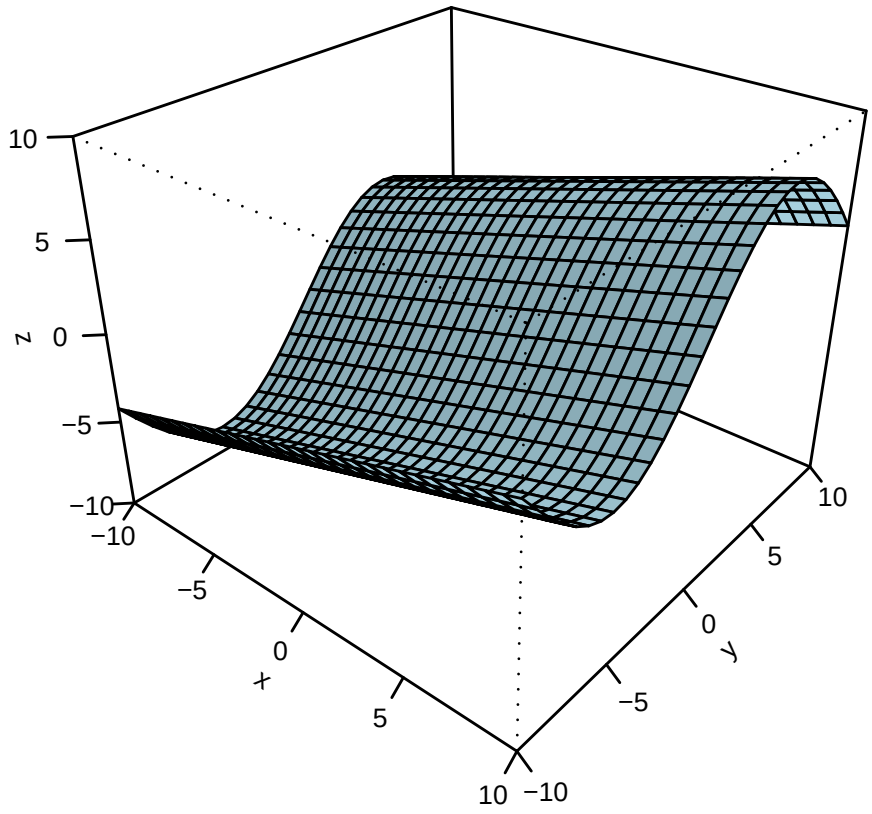
$$x_{t+1} = 0.35x_t + 5\sin(0.3y_t) + \epsilon$$

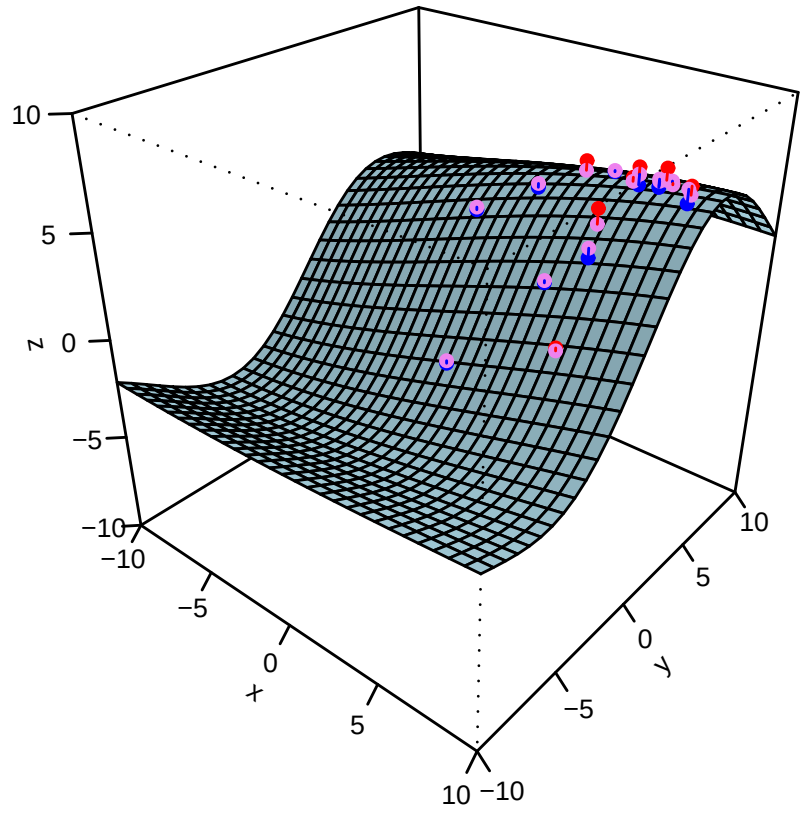
$$y_{t+1} = 0.4y_t + 5\cos(0.3z_t) + \epsilon_2$$

$$z_{t+1} = 0.4z_t + 0.1y_t^2 - 2 + \epsilon_3$$

Stable cycling easy to achieve with nonlinear networks

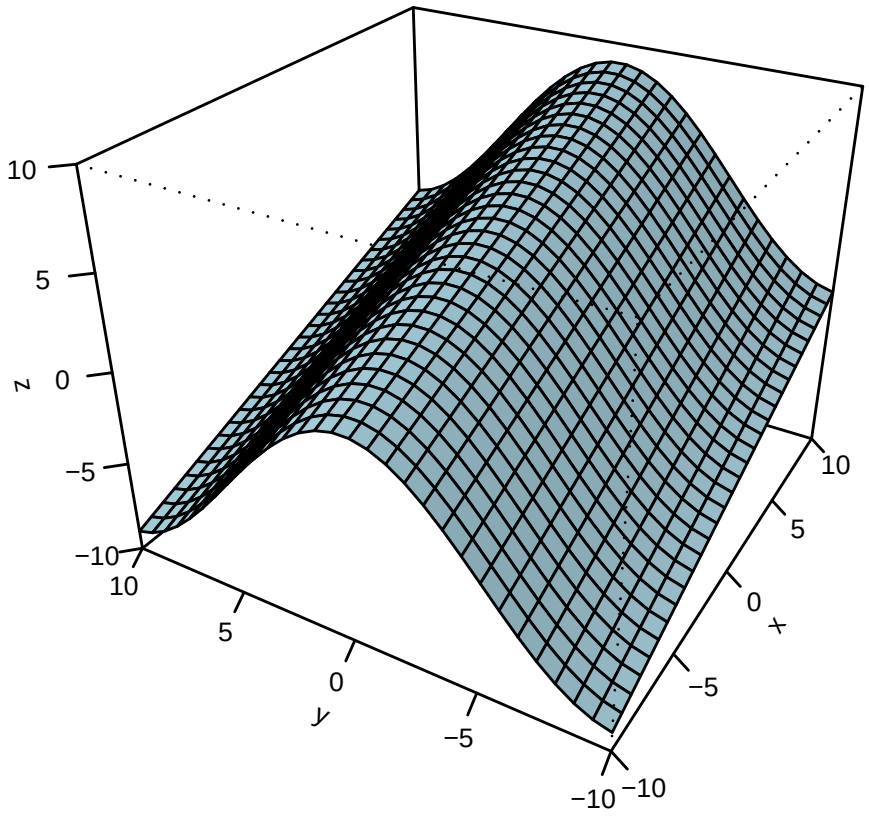
GP on time-series

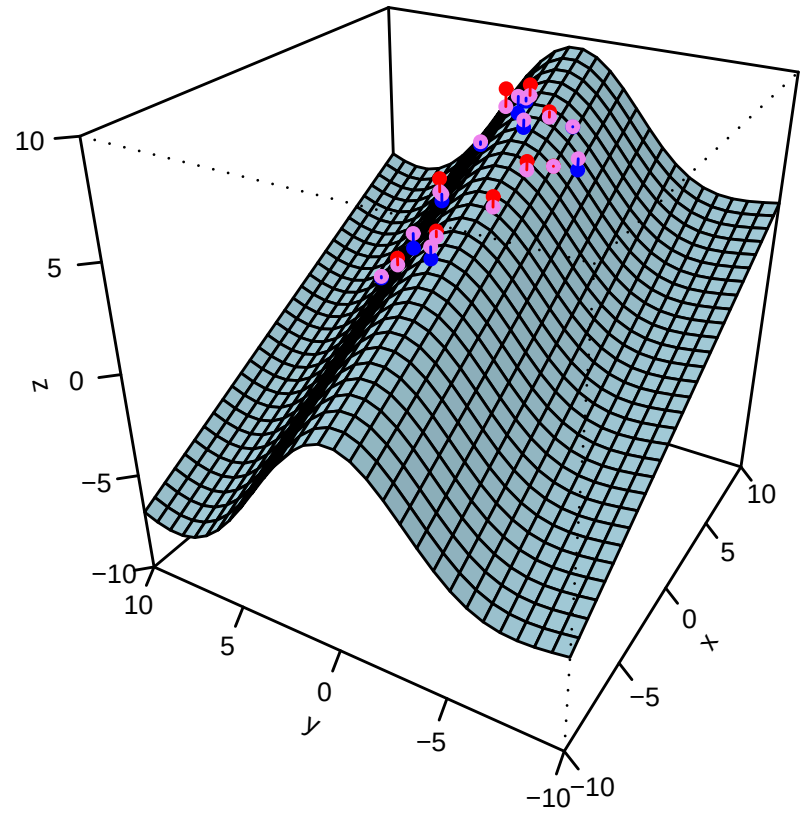




Variable 1: the linear and nonlinear relevance parameters for input 3 are both 0

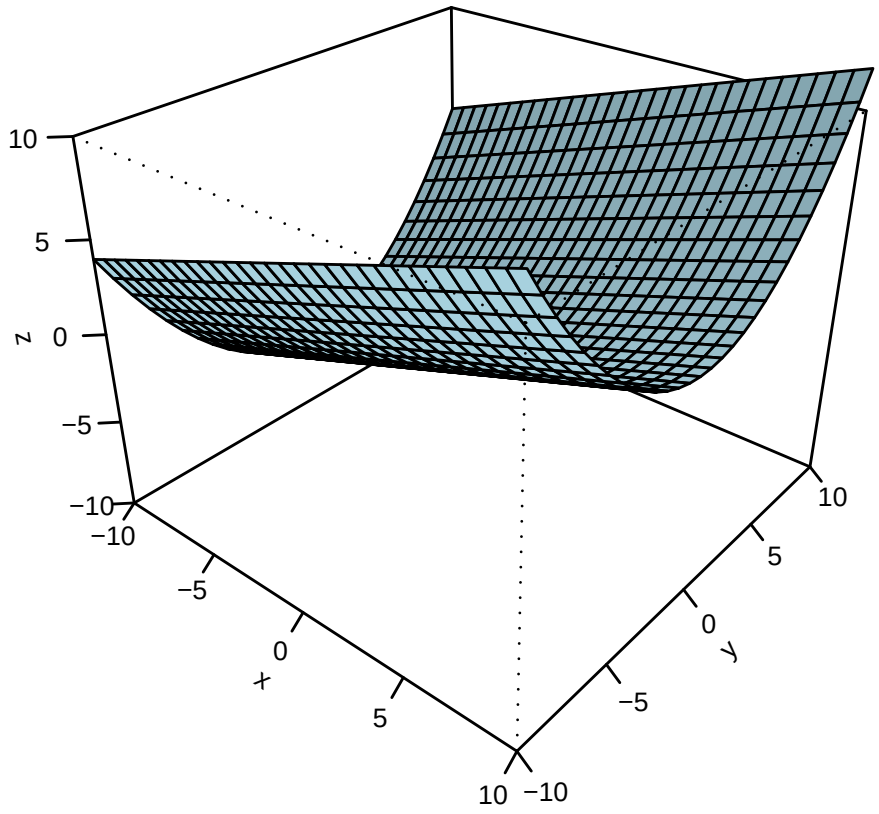
GP on time-series

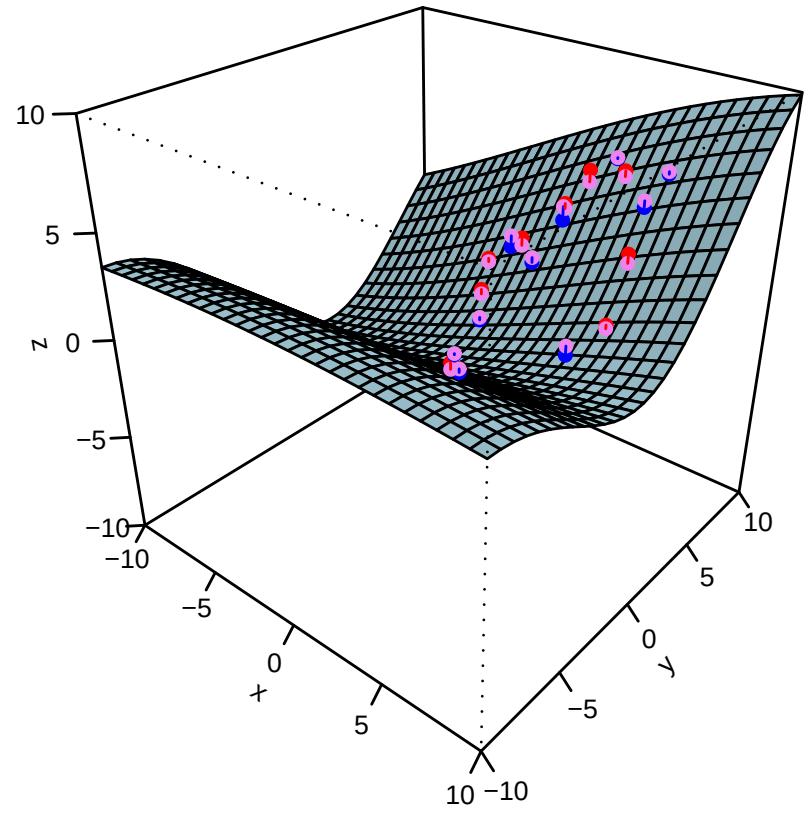




Variable 2: the linear and nonlinear relevance parameters for input 1 are both 0

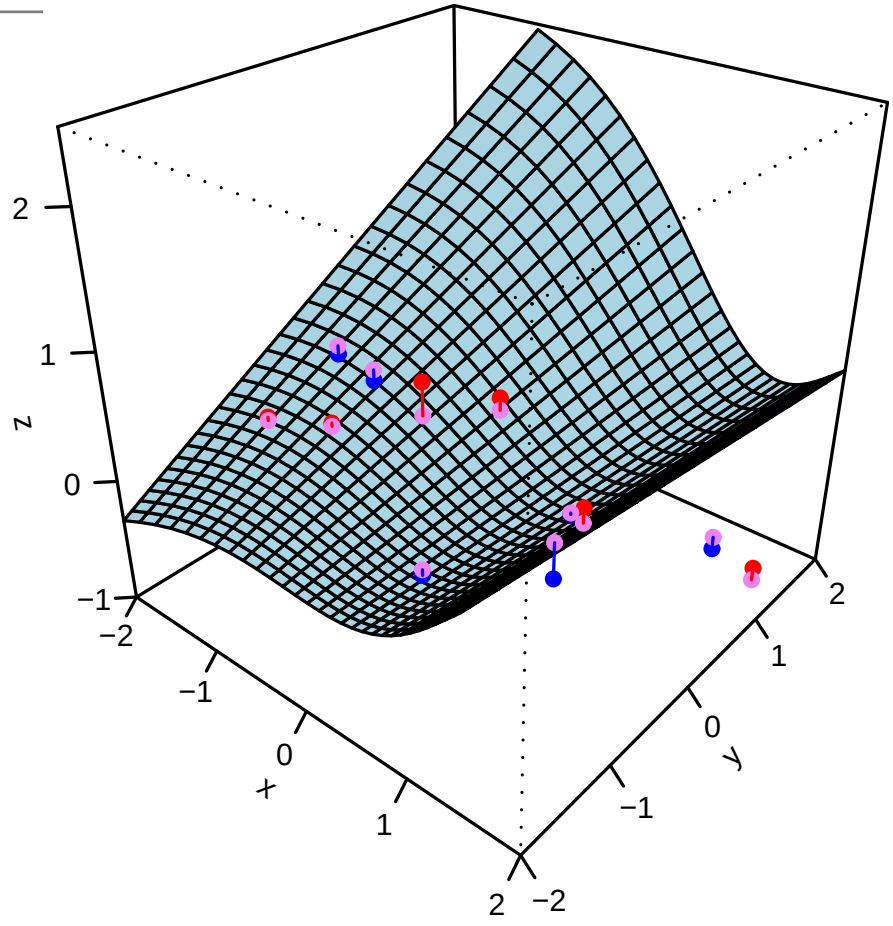
GP on time-series





Variable 3: the linear and nonlinear relevance parameters for input 1 are both 0

Gene network: LHY dependency



Nonlinear relevance:
0.01, 0.01, 0.73, 0.01

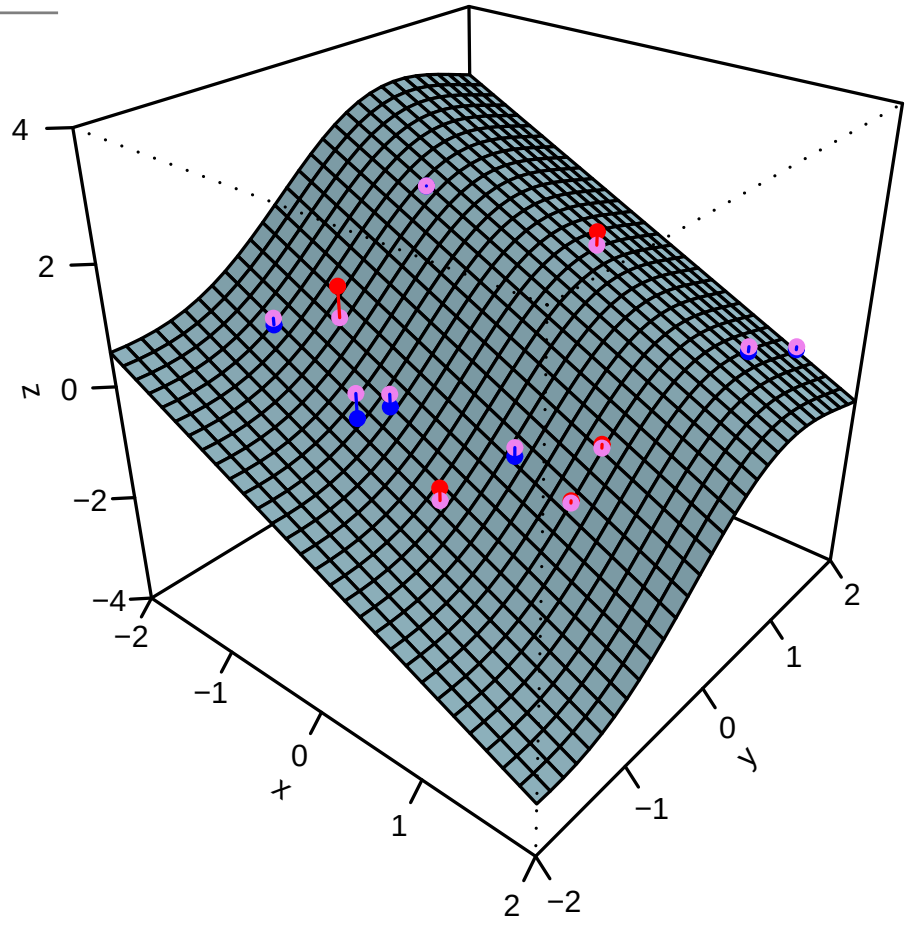
Linear relevance:
0.81, 1.13, 0.45, 0.00

Estimated sd 0.18

No dependency of LHY on PIF3

Nonlinear dependency of LHY on TOC1 and GI, LHY and PIF3 were set to 0

Gene network: GI dependency



Nonlinear relevance:
0.01, 0.00, 0.78, 0.00

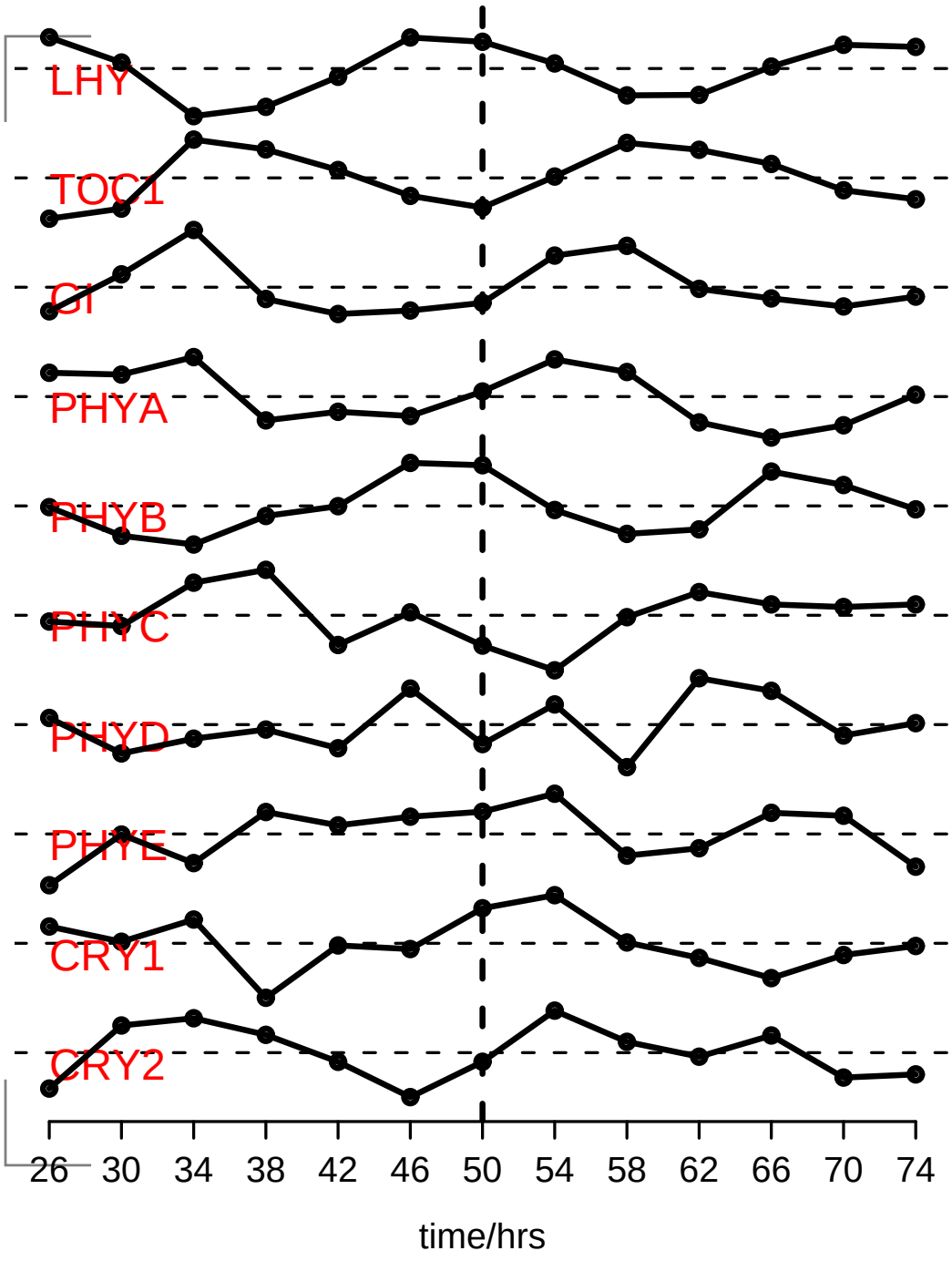
Linear relevance:
0.00, 0.82, 0.17, 0.00

Estimated sd 0.30

No dependency of GI on LHY and PIF3

Linear (negative) dependency of GI on TOC1,
nonlinear (positive) dependency of GI on itself

Light input pathway



Entrainment of 24h rhythm via light input

phytochromes (phy): red, IR

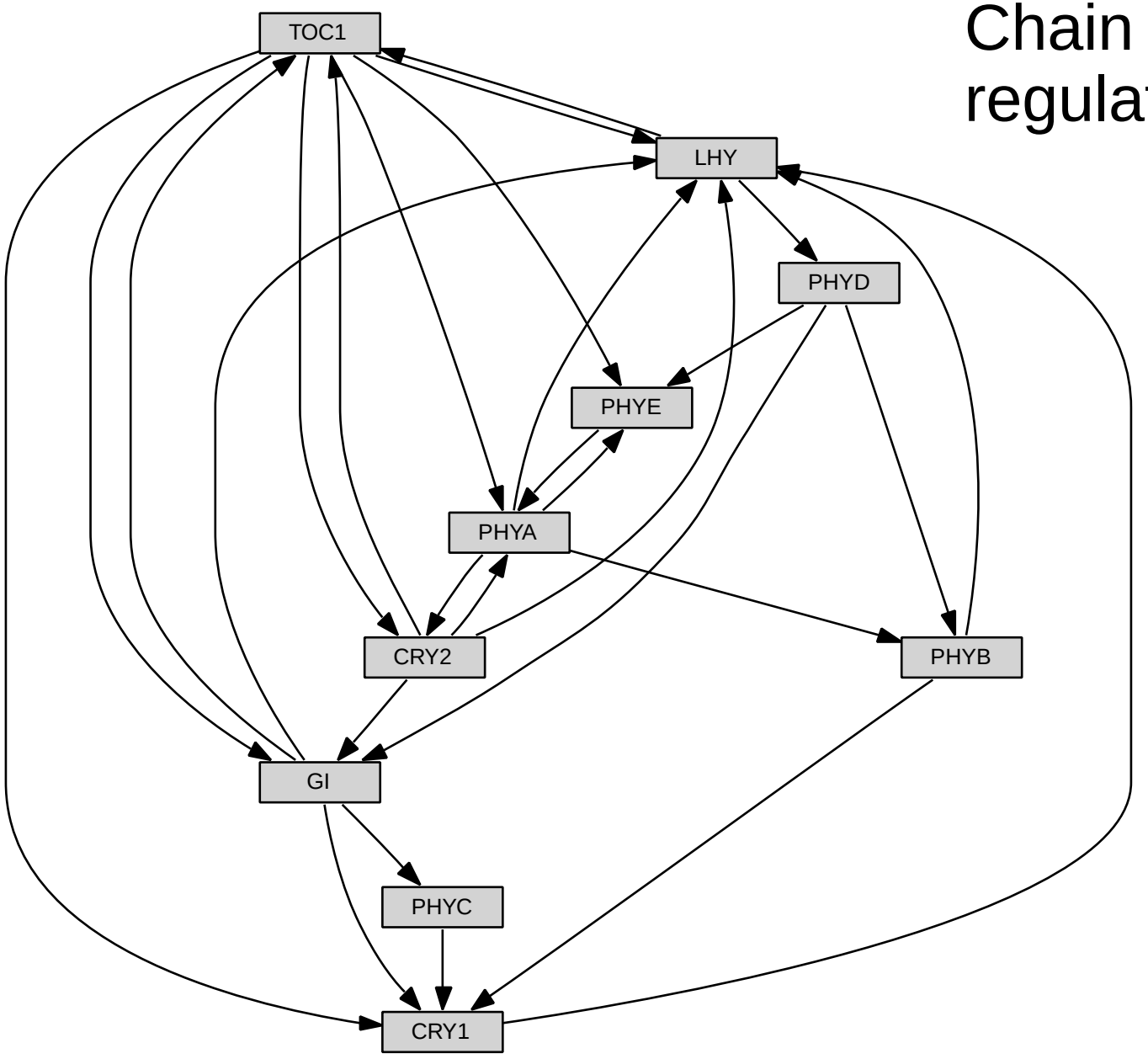
cryptochromes (cry): blue, UV

Even in constant light condition cycling (Cy2, PhyA, PhyB)

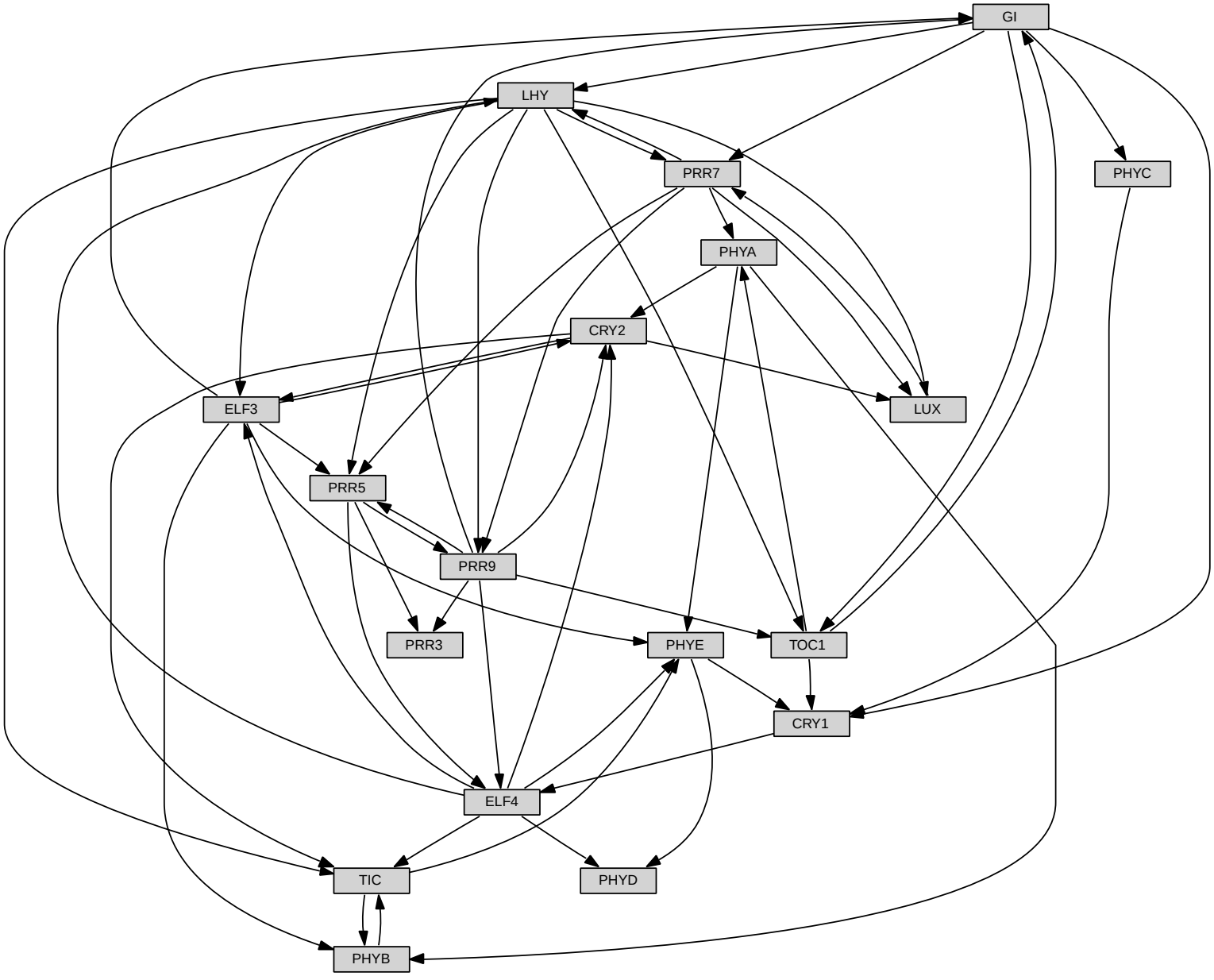
Bidirectional links from central clock?

Light input pathway

Chain of Phy and Cry regulation



Light input and PRR pathway



State space model

$$x_t = f(x_{t-1}) + \epsilon_1$$
$$y_t = Cx_t + \epsilon_2$$

If vector y represents observable variables (genes), use $C = (0, I)$

$f(x) = (f_1(x), \ldots, f_d(x))$ is vector of d **parallel GPs** each trained independently

Extended Kalman filtering with GPs: modify predictive mean and variance

Iterate with MLE type II estimation of relevance parameters: **ARD-EM algorithm for GP**

Extended Kalman filter

$$P(x_i) = N(x_i \mid x_p, V_p)$$
$$x_p = \tilde{\mu}(m_{i-1}, P_{i-1}), \quad V_p = \widetilde{\Sigma}(m_{i-1}, P_{i-1}) + Q$$
$$P(x_i \mid t_i) = N(x_i \mid m_i, P_i)$$
$$m_i = x_p + K(t_i - Cx_p), \quad P_i = (I - KC)V_p$$
$$K = V_p C'(CV_p C' + R)^{-1}$$

Need to calculate mean $\tilde{\mu}(u, S)$ and covariance $V_p = \widetilde{\Sigma}(u, S)$ of parallel GPs for an uncertain input $u \sim N(u, S)$ (similar to J. Quiñonero-Candela, A. Girard, and C. E. Rasmussen, 2003)

Uncertain input for parallel GPs

With covariances C_G and C_L mean and covariance exact, eg

$$\int C_G(x^*, x_j)\, p_G(x^* \mid u, S)\, dx^*$$

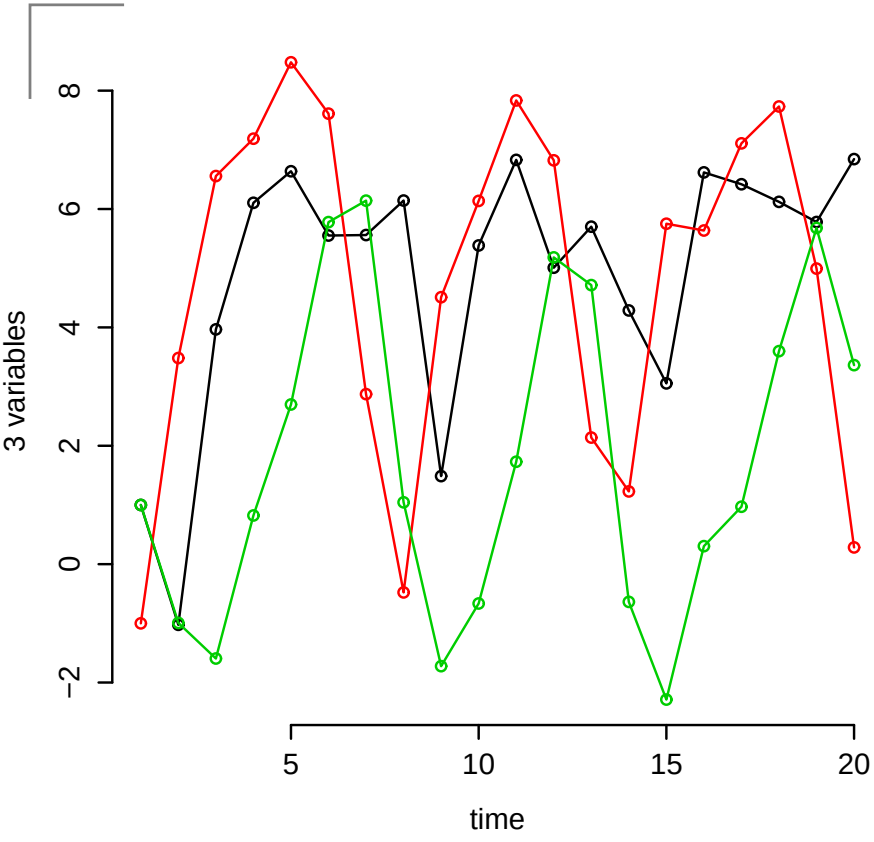
combination of two Gaussians

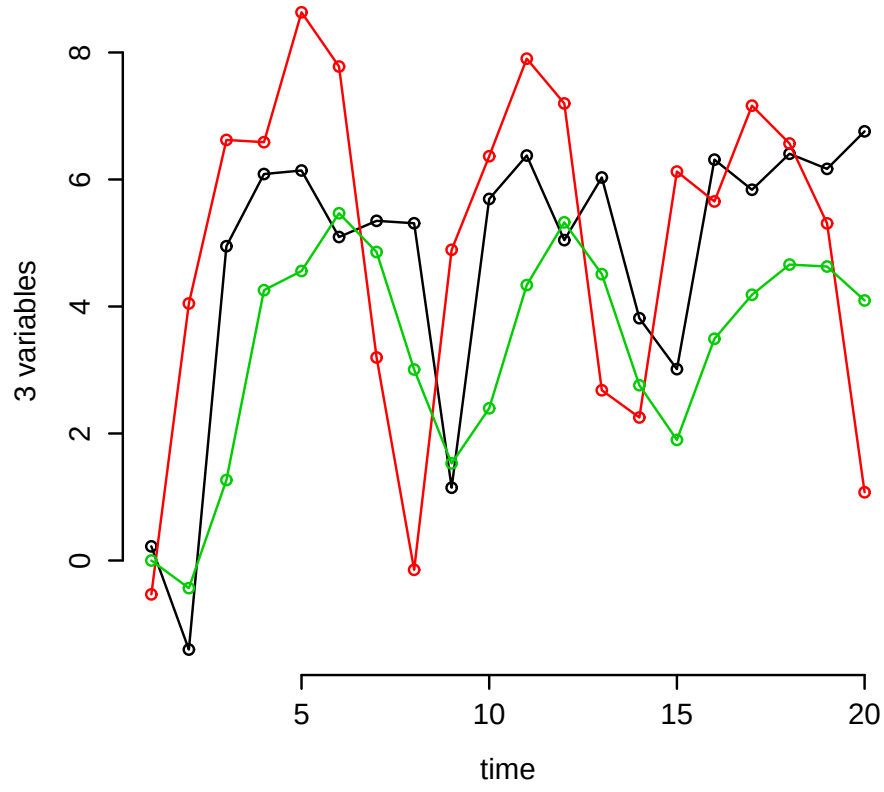
Variance of $f(x^*)$ is

$$E_{x^*}(\widetilde{\Sigma}(x^*)) + \mathrm{var}_{x^*}(\tilde{\mu}(x^*))$$

$\widetilde{\Sigma}(x^*)$ is composed of covariances of each GP
$\mathrm{var}_{x^*}(\tilde{\mu}(x^*))$ **involves covariances across GPs**
(solution along lines of Quiñonero-Candela et al.)

Reconstruction of hidden variable





3rd variable (green) treated as **hidden variable** in GP-EM reconstruction on left-hand side

Conclusion

- Complexity control (Occam's razor) by Bayesian estimation of hyperparameters
- MAP estimation of hyperparameters (Maximum likelihood type II) works fine
- Gaussian processes integrate linear and nonlinear components
- Downside: setting of prior parameters (a and b) above is critical, particularly noise parameter in case of noisy data
- GP EM possible but tricky due to presence of many local optima