

# Probabilistic Inference for Modelling of Transcription Factor Activity

Part of the PUMA project for Propagating Uncertainty in  
Microarray Analysis

**Neil Lawrence** and Magnus Rattray  
School of Computer Science  
University of Manchester

June 13, 2007

# Outline

## 1 Microarray Processing

- Affymetrix GeneChip Arrays

## 2 Transcription Factors

- ChIP-microarray and Transcription Factor Activities
- Transcription Factor Concentrations
- From Simple to Complex Models
- Biological Problem

## 3 Non-linear Response Model

- Linear Response with MLP Kernel
- Non-linear Responses

## 4 Conclusions

# Online Resources

All source code and slides are available online

- This talk available from my home page (see talks link on side).
- Project main page (with links to software)
  - <http://bioinf.manchester.ac.uk/resources/puma/>.
- Additional project homepage
  - <http://www.cs.man.ac.uk/~neill/projects/pipeline/>.

# PUMA Project Outline

## Noise Problems in Microarrays

- Project was motivated by the fact that microarray data is very noisy.
- The aim of the project is to:
  - Assess the level of noise in the estimated gene expression.
  - Propagate the noise through downstream analysis.
- Personnel:
  - **Investigators:** Neil Lawrence, Magnus Rattray
  - **Fellows/Post-docs:** Pei Gao (from July 1st), Marta Milo (Sheffield), Guido Sanguinetti (former post-doc Sheffield)
  - **PhD Students:** Xuejun Liu, Richard Pearson

# Central Dogma

DNA →mRNA →Protein

- Every cell has the same DNA.
- Cells produce different proteins (building blocks of life).
- Level of mRNA produced is known as *gene expression*.
- Has a downstream effect on level of Protein produced.
- Gene expression is controlled by *Transcription factors*.
- Transcription factors themselves are proteins.
  - Feedbacks in these systems lead to gene networks.

# Affymetrix Arrays

## Photolithography and Combinatorial Chemistry

- Affymetrix arrays are a technology for measuring level of mRNA.
- PM (perfect match) probes match the gene sequence.
- MM (mismatch) probes have wrong middle base.
- MM designed to measure non-specific binding.
- Approx 10,000 probe-sets per chip.

# Affymetrix Arrays

## Photolithography and Combinatorial Chemistry



Figure: Affymetrix arrays for human and mouse (image from Wikimedia Commons under GFDL).

# Affymetrix Arrays

## Photolithography and Combinatorial Chemistry

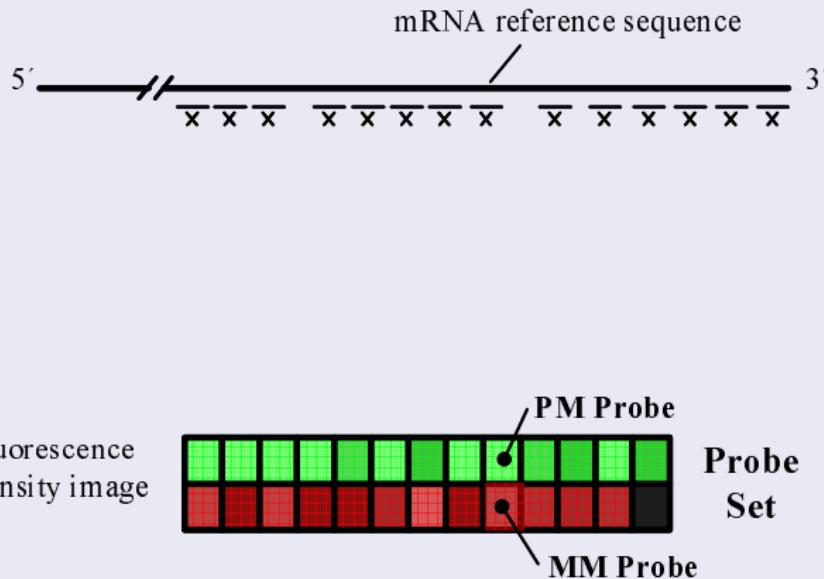


Figure: Affymetrix array schematic

# Affymetrix Arrays

## Photolithography and Combinatorial Chemistry

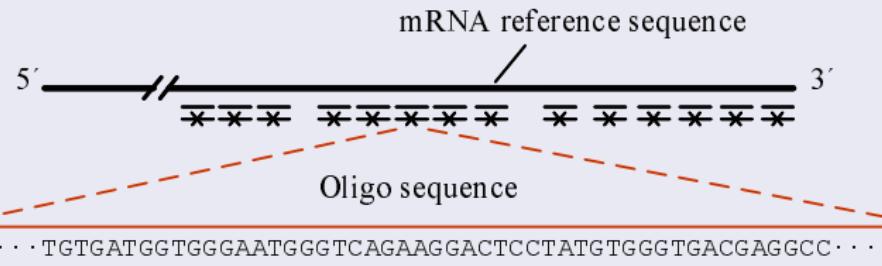


Figure: Affymetrix array schematic

# Affymetrix Arrays

## Photolithography and Combinatorial Chemistry

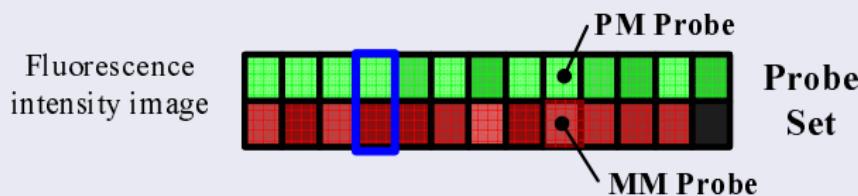
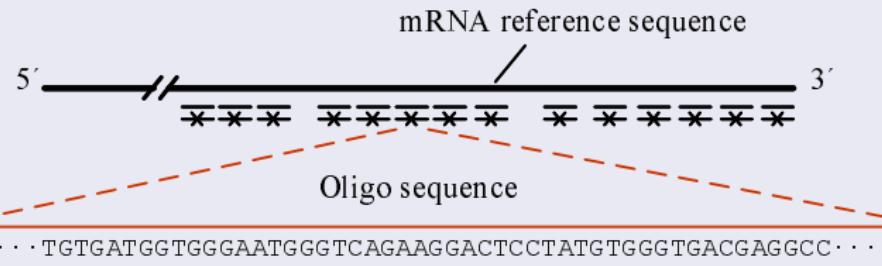


Figure: Affymetrix array schematic

# Affymetrix Arrays

## Photolithography and Combinatorial Chemistry

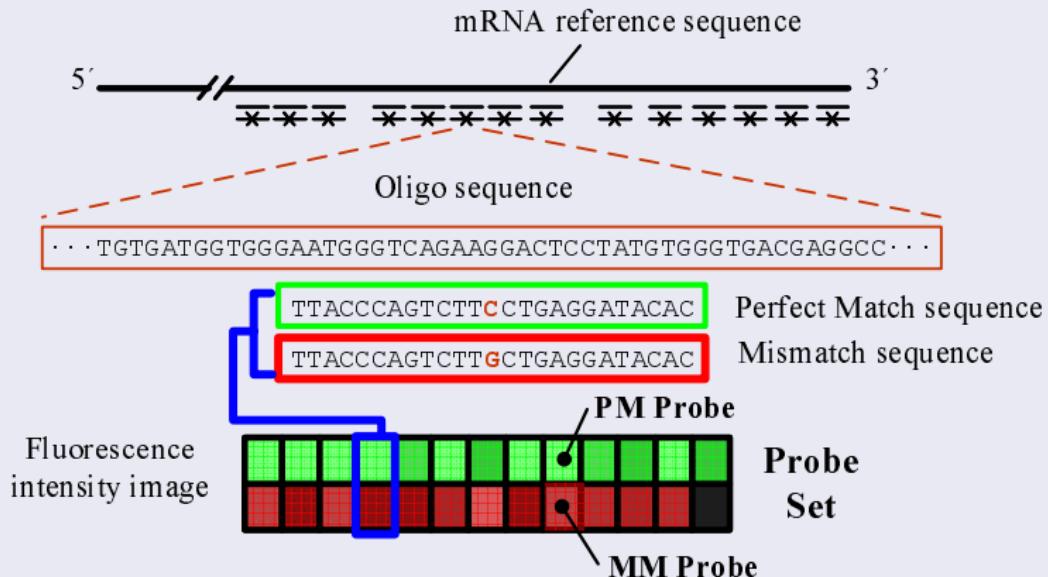


Figure: Affymetrix array schematic

# gMOS Family of Methods

## Gamma Model of Signal (Milo et al., 2003; Liu et al., 2005)

- Most methods return a single expression level estimate.
- The gMOS family of methods additionally provide confidence intervals.
- This confidence intervals can be propagated through higher level analysis.

## gMOS Family of Methods II

### Gamma Model of Signal

$$m_j \sim \text{Ga}(m_j | a, b)$$

$$s_j \sim \text{Ga}(s_j | \alpha, b)$$

$$y_j = m_j + s_j$$

$$y_j \sim \text{Ga}(y_j | a + \alpha, b)$$

$$\text{Ga}(x | a, b) = \frac{b^a}{\Gamma(a)} x^a \exp(-bx)$$

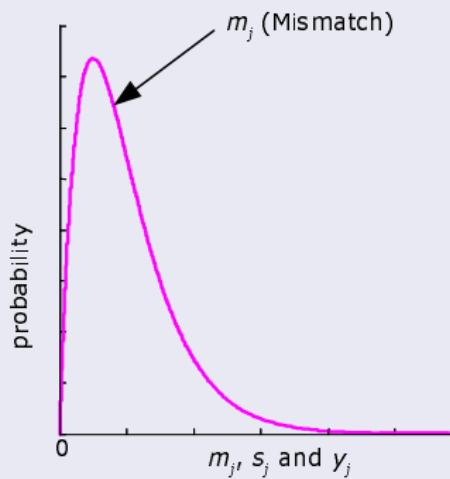


Figure: PDF of  $m_j$ ,  $s_j$  and the implied distribution for  $y_j$ .

## gMOS Family of Methods II

### Gamma Model of Signal

$$m_j \sim \text{Ga}(m_j | a, b)$$

$$s_j \sim \text{Ga}(s_j | \alpha, b)$$

$$y_j = m_j + s_j$$

$$y_j \sim \text{Ga}(y_j | a + \alpha, b)$$

$$\text{Ga}(x | a, b) = \frac{b^a}{\Gamma(a)} x^a \exp(-bx)$$

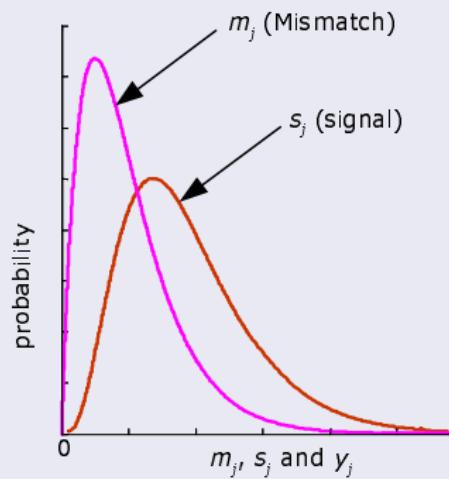


Figure: PDF of  $m_j$ ,  $s_j$  and the implied distribution for  $y_j$ .

## gMOS Family of Methods II

### Gamma Model of Signal

$$m_j \sim \text{Ga}(m_j | a, b)$$

$$s_j \sim \text{Ga}(s_j | \alpha, b)$$

$$y_j = m_j + s_j$$

$$y_j \sim \text{Ga}(y_j | a + \alpha, b)$$

$$\text{Ga}(x | a, b) = \frac{b^a}{\Gamma(a)} x^a \exp(-bx)$$

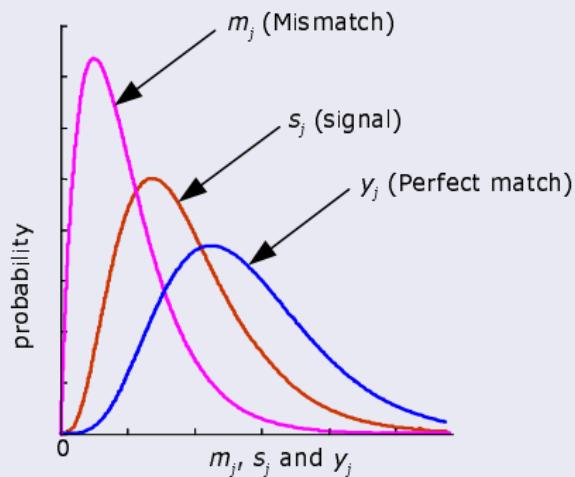


Figure: PDF of  $m_j$ ,  $s_j$  and the implied distribution for  $y_j$ .

## Inferring the Signal

- Maximise likelihood with respect to  $\alpha$ ,  $a$  and  $b$ .
  - Assume independence between  $y_j$  and  $m_j$ ,

$$p(y_j, m_j) = \text{Ga}(y_j | \alpha, b) \text{Ga}(m_j | a, b).$$

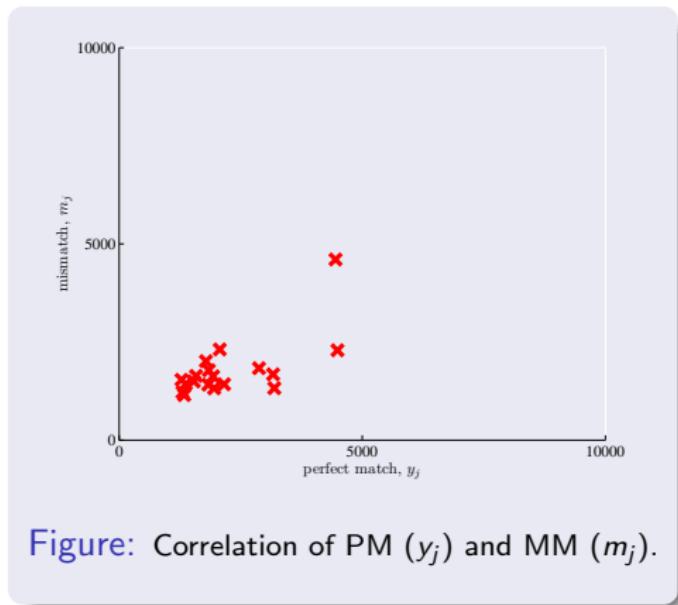
- Use resulting  $\hat{\alpha}$  and  $\hat{b}$  to give distribution over  $s_j$ .

$$p(s_j) = \text{Ga}\left(s_j | \hat{\alpha}, \hat{b}\right).$$

# Modelling Probe Pair Affinity

## mgMOS

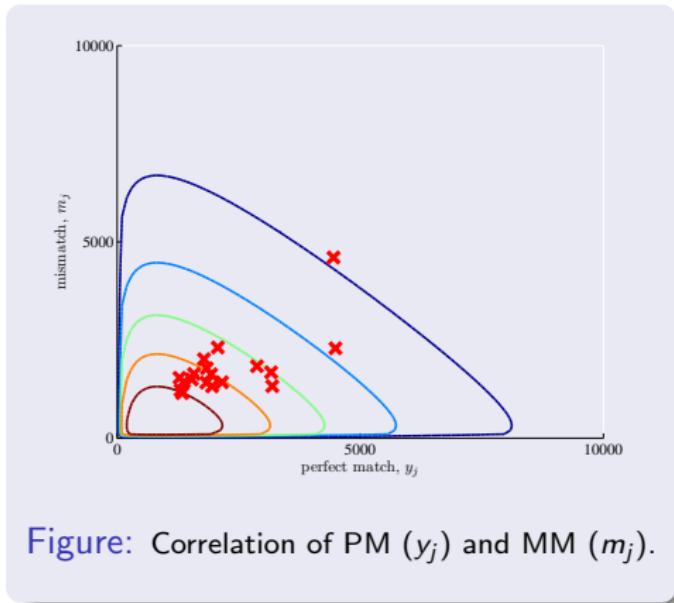
- $y_j$  and  $m_j$  are correlated.
- gMOS makes an independence assumption.
- Correlations arise through shared binding affinity (scale).
- Assume each probe pair has a shared scale  $b_j$ .
- Assume  $b_j \sim \text{Ga}(b_j|c, d)$  and marginalise.



# Modelling Probe Pair Affinity

## mgMOS

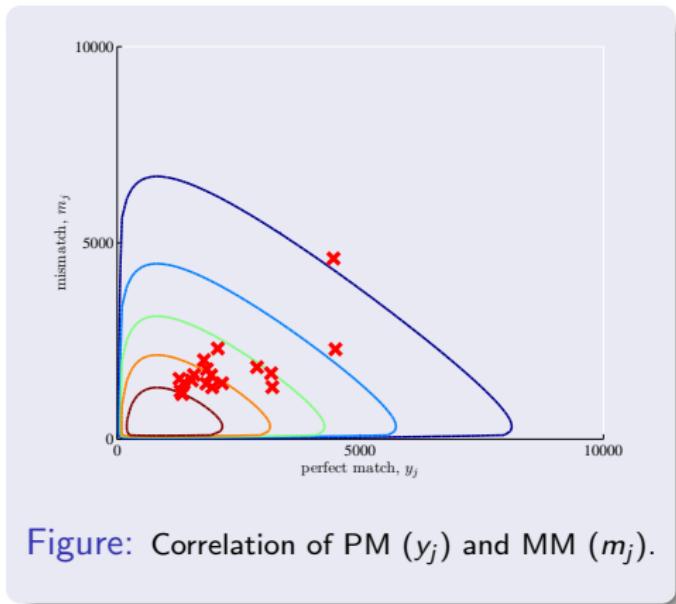
- $y_j$  and  $m_j$  are correlated.
- gMOS makes an independence assumption.
- Correlations arise through shared binding affinity (scale).
- Assume each probe pair has a shared scale  $b_j$ .
- Assume  $b_j \sim \text{Ga}(b_j|c, d)$  and marginalise.



# Modelling Probe Pair Affinity

## mgMOS

- $y_j$  and  $m_j$  are correlated.
- gMOS makes an independence assumption.
- Correlations arise through shared binding affinity (scale).
- Assume each probe pair has a shared scale  $b_j$ .
- Assume  $b_j \sim \text{Ga}(b_j|c, d)$  and marginalise.



# Modelling Probe Pair Affinity

## mgMOS

- $y_j$  and  $m_j$  are correlated.
- gMOS makes an independence assumption.
- Correlations arise through shared binding affinity (scale).
- Assume each probe pair has a shared scale  $b_j$ .
- Assume  $b_j \sim \text{Ga}(b_j|c, d)$  and marginalise.

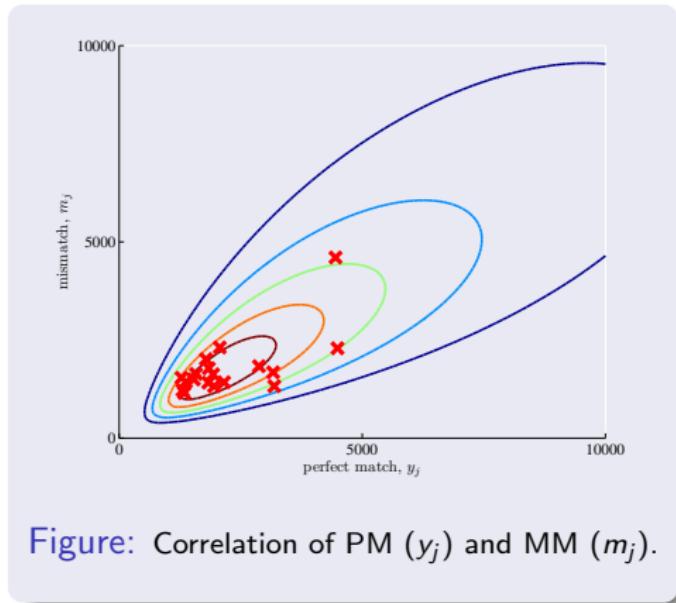
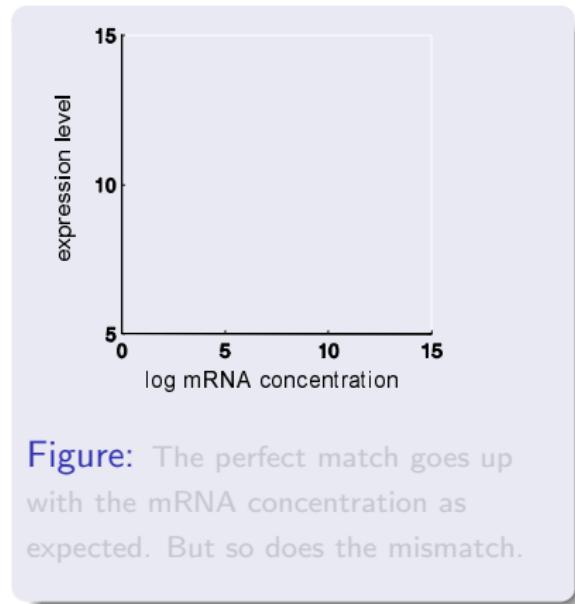


Figure: Correlation of PM ( $y_j$ ) and MM ( $m_j$ ).

# Specific Binding to Mismatch

## Mismatch Effected by Signal

- Affymetrix Latin Square Spike-In data set.
- The perfect match responds to increasing mRNA.
- But so does the mismatch.

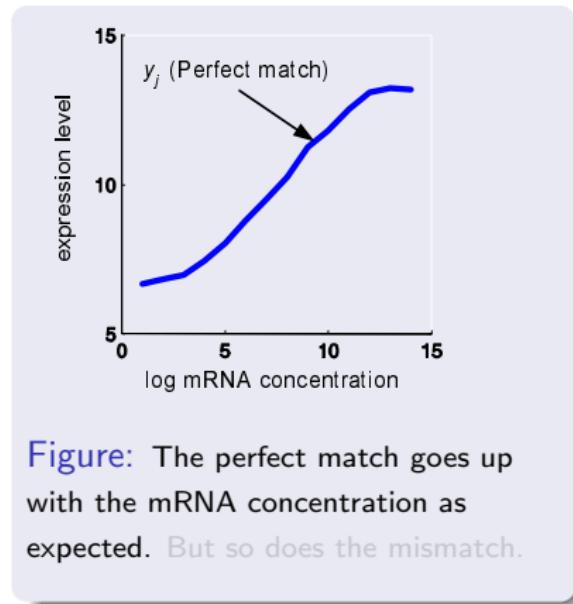


**Figure:** The perfect match goes up with the mRNA concentration as expected. But so does the mismatch.

# Specific Binding to Mismatch

## Mismatch Effected by Signal

- Affymetrix Latin Square Spike-In data set.
- The perfect match responds to increasing mRNA.
- But so does the mismatch.

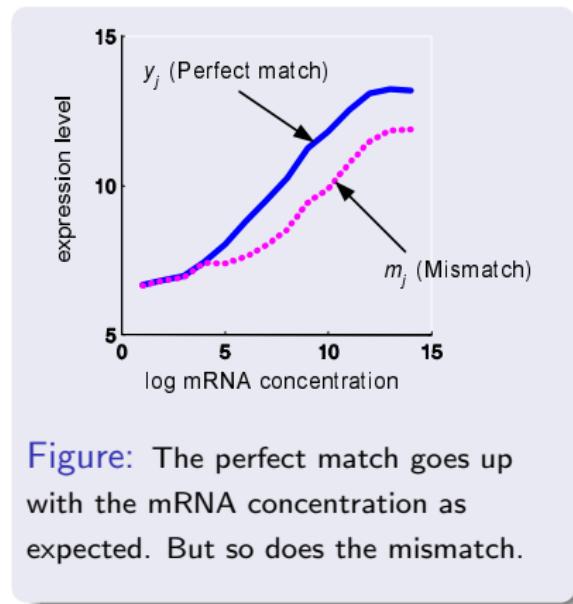


**Figure:** The perfect match goes up with the mRNA concentration as expected. But so does the mismatch.

# Specific Binding to Mismatch

## Mismatch Effected by Signal

- Affymetrix Latin Square Spike-In data set.
- The perfect match responds to increasing mRNA.
- But so does the mismatch.



**Figure:** The perfect match goes up with the mRNA concentration as expected. But so does the mismatch.

# Specific Binding and Multiple Arrays

## multi-mgMOS

- Specific Binding to MM probe:

- Introduce parameter  $\phi$  and assume

$$y_j \sim \text{Ga}(y_j|a + \alpha, b_j), \quad m_j \sim \text{Ga}(m_j|a + \phi\alpha, b_j)$$

- Log normal prior for  $\phi$  and seek a MAP solution.

- Multiple arrays:

- Still take  $b_j \sim \text{Ga}(b_j|c, d)$  but **share  $c$  and  $d$  parameters across chips.**

# Mouse Data Set

<http://www.ncbi.nlm.nih.gov/projects/geo>

Mouse back skin mRNA expression profile for Dab2 (Lin et al., 2004).

RMSE	Root Mean Square Error	
	qr-PCR	x-probe set
MAS 5.0	0.656	0.360
GCRMA	0.694	0.370
multi-mgMOS	0.601	0.233



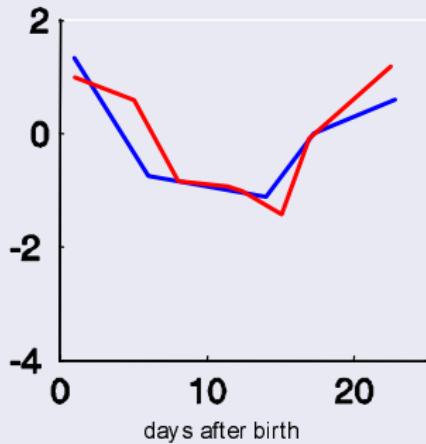
Prediction of Dab2 Expression level from qr-PCR, MAS 5.0, GCRMA and multi-mgMOS.

# Mouse Data Set

<http://www.ncbi.nlm.nih.gov/projects/geo>

Mouse back skin mRNA expression profile for Dab2 (Lin et al., 2004).

RMSE	Root Mean Square Error	
	qr-PCR	x-probe set
MAS 5.0	0.656	0.360
GCRMA	0.694	0.370
multi-mgMOS	0.601	0.233



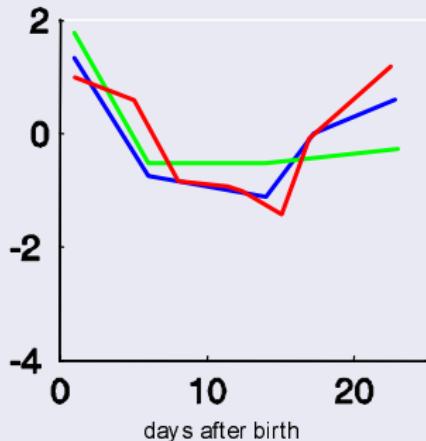
Prediction of Dab2 Expression level from qr-PCR, MAS 5.0, GCRMA and multi-mgMOS.

# Mouse Data Set

<http://www.ncbi.nlm.nih.gov/projects/geo>

Mouse back skin mRNA expression profile for Dab2 (Lin et al., 2004).

RMSE	Root Mean Square Error	
	qr-PCR	x-probe set
MAS 5.0	0.656	0.360
GCRMA	0.694	0.370
multi-mgMOS	0.601	0.233



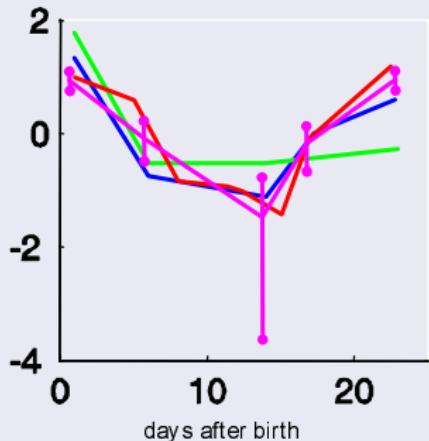
Prediction of Dab2 Expression level from qr-PCR, MAS 5.0, GCRMA and multi-mgMOS.

# Mouse Data Set

<http://www.ncbi.nlm.nih.gov/projects/geo>

Mouse back skin mRNA expression profile for Dab2 (Lin et al., 2004).

RMSE	Root Mean Square Error	
	qr-PCR	x-probe set
MAS 5.0	0.656	0.360
GCRMA	0.694	0.370
multi-mgMOS	0.601	0.233



Prediction of Dab2 Expression level from qr-PCR, MAS 5.0, GCRMA and multi-mgMOS.

# Transcription Factor Activities

## Inferring Activity of Transcription Factors

- Transcription factors control the expression of genes.
- Knowledge of their 'activity' is key to understanding the mechanism behind biological processes.
- Transcription factors are proteins — activity is a combination of their concentration and effect.
- The mRNA concentration of a given transcription factor may be known but:
  - Transcription factors are often lowly expressed — mRNA concentrations difficult to measure.
  - Transcription factors are often post-transcriptionally regulated.

# ChIP Microarrays

## Chromatine Immunoprecipitation (ChIP) Microarrays

- ChIP Microarrays tell us which TFs bind to which genes under certain conditions.
- In effect this gives a structure for the regulatory network.
- Combine this information with gene expression data to obtain transcription factor activities (TFA).

# Transcription Factor Activities

## Evaluating Activities of Transcription Factors

- Several approaches based on regression (Liao et al., 2003; Gao et al., 2004; Boulesteix and Strimmer, 2005; Alter and Golub, 2004)
- Assume a gene's expression is given by a linear relationship

$$\mathbf{y}_i = \mathbf{B}\mathbf{x}_i + \boldsymbol{\epsilon}_i.$$

$\mathbf{y}_i \in \mathbb{R}^{T \times 1}$  is the expression profile of the  $i$ th gene,

$\mathbf{x}_i \in \{0, 1\}^{q \times 1}$  indicates which transcription factors bind to the  $i$ th gene

$\mathbf{B} \in \mathbb{R}^{T \times q}$  is the matrix of TFAs.

$$\boldsymbol{\epsilon}_i \sim N(\mathbf{0}, \sigma^2 \mathbf{I})$$

- Problem: the matrix  $\mathbf{B}$  is *not* gene specific. It gives average TFA across genes.

# Gene Specific TFAs

## Associate TFAs to Genes (Sanguinetti et al., 2006)

- Introduce gene specific TFAs,

$$\mathbf{y}_i = \mathbf{B}_i \mathbf{x}_i + \boldsymbol{\epsilon}_i.$$

- Parameter Explosion
  - Assume prior distribution for  $\mathbf{B}_i$ .

$$p(\mathbf{B}) = \prod_{i=1}^N p(\mathbf{B}_i) = \prod_{i=1}^N \prod_{t=1}^{N_T} p(\mathbf{b}_{i,t})$$

$$p(\mathbf{b}_{i,t}) = N(\mathbf{b}_{i,t} | \mathbf{0}, \Sigma)$$

$\mathbf{b}_{i,t} \in \mathbb{R}^{q \times 1}$  is the vector of TFAs for each TF associated with the  $i$ th gene at time  $t$

# Gene Specific TFAs

## Associate TFAs to Genes (Sanguinetti et al., 2006)

- Introduce gene specific TFAs,

$$\mathbf{y}_i = \mathbf{B}_i \mathbf{x}_i + \boldsymbol{\epsilon}_i.$$

- Parameter Explosion

- Assume prior distribution for  $\mathbf{B}_i$ .

$$p(\mathbf{B}) = \prod_{i=1}^N p(\mathbf{B}_i) = \prod_{i=1}^N \prod_{t=1}^T p(\mathbf{b}_{i,t})$$

$$p(\mathbf{b}_{i,t}) = N(\mathbf{b}_{i,t} | \mathbf{0}, \Sigma)$$

$\mathbf{b}_{i,t} \in \mathbb{R}^{q \times 1}$  is the vector of TFAs for each TF associated with the  $i$ th gene at time  $t$

# Gene Specific TFAs

## Associate TFAs to Genes (Sanguinetti et al., 2006)

- Introduce gene specific TFAs,

$$\mathbf{y}_i = \mathbf{B}_i \mathbf{x}_i + \boldsymbol{\epsilon}_i.$$

- Parameter Explosion
  - Assume prior distribution for  $\mathbf{B}_i$ .

$$p(\mathbf{B}) = \prod_{i=1}^N p(\mathbf{B}_i) = \prod_{i=1}^N \prod_{t=1}^T p(\mathbf{b}_{i,t})$$

$$p(\mathbf{b}_{i,t}) = N(\mathbf{b}_{i,t} | \mathbf{0}, \Sigma)$$

$\mathbf{b}_{i,t} \in \mathbb{R}^{q \times 1}$  is the vector of TFAs for each TF associated with the  $i$ th gene at time  $t$

# Temporal Continuity of TFAs

## Time Course Experiments

- Introduce concept of temporal continuity with Gaussian distribution.

$$p(\mathbf{b}_{i,t} | \mathbf{b}_{i,t-1}) = N(\mathbf{b}_{i,t} | \gamma \mathbf{b}_{i,t-1} + (1 - \gamma) \boldsymbol{\mu}, (1 - \gamma^2) \boldsymbol{\Sigma})$$

The temporal continuity,  $\gamma$  is between 0 and 1.

# Temporal Continuity of TFAs II

## Effect of $\gamma$

- When  $\gamma = 0$  we recover

$$p(\mathbf{b}_{i,t}) = N(\mathbf{b}_{i,t} | \boldsymbol{\mu}, \boldsymbol{\Sigma})$$

which is equivalent to the original independent model.

- As  $\gamma \rightarrow 1$  we recover

$$p(\mathbf{b}_{i,t} | \mathbf{b}_{i,t-1}) = \lim_{\sigma^2 \rightarrow 0} N(\mathbf{b}_{i,t} | \mathbf{b}_{i,t-1}, \sigma^2 \mathbf{I})$$

which is appropriate if the 'time points' are in fact biological replicates.

# Results on TFAs

## Yeast Cell Cycle Data with ChIP-on-chip 204 TFs

- Yeast cell cycle *cdc15* data set (Spellman et al., 1998).
- ChIP-on-chip from 113 TFs (Lee et al., 2002).
- 24 experimental points in time series data.
- Compare with non-specific TFAs obtained by Regression.

# Results on TFAs II

## Graphs of TFAs

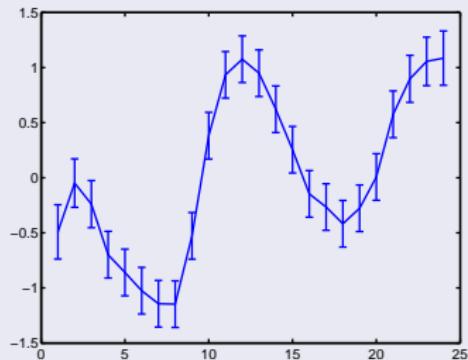
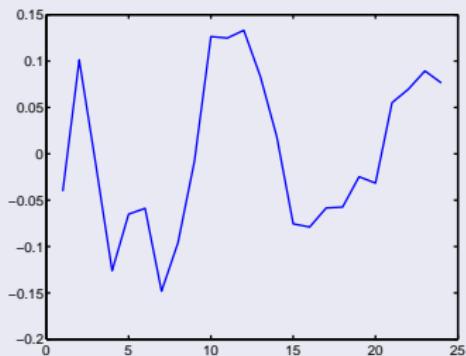


Figure: TFAs of ACE2 from the Spellman data. *Left:* TFA obtained by regression *Right:* gene specific TFA for average of  $\mathbf{B}_i$  across genes.

## Results on TFAs II

### Graphs of TFAs

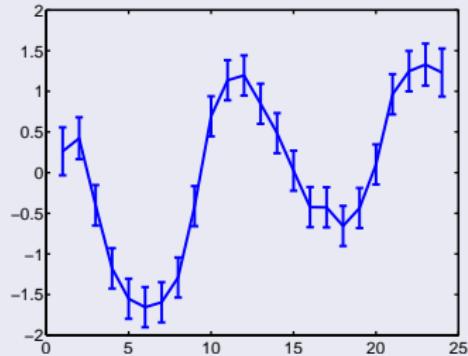
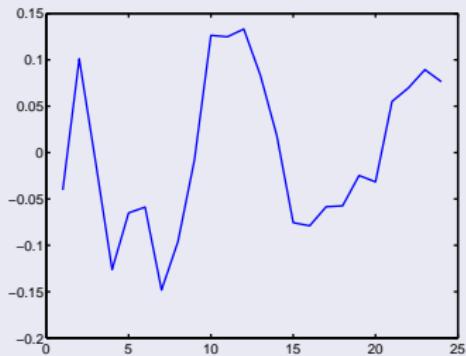


Figure: TFAs of ACE2 from the Spellman data. *Left:* TFA obtained by regression *Right:* gene specific TFA SCW11.

# Results on TFAs II

## Graphs of TFAs

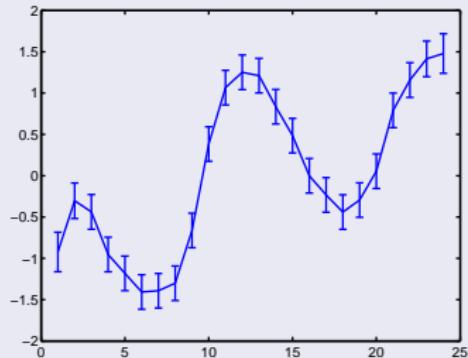
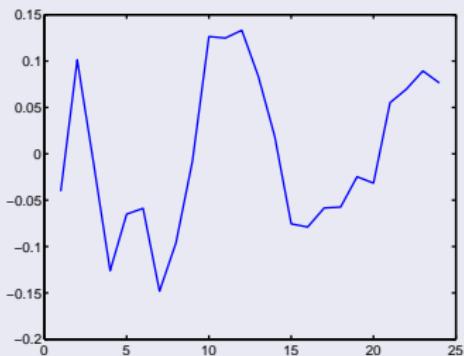


Figure: TFAs of ACE2 from the Spellman data. *Left:* TFA obtained by regression *Right:* gene specific TFA CTS1.

# Results on TFAs II

## Graphs of TFAs

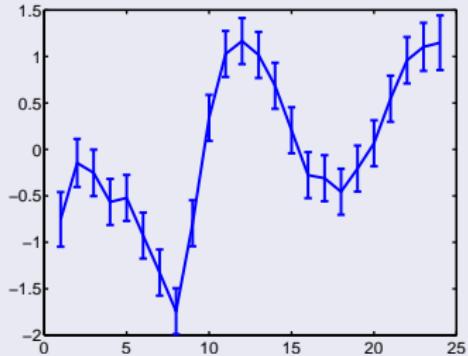
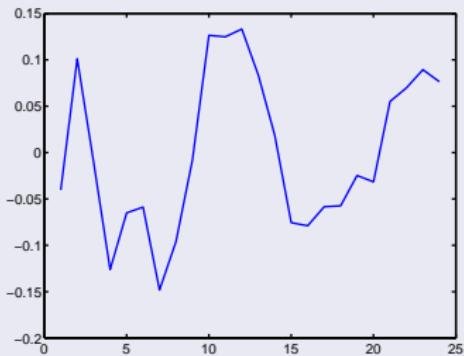


Figure: TFAs of ACE2 from the Spellman data. *Left:* TFA obtained by regression *Right:* gene specific TFA YER124C.

## Results on TFAs II

### Graphs of TFAs

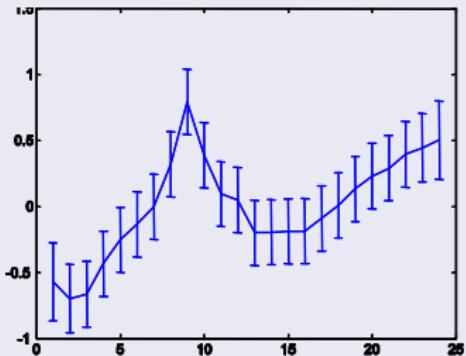
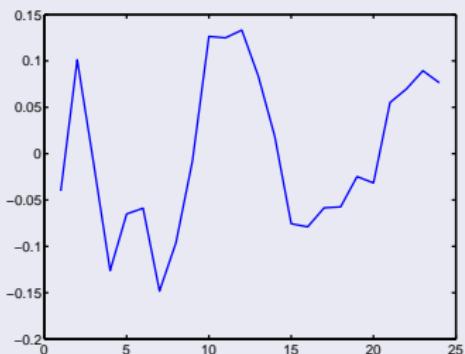


Figure: TFAs of ACE2 from the Spellman data. *Left:* TFA obtained by regression *Right:* gene specific TFA YKL51C.

# Separation of Concentration and Effect

## Splitting the Activity into Component Parts

- TFA is a combination of:
  - TF concentration.
  - TF effect.
- Model expression by splitting the two:

$$\mathbf{y}_i = (\mathbf{B} \odot \mathbf{X}) \mathbf{c}_t + \epsilon_t$$

where  $\odot$  is the Hadamard (element by element) product.

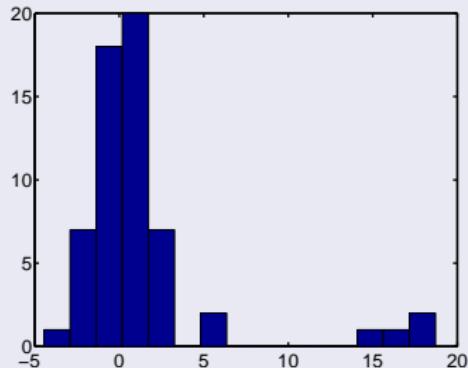
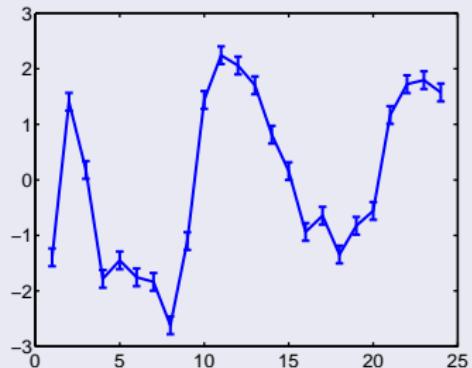
$\mathbf{B} \in \mathbb{R}^{N \times q}$  is a matrix of each TFs effect on each gene.

$\mathbf{c}_t \in \mathbb{R}^{q \times 1}$  is concentration of each TF at time  $t$ .

- Bayesian treatment of  $\mathbf{c}$  and  $\mathbf{B}$  through a variational approach.

# TF Concentration Results

## Concentration of ACE2



**Figure:** *Left:* concentration of ACE2 and *right:* effect of ACE2 on its target genes as a histogram.

# TF Concentration Results II

## Nice ACE2 Stories in Results

- ACE2 four most significant targets: CTS1, DSE1, DSE2, SCW11.
  - Evidence to back this up comes from biological literature.
  - CTS1 relationship is known.
  - DSE1 and DSE2 are involved in cell wall degradation causing daughter to separate from parent.
  - SCW11's function is unclear but protein is localised at cell wall.
- Negative regulation of NCE4
  - Not documented, but ACE2 terminates mitosis & NCE4 ensures DNA stability during replication

# More Complex Model

## Complex Models on Small Networks

- Simple linear models allow genome wide analysis of TFAs.
- We now consider a more complex model on a much smaller network.

# Differential Equation Model

## Inference of p53 Concentration

- p53 is an important in cancer.
- Many targets of p53 are not shared with other TFs.

## Differential Equation model

- Simple linear model differential equation model recently used by Barenco et al. (2006).
- Initially inferred transcription factor concentrations using Markov Chain Monte Carlo ( $10^7$  iterations). Now use maximum likelihood and curvature.
- We repeat their experiments with Gaussian processes.

# Simple Linear Model

## Linear model of regulation

$$\frac{dy_i(t)}{dt} = B_i + S_i f(t) - D_i y_i(t)$$

where:

- $y_i(t)$  — expression of the  $i$ th gene at time  $t$ .
- $f(t)$  — concentration of the transcription factor at time  $t$ .
- $D_i$  — gene's decay rate.
- $B_i$  — basal transcription rate.
- $S_i$  — sensitivity to the transcription factor.

# Equation Solution

## Solve via Laplace Transforms

- Solution to the equation:

$$y_i(t) = \frac{B_i}{D_i} + S_i \exp(-D_i t) \int_0^t f(u) \exp(D_i u) du.$$

If  $f(t)$  is a zero mean Gaussian process then  $y_i(t)$  is also a Gaussian process with mean  $\frac{B_i}{D_i}$ .

# Two Properties of GPs

## Integral of Gaussian Process

The integral of a GP is also a GP,

$$f(t) \sim N(\mathbf{0}, \mathbf{K}_{ff})$$

and

$$g(t) = \int_0^t f(u) du$$

then

$$g(t) \sim N(\mathbf{0}, \mathbf{K}_{gg}),$$

where

$$k_{gg}(t, t') = \int_0^t \int_0^{t'} k_{ff}(u, u') du du'$$

# Two Properties of GPs

Product with deterministic function

The integral of a GP is also a GP,

$$f(t) \sim N(\mathbf{0}, \mathbf{K}_{ff}),$$

and

$$g(t) = f(t) h(t)$$

where  $h(t)$  is a deterministic function then,

$$g(t) \sim N(\mathbf{0}, \mathbf{K}_{gg}),$$

where

$$k_{gg}(t, t') = h(t) k_{ff}(t, t') h(t')$$

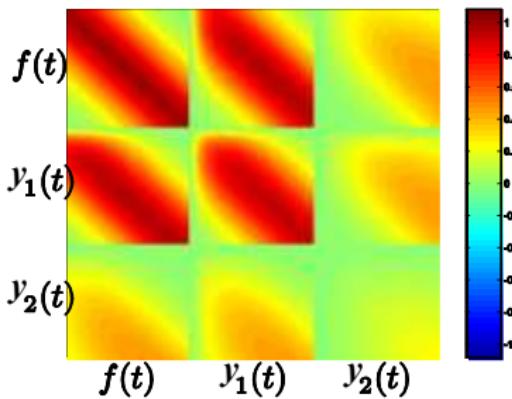
# Covariance for Transcription Model

RBF Kernel function for  $f(t)$

$$y_i(t) = \frac{B_i}{D_i} + S_i \exp(-D_i t) \int_0^t f(u) \exp(D_i u) du.$$

- Joint distribution for  $x_1(t)$ ,  $x_2(t)$  and  $f(t)$ .
- Here:

$D_1$	$S_1$	$D_2$	$S_2$
5	5	0.5	0.5



# Joint Sampling of $y(t)$ and $f(t)$ from Covariance

gpsimTest

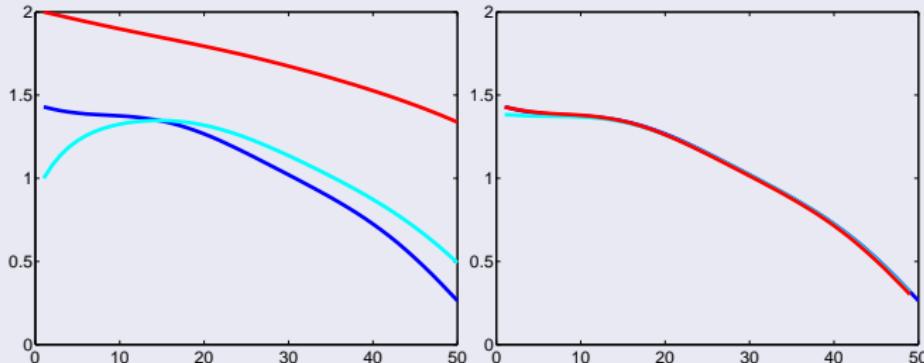


Figure: *Left:* joint samples from the transcription covariance, *blue*:  $f(t)$ , *cyan*:  $y_1(t)$  and *red*:  $y_2(t)$ . *Right:* numerical solution for  $f(t)$  of the differential equation from  $y_1(t)$  and  $y_2(t)$  (blue and cyan). True  $f(t)$  included for comparison.

# Joint Sampling of $y(t)$ and $f(t)$ from Covariance

gpsimTest

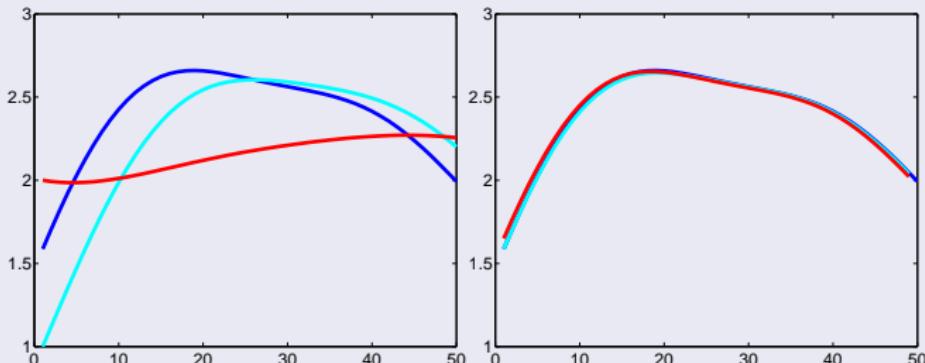


Figure: *Left:* joint samples from the transcription covariance, *blue*:  $f(t)$ , *cyan*:  $y_1(t)$  and *red*:  $y_2(t)$ . *Right:* numerical solution for  $f(t)$  of the differential equation from  $y_1(t)$  and  $y_2(t)$  (blue and cyan). True  $f(t)$  included for comparison.

# Joint Sampling of $y(t)$ and $f(t)$ from Covariance

gpsimTest

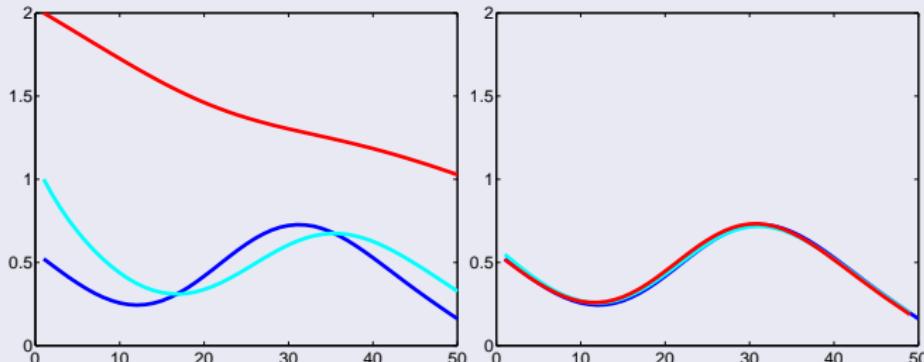


Figure: *Left:* joint samples from the transcription covariance, *blue:*  $f(t)$ , *cyan:*  $y_1(t)$  and *red:*  $y_2(t)$ . *Right:* numerical solution for  $f(t)$  of the differential equation from  $y_1(t)$  and  $y_2(t)$  (blue and cyan). True  $f(t)$  included for comparison.

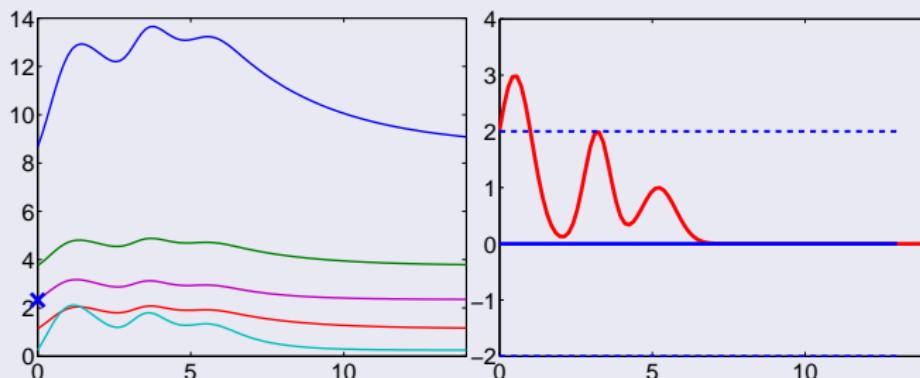
# Artificial Data

## Toy Problem

- Results from an artificial data set.
- We used a 'known TFC' and derived six 'mRNA profiles'.
  - Known TFC composed of three Gaussian basis functions.
  - mRNA profiles derived analytically.
- Fourteen subsamples were taken and corrupted by noise.
- This 'data' was then used to:
  - Learn decays, sensitivities and basal transcription rates.
  - Infer a posterior distribution over the missing TFC.

# Artificial Data Results

demToyProblem1



**Figure:** *Left:* The TFC,  $f(t)$ , which drives the system. *Middle:* Five gene mRNA concentration profiles each obtained by using different parameter sets  $\{B_i, S_i, D_i\}_{i=1}^5$  (lines) along with noise corrupted 'data'. *Right:* The inferred TFC (with error bars).

# Artificial Data Results

demToyProblem1

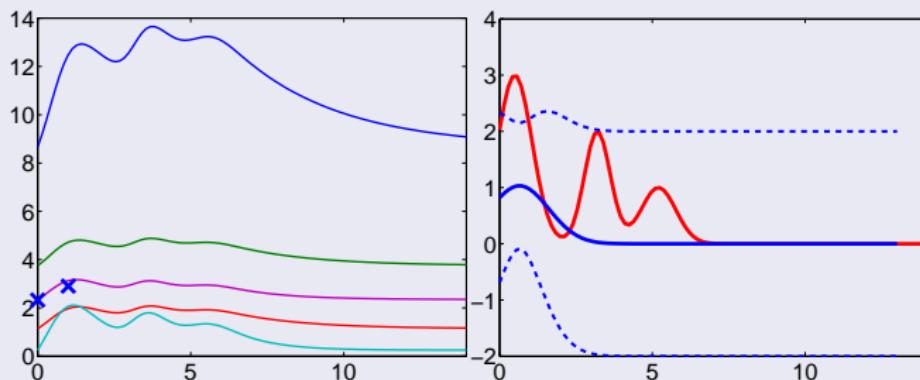


Figure: *Left*: The TFC,  $f(t)$ , which drives the system. *Middle*: Five gene mRNA concentration profiles each obtained by using different parameter sets  $\{B_i, S_i, D_i\}_{i=1}^5$  (lines) along with noise corrupted 'data'. *Right*: The inferred TFC (with error bars).

# Artificial Data Results

demToyProblem1

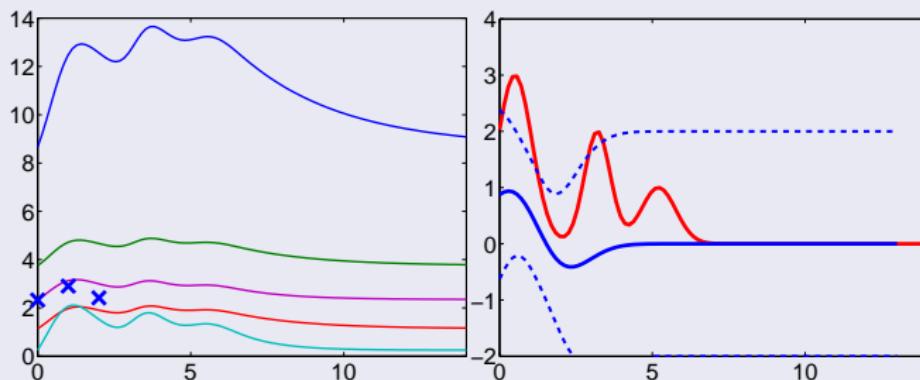


Figure: *Left*: The TFC,  $f(t)$ , which drives the system. *Middle*: Five gene mRNA concentration profiles each obtained by using different parameter sets  $\{B_i, S_i, D_i\}_{i=1}^5$  (lines) along with noise corrupted 'data'. *Right*: The inferred TFC (with error bars).

# Artificial Data Results

demToyProblem1

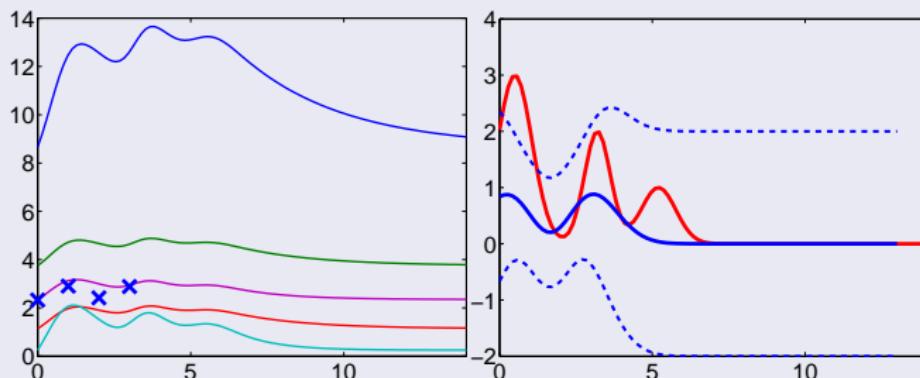
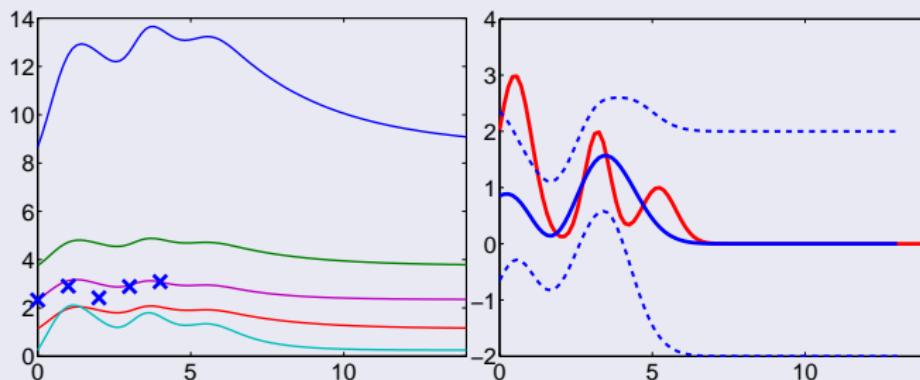


Figure: *Left*: The TFC,  $f(t)$ , which drives the system. *Middle*: Five gene mRNA concentration profiles each obtained by using different parameter sets  $\{B_i, S_i, D_i\}_{i=1}^5$  (lines) along with noise corrupted 'data'. *Right*: The inferred TFC (with error bars).

# Artificial Data Results

demToyProblem1



**Figure:** *Left:* The TFC,  $f(t)$ , which drives the system. *Middle:* Five gene mRNA concentration profiles each obtained by using different parameter sets  $\{B_i, S_i, D_i\}_{i=1}^5$  (lines) along with noise corrupted 'data'. *Right:* The inferred TFC (with error bars).

# Artificial Data Results

demToyProblem1

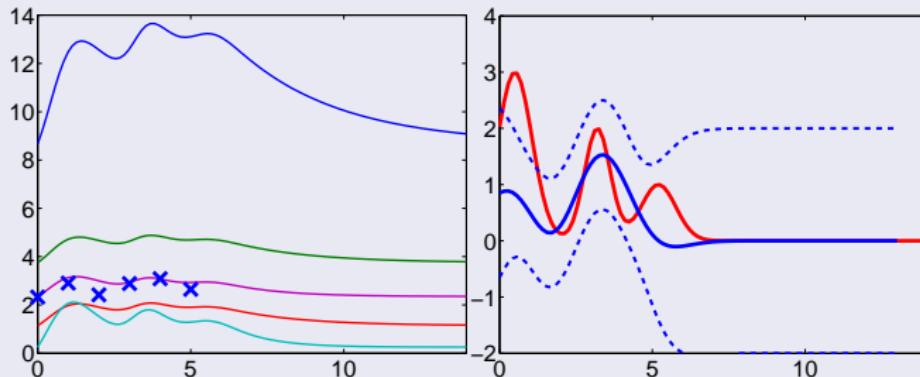


Figure: *Left:* The TFC,  $f(t)$ , which drives the system. *Middle:* Five gene mRNA concentration profiles each obtained by using different parameter sets  $\{B_i, S_i, D_i\}_{i=1}^5$  (lines) along with noise corrupted 'data'. *Right:* The inferred TFC (with error bars).

# Artificial Data Results

demToyProblem1

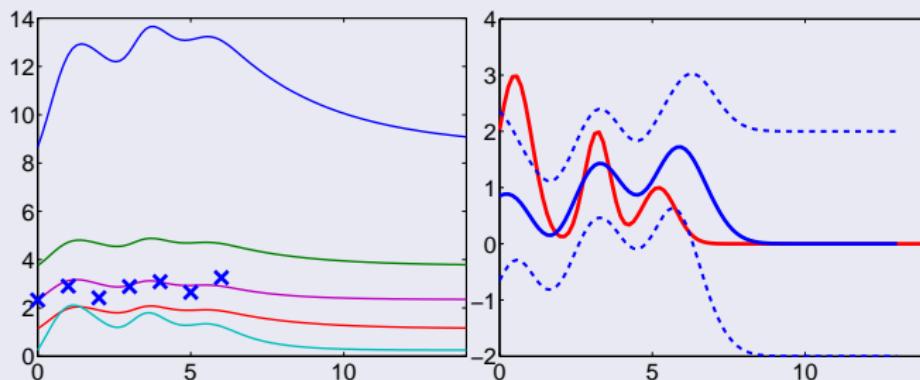


Figure: *Left*: The TFC,  $f(t)$ , which drives the system. *Middle*: Five gene mRNA concentration profiles each obtained by using different parameter sets  $\{B_i, S_i, D_i\}_{i=1}^5$  (lines) along with noise corrupted 'data'. *Right*: The inferred TFC (with error bars).

# Artificial Data Results

demToyProblem1

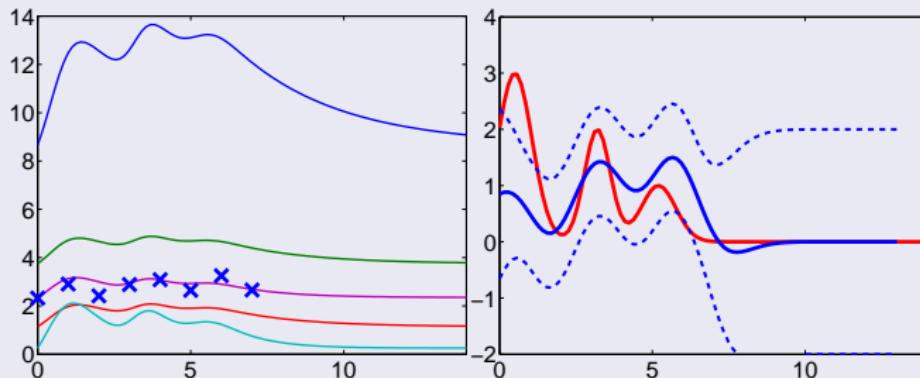


Figure: *Left*: The TFC,  $f(t)$ , which drives the system. *Middle*: Five gene mRNA concentration profiles each obtained by using different parameter sets  $\{B_i, S_i, D_i\}_{i=1}^5$  (lines) along with noise corrupted 'data'. *Right*: The inferred TFC (with error bars).

# Artificial Data Results

demToyProblem1

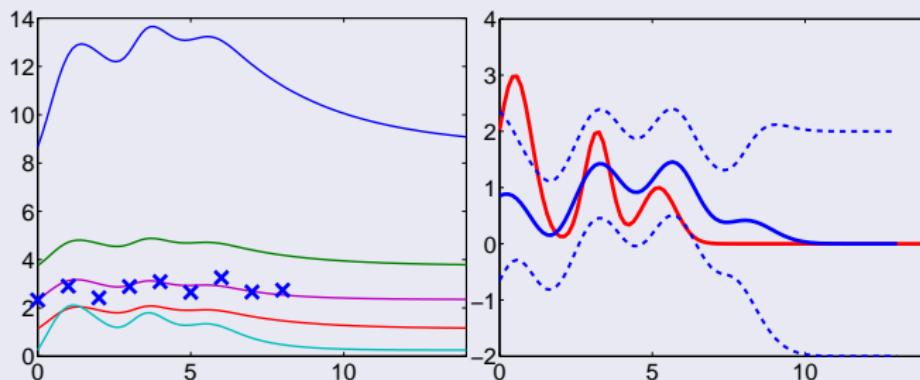


Figure: *Left*: The TFC,  $f(t)$ , which drives the system. *Middle*: Five gene mRNA concentration profiles each obtained by using different parameter sets  $\{B_i, S_i, D_i\}_{i=1}^5$  (lines) along with noise corrupted 'data'. *Right*: The inferred TFC (with error bars).

# Artificial Data Results

demToyProblem1

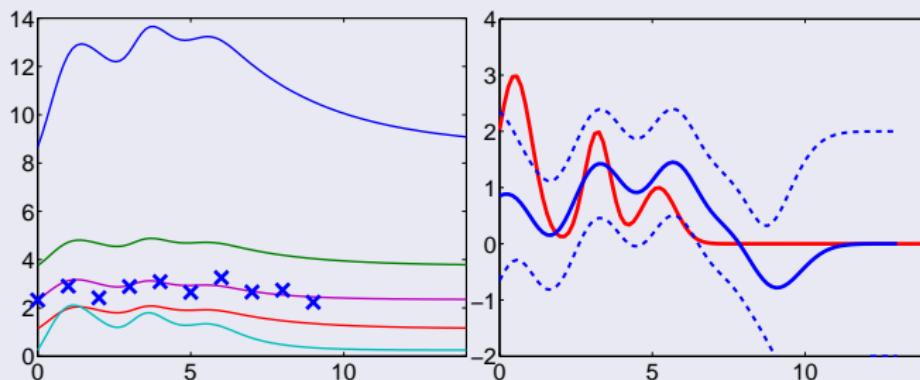
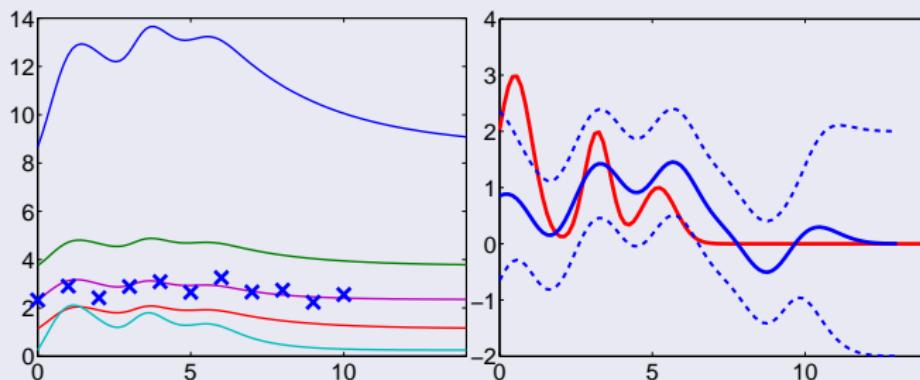


Figure: *Left*: The TFC,  $f(t)$ , which drives the system. *Middle*: Five gene mRNA concentration profiles each obtained by using different parameter sets  $\{B_i, S_i, D_i\}_{i=1}^5$  (lines) along with noise corrupted 'data'. *Right*: The inferred TFC (with error bars).

# Artificial Data Results

demToyProblem1



**Figure:** *Left:* The TFC,  $f(t)$ , which drives the system. *Middle:* Five gene mRNA concentration profiles each obtained by using different parameter sets  $\{B_i, S_i, D_i\}_{i=1}^5$  (lines) along with noise corrupted 'data'. *Right:* The inferred TFC (with error bars).

# Artificial Data Results

demToyProblem1

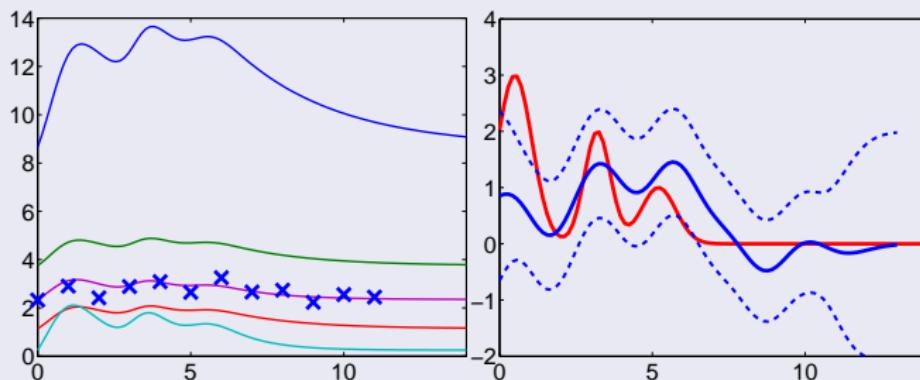
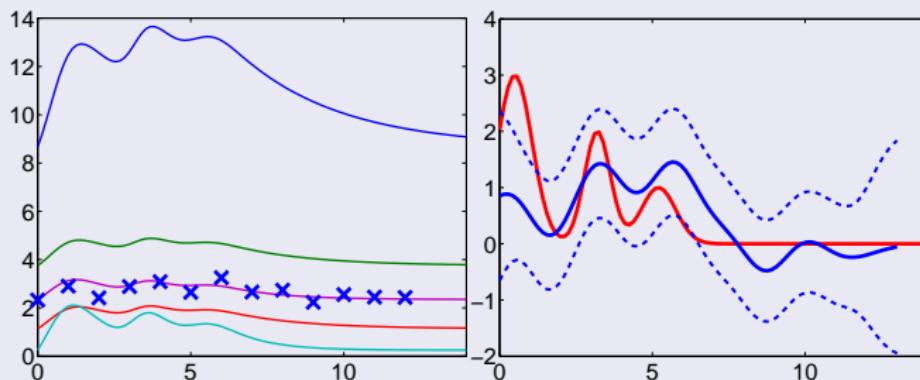


Figure: *Left*: The TFC,  $f(t)$ , which drives the system. *Middle*: Five gene mRNA concentration profiles each obtained by using different parameter sets  $\{B_i, S_i, D_i\}_{i=1}^5$  (lines) along with noise corrupted 'data'. *Right*: The inferred TFC (with error bars).

# Artificial Data Results

demToyProblem1



**Figure:** *Left:* The TFC,  $f(t)$ , which drives the system. *Middle:* Five gene mRNA concentration profiles each obtained by using different parameter sets  $\{B_i, S_i, D_i\}_{i=1}^5$  (lines) along with noise corrupted 'data'. *Right:* The inferred TFC (with error bars).

# Artificial Data Results

demToyProblem1

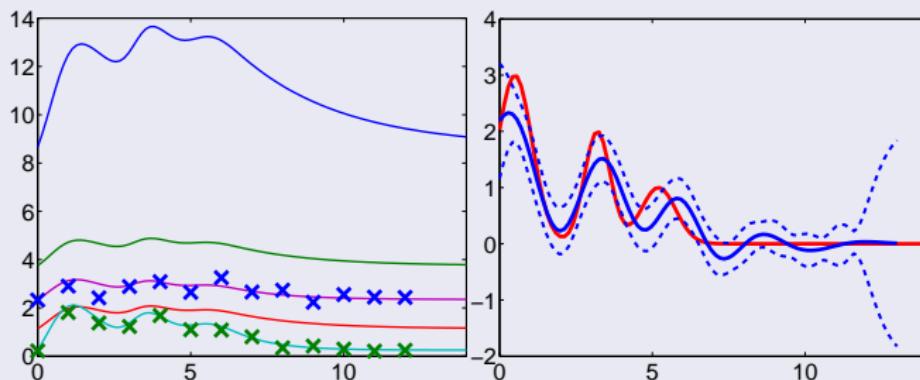


Figure: *Left*: The TFC,  $f(t)$ , which drives the system. *Middle*: Five gene mRNA concentration profiles each obtained by using different parameter sets  $\{B_i, S_i, D_i\}_{i=1}^5$  (lines) along with noise corrupted 'data'. *Right*: The inferred TFC (with error bars).

# Artificial Data Results

demToyProblem1

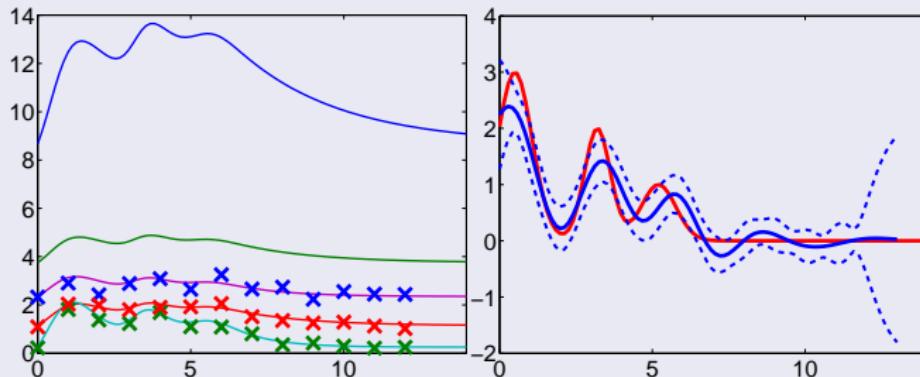


Figure: *Left*: The TFC,  $f(t)$ , which drives the system. *Middle*: Five gene mRNA concentration profiles each obtained by using different parameter sets  $\{B_i, S_i, D_i\}_{i=1}^5$  (lines) along with noise corrupted 'data'. *Right*: The inferred TFC (with error bars).

# Artificial Data Results

demToyProblem1

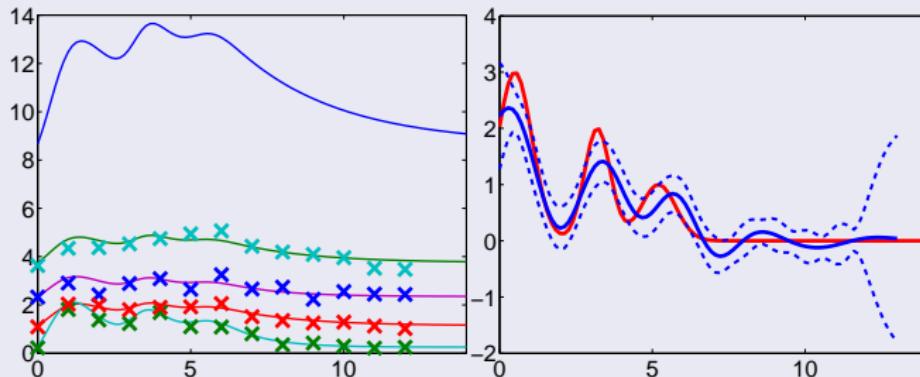


Figure: *Left*: The TFC,  $f(t)$ , which drives the system. *Middle*: Five gene mRNA concentration profiles each obtained by using different parameter sets  $\{B_i, S_i, D_i\}_{i=1}^5$  (lines) along with noise corrupted 'data'. *Right*: The inferred TFC (with error bars).

# Artificial Data Results

demToyProblem1

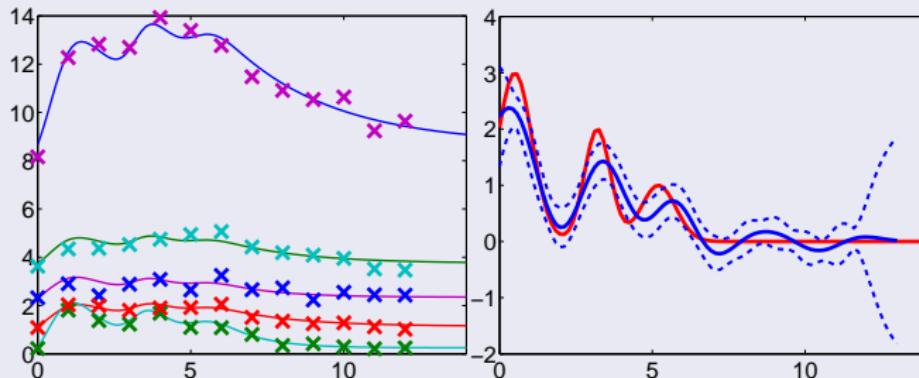


Figure: *Left:* The TFC,  $f(t)$ , which drives the system. *Middle:* Five gene mRNA concentration profiles each obtained by using different parameter sets  $\{B_i, S_i, D_i\}_{i=1}^5$  (lines) along with noise corrupted 'data'. *Right:* The inferred TFC (with error bars).

# Results

## Linear System

- Recently published biological data set studied using linear response model by Barenco et al. (2006).
- Study focused on the tumour suppressor protein p53.
- mRNA abundance measured for five targets: *DDB2*, *p21*, *SESN1/hPA26*, *BIK* and *TNFRSF10b*.
- Quadratic interpolation for the mRNA production rates to obtain gradients.
- They used MCMC sampling to obtain estimates of the model parameters  $B_j$ ,  $S_j$ ,  $D_j$  and  $f(t)$ .

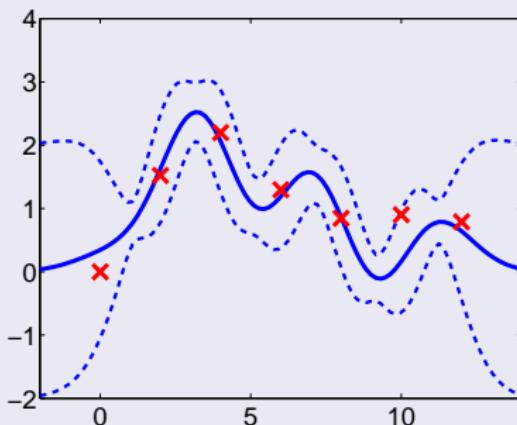
# Linear response analysis

## Experimental Setup

- We analysed data using the linear response model.
- Raw data was processed using the mmgMOS model of Liu et al. (2005) which provides variance as well as expression level.
- We present posterior distribution over TFCs.
- Results of inference on the values of the hyperparameters  $B_j$ ,  $S_j$  and  $D_j$ .
  - Samples from the posterior distribution were obtained using Hybrid Monte Carlo (see e.g. Neal, 1996).

# Linear Response Results

demBarenco1



**Figure:** Predicted protein concentration for p53. Solid line is mean, dashed lines 95% credibility intervals. The prediction of (Barenco et al., 2006) was pointwise and is shown as crosses.

# Results — Transcription Rates

## Estimation of Equation Parameters demBarenco1

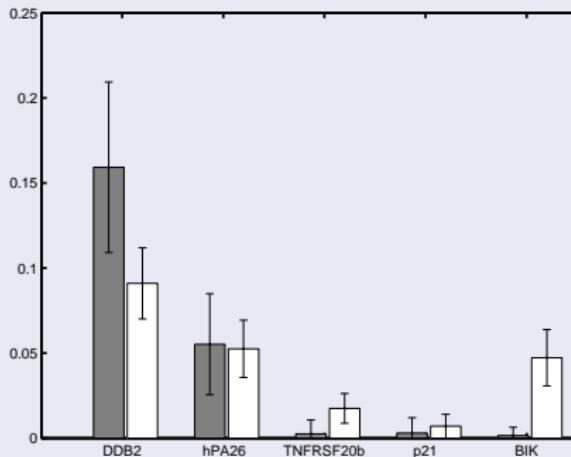


Figure: Basal transcription rates. Our results (black) compared with Barenco et al. (2006) (white).

# Results — Transcription Rates

## Estimation of Equation Parameters demBarenc01

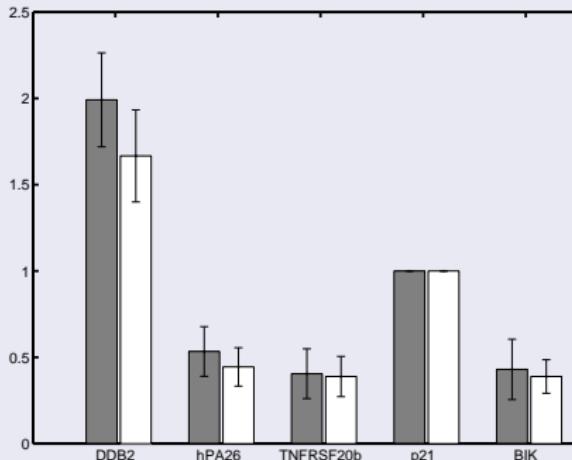


Figure: Sensitivities. Our results (black) compared with Barenco et al. (2006) (white).

# Results — Transcription Rates

## Estimation of Equation Parameters demBarenc01

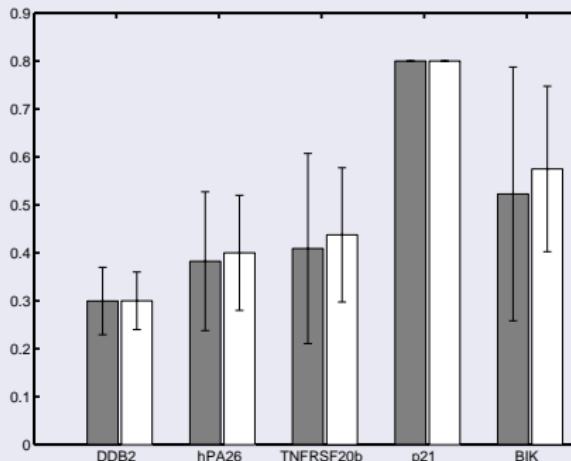


Figure: Decays. Our results (black) compared with Barenco et al. (2006) (white).

# Linear Response Discussion

## GP Results

- Note oscillatory behaviour, possible artifact of RBF covariance Rasmussen and Williams (see page 123 in 2006).
- Results are in good accordance with the results obtained by Barenco et al..
- Differences in estimates of the basal transcription rates probably due to:
  - different methods used for probe-level processing of the microarray data.
  - Our failure to constrain  $f(0) = 0$ .
- Our results take about 13 minutes to produce Barenco et al. required 10 million iterations of Monte Carlo.

# Non-linear Response Model

## More Realistic Response

- Transcription factor concentrations are positive, but direct samples from a GP will not be.
- Linear models don't account for saturation.
- *Solution:* model response using a positive nonlinear function.

# Formalism

## Non-linear Response

- Introduce a non-linearity  $g(\cdot)$  parameterised by  $\theta_j$

$$\frac{dx_j}{dt} = B_j + g(f(t), \theta_j) - D_j x_j$$

$$x_j(t) = \frac{B_j}{D_j} + \exp(-D_j t) \int_0^t du g(f(u), \theta_j) \exp(D_j u) .$$

- The induced distribution of  $x_j(t)$  is no longer a GP.
- Derive the functional gradient and learn a MAP solution for  $f(t)$ .
- Also compute Hessian so we can approximate the marginal likelihood.

## Example: linear response

### Using non-RBF kernels

- Start by taking  $g(\cdot)$  to be linear.
- Provides 'sanity check' and allows arbitrary covariance functions.
- Avoids double numerical integral that would normally be required.

# Response Results

demBarencoMap1, demBarencoMap2

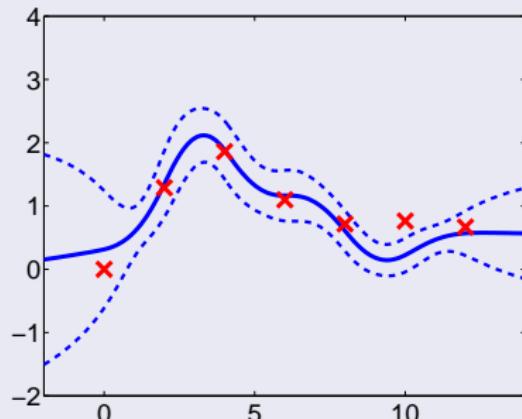
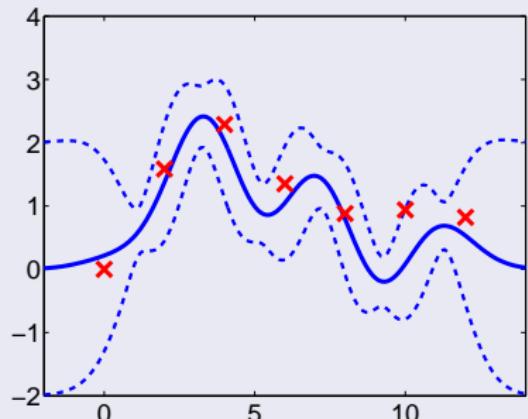


Figure: Left: RBF prior on  $f$  (log likelihood -101.4); Right: MLP prior on  $f$  (log likelihood -105.6).

# Non-linear response analysis

## Non-linear responses

- Exponential response model (constrains protein concentrations positive).
- $\log(1 + \exp(f))$  response model.
- $\frac{3}{1 + \exp(-f)}$
- Inferred MAP solutions for the latent function  $f$  are plotted below.

# $\exp(\cdot)$ Response Results

demBarencoMap3, demBarencoMap4

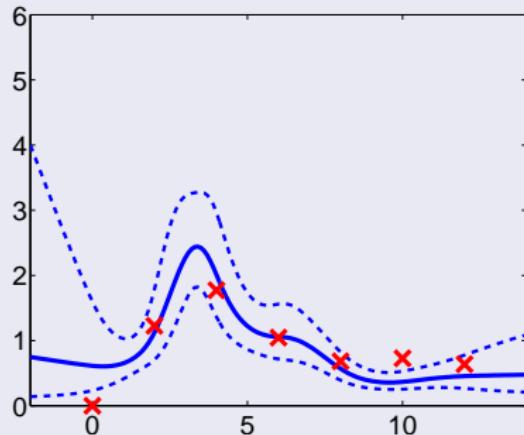
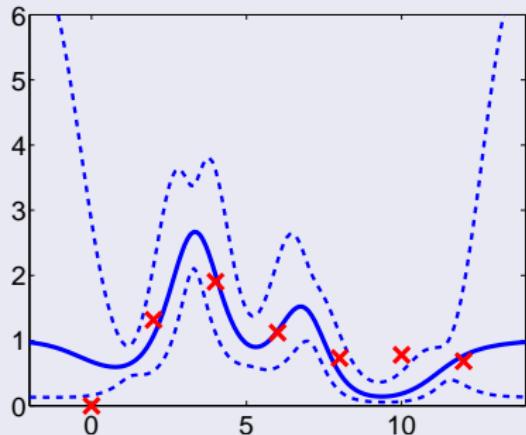


Figure: Left: shows results of using a squared exponential prior covariance on  $f$  (log likelihood -100.6); Right: shows results of using an MLP prior covariance on  $f$  (log likelihood -106.4).

# $\log(1 + \exp(f))$ Response Results

demBarencoMap5, demBarencoMap6

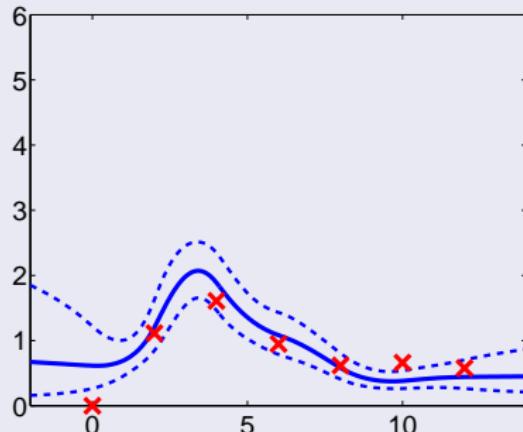
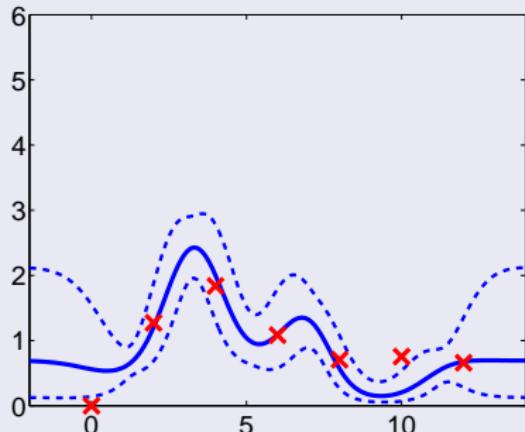


Figure: Left: shows results of using a squared exponential prior covariance on  $f$  (log likelihood -100.9); Right: shows results of using an MLP prior covariance on  $f$  (log likelihood -110.0).

# $\frac{3}{1+\exp(-f)}$ Response Results

## demBarencoMap7, demBarencoMap8

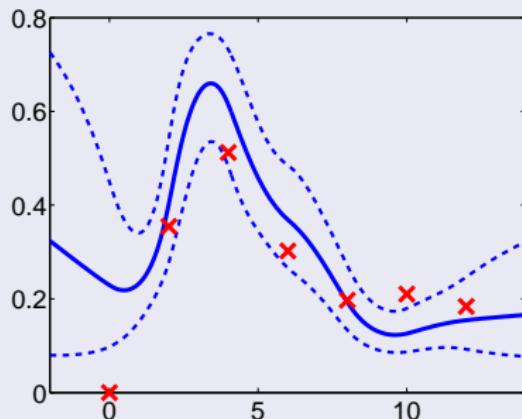
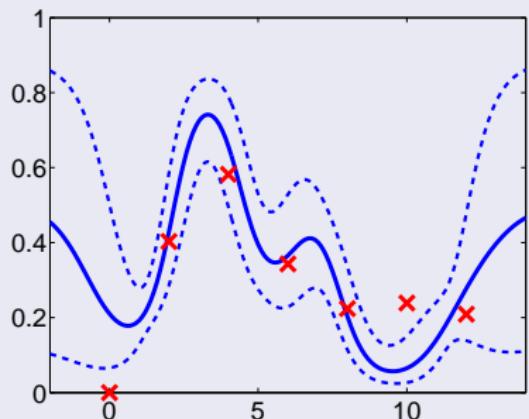


Figure: Left: shows results of using a squared exponential prior covariance on  $f$  (log likelihood -104.1); Right: shows results of using an MLP prior covariance on  $f$  (log likelihood -111.2).

# Transcription Model Summary

## Progress so far and Future work

- Elegant solution of a problem with indirect observations.
- Already extended to non-linear response equations (using Laplace approximation).
- Expect to extend it to systems with *multiple transcription factors*.

# Summary

## PUMA: Propagation of Uncertainty in Microarray Analysis

- Level of Noise in the Array can be Assesed (gMOS methods).
- Probabilistic Models can:
  - Improve selection of over-expressed genes (PPLR) — Appendix
  - Clean up gene expression profiles (NPPCA) — Appendix
- Simple (log-linear) probabilistic models can be used with network connectivity data to
  - To infer *genome wide* transcription factor activities (chipdyno).
  - To infer *genome wide* transcription factor protein 'concentrations' (chipvar).
- Gaussian processes & differential equations for complex interactions.
- And finally ...

# Acknowledgements

## Team:

- Principal Investigators
  - Neil Lawrence and Magnus Rattray
- gMOS family of Methods and PPLR
  - Xuejun Liu and Marta Milo
- Uncertainty Propagation through PCA
  - Marta Milo and Guido Sanguinetti
- Inference of Transcription Factor Activities
  - Guido Sanguinetti and **3 year RA Position Available!!**
- BBSRC award “Improved Processing of Microarray Data Using Probabilistic Models”.

# References

O. Alter and G. H. Golub. *Proceedings of the National Academy of Sciences USA*, 101(47):16577–16582, 2004.

M. Barenco, D. Tomescu, D. Brewer, R. Callard, J. Stark, and M. Hubank. *Genome Biology*, 7(3):R25, 2006.

A.-L. Boulesteix and K. Strimmer. *Theor. Biol. Med. Model.*, 2(23):1471–16582, 2005.

S. E. Choe, M. Boutros, A. M. Michelson, G. M. Church, and M. S. Halfon. *Genome Biology*, 6(R16), 2005.

F. Gao, B. C. Foat, and H. J. Bussemaker. *BMC Bioinformatics*, 5(31):1471–2105, 2004.

T. I. Lee, N. J. Rinaldi, F. Robert, D. T. Odom, Z. Bar-Joseph, G. K. Gerber, N. M. Hannett, C. T. Harbison, C. M. Thompson, I. Simon, J. Zeitlinger, E. G. Jennings, H. L. Murray, D. B. Gordon, B. Ren, J. J. Wyrick, J.-B. Tagne, T. L. Volkert, E. Fraenkel, D. K. Gifford, and R. A. Young. *Science*, 298(5594):799–804, 2002.

J. C. Liao, R. Boscolo, Y.-L. Yang, L. M. Tran, C. Sabatti, and V. P. Roychowdhury. *Proceedings of the National Academy of Sciences*

# Differential Gene Expression

## Probability of Positive Log Ratio(Liu et al., 2006)

- Differential gene expression is normally assessed with log ratios of gene expression.

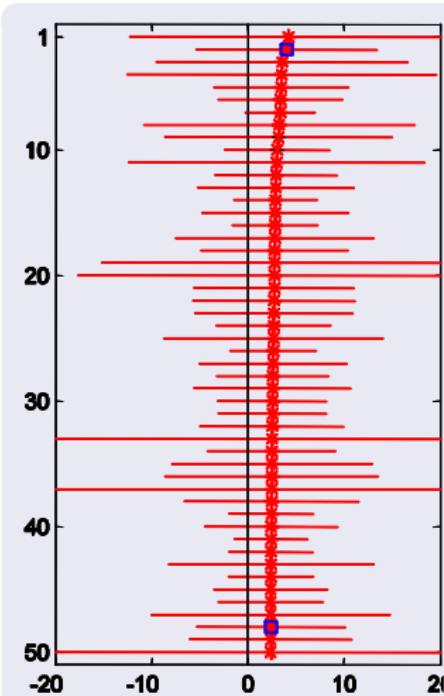
$$r_{ij} = \log \frac{s_i}{s_j}$$

- This measure is very sensitive to noise at low expression levels.
- Use variance of expression to obtain Probability of Positive Log Ratio (PPLR).

# PPLR Results

Golden spike-in dataset (Choe et al., 2005)

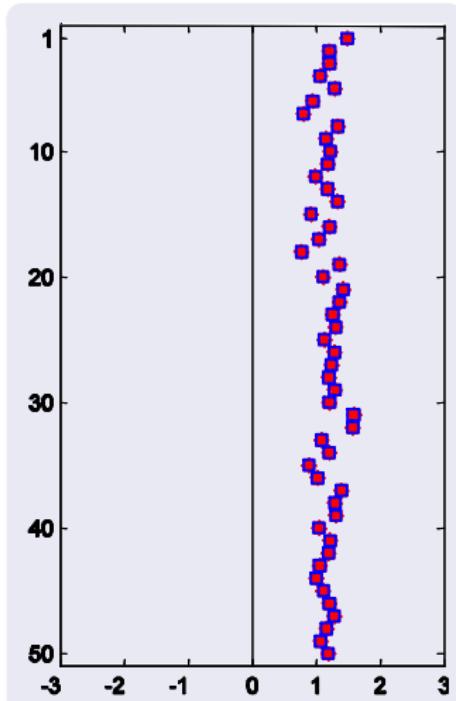
- Ranking (y-axis) against log ratio (x-axis) for.
  - Ranking by Expected Log Ratio.**
  - Ranking by PPLR.
- Red stars indicate expected log ratio.
- Red lines indicate error bars.
- Blue squares indicates genes that were spiked-in.



# PPLR Results

Golden spike-in dataset (Choe et al., 2005)

- Ranking (y-axis) against log ratio (x-axis) for:
  - Ranking by Expected Log Ratio.
  - Ranking by PPLR.
- Red stars indicate expected log ratio.
- Red lines indicate error bars.
- Blue squares indicates genes that were spiked-in.



# Cleaning up Profiles

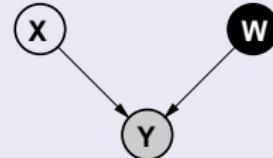
## Converting Noisy Profiles to Clean

- If we can 'clean up' the profiles we can use in other methods.
- Construct a probabilistic model for the data and corruption process.
- Work with posterior distribution over cleaned up profile.
- We designed a heteroschedastic Probabilistic PCA for doing this (Sanguinetti et al., 2005).

# Probabilistic PCA

## Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- Latent variable approach:
  - Define Gaussian prior over *latent space*,  $\mathbf{X}$ .
  - Integrate out *latent variables*.



$$p(\mathbf{Y}|\mathbf{X}, \mathbf{W}) = \prod_{i=1}^n N(\mathbf{y}_{i,:} | \mathbf{W}\mathbf{x}_{i,:} + \boldsymbol{\mu}, \sigma^2 \mathbf{I})$$

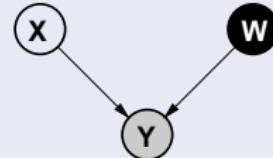
$$p(\mathbf{X}) = \prod_{i=1}^n N(\mathbf{x}_{i,:} | \mathbf{0}, \mathbf{I})$$

$$p(\mathbf{Y}|\mathbf{W}) = \prod_{i=1}^n N(\mathbf{y}_{i,:} | \boldsymbol{\mu}, \mathbf{W}\mathbf{W}^T + \sigma^2 \mathbf{I})$$

# Probabilistic PCA

## Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- Latent variable approach:
  - Define Gaussian prior over *latent space*,  $\mathbf{X}$ .
  - Integrate out *latent variables*.



$$p(\mathbf{Y}|\mathbf{X}, \mathbf{W}) = \prod_{i=1}^n N(\mathbf{y}_{i,:} | \mathbf{W}\mathbf{x}_{i,:} + \boldsymbol{\mu}, \sigma^2 \mathbf{I})$$

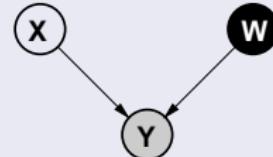
$$p(\mathbf{X}) = \prod_{i=1}^n N(\mathbf{x}_{i,:} | \mathbf{0}, \mathbf{I})$$

$$p(\mathbf{Y}|\mathbf{W}) = \prod_{i=1}^n N(\mathbf{y}_{i,:} | \boldsymbol{\mu}, \mathbf{W}\mathbf{W}^T + \sigma^2 \mathbf{I})$$

# Probabilistic PCA

## Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- Latent variable approach:
  - Define Gaussian prior over *latent space*,  $\mathbf{X}$ .
  - Integrate out *latent variables*.



$$p(\mathbf{Y}|\mathbf{X}, \mathbf{W}) = \prod_{i=1}^n N(\mathbf{y}_{i,:} | \mathbf{W}\mathbf{x}_{i,:} + \boldsymbol{\mu}, \sigma^2 \mathbf{I})$$

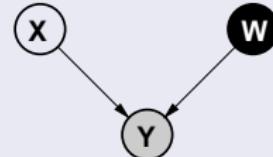
$$p(\mathbf{X}) = \prod_{i=1}^n N(\mathbf{x}_{i,:} | \mathbf{0}, \mathbf{I})$$

$$p(\mathbf{Y}|\mathbf{W}) = \prod_{i=1}^n N(\mathbf{y}_{i,:} | \boldsymbol{\mu}, \mathbf{W}\mathbf{W}^T + \sigma^2 \mathbf{I})$$

# Probabilistic PCA

## Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- Latent variable approach:
  - Define Gaussian prior over *latent space*,  $\mathbf{X}$ .
  - Integrate out *latent variables*.



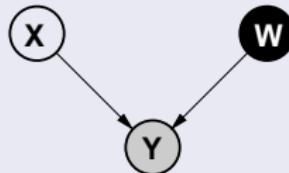
$$p(\mathbf{Y}|\mathbf{X}, \mathbf{W}) = \prod_{i=1}^n N(\mathbf{y}_{i,:} | \mathbf{W}\mathbf{x}_{i,:} + \boldsymbol{\mu}, \sigma^2 \mathbf{I})$$

$$p(\mathbf{X}) = \prod_{i=1}^n N(\mathbf{x}_{i,:} | \mathbf{0}, \mathbf{I})$$

$$p(\mathbf{Y}|\mathbf{W}) = \prod_{i=1}^n N(\mathbf{y}_{i,:} | \boldsymbol{\mu}, \mathbf{W}\mathbf{W}^T + \sigma^2 \mathbf{I})$$

## Probabilistic PCA II

## Probabilistic PCA Max. Likelihood Soln (Tipping and Bishop, 1999)



$$p(\mathbf{Y}|\mathbf{W}) = \prod_{i=1}^n N(\mathbf{y}_{i,:}|\boldsymbol{\mu}, \mathbf{W}\mathbf{W}^T + \sigma^2\mathbf{I})$$

## Probabilistic PCA II

## Probabilistic PCA Max. Likelihood Soln (Tipping and Bishop, 1999)

$$p(\mathbf{Y}|\mathbf{W}) = \prod_{i=1}^n N(\mathbf{y}_{i,:}|\boldsymbol{\mu}, \mathbf{C}), \quad \mathbf{C} = \mathbf{W}\mathbf{W}^T + \sigma^2\mathbf{I}$$

$$\log p(\mathbf{Y}|\mathbf{W}) = -\frac{n}{2} \log |\mathbf{C}| - \frac{1}{2} \text{tr}(\mathbf{C}^{-1} \tilde{\mathbf{Y}}^T \tilde{\mathbf{Y}}) + \text{const.}$$

Where  $\tilde{\mathbf{Y}}$  is the matrix  $\mathbf{Y}$  with  $\boldsymbol{\mu}$  removed. If  $\mathbf{U}_q$  are first  $q$  principal eigenvectors of  $n^{-1} \tilde{\mathbf{Y}}^T \tilde{\mathbf{Y}}$  and the corresponding eigenvalues are  $\Lambda_q$ ,

$$\mathbf{W} = \mathbf{U}_q \mathbf{L} \mathbf{V}^T, \quad \mathbf{L} = (\Lambda_q - \sigma^2 \mathbf{I})^{\frac{1}{2}}$$

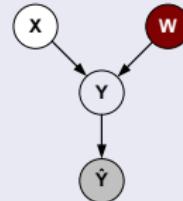
where  $\mathbf{V}$  is an arbitrary rotation matrix.

$$\boldsymbol{\mu} = n^{-1} \sum_{i=1}^n \mathbf{y}_{i,:}$$

# Heteroschedastic Probabilistic PCA

## Heteroschedastic PPCA

- Define *linear-Gaussian relationship* between latent variables and  $\mathbf{Y}$ .
- Define a *further Gaussian relationship* to corrupted profiles  $\hat{\mathbf{Y}}$ .
  - $\mathbf{D}_i$  is a diagonal matrix of estimated variances.
- Integrate out *latent variables*.



$$p(\mathbf{Y}|\mathbf{X}, \mathbf{W}) = \prod_{i=1}^n N(\mathbf{y}_{i,:} | \mathbf{W}\mathbf{x}_{i,:} + \boldsymbol{\mu}, \sigma^2 \mathbf{I})$$

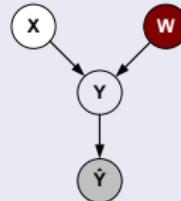
$$p(\hat{\mathbf{y}}_{i,:} | \mathbf{y}_{i,:}) = N(\hat{\mathbf{y}}_{i,:} | \mathbf{y}_{i,:}, \mathbf{D}_i)$$

$$p(\hat{\mathbf{Y}}|\mathbf{W}) = \prod_{i=1}^n N(\mathbf{y}_{i,:} | \boldsymbol{\mu}, \mathbf{W}\mathbf{W}^T + \sigma^2 \mathbf{I} + \mathbf{D}_i)$$

# Heteroschedastic Probabilistic PCA

## Heteroschedastic PPCA

- Define *linear-Gaussian relationship* between latent variables and  $\mathbf{Y}$ .
- Define a *further Gaussian relationship* to corrupted profiles  $\hat{\mathbf{Y}}$ .
  - $\mathbf{D}_i$  is a diagonal matrix of estimated variances.
- Integrate out *latent variables*.



$$p(\mathbf{Y}|\mathbf{X}, \mathbf{W}) = \prod_{i=1}^n N(\mathbf{y}_{i,:} | \mathbf{W}\mathbf{x}_{i,:} + \boldsymbol{\mu}, \sigma^2 \mathbf{I})$$

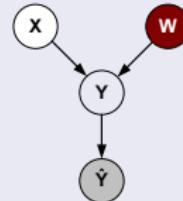
$$p(\hat{\mathbf{y}}_{i,:} | \mathbf{y}_{i,:}) = N(\hat{\mathbf{y}}_{i,:} | \mathbf{y}_{i,:}, \mathbf{D}_i)$$

$$p(\hat{\mathbf{Y}}|\mathbf{W}) = \prod_{i=1}^n N(\mathbf{y}_{i,:} | \boldsymbol{\mu}, \mathbf{W}\mathbf{W}^T + \sigma^2 \mathbf{I} + \mathbf{D}_i)$$

# Heteroschedastic Probabilistic PCA

## Heteroschedastic PPCA

- Define *linear-Gaussian relationship* between latent variables and  $\mathbf{Y}$ .
- Define a *further Gaussian relationship* to corrupted profiles  $\hat{\mathbf{Y}}$ .
  - $\mathbf{D}_i$  is a diagonal matrix of estimated variances.
- Integrate out *latent variables*.



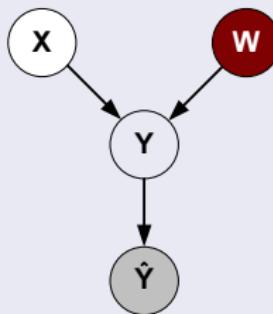
$$p(\mathbf{Y}|\mathbf{X}, \mathbf{W}) = \prod_{i=1}^n N(\mathbf{y}_{i,:} | \mathbf{W}\mathbf{x}_{i,:} + \boldsymbol{\mu}, \sigma^2 \mathbf{I})$$

$$p(\hat{\mathbf{y}}_{i,:} | \mathbf{y}_{i,:}) = N(\hat{\mathbf{y}}_{i,:} | \mathbf{y}_{i,:}, \mathbf{D}_i)$$

$$p(\hat{\mathbf{Y}}|\mathbf{W}) = \prod_{i=1}^n N(\mathbf{y}_{i,:} | \boldsymbol{\mu}, \mathbf{W}\mathbf{W}^T + \sigma^2 \mathbf{I} + \mathbf{D}_i)$$

## Heteroschedastic PPCA II

## Heteroschedastic PPCA Max. Likelihood Soln (Sanguinetti et al., 2005)



$$p(\mathbf{Y}|\mathbf{W}) = \prod_{i=1}^n N \left( \mathbf{y}_{i,:} | \boldsymbol{\mu}, \mathbf{W}\mathbf{W}^T + \sigma^2 \mathbf{I} + \mathbf{D}_i \right)$$

## Heteroschedastic PPCA II

## Heteroschedastic PPCA Max. Likelihood Soln (Sanguinetti et al., 2005)

$$p(\mathbf{Y}|\mathbf{W}) = \prod_{i=1}^n N\left(\mathbf{y}_{i,:}|\boldsymbol{\mu}, \mathbf{W}\mathbf{W}^T + \sigma^2\mathbf{I} + \mathbf{D}_i\right)$$

- Can no longer solve via eigenvalue problem.
- We use an EM algorithm.
  - A major problem is the strong correlation between  $\mathbf{W}$  and  $\boldsymbol{\mu}$ .
  - We use some tricks to speed up convergence.
- Software available in R and MATLAB.

# Heteroschedastic PPCA Results

## Mouse Cochlear Dataset

- Data from a conditionally immortal cell line extracted from mouse cochlear epithelial cells.
- Twelve samples from 14 days of differentiation after extraction at E13.5 (Rivolta et al., 2002).
- Experimental setup:
  - Perform HPPCA/PCA on the data.
  - Extract 50 genes most associated with 2nd principal component
  - Cluster original profiles and reconstructed profiles.

## Heteroschedastic PPCA Results

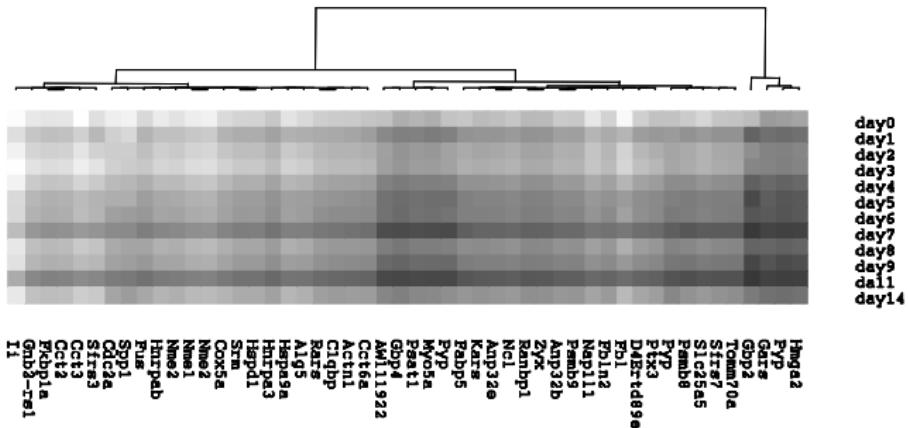


Figure: Hierarchical Clustering on Corrected Profiles.

## Heteroschedastic PPCA Results

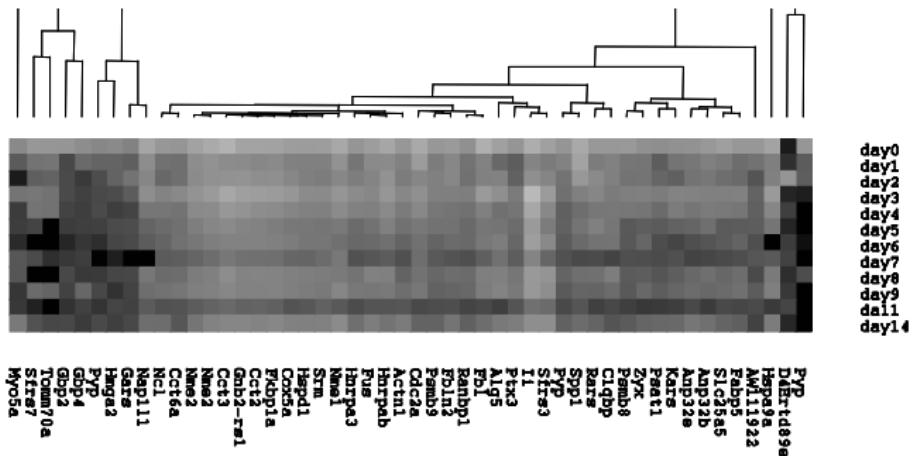


Figure: Hierarchical Clustering on Uncorrected Profiles.

# Heteroschedastic PPCA Results

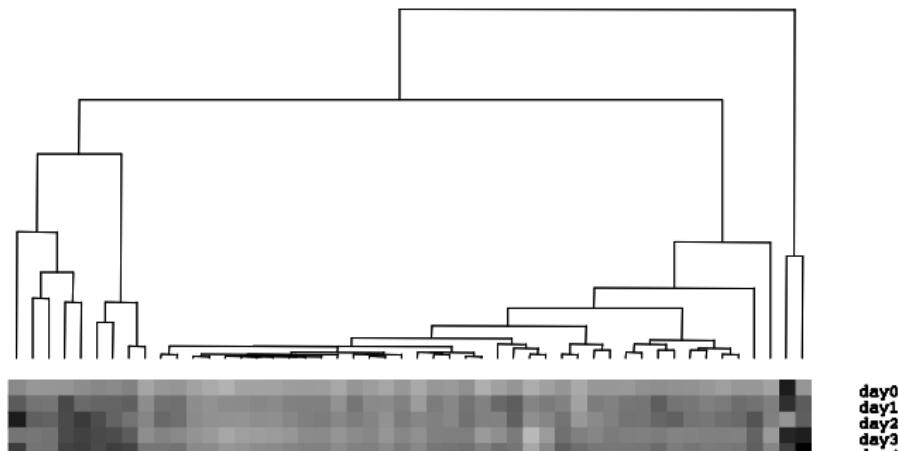


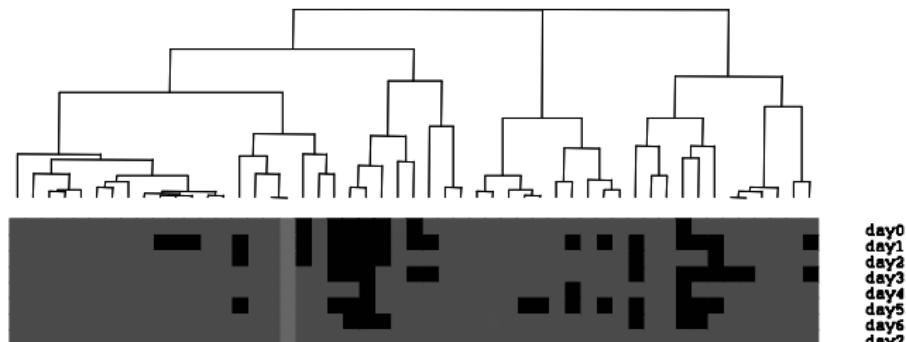
Figure: Hierarchical Clustering on Uncorrected Profiles.

## Heteroschedastic PPCA Results



Figure: Hierarchical Clustering on genes selected by normal PCA.

# Heteroschedastic PPCA Results



**Figure:** Hierarchical Clustering on genes selected by normal PCA.