Probabilistic Principal Surfaces For Yeast Gene Microarray Data Mining

Roberto Tagliaferri

Department of Mathematics and Informatics

University of Salerno, Italy

rtagliaferri@unisa.it

AstroNeural Collaboration

Astroneural
DMI - Università di Salerno
DSF - Università Federico II Napoli

- Napoli
  - Department of Physical Sciences
    - **Roberto Amato**
    - **Carmine Del Mondo**
    - **Natalia Deniskina**
    - **Ciro Donalek**
    - **Giuseppe Longo**
    - **Gennaro Miele**
  - Telethon Institute for Genetics and Medicine
    - **Diego Di Bernardo**
- Salerno
  - Department of Mathematics and Informatics
    - **Angelo Ciaramella**
    - **Giancarlo Raiconi**
    - **Antonino Staiano**
    - **Roberto Tagliaferri**

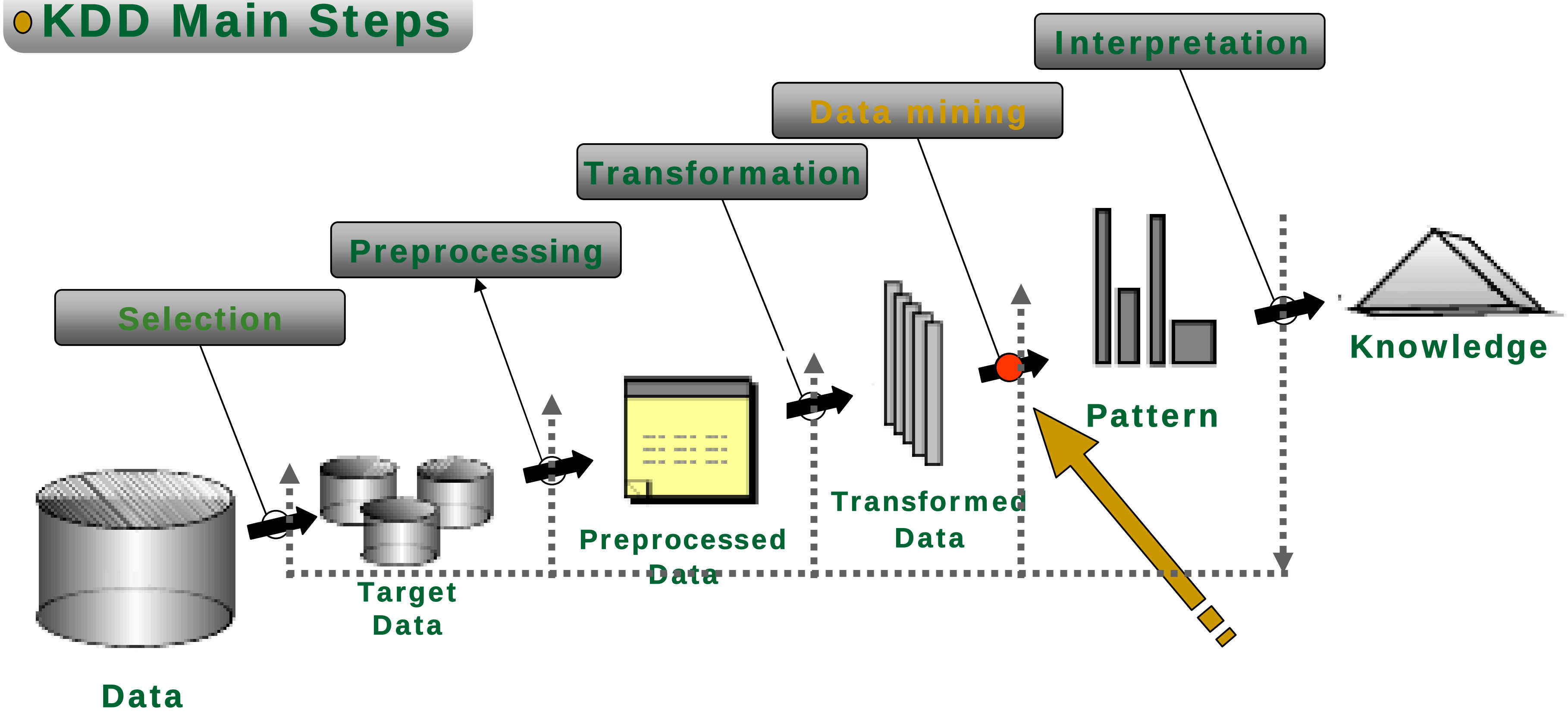
Outline

- Introduction
  - Knowledge Discovery in Databases
  - Data Mining
- Latent Variable Models
  - Probabilistic Principal Surfaces
    - ✓ Spherical PPS
- PPS and Data Mining
  - PPS for high-D data visualization
- A case study: Yeast gene microarray data
  - Preprocessing: Noise Estimation Method and Nonlinear PCA
  - Clustering: Neg-Entropy based algorithm
- Conclusions

Introduction

Knowledge Discovery in Databases (KDD)

- KDD Main Steps



Process involved in whatever data-rich field aimed to extract meaningful information from data

Introduction

Data Mining

- *Data Mining is a key step in KDD process aimed to find meaningful patterns in the data.*
- Data Mining Methods
  - Regression
  - Classification
  - Clustering
  - Data Visualization

Latent Variable Models

- Latent variable models are probabilistic models which generate a probability density function underlying a set of data in a multidimensional input space.

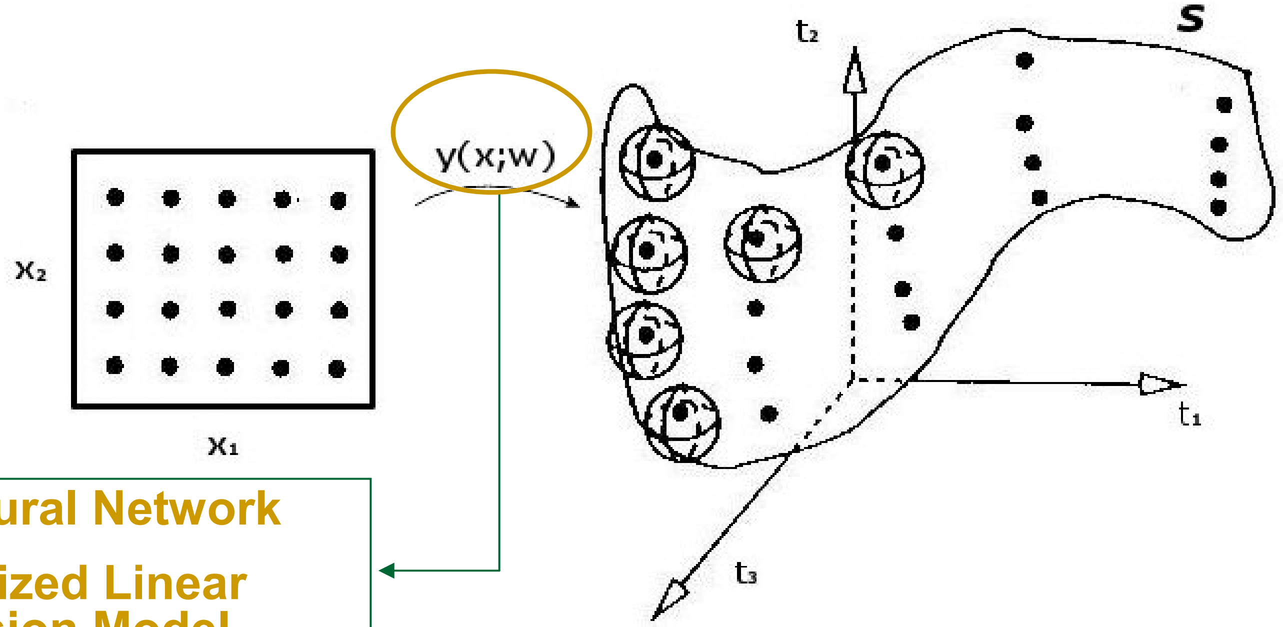
- The probability density function is a mixture of Gaussian expressed in terms of a smaller number of latent variables lying in another space called "*latent space*".

- The latent space is usually 2 or 3 dimensional, therefore, by using the Bayes theorem, one can derive the input data density function in the latent space.

- Hence, the data itself and the density function can be visualized in the latent space.

Latent Variable Models

- **Goal**: to express the distribution $p(\boldsymbol{t})$ of the variable $\boldsymbol{t}=(t_1,\ldots,t_D)$ in terms of a smaller number of latent variables $\boldsymbol{x}=(x_1,\ldots,x_Q)$, $Q<D$. The link between the latent and data spaces is obtained by the nonlinear function y(x,w).



- **RBF Neural Network**
- **Generalized Linear Regression Model**

Probabilistic Principal Surfaces

- Nonlinear latent variable model in which a mixture of Gaussians in the input space is built

$$p(\mathbf{t} \mid \mathbf{W}, \boldsymbol{\Sigma}) = \frac{1}{M} \sum_{m=1}^{M} p(\mathbf{t} \mid \mathbf{x}, \mathbf{W}, \boldsymbol{\Sigma})$$

Each mixture component is a Gaussian Distribution with mean y(**x**,**W**) and covariance Σ:

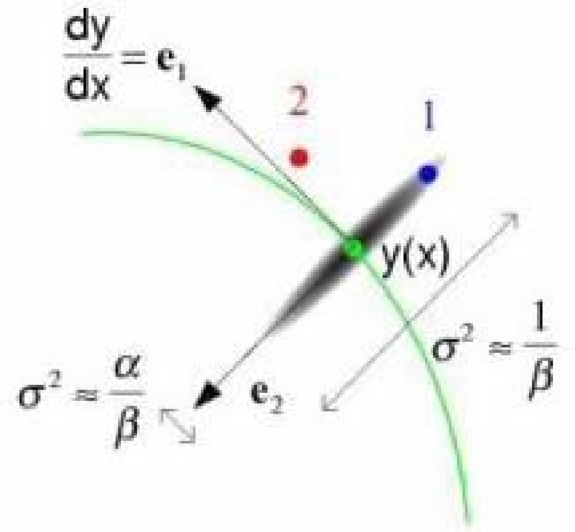
$$p(\mathbf{t} \mid \mathbf{x}) = \frac{1}{(2\pi)^{D/2} \left|\boldsymbol{\Sigma}\right|^{1/2}} \exp\left(-\frac{1}{2} (\mathbf{t} - \mathbf{y}(\mathbf{x}, \mathbf{W}))^T \boldsymbol{\Sigma}^{-1} (\mathbf{t} - \mathbf{y}(\mathbf{x}, \mathbf{W})) \right)$$

Probabilistic Principal Surfaces

- The covariance has the following form

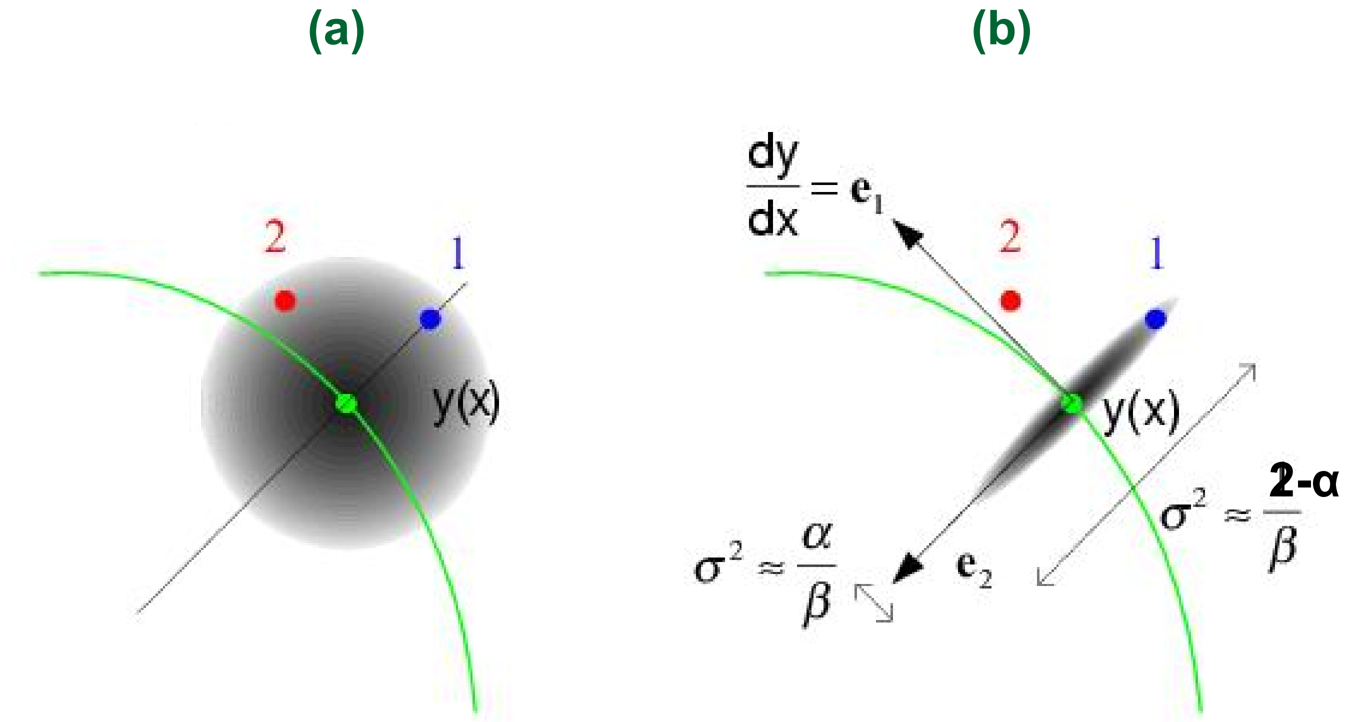
$$\Sigma(\mathbf{x}) = \frac{\alpha}{\beta}\sum_{q=1}^{Q}\mathbf{e}_q(\mathbf{x})\mathbf{e}_q^T(\mathbf{x}) + \frac{(D-\alpha Q)}{\beta(D-Q)}\sum_{d=Q+1}^{D}\mathbf{e}_d(\mathbf{x})\mathbf{e}_d^T(\mathbf{x})$$

- $\{\mathbf{e}_q(\mathbf{x})\}_{q=1,\ldots,Q}$ set of orthonormal vectors tangential to the manifold at $y(\mathbf{x};\mathbf{W})$
- $\{\mathbf{e}_d(\mathbf{x})\}_{d=Q+1,\ldots,D}$ set of orthonormal vectors orthogonal to the manifold at $y(\mathbf{x};\mathbf{W})$
- *α is called clamping factor* $0<\alpha<D/Q$



$\frac{dy}{dx} = \mathbf{e}_1$
2
1
y(x)
$\sigma^2 \approx \frac{1}{\beta}$
$\sigma^2 \approx \frac{\alpha}{\beta}$
$\mathbf{e}_2$

Probabilistic Principal Surfaces



Under a spherical Gaussian model, points ***1*** and ***2*** have equal influence on the center node *y(**x**)* (a) PPS have an oriented covariance matrix so point ***1*** is probabilistically closer to the center node *y(**x**)* than point ***2*** (b)

Probabilistic Principal Surfaces

- Based on a generalized EM for parameters W, α, β
- In practice, however, α is kept fixed, and only W and β are computed
- Computationally complex but fast convergence

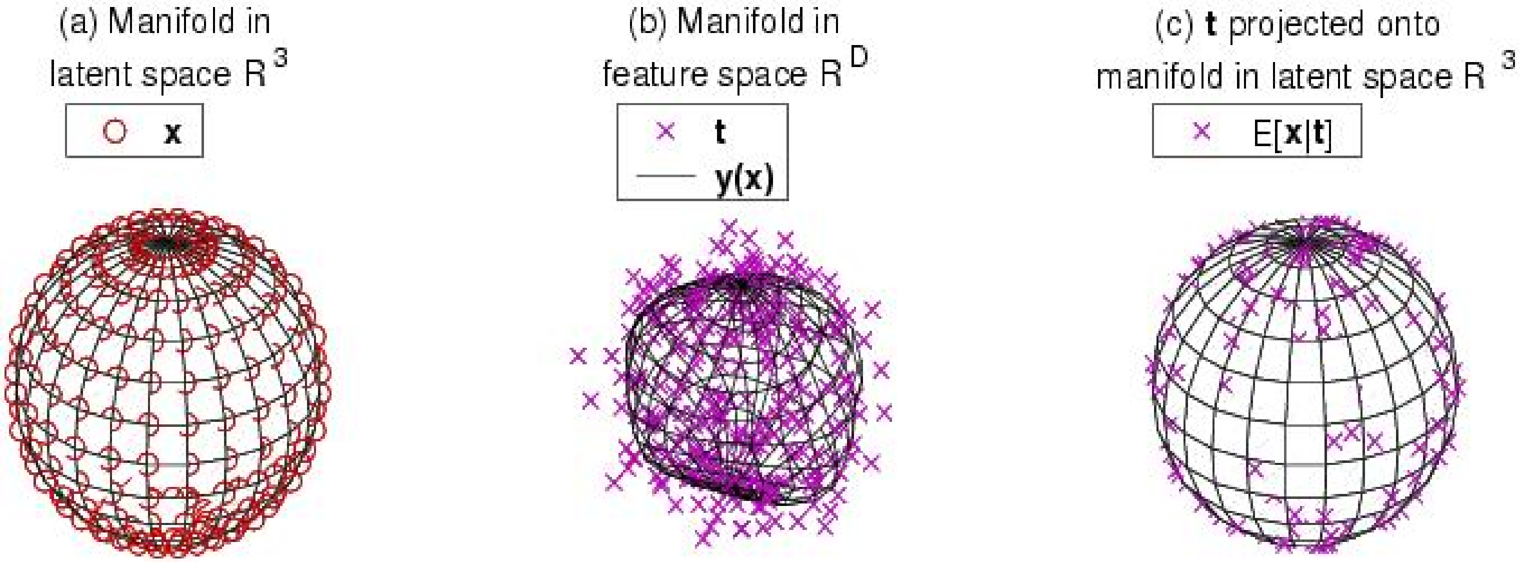
Probabilistic Principal Surfaces

Spherical PPS

- Manifold composed by nodes regularly arranged on the surface of a sphere in *3D* space (*Q=3*)
- Use manifold as a classification reference template
- Use projections for visualizations

Probabilistic Principal Surfaces

Spherical PPS



(a) The spherical manifold in R^3 latent space.
(b) The spherical manifold in R^3 data space.
(c) Projection of data point **t** onto the latent spherical manifold.

PPS & Data Mining

Spherical PPS for visualization

- *Probabilistic Projection*: the projected latent coordinate is computed as a linear combination of all latent nodes weighted by the responsibility matrix,

$$\mathbf{x}_n^{proj} \equiv \langle \mathbf{x} \mid \mathbf{t}_n \rangle = \int \mathbf{x} p(\mathbf{x} \mid \mathbf{t}_n) dx = \sum_{m=1}^{M} r_{mn} \mathbf{x}_m$$

- Since $||\boldsymbol{x}_m||=1$ for $m=1,\ldots,M$ and $\Sigma_m r_{mn}=1$ for $n=1,\ldots,N$, all projections lie within the sphere, i.e. $||\boldsymbol{x}_m||\leq 1$ *and*

- r_{mn} is the responsibility of latent variable $\boldsymbol{x}_m$ with respect to data point $\boldsymbol{t}_n$

$$p(\mathbf{x}_m \mid \mathbf{t}_n) = \frac{p(\mathbf{t}_n \mid \mathbf{x}_m, \mathbf{W}, \beta) p(\mathbf{x}_m)}{\sum_{m'=1}^{M} p(\mathbf{t}_n \mid \mathbf{x}_{m'}, \mathbf{W}, \beta) p(\mathbf{x}_{m'})}$$

Case Study

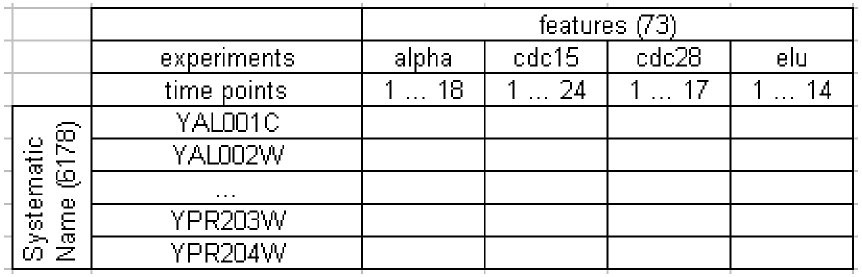
Yeast Gene Microarray Data

- P. T. Spellman et al., **Comprehensive Identification of Cell Cycle-regulated Genes of the Yeast *Saccharomyces cerevisiae* by Microarray Hybridization,** Molecular Biology of the Cell, Vol. 9, 3273-3297, December, 1998
- 6178 genes each one subject to 6 experiments:
  - cln3
  - clb2
  - alpha factor arrest
  - cdc15 temperature-sensitive mutant
  - cdc28
  - elutriation
- 73 features associate to each gene. After a preprocessing phase the features were reduced to 32.

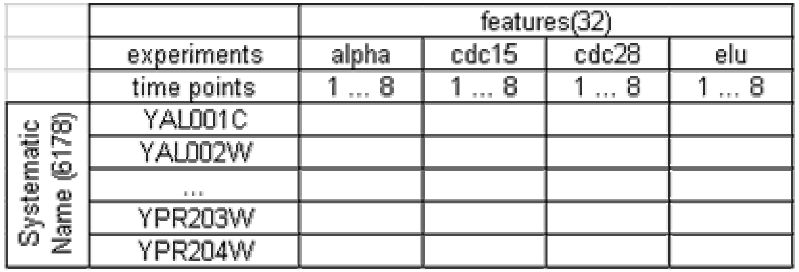
Case Study

Computational Steps

1. PREPROCESSING: Noise Estimation Method and Nonlinear PCA

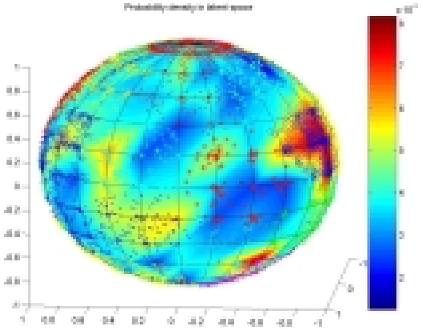


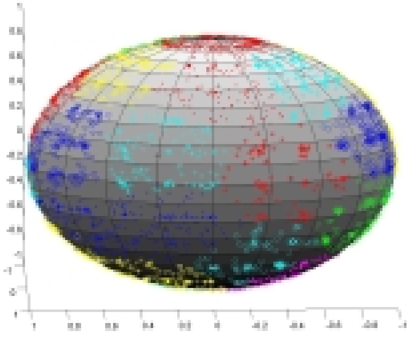
		features (73)			
	experiments	alpha	cdc15	cdc28	elu
	time points	1 … 18	1 … 24	1 … 17	1 … 14
Systematic Name (6178)	YAL001C				
	YAL002W				
	…				
	YPR203W				
	YPR204W				



		features(32)			
	experiments	alpha	cdc15	cdc28	elu
	time points	1 … 8	1 … 8	1 … 8	1 … 8
Systematic Name (6178)	YAL001C				
	YAL002W				
	…				
	YPR203W				
	YPR204W				

2. DATA MINING: 3D Spherical PPS and Clustering





Case Study

Gene Noise Estimation Method

The genes behaviour is periodic. The period is the cell cycle.

This implies that a gene behaviour, sampled for two cell cycles, can be considered as two measurements of the same thing.

This can be used to obtain an estimation for the uncertainty of the measurement.

Case Study

Gene Noise Estimation Method

Cell cycle duration, i.e. period, depends on some parameters such as temperature, nutrient source, density of cells and so on (for our experiments, periods were in the limits 90 ± 11 min).

To find the exact period length of each experiment we divided the gene time series in two parts and searched for (moving the cutting point in the interval 90 ± 11) the point of best correlation between the two parts.

Case Study

Gene Noise Estimation Method

Once obtained the period length, we have computed the noise/signal ratio of each gene, considering:

the difference between the two periods of each gene as an estimation of its noise;

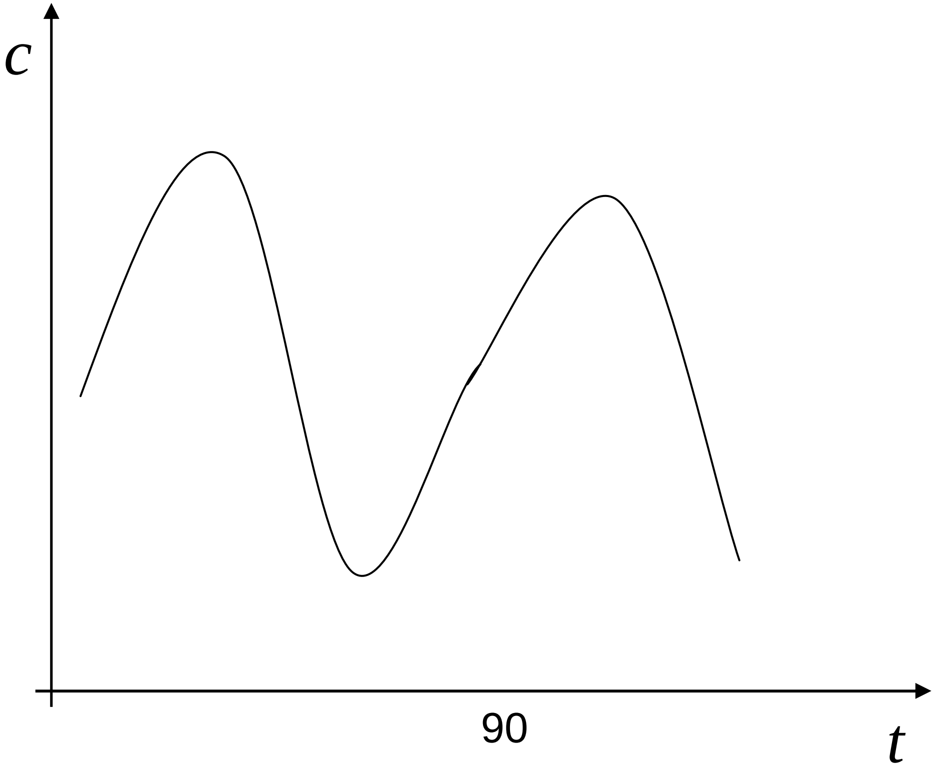
the mean of the two periods as the “real” signal of the gene.

This value was used to exclude too noisy genes.

This estimation is accomplished independently for each experiment.

Case Study

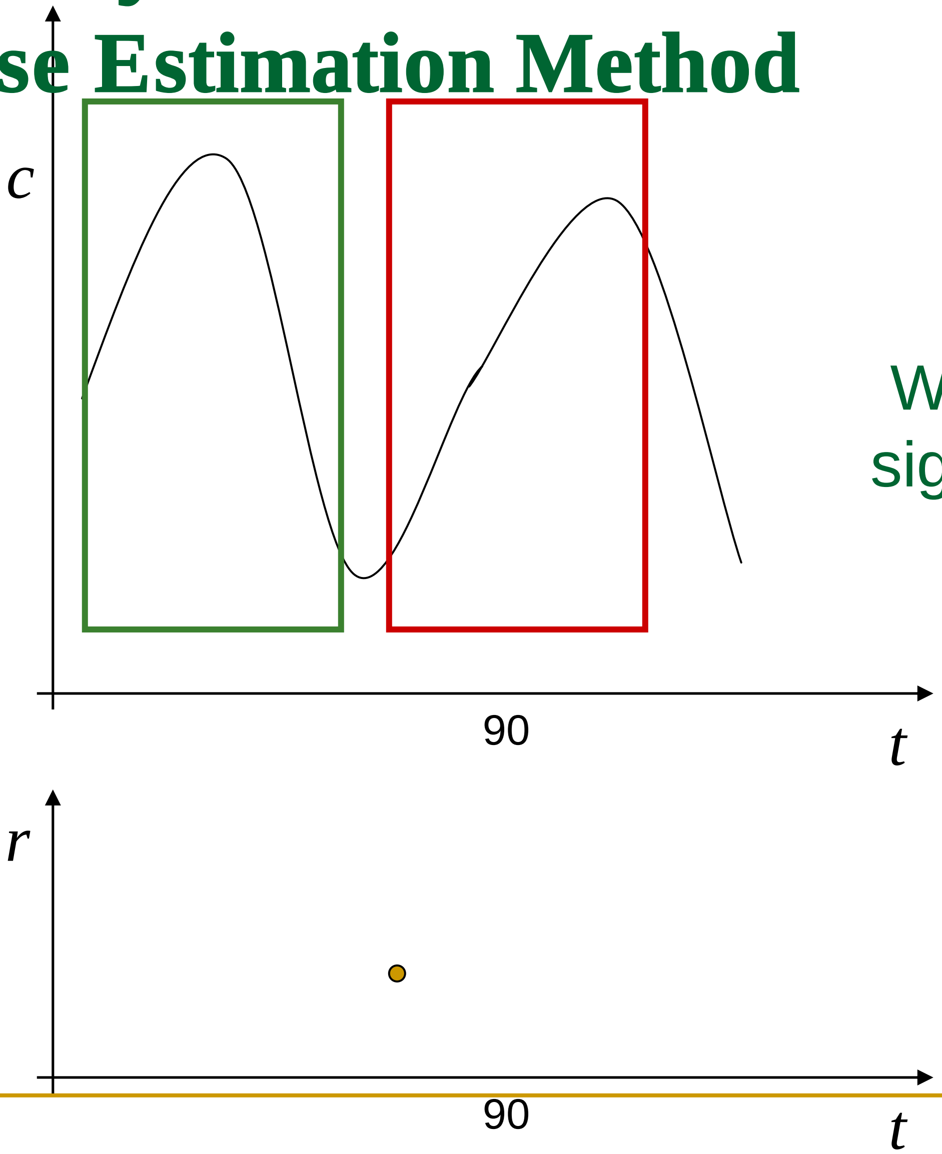
Gene Noise Estimation Method



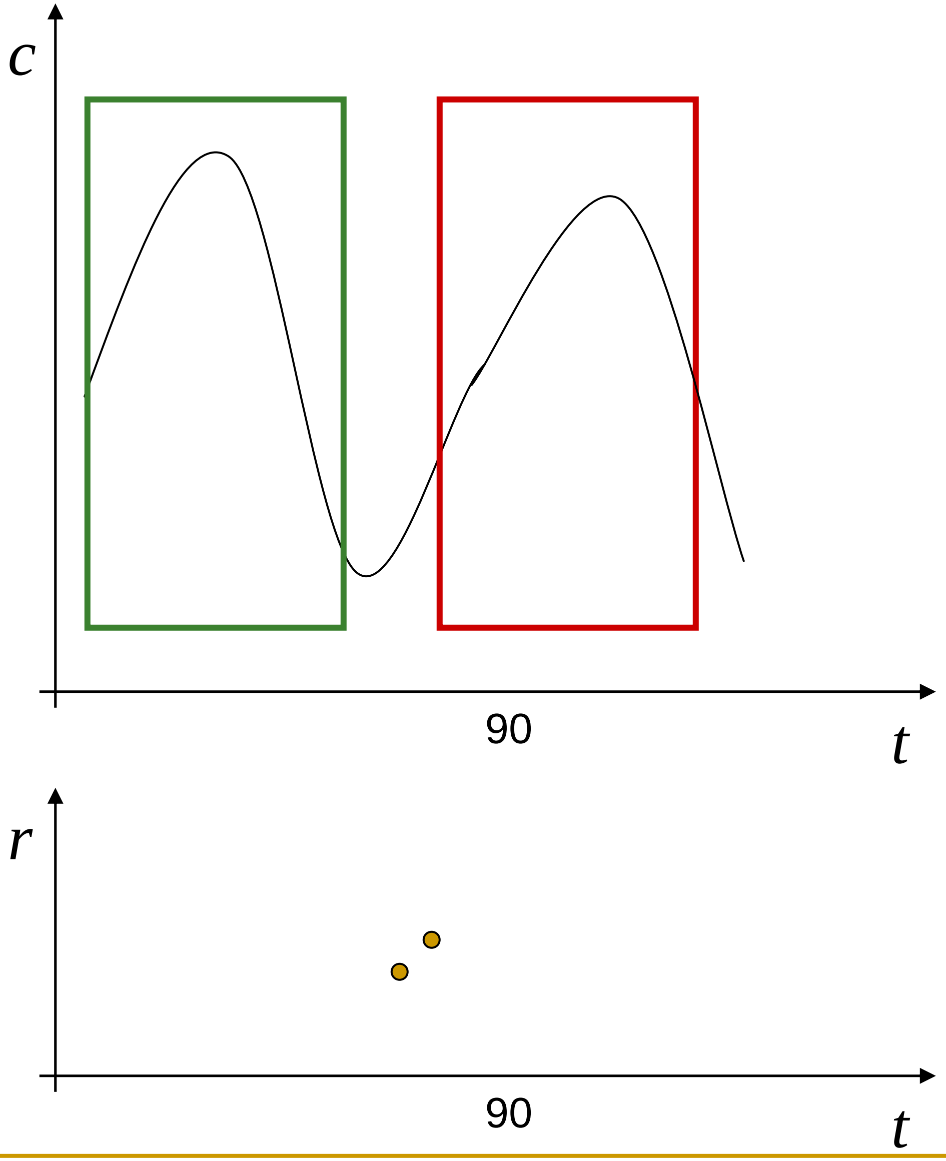
Consider a generic gene signal over an experiment

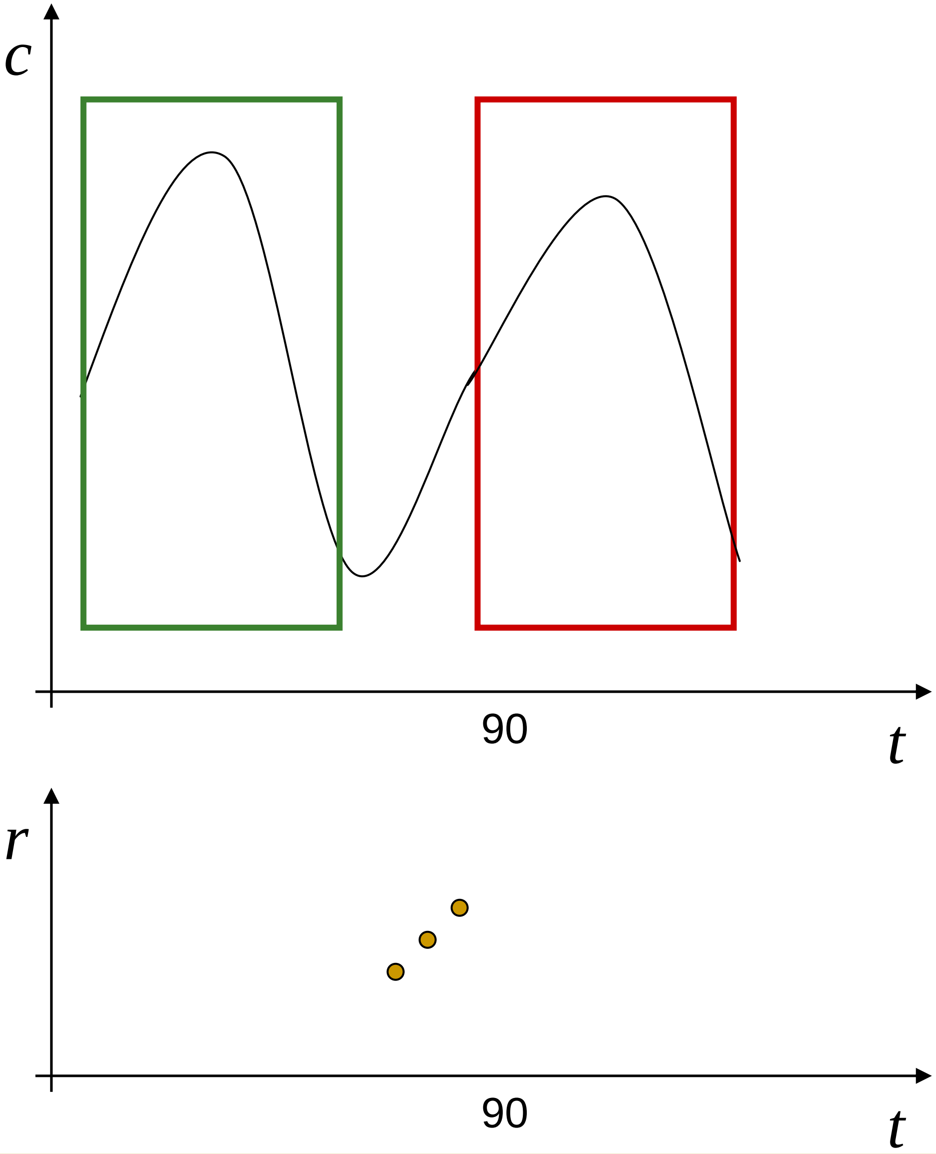
Case Study

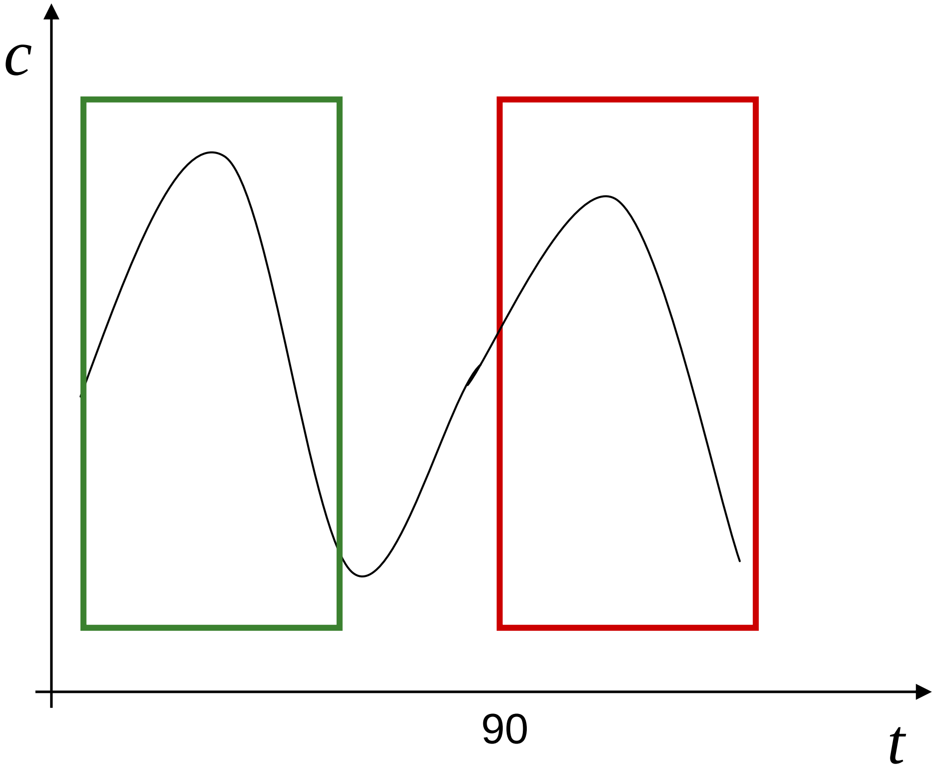
Gene Noise Estimation Method

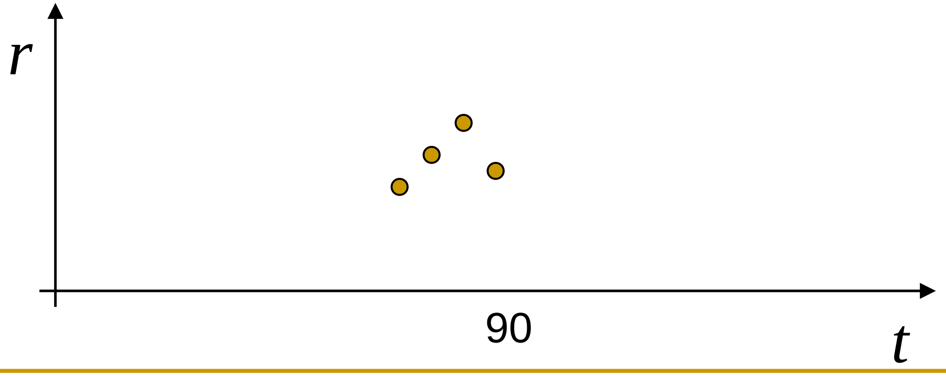


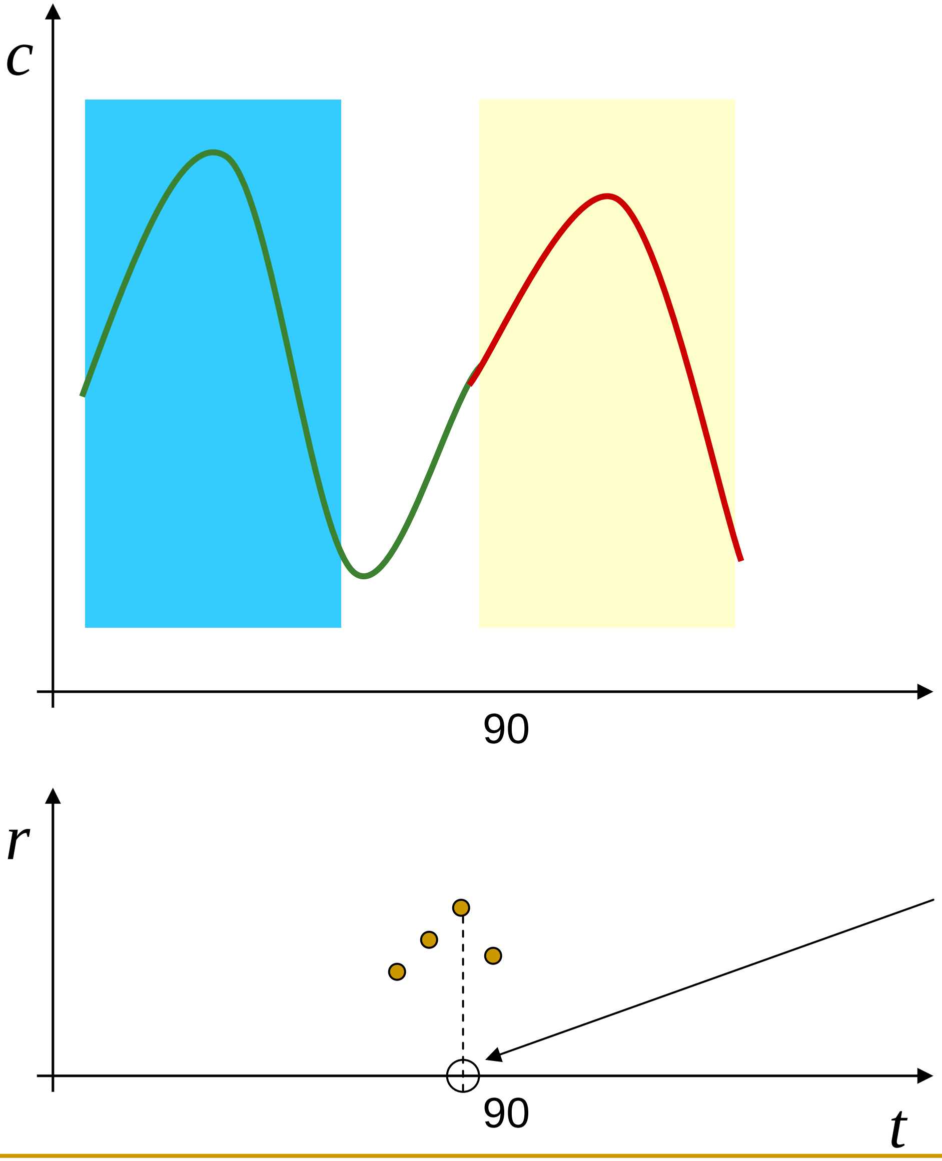
We estimate the signal time period


c
90
t
r
90
t

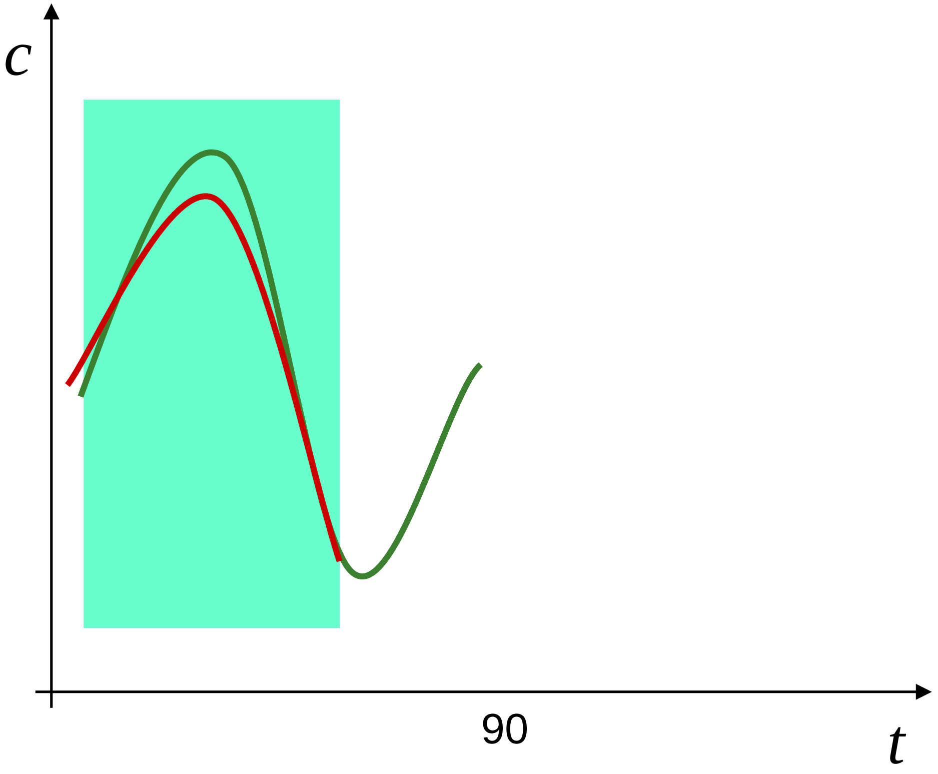

c
90
t
r
90
t


c
90
t


r
90
t



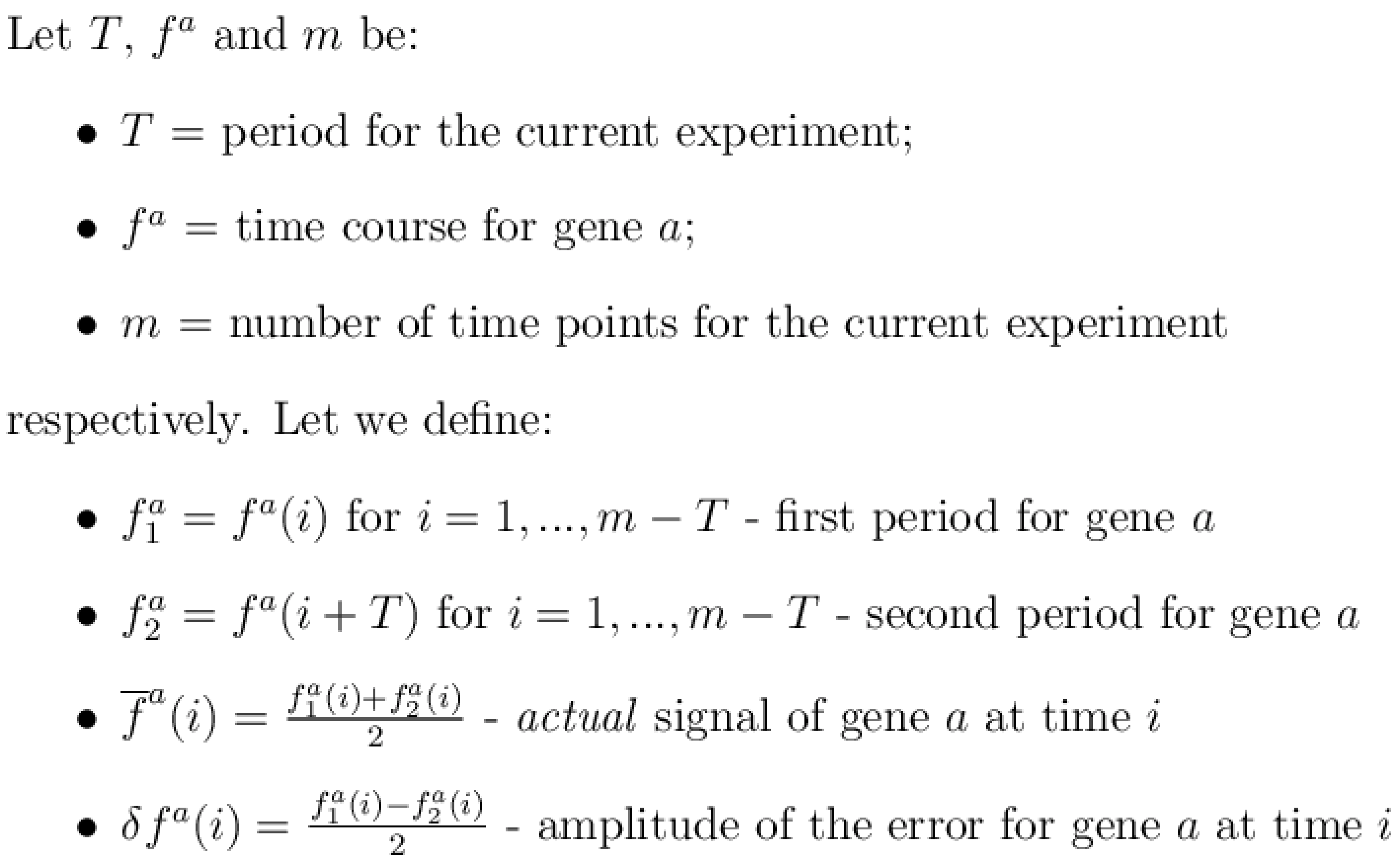
Best correlation point



The signals (before and after the cutting point) are superimposed: the average between them it’s the “true” signal. The difference is our estimate of the noise

Case Study

Gene Noise Estimation Method

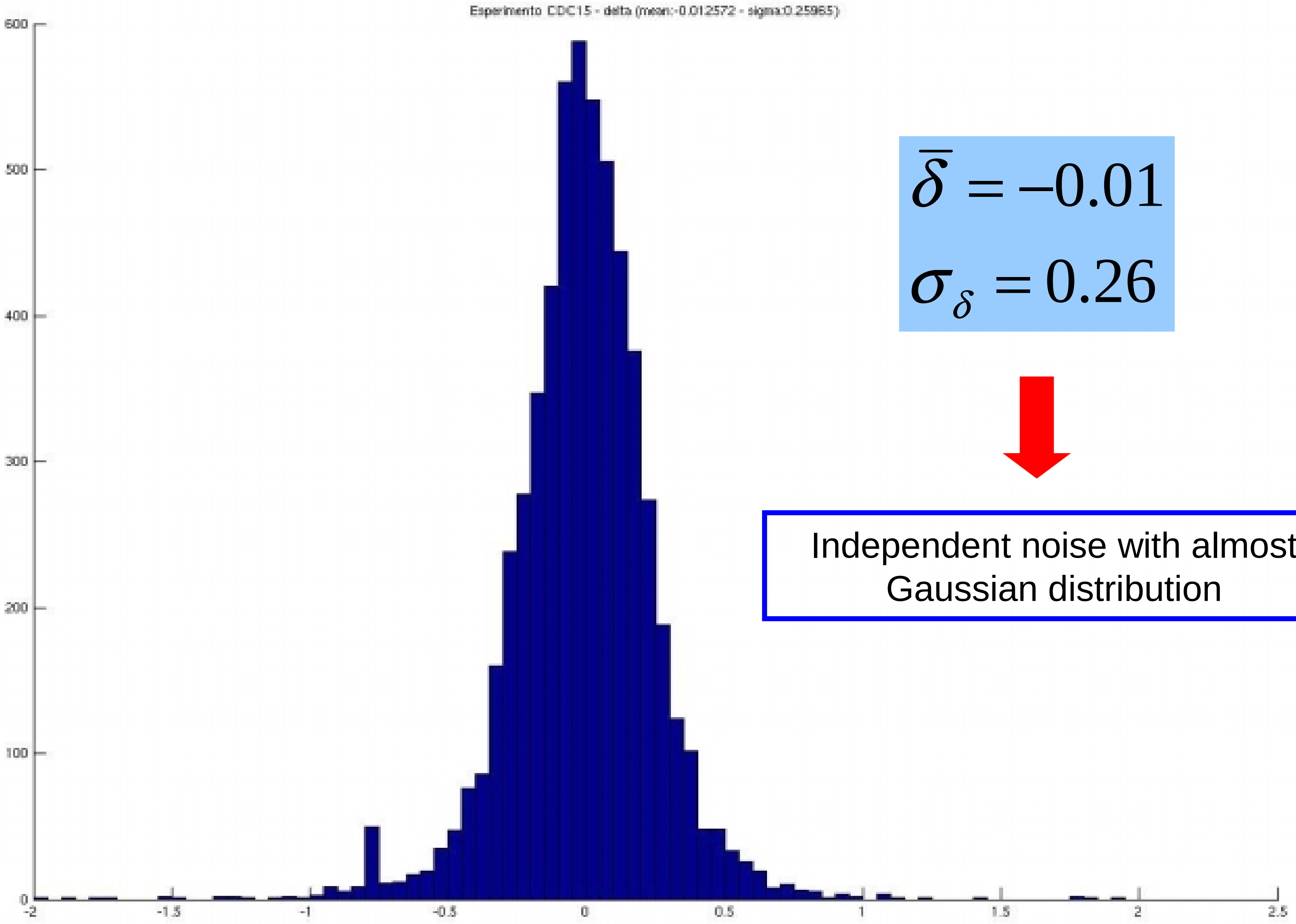


Let T, f^a and m be:

- T = period for the current experiment;
- f^a = time course for gene a;
- m = number of time points for the current experiment

respectively. Let we define:

- $f_1^a = f^a(i)$ for $i = 1, ..., m - T$ - first period for gene a
- $f_2^a = f^a(i + T)$ for $i = 1, ..., m - T$ - second period for gene a
- $\overline{f}^a(i) = \frac{f_1^a(i)+f_2^a(i)}{2}$ - *actual* signal of gene a at time i
- $\delta f^a(i) = \frac{f_1^a(i)-f_2^a(i)}{2}$ - amplitude of the error for gene a at time i

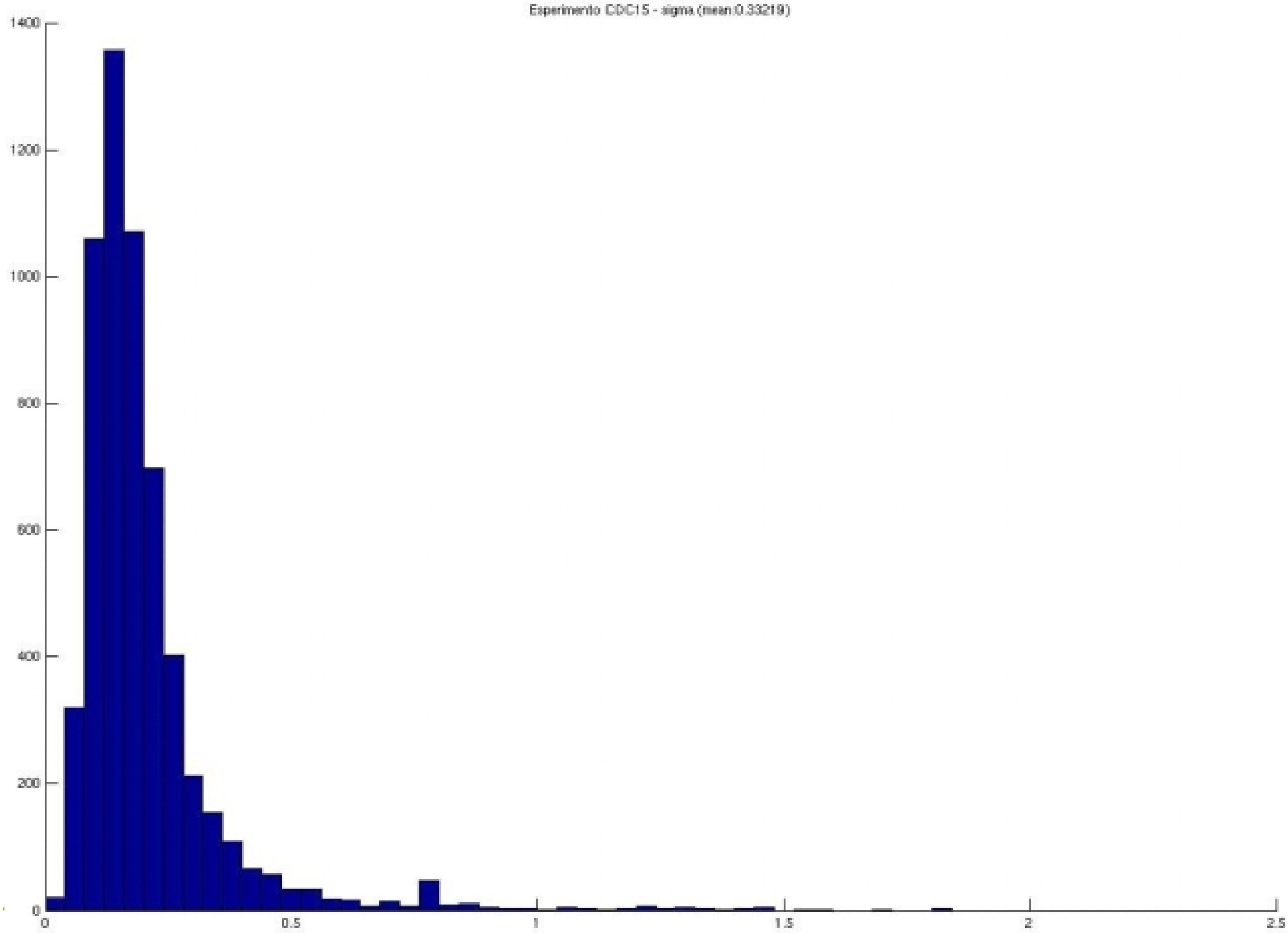


Distribution of $\delta^a = \frac{1}{m}\sum_{i=1}^{m} \delta f^a(i)$

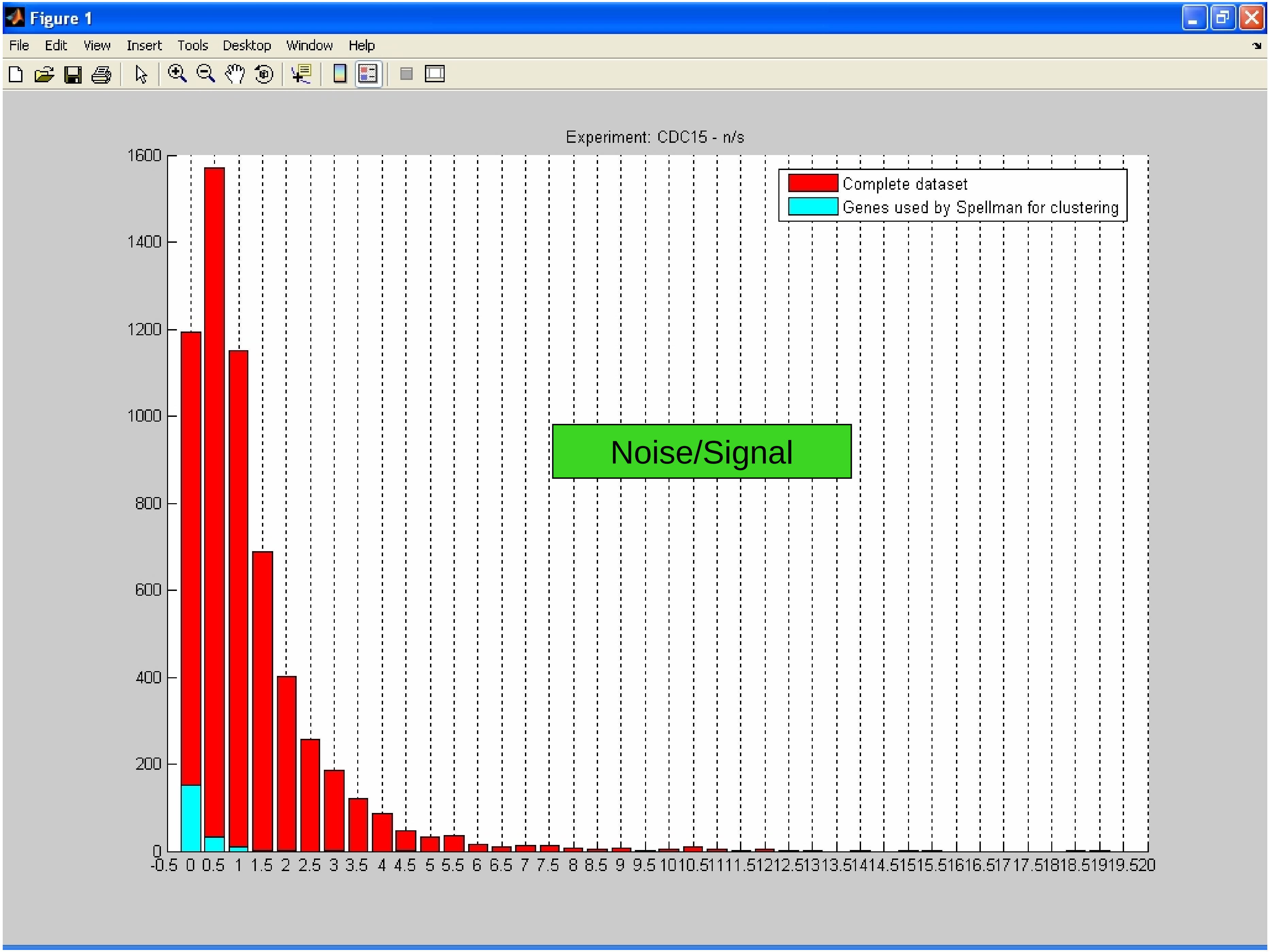
$$\bar{\delta} = -0.01$$

$$\sigma_\delta = 0.26$$

Independent noise with almost Gaussian distribution



Distribution of $\sigma^a = \sqrt{\frac{1}{m}\sum_{i=1}^{m}[\delta f^a(i)]^2}$ i.e. mean amplitude of the error



Noise/Signal

Case Study

Preprocessing (nonlinear PCA)

- The data of the experiments are unevenly sampled;
- To extract the features from the experiments we apply a non-linear Principal Component Analysis;
- In details, we apply for each experiment the non-linear PCA to extract the components (1 in our case) to obtain the features.

Case Study

Neg-entropy based Clustering (NEC)

- Starting from the PPS density function we cluster its Gaussian components using information based on entropy.
- Several approaches have been introduced based on the *hypothesis test* or *Kullback-Leibler* divergence
- We introduce an approach based on the *Neg-entropy*
- The algorithm permits to agglomerate automatically the clusters using non-Gaussianity information

Case Study

NEC: neg-entropy

- Neg-entropy is based on the information-theoretic quantity of differential entropy

- It is used to obtain a measure of non-Gaussianity that is zero for a Gaussian variable:

$$J(\mathbf{x}) = H(\mathbf{x}_{\mathbf{Gauss}}) - H(\mathbf{x})$$

where $\mathbf{x}_{\mathbf{Gauss}}$ is a Gaussian random variable of the same correlation (and covariance) matrix as $\mathbf{x}$

- Neg-entropy is always non-negative and it is zero if and only if $\mathbf{x}$ has a Gaussian distribution

Case Study

NEC: approximate neg-entropy

- The classical method to approximate neg-entropy is using high-order cumulants

$$J(\mathbf{x}) \approx \frac{1}{12} E\left\{\mathbf{x}^3\right\}^2 + \frac{1}{4}\mathrm{kurt}(\mathbf{x})^2$$

where **kurt** is the kurtosis

- A different and more robust approximation of the neg-entropy is

$$J(\mathbf{x}) \propto \left[E\{G(\mathbf{x})\} - E\{G(\upsilon)\}\right]^2$$

where υ is a standardized Gaussian variable and **x** has zero mean and unit variance

- We note that choosing a *G* that does not grow fast, one obtains more robust estimators. The following choices of *G* have proved very useful:

$$G^1 = \frac{1}{a}\log\cosh(a\mathbf{x}) \quad - \quad G^2 = \frac{1}{4}\mathbf{x}^4 \quad - \quad G^3 = -\frac{1}{a}e^{-a\frac{\mathbf{x}^2}{2}}$$

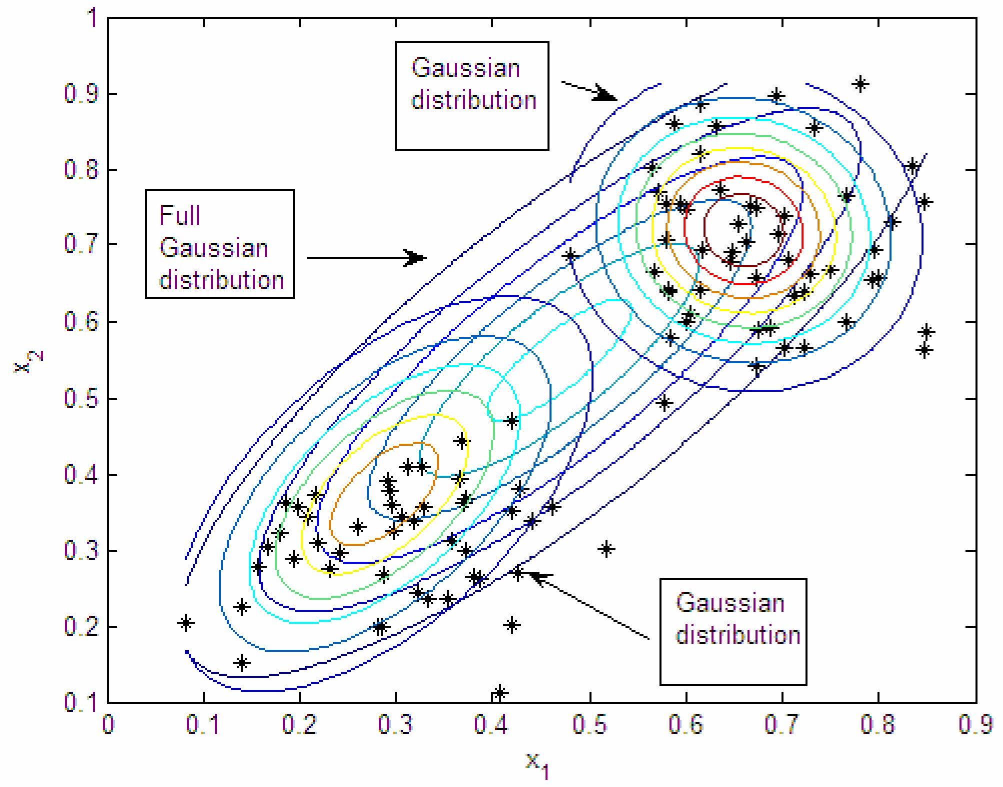
Case Study

NEC: algorithm

- Starts from M clusters (one for each PPS mixture component);
- Agglomerates two components, i and j:
    - if the new cluster candidate Neg-entropy value of is less of a fixed threshold
        - then i U j replaces clusters i and j. i U j becomes cluster i and j=j+1;
        - else j=j+1
    - the steps are repeated until all the components are processed
- Ends with the final number of clusters.

Case Study

NEC: Gaussians not merged by the algorithm

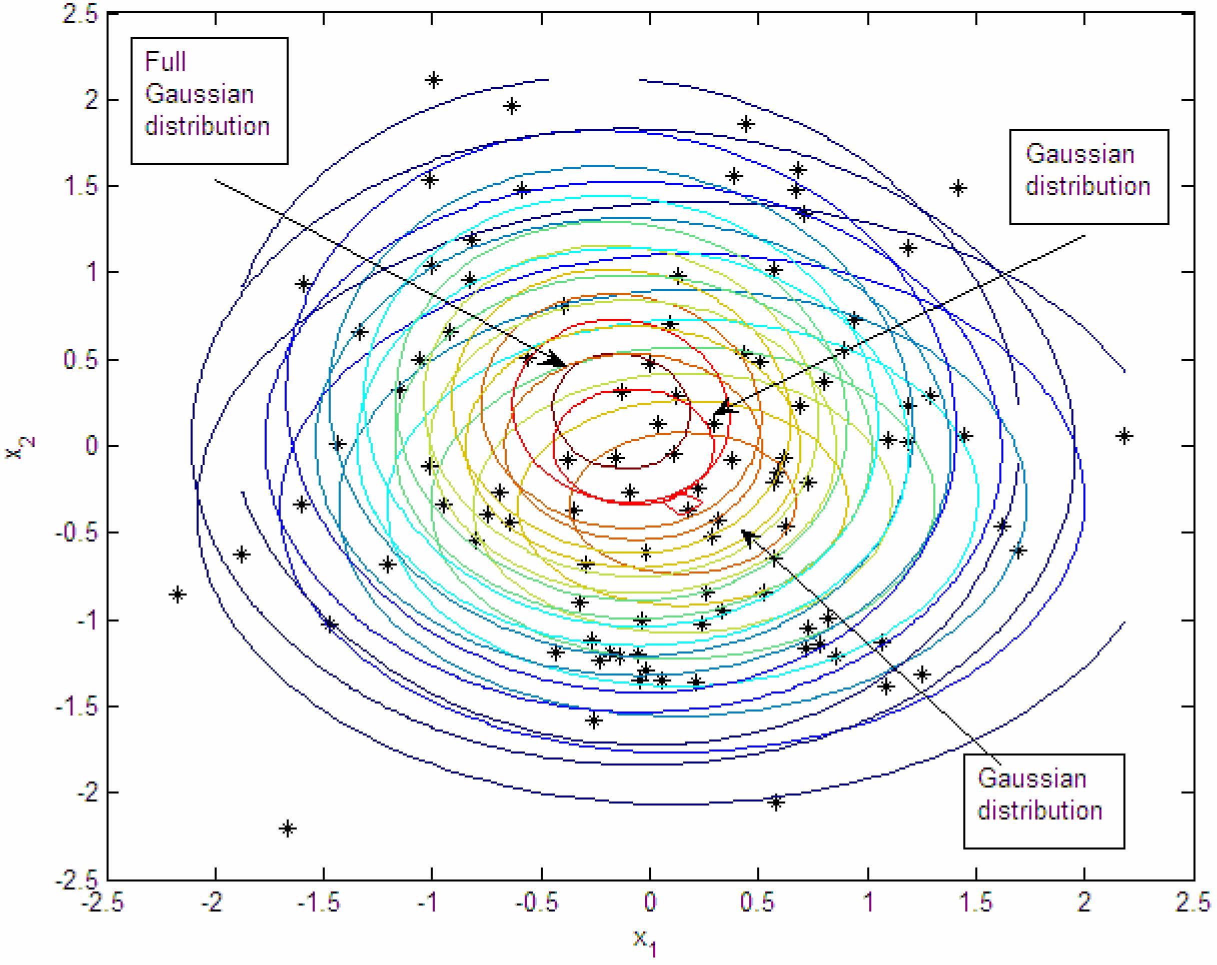


NegE=750

Case Study

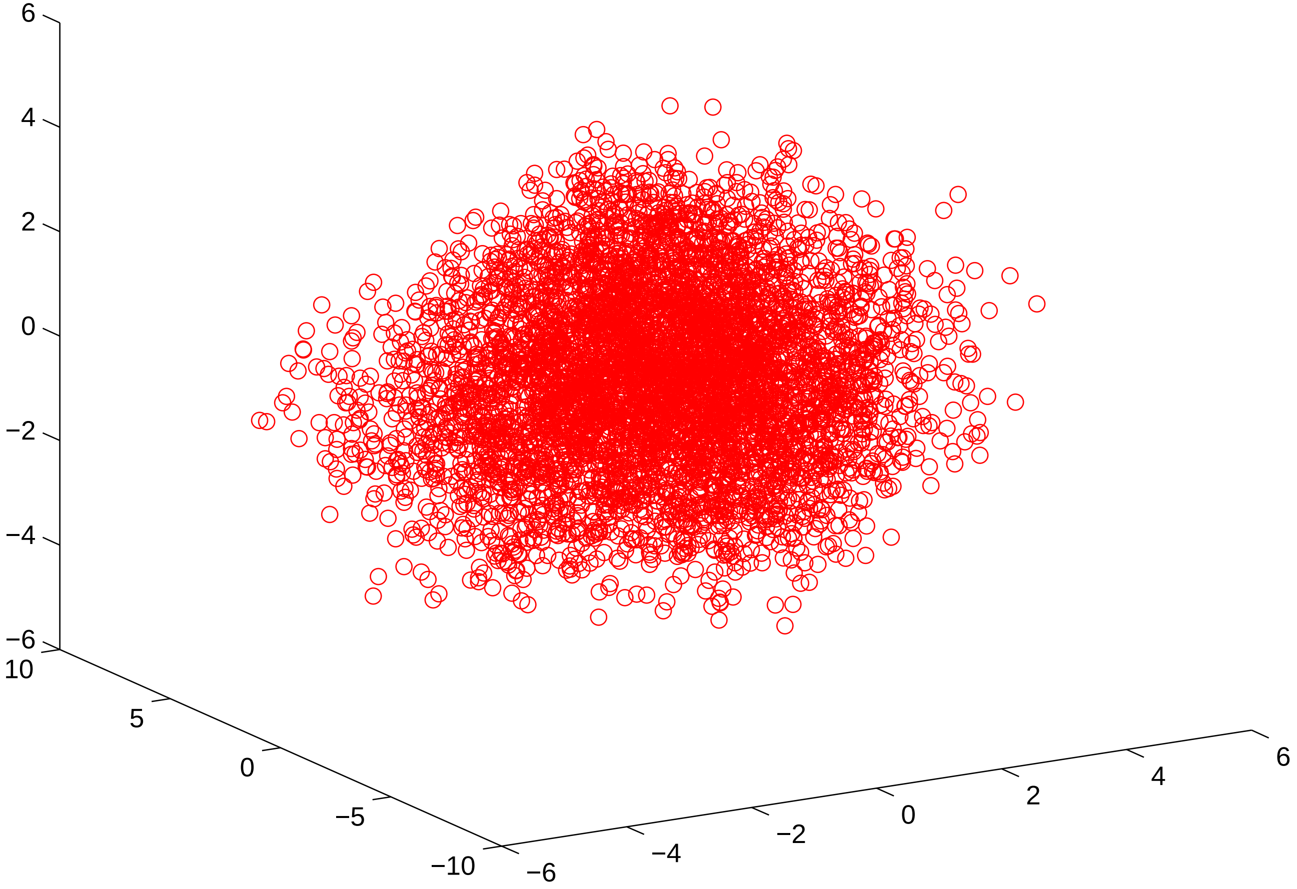
NEC: two merged Gaussian distributions

NegE=4



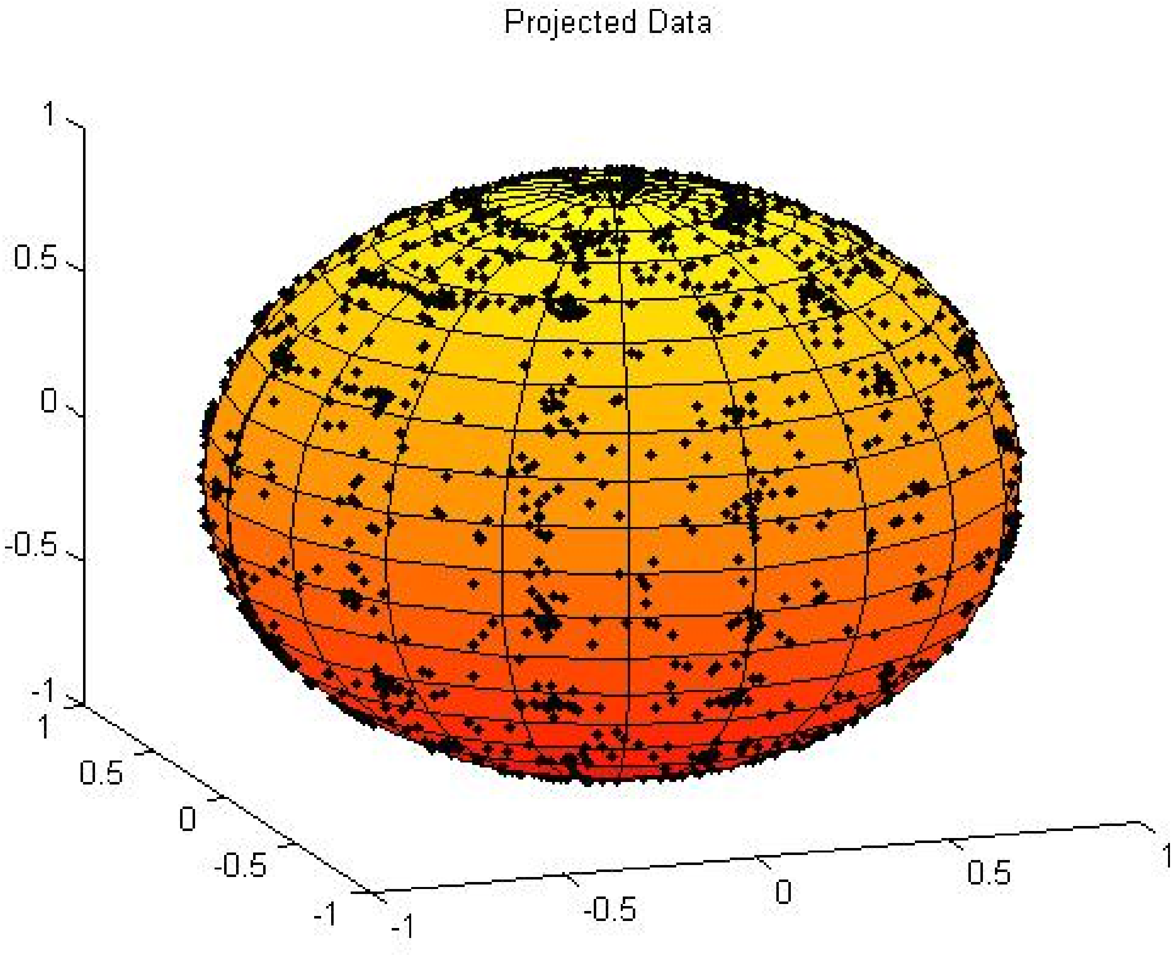
Case Study

3D PCA of Yeast Gene Microarray Data



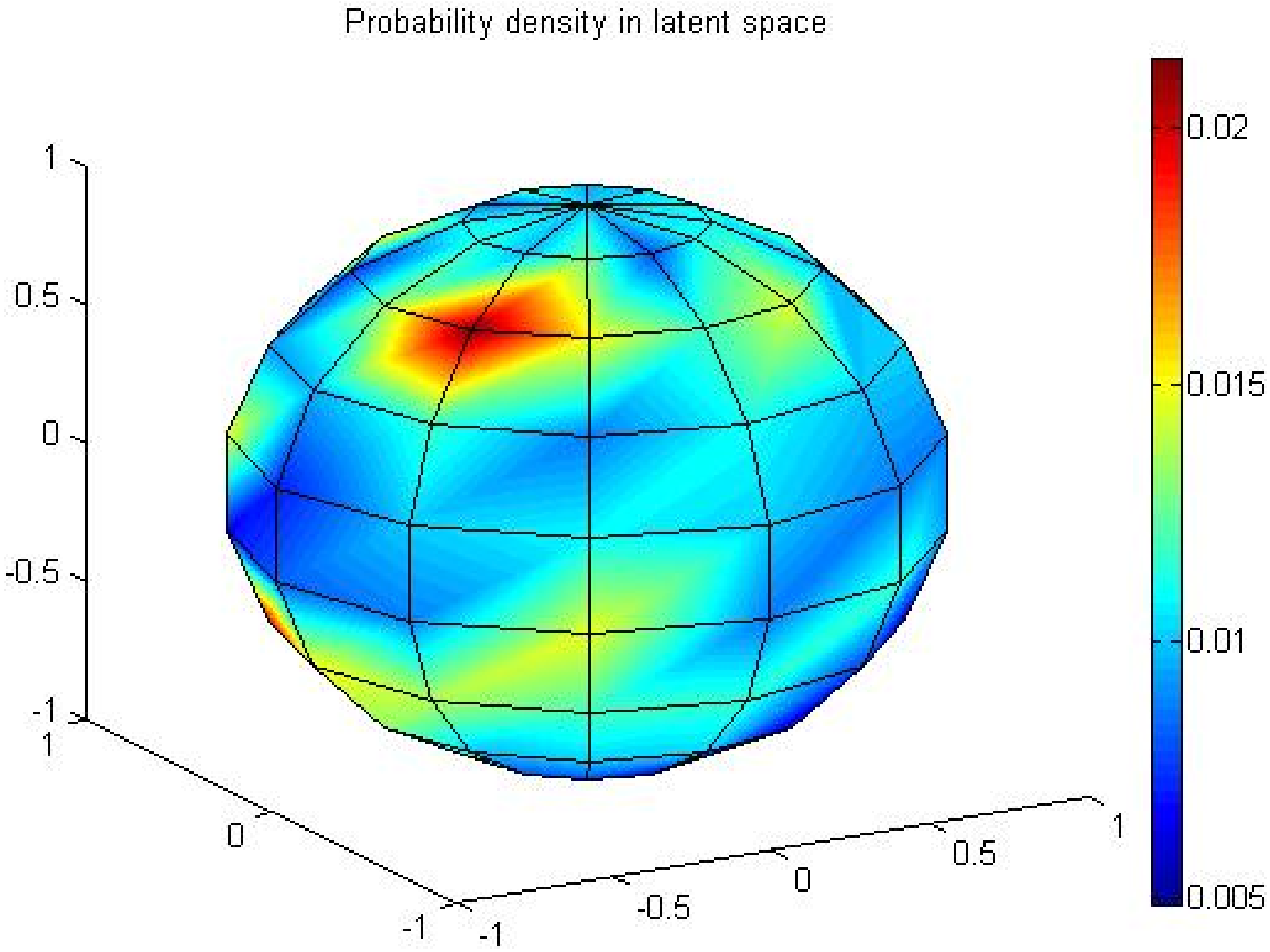
Case Study

PPS: data point projections



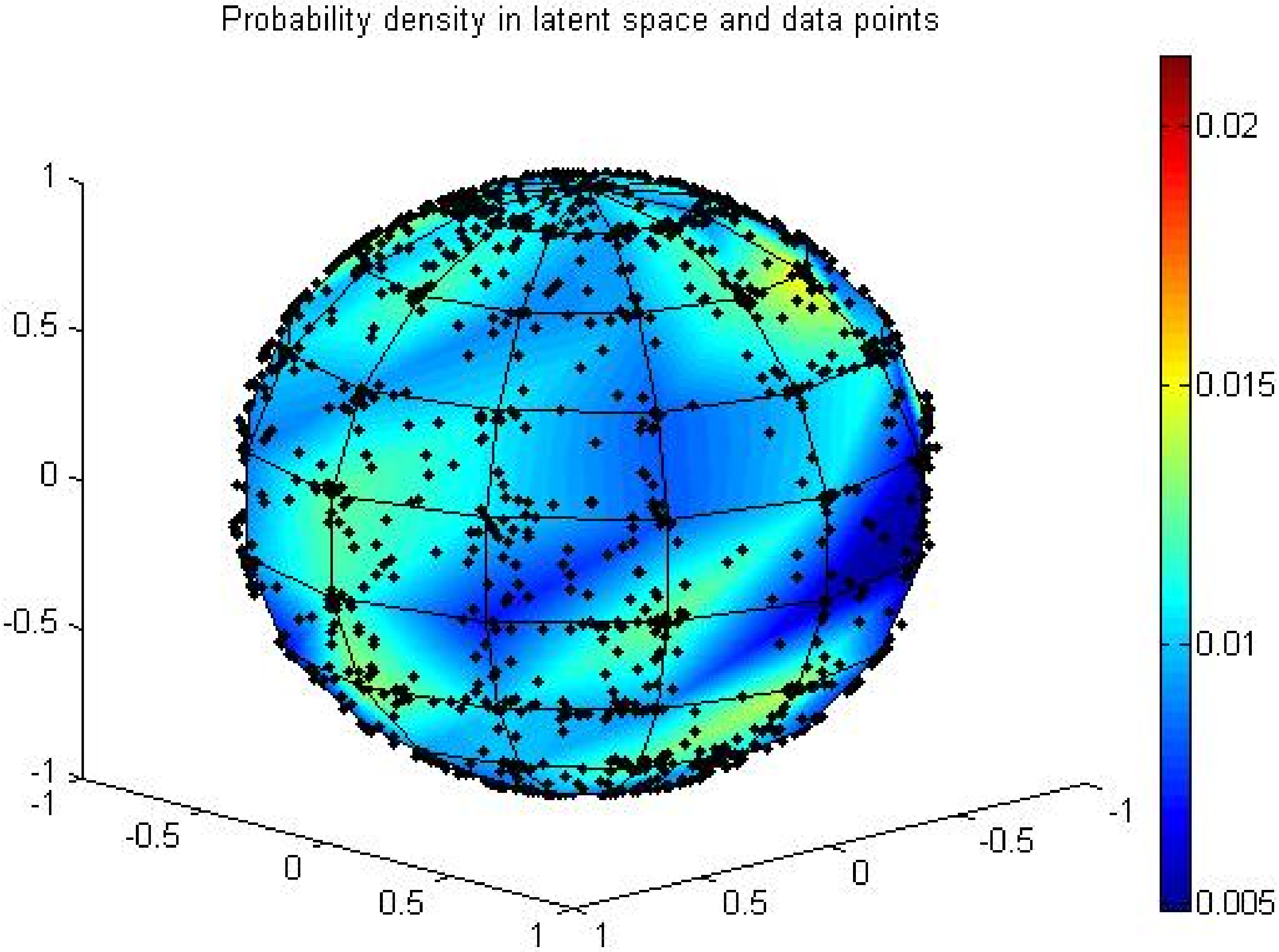
Case Study

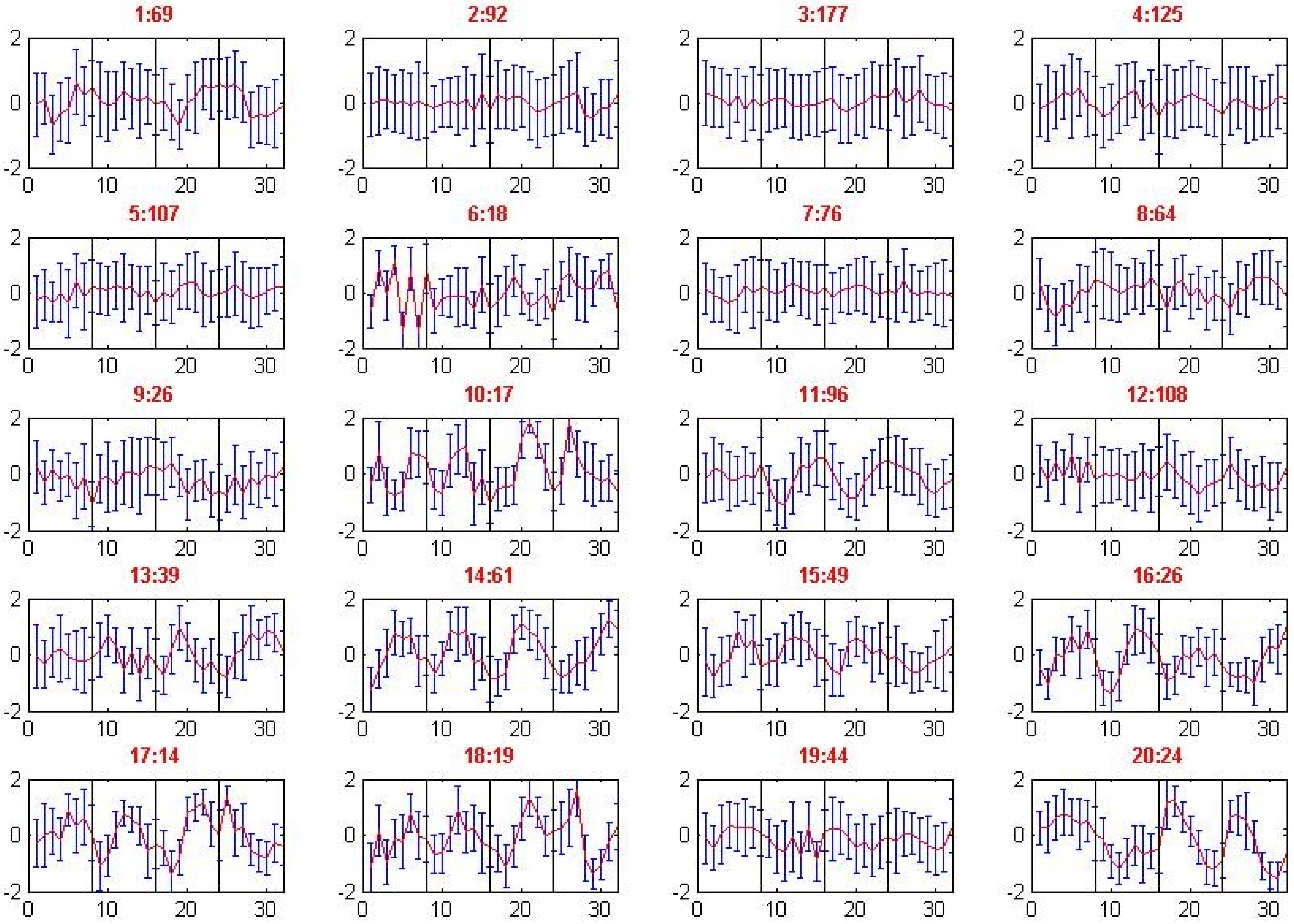
PPS: pdf

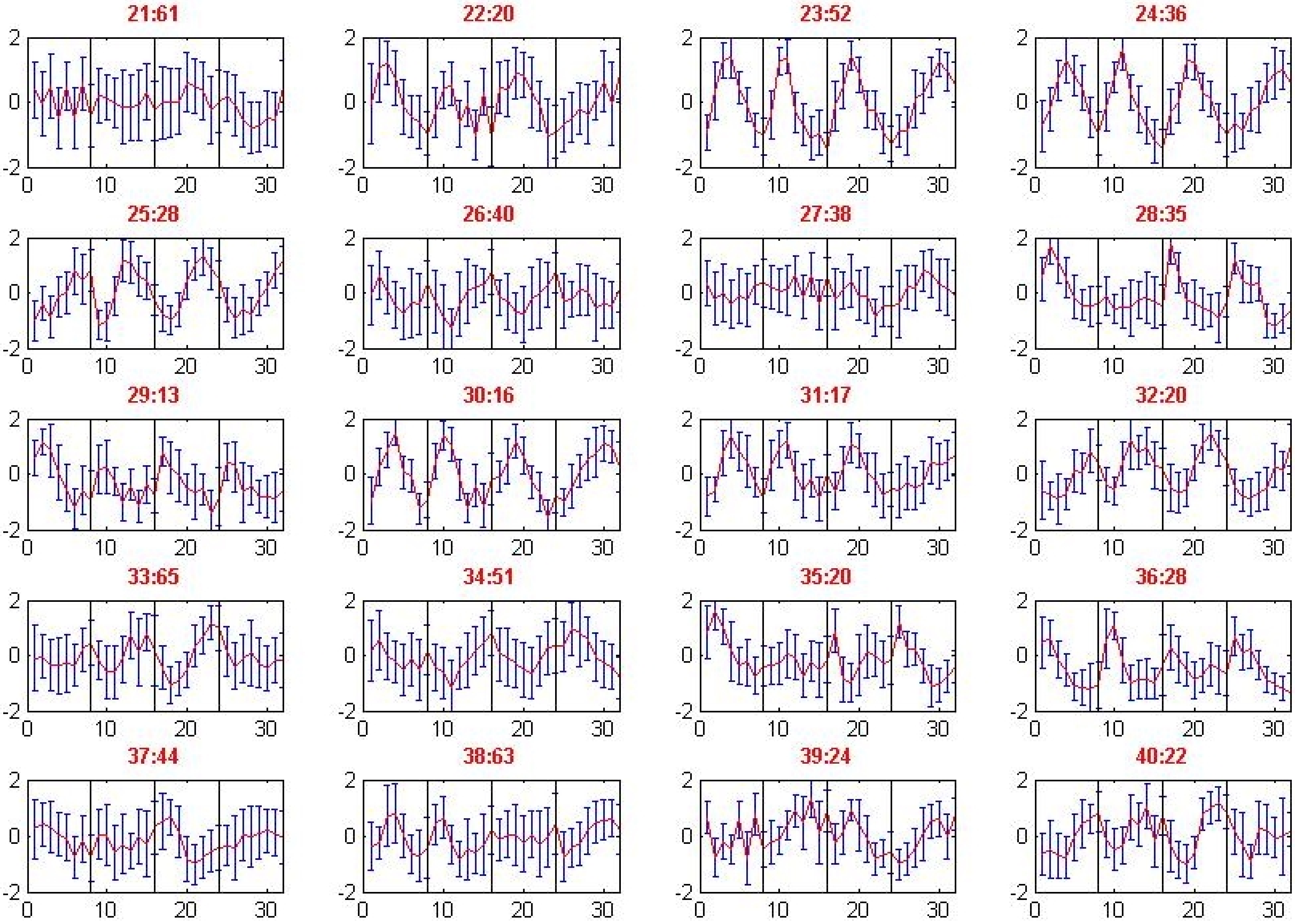


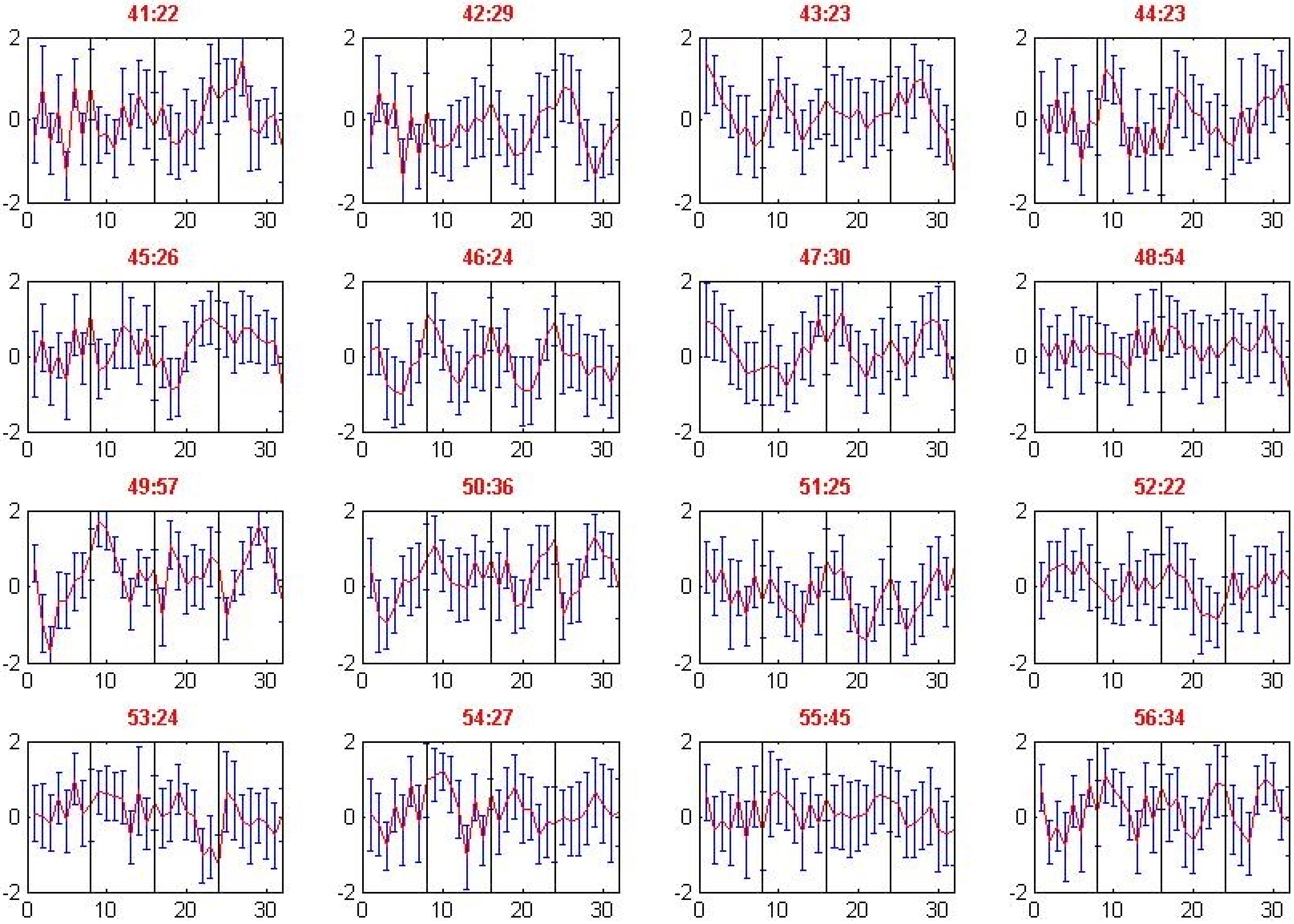
Case Study

PPS: pdf and data point projections




1:69
2:92
3:177
4:125
5:107
6:18
7:76
8:64
9:26
10:17
11:96
12:108
13:39
14:61
15:49
16:26
17:14
18:19
19:44
20:24
2
0
-2
0
10
20
30

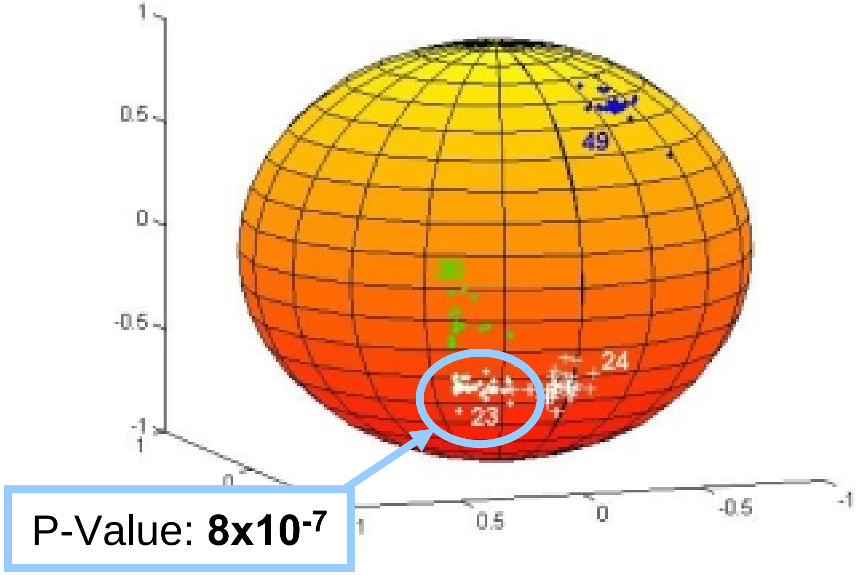

21:61
22:20
23:52
24:36
25:28
26:40
27:38
28:35
29:13
30:16
31:17
32:20
33:65
34:51
35:20
36:28
37:44
38:63
39:24
40:22
2
0
-2
0
10
20
30


41:22
42:29
43:23
44:23
45:26
46:24
47:30
48:54
49:57
50:36
51:25
52:22
53:24
54:27
55:45
56:34
2
0
-2
0
10
20
30

Case Study

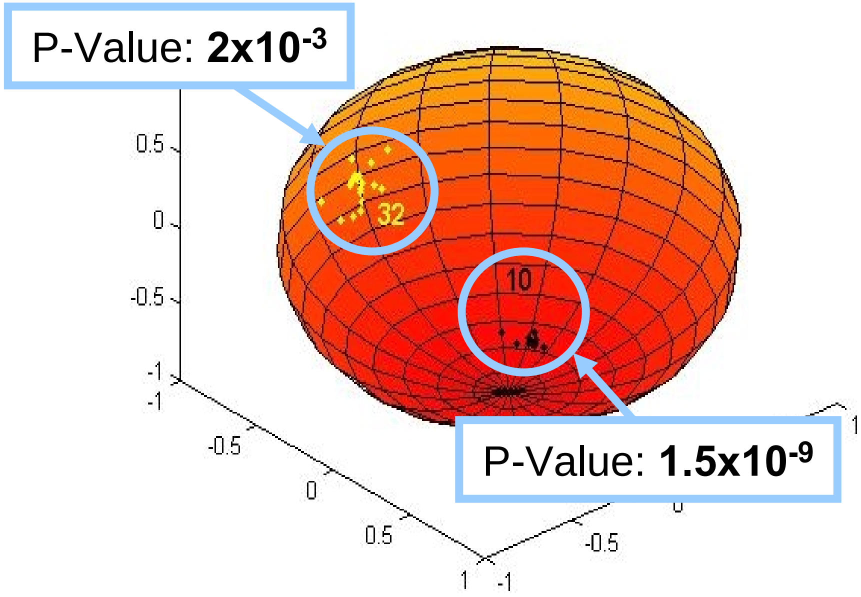
NEC Results

Front view



P-Value: **8x10^{-7}**

Back view



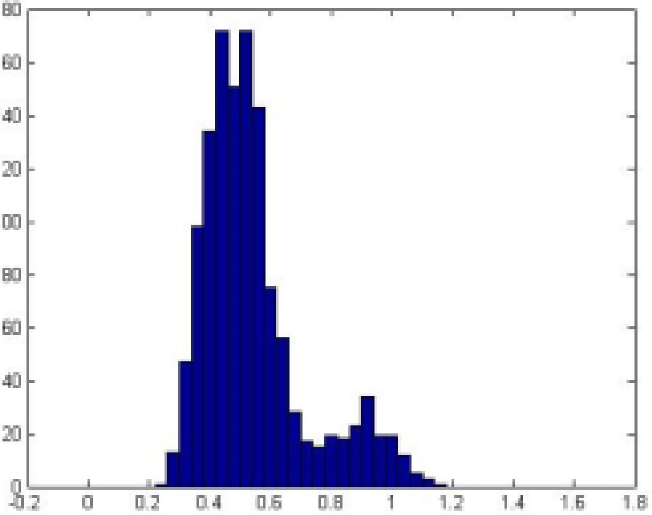
P-Value: **2x10^{-3}**

P-Value: **1.5x10^{-9}**

Case Study

About the threshold

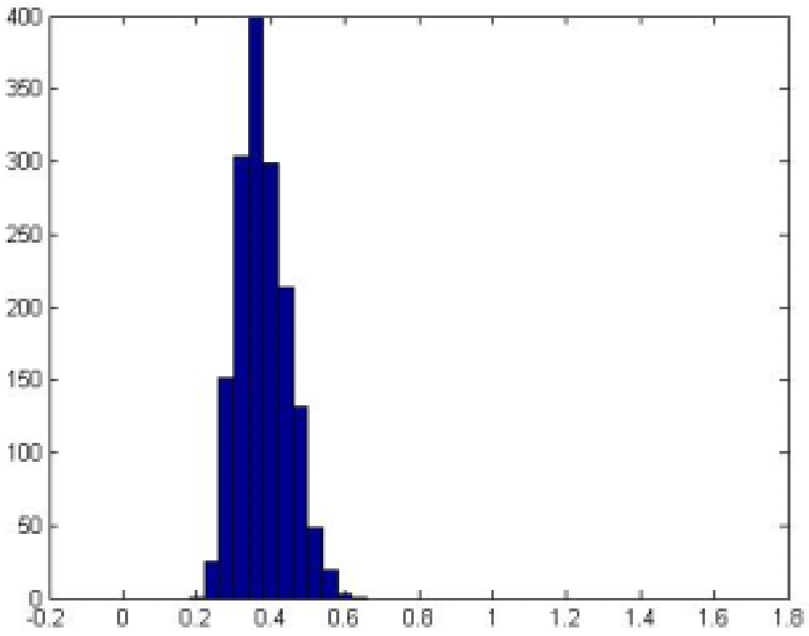
- Choosing of the right value for the threshold is critical.
- A value too high can produce **few** clusters **but** with a multimodal distribution of the distances



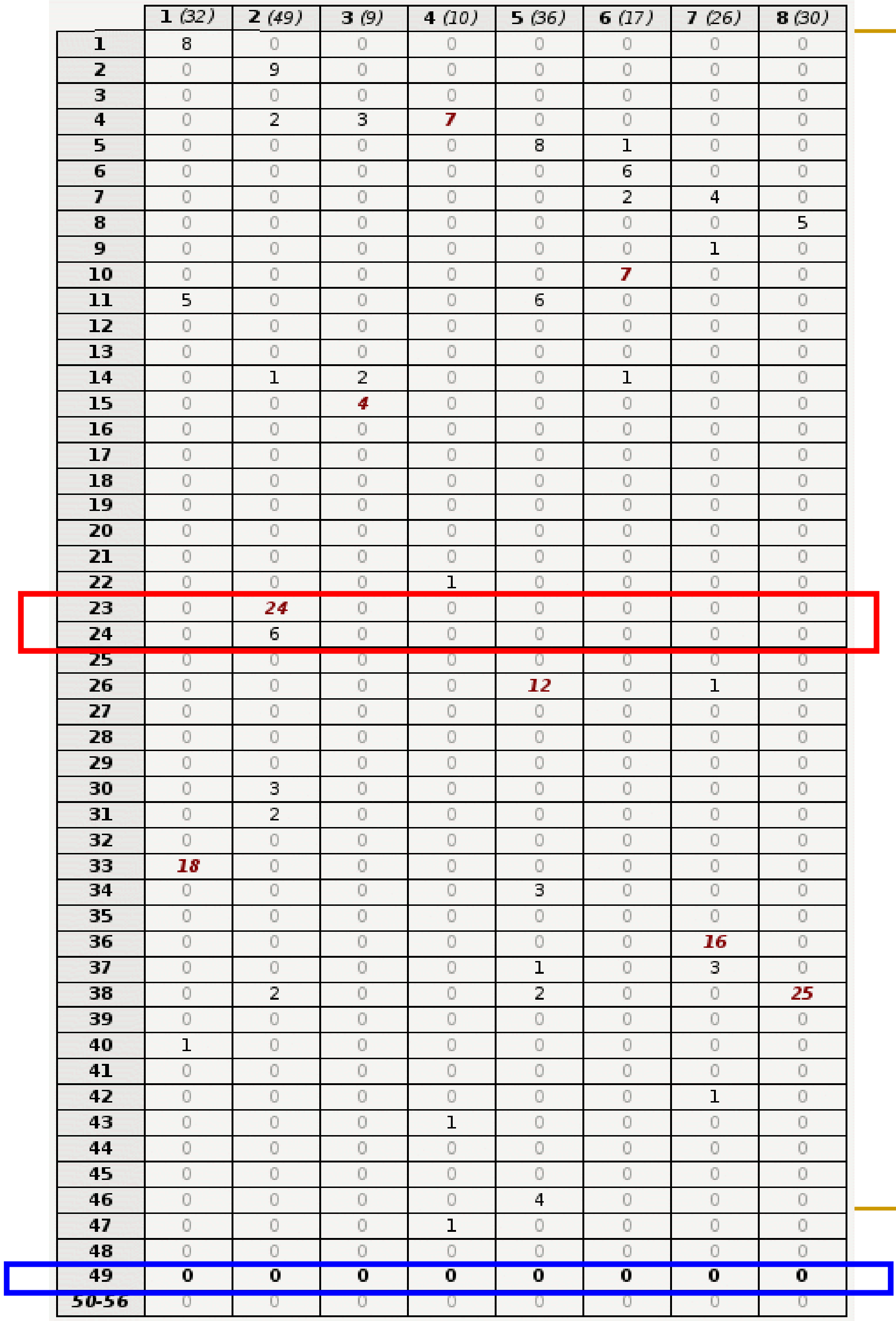
Cluster 34

Case Study

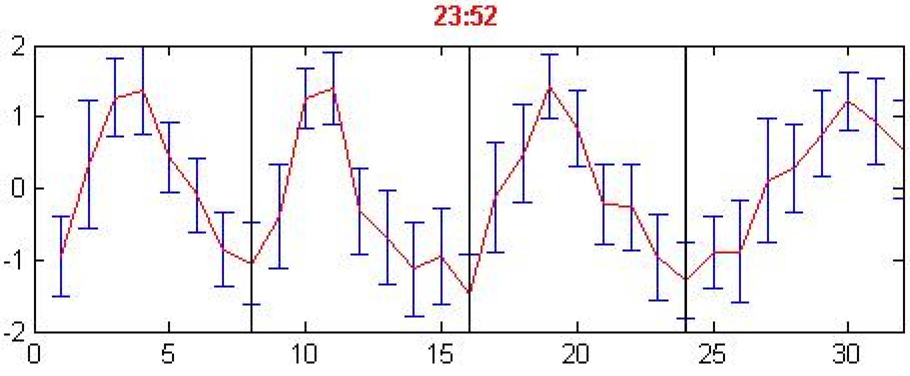
About the threshold…



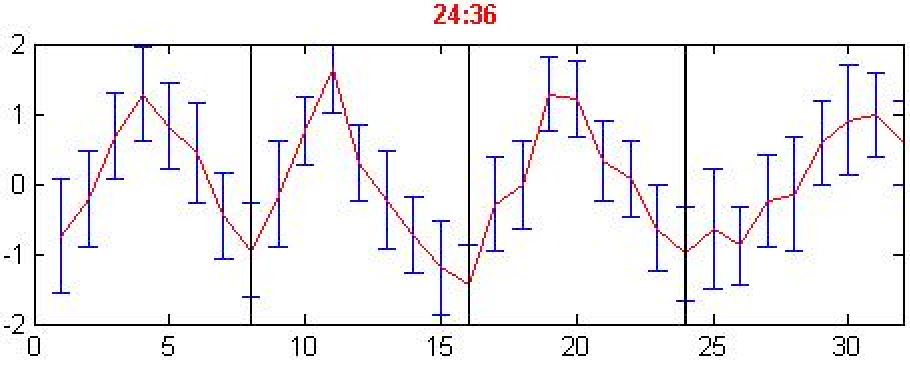
- A right value, instead, produce clusters with a mean distance consistent with the noise estimation we found in the first step



	1 (32)	2 (49)	3 (9)	4 (10)	5 (36)	6 (17)	7 (26)	8 (30)
1	8	0	0	0	0	0	0	0
2	0	9	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	0	2	3	*7*	0	0	0	0
5	0	0	0	0	8	1	0	0
6	0	0	0	0	0	6	0	0
7	0	0	0	0	0	2	4	0
8	0	0	0	0	0	0	0	5
9	0	0	0	0	0	0	1	0
10	0	0	0	0	0	*7*	0	0
11	5	0	0	0	6	0	0	0
12	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0
14	0	1	2	0	0	1	0	0
15	0	0	*4*	0	0	0	0	0
16	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0
22	0	0	0	1	0	0	0	0
23	0	*24*	0	0	0	0	0	0
24	0	6	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0
26	0	0	0	0	*12*	0	1	0
27	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0	0
30	0	3	0	0	0	0	0	0
31	0	2	0	0	0	0	0	0
32	0	0	0	0	0	0	0	0
33	*18*	0	0	0	0	0	0	0
34	0	0	0	0	3	0	0	0
35	0	0	0	0	0	0	0	0
36	0	0	0	0	0	0	*16*	0
37	0	0	0	0	1	0	3	0
38	0	2	0	0	2	0	0	*25*
39	0	0	0	0	0	0	0	0
40	1	0	0	0	0	0	0	0
41	0	0	0	0	0	0	0	0
42	0	0	0	0	0	0	1	0
43	0	0	0	1	0	0	0	0
44	0	0	0	0	0	0	0	0
45	0	0	0	0	0	0	0	0
46	0	0	0	0	4	0	0	0
47	0	0	0	1	0	0	0	0
48	0	0	0	0	0	0	0	0
49	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
50-56	0	0	0	0	0	0	0	0



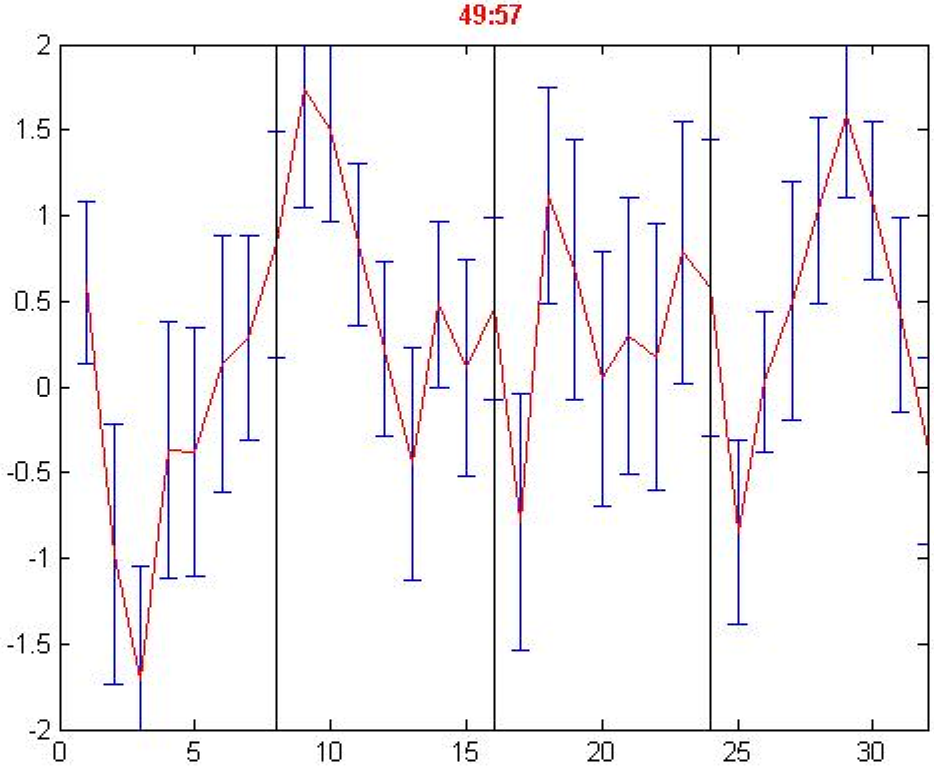
23:52



24:36

Case Study

Cluster 49…



P-Value: 1.6 x 10^{-21}

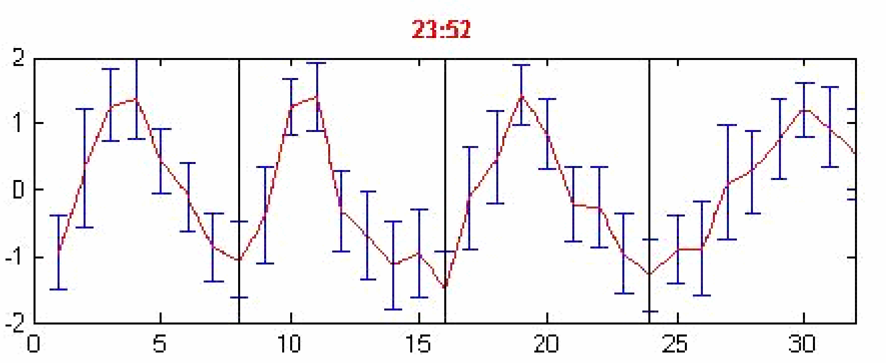
Case Study

Cluster 49

- 56 genes (many of them not cell- cycle regulated).
- No interesection with Spellman's cluster.
- 26 out of 56 genes are the components of nucleus and membrane- bound organelles, and are involved in two processes, cytoplasm organization and biogenesis and ribosome biogenesis and assembly.
  - 46% of these 26 genes have the functions of RNA binding and catalysing the ATF separation reactions during unwinding RNA helix.
- Some of 24 rested genes are involved in biopolymer metabolism.
- There are **half of genes among these 24** rested genes with **unknown functions (could be object of further analysis)**.

Case Study

Cluster 23



- 29 genes
- p-value = 8x10-7
- 48,98% intersection with Spellman CLN2 cluster.
  - Most of these genes are strongly cell-cycle regulated, peak expression occurs in mid–G1 phase
  - strongly induced by GAL-CLN3 but are strongly repressed by GAL-CLB2.
  - All these genes are involved in DNA replication.
- The rest of cluster contains some genes with **unknown functions**

Conclusions …

- Spherical PPS exhibits a number of attractive abilities for classification and visualization of high-D data
- The spherical manifold is able to better characterize and represent the periphery and the sparsity of high-D data due to the *curse of dimensionality*
- Overcome border effects as in rectangular manifold (GTM) and grid (SOM)

Conclusions …

- Visualization is an important tool in data mining applications for all types of user

- The domain expert must be involved in the process

- Interaction with the plots allows the user to query the data more effectively.

Conclusions

- We built a graphical user interface which allows to interact with the data projected on a unit sphere surface

- A user is allowed to
  - Interact with data by selecting points on the latent manifold retrieving the corresponding source in the original catalog

  - The user is able to localize clusters of data on the sphere which correspond to clusters of similar data in the input space

- Useful for genetic data mining, but general enough to address a large number of application fields

Bibliography

- K. Chang, J. Ghosh, ***A Unified Model for Probabilistic Principal Surfaces***, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 23, No. 1, 2001

- P. T. Spellman et al., ***Comprehensive Identification of Cell Cycle-regulated Genes of the Yeast Saccharomyces cerevisiae by Microarray Hybridization***, Molecular Biology of the Cell, Vol. 9, 3273-3297, December, 1998

- A. Staiano, ***Unsupervised Neural Networks for the Extraction of Scientific Information from Astronomical Data***, PhD Thesis, University of Salerno, 2003

- R. Tagliaferri, A. Ciaramella, et al., ***Spectral analysis of stellar light curves by means of neural networks***, Astronomy and Astrophysics Supplement Series, 137:391--405, 1999