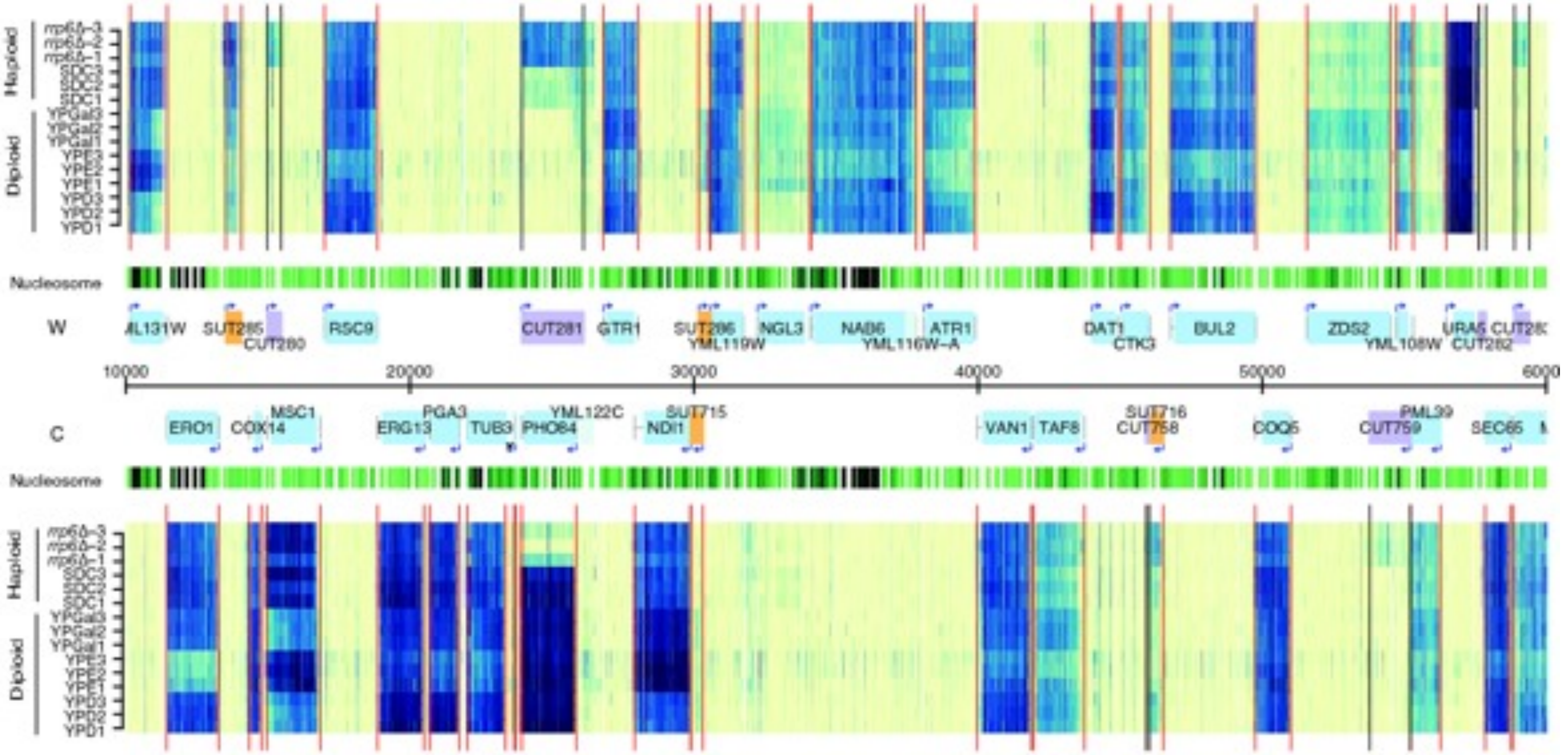
Bioconductor tools for microarray analysis

“Preprocessing”: normalization & error models



Wolfgang Huber
EMBL

BIOCONDUCTOR

An international open source and open development software project for the analysis of genomic data

Use the statistical environment and language R as the integrating middleware

Design principles: rapid development, code re-use

Six-monthly release cycle; release 1.0 in March 2003 (15 packages), ..., release 2.6 on 23.4.2010 (389 packages)

BIOCONDUCTOR

Goals

Provide access to powerful statistical and graphical methods for the analysis of genomic data

Facilitate the integration of biological metadata (e.g. EntrezGene, BioMarts, PubMed) in the analysis of experimental data

Promote the development of accessible, extensible, transparent and well-documented software

Promote reproducible research

Provide training in computational and statistical methods

BIOCONDUCTOR

Best known for microarray data analysis, but has now also expanded into:

Graph data structures and visualisation

Next generation sequencing, genotyping, association studies

Efficient infrastructure for computing with character sequences, intervals

Cell-based assays, flow cytometry, automated microscopy

Good scientific software is like a good scientific publication

Reproducible

Subject to peer-review

Easy to access and use by others

Builds on the work of others

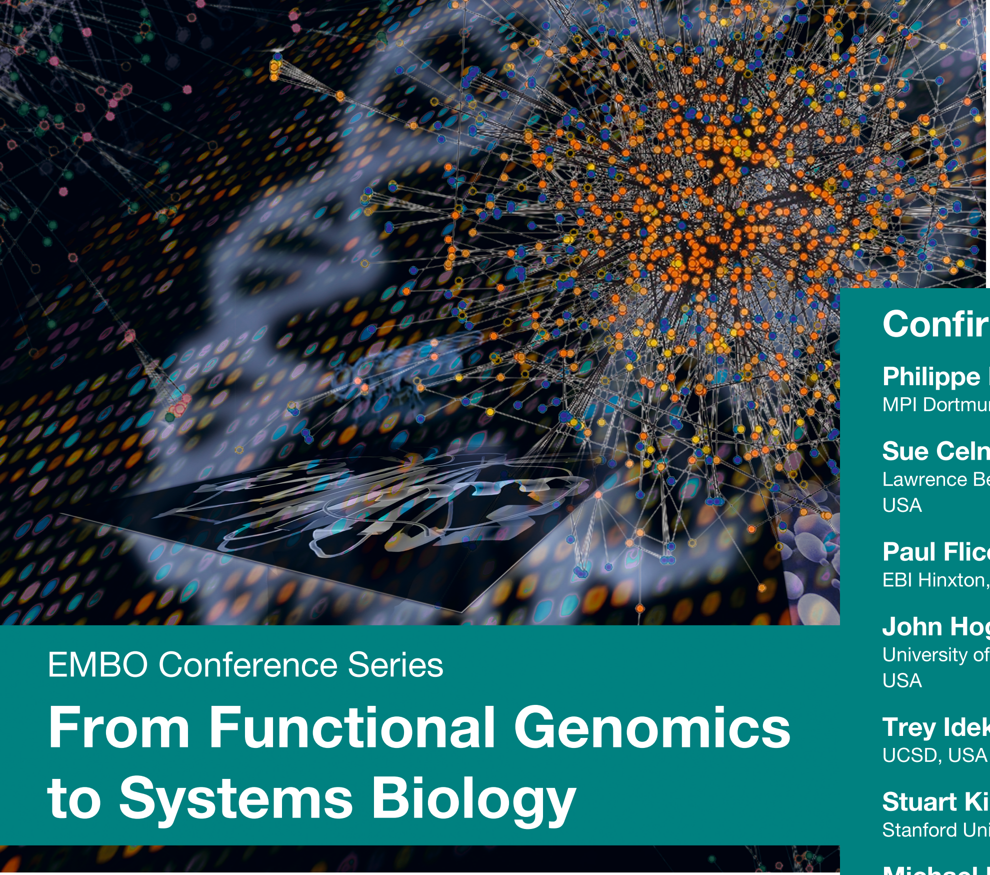
Others can build their work on top of it

European Bioconductor Short Course:
Brixen, South Tyrol, June 2003, ..., 2010

Bioconductor Conference:
Seattle, WA, 28-30 July 2010

Developer Meeting:
Heidelberg, 17-18 Nov 2010

Many further short courses & developer meetings: see www.bioconductor.org!



EMBO Conference Series

From Functional Genomics to Systems Biology

13–16 November 2010

EMBL Heidelberg, Germany
Advanced Training Centre

Confirmed Speakers

Philippe Bastiaens
MPI Dortmund, Germany

Sue Celniker
Lawrence Berkeley Nat. Lab, USA

Paul Flicek
EBI Hinxton, UK

John Hogenesch
University of Pennsylvania, USA

Trey Ideker
UCSD, USA

Stuart Kim
Stanford University, USA

Michael Levine
UC Berkeley, USA

Jason Lieb
UNC Chapel Hill, USA

Denis Noble
University of Oxford, UK

Erin O'Shea
Harvard MCB, USA

Lucas Pelkmans
ETH Zurich, Switzerland

Aviv Regev
Broad Institute, USA

Bing Ren
UCSD, USA

Ben Scheres
ETH Zurich, Switzerland

Sandy Schmid
The Scripps Research Institute, USA

Luis Serrano
Center for Genomic Regulation, Spain

Mike Snyder
Yale University, USA

Alex Stark
IMP Vienna, Austria

Olga Troyanskaya
Princeton University, USA

Michael Tyers
University of Edinburgh, UK

Jonathan Weissman
UCSF, USA

Rick Young
Whitehead Institute, USA

Organisers

Eileen Furlong
EMBL Heidelberg, Germany

Frank Holstege
University Medical Centre Utrecht, The Netherlands

Marian Walhout
UMASS Medical School, USA

Topics

- Transcriptional control
- Systems analysis of basic cellular processes
- Regulatory networks
- Single cell biology
- Moving from genotype to phenotype
- Modeling complex systems

Brief history

Late 1980s: Poustka, Lennon, Lehrach: cDNAs spotted on nylon membranes

1990s: Affymetrix adapts microchip production technology for in situ oligonucleotide synthesis („commercial and heavily patent-fenced“)

1990s: Brown lab in Stanford develops two-colour spotted array technology („open and free“)

1998: Yeast cell cycle expression profiling on spotted arrays (Spellmann) and Affymetrix (Cho)

1999: Tumor type discrimination based on mRNA profiles (Golub)

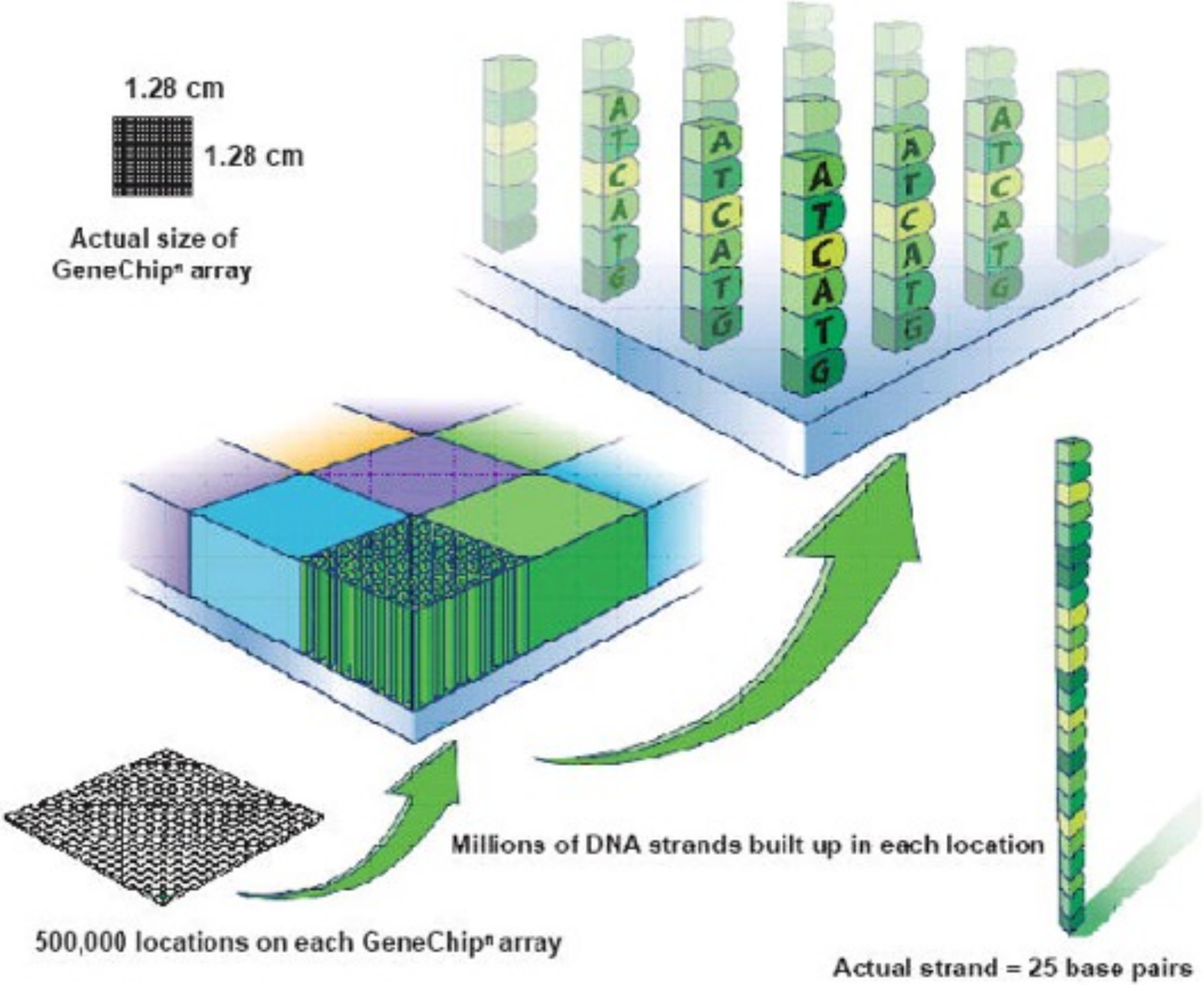
2000-ca. 2004: Affymetrix dominates the microarray market

Since ~2003: Nimblegen, Illumina, Agilent (and many others)

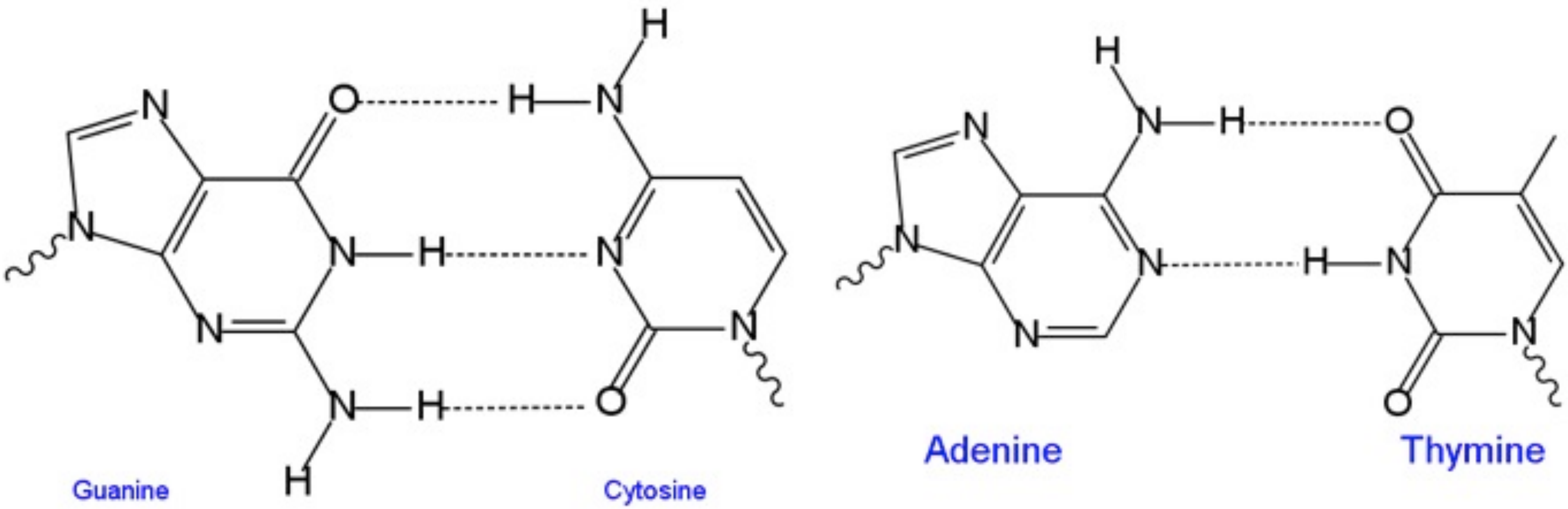
Throughout 2000‘s: CGH, CNVs, SNPs, ChIP, tiling arrays

Since ~2007: Next-generation sequencing (454, Solexa, ABI Solid,...)

Oligonucleotide microarrays

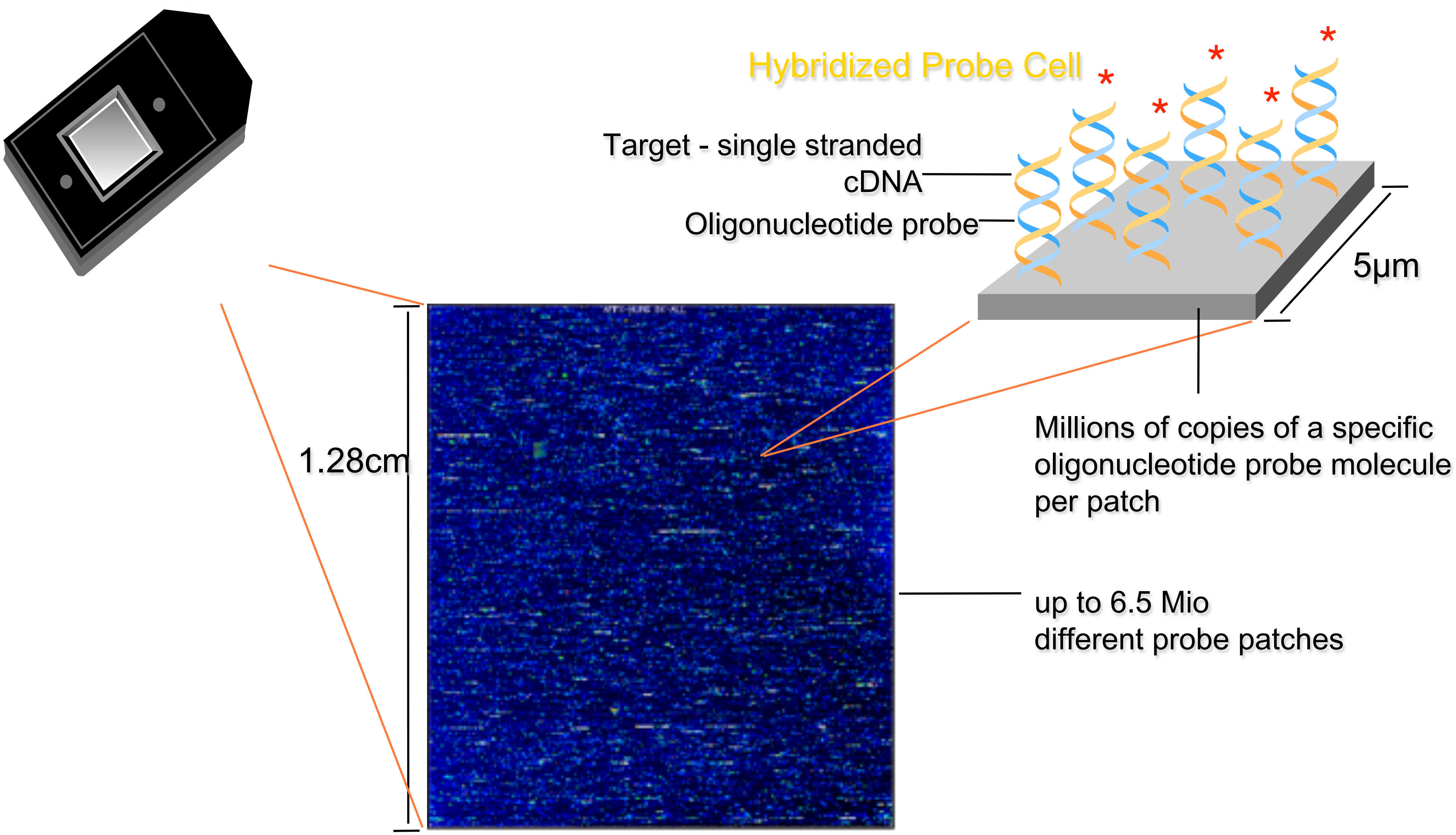

1.28 cm
1.28 cm
Actual size of GeneChip® array
A T C A T G
Millions of DNA strands built up in each location
500,000 locations on each GeneChip® array
Actual strand = 25 base pairs

Base Pairing



Ability to use hybridisation for constructing specific + sensitive probes at will is unique to DNA (cf. proteins, RNA, metabolites)

Oligonucleotide microarrays



GeneChip

Hybridized Probe Cell

Target - single stranded cDNA

Oligonucleotide probe

5µm

1.28cm

Millions of copies of a specific oligonucleotide probe molecule per patch

up to 6.5 Mio
different probe patches

Image of array after hybridisation and staining

Probe sets

GeneChip® Expression Array Design

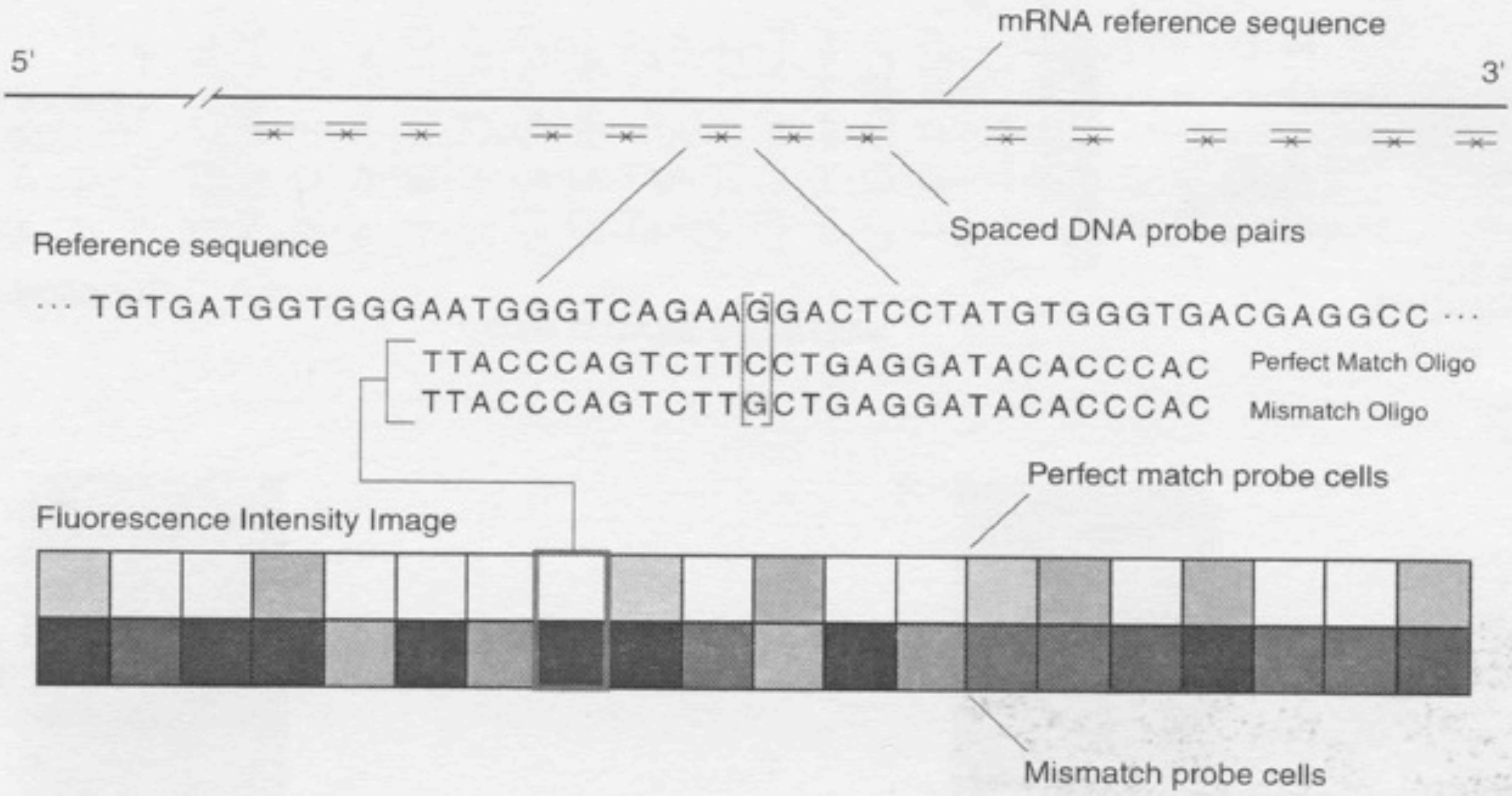

mRNA reference sequence
5'
3'
Spaced DNA probe pairs
Reference sequence
··· TGTGATGGTGGGAATGGGTCAGAAGGACTCCTATGTGGGTGACGAGGCC ···
TTACCCAGTCTTCCTGAGGATACACCCAC
Perfect Match Oligo
TTACCCAGTCTTGCTGAGGATACACCCAC
Mismatch Oligo
Perfect match probe cells
Fluorescence Intensity Image
Mismatch probe cells

Figure 1-3 Expression tiling strategy

Terminology for transcription arrays

Each target molecule (transcript) is represented by several oligonucleotides of (intended) length 25 bases

Probe: one of these 25-mer oligonucleotides

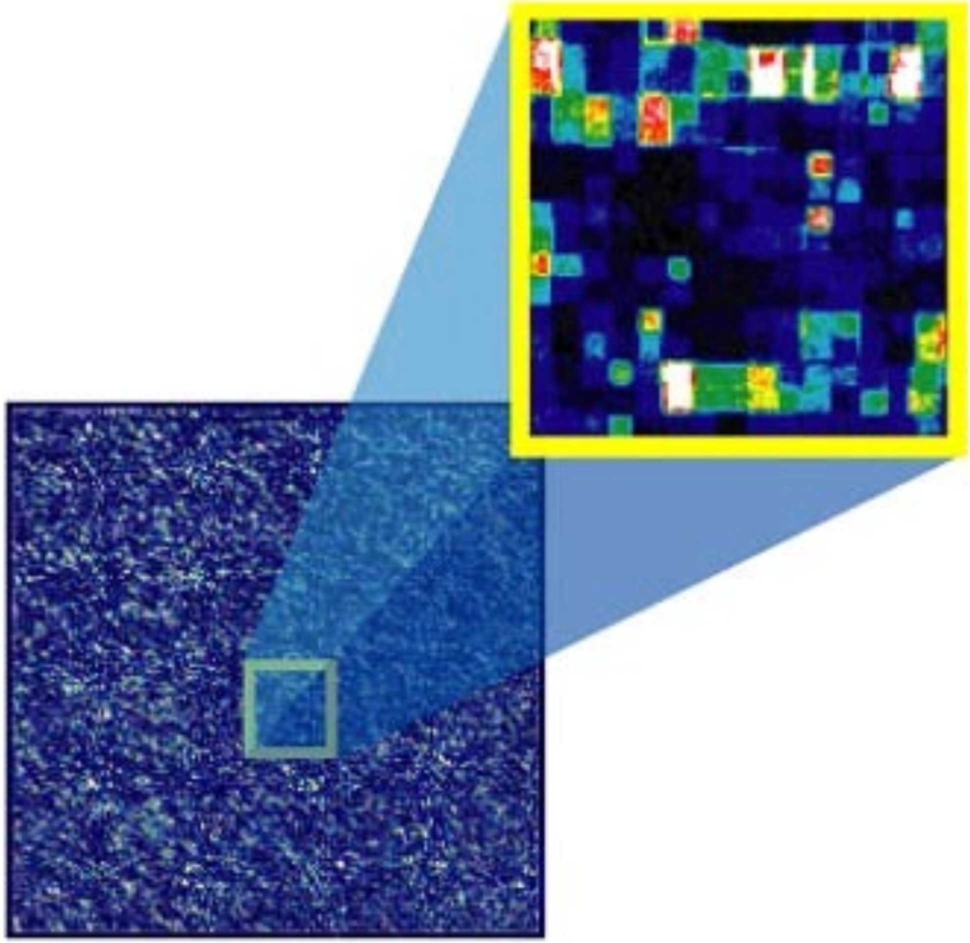
Probe set: a collection of probes (e.g. 11) targeting the same transcript

MGED/MIAME: „probe“ is ambiguous!

Reporter: the sequence

Feature: a physical patch on the array with molecules intended to have the same reporter sequence (one reporter can be represented by multiple features)

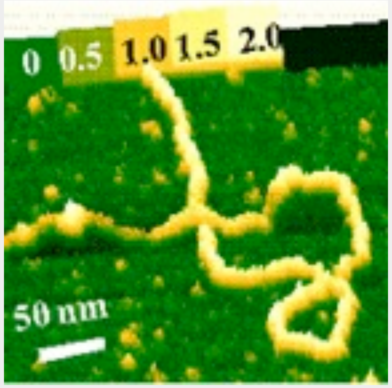
Image analysis



- several dozen pixels per feature
- segmentation
- summarisation into one number representing the intensity level for this feature

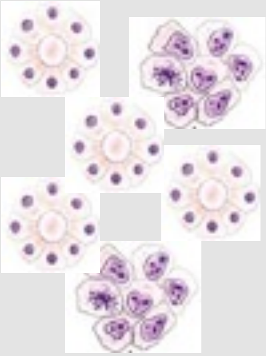
→ CEL file

μarray data



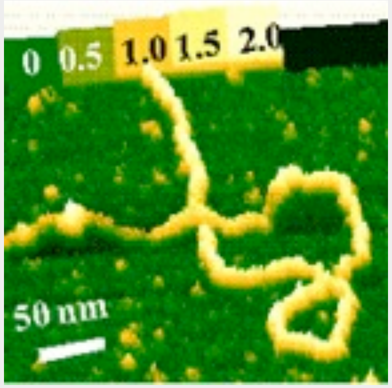
arrays:
probes = gene-specific DNA strands

μarray data



samples:

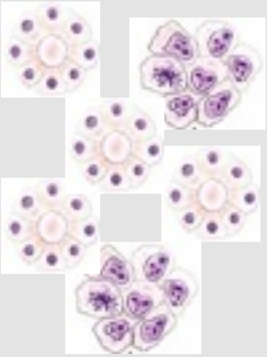
mRNA from tissue biopsies, cell lines



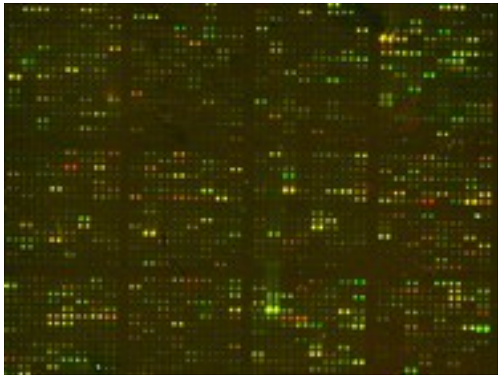
arrays:

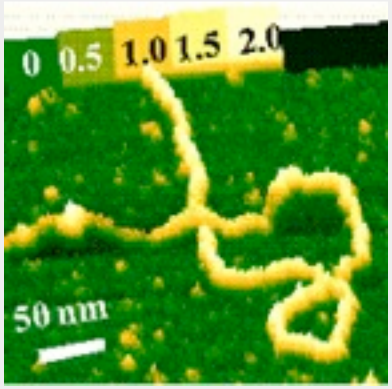
probes = gene-specific DNA strands

μarray data



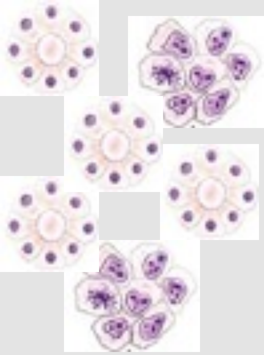
samples:
mRNA from tissue biopsies, cell lines



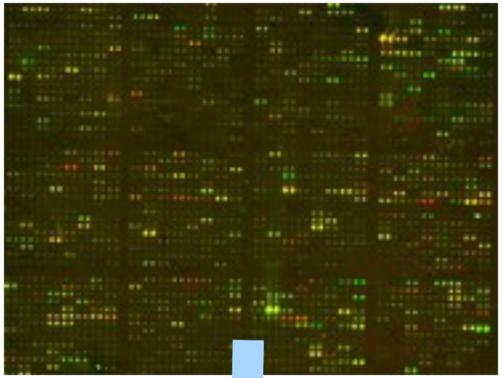


arrays:
probes = gene-specific DNA strands

μarray data

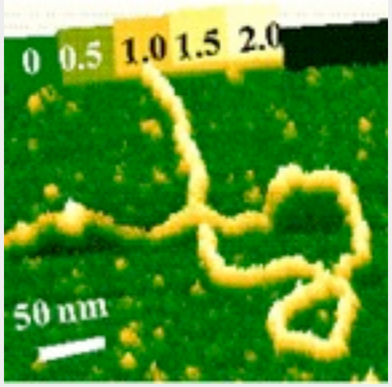


samples:
mRNA from tissue biopsies, cell lines



fluorescent detection of the amount of sample-probe binding

	tissue A	tissue B	tissue C
ErbB2	0.02	1.12	2.12
VIM	1.1	5.8	1.8
ALDH4	2.2	0.6	1.0
CASP4	0.01	0.72	0.12
LAMA4	1.32	1.67	0.67
MCAM	4.2	2.93	3.31



arrays:
probes = gene-specific DNA strands

Microarray Infrastructure in Bioconductor

Platform-specific data import and initial processing

Affymetrix 3’ IVT (e.g. Human U133 Plus 2.0, Mouse 430 2.0):

`affy`

Affymetrix Exon (e.g. Human Exon 1.0 ST):

`oligo, exonmap, xps`

Affymetrix SNP arrays:

`oligo`

Nimblegen tiling arrays (e.g. for ChIP-chip):

`Ringo`

Affymetrix tiling arrays (e.g. for ChIP-chip):

`Starr`

Illumina bead arrays:

`beadarray, lumi`

http://www.bioconductor.org/docs/workflows/oligoarrays

Flexible data import

Using generic `R` I/O functions and constructors

`Biobase`

`limma`

Chapter *Two Color Arrays* in the useR-book.

`limma` user guide

Normalisation and quality assessment

`preprocessCore`

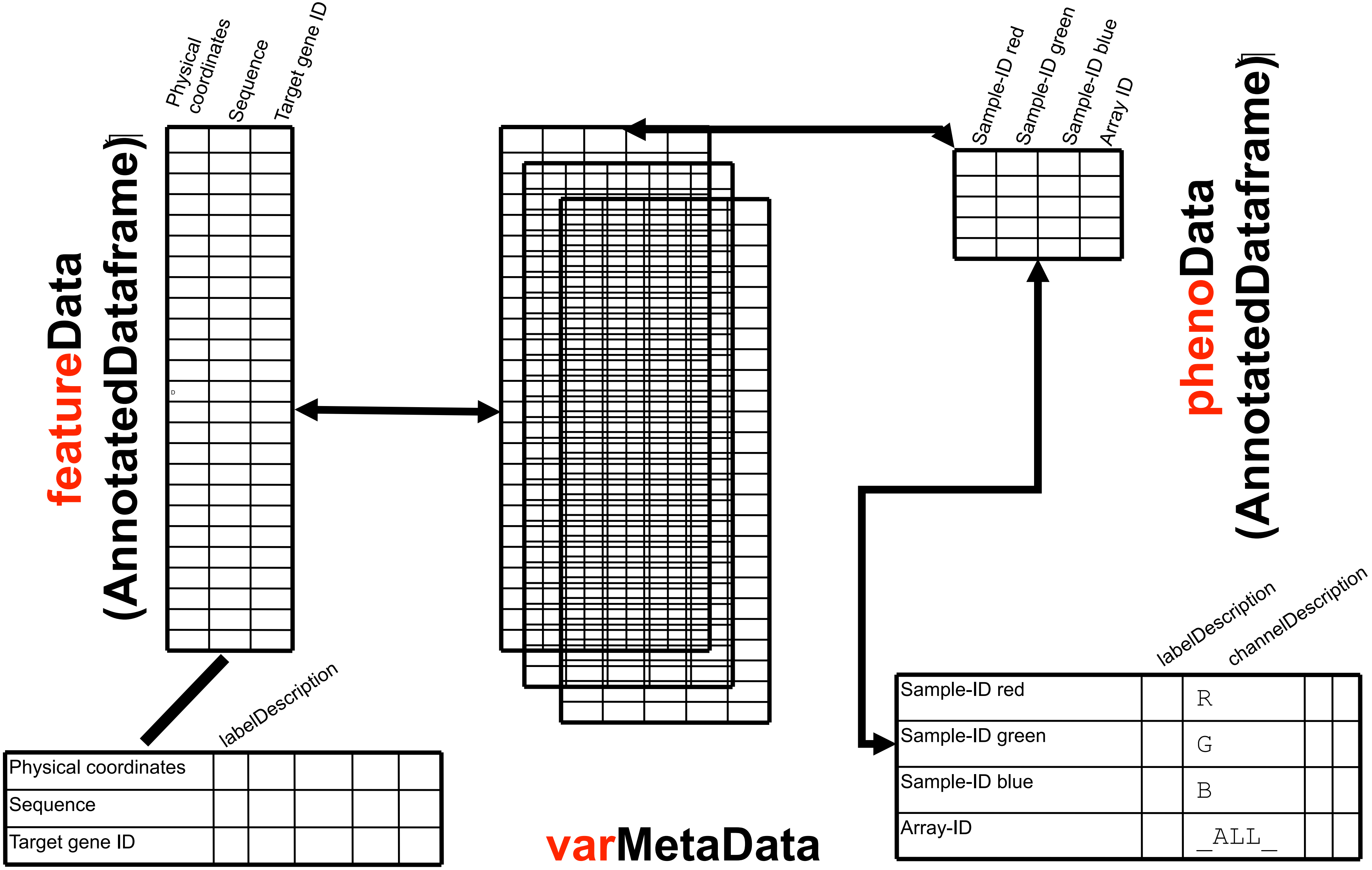
`limma`

`vsn`

`arrayQualityMetrics`

NChannelSet

assayData can contain N=1, 2, ..., matrices of the same size



	labelDescription				
Physical coordinates					
Sequence					
Target gene ID					

	labelDescription	channelDescription		
Sample-ID red		R		
Sample-ID green		G		
Sample-ID blue		B		
Array-ID		_ALL_		

Annotation / Metadata

Keeping data together with the metadata (about reporters, target genes, samples, experimental conditions, ...) is one of the major principles of Bioconductor

- avoid alignment bugs
- facilitate discovery

Often, the same microarray design is used for multiple experiments. Duplicating that metadata every time would be inefficient, and risk versioning mismatches ⇒

instead of `featureData`, just keep a pointer to an annotation package.

(In principle, one could also want to do this for samples.)

Annotation infrastructure for Affymetrix

For `affy`:

`hgu133plus2.db` “all available” information about target genes

`hgu133plus2cdf` maps the physical features on the array to probesets

`hgu133plus2probe` nucleotide sequence of the features (e.g. for `gcrma`)

For `oligo`:

`pd.*` packages should rationalise and simplify this - but not there yet....

Genotyping

`crlmm` Genotype Calling (CRLMM) and Copy Number Analysis tool for Affymetrix SNP 5.0 and 6.0 and Illumina arrays.

`snpMatrix`

.... others

See also:
Genome-wide association study of CNVs in 16,000 cases of eight common diseases and 3,000 shared controls, The Wellcome Trust Case Control Consortium, Nature 464, 713-720 (Box 1).

Transcriptomics

Microarray Analysis Tasks

Data import

reformating and setup/curation of the metadata

Normalisation

Quality assessment & control

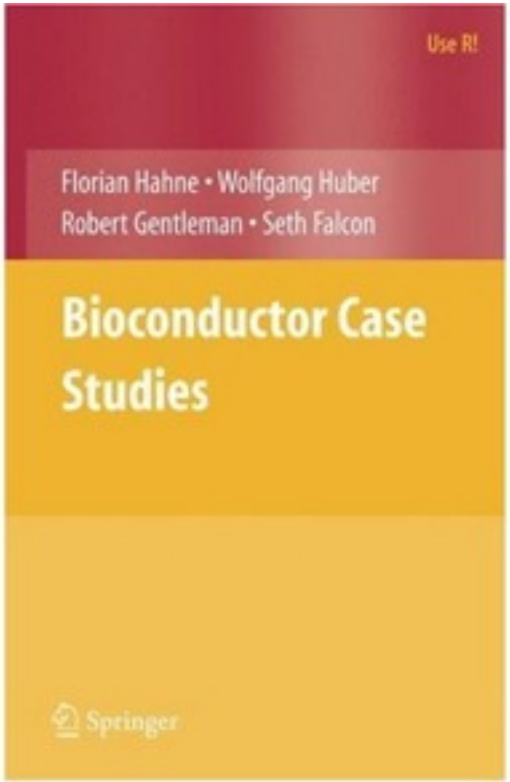
Differential expression

Using gene-level annotation

Gene set enrichment analysis

Clustering & Classification

Integration of other datasets



Use R!

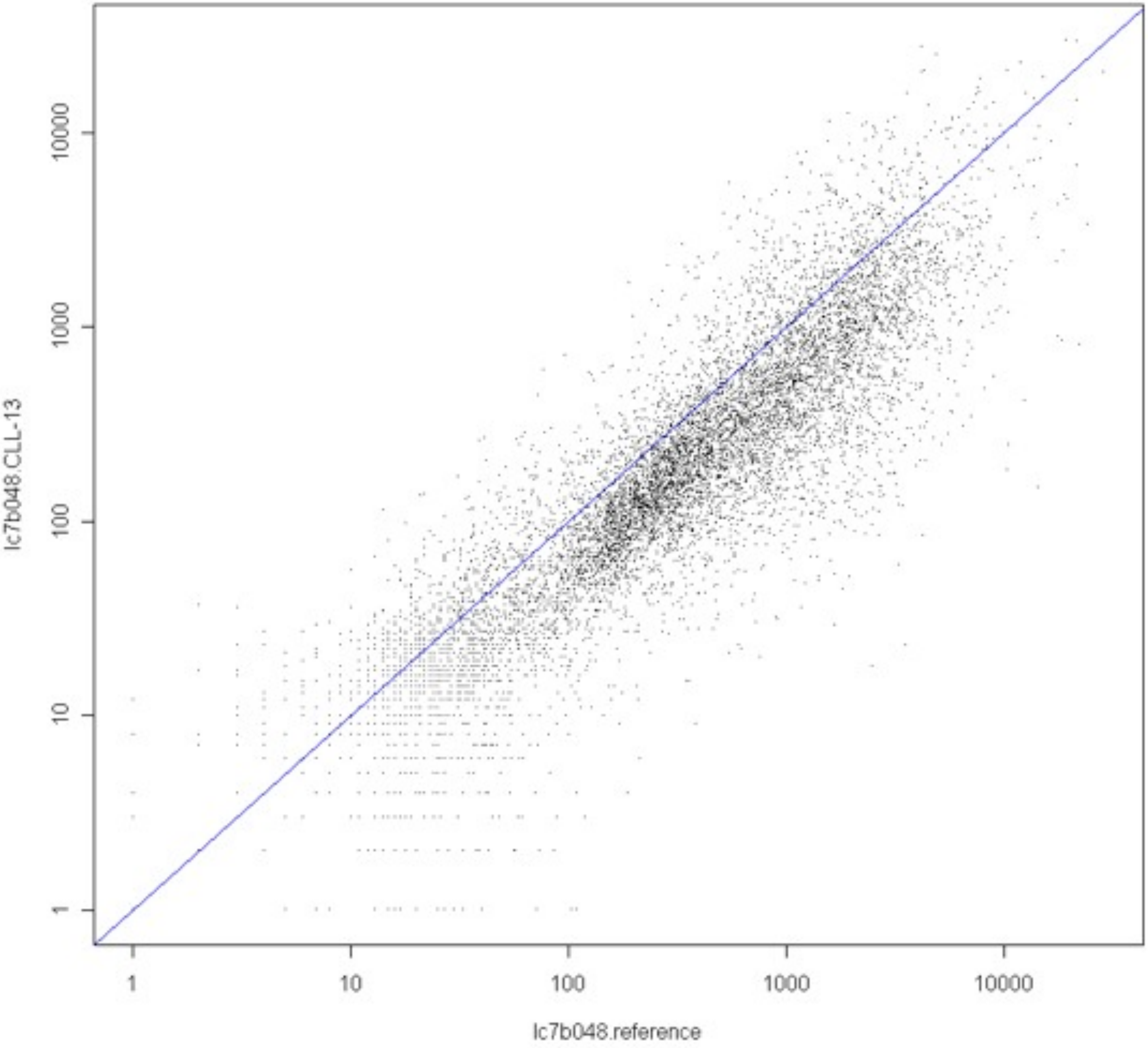
Florian Hahne • Wolfgang Huber
Robert Gentleman • Seth Falcon

Bioconductor Case Studies

Springer

Why do you need ‘normalisation’?

Systematic drift effects



From: lymphoma dataset

vsn package

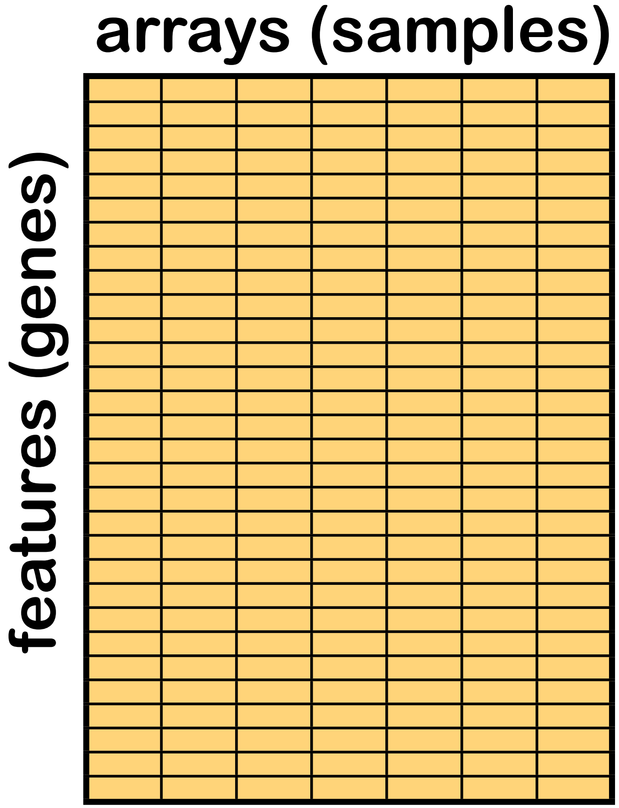
Alizadeh et al., Nature 2000

Quantile normalisation

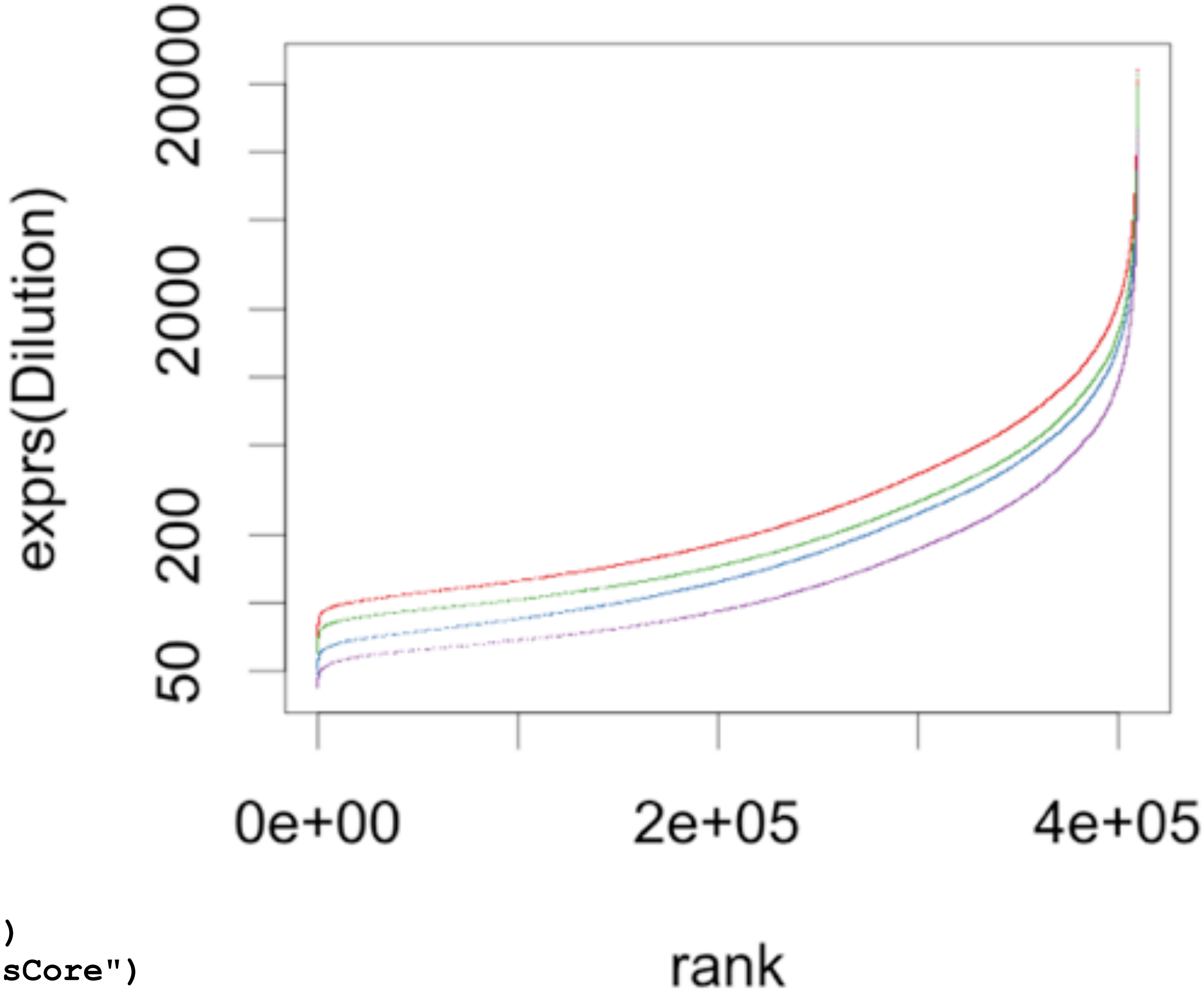
Within each column (array), replace the intensity values by their rank

For each rank, compute the average of the intensities with that rank, across columns (arrays)

Replace the ranks by those averages



Ben Bolstad 2001

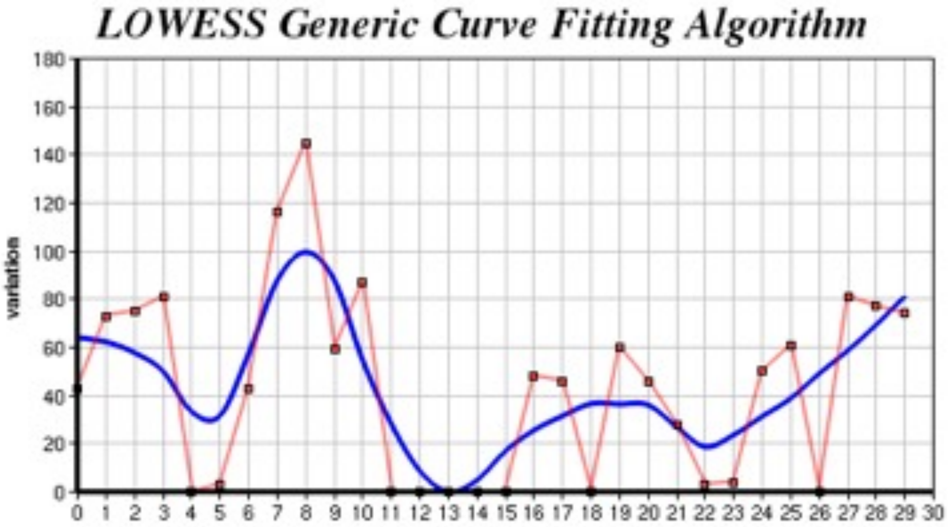


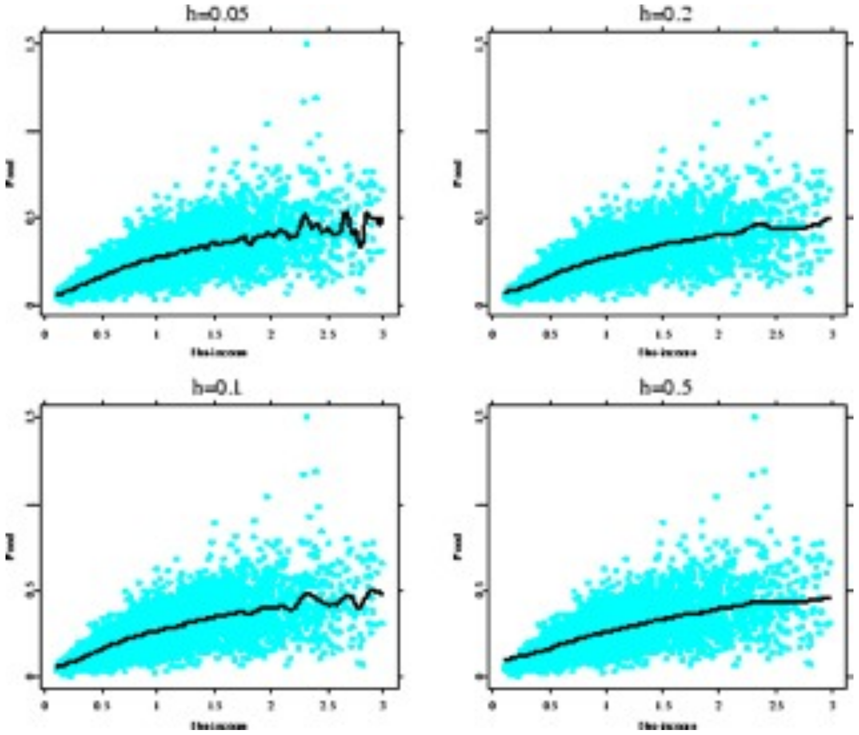
```
library("affydata")
library("preprocessCore")
library("RColorBrewer")
data("Dilution")
nr = apply(exprs(Dilution), 2, rank)
nq = normalize.quantiles(exprs(Dilution))
matplot(nr, exprs(Dilution), pch=".", log="y",
           xlab="rank", col=brewer.pal(9,"Set1"))
```

loess normalisation

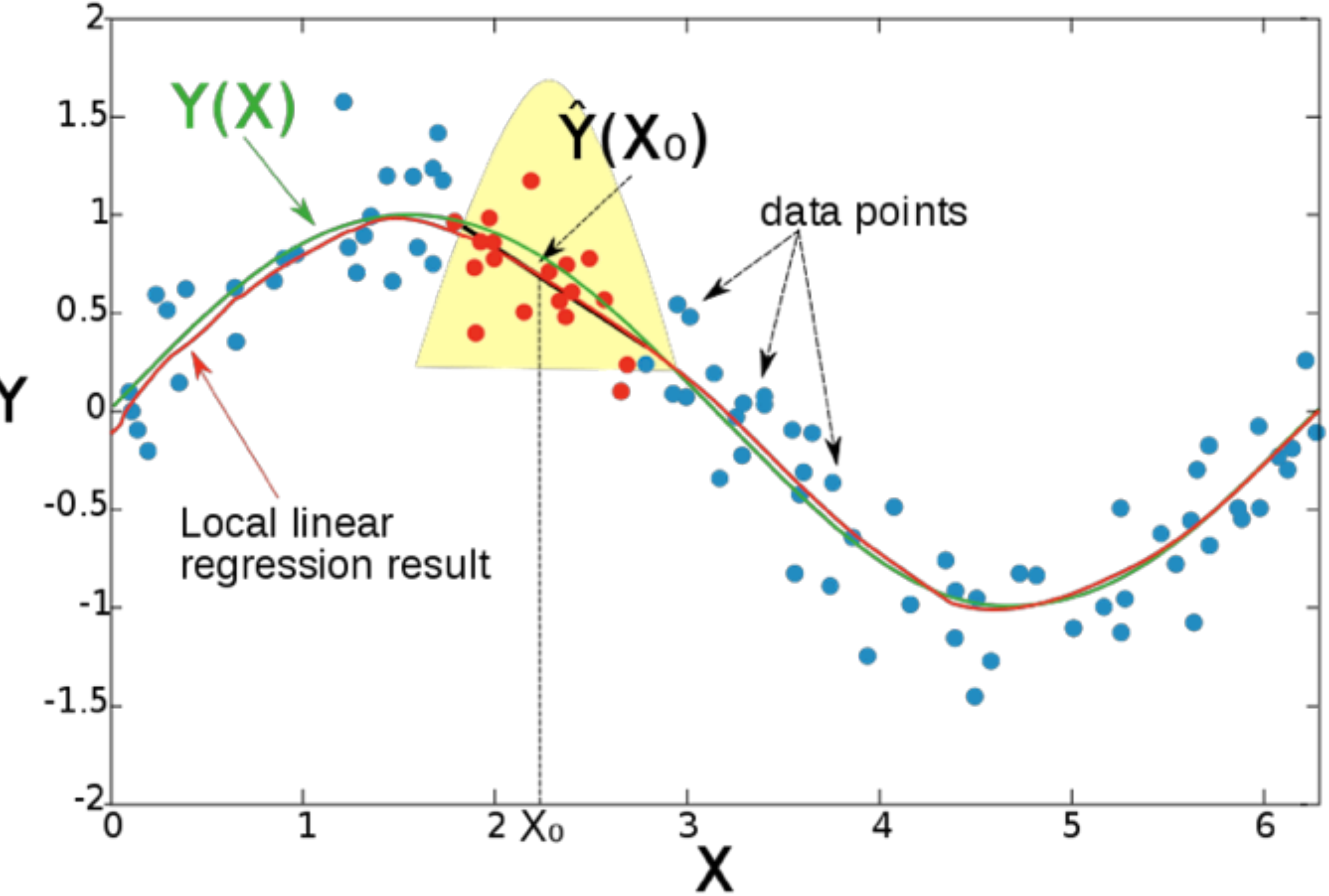
"loess" normalisation

loess (locally weighted scatterplot smoothing): an algorithm for robust local polynomial regression by W. S. Cleveland and colleagues (AT&T, 1980s) and handily available in R





Local polynomial regression



Local polynomial regression

Global polynomial regression

$$y(x) = a_p x^p + \ldots + a_2 x^2 + a_1 x + a_0 + \varepsilon$$

applied to data $(x_1, y_1), \ldots, (x_n, y_n)$, with equal weights

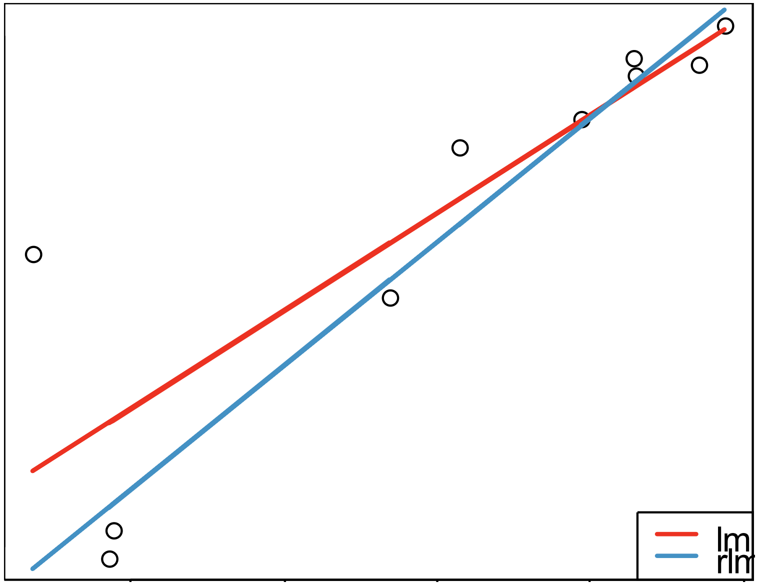
resulting in global fit $(a_p, \ldots, a_1)$

Local polynomial regression around ν

with weights $h_b(x - \nu)$

resulting in local fit $(a_p(v), \ldots, a_1(v))$

Making regression against outliers

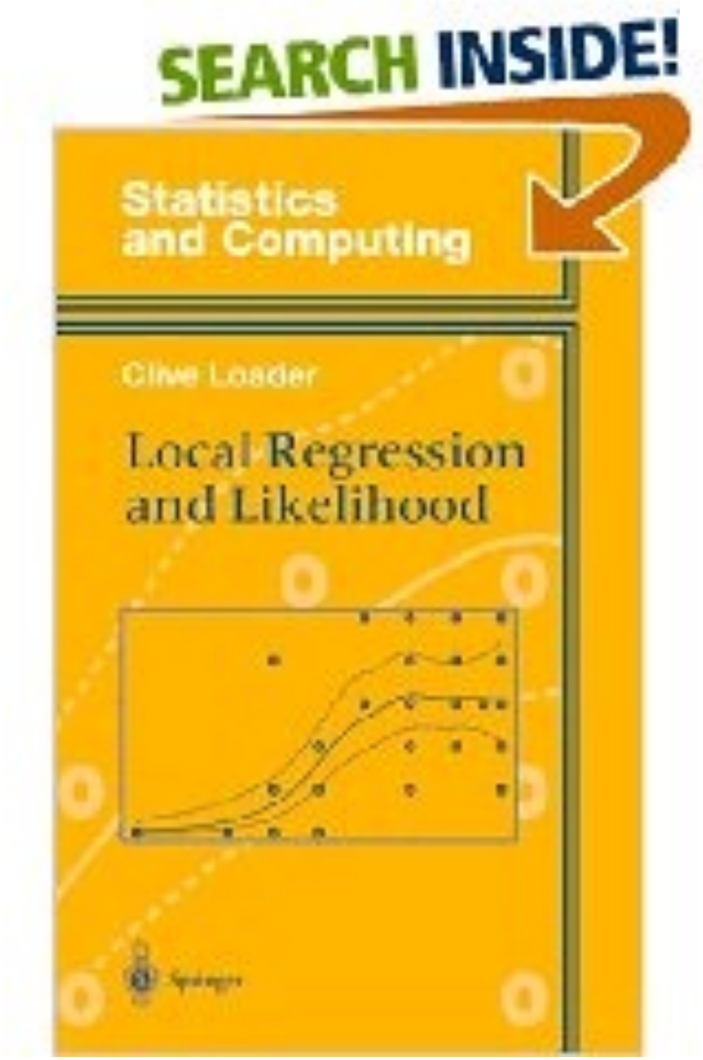


$$\text{OLS:}\quad \sum_{i=1}^{n}\left(y_i - f(x_i)\right)^2 \rightarrow \min$$

$$\text{M-est.:}\quad \sum_{i=1}^{n} M\left(y_i - f(x_i)\right) \rightarrow \min$$

$$\text{LTS:}\quad \mathsf{Q}\left(\{y_i - f(x_i) \mid i = 1,\dots,n\}\right) \rightarrow \min$$

P.J. Huber: *Robust Statistics*
P. Rousseeuw: *Robust regression and outlier detection*



C. Loader

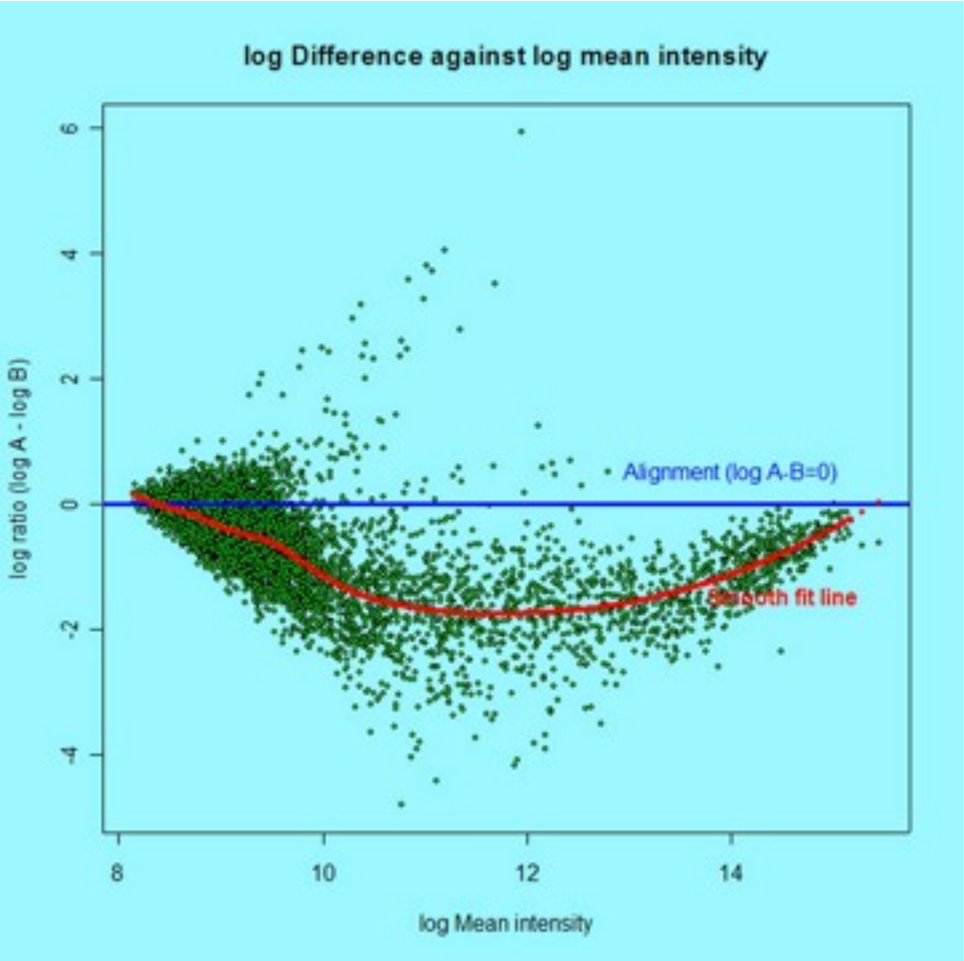
Local Regression and Likelihood

Springer Verlag

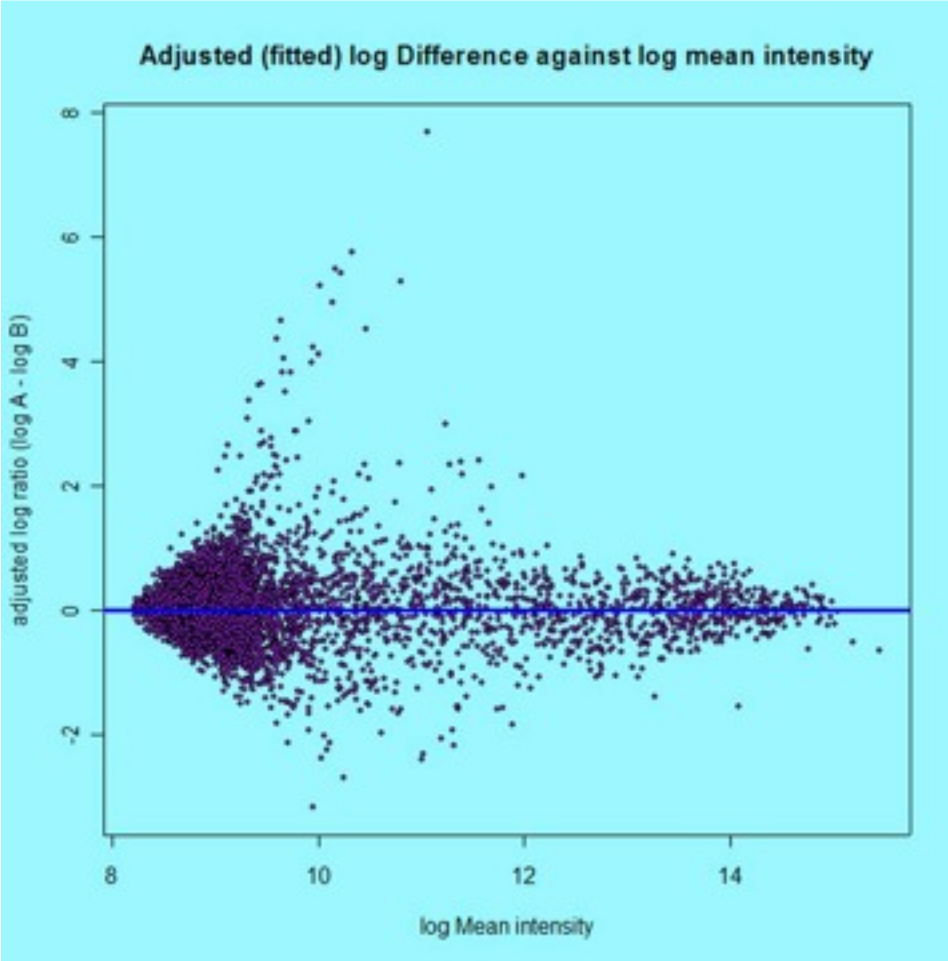
loess normalisation

- local polynomial regression of *M* against *A*
- 'normalised' M-values are the residuals

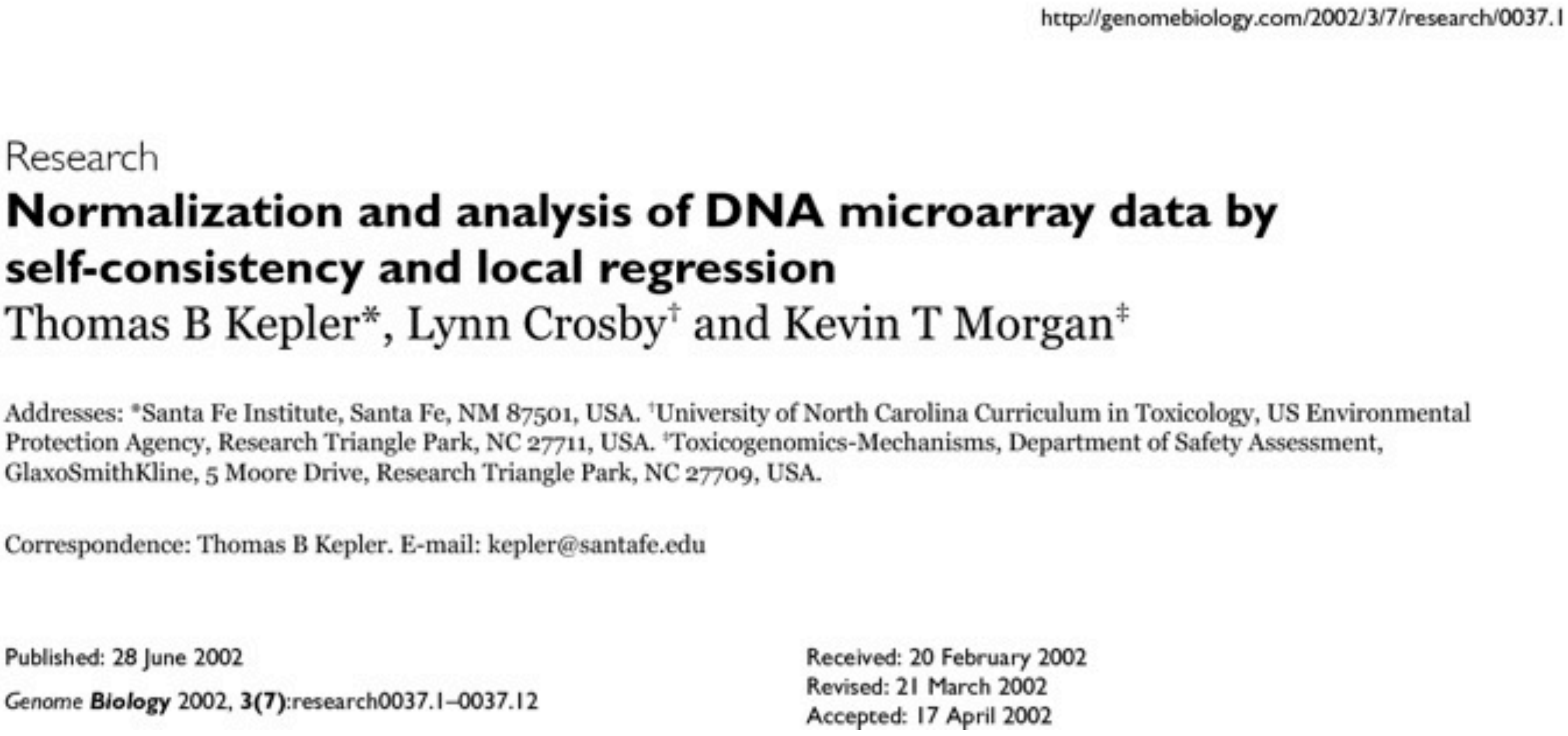
before



after



local polynomial regression normalisation in >2 dimensions



http://genomebiology.com/2002/3/7/research/0037.1

Research

Normalization and analysis of DNA microarray data by self-consistency and local regression

Thomas B Kepler[*], Lynn Crosby[†] and Kevin T Morgan[‡]

Addresses: [*]Santa Fe Institute, Santa Fe, NM 87501, USA. [†]University of North Carolina Curriculum in Toxicology, US Environmental Protection Agency, Research Triangle Park, NC 27711, USA. [‡]Toxicogenomics-Mechanisms, Department of Safety Assessment, GlaxoSmithKline, 5 Moore Drive, Research Triangle Park, NC 27709, USA.

Correspondence: Thomas B Kepler. E-mail: kepler@santafe.edu

Published: 28 June 2002

Genome **Biology** 2002, **3(7)**:research0037.1–0037.12

Received: 20 February 2002
Revised: 21 March 2002
Accepted: 17 April 2002

n-dimensional local regression model for microarray normalisation

$$Y_{kij} = \alpha_k + \nu_{ij}(\alpha_k) + \delta_{ik} + \sigma(\alpha_k)\varepsilon_{kij}$$

Y_{kij}: log-intensity of gene k in condition i, replicate j

α_k : baseline value gene k (A-value)

δ_{ik}: effect of treatment i on gene k

$\nu_{ij}(\alpha_k)$: intensity-dependent normalisation function for array ij

$\sigma(\alpha_k)$: intensity-dependent error scale function

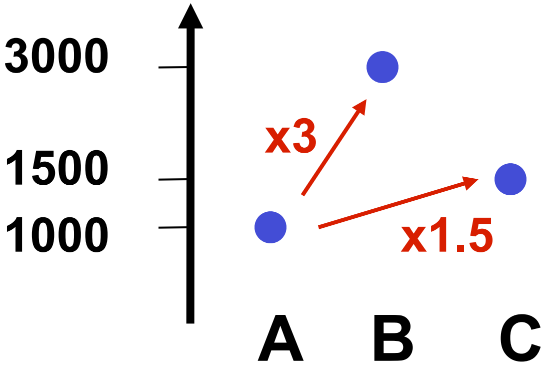
ε_{kij}: i.i.d. error term

An algorithm for fitting this robustly is described (roughly) in the paper. They only provided software as a binary for Windows. The paper has 129 citations in according to Google scholar (6/2010), but the method has not found much use.

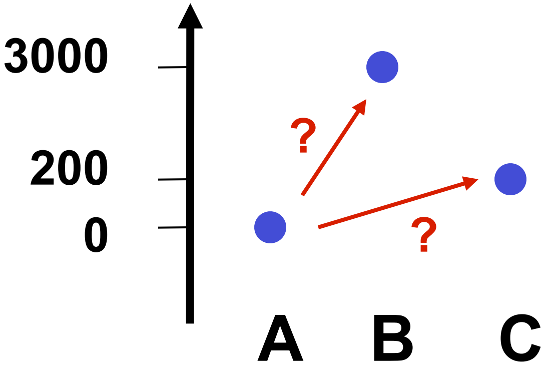
Estimating relative expression (fold-changes)

▶ ratios and fold changes

Fold changes are useful to describe continuous changes in expression



But what if the gene is “off” (below detection limit) in one condition?



▶ ratios and fold changes

The idea of the log-ratio (base 2)

- **0: no change**
- **+1: up by factor of $2^1 = 2$**
- **+2: up by factor of $2^2 = 4$**
- **-1: down by factor of $2^{-1} = 1/2$**
- **-2: down by factor of $2^{-2} = ¼$**

▶ ratios and fold changes

The idea of the log-ratio (base 2)

0: no change
+1: up by factor of $2^1 = 2$
+2: up by factor of $2^2 = 4$
-1: down by factor of $2^{-1} = 1/2$
-2: down by factor of $2^{-2} = ¼$

A unit for measuring changes in expression: assumes that a change from 1000 to 2000 units has a similar biological meaning to one from 5000 to 10000.
…. data reduction

ratios and fold changes

The idea of the log-ratio (base 2)

- 0: no change
- +1: up by factor of $2^{1} = 2$
- +2: up by factor of $2^{2} = 4$
- -1: down by factor of $2^{-1} = 1/2$
- -2: down by factor of $2^{-2} =$ ¼

A unit for measuring changes in expression: assumes that a change from 1000 to 2000 units has a similar biological meaning to one from 5000 to 10000.

…. data reduction

What about a change from 0 to 500?

- conceptually
- noise, measurement precision

▶ What is wrong with microarray data?

Many data are measured in definite units:

- time in seconds
- lengths in meters
- energy in Joule, etc.

Climb Mount Plose (2465 m) from Brixen (559 m) with weight of 76 kg, working against a gravitation field of strength 9.81 m/s^2 :

$$(2465 - 559) \cdot 76 \cdot 9.81 \;\; m\ kg\ m/s^2$$

$$= 1\ 421\ 037\ kg\ m^2\ s^{-2}$$

$$= 1\ 421.037\ kJ$$

▶ What is wrong with microarray data?

Many data are measured in definite units:

- time in seconds
- lengths in meters
- energy in Joule, etc.

Climb Mount Plose (2465 m) from Brixen (559 m) with weight of 76 kg, working against a gravitation field of strength 9.81 m/s^2 :

$$(2465 - 559) \cdot 76 \cdot 9.81 \;\; m \, kg \, m/s^2$$
$$= 1\,421\,037 \; kg \, m^2 \, s^{-2}$$
$$= 1\,421.037 \; kJ$$

Two component error model and variance stabilisation

The two component model

measured intensity = offset + gain × true abundance

$$y_{ik} = a_{ik} + b_{ik} x_k$$

$$a_{ik} = a_i + \varepsilon_{ik}$$

a_i per-sample offset

ε_{ik} additive noise

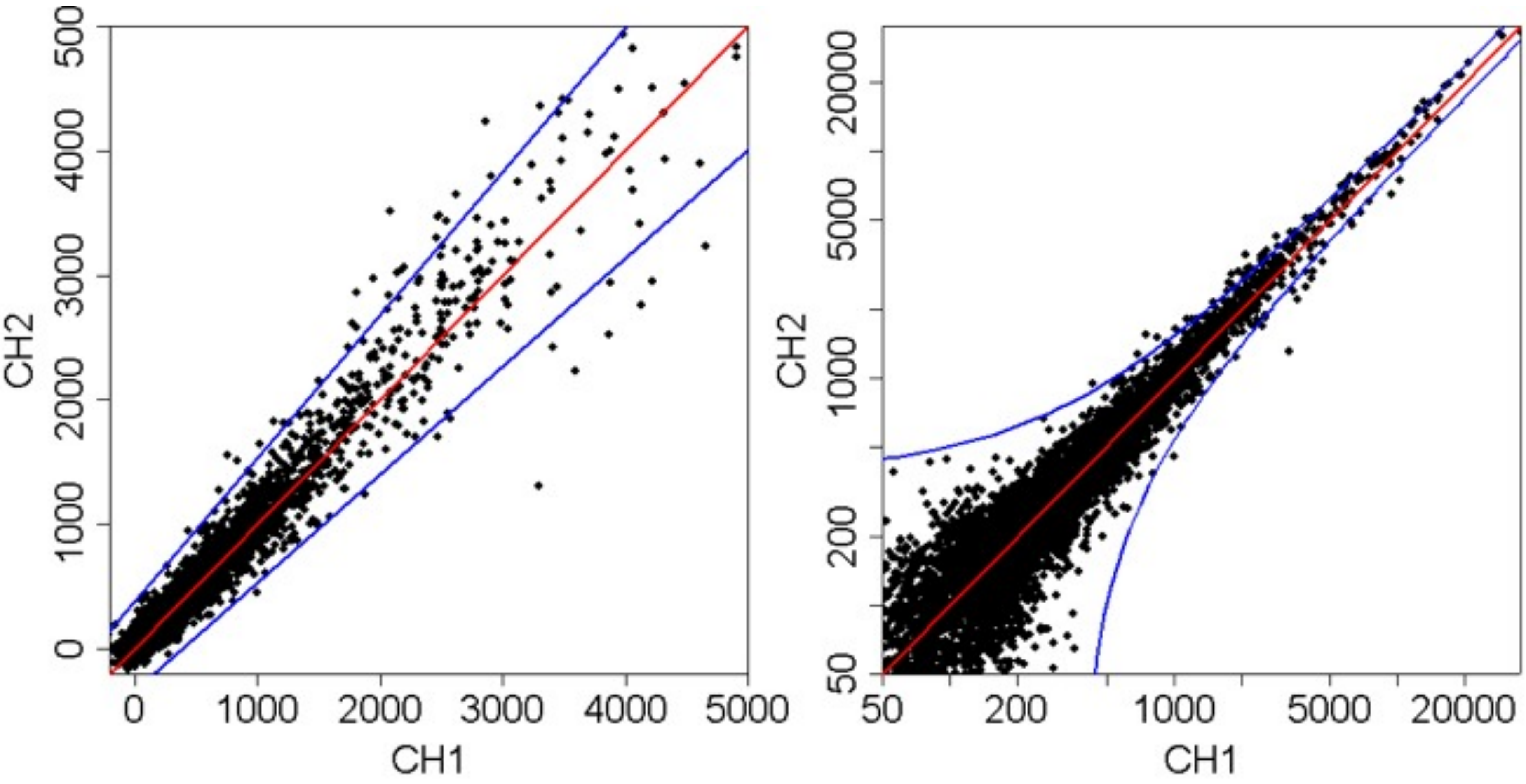
$$b_{ik} = b_i \, b_k \, \exp(\eta_{ik})$$

b_i per-sample gain factor

b_k sequence-wise probe efficiency

η_{ik} multiplicative noise

The two-component model

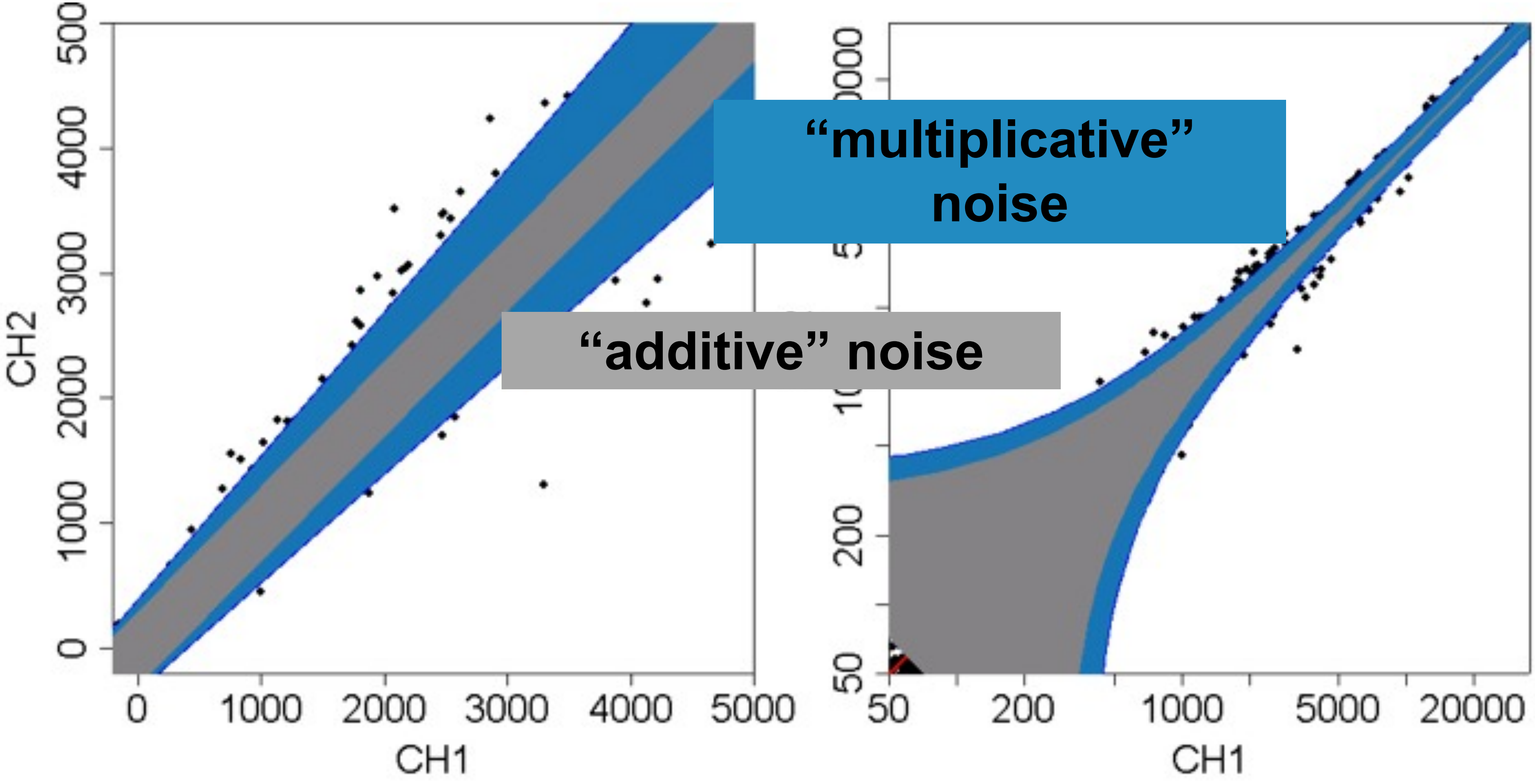


raw scale

log scale

B. Durbin, D. Rocke, JCB 2001

The two-component model



“multiplicative” noise

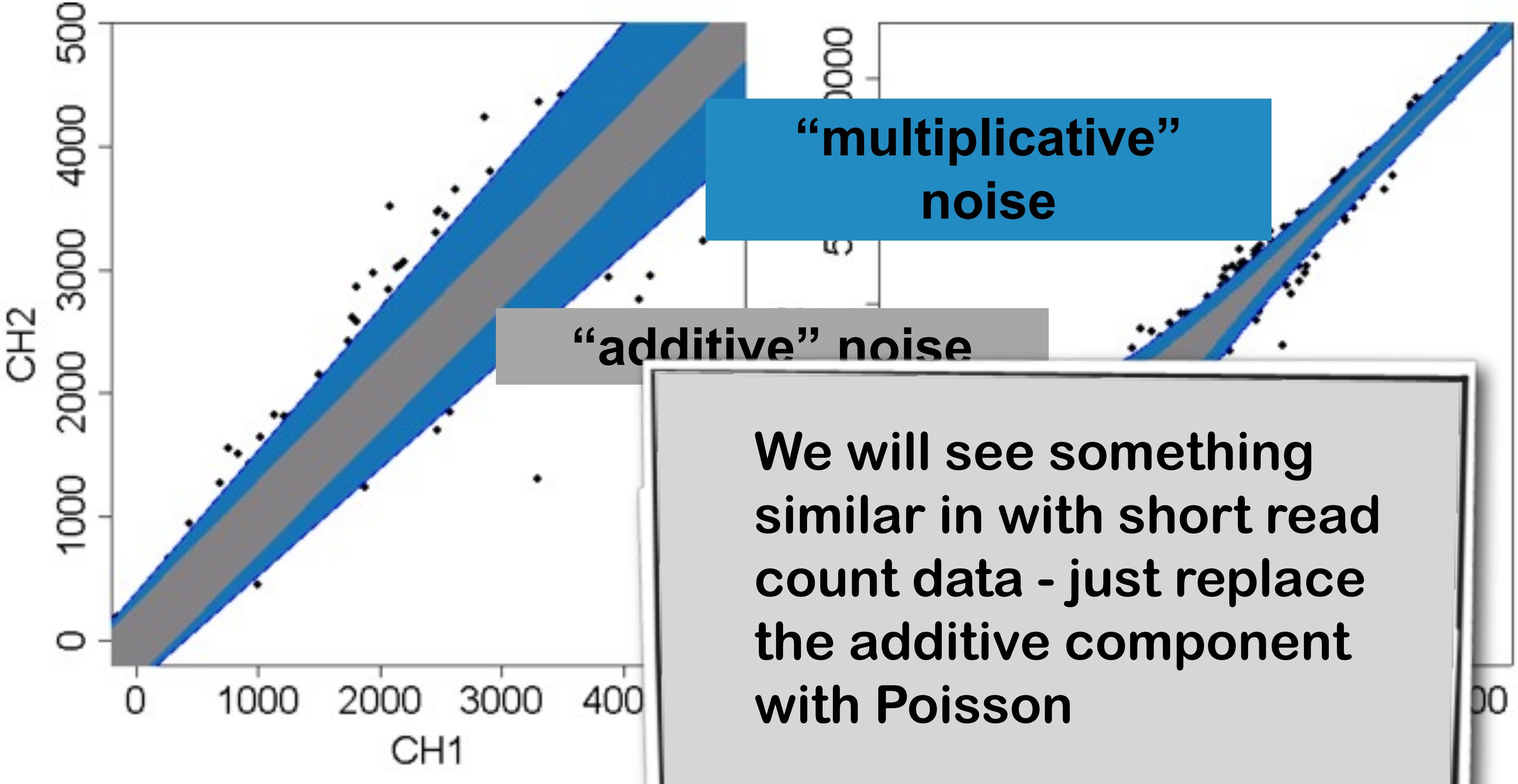
“additive” noise

raw scale

log scale

B. Durbin, D. Rocke, JCB 2001

The two-component model



“multiplicative” noise

“additive” noise

We will see something similar in with short read count data - just replace the additive component with Poisson

raw scale

log scale

B. Durbin, D. Rocke, JCB 2001

The additive-multiplicative error model

Trey Ideker et al.: JCB (2000)

David Rocke and Blythe Durbin: JCB (2001), Bioinformatics (2002)

Use for robust affine regression normalisation: W. Huber, Anja von Heydebreck et al. Bioinformatics (2002).

For background correction in RMA: R. Irizarry et al., Biostatistics (2003).

▶ Parameterization

$$y = a + \varepsilon + bx\,(1+\eta)$$

$$y = a + \varepsilon + bx\,\exp(\eta)$$

two practically equivalent forms ($\eta \ll 1$)

a: average background	on one array, for one color, the same for all features	also dependent on the reporter sequence
ε: background fluctuations	same distribution in whole experiment	different distributions
b: average gain factor	on one array, for one color, the same for all features	intensity dependent
η: gain fluctuations	same distribution in whole experiment	different distributions

▶ variance stabilizing transformations

X_u a family of random variables with

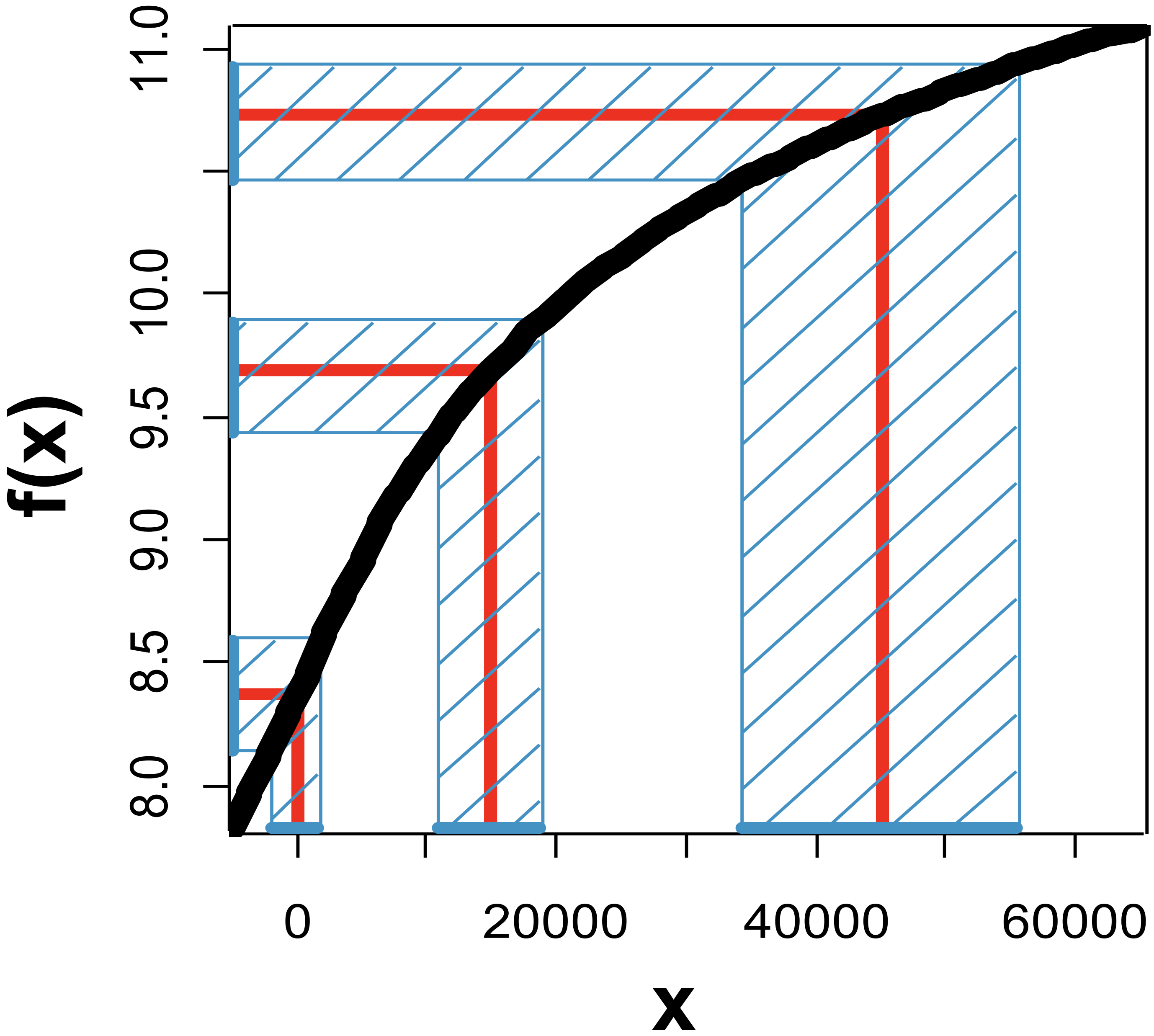
E(X_u) = u and Var(X_u) = $v(u)$. Define

$$f(x) = \int^{x} \frac{du}{\sqrt{v(u)}}$$

Then, var $f(X_u)$ ≈ does not depend on u

Derivation: linear approximation,
relies on smoothness of $v(u)$.

▶ variance stabilizing transformation



▶ variance stabilizing transformations

$$\boxed{f(x) = \int^{x} \frac{1}{\sqrt{v(u)}} du}$$

1.) constant variance ('additive') $v(u) = s^2 \quad \Rightarrow \quad f \propto u$

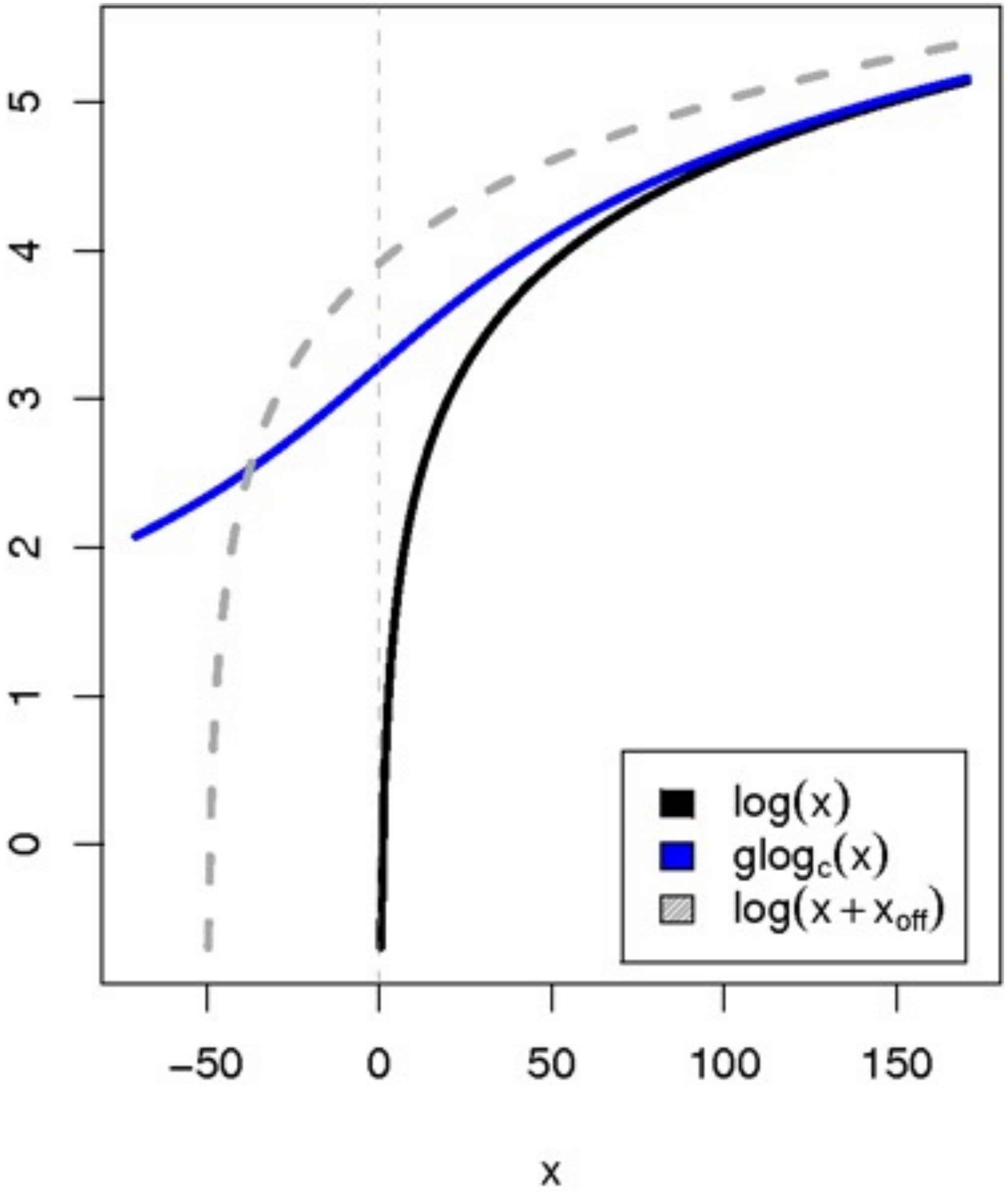
2.) constant CV ('multiplicative') $v(u) \propto u^2 \Rightarrow f \propto \log u$

3.) offset $v(u) \propto (u + u_0)^2 \quad \Rightarrow \quad f \propto \log(u + u_0)$

4.) additive and multiplicative

$$v(u) \propto (u + u_0)^2 + s^2 \quad \Rightarrow \quad f \propto \operatorname{arsinh} \frac{u + u_0}{s}$$

▶ the “glog” transformation



$$\mathrm{glog}_2(x,c) = \log_2\left(\frac{x+\sqrt{x^2+c^2}}{2}\right)$$

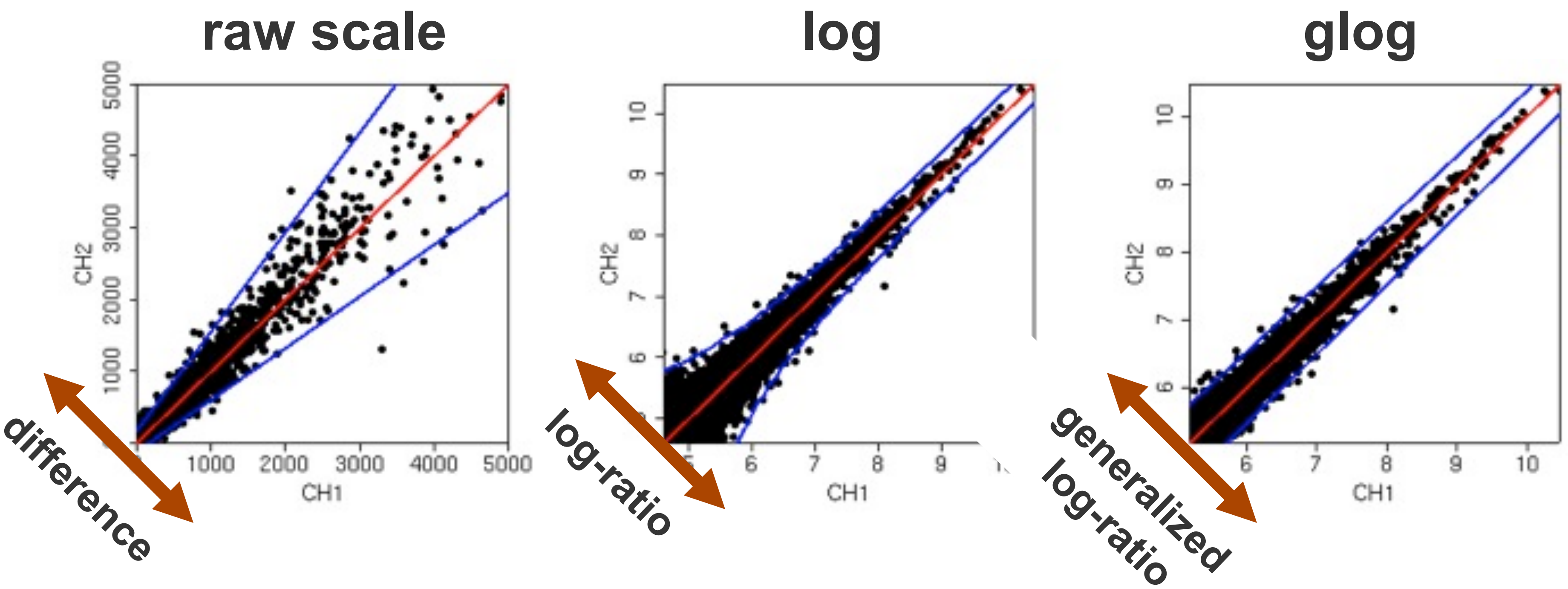
$$\mathrm{glog}_e(x,1) + \log_e 2 = \mathrm{arsinh}(x)$$

P. Munson, 2001

D. Rocke & B. Durbin, ISMB 2002

W. Huber et al., ISMB 2002

glog



raw scale

log

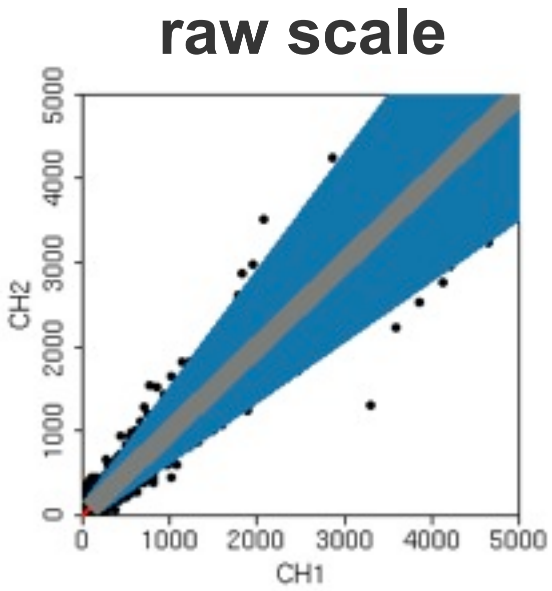
glog

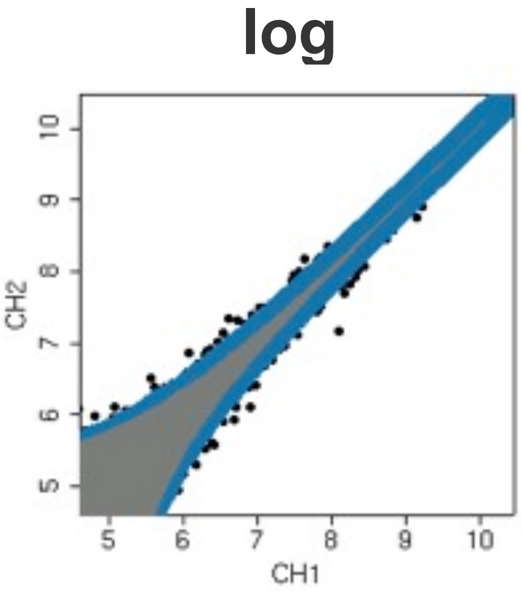
difference

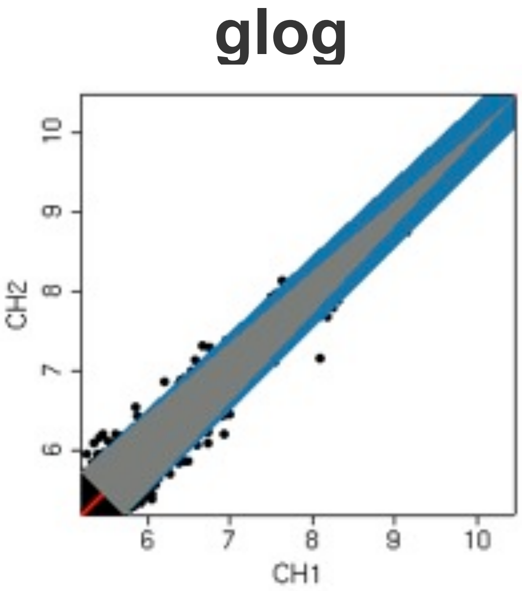
log-ratio

generalized log-ratio

▶ glog



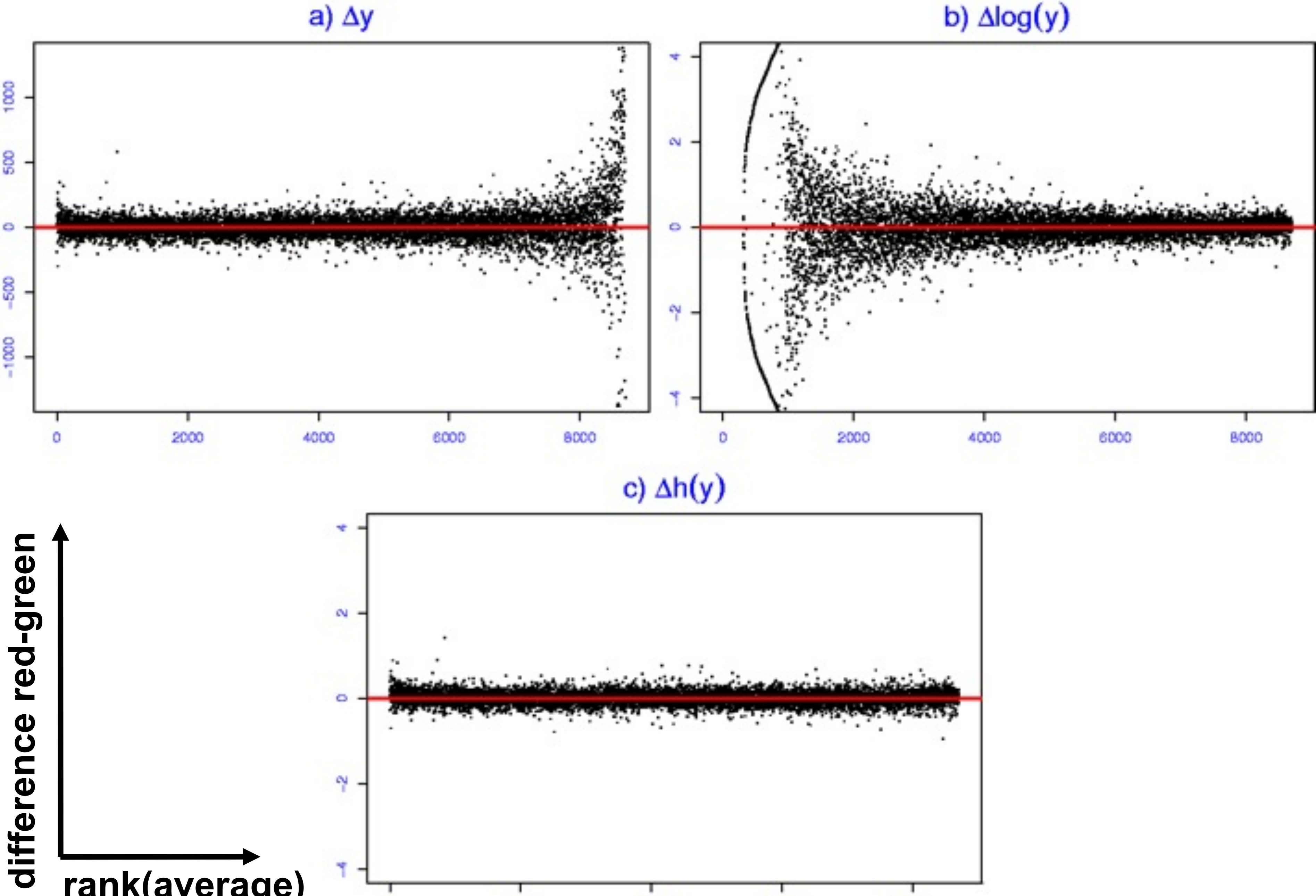




variance:

▨ constant part

🟦 proportional part


a) Δy
1000
500
0
-500
-1000
0
2000
4000
6000
8000
b) Δlog(y)
4
2
0
-2
-4
0
2000
4000
6000
8000
c) Δh(y)
4
2
0
-2
-4
0
2000
4000
6000
8000
difference red-green
rank(average)

Parameter estimation

$$\operatorname{arsinh}\frac{y_{ki} - a_i}{b_i} = \mu_k + \varepsilon_{ki}, \qquad \varepsilon_{ki} : N(0, c^2)$$

Parameter estimation

$$\operatorname{arsinh}\frac{y_{ki}-a_i}{b_i} = \mu_k + \varepsilon_{ki}, \qquad \varepsilon_{ki} : N(0, c^2)$$

measured intensity = offset + gain * true abundance

$$y_{ik} = a_{ik} + b_{ik} x_{ik}$$

$$a_{ik} = a_i + L_{ik} + \varepsilon_{ik}$$

a_i per-sample offset

L_{ik} local background provided by image analysis

$\varepsilon_{ik} \sim N(0, b_i^2 s_1^2)$

"additive noise"

$$b_{ik} = b_i \, b_k \, \exp(\eta_{ik})$$

b_i per-sample normalization factor

b_k sequence-wise labeling efficiency

$\eta_{ik} \sim N(0, s_2^2)$

"multiplicative noise"

Parameter estimation

$$\boxed{\operatorname{arsinh}\frac{y_{ki} - a_i}{b_i} = \mu_k + \varepsilon_{ki}, \qquad \varepsilon_{ki} : N(0, c^2)}$$

Parameter estimation

$$\operatorname{arsinh}\frac{y_{ki} - a_i}{b_i} = \mu_k + \varepsilon_{ki}, \qquad \varepsilon_{ki} \; : \; N(0, c^2)$$

- **maximum likelihood estimator: straightforward – but sensitive to deviations from normality**

Parameter estimation

$$\operatorname{arsinh}\frac{y_{ki} - a_i}{b_i} = \mu_k + \varepsilon_{ki}, \qquad \varepsilon_{ki} : N(0, c^2)$$

- maximum likelihood estimator: straightforward – but sensitive to deviations from normality

- model holds for genes that are unchanged; differentially transcribed genes act as outliers.

Parameter estimation

$$\operatorname{arsinh}\frac{y_{ki} - a_i}{b_i} = \mu_k + \varepsilon_{ki}, \qquad \varepsilon_{ki} : N(0, c^2)$$

o maximum likelihood estimator: straightforward – but sensitive to deviations from normality

o model holds for genes that are unchanged; differentially transcribed genes act as outliers.

o robust variant of ML estimator, à la Least Trimmed Sum of Squares regression.

Parameter estimation

$$\operatorname{arsinh}\frac{y_{ki} - a_i}{b_i} = \mu_k + \varepsilon_{ki}, \qquad \varepsilon_{ki} \ : \ N(0, c^2)$$

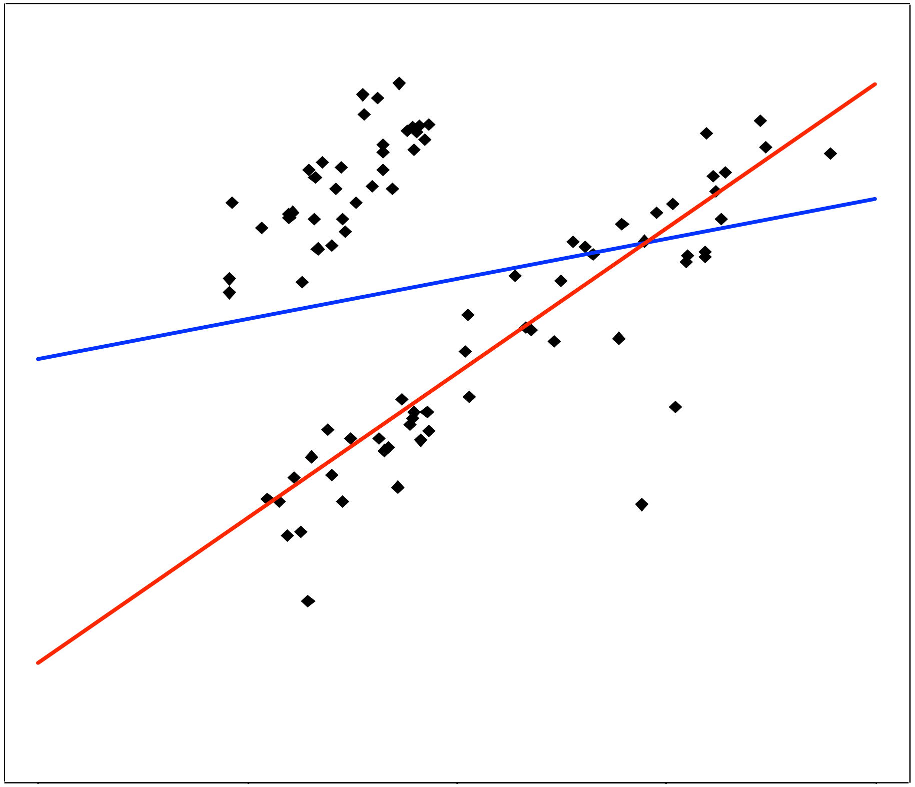
o maximum likelihood estimator: straightforward – but sensitive to deviations from normality

o model holds for genes that are unchanged; differentially transcribed genes act as outliers.

o robust variant of ML estimator, à la Least Trimmed Sum of Squares regression.

o works well as long as <50% of genes are differentially transcribed (and may still work otherwise)

Least trimmed sum of squares regression



minimize

$$\sum_{i=1}^{\frac{n}{2}} \left(y_{(i)} - f\left(x_{(i)}\right)\right)^2$$

P. Rousseeuw, 1980s

- least sum of squares
- least trimmed sum of squares

“usual” log-ratio

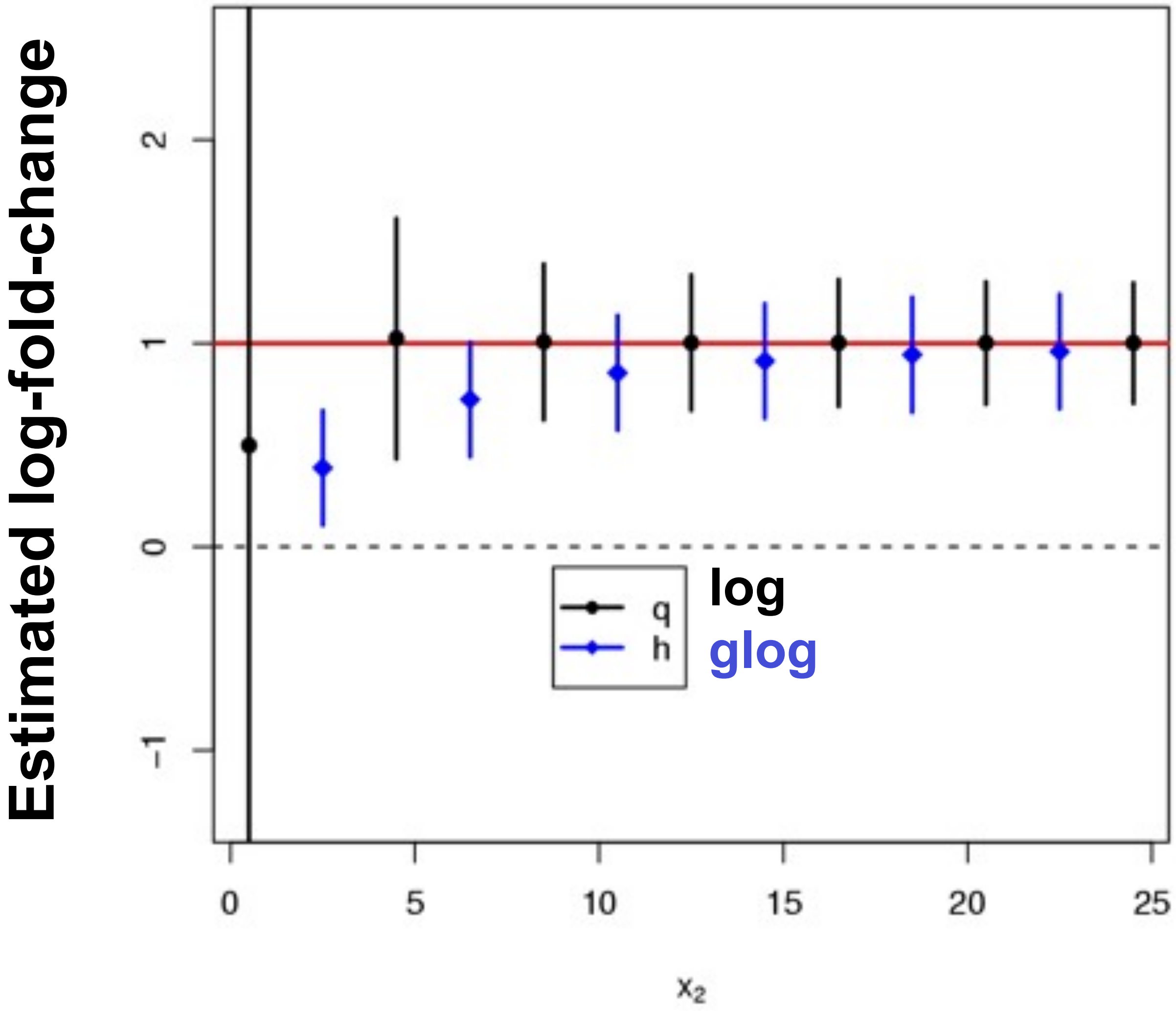
$$\log \frac{x_1}{x_2}$$

'glog' (generalized log-ratio)

$$\log \frac{x_1 + \sqrt{x_1^2 + c_1^2}}{x_2 + \sqrt{x_2^2 + c_2^2}}$$

c_1, c_2 are experiment specific parameters (~level of background noise)

Variance Bias Trade-Off



Signal intensity

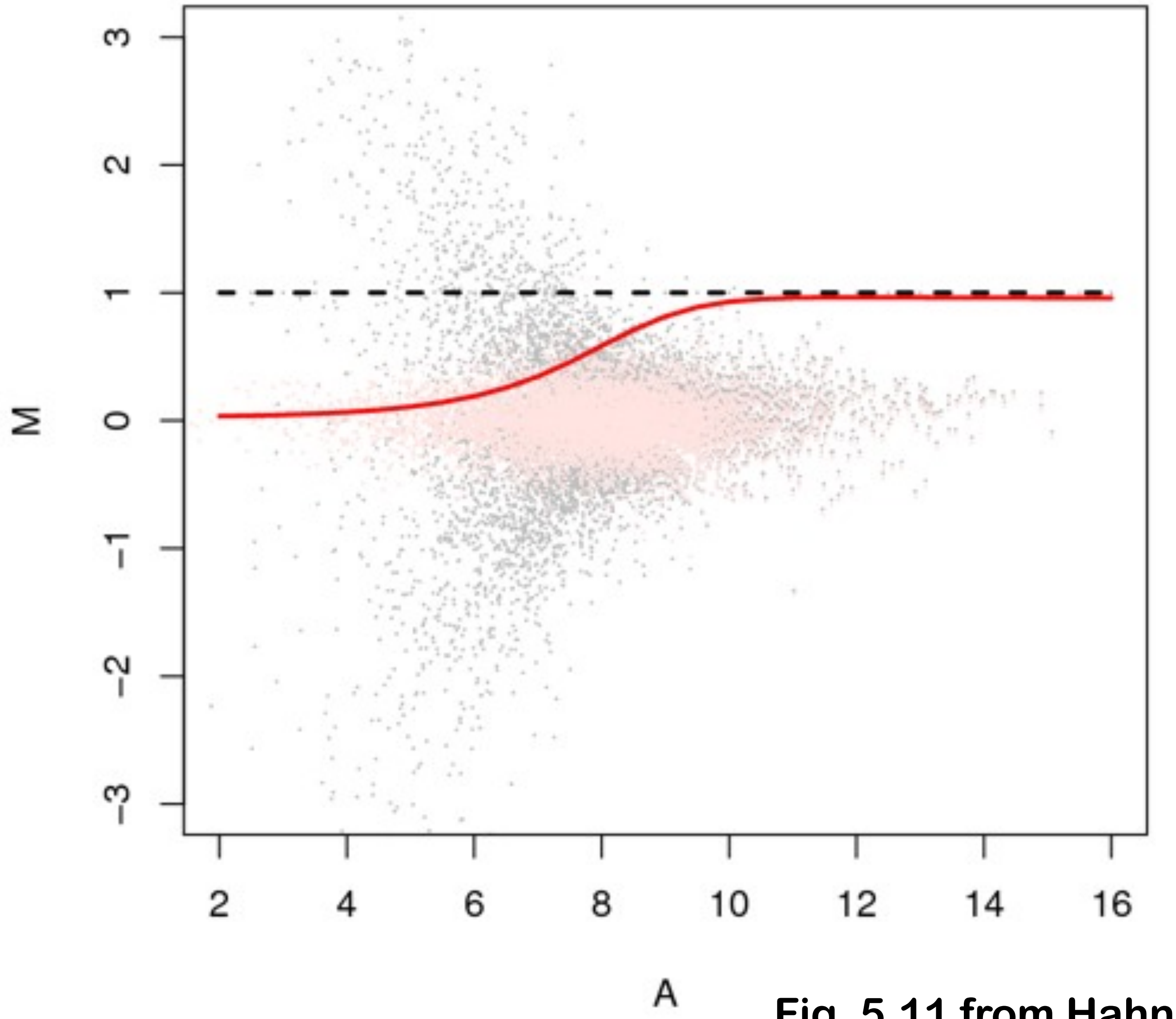
▶ Variance-bias trade-off and shrinkage estimators

Shrinkage estimators:
a general technology in statistics:
pay a small price in bias for a large decrease of variance, so overall the mean-squared-error (MSE) is reduced.

Particularly useful if you have few replicates.

Generalized log-ratio is a shrinkage estimator for log fold change

▶ Variance-bias trade-off and shrinkage estimators



Same-same comparison

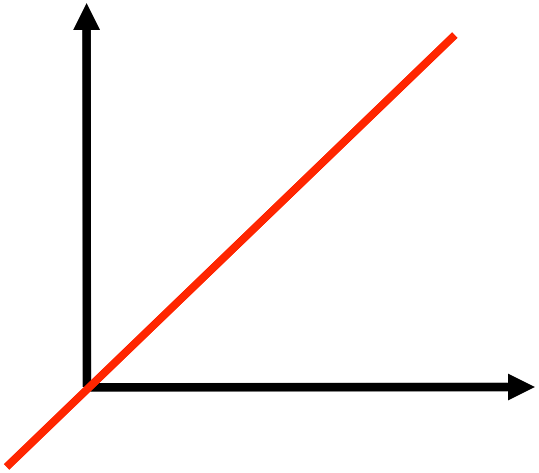
log-ratio

glog-ratio

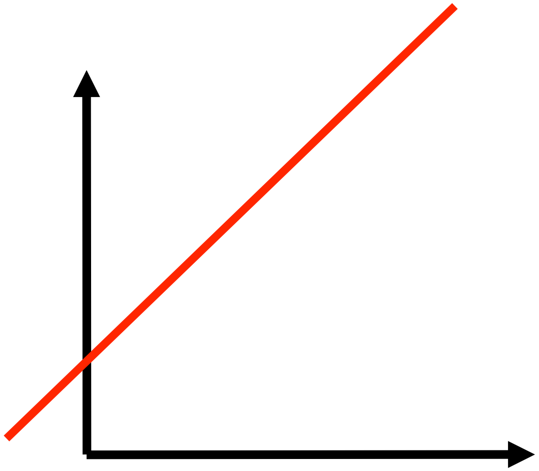
Lines: 29 data points with *observed* ratio of 2

Fig. 5.11 from Hahne et al. (useR book)

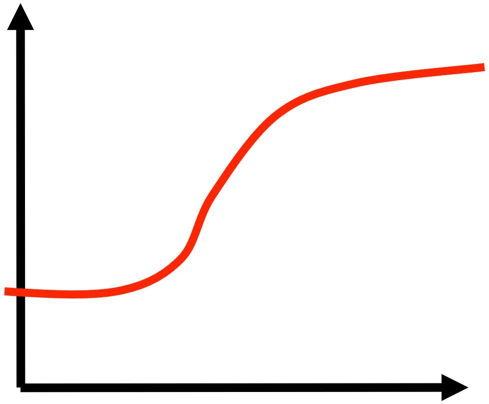
▶ Linear and Non-linear



linear



affine linear



“genuinely”
non-linear

Always affine?

vsn provides a combination of glog-transformation and affine between-array[*] normalisation

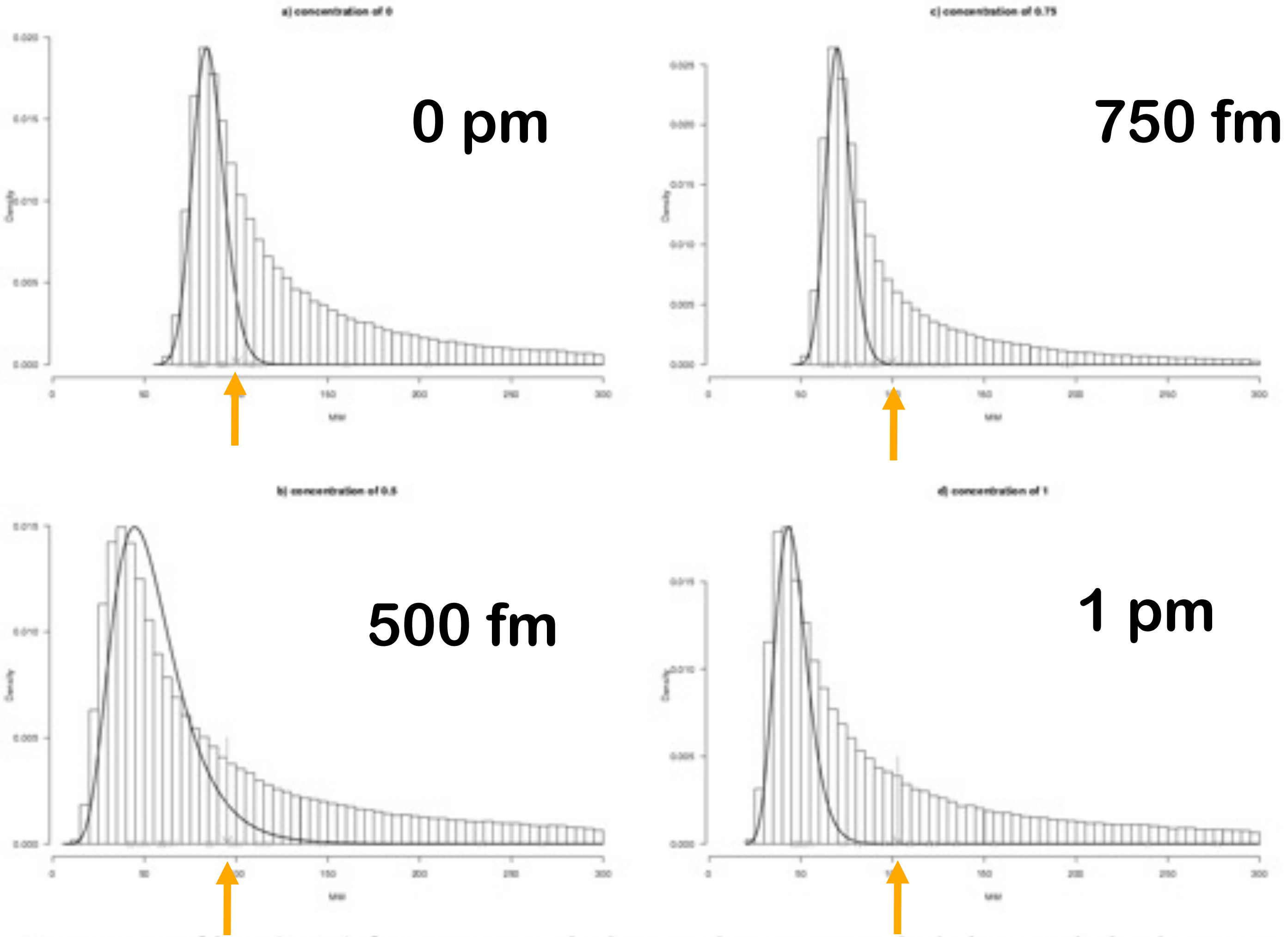
What if you want to normalise for genuine non-linear effects, and still use the transformation?

Set parameter `calib` in `vsn2` function to `none` (default: `affine`) and do your own normalisation beforehand (do not (log-)transform). The vignette shows an example for use with quantile normalisation.

[*] print-tip groups or other stratifications are also possible

Background

Background correction



0 pm

750 fm

500 fm

1 pm

Fig. 5. Histograms of $\log_2(MM)$ for a array in which no probe-set was spiked along with the three arrays in which BioB-5 was spiked-in at concentrations of 0.5, 0.75, and 1 pM. The observed PM values for the 20 probes associated with BioB-5 are marked with crosses and the average with an arrow. The black curve represents the log normal distribution obtained from left-of-the-mode data.

Irizarry et al.
Biostatistics
2003

RMA Background correction

$PM = B + S$

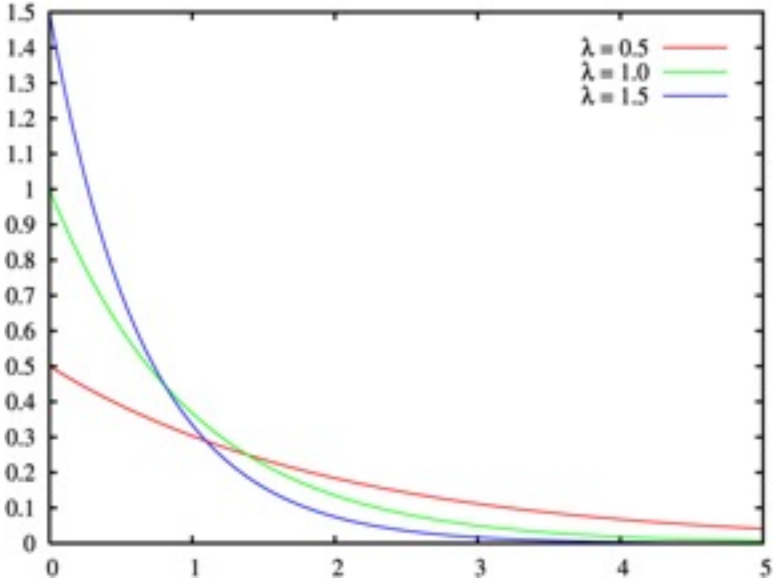
$B \sim$ log-normal with mean and sd read off MM values

$S \sim$ exponential

$\Rightarrow$ closed form expression for $E[S \mid PM]$,

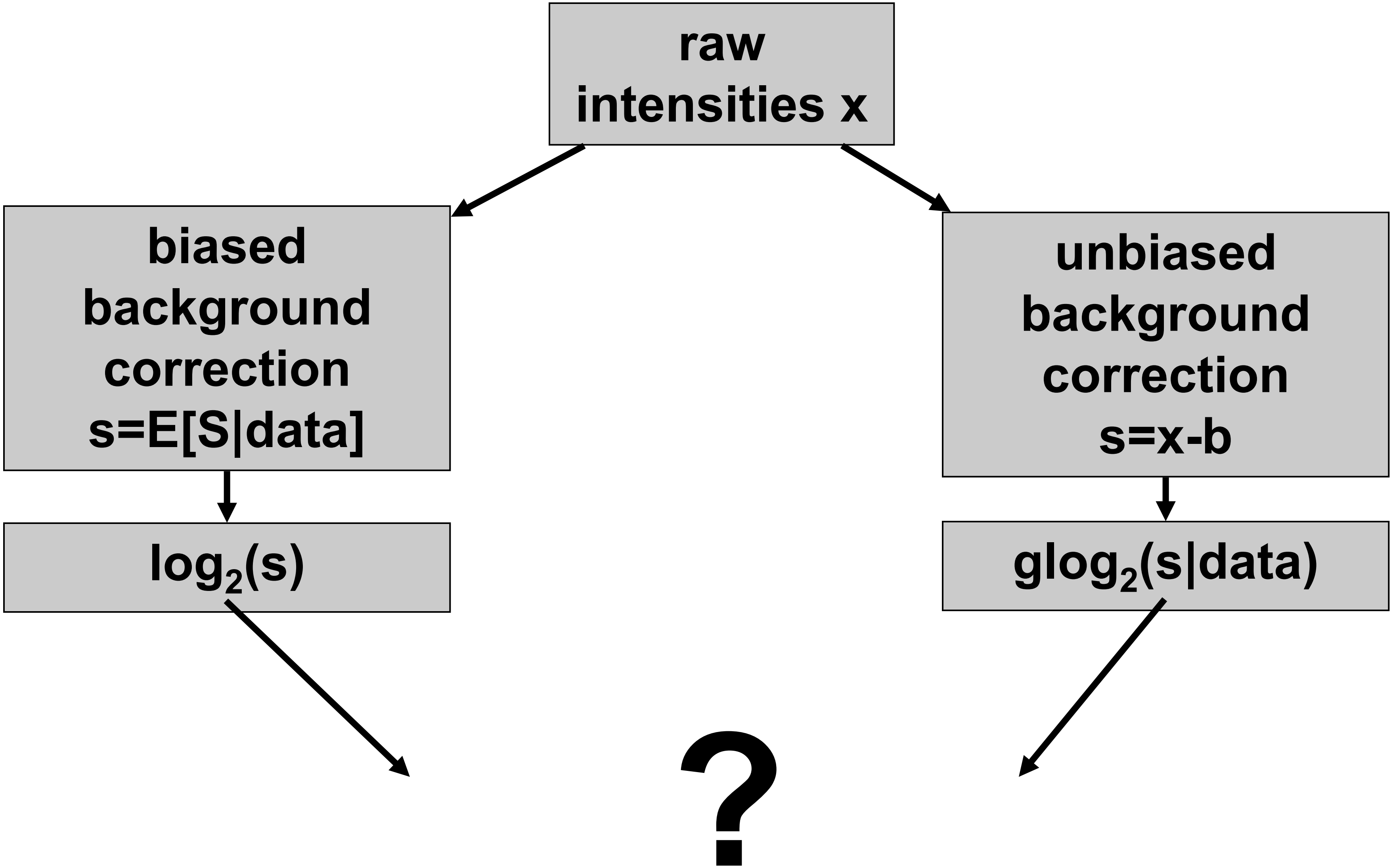
use this as $\hat{s} \quad (> 0)$.

(NB, $P[S > 0] = 1$ is not realistic)



Irizarry et al. (2002)

Background correction:



raw intensities x

biased background correction $s=E[S|\text{data}]$

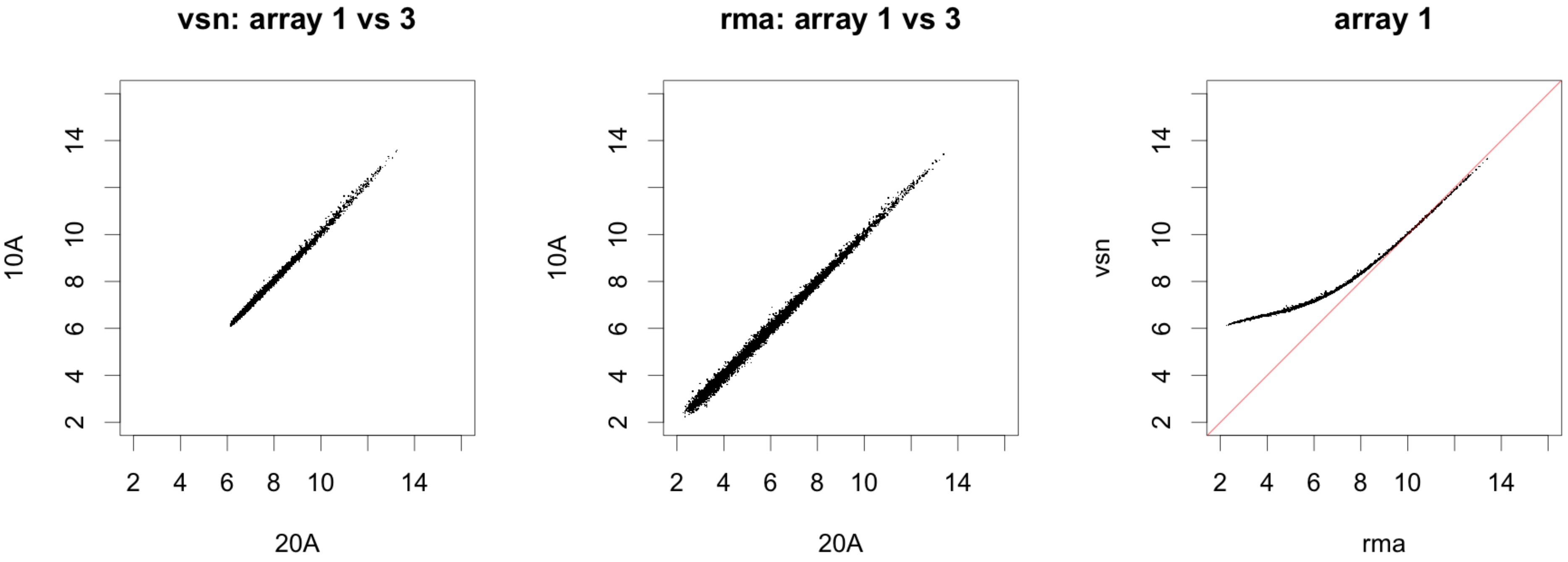
unbiased background correction $s=x-b$

$\log_2(s)$

$\text{glog}_2(s|\text{data})$

?

Comparison between RMA and VSN background correction



vsn package vignette

Summaries for Affymetrix genechip probe sets

Data and notation

PM_{ikg} , MM_{ikg} = Intensities for perfect match and mismatch probe k for gene g on chip i

$i = 1,\ldots, n$ one to hundreds of chips

$k = 1,\ldots, J$ usually 11 probe pairs

$g = 1,\ldots, G$ tens of thousands of probe sets.

Tasks:

calibrate (normalize) the measurements from different chips (samples)

summarize for each probe set the probe level data, i.e., 11 PM and MM pairs, into a single expression measure.

compare between chips (samples) for detecting differential expression.

Expression measures: MAS 4.0

Affymetrix GeneChip MAS 4.0 software used AvDiff, a trimmed mean:

$$AvDiff = \frac{1}{\# K} \sum_{k \in K} (PM_k - MM_k)$$

- sort $d_k = PM_k - MM_k$
- exclude highest and lowest value
- K := those pairs within 3 standard deviations of the average

Expression measures
MAS 5.0

Instead of MM, use "repaired" version CT

CT = MM if *MM<PM*

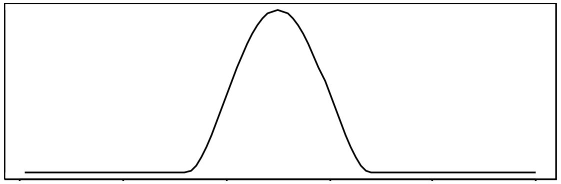
= PM / "typical log-ratio" if *MM>=PM*

Signal = Weighted mean of the values log(PM-CT)

weights follow Tukey Biweight function

(location = data median,

scale a fixed multiple of MAD)



Expression measures: Li & Wong

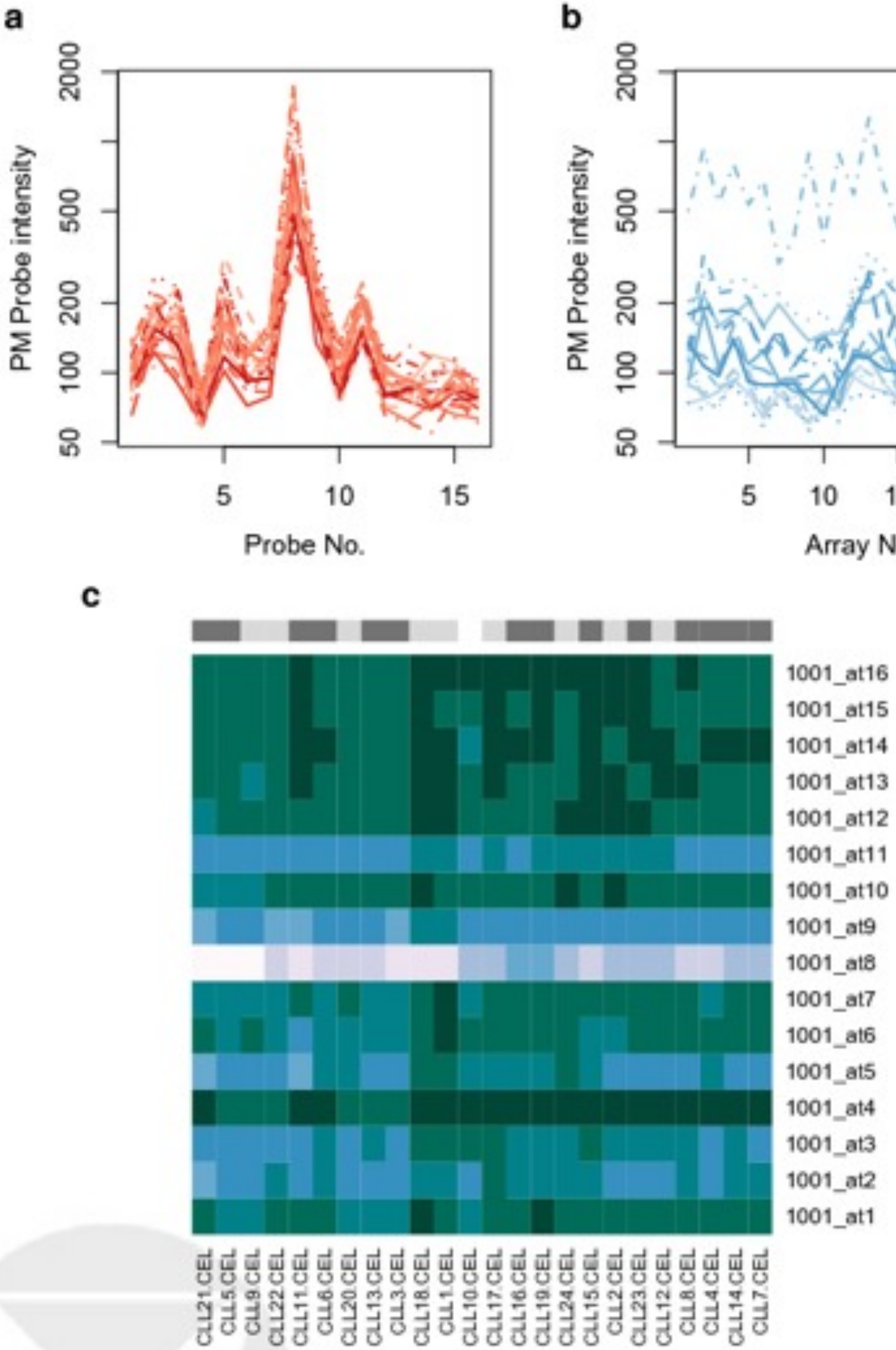
dChip **fits a model for each gene**

$$PM_{ki} - MM_{ki} = \theta_k \phi_i + \varepsilon_{ki}, \qquad \varepsilon_{ki} \propto N(0, \sigma^2)$$

where

- ϕ_i **: expression measure for the gene in sample *i***
- θ_k **: probe effect**

ϕ_i **is estimated by maximum likelihood**


a
2000
500
200
100
50
PM Probe intensity
5
10
15
Probe No.
b
2000
500
200
100
50
PM Probe intensity
5
10
15
20
Array No.
c
1001_at16
1001_at15
1001_at14
1001_at13
1001_at12
1001_at11
1001_at10
1001_at9
1001_at8
1001_at7
1001_at6
1001_at5
1001_at4
1001_at3
1001_at2
1001_at1
CLL21.CEL
CLL5.CEL
CLL9.CEL
CLL22.CEL
CLL11.CEL
CLL6.CEL
CLL20.CEL
CLL13.CEL
CLL3.CEL
CLL18.CEL
CLL1.CEL
CLL10.CEL
CLL17.CEL
CLL16.CEL
CLL19.CEL
CLL24.CEL
CLL15.CEL
CLL2.CEL
CLL23.CEL
CLL12.CEL
CLL8.CEL
CLL4.CEL
CLL14.CEL
CLL7.CEL

Expression measures
RMA: Irizarry et al. (2002)

dChip

$$Y_{ki} = \theta_k \phi_i + \varepsilon_{ki}, \qquad \varepsilon_{ki} \propto N(0,\sigma^2)$$

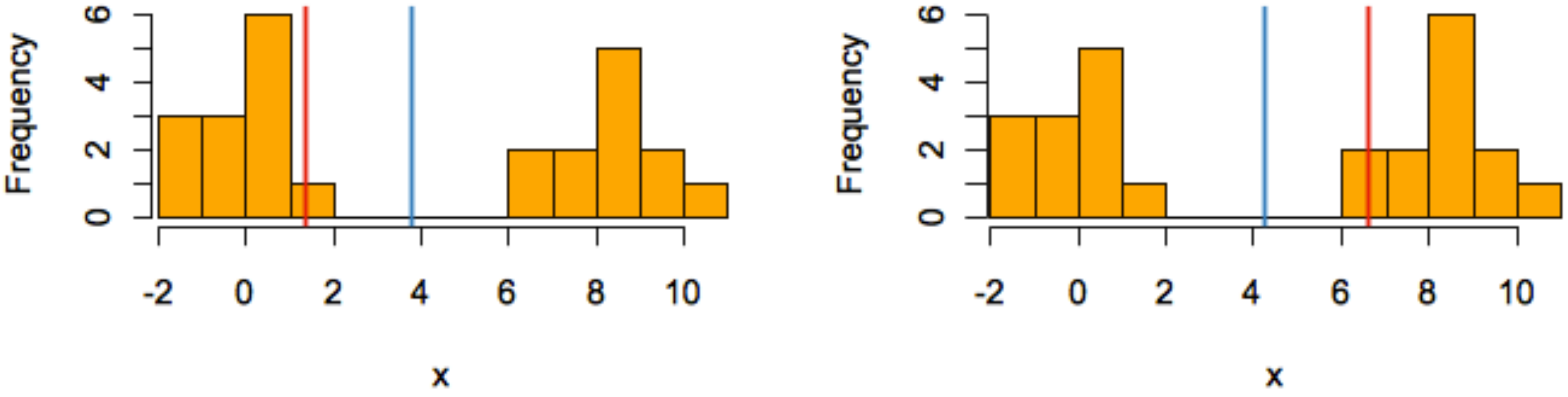
RMA

$$\log_2 Y_{ki} = a_k + b_i + \varepsilon_{ki}$$

***b*$_i$ is estimated using the robust method median polish (successively remove row and column medians, accumulate terms, until convergence).**

However, median (and hence median polish) is not always so robust...

- median
- trimmed mean (0.15)

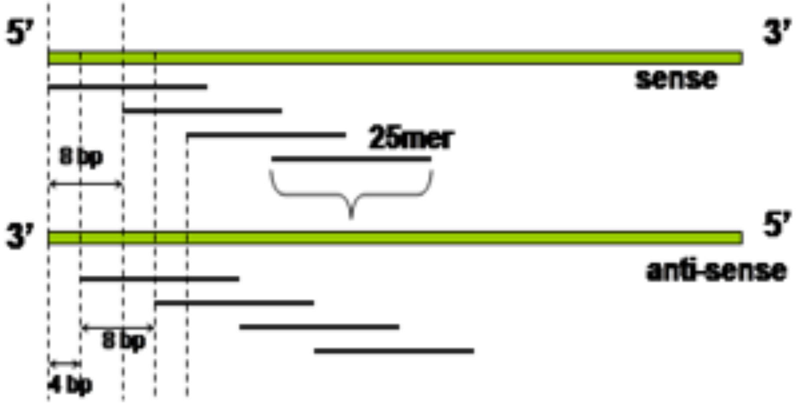


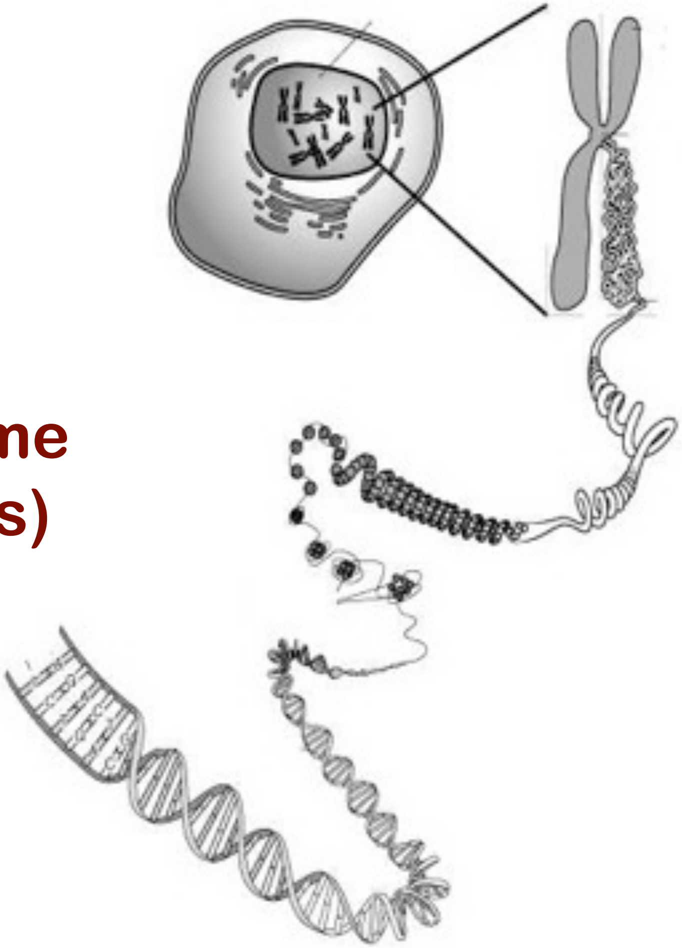
See also: Casneuf T. et al. (2007), In situ analysis of cross-hybridisation on microarrays and the inference of expression correlation. BMC Bioinformatics 2007;8(1): 461

Probe effect adjustment by using gDNA reference

Huber et al., Bioinformatics 2006

Genechip *S. cerevisiae* Tiling Array

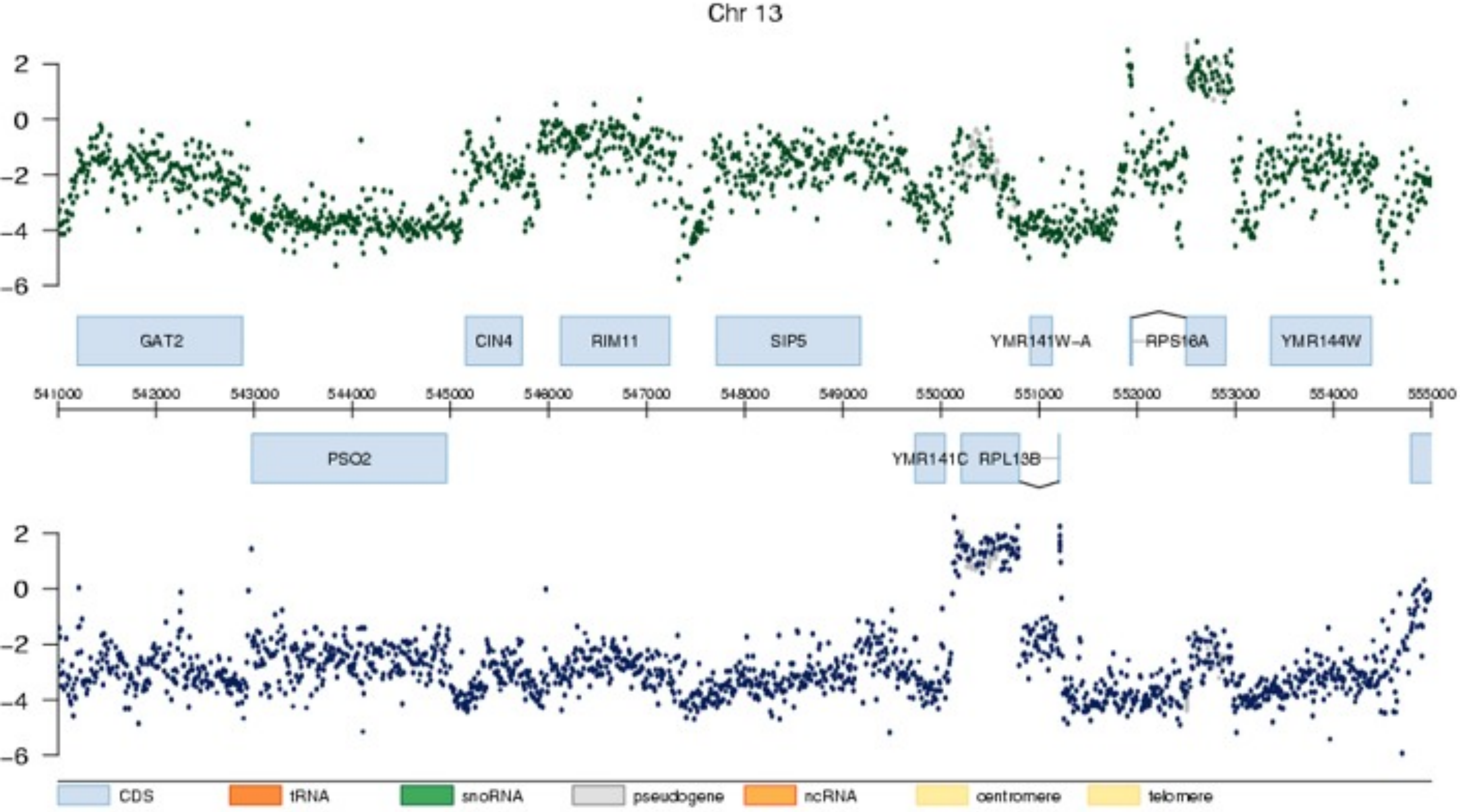




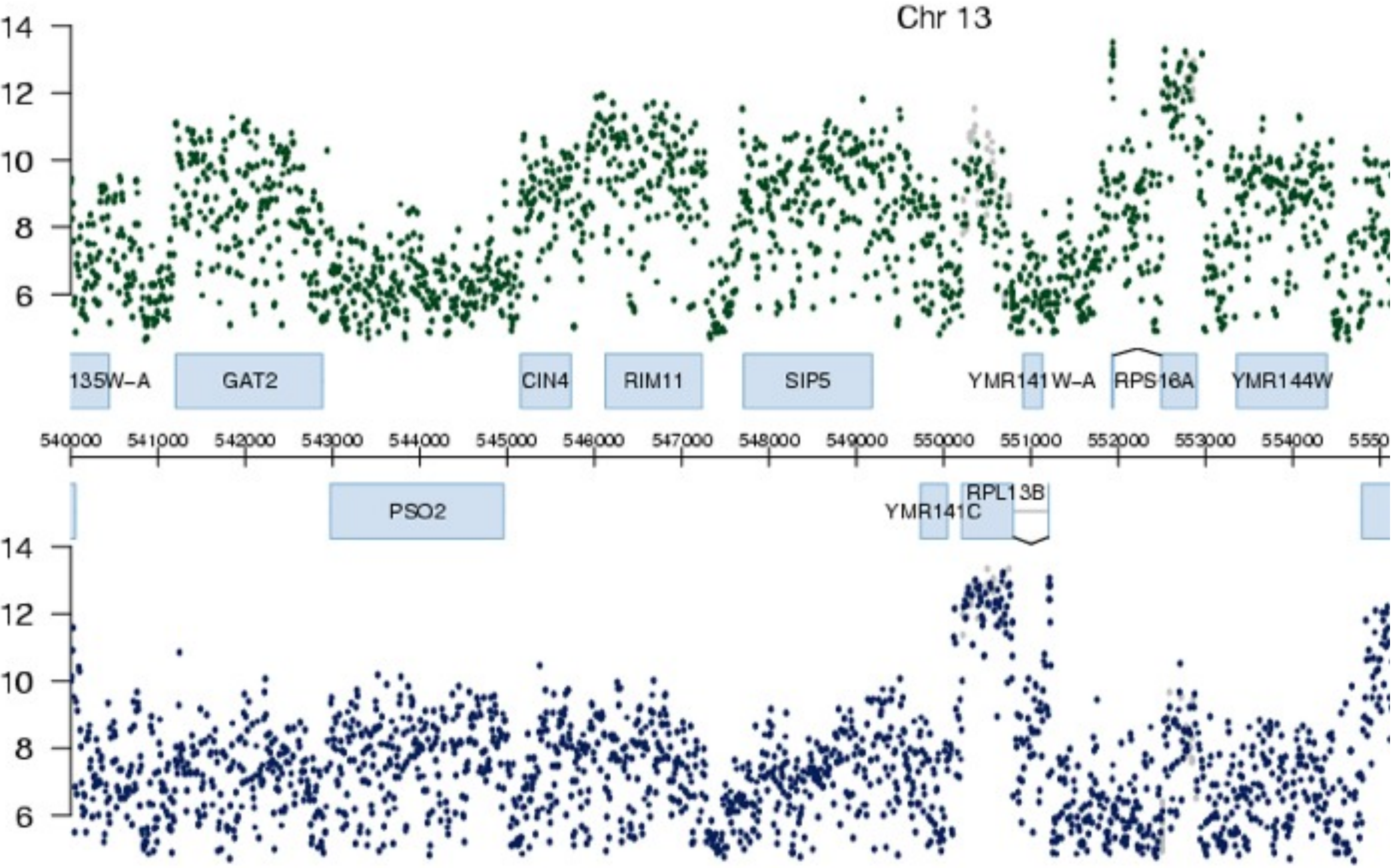
4 bp tiling path over complete genome
(12 M basepairs, 16 chromosomes)
Sense and Antisense strands
6.5 Mio oligonucleotides
5 μm feature size

manufactured by Affymetrix
designed by Lars Steinmetz (EMBL & Stanford Genome Center)

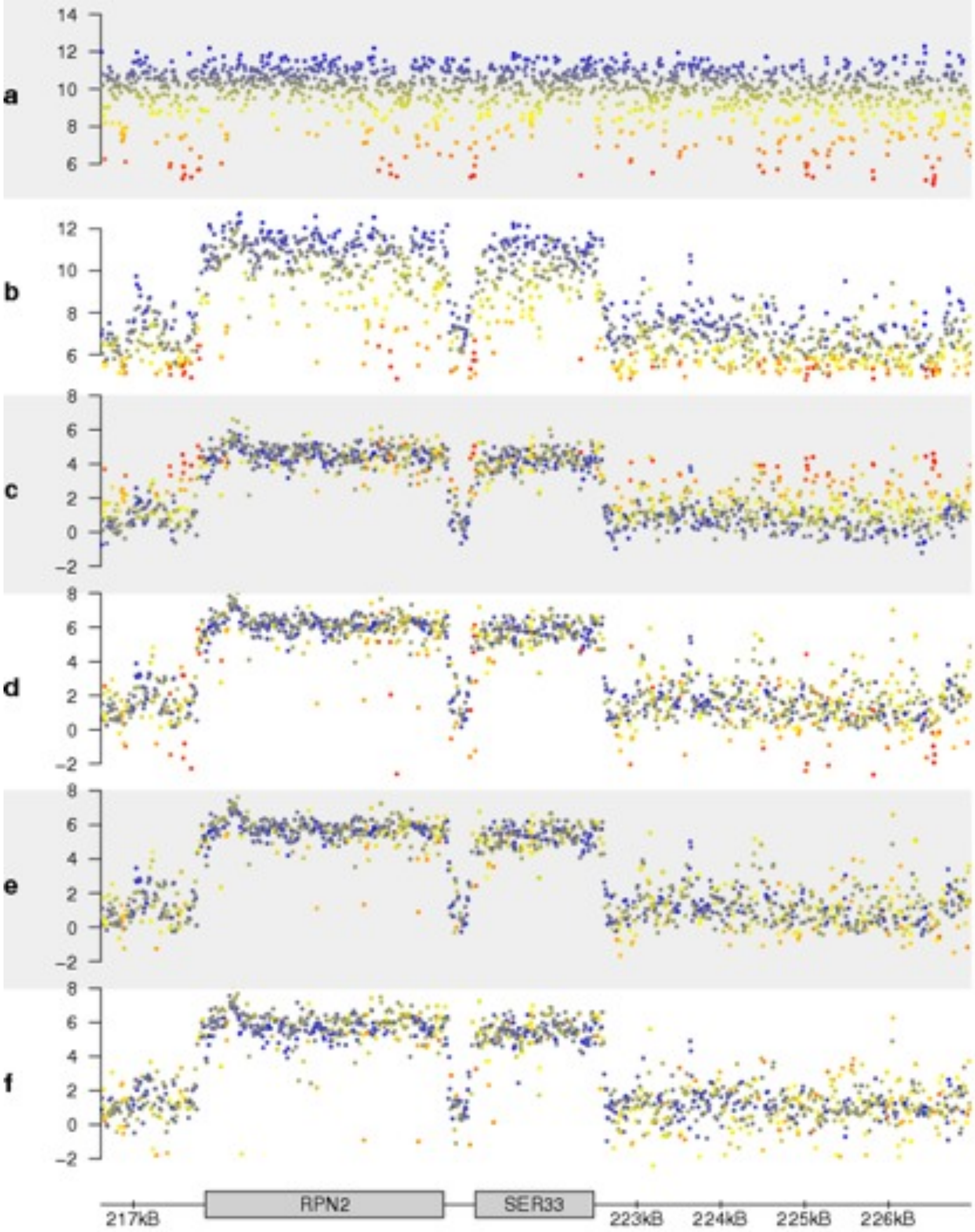
RNA Hybridization



Before normalization



Probe specific response normalization



S/N

$$\log_2 s_i$$

$$\log_2 y_i$$

3.22

$$q_i = \log_2 \frac{y_i}{s_i}$$

3.47

$$q_i = \mathrm{glog}_2 \frac{y_i - b(s_i)}{s_i}$$

4.04

remove 'dead' probes

4.58

$$q_i = \mathrm{glog}_2 \frac{PM_i - MM_i}{s_i}$$

4.36

Probe-specific response normalization

$$q_i = \operatorname{glog}_2 \frac{y_i - b(s_i)}{s_i}$$

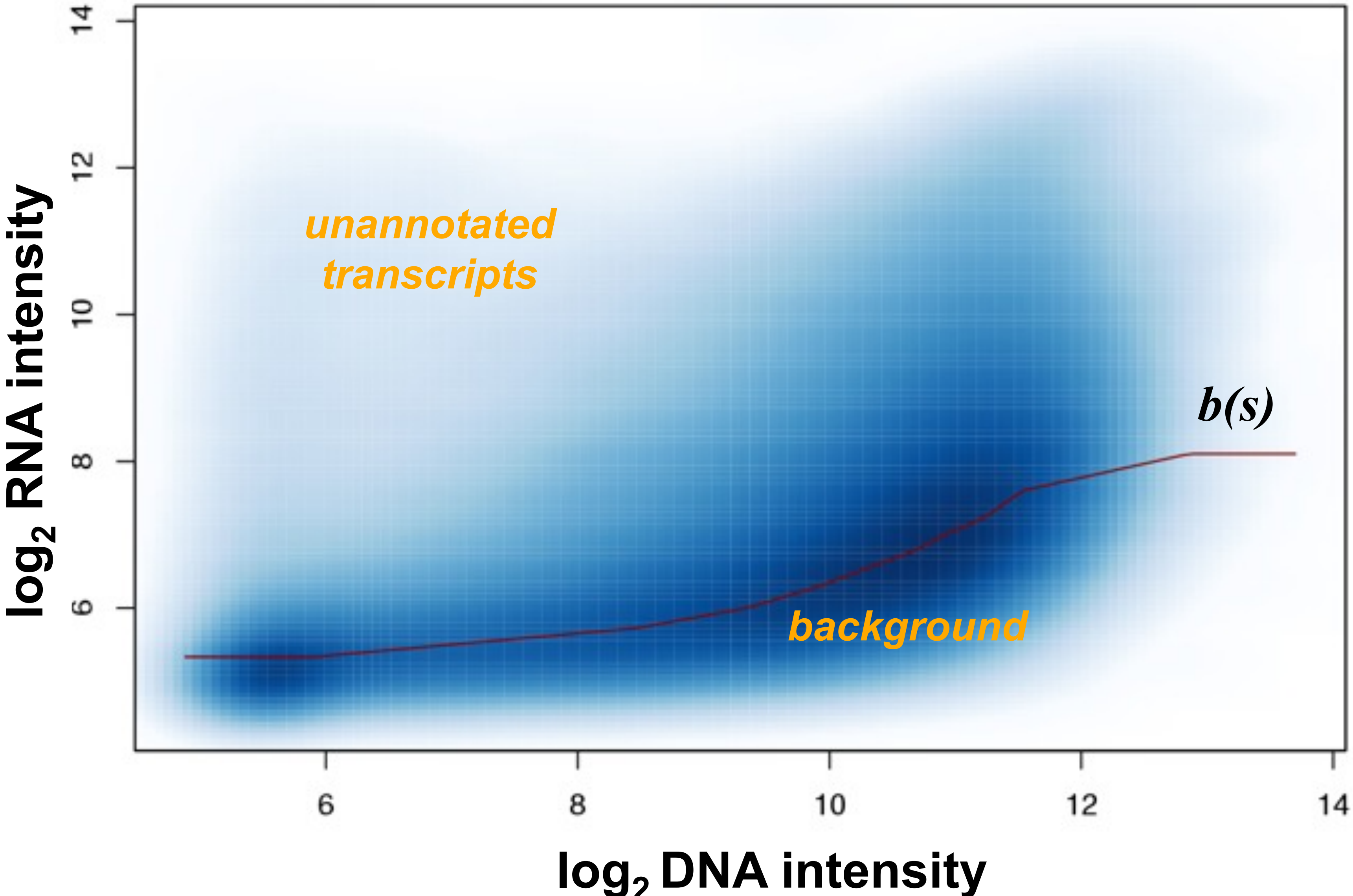
s_i probe specific response factor.

Estimate taken from DNA hybridization data

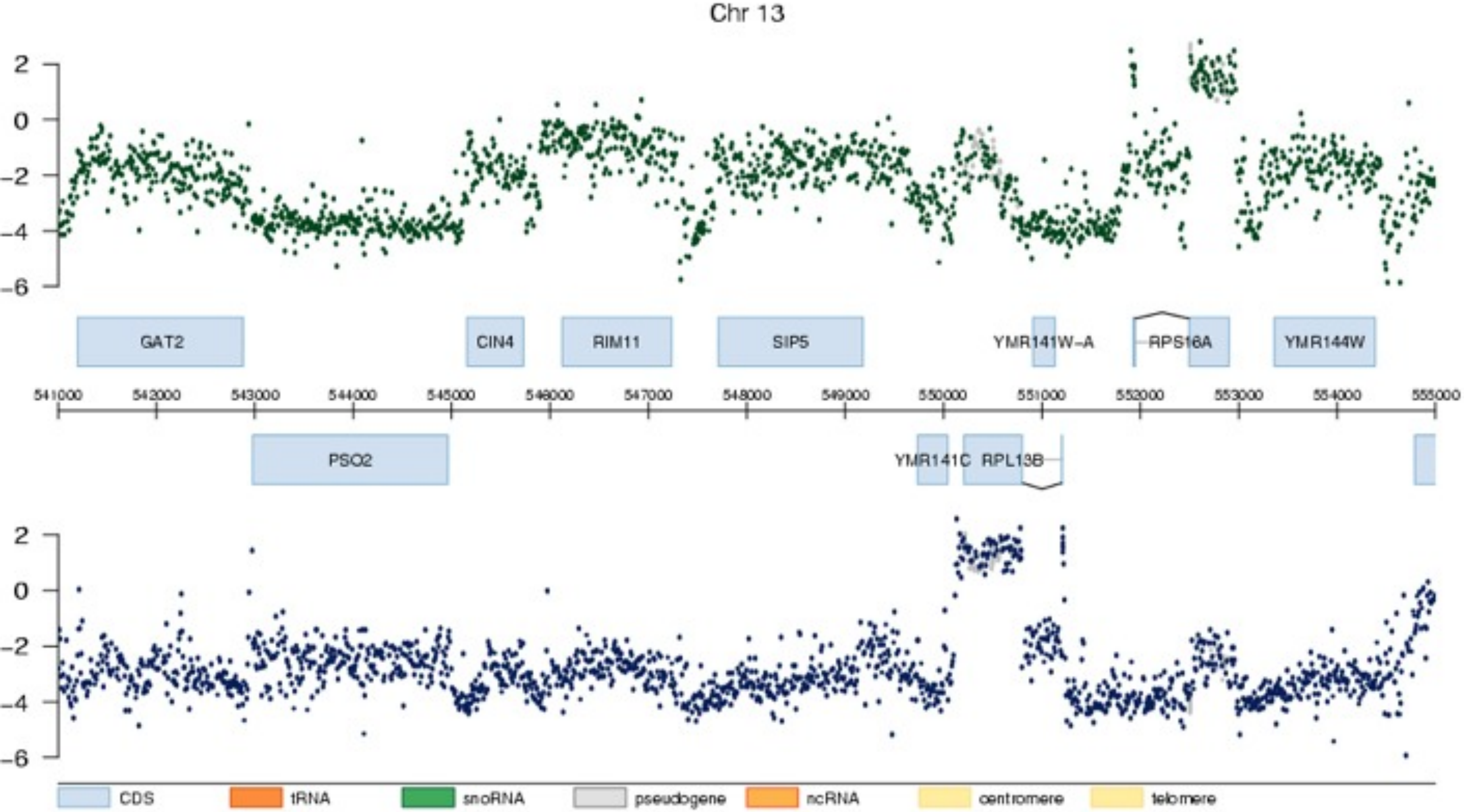
$b_i = b(s_i)$ probe specific background term.

Estimation: for strata of probes with similar *s_i*, estimate *b* through location estimator of distribution of intergenic probes, then interpolate to obtain continuous *b(s)*

Estimation of *b*: joint distribution of (DNA, RNA) values of intergenic PM probes



After normalization



Quality assessment

▶ References

Bioinformatics and computational biology solutions using R and Bioconductor, R. Gentleman, V. Carey, W. Huber, R. Irizarry, S. Dudoit, Springer (2005).

Variance stabilization applied to microarray data calibration and to the quantification of differential expression. W. Huber, A. von Heydebreck, H. Sültmann, A. Poustka, M. Vingron. Bioinformatics 18 suppl. 1 (2002), S96-S104.

Exploration, Normalization, and Summaries of High Density Oligonucleotide Array Probe Level Data. R. Irizarry, B. Hobbs, F. Collins, …, T. Speed. Biostatistics 4 (2003) 249-264.

Error models for microarray intensities. W. Huber, A. von Heydebreck, and M. Vingron. Encyclopedia of Genomics, Proteomics and Bioinformatics. John Wiley & sons (2005).

Normalization and analysis of DNA microarray data by self-consistency and local regression. T.B. Kepler, L. Crosby, K. Morgan. Genome Biology. 3(7):research0037 (2002)

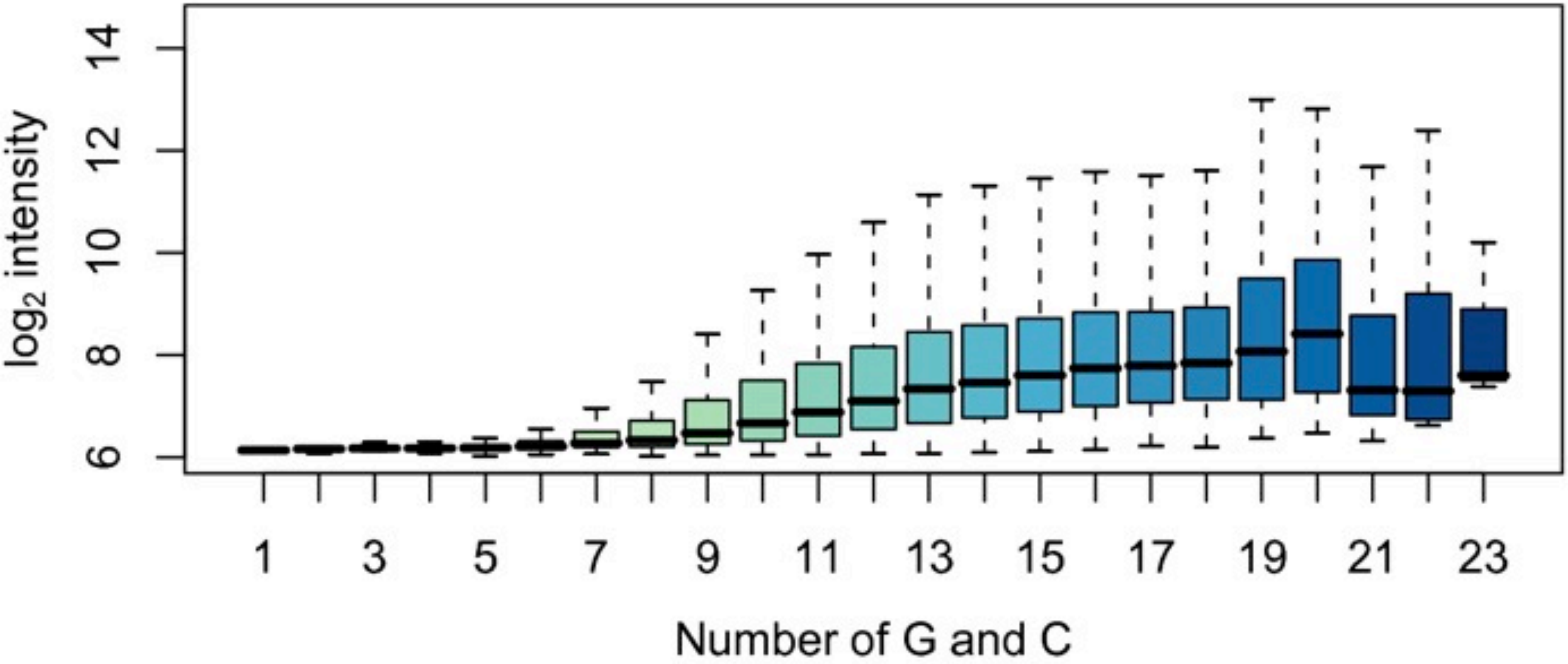
Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiments. S. Dudoit, Y.H. Yang, M. J. Callow, T. P. Speed. Technical report # 578, August 2000 (UC Berkeley Dep. Statistics)

A Benchmark for Affymetrix GeneChip Expression Measures. L.M. Cope, R.A. Irizarry, H. A. Jaffee, Z. Wu, T.P. Speed. Bioinformatics (2003).

….many, many more...

Acknowledgements

Anja von Heydebreck (Merck, Darmstadt)
Robert Gentleman (Genentech, San Francisco)
Günther Sawitzki (Uni Heidelberg)
Martin Vingron (MPI, Berlin)
Rafael Irizarry (JHU, Baltimore)
Terry Speed (UC Berkeley)
Judith Boer (Uni Leiden)
Anke Schroth (Wiesloch)
Friederike Wilmer (Qiagen Hilden)
Jörn Tödling (Inst. Curie, Paris)
Lars Steinmetz (EMBL Heidelberg)
Audrey Kauffmann (Bergonié, Bordeaux)


log2 intensity
6
8
10
12
14
1
3
5
7
9
11
13
15
17
19
21
23
Number of G and C

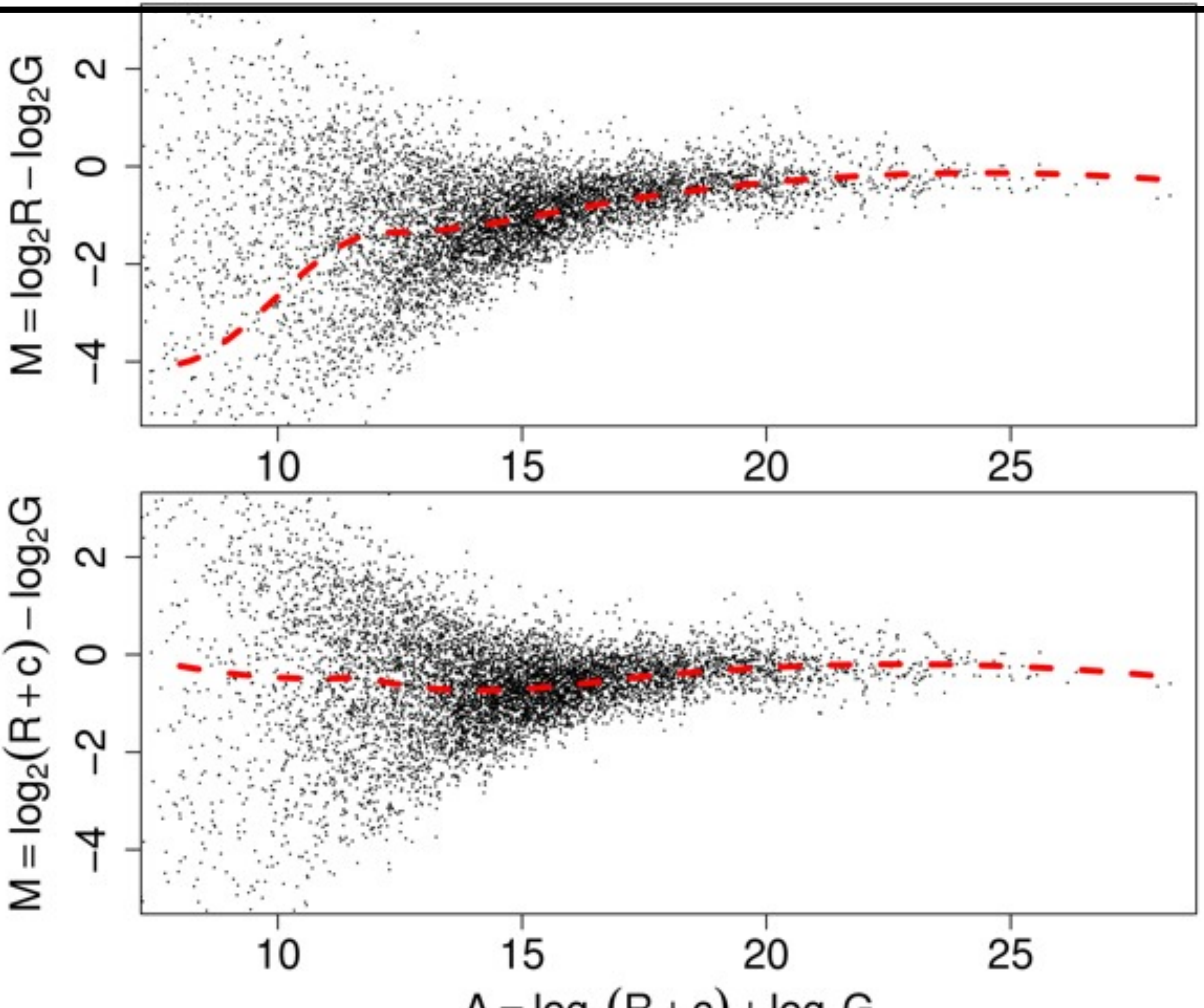
▶ What about non-linear effects

o Microarrays can be operated in a linear regime, where fluorescence intensity increases proportionally to target abundance (see e.g. Affymetrix dilution series)

Two reasons for non-linearity:

o At the high intensity end: saturation/quenching. This can (and should) be avoided experimentally - loss of data!

o At the low intensity end: background offsets, instead of $y=k \cdot x$ we have $y=k \cdot x+x_0$, and in the log-log plot this can look curvilinear. But this is an affine-linear effect and can be correct by affine normalization. Local polynomial regression may be OK, but tends to be less efficient.


M = log₂R − log₂G
2
0
−2
−4
10
15
20
25
M = log₂(R + c) − log₂G
2
0
−2
−4
10
15
20
25
A = log₂(R + c) + log₂G