

Microarray Data Analysis

Gene Filtering, Missing Values Imputation

(Affymetrix GeneChip)

國立臺灣大學 資訊所

Course: 生物資訊之統計與計算方法

2007/03/29

吳漢銘

hmwu@stat.sinica.edu.tw

<http://idv.sinica.edu.tw/hmwu/>

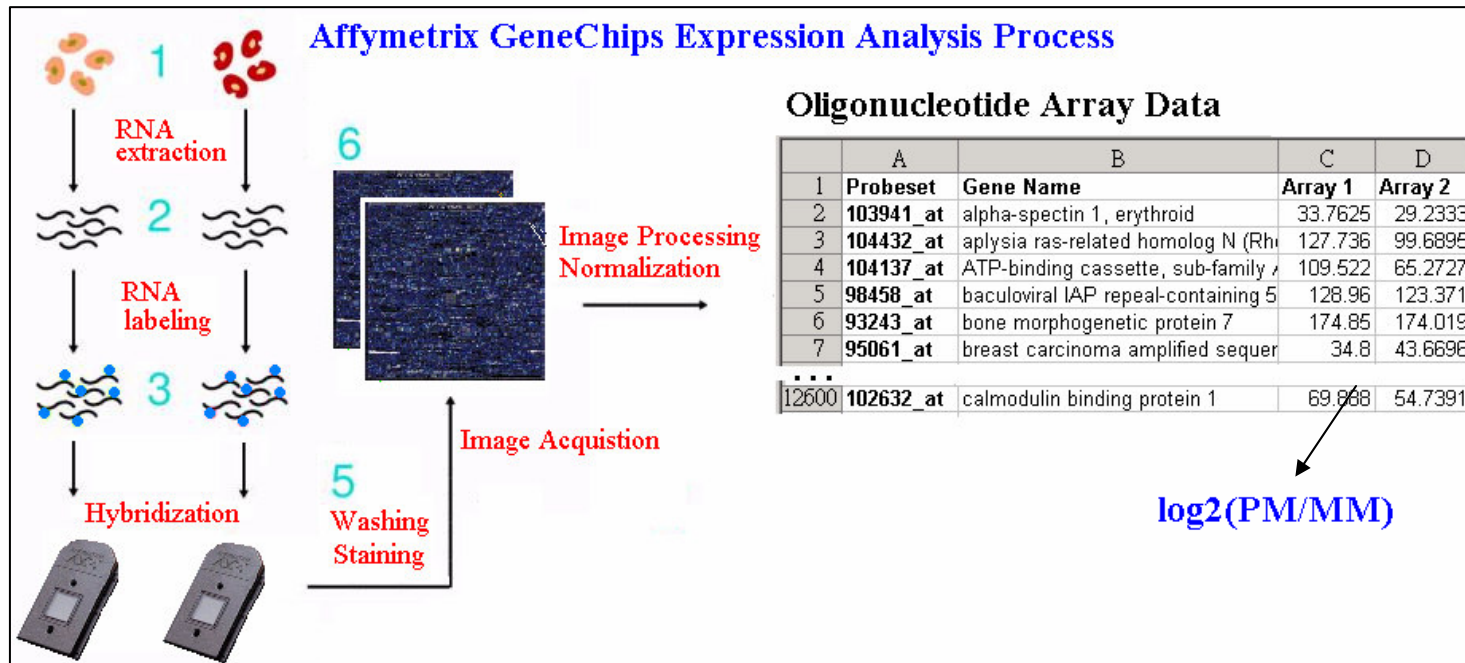
Institute of Statistical Science, Academia Sinica

中央研究院 統計科學研究所

Outlines



2/13



- **MSA5: Detection Calls**
- **dChip: Filter Genes**
- **Missing Values Imputation: KNN, SVD**

MSA5: Detection Calls



- **Answers:** “Is the transcript of a particular gene Present or Absent?”
- **Absent** means that the expression level is below the threshold of detection. That is, the expression level is not provably different from zero.
- **Advantage:** easy to filter and easy to interpret: we may only want to look at genes whose transcripts are detectable in a particular experiment.

Method

There are four steps to the method:

1. Remove saturated probe pairs and ignore probe pairs wherein $PM \sim MM + \tau$
2. Calculate the discrimination scores. (This tells us how different the PM and MM cells are.)
3. Use Wilcoxon’s rank test to calculate a significance or p -value. (This tells us how confident we can be about a certain result.)
4. Compare the p -value with our preset significance levels to make the call.

Saturation

If a mismatch cell is saturated $MM \geq 46000$, the corresponding probe pair is not used in further computations. We also discard pairs where PM and MM are within τ of each other.

Discrimination Score

$$R_i = \frac{PM_i - MM_i}{PM_i + MM_i}$$

Computing p -values:

The one-sided Wilcoxon’s Signed Rank Test

$$H_0: \text{median}(R_i - \tau) = 0$$

$$H_1: \text{median}(R_i - \tau) > 0$$

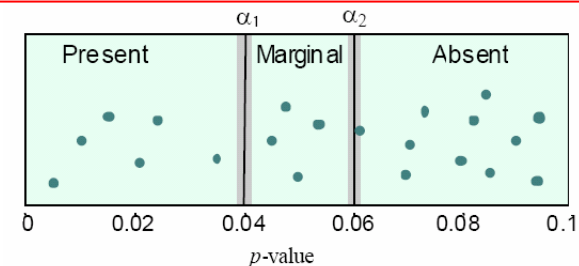
Making the call

We set two significance levels α_1 and α_2 such that $0 < \alpha_1 < \alpha_2 < 0.5$

default $\alpha_1 = 0.04$ (16-20 probe pairs)

default $\alpha_2 = 0.06$ (16-20 probe pairs)

The null hypothesis is that the target is absent (zero effect on the probes).



Significance levels α_1 and α_2 define cut-offs of p -values for making calls.

dChip: Filter Genes



4/13

1. $A < SD/mean < B$

$A < SD$ (for logged data) $< B$

A gene is variable enough compared to its mean expression level to contain interesting information ($> A$), but not so variable that nothing can be learned ($< B$).

2. Presence call $> X\%$

Narrows genes with a positive presence call in a certain percentage ($> X\%$) of the samples.

3. $A < \text{Median}(SD/\text{Mean}) < B$

4. Expression level $> Y$ in $X\%$

Since low expression estimates are sometimes unreliable, we may want to limit our analysis to genes that are expressed above some threshold ($> Y$) in a certain percentage ($X\%$) of the samples.

Filter Genes

Filter genes

Criterion

(1) Variation across samples (after pooling replicate arrays) :
0.5 < Standard deviation / Mean < 10

(2) P call % in the arrays used >= 70 %

(3) Variation within replicate arrays called Present:
0 < Median(Standard deviation / Mean) < 0.5

(4) The expression level is >= 20 in >= 50 % samples

Filter on gene list: using all genes

Filtered gene list: D:\BioInformatics\Web-Oligo\10-Software\dChi make sure the file is closed

Help Options...


確定 取消 適用(A)

<http://www.biostat.harvard.edu/complab/dchip/>

Useful Reference



5/13



A community resource for cereal microarrays

Search Website for

News | Data Access | Probe Set Info | Analysis & Viz. | Database Overview | Tools | My BarleyBase | Feedback

Hello! Guest! Please [Login](#) or [Register!](#) [Log Out](#)

Tutorial on Expression Profile Filters

Table of Contents

- [Introduction](#)
- [1. Absolute Value Filter](#)
- [2. MAS5.0 Call Filter](#)
- [3. Variation Filter](#)
- [4. Fold Change Filter](#)
- [5. Statistical Test Filter](#)
- [6. ANOVA Filter](#)
- [7. Variation Rank Filter](#)
- [8. Composite Filters Customized](#)
- [9. Usage of Expression Profile Filters](#)

<http://www.barleybase.org/filtertut.php>

GeneSpring Tutorials

<http://www.chem.agilent.com/Scripts/Generic.ASP?IPa ge=34743&indcol=Y&prodcol=Y>

GeneSpring User Manual

<http://www.chem.agilent.com/cfusion/faq/faq2.cfm?subs ection=78§ion=20&faq=1118&lang=en>



GeneSpring User Manual

Version 7.0

9

Filtering Data

This chapter explains how to use the basic and advanced filtering tools in GeneSpring. It covers the following topics:

- [Filtering](#)
- [Filtering on Gene Lists](#)
- [Using Advanced Filters](#)
- [Filtering Data Objects Assigned to Projects](#)

Filtering

Using GeneSpring's sophisticated filtering tools, you can identify genes that are affected by novel drug treatments or experimental conditions. A variety of intuitive visual interfaces allow even novice users to select genes with specific expression patterns.

GeneSpring offers visually-intuitive filtering tools for both entry-level and advanced users. All visual filtering windows generate graphs of results in real-time. These filters allow researchers to exclude particular conditions, set minimum and maximum values, and choose specific gene lists to filter.

GeneSpring also has an advanced filtering window designed for power users. The advanced filtering window allows you to create complex Boolean expressions to identify genes with a highly-specific expression pattern.

Once created, filters can easily be saved to standardize critical laboratory procedures, or can be shared with other researchers using Signet.

Filtering on Gene Lists

Gene filtering is a simple, but effective way to sort through the large amounts of expression data. Filtering enables you to evaluate the quality of sample before performing data analysis or identify interesting genes for further study after analysis. This section includes the following topics:



- [Gene Filters](#)
- [Filtering Menu](#)
- [Filter Window](#)
- [Data Types for Restrictions](#)

Missing Values Estimation for Microarray Data

Missing values imply a loss of information

- Many analysis techniques that require complete data matrices: such as **hierarchical clustering**, **k-means clustering**, and **self-organizing maps**.
- May benefit from using more accurately estimated missing values.

Possible Solution

1. Exclude missing values from subsequent analysis. 
2. Repeat the experiment **Expensive.** *May be of scientific interest !*
3. Missing values in replicated design.
4. Adjust dissimilarity measures. (e.g., pairwise deletion.)
5. Modify clustering methods that can deal with missing values.
6. **Imputation of missing values.** 

Sources of Missing Values



7/13

■ Various Reasons

- a feature of the robotic apparatus may fail,
- a scanner may have insufficient resolution,
- simply dust or scratches on the slide (image corruption),
- spots with dust particles, irregularities, ...

■ Mathematical transformation

- undefined mathematical transformed:
e.g., corrected intensities values that are **negative** or **zero**, a subsequent log-transformation will yield missing values.

■ Flag

Spots may be flagged as **absent** or **feature not found** when nothing is printed in the location of a spot.

- the imaging software cannot detect any fluorescence at the spot,
- expression readings that are barely above the background correction,
- the expression intensity ratio is undefined: */0, 0/*.

GenePix

Good=100. Bad=-100. Not Found=-50. Absent=-75. unflagged=0.

Statistical Classification of Missing Data

It helpful to classify missing values on the basis of the **stochastic mechanism** that produces them.

Missing Completely At Random (MCAR)

- Missingness is **independent** of their own **unobserved values** and the **observed data**.
- Arising from chance events that are **unrelated to the nature** of the investigation.
- **e.g.**, A spot that is obscured accidentally by a dust particle.

Missing At Random (MAR)

- Missingness **does not depend** on their on **unobserved value** but does **dependent** on the **observed data**.

Missing Not At Random (MNAR)

- Missingness **depend** on their own **unobserved values**.
 - missingness depends on the fact that their raw intensity values are zero or small.
 - **e.g.**, Spots that show no fluorescence or that have undefined log-intensities because their background-corrected intensities are negative.
- The missing values may give clues to **systematic aspects of the problem**.
 - If missing values do occur by chance among a set of **replicates**, the observed members of the set can stand in for the missing, albeit with some loss of statistical precision.

Imputation: methods rely on the missingness being of the **MCAR** type.

Imputation of Missing Values



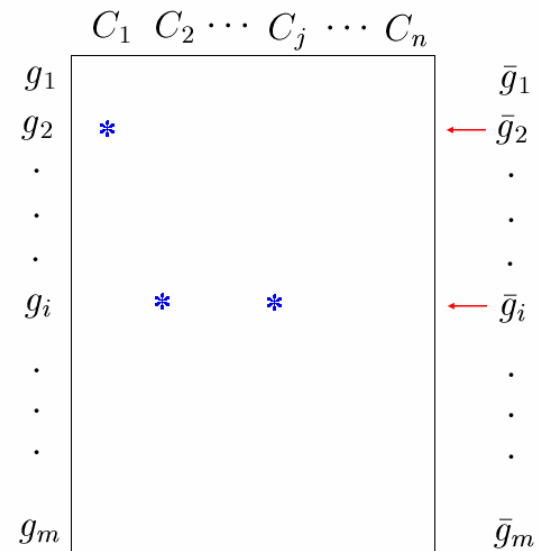
■ Missing log2 transformed data are replaced by *zeros* or by an *average* expression over the row ("row average").

■ Row average assumes that the expression of a gene in one of the experiments is *similar* to its expression in a different experiment, which is often not true in microarray experiments.

■ **Main weakness:**

- it makes no serious attempt to model the connection of the missing values to the observed data.
- since these methods do not take into consideration the *correlation structure* of the data.
- not very effective (Troyanskaya et al, 2001)

■ **Useful:** where an initial imputation is required an iterative imputation method.



$$\widehat{C_j(g_i)} = \bar{g}_i$$
$$i = 1, 2, \dots, m.$$
$$j = 1, 2, \dots, n.$$

zero's
row average
row median

K-Nearest Neighbors Imputation



KNNImpute: a missing value estimation method to **minimize** data modeling assumptions and take advantage of the **correlation structure** of the gene expression data.

- Results are adequate and relatively insensitive to values of **k** **between 10 and 20.** (Troyanskaya et al, 2001)

- Euclidean distance** appeared to be a sufficiently accurate norm.

	C_1	C_2	\dots	C_j	\dots	C_n
g_1	*	✓		*	✓	✓
g_2	■	✓		✓	✓	✓
\cdot						
\cdot	■	✓		*	✓	✓
\cdot						
g_i	■	✓		✓	✓	✓
\cdot						
\cdot						
g_m		*		*		

KNNImpute

Model:

$$\{g_{(k)}, k = 1, 2, \dots, K\} = \text{args} \max_k \text{Corr}(g_1, g_i)$$

$$\{g_{(k)}, k = 1, 2, \dots, K\} = \text{args} \min_k \text{Dist}(g_1, g_i)$$

C: Observed C_i 's without missing values

Imputation:

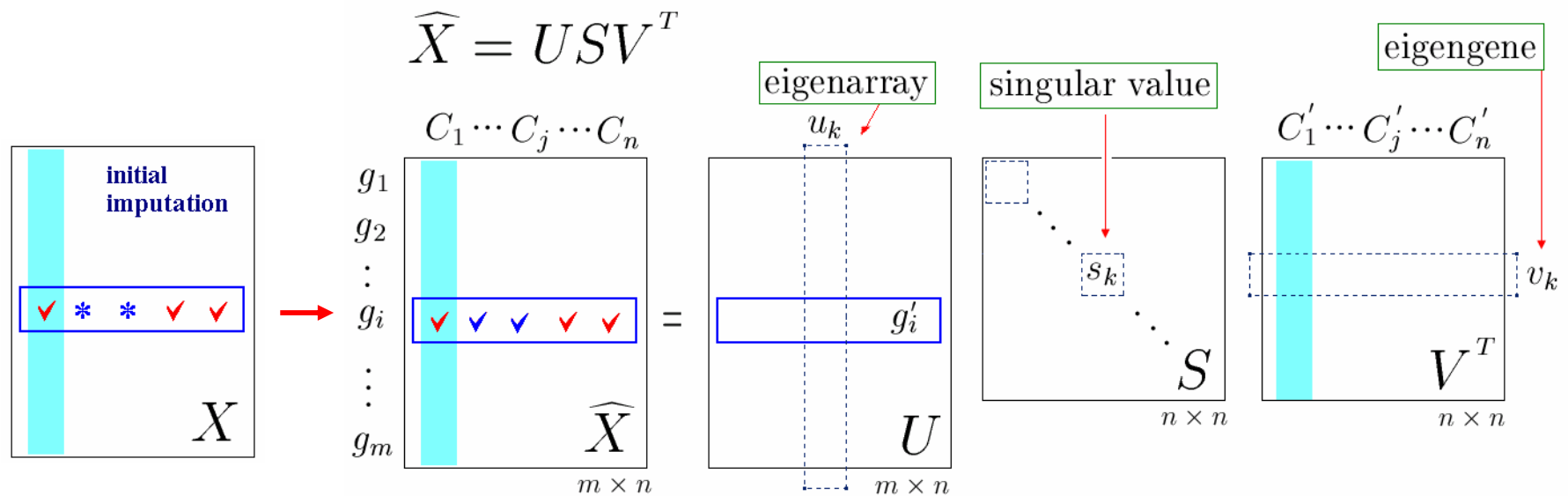
Average $C_1(\widehat{g}_1) = \frac{1}{K} \sum_{k=1}^K C_1(g_k)$

Weighted Average $C_1(\widehat{g}_1) = \frac{\sum_{k=1}^K w_k C_1(g_k)}{\sum_{k=1}^K w_k}$

$$w_k = \frac{1}{\sum_{i \in C} [C_j(g_k) - C_1(g_1)]^2}$$

- Euclidean distance measure is often **sensitive to outliers**, which could be present in microarray data.
- Log-transformed data** seems to sufficiently reduce the effect of outliers on genes similarity determination.

Singular Value Decomposition Imputation



Could Extend to Iterative approach

- [Troyanskaya O, Cantor M, Sherlock G, Brown P, Hastie T, Tibshirani R, Botstein D, Altman RB. \(2001\), Missing value estimation methods for DNA microarrays. Bioinformatics 17\(6\), 520-525.](#)
- Trevor Hastie , Robert Tibshirani, Gavin Sherlock , Michael Eisen , Patrick Brown , David Botstein. (1999). [Imputing Missing Data for Gene Expression Arrays](#), Technical Report.

SVDimpute

Model:

$$g_i(C) = \beta_0 + \sum_{k=1}^K \beta_k v_{(k)}(C)$$

C : Observed C_i 's without missing values

Imputation:

$$g_i(\widehat{C}) = \hat{\beta}_0 + \sum_{k=1}^K \hat{\beta}_k v_{(k)}(\widehat{C})$$

Evaluation of Imputation Methods



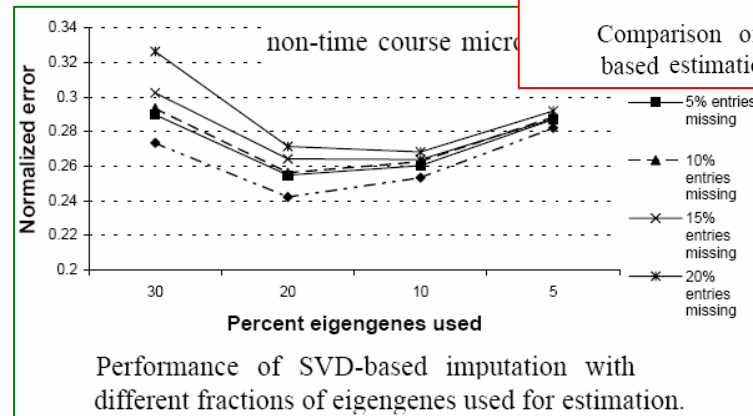
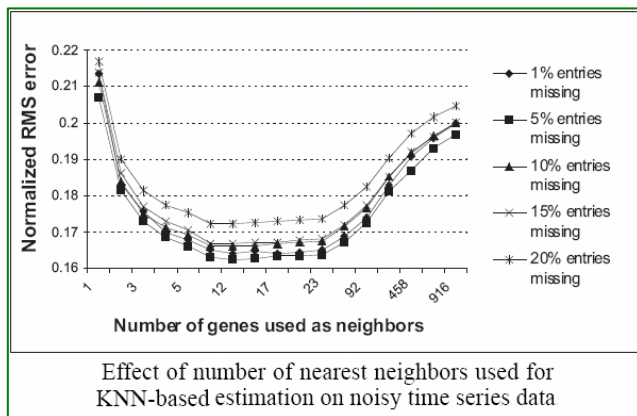
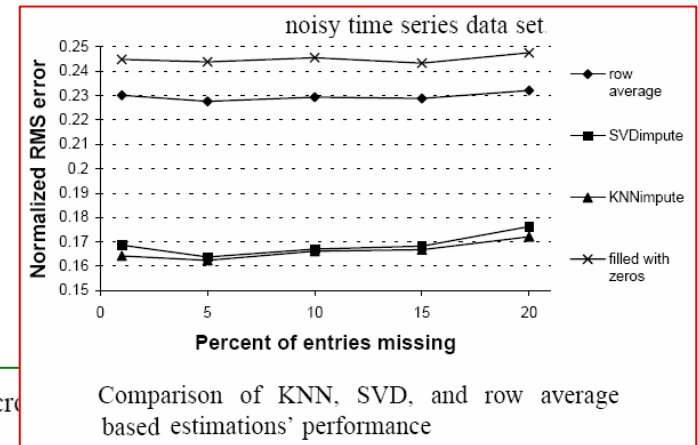
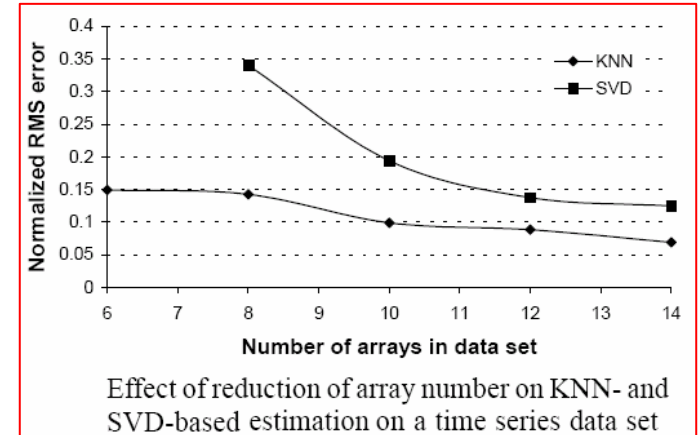
Troyanskaya O, Cantor M, Sherlock G, Brown P, Hastie T, Tibshirani R, Botstein D, Altman RB. (2001), Missing value estimation methods for DNA microarrays. *Bioinformatics* 17(6), 520-525.

Data sets:

- **Non-Time Series:** Gasch et al., (2000): 755 genes, 173 arrays.
- **Time Series:** DeRisi et al., (1997): 6135 genes, 7 arrays.
- **Noisy Times Series:** Spellman et al., (1998): 509 genes, 77 arrays.

Criteria: normalized root mean squared error (NRMSE)

$$\text{NRMSE} = \frac{\sqrt{\text{mean}[(y_{\text{guess}} - y_{\text{ans}})^2]}}{\text{std}[y_{\text{ans}}]}$$



Reference



Singular Value Decomposition Imputation

- Troyanskaya O, Cantor M, Sherlock G, Brown P, Hastie T, Tibshirani R, Botstein D, Altman RB. (2001), Missing value estimation methods for DNA microarrays. *Bioinformatics* 17(6), 520-525.
- Trevor Hastie , Robert Tibshirani, Gavin Sherlock , Michael Eisen , Patrick Brown , David Botstein. (1999). *Imputing Missing Data for Gene Expression Arrays*, Technical Report.

Local Least Square Imputation

- Bo TH, Dysvik B, Jonassen I. LSImpute: accurate estimation of missing values in microarray data with least squares methods. *Nucleic Acids Res.* 2004 Feb 20;32(3):e34.
- Hyunsoo Kimy, Gene H. Golubz, and Haesun Parky. (2004). *Missing Value Estimation for DNA Microarray Gene Expression Data: Local Least Squares Imputation*, *Bioinformatics Advance* Access published August 27, 2004.

Bayesian

- Oba S, Sato M-A, Takemasa I, Monden M, Matsubara K-I, Ishii S: A Bayesian missing value estimation method for gene expression profile data,. *Bioinformatics* 2003, 19:2088-2096.
- Zhou X, Wang X, Dougherty ER: Missing-value estimation using linear and non-linear regression with Bayesian gene selection. *Bioinformatics* 2003, 19:2302-2307.

GMCimpute

- Ouyang M, Welsh WJ, Georgopoulos P. Gaussian mixture clustering and imputation of microarray data. *Bioinformatics.* 2004 Apr 12;20(6):917-23. Epub 2004 Jan 29.

Others

- Kim KY, Kim BJ, Yi GS. Reuse of imputed data in microarray analysis increases imputation efficiency. *BMC Bioinformatics.* 2004 Oct 26;5(1):160.
- Shmuel Friedland, Amir Niknejad, and Laura Chiharaz. (2004). *A Simultaneous Reconstruction of Missing Data in DNA Microarrays*, Institute for Mathematics and its Applications,.
- Alexandre G de Brevern, Serge Hazout and Alain Malpertuy. (2004). Influence of microarrays experiments missing values on the stability of gene groups by hierarchical clustering, *BMC Bioinformatics* Volume 5.