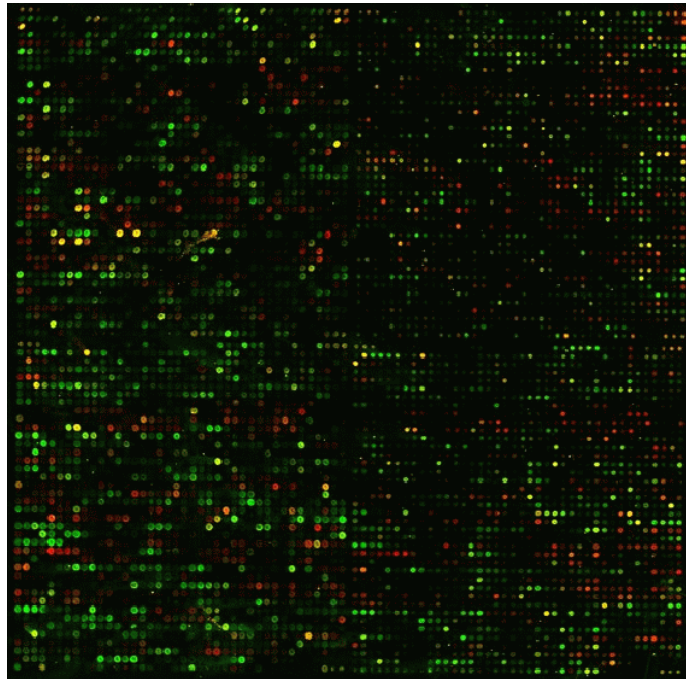


# Pre-processing: spotted DNA microarrays



# Terminology

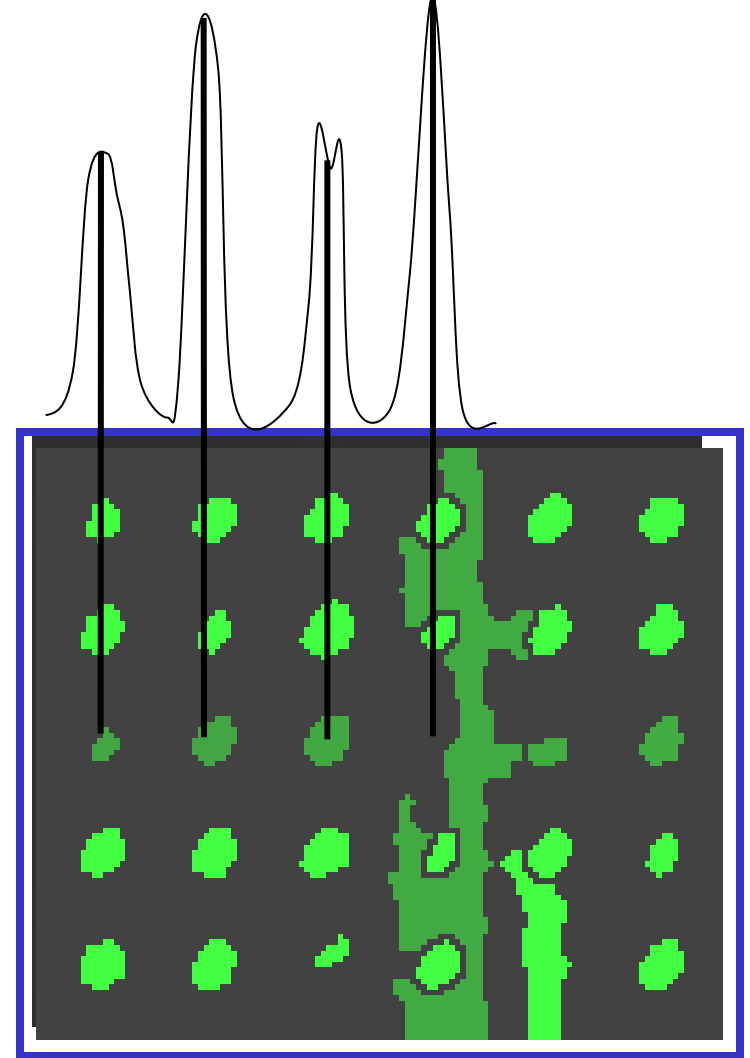
- **Target:** DNA hybridized to the array, mobile substrate.
- **Probe:** DNA spotted on the array, aka. spot, immobile substrate.
- **Sector:** collection of spots printed using the same print-tip (or pin), aka. **print-tip-group**, pin-group, spot matrix, grid.
- The terms **slide** and **array** are often used to refer to the printed microarray.
- **Batch:** collection of microarrays with the same probe layout.
- **Cy3 = Cyanine 3 = green dye.**
- **Cy5 = Cyanine 5 = red dye.**

# Image analysis

- The **raw data** from a cDNA microarray experiment consist of pairs of **image files**, 16-bit TIFFs, one for each of the dyes.
- **Image analysis** is required to extract measures of the red and green fluorescence intensities, **R** and **G**, for each spot on the array.

# Image analysis

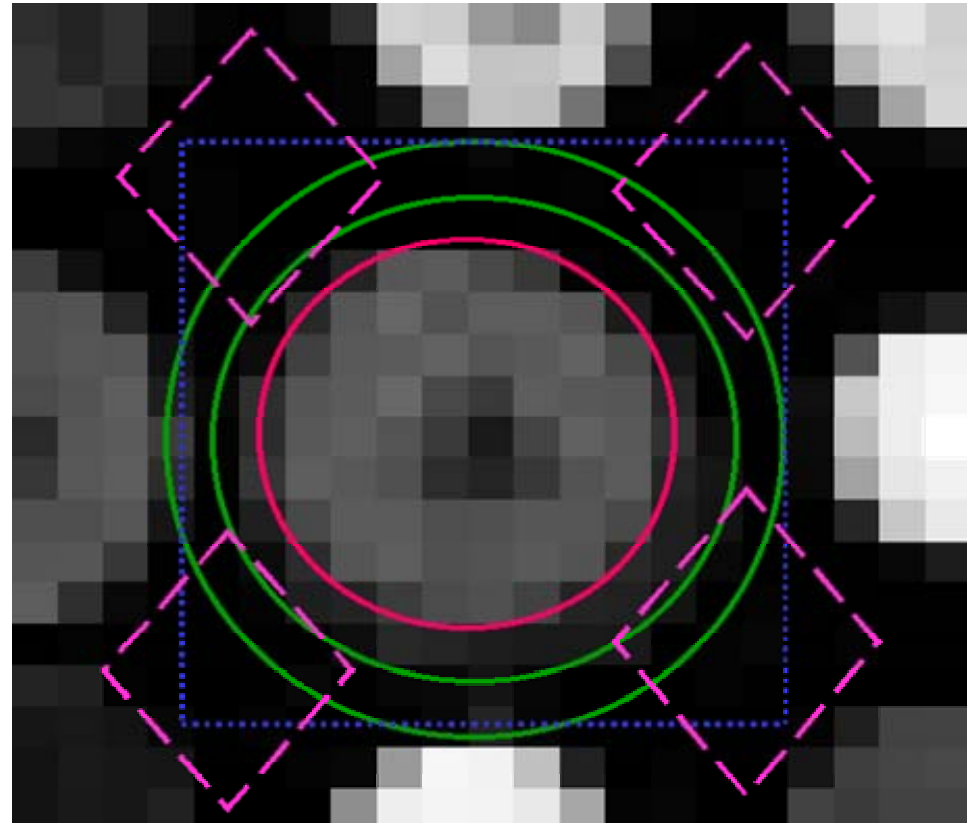
- 1. Addressing.** Estimate location of spot centers.
- 2. Segmentation.** Classify pixels as foreground (signal) or background.
- 3. Information extraction.** For each spot on the array and each dye
  - foreground intensities;
  - background intensities;
  - quality measures.



→ **R** and **G** for each spot on the array.

# Local background

- GenePix
- QuantArray
- ScanAnalyze

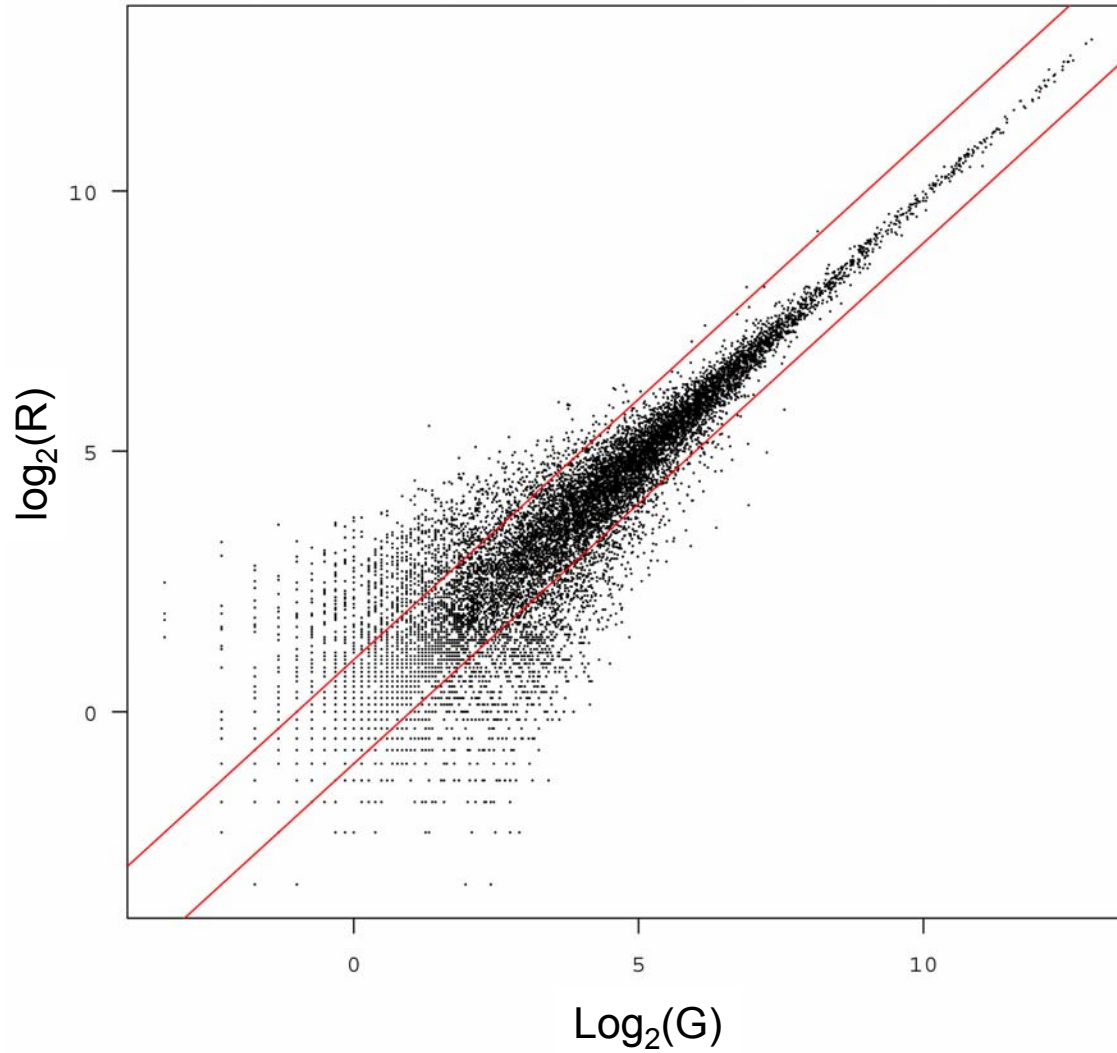


**Spot uses Morphological opening**

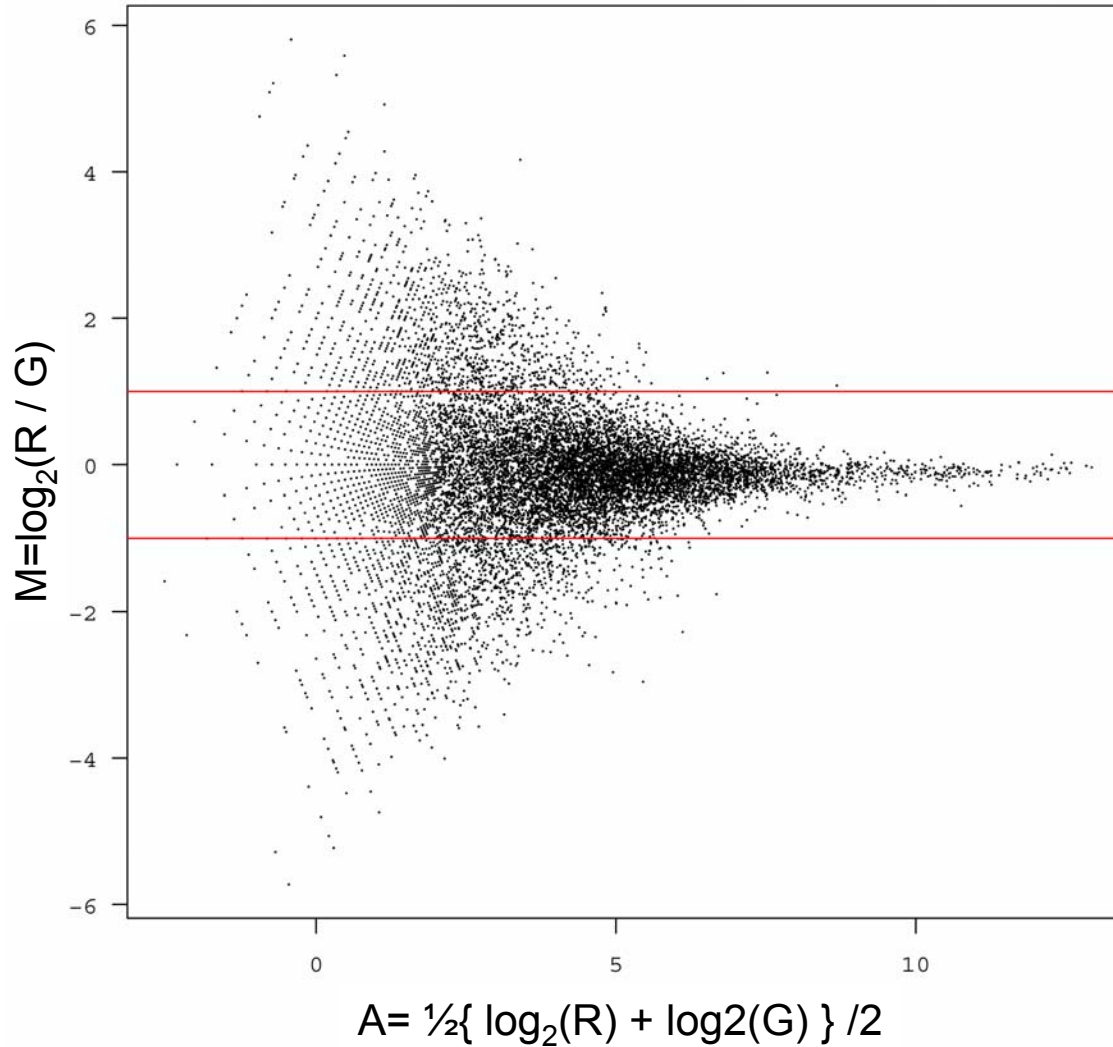
# Single-slide data display

- Usually: R vs. G  
 $\log_2 R$  vs.  $\log_2 G$ .
- Preferred  
 $M = \log_2 R - \log_2 G$   
vs.  $A = (\log_2 R + \log_2 G)/2$ .
- An MA-plot amounts to a  $45^\circ$  clockwise rotation of a  $\log_2 R$  vs.  $\log_2 G$  plot followed by scaling.

# RvG Plot



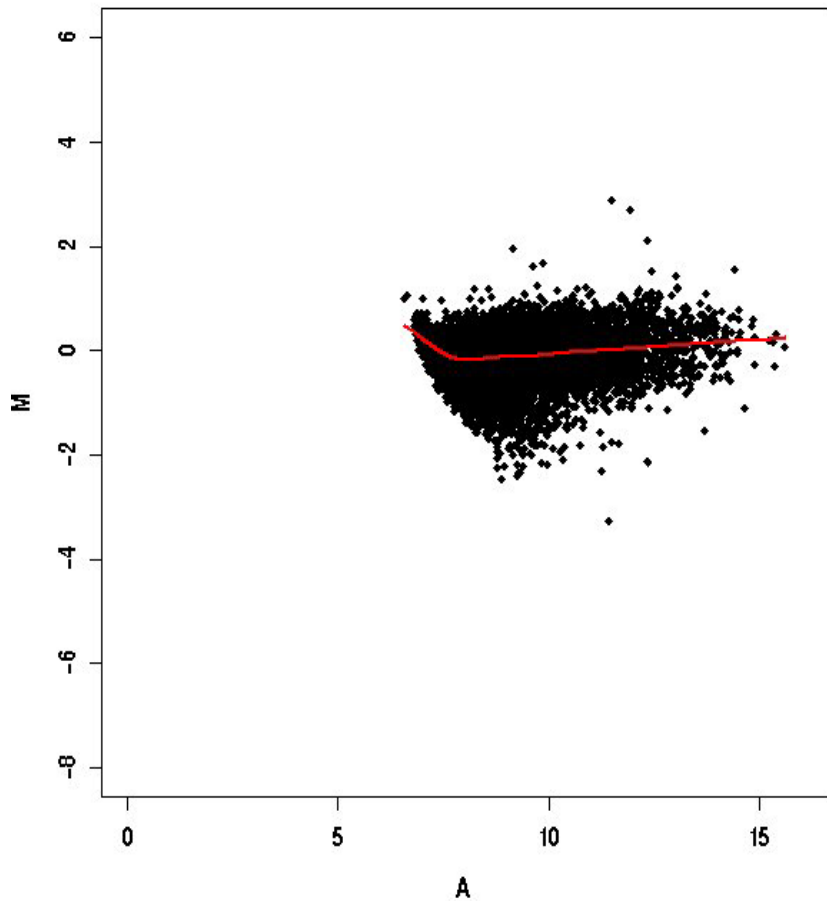
# MvA Plot



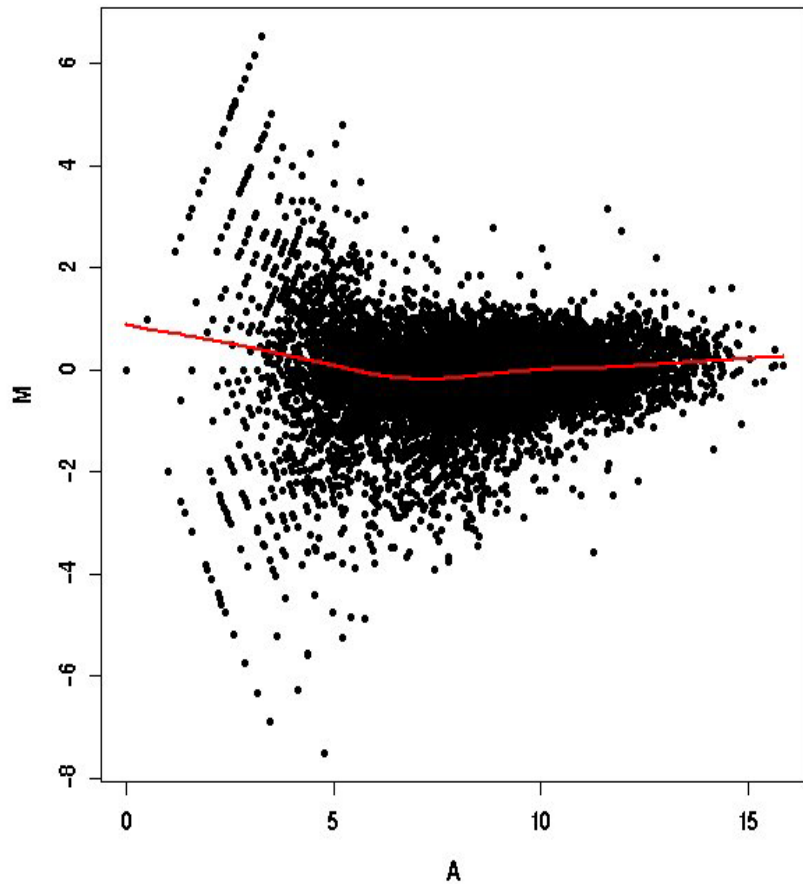


# Background matters

Morphological opening



Local background



$M = \log_2 R - \log_2 G$  vs.  $A = (\log_2 R + \log_2 G)/2$

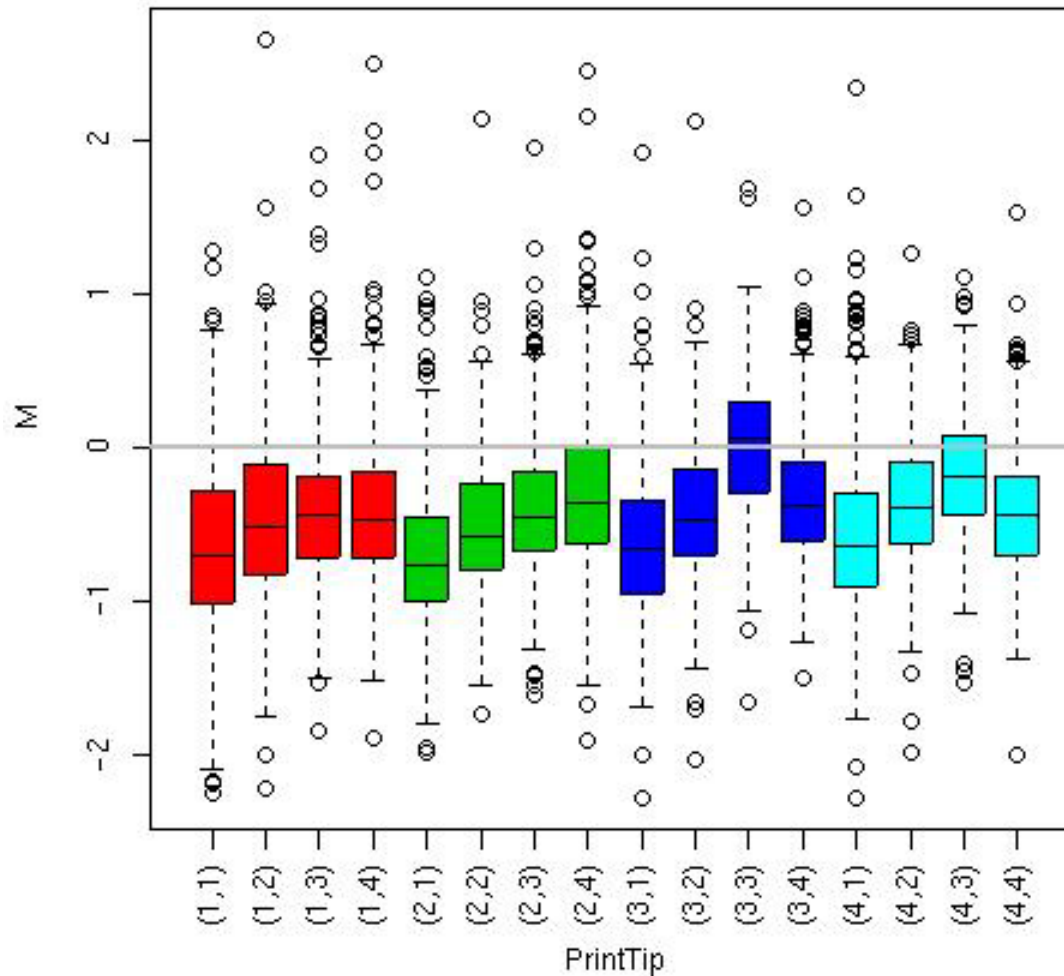
# Diagnostic plots

- **Diagnostics plots** of spot statistics  
E.g. red and green log-intensities, intensity log-ratios  $M$ , average log-intensities  $A$ , spot area.
  - Boxplots;
  - 2D spatial images;
  - Scatter-plots, e.g. MA-plots;
  - Density plots.
- **Stratify** plots according to layout parameters, e.g. print-tip-group, plate.

# Boxplots by print-tip-group

Swirl 93 array: pre-normalization log-ratio M

Intensity  
log-ratio, M

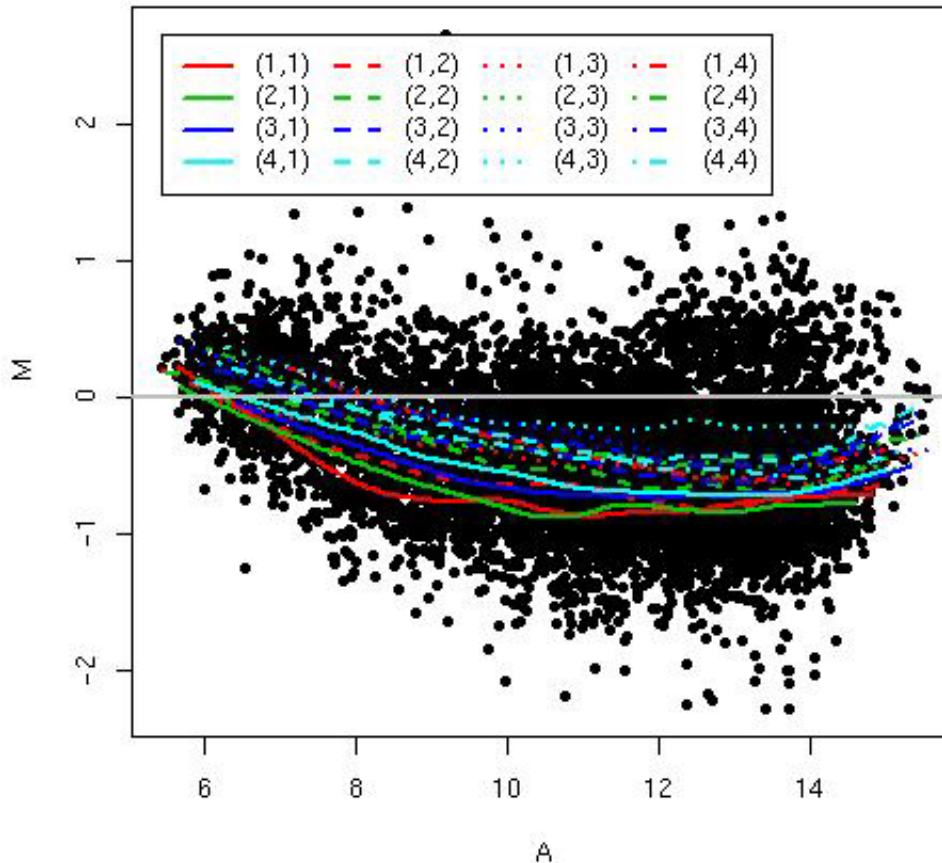


# MA-plot by print-tip-group

$$M = \log_2 R - \log_2 G, \quad A = (\log_2 R + \log_2 G)/2$$

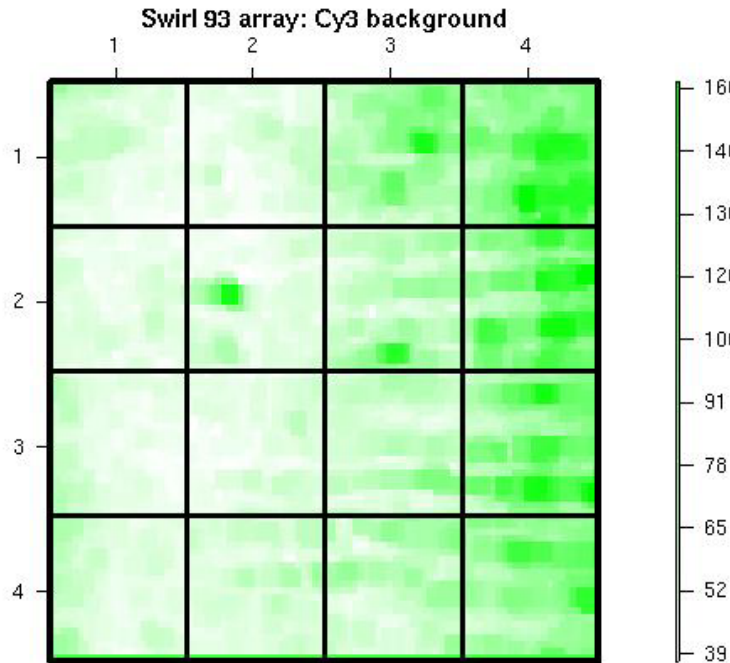
Swirl 93 array: pre-normalization log-ratio M

Intensity  
log-ratio, M

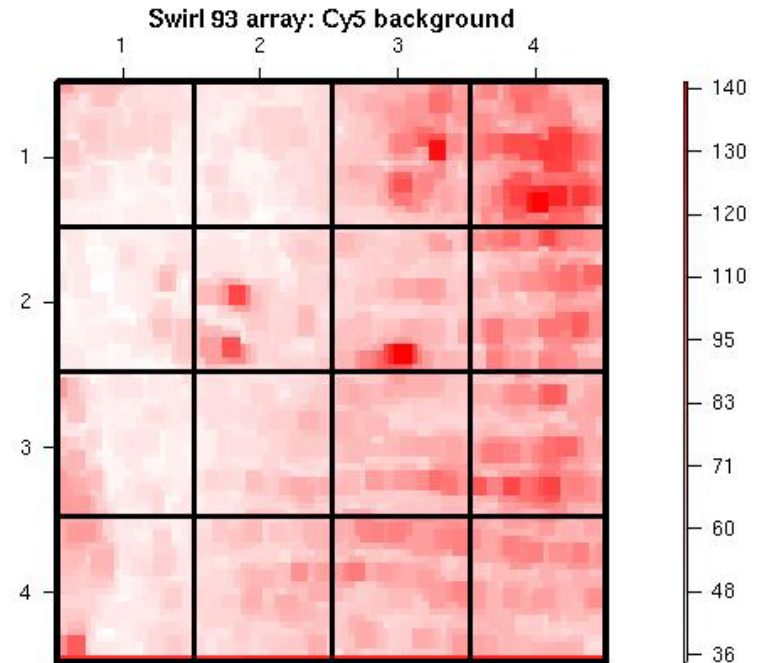


Average  
log-intensity, A

# 2D spatial images



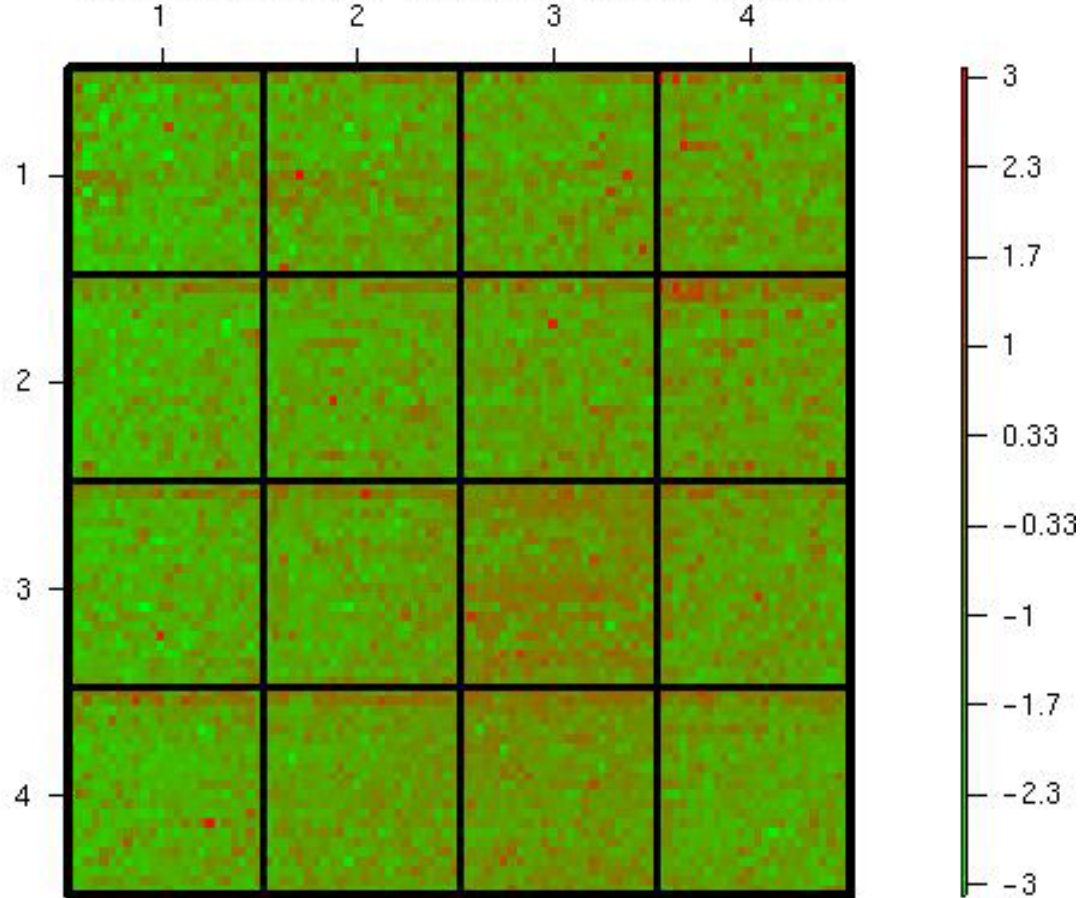
**Cy3 background intensity**



**Cy5 background intensity**

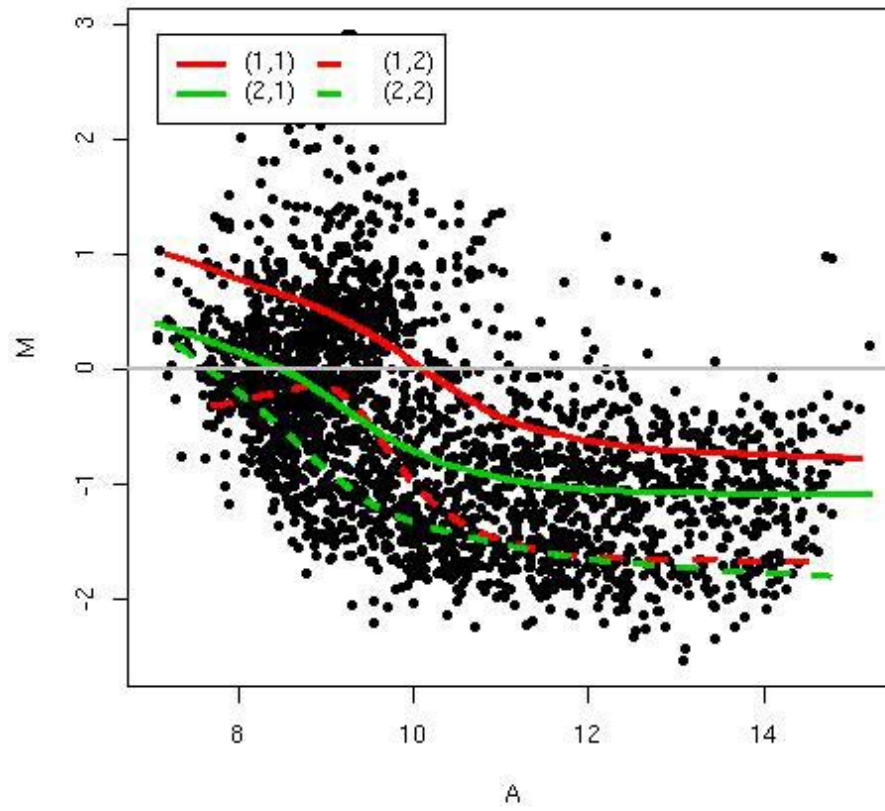
# 2D spatial images

Swirl 93 array: pre-normalization log-ratio M



Intensity  
log-ratio, M

# Normalization



# Normalization

- After image processing, we have measures of the red and green fluorescence intensities, **R** and **G**, for each spot on the array.
- **Normalization** is needed to ensure that differences in intensities are indeed due to differential expression, and not some printing, hybridization, or scanning artifact.
- Normalization is necessary before any analysis which involves within or between slides comparisons of intensities, e.g., clustering, testing.



# Normalization

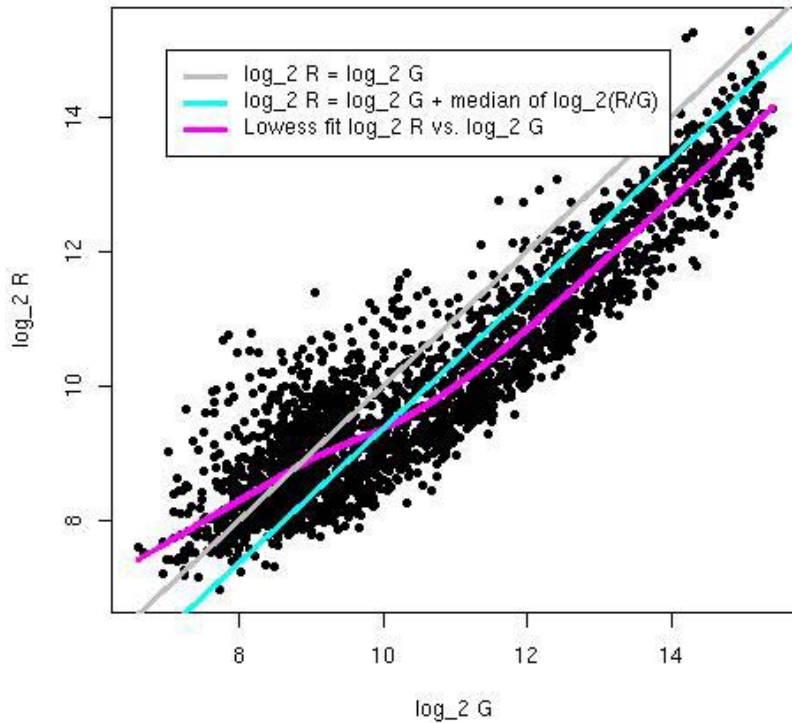
- Identify and remove the effects of **systematic variation** in the measured fluorescence intensities, other than differential expression, for example
  - different labeling efficiencies of the dyes;
  - different amounts of Cy3- and Cy5-labeled mRNA;
  - different scanning parameters;
  - print-tip, spatial, or plate effects, etc.

# Normalization

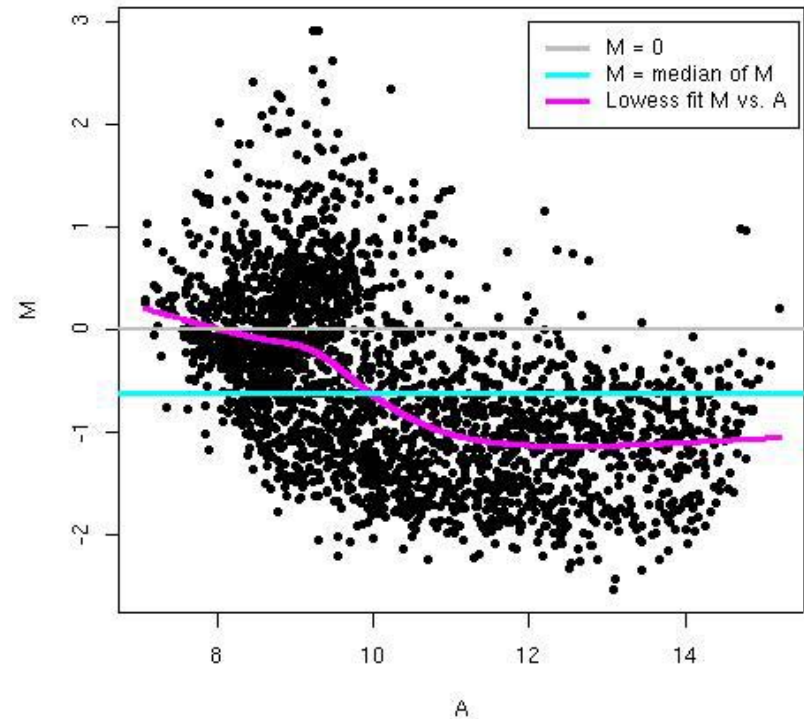
- The need for normalization can be seen most clearly in **self-self hybridizations**, where the same mRNA sample is labeled with the Cy3 and Cy5 dyes.
- The imbalance in the red and green intensities is usually **not constant** across the spots within and between arrays, and can vary according to overall spot intensity, location, plate origin, etc.
- These factors should be considered in the normalization.

# Self-self hybridization

## $\log_2 R$ vs. $\log_2 G$



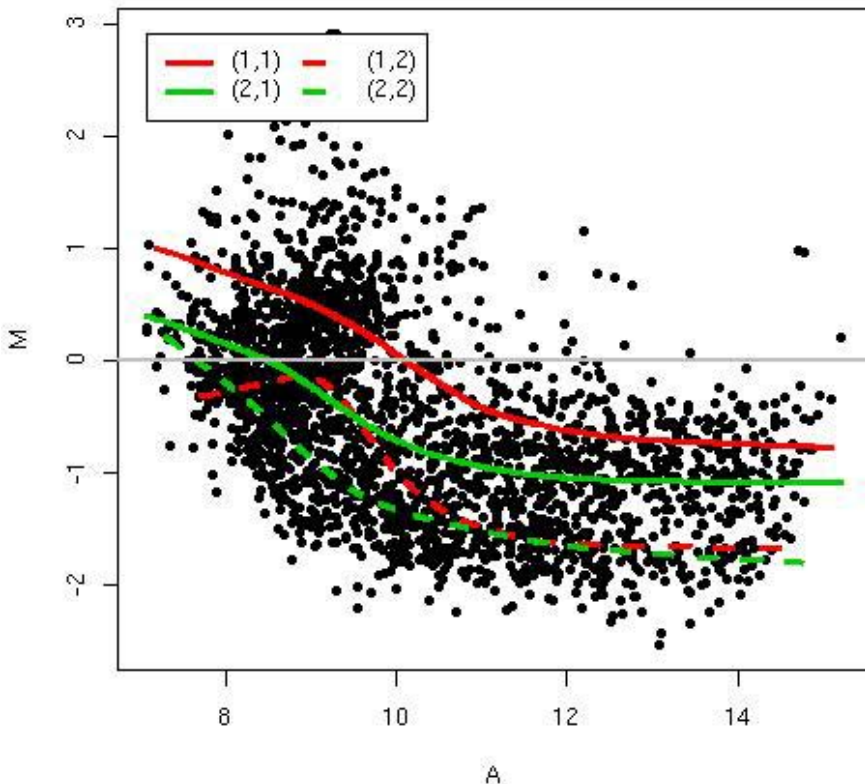
## M vs. A



$$M = \log_2 R - \log_2 G, \quad A = (\log_2 R + \log_2 G)/2$$

# Self-self hybridization

**M vs. A**



Robust local regression  
within sectors  
(print-tip-groups)  
of intensity log-ratio M  
on average log-intensity  
A.

$$M = \log_2 R - \log_2 G, \quad A = (\log_2 R + \log_2 G)/2$$

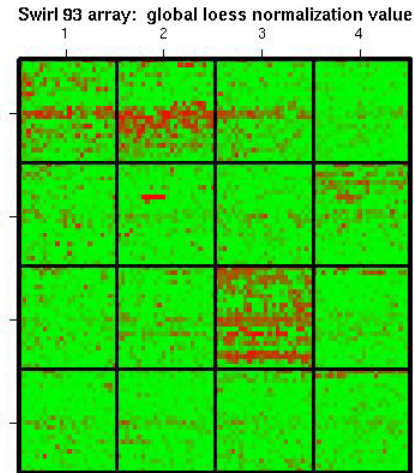
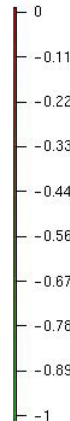
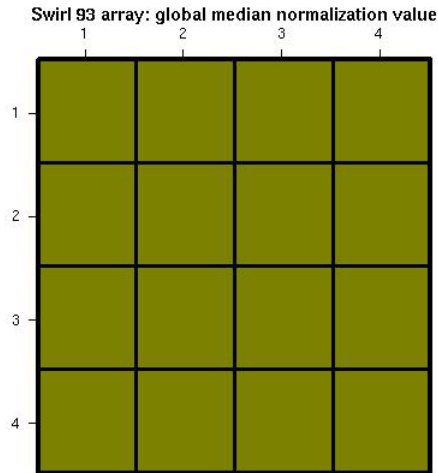
# Example of Normalization

$$\log_2 R/G \leftarrow \log_2 R/G - L(\text{intensity, sector, ...})$$

- **Constant normalization:** L is constant
- **Adaptive normalization:** L depends on a number of predictor variables, such as spot intensity A, sector, plate origin.
  - Intensity-dependent normalization.
  - Intensity and sector-dependent normalization.
  - 2D spatial normalization.
  - Other variables: time of printing, plate, etc.
  - Composite normalization. Weighted average of several normalization functions.

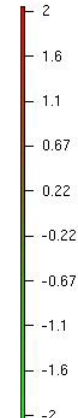
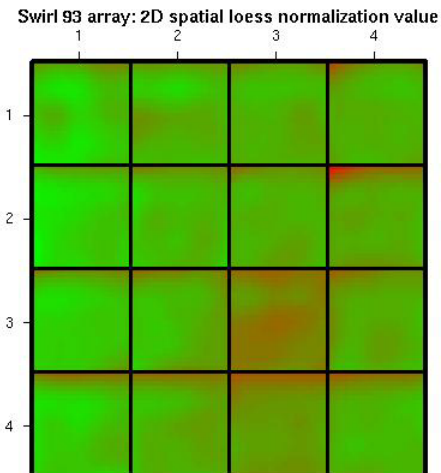
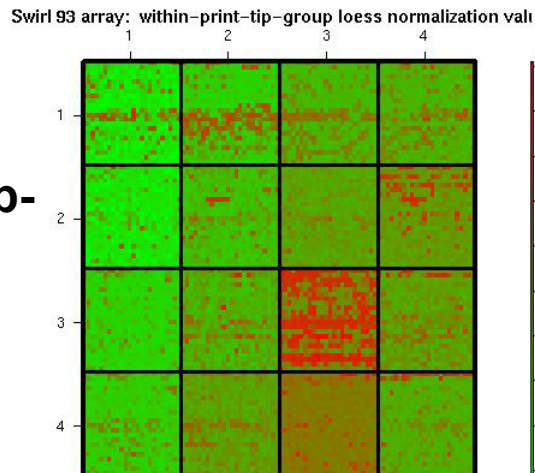
# 2D images of L values

**Global median normalization**



**Global loess normalization**

**Within-print-tip-group loess normalization**

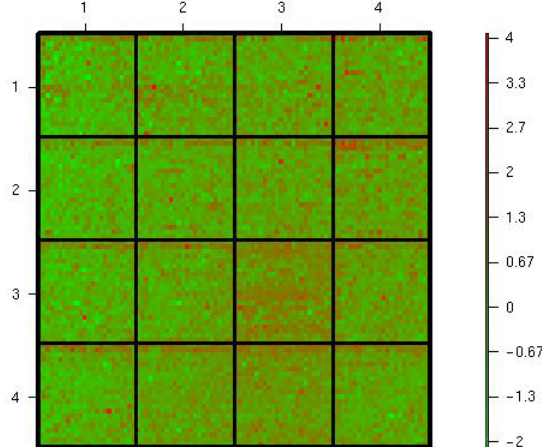


**2D spatial normalization**

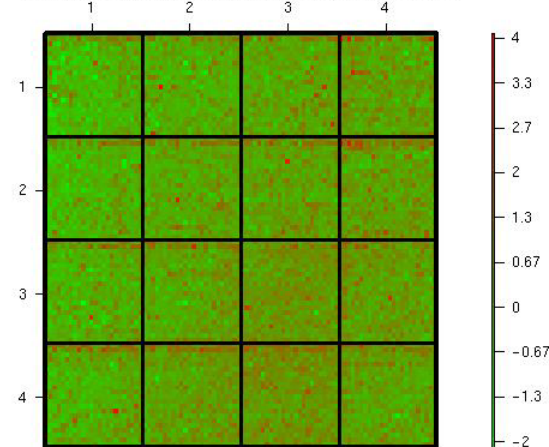
# 2D images of normalized M-L

**Global median normalization**

Swirl 93 array: global median normalization log-ratio M



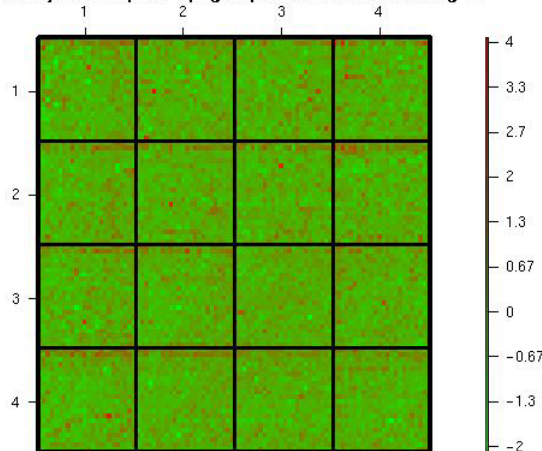
Swirl 93 array: global loess normalization log-ratio M



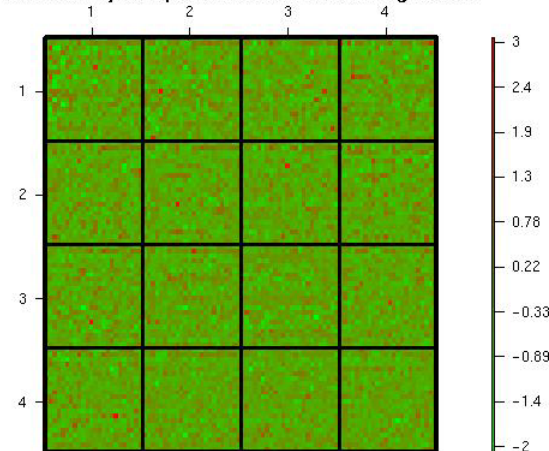
**Global loess normalization**

**Within-print-tip-group loess normalization**

Swirl 93 array: within-print-tip-group loess normalization log-ratio M



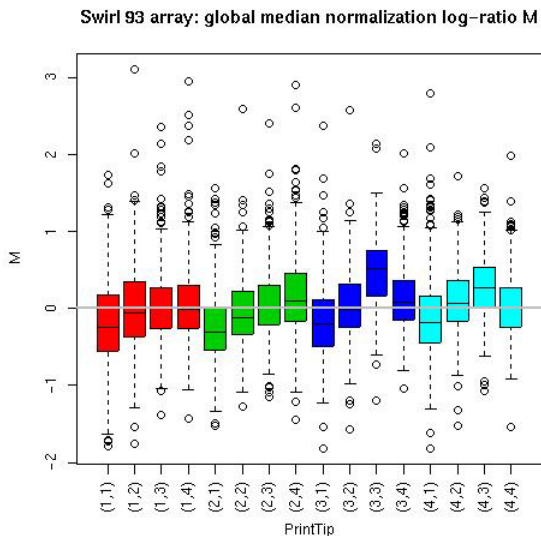
Swirl 93 array: 2D spatial loess normalization log-ratio M



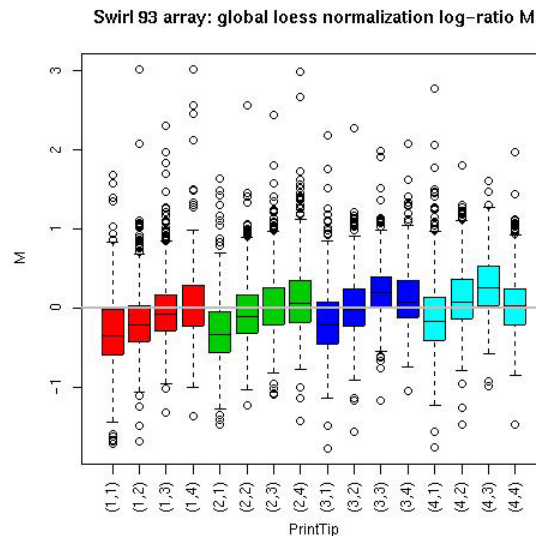
**2D spatial normalization**

# Boxplots of normalized M-L

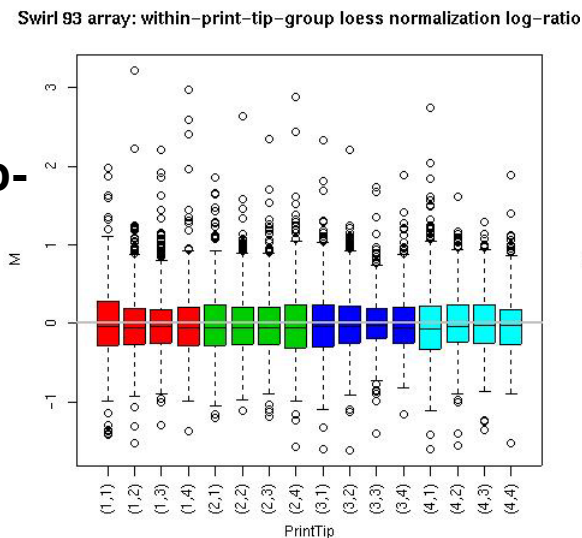
Global median normalization



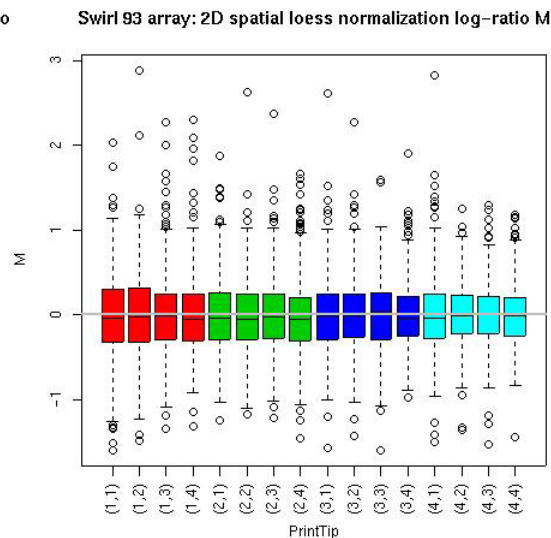
Global loess normalization



Within-print-tip-group loess normalization



2D spatial normalization

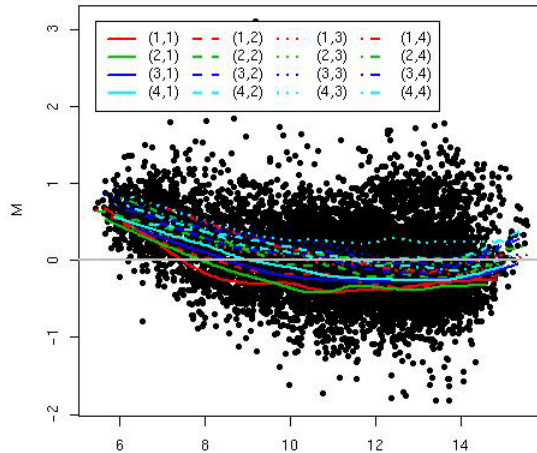




# MA-plots of normalized M-L

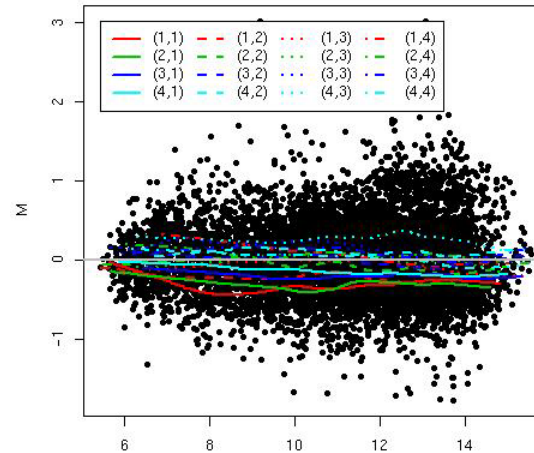
**Global median normalization**

Swirl 93 array: global median normalization log-ratio M



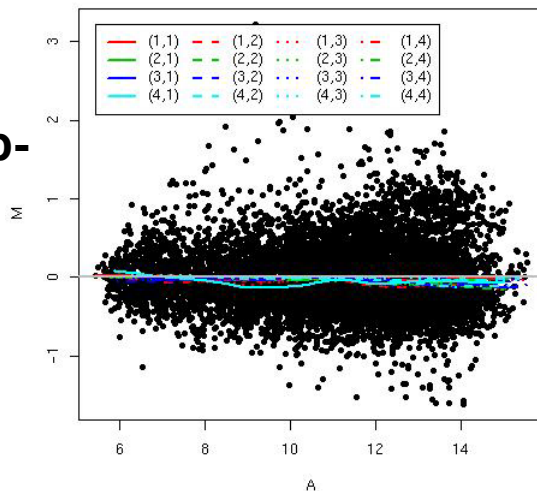
**Global loess normalization**

Swirl 93 array: global loess normalization log-ratio M



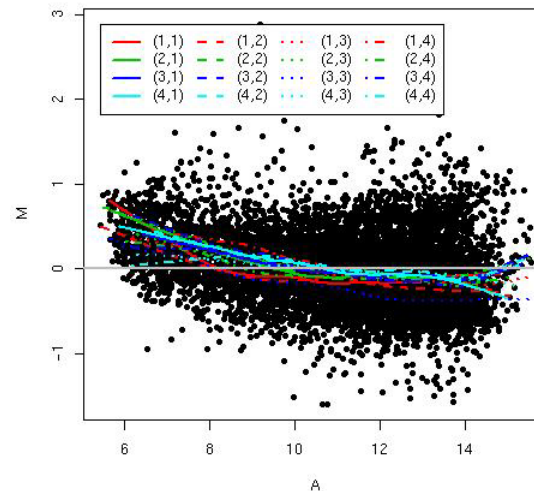
**Within-print-tip-group loess normalization**

Swirl 93 array: within-print-tip-group loess normalization log-ratio



**2D spatial normalization**

Swirl 93 array: 2D spatial loess normalization log-ratio M



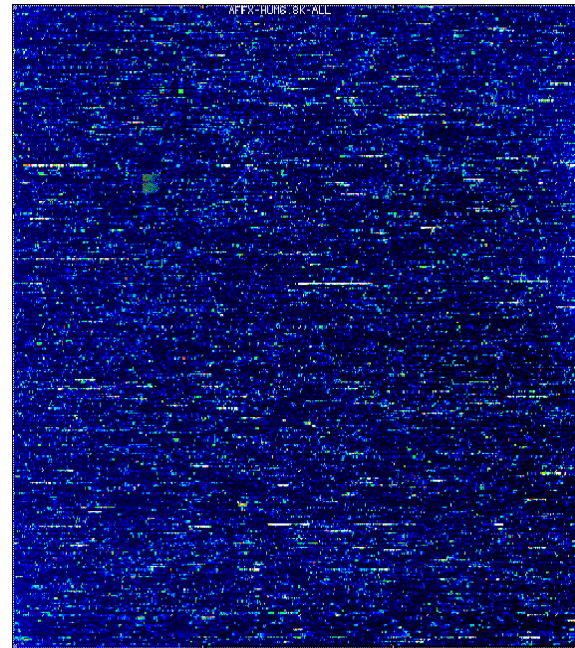
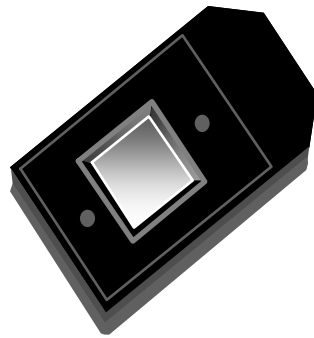
# Some References

- Dudoit, Yang, Callow, and Speed: *Statistica Sinica* (2002)
- Dudoit and Yang (2002) Chap 2 in *The Analysis of Gene Expression Data*
- Yang, Buckley, Dudoit, and Speed: *JCGS* (2002)
- Kerr and Churchill: *Biostatistics* (2001)
- Colantuoni, Henry, Zeger, and Pevsner: *Bioinformatics* (2002)

# **marray**: Pre-processing spotted DNA microarray data

- **marrayClasses**:
  - class definitions for cDNA microarray data (MIAME);
  - basic methods for manipulating microarray objects: printing, plotting, subsetting, class conversions, etc.
- **marrayInput**:
  - reading in intensity data and textual data describing probes and targets;
  - automatic generation of microarray data objects;
  - widgets for point & click interface.
- **marrayPlots**: diagnostic plots.
- **marrayNorm**: robust adaptive location and scale normalization procedures.

# Pre-processing: oligonucleotide chips



# Probe-pair set

## GeneChip® Expression Array Design

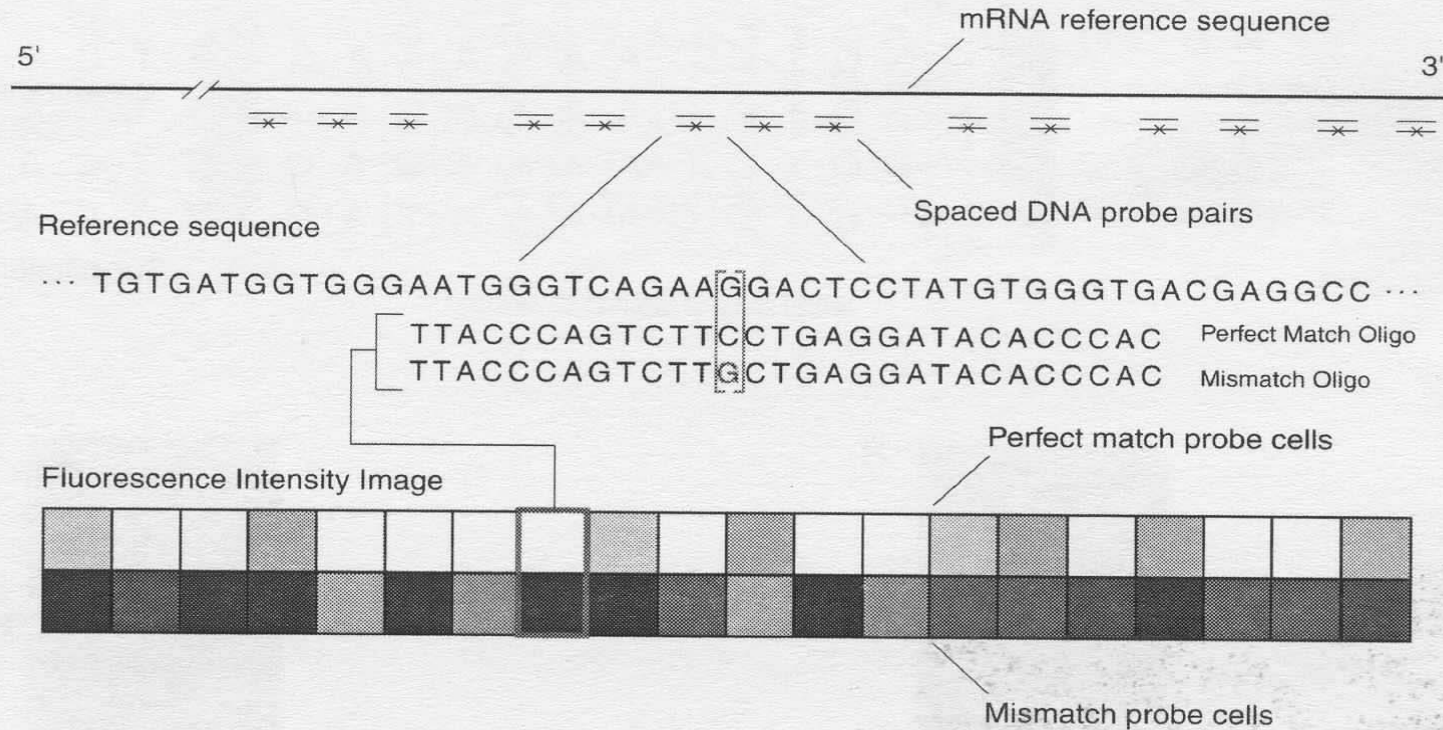
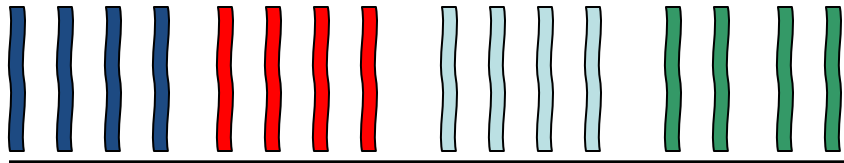
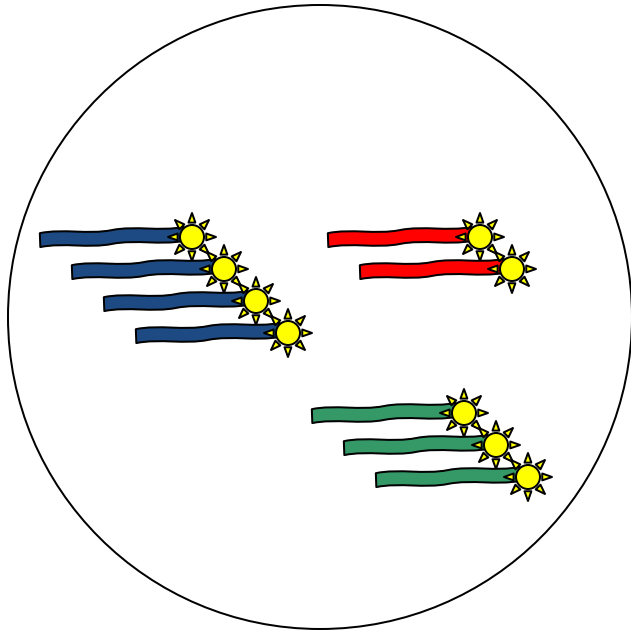


Figure 1-3 Expression tiling strategy

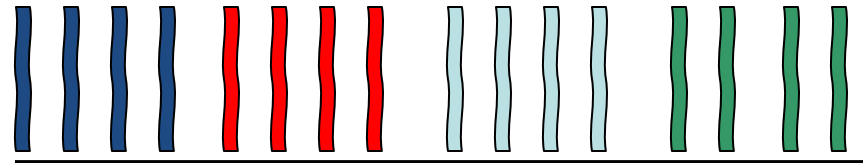
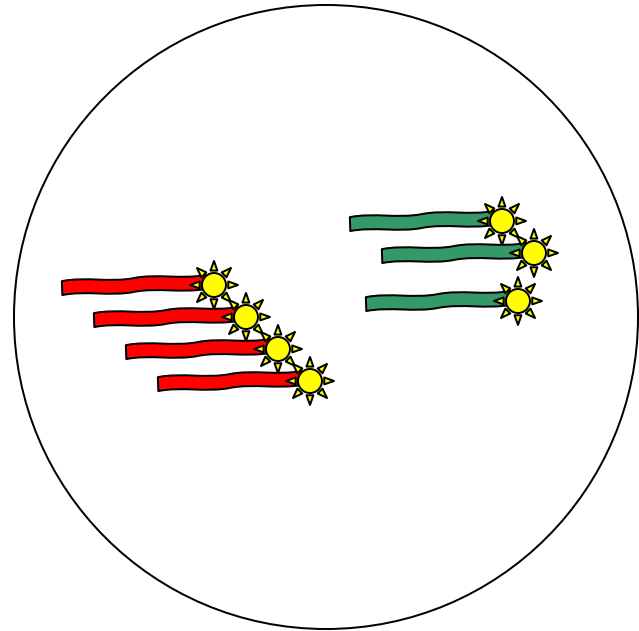
# Before Hybridization

Sample 1



Array 1

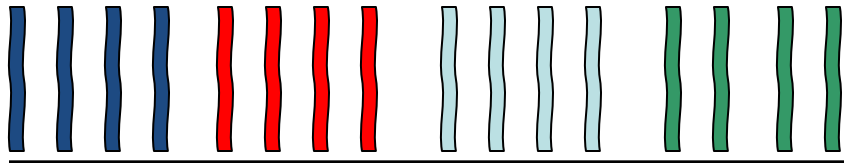
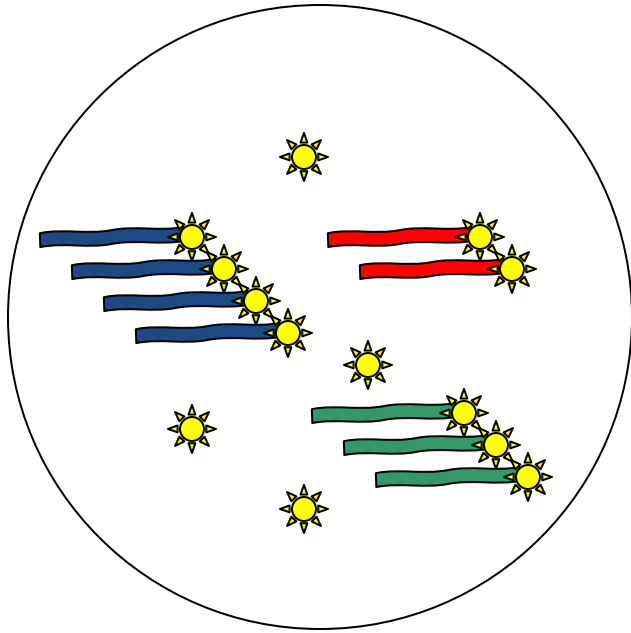
Sample 2



Array 2

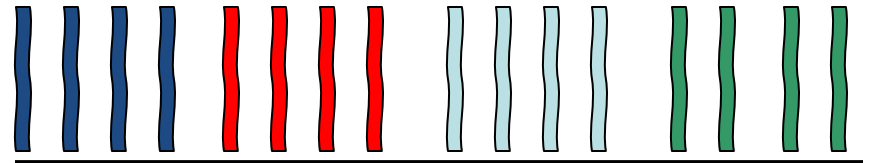
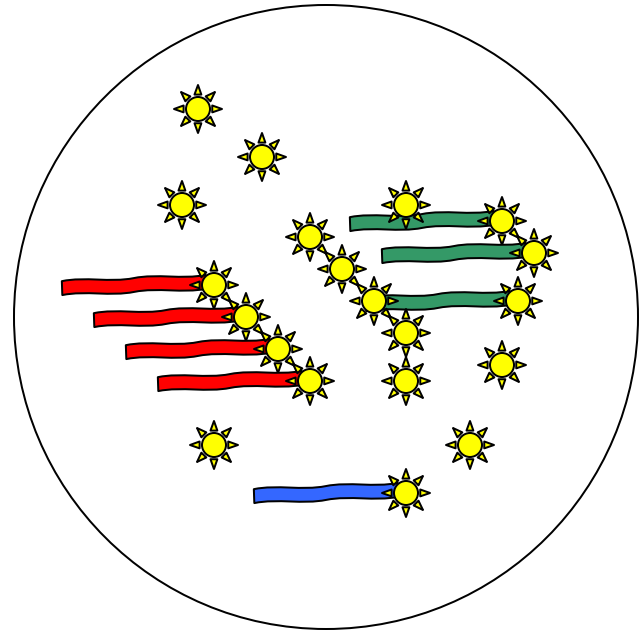
# More Realistic

Sample 1



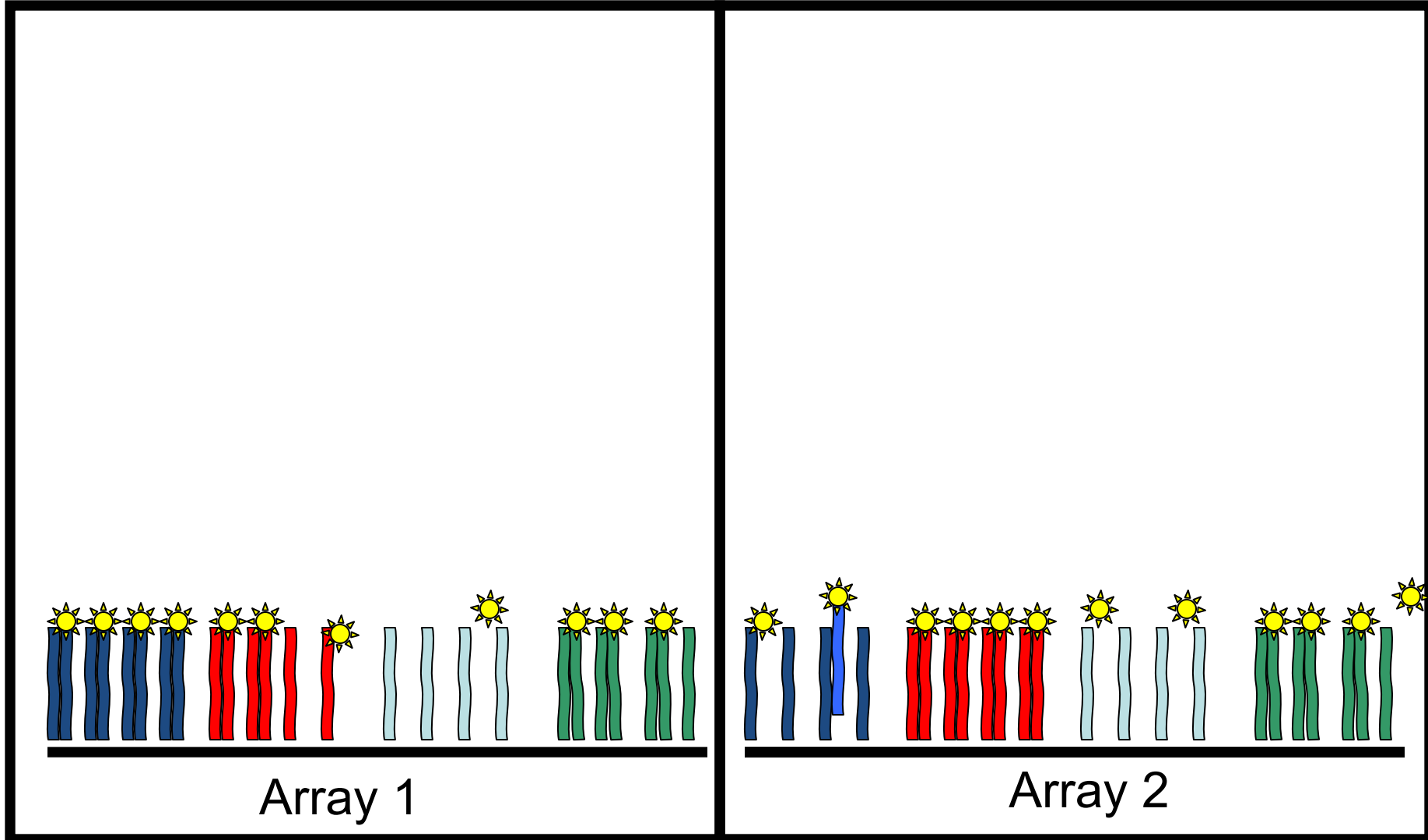
Array 1

Sample 2



Array 2

# Non-specific Hybridization





# GeneChip® Expression Array Design

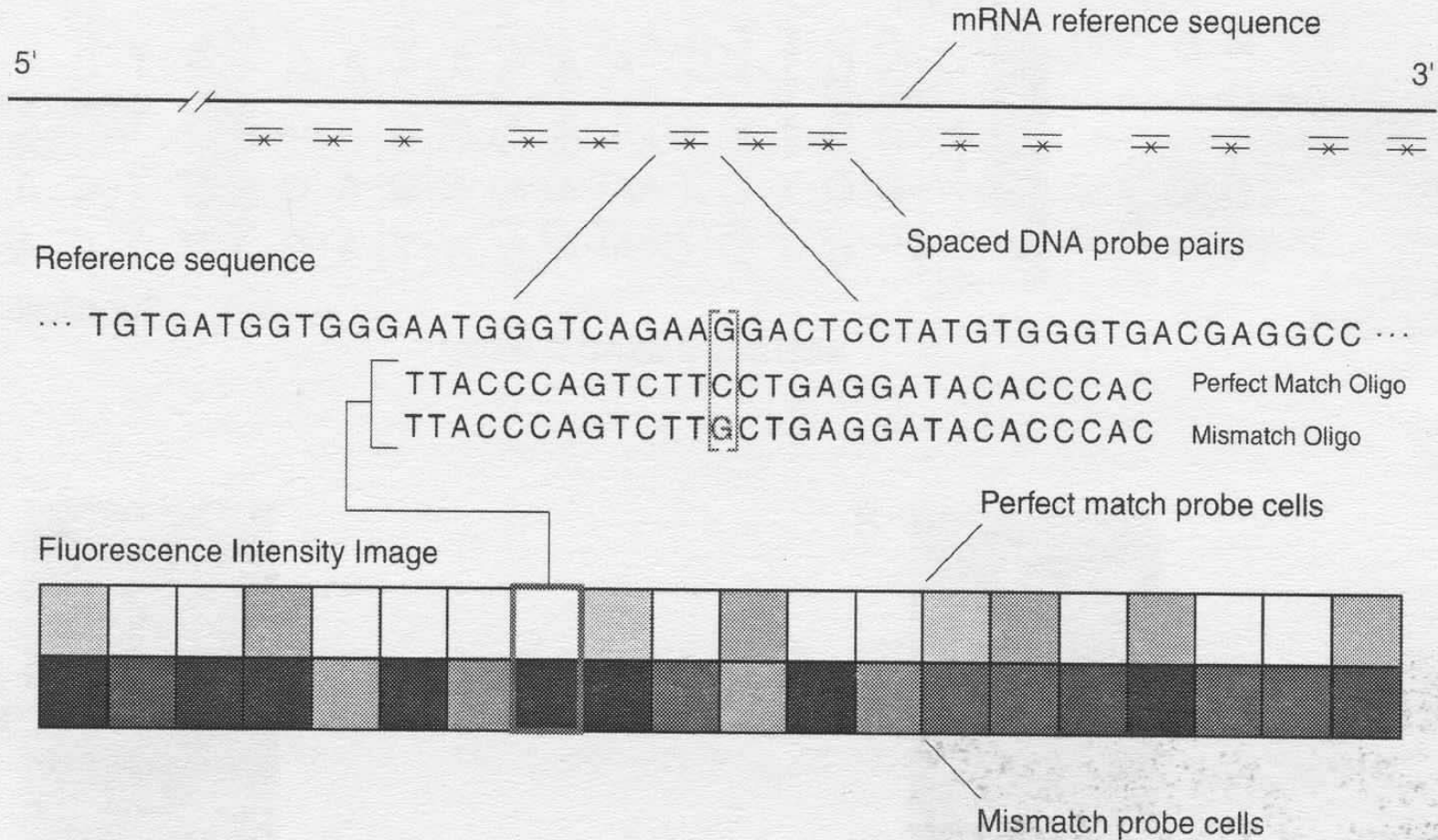


Figure 1-3 Expression tiling strategy

# Terminology

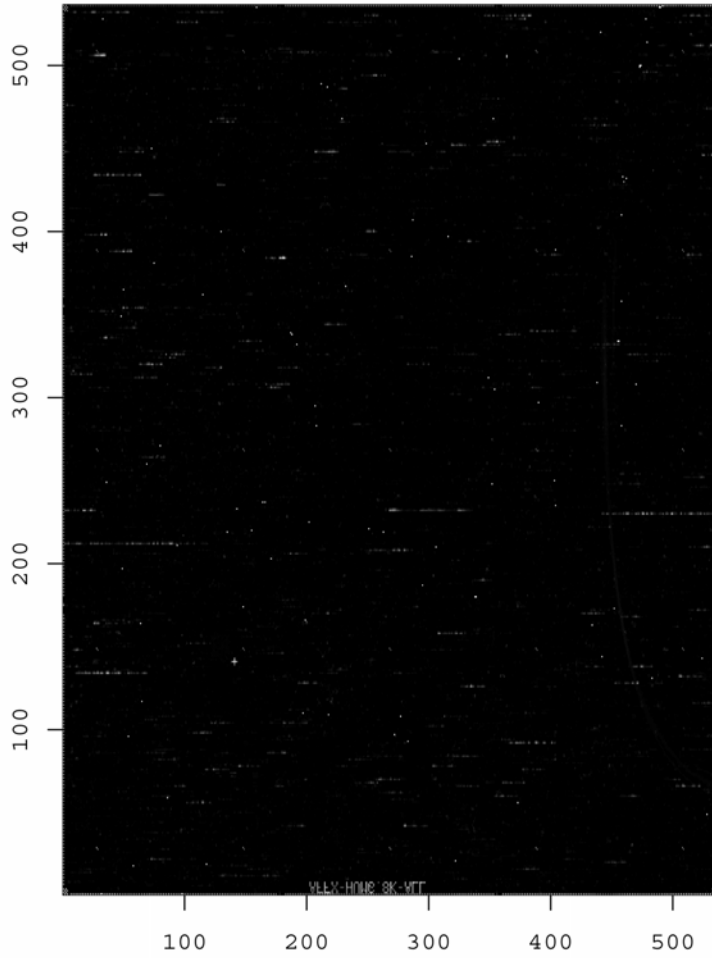
- Each gene or portion of a gene is represented by 16 to 20 oligonucleotides of 25 base-pairs.
- **Probe**: an oligonucleotide of 25 base-pairs, i.e., a 25-mer.
- **Perfect match (PM)**: A 25-mer complementary to a reference sequence of interest (e.g., part of a gene).
- **Mismatch (MM)**: same as PM but with a single homomeric base change for the middle (13<sup>th</sup>) base (transversion purine <-> pyrimidine, G <->C, A <->T) .
- **Probe-pair**: a (PM,MM) pair.
- **Probe-pair set**: a collection of probe-pairs (11 to 20) related to a common gene or fraction of a gene.
- **Affy ID**: an identifier for a probe-pair set.
- The purpose of the MM probe design is to measure non-specific binding and background noise.

# Why Analyze Probe Level Data?

- Quality control
  - Spatial Effects
  - RNA degradation (Leslie Cope)
- Detection of defective probes
- Transcript sequence “estimates” change
- Ways to reduce to expression measure  
keep improving

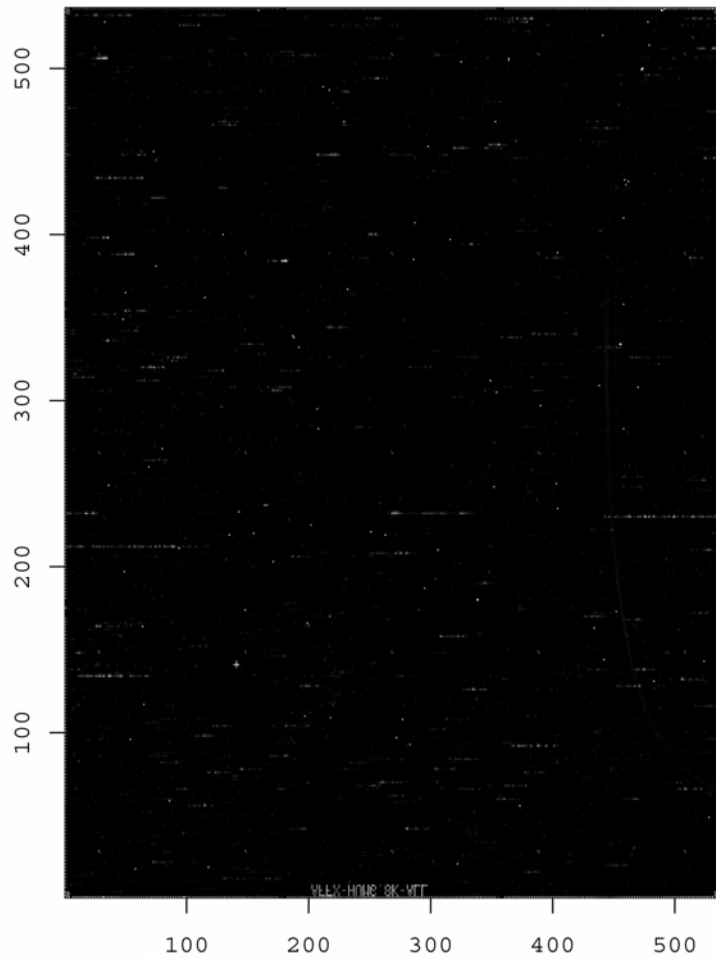
# QC

raw values

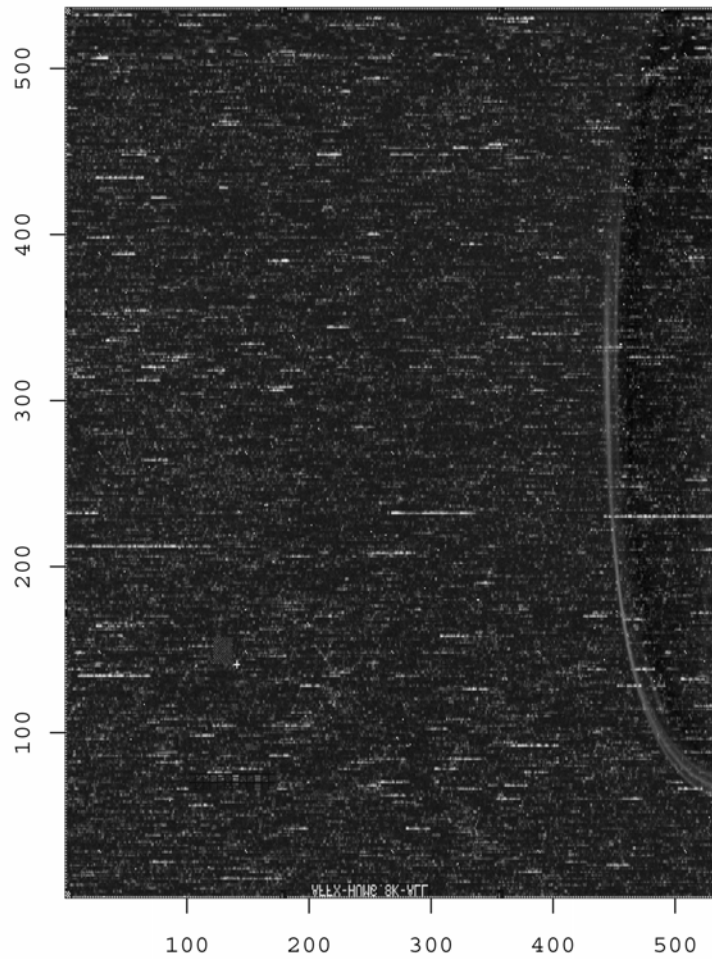


# QC

raw values



log2-transformed values



# Statistical Problem

- Each gene is represented by 20 pairs (PM and MM) of probe intensities
- Each array has 8K-20K genes
- Usually there are various arrays
- Obtain measure for each gene on each array:  
Summarize 20 pairs
- Background correction and normalization are issues

# Default until 2002 (MAS 4.0)

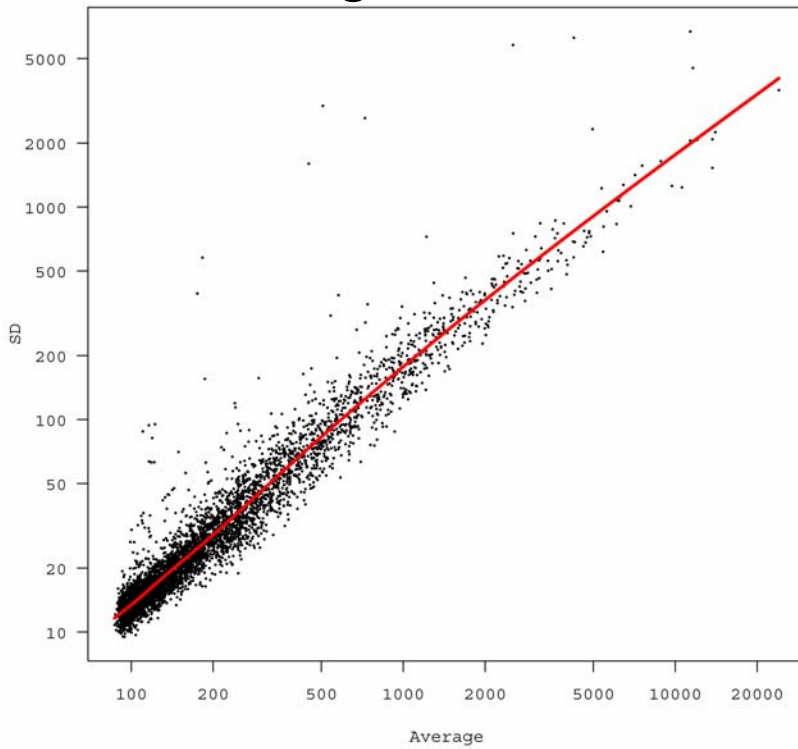
- GeneChip<sup>®</sup> software used *Avg.diff*

$$Avg.diff = \frac{1}{|A|} \sum_{j \in A} (PM_j - MM_j)$$

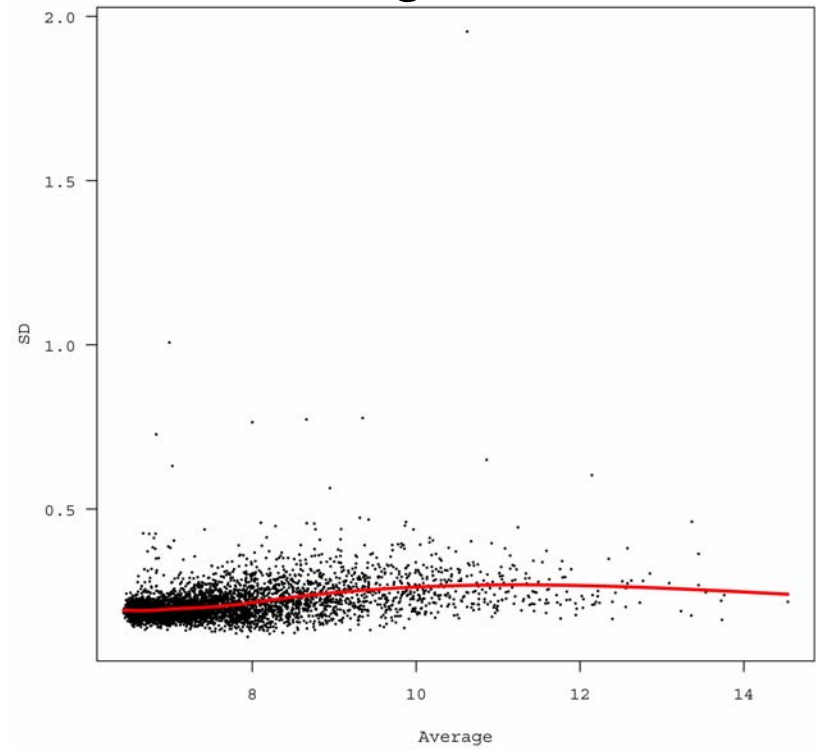
- with A a set of “suitable” pairs chosen by software.
- Obvious Problems:
  - Many negative expression values
  - No log transform

# Why use log?

## Original Scale



## Log Scale





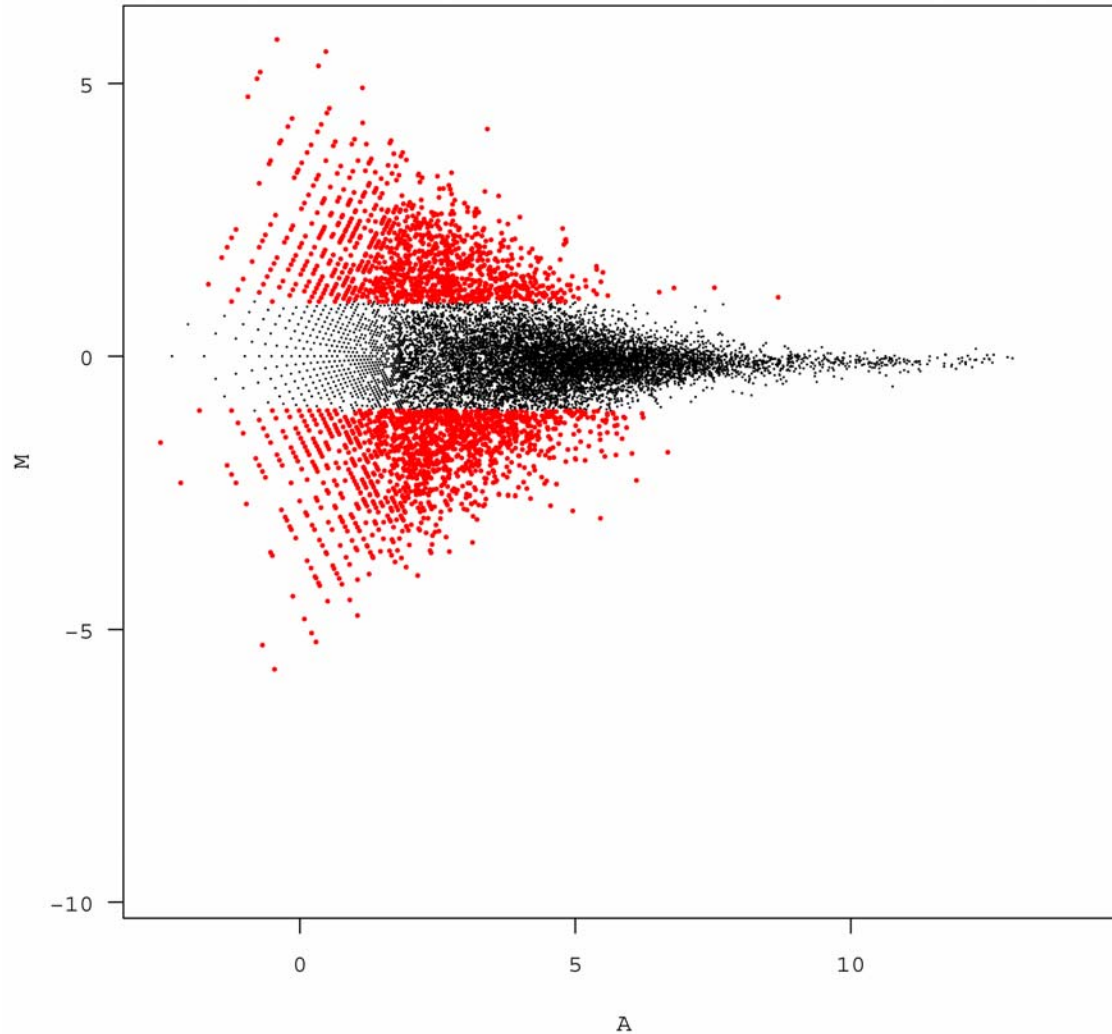
# Current default (MAS 5.0)

- GeneChip<sup>®</sup> new version uses something else

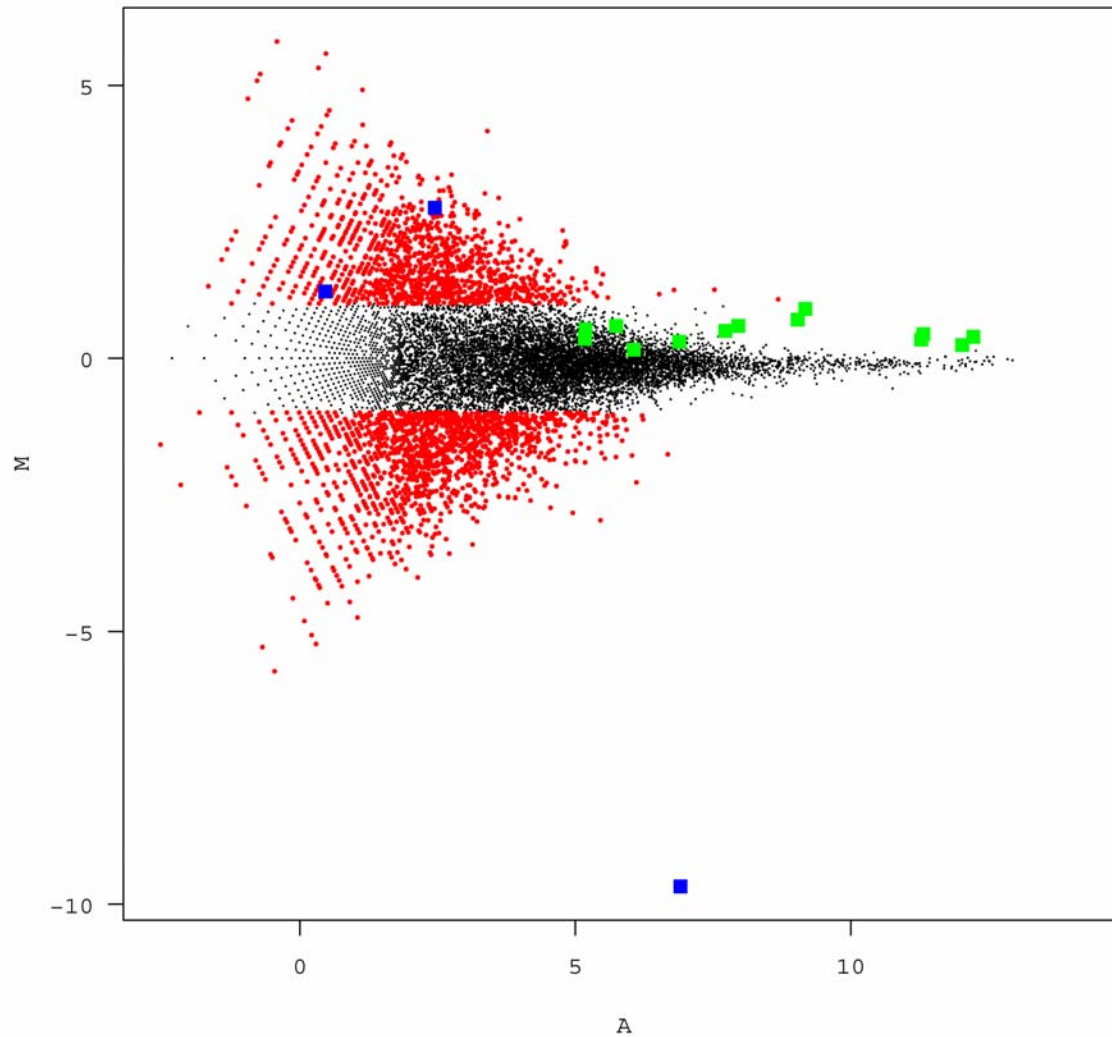
$$signal = TukeyBiweight\{\log(PM_j - MM_j^*)\}$$

- with  $MM^*$  a version of MM that is never bigger than PM.
- Ad-hoc background procedure and scale normalization are used.

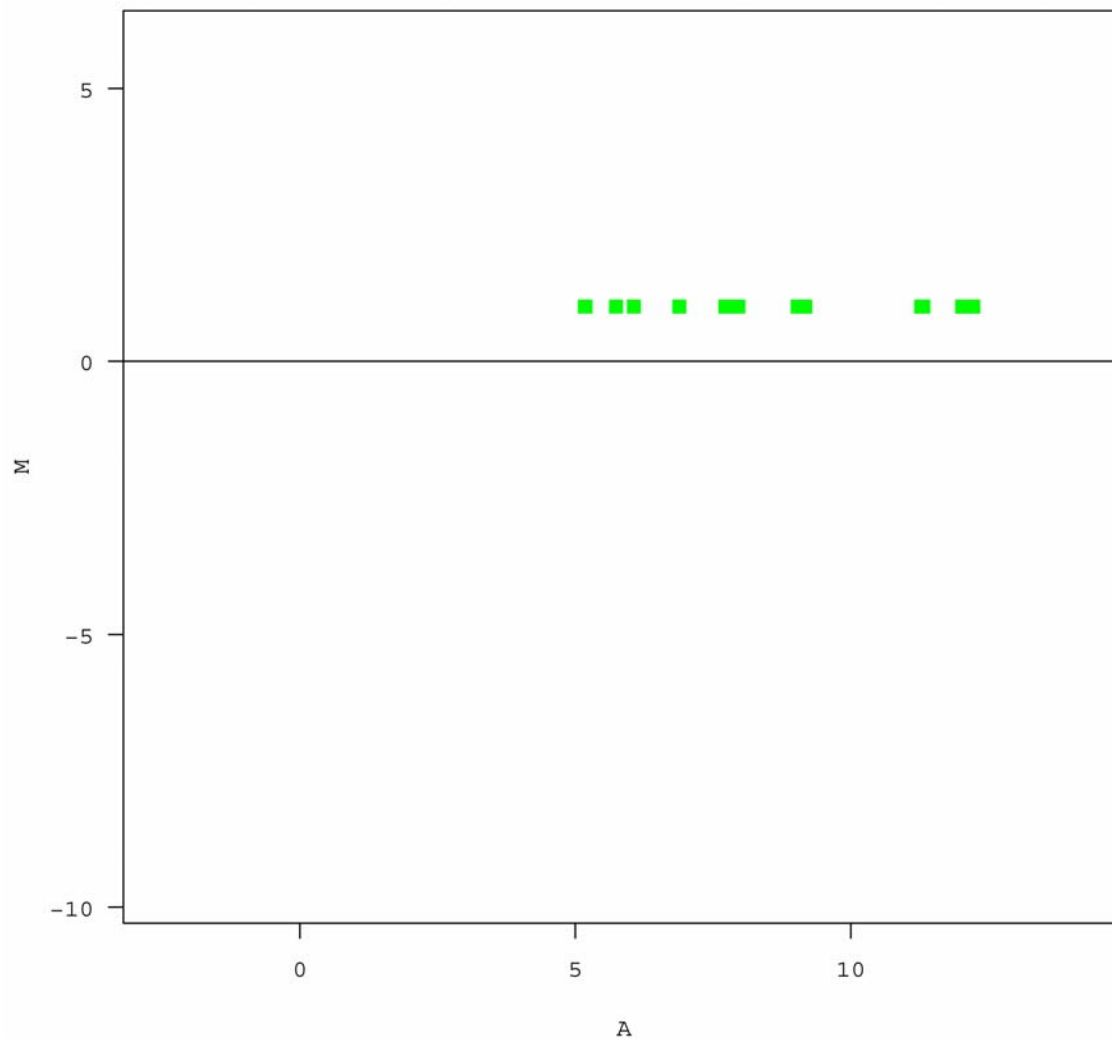
# Can this be improved?



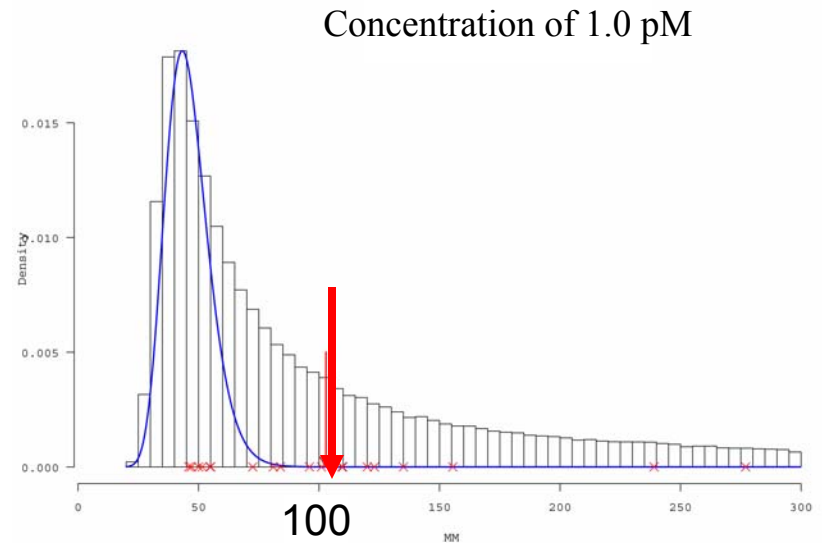
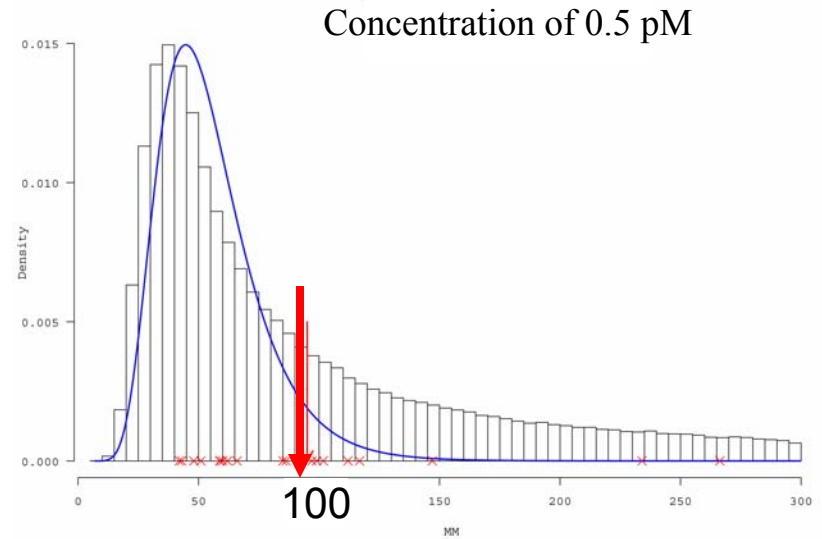
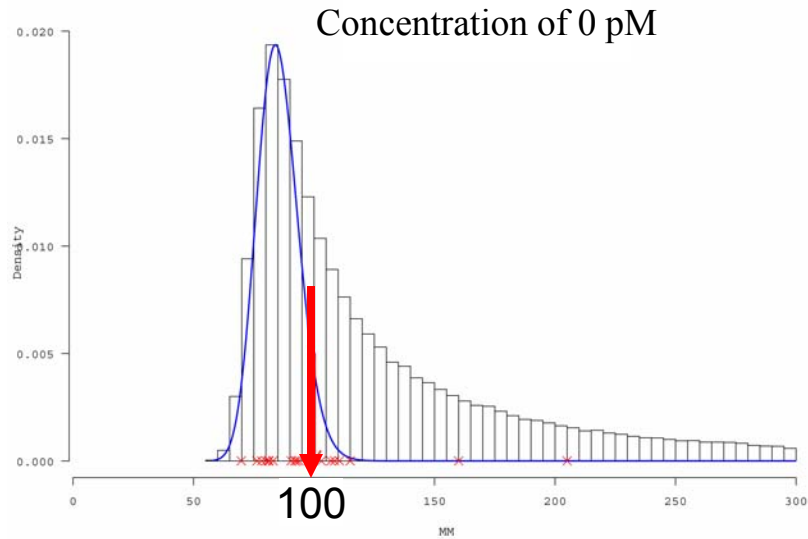
# Use Spike-In Experiment



# Use Spike-In Experiment

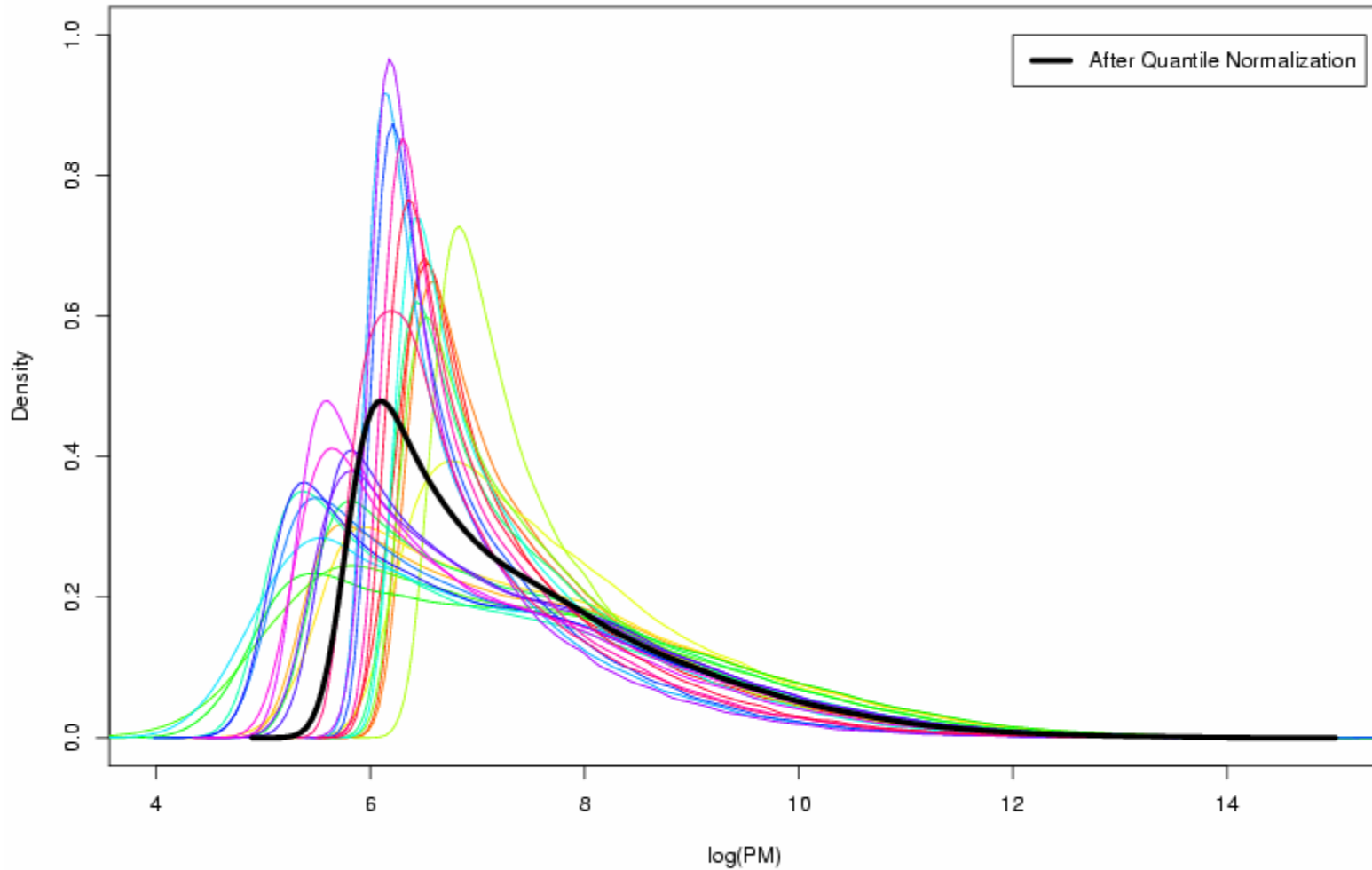


# Why background correct?

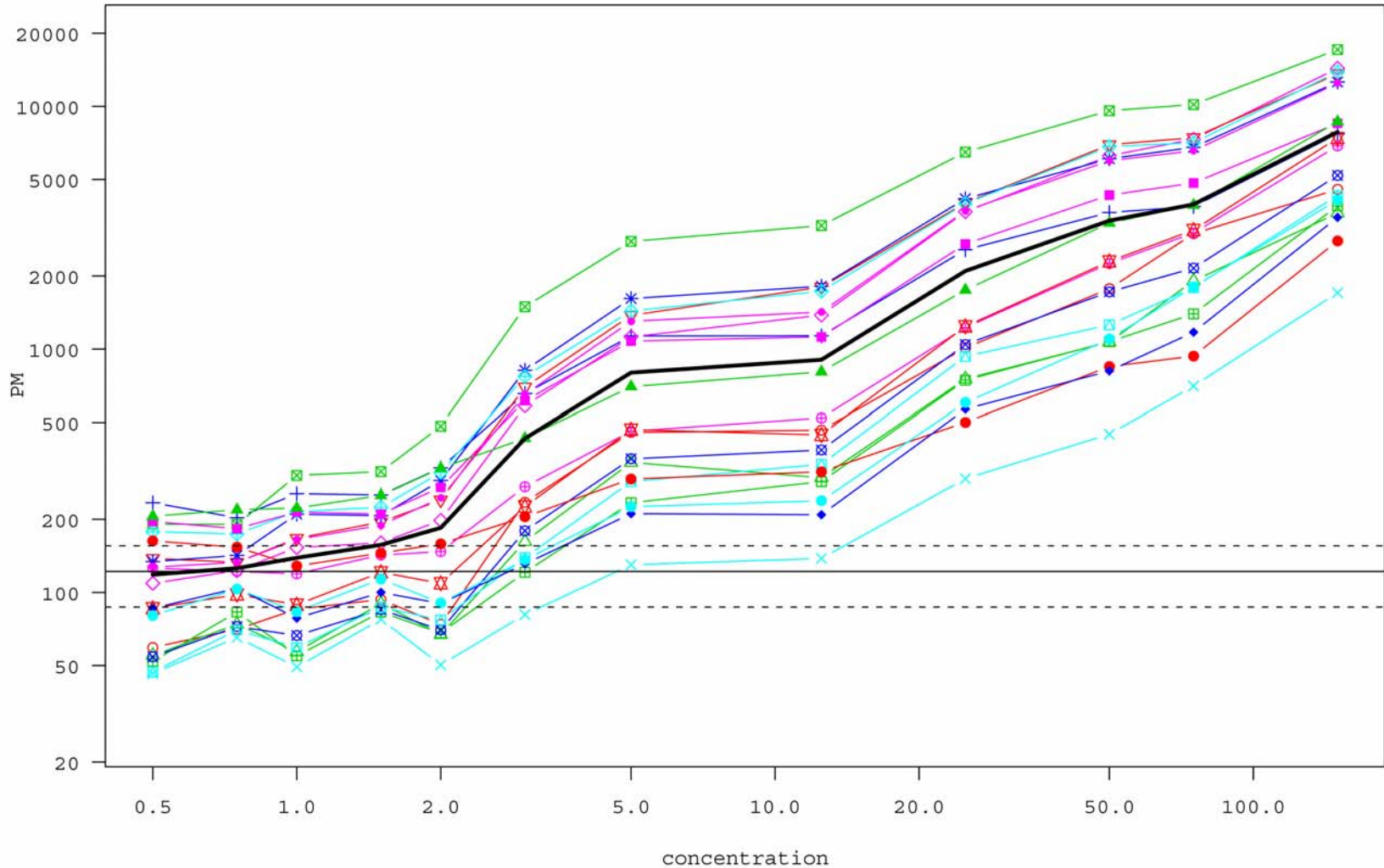


# Why normalize?

Density of PM probe intensities for Spike-In chips



# Why fit statistical models to obtain summaries?

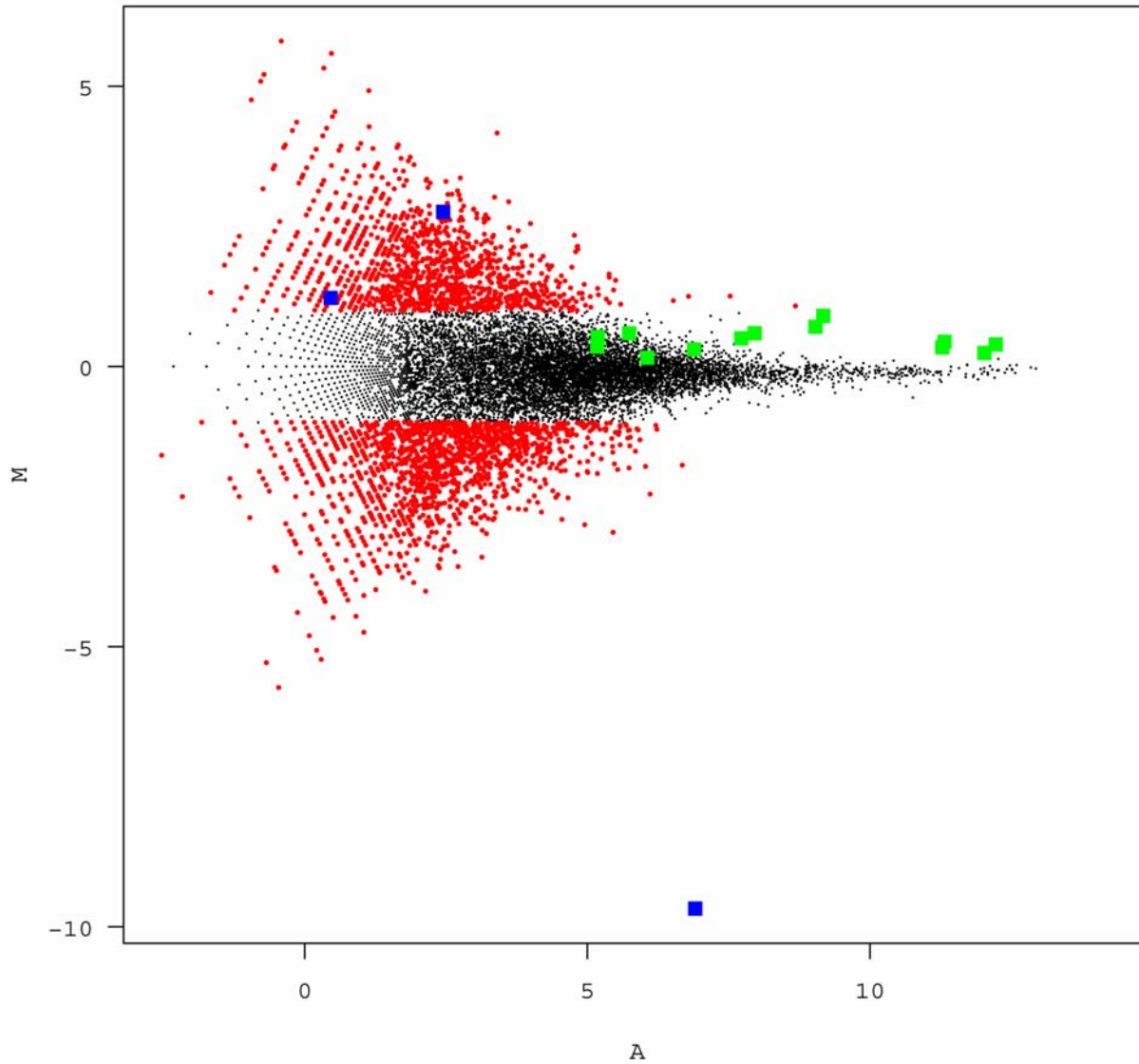


# Example of use of statistical models

- Instead of subtracting MM
- Assume  $PM = B + S$
- To estimate  $S$ , use expectation:  $E[S|B+S]$
- After normalization, assume:
$$\log_2 S_{ij} = E_i + P_j + \varepsilon_{ij}$$
- Estimate  $E_i$  using robust procedure
- We call this procedure **RMA**
- Does it make a difference?



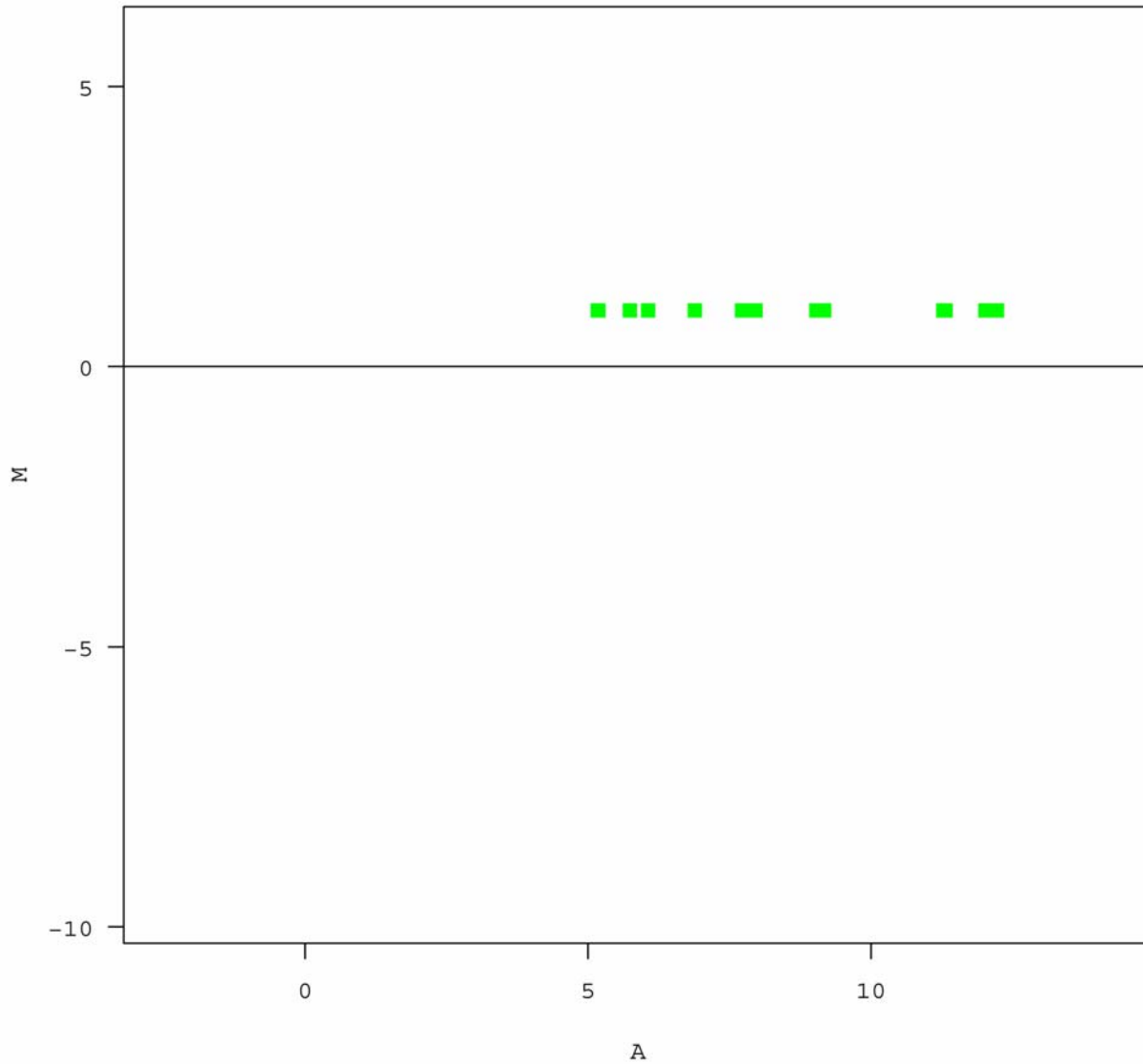
# MAS 5.0



Ranks

- 1
- 270
- 2074
- 3063
- 3935
- 4639
- 4652
- 5149
- 5372
- 5947
- 6448
- 6870
- 7037
- 7549
- 8429
- 9721

# Perfect



Ranks

1

2

3

4

5

6

7

8

9

10

11

12

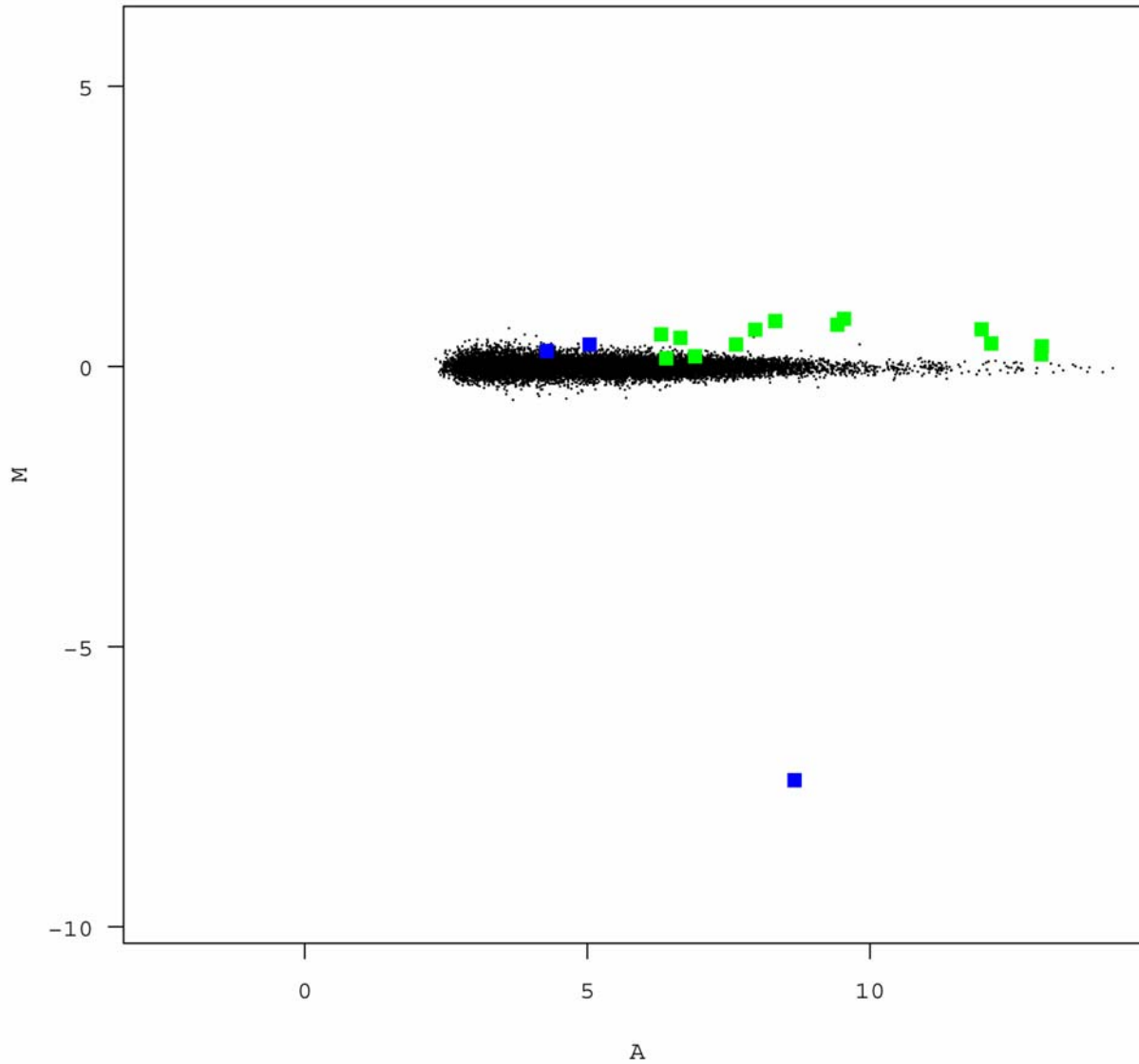
13

14

15

16

# RMA



Ranks

1

2

3

4

6

7

10

16

45

56

58

88

406

999

1643

2739

# Some References

- Li and Wong: PNAS (2001)
- Irizarry et al: Biostatistics (2003)
- Irizarry et al: NAR (2003)
- Bolstad et al: Bioinformatics (2003)

# Differential gene expression

# Data Reduction in Microarray Experiments

Images



Intensities (normalization)



Expression measures (normalization)



**Score**



**Choose a cut off:** report a list of differentially expressed genes and error rate

## Differential gene expression

- Identify genes whose expression levels are **associated** with a response or covariate of interest
  - clinical outcome such as survival, response to treatment, tumor class;
  - covariate such as treatment, dose, time.
- **Estimation**: In a statistical framework, assigning a **score** can be viewed as estimating an effects of interest (e.g. difference in means, slope, interaction). We can also take the **variability** of these estimates into account.
- **Testing**: In a statistical framework, **deciding on a cut-off** can be viewed as an assessment of the statistical **significance** of the observed associations.

## Example: Two populations

A common problem is to find genes that are differentially expressed in two populations.

Many method papers appear in both statistical and molecular biology literature.

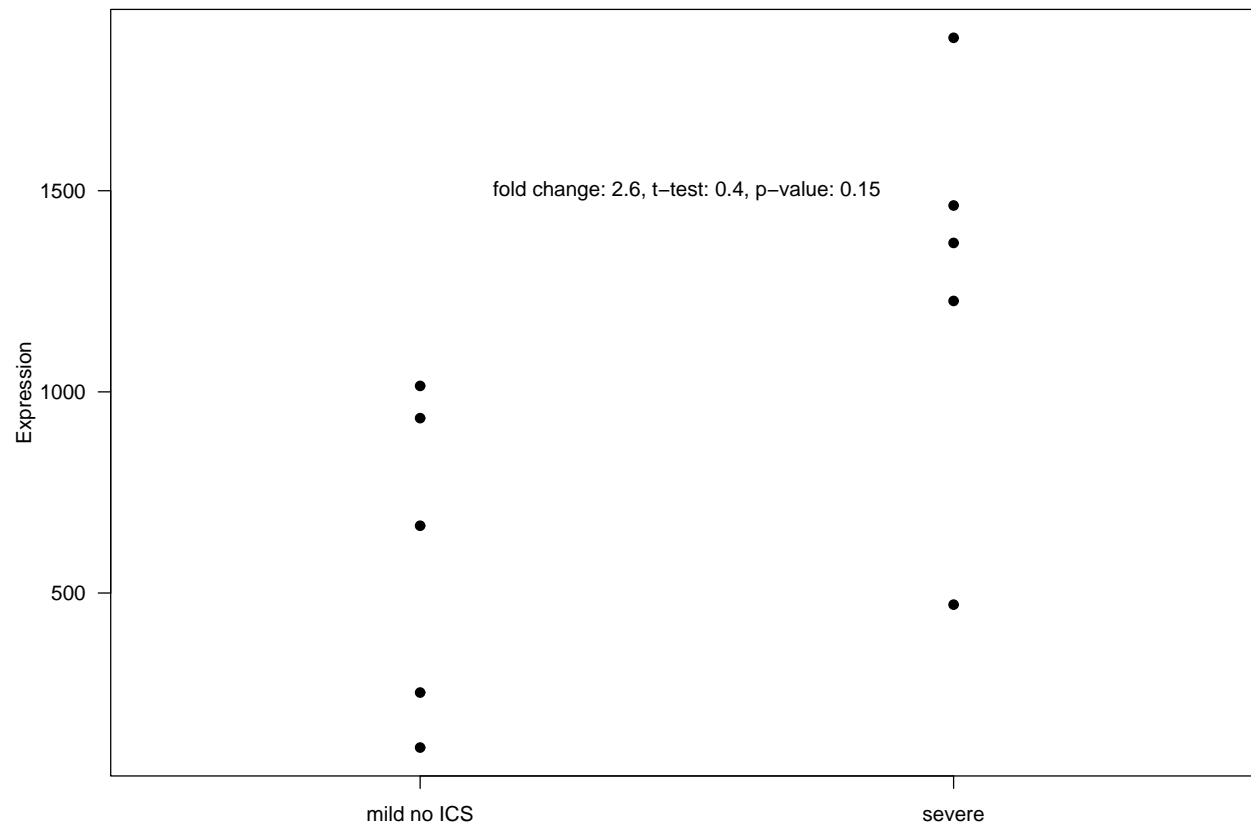
The proposed scores range from:

- ad-hoc summaries of fold-change,
- variantes on the t-test,
- and posterior means obtained from Bayesian or empirical Bayes methods.

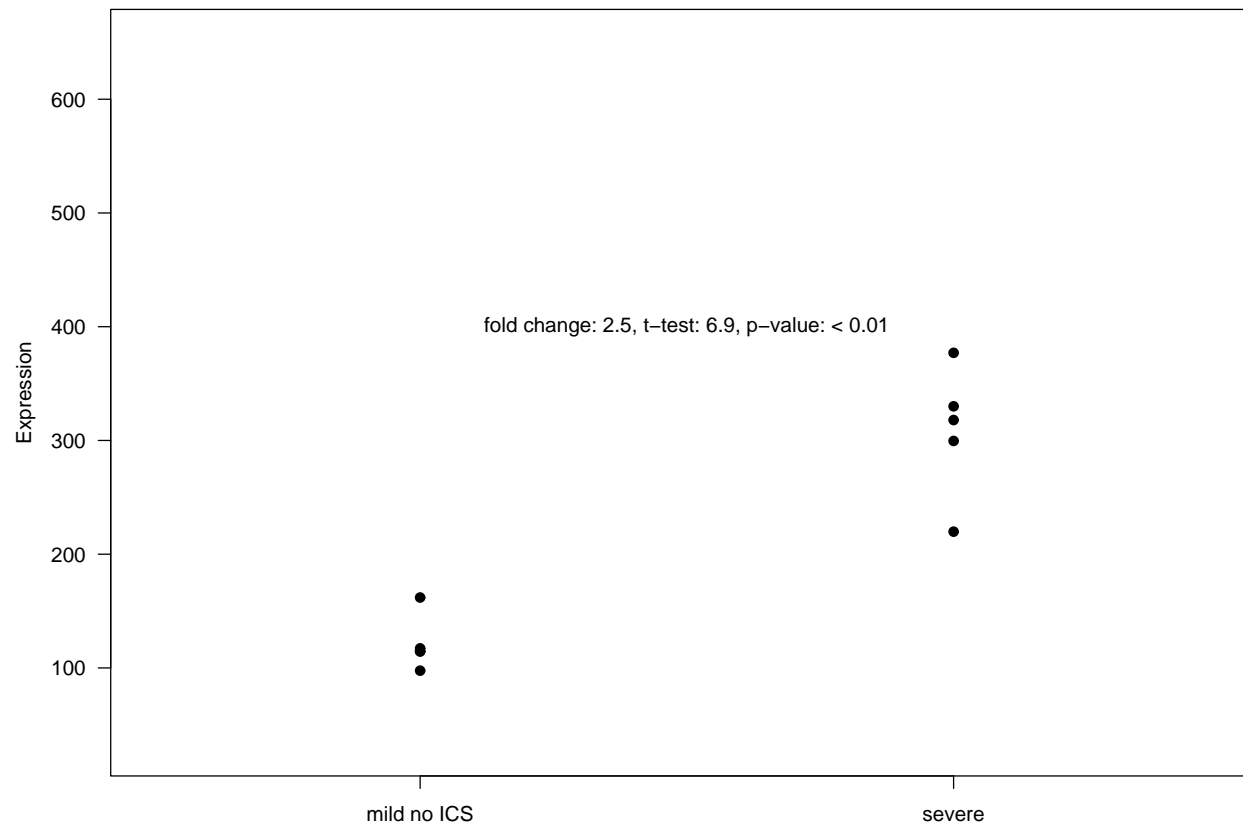
**What's the difference?** Mainly the way in which the variation within population is incorporated



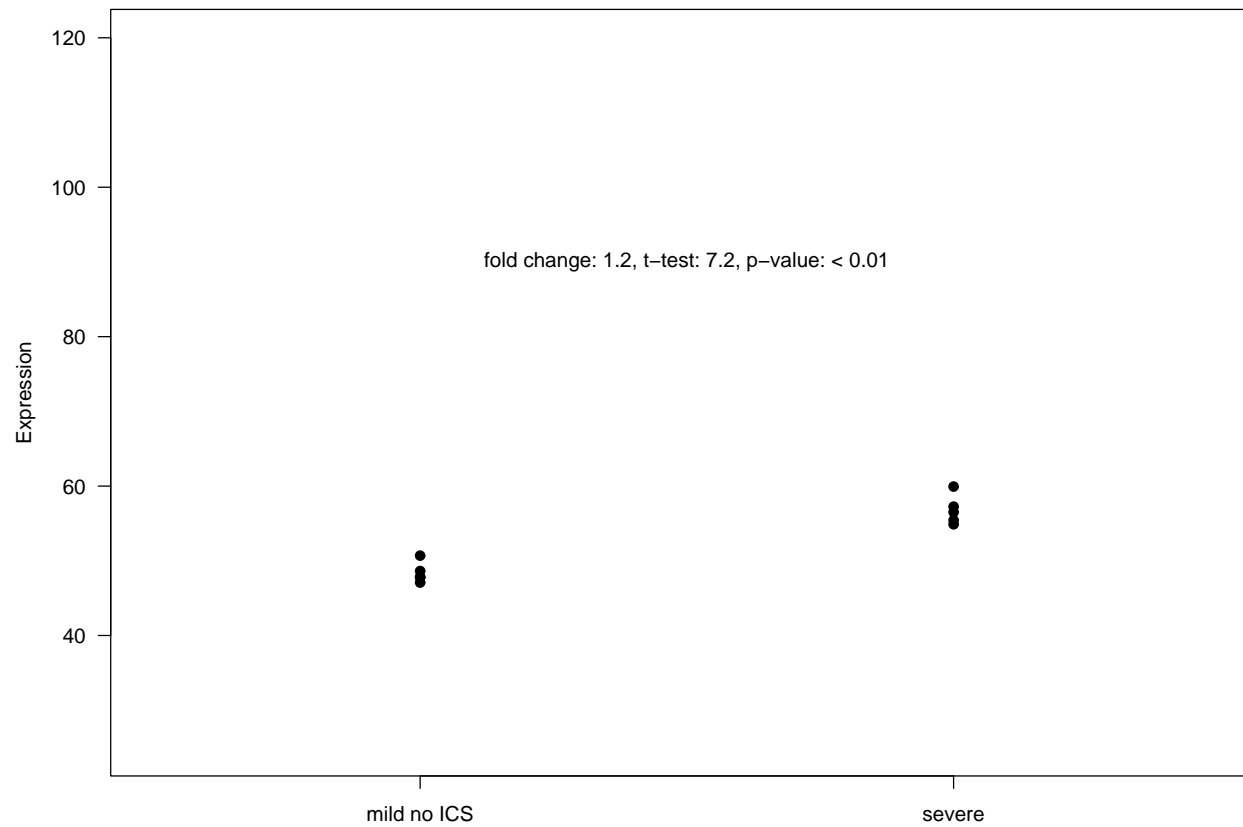
# Should we consider variability of estimate?



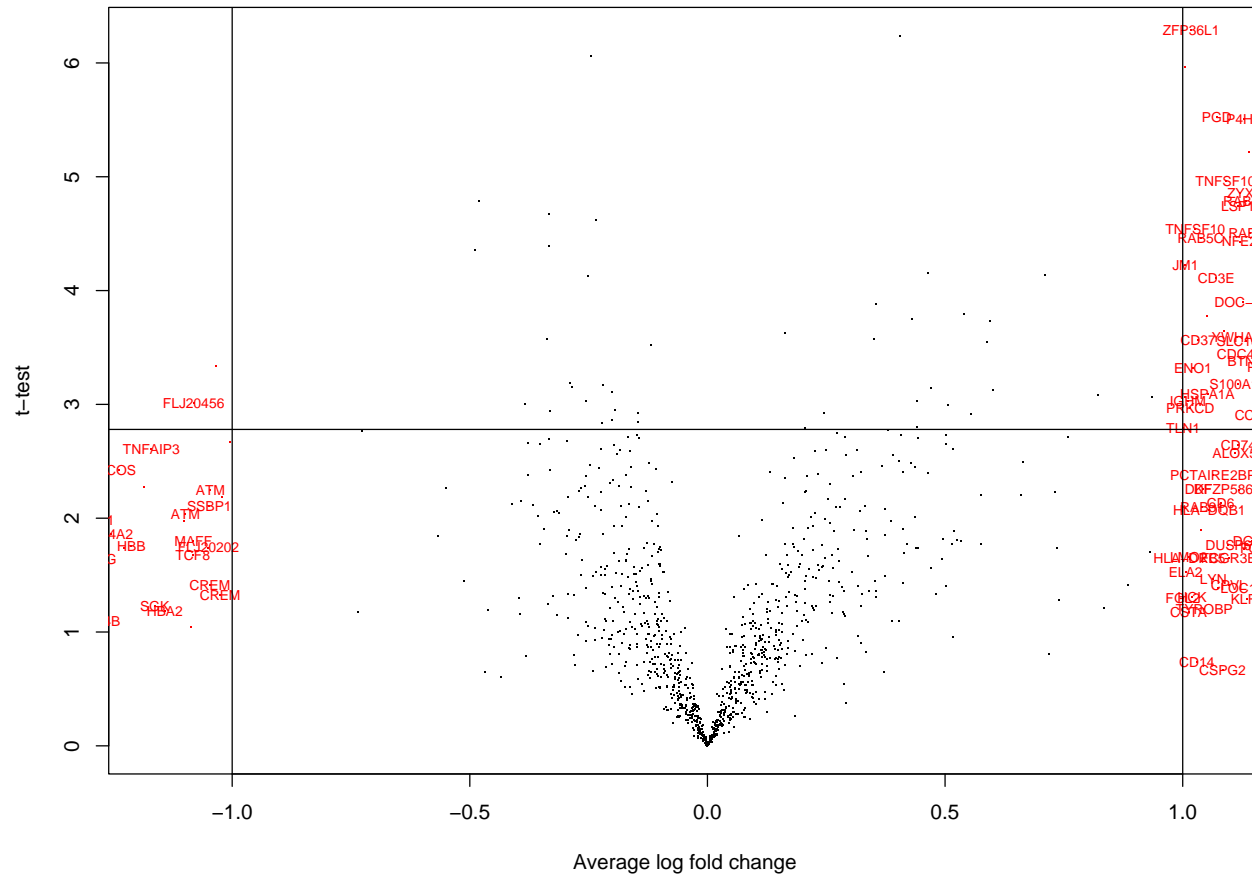
## Should we consider variability of estimate?



## Should we consider variability of estimate?



# Volcano Plots



## Some Examples

**Notation:** log expression, population  $i$ , gene  $j$ , array  $k$ :

$$Y_{jk}(i), j = 1, \dots, J, k = 1, \dots, K = K_1 + K_2, i = 1, 2.$$

- log fold-change:  $\bar{Y}_{j(2)} - \bar{Y}_{j(1)}.$

- t-statistic:  $\frac{\bar{Y}_{j(2)} - \bar{Y}_{j(1)}}{s_j}$

- SAM shrunken-t:  $\frac{\bar{Y}_{j(2)} - \bar{Y}_{j(1)}}{s_j + s_0}.$

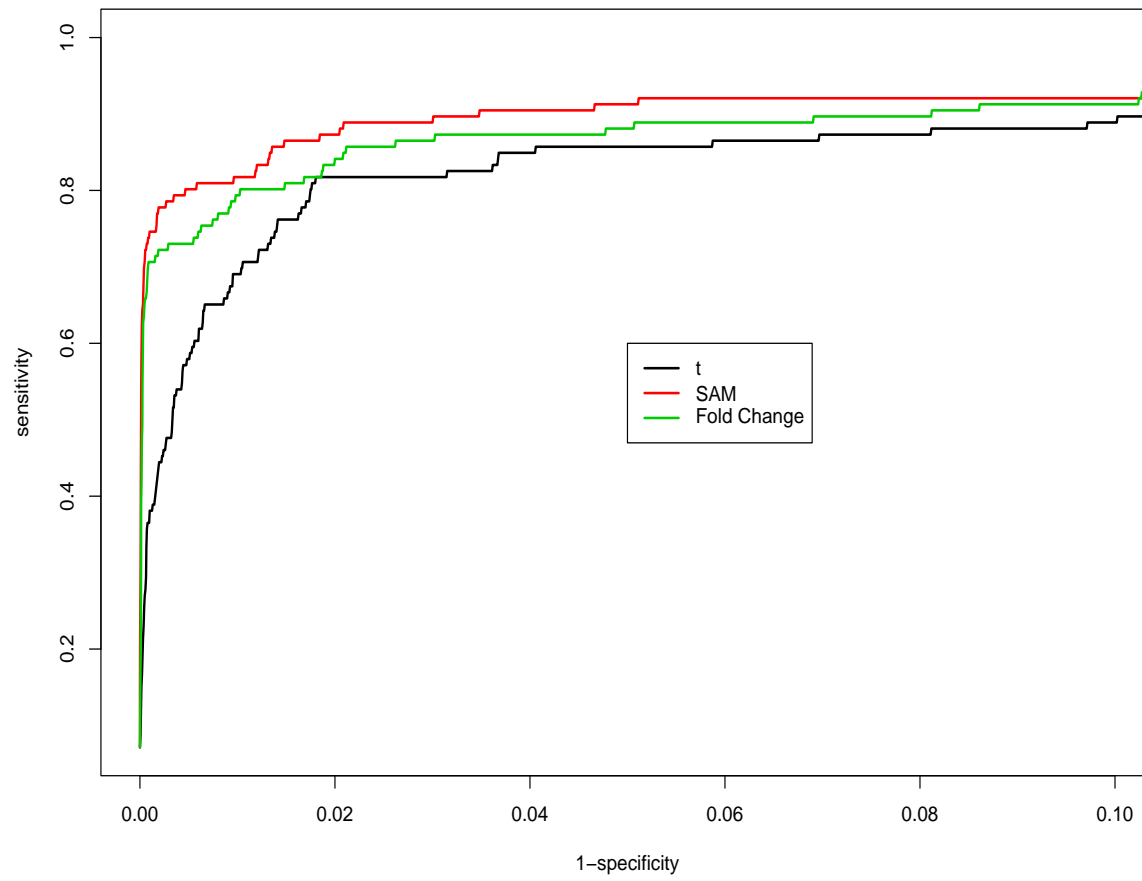
- Wilcoxon rank-sum

- Bayesian (e.g., Baldi and Long):  $\frac{\bar{Y}_{j(2)} - \bar{Y}_{j(1)}}{\sqrt{(1-w)s_j^2 + ws_0^2}}.$

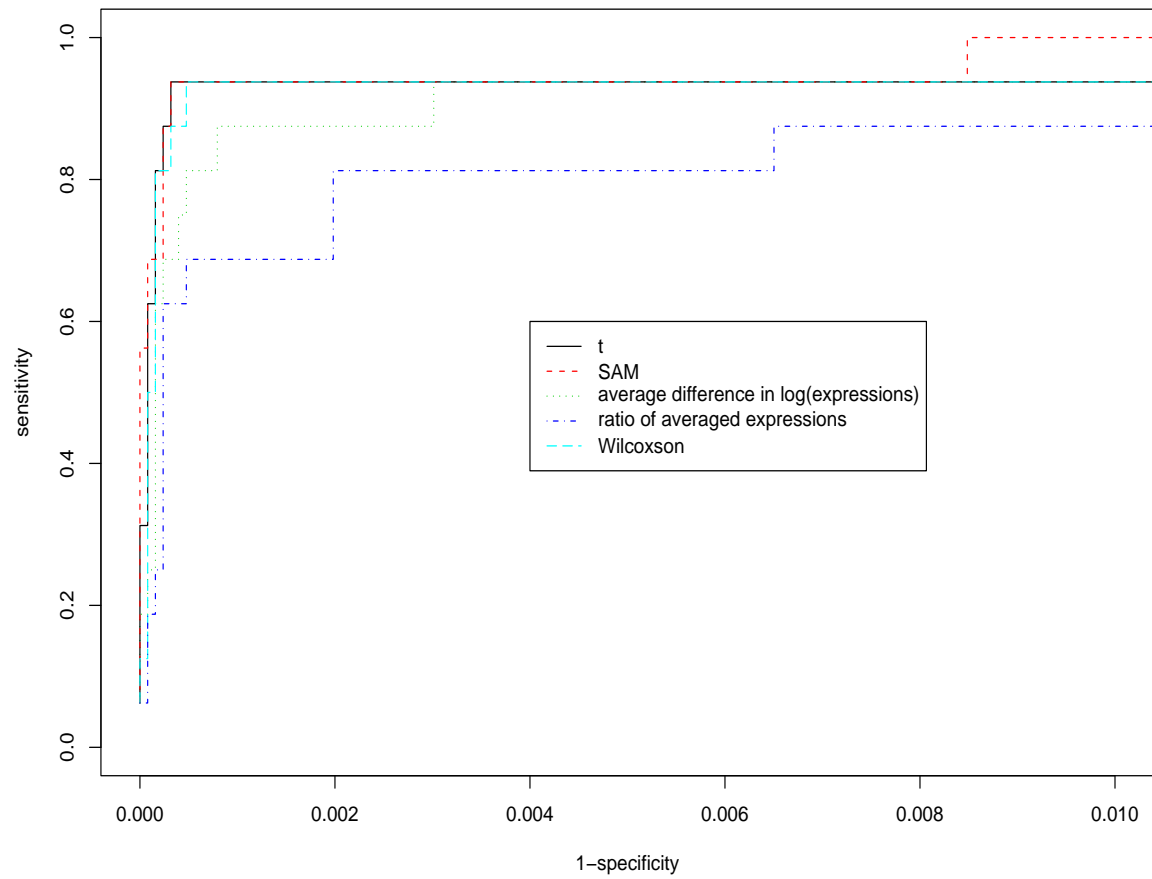
## Does it make a difference?

- Data:  
Spike-in data from Affymetrix, 16 spike-in genes with known spikein concentrations
- Properties of “good method”
  - rank truly differentially expressed genes higher than non-differential ones → sensitivity, specificity
  - ROC curves

**N=3**



**N=12**





## Hypothesis testing

Once you have a score for each gene, how do you decide on a cut-off? p-values are popular. Are they appropriate?

- Test for each gene **null hypothesis**: no differential expression.

$H_g$  :      the expression level of gene  $g$   
                  is not associated with the covariate or response.

Two types of errors can be committed

- **Type I error** or **false positive**  
say that a gene is differentially expressed when it is not, i.e.,  
reject a *true null* hypothesis.
- **Type II error** or **false negative**  
fail to identify a truly differentially expressed gene, i.e.,  
fail to reject a *false null* hypothesis.

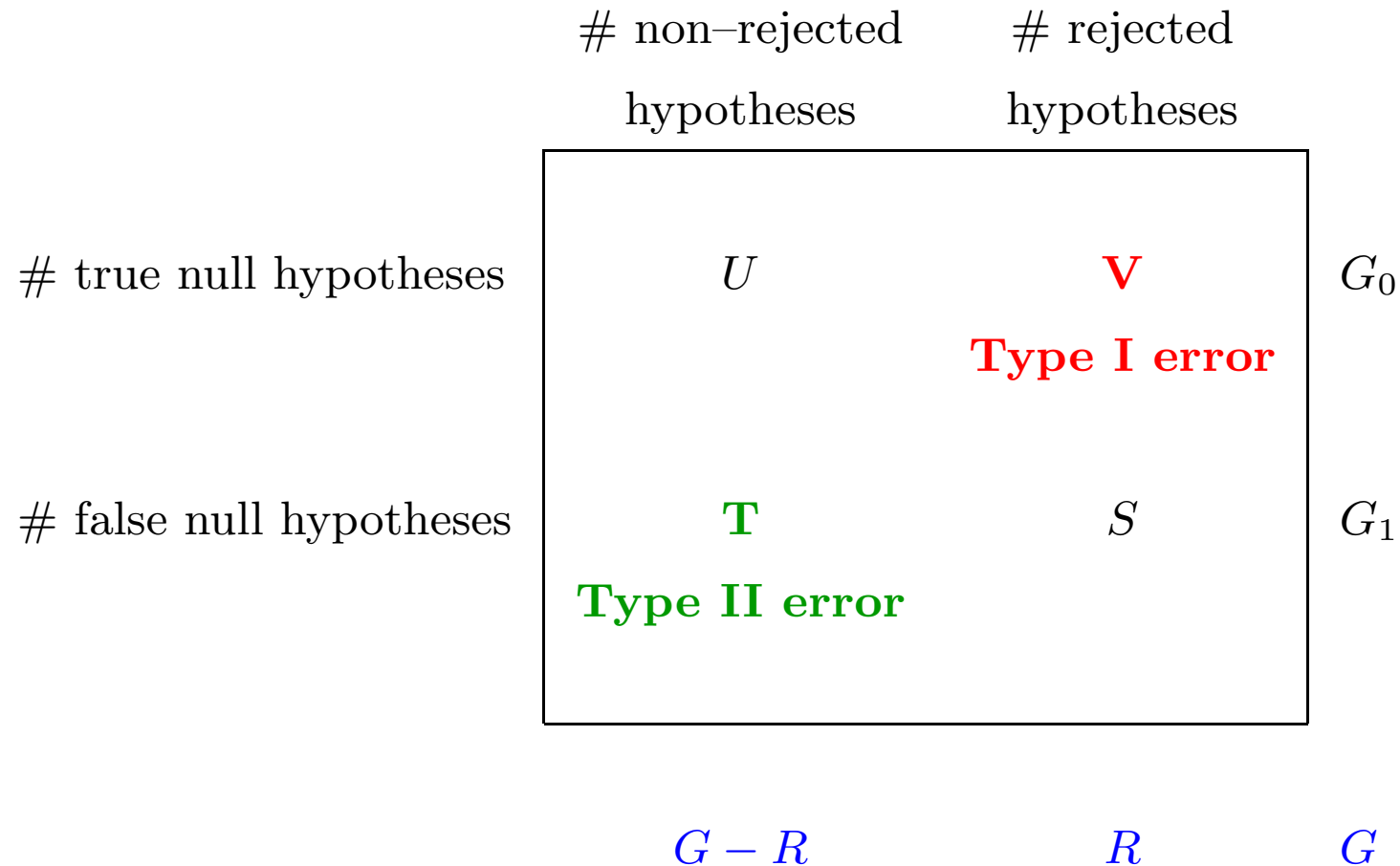
## Multiple hypothesis testing

- Large multiplicity problem: thousands of hypotheses are tested simultaneously!
  - Increased chance of false positives.
  - E.g. chance of at least one  $p$ -value  $< \alpha$  for  $G$  independent tests is  $1 - (1 - \alpha)^G$  and converges to one as  $G$  increases. For  $G = 1,000$  and  $\alpha = 0.01$ , this chance is 0.9999568!
  - Individual  $p$ -values of 0.01 no longer correspond to significant findings.
- Need to **adjust for multiple testing** when assessing the statistical significance of the observed associations.

## Multiple hypothesis testing

- Define an appropriate **Type I error** or **false positive rate**.
- Develop multiple testing procedures that
  - provide **strong control** of this error rate,
  - are **powerful** (few false negatives),
  - take into account the **joint distribution** of the test statistics.
- Report **adjusted  $p$ -values** for each gene which reflect the **overall** Type I error rate for the experiment.
- **Resampling** methods are useful tools to deal with the unknown joint distribution of the test statistics.

# Multiple hypothesis testing



*From Benjamini & Hochberg (1995).*

## Three Examples

### **FWER(Family-Wise Error Rate)**

Probability of including at least one non-differentially expressed genes into your list:  $p(V > 0)$

**False discovery rate (FDR).** The FDR of Benjamini & Hochberg (1995) is the expected proportion of Type I errors among the rejected hypotheses, i.e.,

$$FDR = E(Q),$$

$$Q \equiv \begin{cases} V/R, & \text{if } R > 0, \\ 0, & \text{if } R = 0. \end{cases}$$

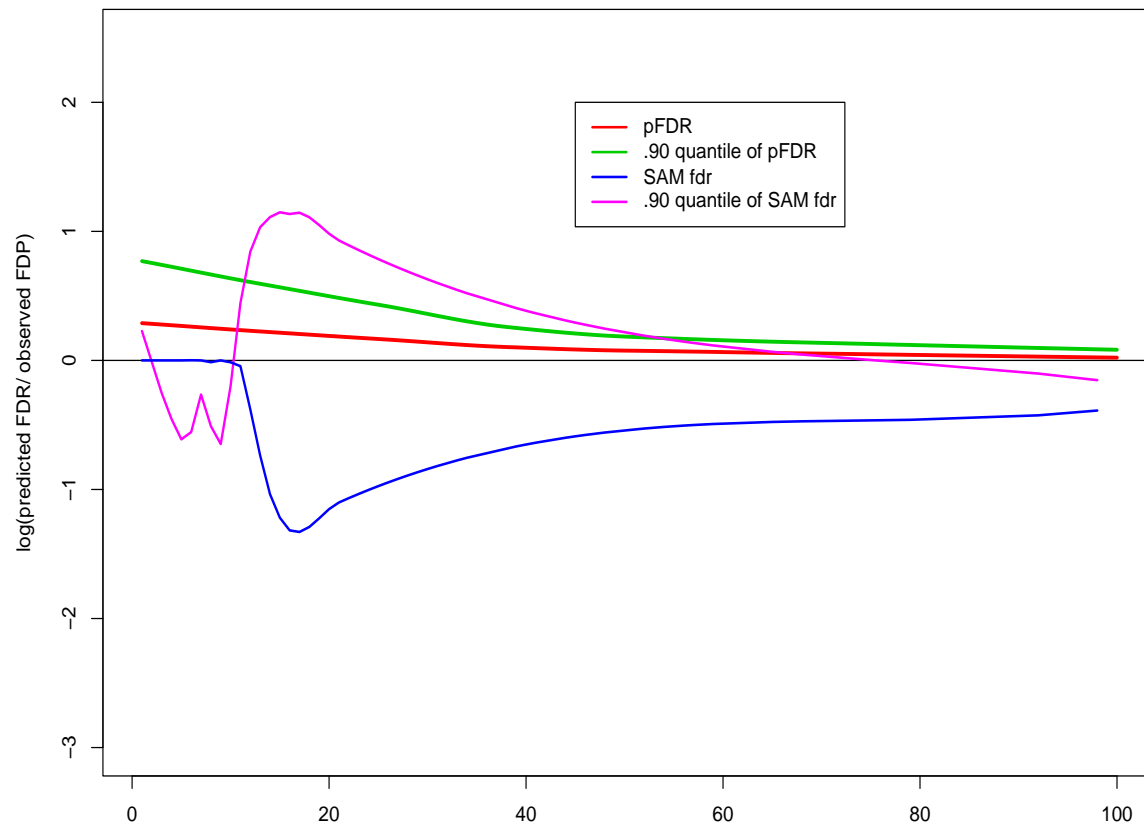
**pFDR.** Expected proportion of false discoveries among the genes in your list conditioning on “at least one gene is included in the differential list”:  $E(Q|R > 0)$

## Does it make a difference?

- Data:  
Spike-in data from Affymetrix, 14 spike-in genes with known concentrations
- Properties of “good method”: reported error rate close to true error rate

$$\log \left( \frac{\text{predicted error rate}}{\text{observed error rate}} \right) \approx 0$$

## Log ratio of predicted and observed error rates



# Demo

- We will demonstrate how to go from a probe level data from two samples hybridized to six Affymetrix arrays to a list of *candidate genes*
- Bioconductor packages used:
  - **affy**: Preprocessing probe level data
  - **Biobase**: organizes expression level data
  - **multtest**: functions for multiple testing



# Affymetrix files

- Main software from Affymetrix company, *MicroArray Suite - MAS*, now version 5.
- **DAT** file: Image file,  $\sim 10^7$  pixels,  $\sim 50$  MB.
- **CEL** file: Cell intensity file, probe level PM and MM values.
- **CDF** file: Chip Description File. Describes which probes go in which probe sets and the location of probe-pair sets (genes, gene fragments, ESTs).

# **affy**: Pre-processing Affymetrix data

- Class definitions for probe-level data: **AffyBatch**, **ProbSet**, **Cdf**, **CEL**.
- Basic methods for manipulating microarray objects: printing, plotting, subsetting.
- Functions and widgets for data input from **CEL** and **CDF** files, and automatic generation of microarray data objects.
- Diagnostic plots: 2D spatial images, density plots, boxplots, MA-plots, etc.

# **affy** classes: **AffyBatch**

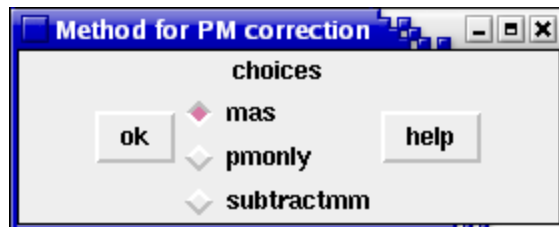
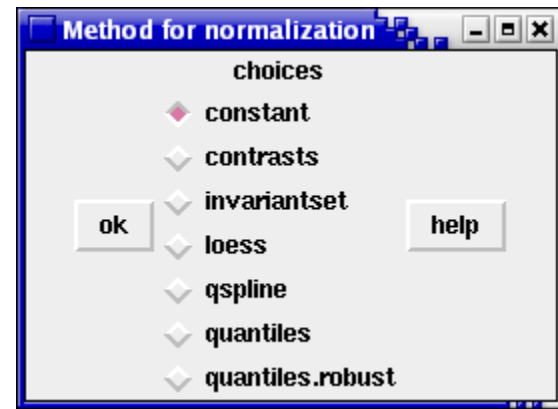
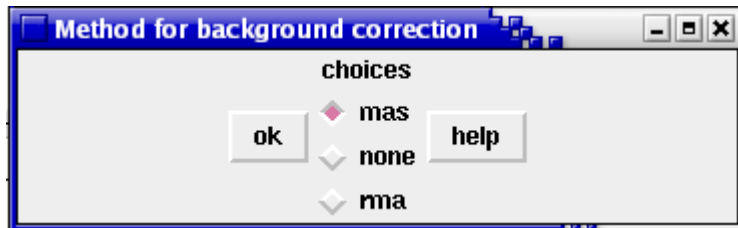
**Probe-level intensity data for a batch of arrays (same CDF)**

<b>cdfName</b>	Name of CDF file for arrays in the batch	
<b>nrow</b>	<b>ncol</b>	Dimensions of the array
<b>exprs</b>	<b>se.exprs</b>	Matrices of probe-level intensities and SEs rows → probe cells, columns → arrays.
<b>phenoData</b>	Sample level covariates, instance of class <b>phenoData</b>	
<b>annotation</b>	Name of annotation data	
<b>description</b>	MIAME information	
<b>notes</b>	Any notes	

# CDF data packages

- Data packages containing necessary CDF information are available at [www.bioconductor.org](http://www.bioconductor.org).
- Packages contain **environment** objects, which provide mappings between AffyIDs and matrices of probe locations,  
rows → probe-pairs, columns → PM, MM (e.g., 20X2 matrix for hu6800).
- **cdfName** slot of **AffyBatch**.
- **HGU95Av2** and **HGU133A** provided in **affy** package.

# Expression measures: expresso



**expresso (widget=TRUE)**

# Acknowledgements

- **Bioconductor core team**
  - **Ben Bolstad**, Biostatistics, UC Berkeley
  - **Vincent Carey**, Biostatistics, Harvard
  - **Francois Collin**, GeneLogic
  - **Leslie Cope**, JHU
  - **Laurent Gautier**, Technical University of Denmark, Denmark
  - **Yongchao Ge**, Statistics, UC Berkeley
  - **Robert Gentleman**, Biostatistics, Harvard
  - **Jeff Gentry**, Dana-Farber Cancer Institute
  - **John Ngai Lab**, MCB, UC Berkeley
  - **Juliet Shaffer**, Statistics, UC Berkeley
  - **Terry Speed**, Statistics, UC Berkeley
  - **Zhijin Wu**, Biostatistics, JHU
  - **Yee Hwa (Jean) Yang**, Biostatistics, UCSF
  - **Jianhua (John) Zhang**, Dana-Farber Cancer Institute
  - Spike-in and dilution datasets:
    - **Gene Brown's group**, Wyeth/Genetics Institute
    - **Uwe Scherf's group**, Genomics Research & Development, GeneLogic.
  - **GeneLogic** and **Affymetrix** for permission to use their data.

# Supplemental Slides

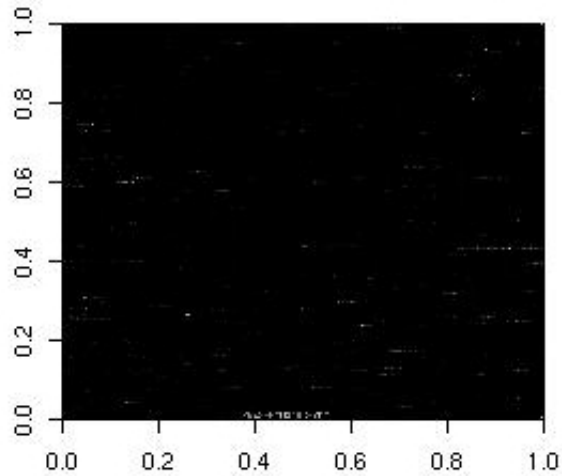
# Diagnostic plots

- See demo (`affy`).
- **Diagnostic plots** of probe-level intensities, PM and MM.
  - **image**: 2D spatial color images of log intensities (`AffyBatch`, `Cel`).
  - **boxplot**: boxplots of log intensities (`AffyBatch`).
  - **mva.pairs**: scatter-plots with fitted curves (apply `exprs`, `pm`, or `mm` to `AffyBatch` object).
  - **hist**: density plots of log intensities (`AffyBatch`).

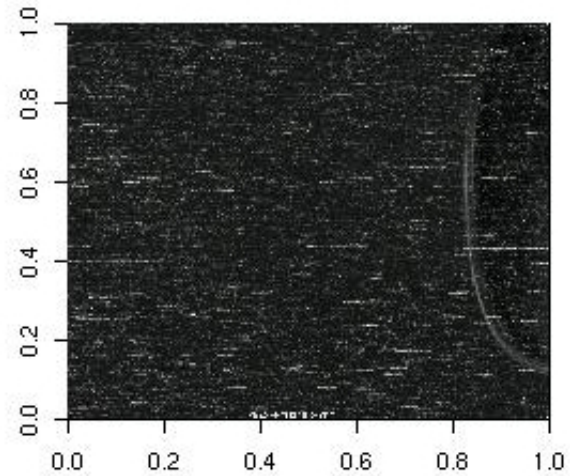


# image

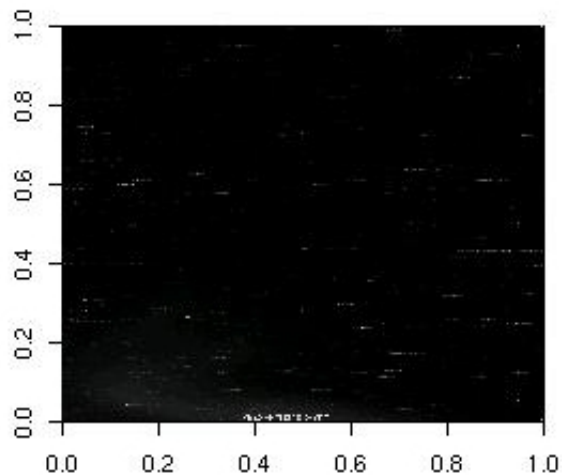
read from file: HIVControl4A.CEL.gz



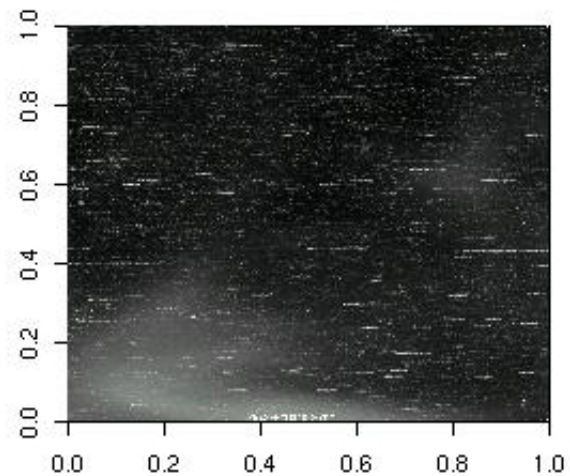
read from file: HIVControl4A.CEL.gz



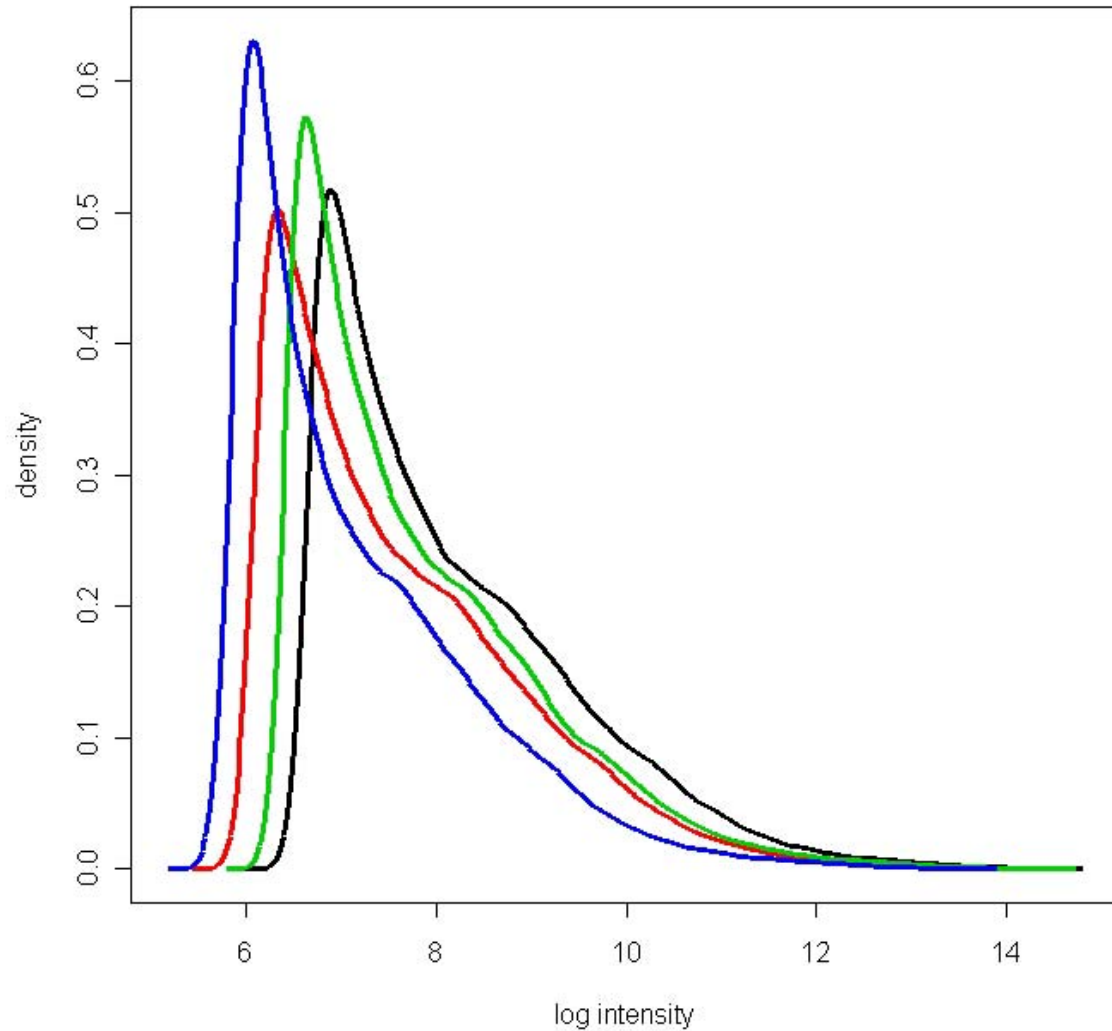
read from file: HIVControl4B.CEL.gz



read from file: HIVControl4B.CEL.gz



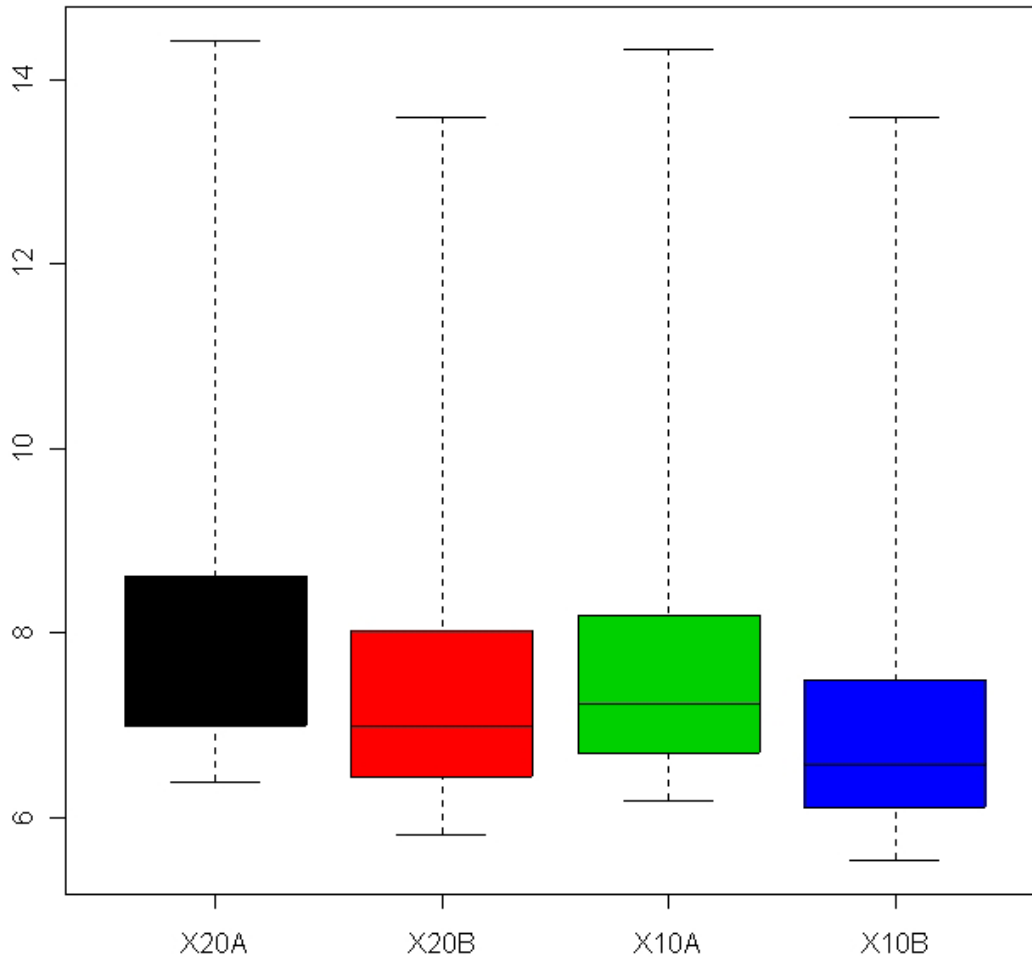
# hist



```
hist(Dilution,col=1:4,type="l",lty=1,lwd=3)
```

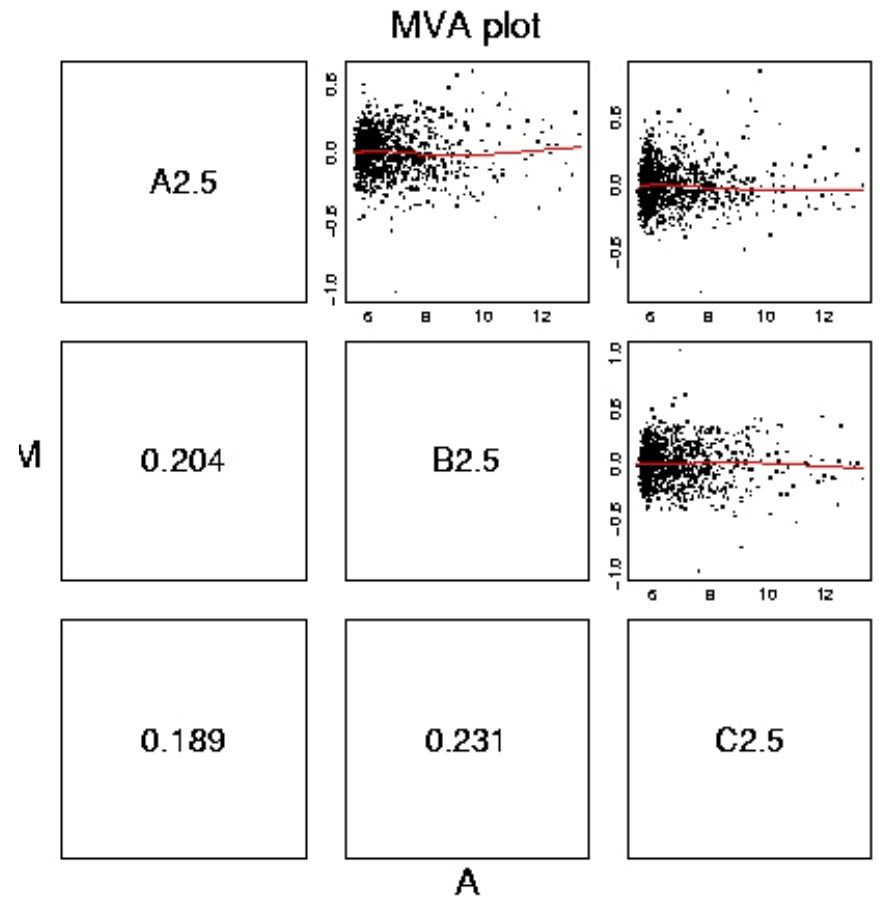
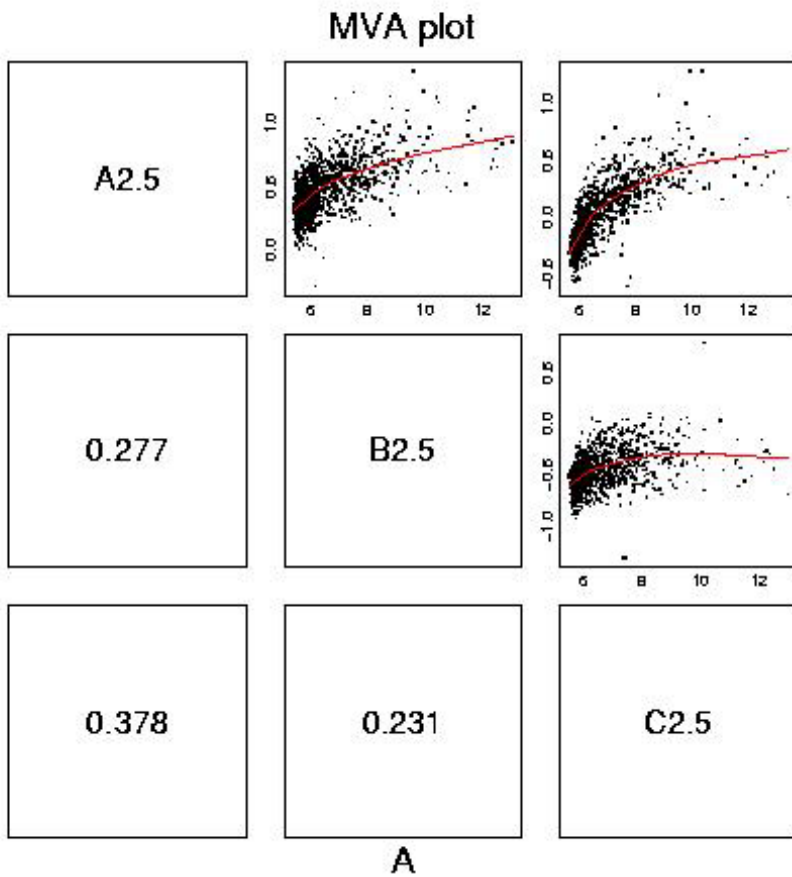
# boxplot

Small part of dilution study



```
boxplot(Dilution, col=1:4)
```

# mva.pairs



# Expression measures

- **expresso**: Choice of common methods for
  - background correction: `bgcorrect.methods`
  - normalization: `normalize.AffyBatch.methods`
  - probe specific corrections: `pmcorrect.methods`
  - expression measures: `express.summary.stat.methods`.
- **rma**: Fast implementation of RMA (Irizarry et al., 2003): model-based background correction, quantile normalization, median polish expression measures.
- **express**: Implementing your own method for computing expression measures.
- **normalize**: Normalization procedures in `normalize.AffyBatch.methods` or `normalize.methods(object)`.

# Probe sequence analysis

- Examine probe intensity based on location relative to 5' end of RNA sequence of interest.
- Expect probe intensities to be lower at 5' end compared to 3' of mRNA.
- E.g.

```
deg<-AffyRNAdeg (Dilution)
```

```
plotAffyRNAdeg (deg)
```

# multtest package

- Multiple testing procedures for controlling
  - **Family-Wise Error Rate - FWER**: Bonferroni, Holm (1979), Hochberg (1986), Westfall & Young (1993) maxT and minP;
  - **False Discovery Rate - FDR**: Benjamini & Hochberg (1995), Benjamini & Yekutieli (2001).
- Tests based on t- or F-statistics for one- and two-factor designs.
- **Permutation procedures** for estimating adjusted p-values.
- Fast permutation algorithm for minP adjusted p-values.
- Documentation: tutorial on multiple testing.

# marrayLayout class

## Array layout parameters

maNspots

Total number of spots

maNgr

maNgc

Dimensions of grid matrix

maNsr

maNsc

Dimensions of spot matrices

maSub

Current subset of spots

maPlate

Plate IDs for each spot

maControls

Control status labels for each spot

maNotes

Any notes



# marrayRaw class

## Pre-normalization intensity data for a batch of arrays

maRf	maGf	Matrix of red and green foreground intensities
maRb	maGb	Matrix of red and green background intensities
maW		Matrix of spot quality weights
maLayout		Array layout parameters - <b>marrayLayout</b>
maGnames		Description of spotted probe sequences - <b>marrayInfo</b>
maTargets		Description of target samples - <b>marrayInfo</b>
maNotes		Any notes

# marrayNorm class

## Post-normalization intensity data for a batch of arrays

maA	Matrix of average log intensities, A	
maM	Matrix of normalized intensity log ratios, M	
maMloc	maMscale	Matrix of location and scale normalization values
maW	Matrix of spot quality weights	
maLayout	Array layout parameters - <code>marrayLayout</code>	
maGnames	Description of spotted probe sequences - <code>marrayInfo</code>	
maTargets	Description of target samples - <code>marrayInfo</code>	
maNormCall	Function call	
maNotes	Any notes	

# `marrayInput` package

- `marrayInput` provides functions for reading microarray data into R and creating microarray objects of class `marrayLayout`, `marrayInfo`, and `marrayRaw`.
- Input
  - Image quantitation data, i.e., output files from image analysis software.  
E.g. `.gpr` for **GenePix**, `.spot` for **Spot**.
  - Textual description of probe sequences and target samples.  
E.g. `gal` files, `god` lists.

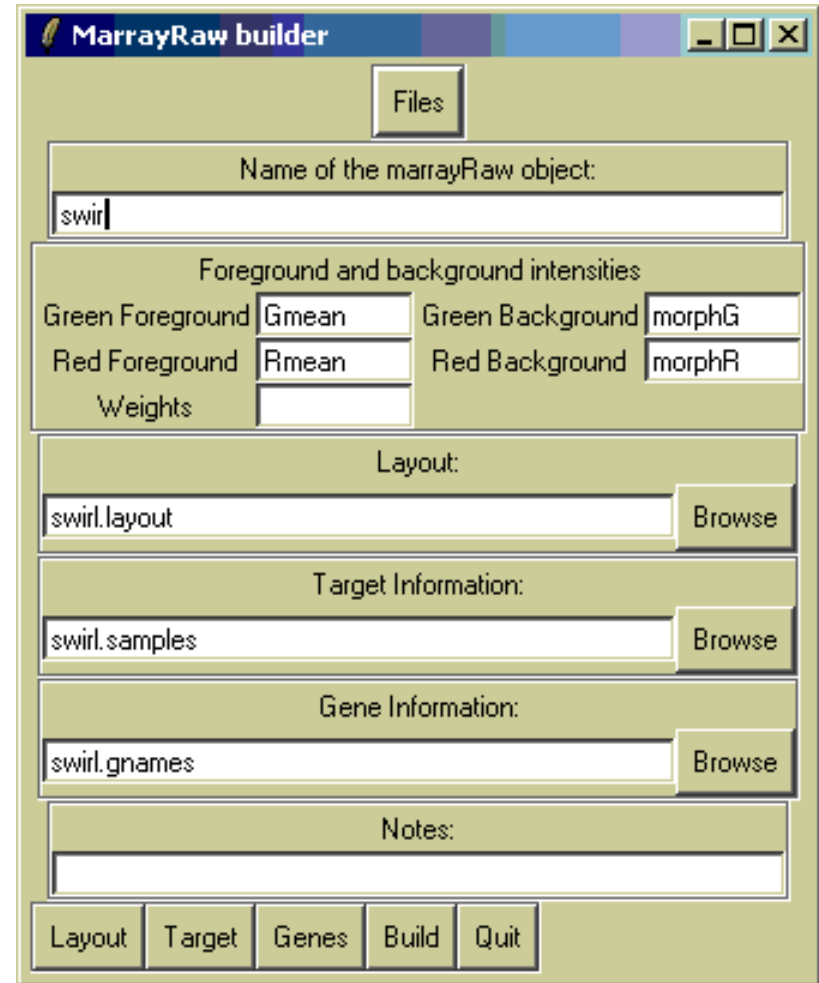
# marrayInput package

- Widgets for graphical user interface

`widget.marrayLayout`,

`widget.marrayInfo`,

`widget.marrayRaw`.



The screenshot shows a window titled "MarrayRaw builder" with a standard Windows-style title bar. The window contains several sections for configuring a marrayRaw object:

- Files:** A button labeled "Files" is located at the top right.
- Name of the marrayRaw object:** A text input field containing the text "swirl".
- Foreground and background intensities:** A section with four input fields: "Green Foreground" (Gmean), "Green Background" (morphG), "Red Foreground" (Rmean), and "Red Background" (morphR). Below these is a "Weights" label and an empty input field.
- Layout:** A text input field containing "swirl.layout" and a "Browse" button to its right.
- Target Information:** A text input field containing "swirl.samples" and a "Browse" button to its right.
- Gene Information:** A text input field containing "swirl.gnames" and a "Browse" button to its right.
- Notes:** A large empty text area for entering notes.
- Buttons:** A row of five buttons at the bottom: "Layout", "Target", "Genes", "Build", and "Quit".

# marrayPlots package

- See demo (`marrayPlots`).
- **Diagnostic plots** of spot statistics.  
E.g. red and green log intensities, intensity log ratios  $M$ , average log intensities  $A$ , spot area.
  - `maImage`: 2D spatial color images.
  - `maBoxplot`: boxplots.
  - `maPlot`: scatter-plots with fitted curves and text highlighted.
- **Stratify** plots according to layout parameters such as `print-tip-group`, `plate`.  
E.g. MA-plots with loess fits by `print-tip-group`.

# marrayNorm package

- **maNormMain**: main normalization function, allows **robust adaptive location and scale normalization** for a batch of arrays
  - intensity or A-dependent location normalization (**maNormLoess**);
  - 2D spatial location normalization (**maNorm2D**);
  - median location normalization (**maNormMed**);
  - scale normalization using MAD (**maNormMAD**);
  - composite normalization;
  - your own normalization function.
- **maNorm**: simple wrapper function.  
**maNormScale**: simple wrapper function for scale normalization.

# **marrayTools** package

- The **marrayTools** package provides additional functions for handling two-color spotted microarray data (see devel. version).
- The **spotTools** and **gpTools** functions start from Spot and GenePix image analysis output files, respectively, and automatically
  - read in these data into R,
  - perform standard normalization (within print-tip-group loess),
  - create a directory with a standard set of diagnostic plots (jpeg format), excel files of quality measures, and tab delimited files of normalized log ratios  $M$  and average log intensities  $A$ .

# swirl dataset

- Microarrays:
  - 8,448 probes (768 controls);
  - 4 x 4 grid matrix;
  - 22 x 24 spot matrices.
- 4 hybridizations: swirl mutant and wild type mRNA.
- Data stored in object of class `marrayRaw`: `data(swirl)`.
- ```
> maInfo(maTargets(swirl))[,3:4]
```

|   | experiment Cy3 | experiment Cy5 |
|---|----------------|----------------|
| 1 | swirl          | wild type      |
| 2 | wild type      | swirl          |
| 3 | swirl          | wild type      |
| 4 | wild type      | swirl          |



# Scale normalization

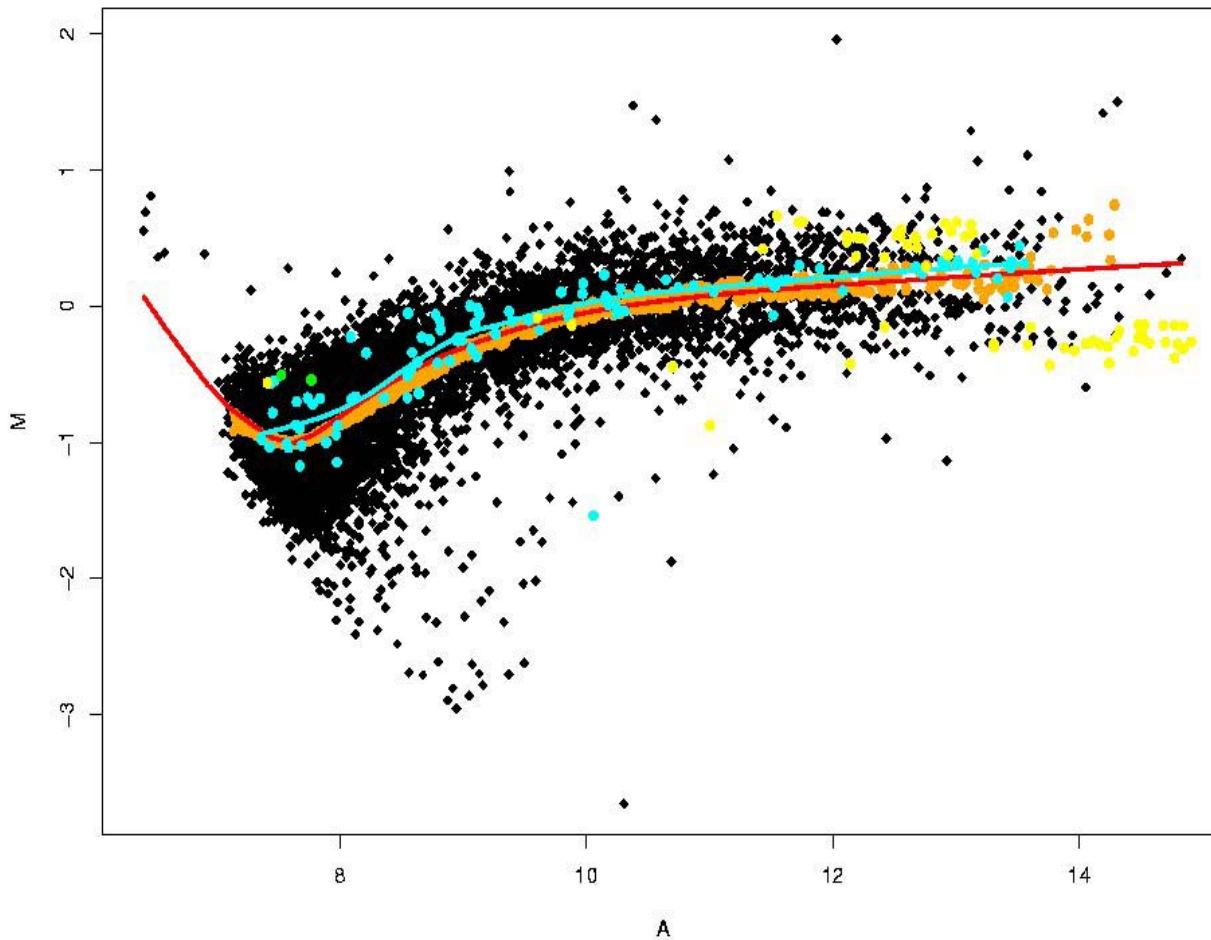
- For print-tip-group scale normalization, assume all print-tip-groups have the same spread in  $M$ .
- Denote **true** and **observed** log-ratio by  $\mu_{ij}$  and  $M_{ij}$ , resp., where  $M_{ij} = a_i \mu_{ij}$ , and  $i$  indexes print-tip-groups and  $j$  spots. Robust estimate of  $a_i$  is

$$\hat{a}_i = \frac{MAD_i}{\sqrt[I]{\prod_{i=1}^I MAD_i}}$$

where  $MAD_i$  is MAD of  $M_{ij}$  in print-tip-group  $i$ .

- Similarly for between-slides scale normalization.

# Microarray sample pool



MSP

Rank invariant

Housekeeping

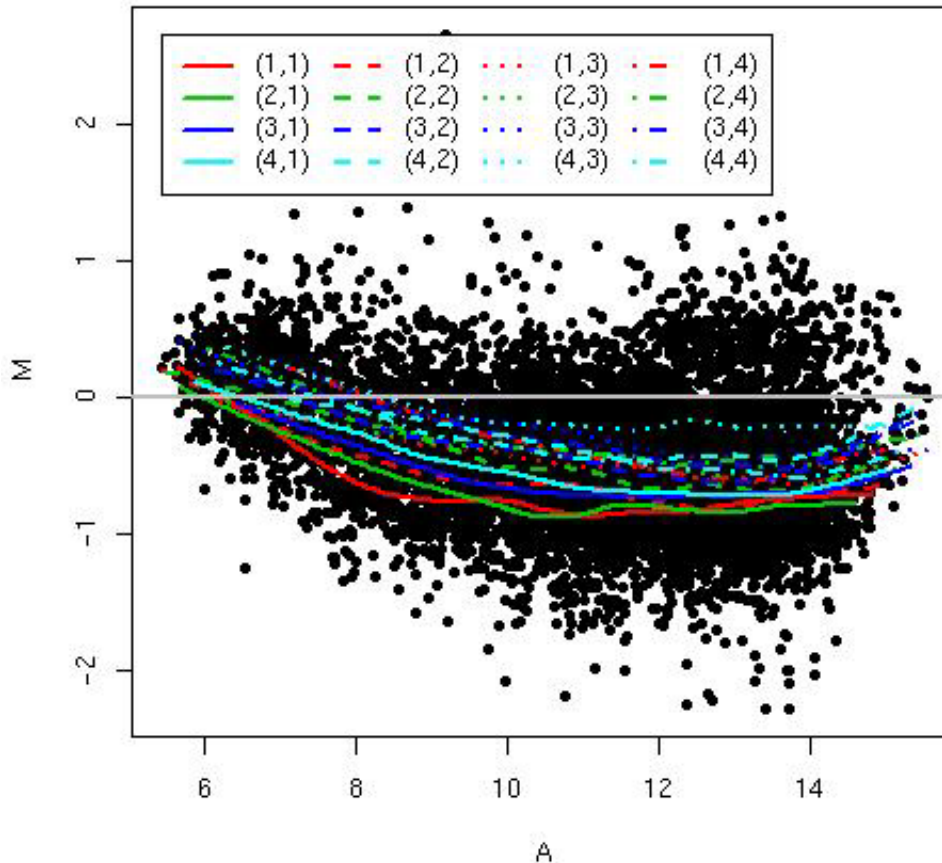
Tubulin, GAPDH

# MA-plot by print-tip-group

## maPlot

$$M = \log_2 R - \log_2 G, \quad A = (\log_2 R + \log_2 G)/2$$

Swirl 93 array: pre-normalization log-ratio M



Intensity  
log ratio, M

Average  
log intensity, A