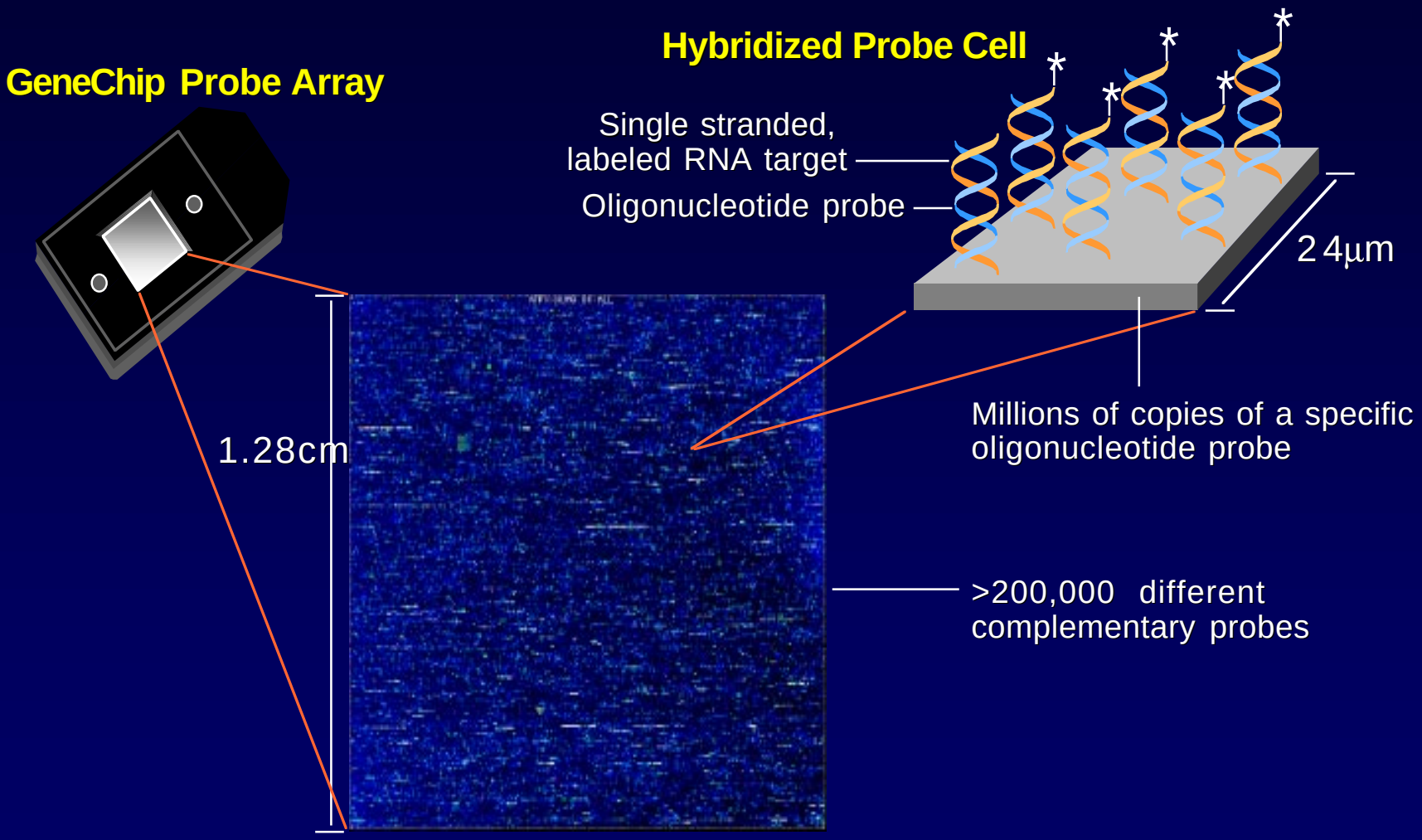
# Introduction to Affymetrix GeneChip data

Stat 246, Spring 2002, Week 16

# Summary

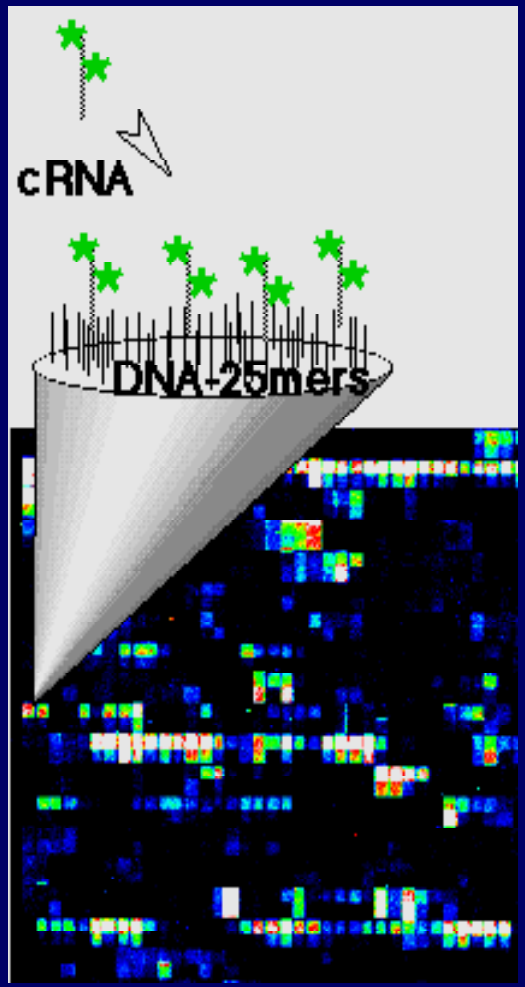
- Review of technology
- Probeset summaries
- What we do: our 4 steps
- Assessing the technology and the different expression measures
- How robustness works

# Probe arrays




Compliments of D. Gerhold

# Image analysis




- About 100 pixels per probe cell
- These intensities are combined to form one number representing expression for the probe cell oligo
- Possibly room for improvement

# GeneChip® Expression Array Design

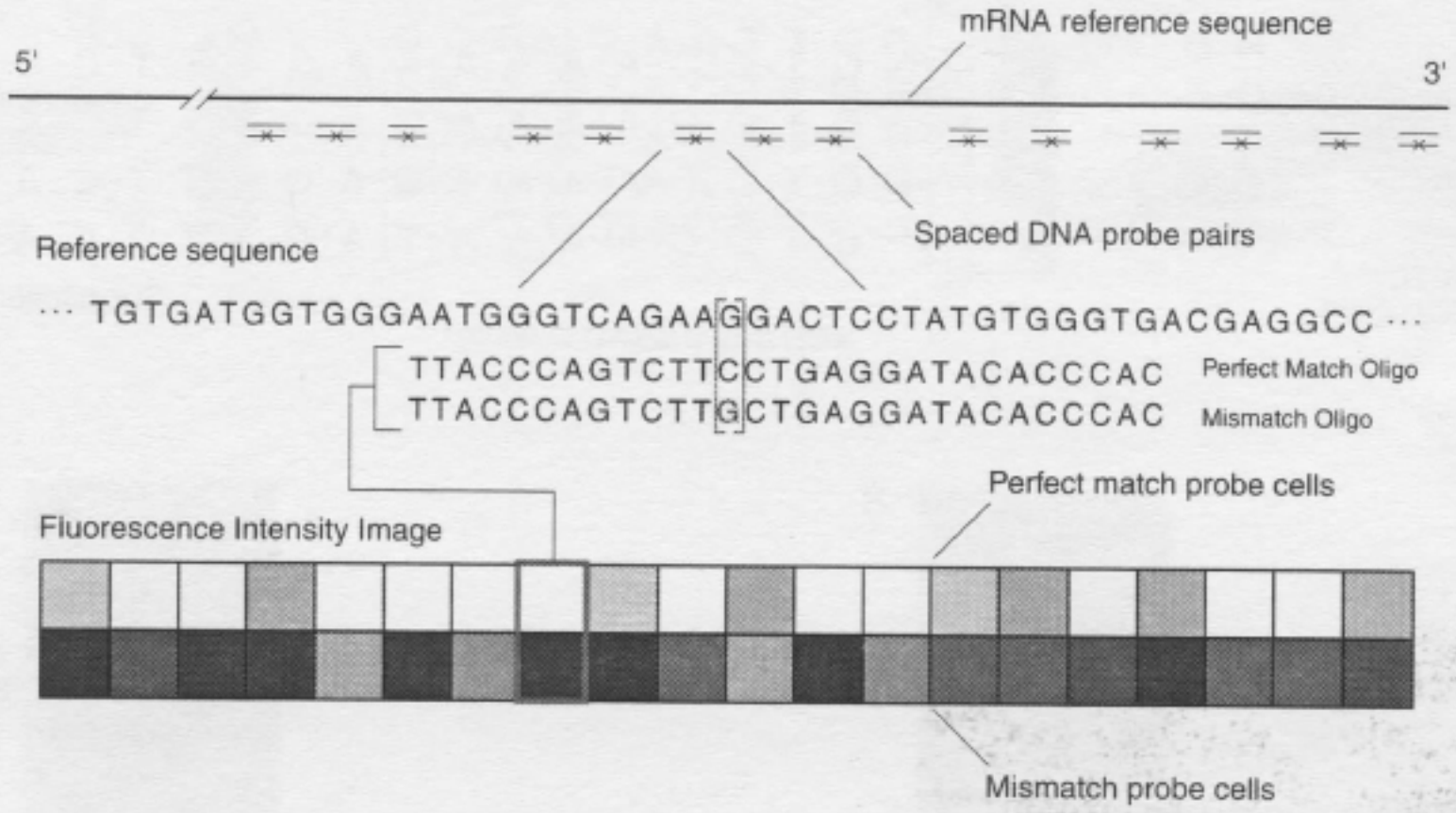


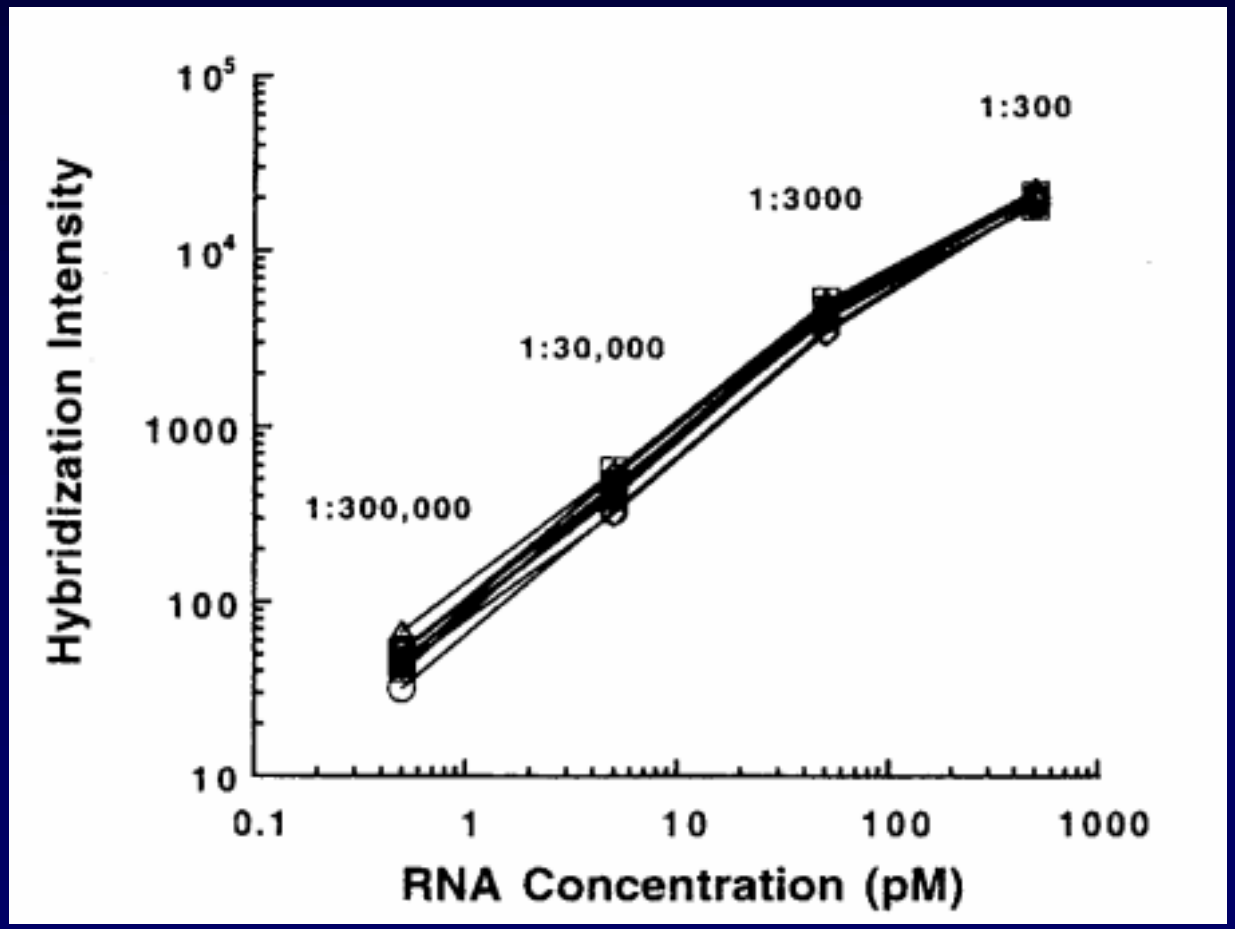

Figure 1-3 Expression tiling strategy

## The big picture

- Summarize 20 PM,MM pairs (probe level data) into one number for each probe set (gene)
- We call this number an expression measure
- Affymetrix GeneChip Software has defaults.
- Does it work? Can it be improved?

# Where is the evidence that it works?

Lockhart et. al. Nature Biotechnology 14 (1996)



## Comments

- The chips used in Lockhart et. al. contained around 1000 probes per gene
- Current chips contain 11-20 probes per gene
- These are quite different situations
- We haven’t seen a plot like the previous one for current chips

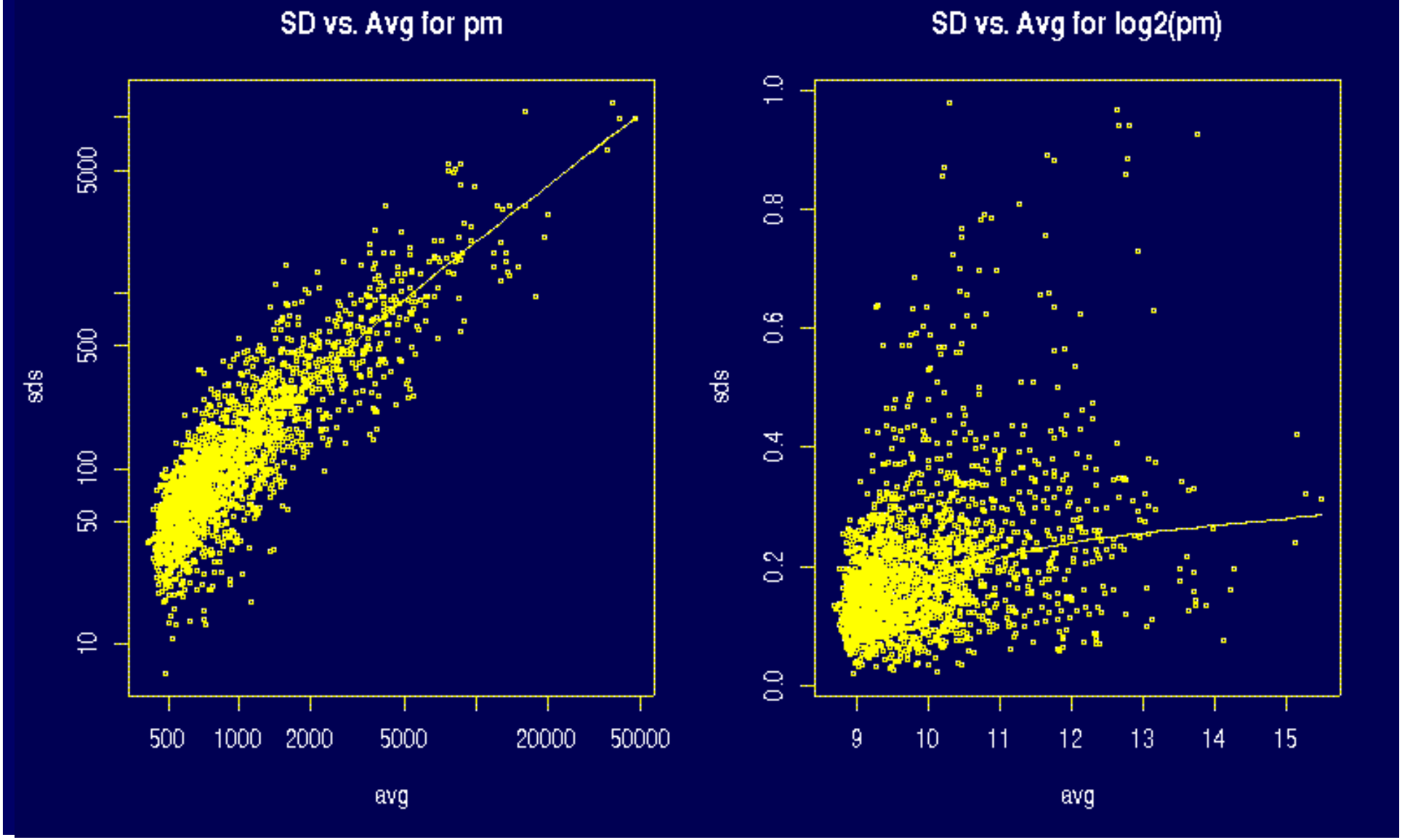
# Some possible problems

What if

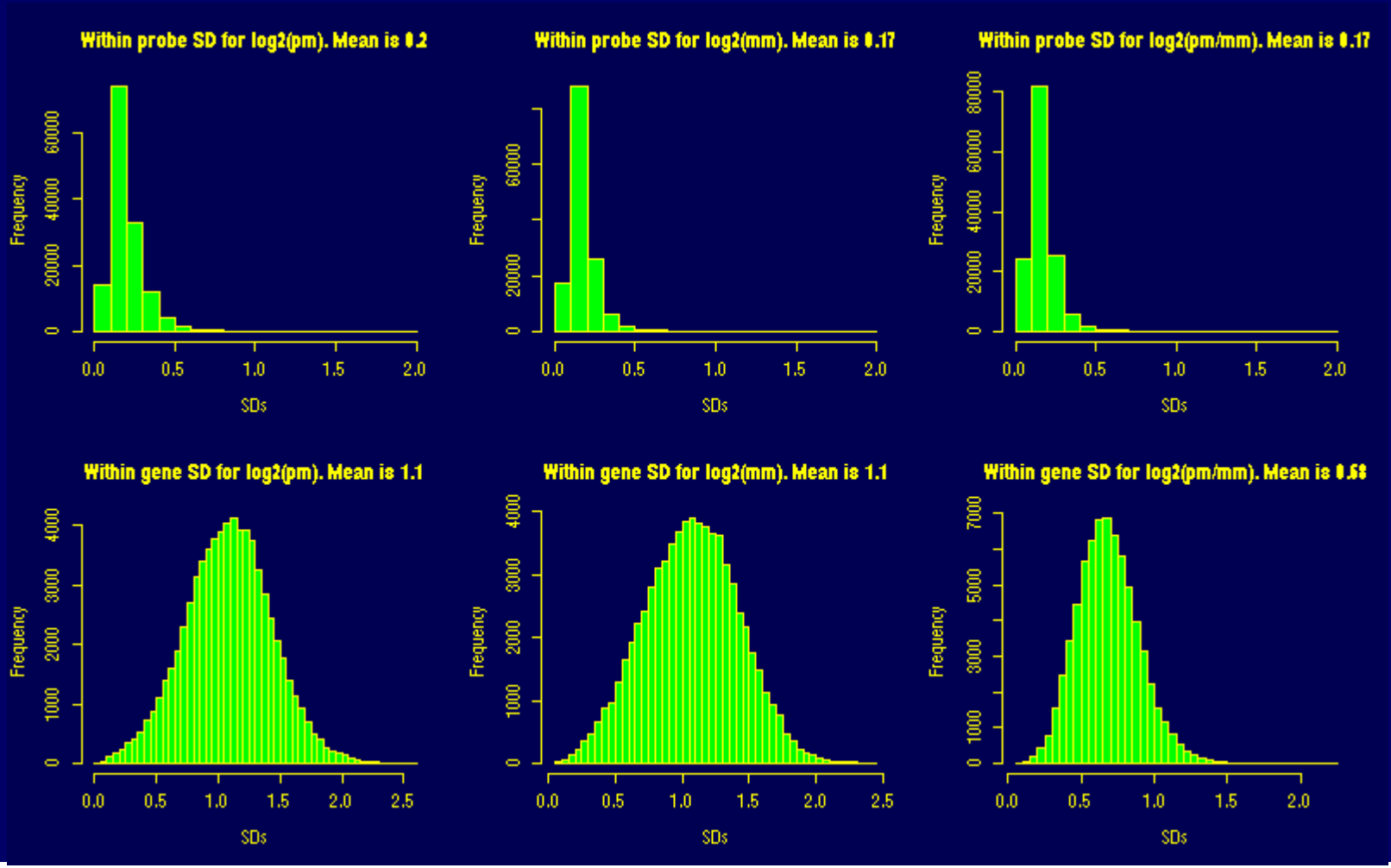
- a small number of the probe pairs hybridize much better than the rest?
- removing the middle base does not make a difference for some probes?
- some MMs are PMs for some other gene?
- there is need for normalization?

We explore these possibilities using a variety of data sets

# SD vs. Avg (across replicate chips)



# ANOVA: Strong probe effect: 5 times bigger than gene effect



# Competing measures of expression

- GeneChip– older software uses *Avg.diff*

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

with A a set of suitable pairs chosen by software. 30%-40-% can be <0.

- *Log $PM_j/MM_j$* was also used.
- For differential expression Avg.diffs are compared between chips.

# Competing measures of expression, 2

- Li and Wong fit a model

$$PM_{ij} - MM_{ij} = \theta_i \phi_j + \varepsilon_{ij},\ \ \varepsilon_{ij} \propto N(0, \sigma^2)$$

  They consider $\theta_i$ to be expression in chip *i*
- Efron *et al* consider *log PM - 0.5 log MM.* It is much less frequently <0.
- Another summary is the second largest PM, $PM_{(2)}$

# Competing measures of expression, 3

- GeneChip – newest version uses something else, namely

$$\text{signal} = \text{TukeyBiweight}\{\log(PM_j - MM^*_j)\}$$

with *MM** a version of MM that is never bigger than PM.

# Competing measures of expression, 4

- Why not stick to what has worked for cDNA?

$$\frac{1}{|\mathrm{A}|}\sum_{j\in A}\log_2(PM_j - BG)$$

Again A is a suitable set of pairs.

Care needed with BG, and we need to robustify.

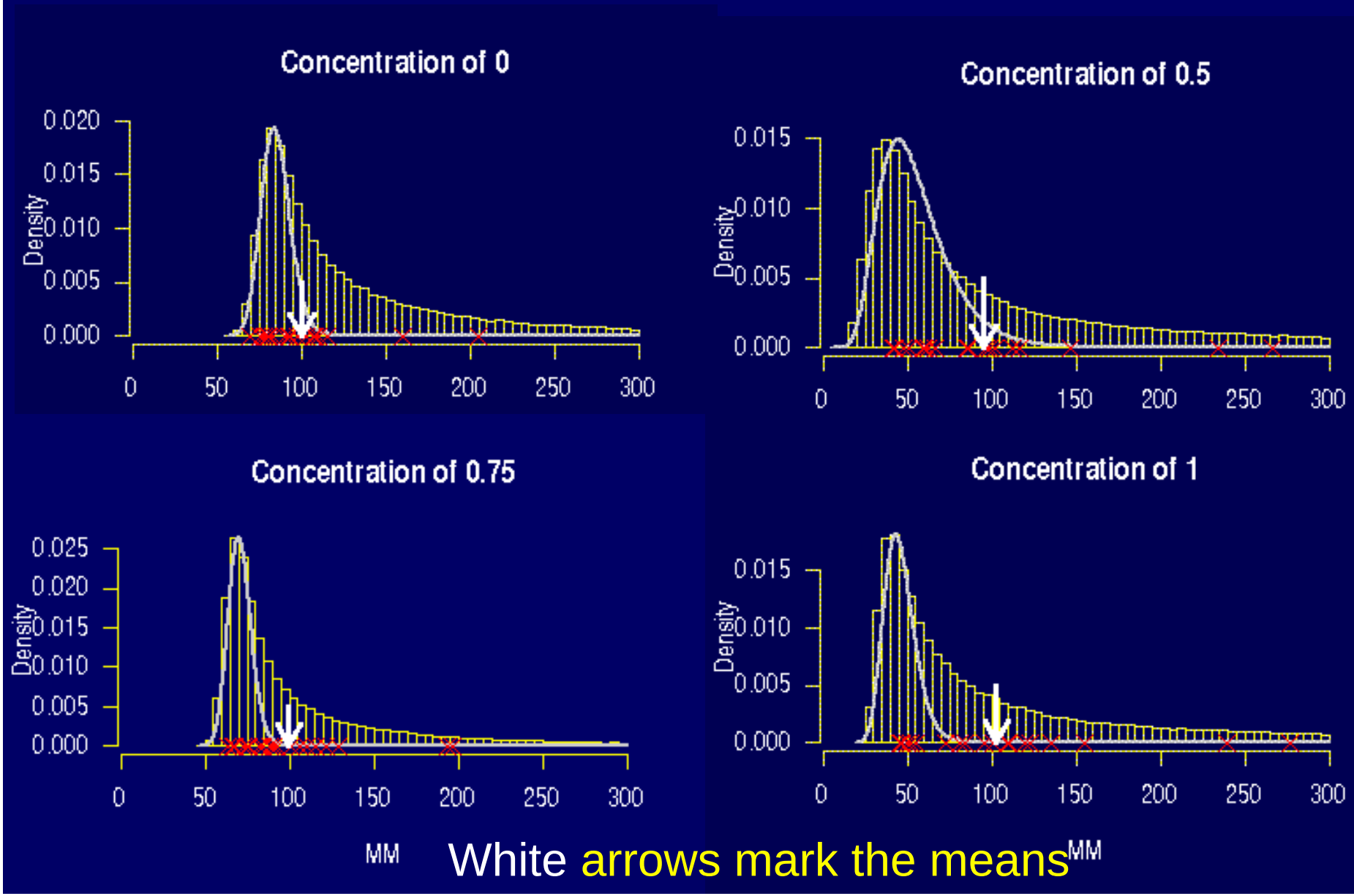
# What we do: four steps

We use **only PM**, and ignore MM. Also, we

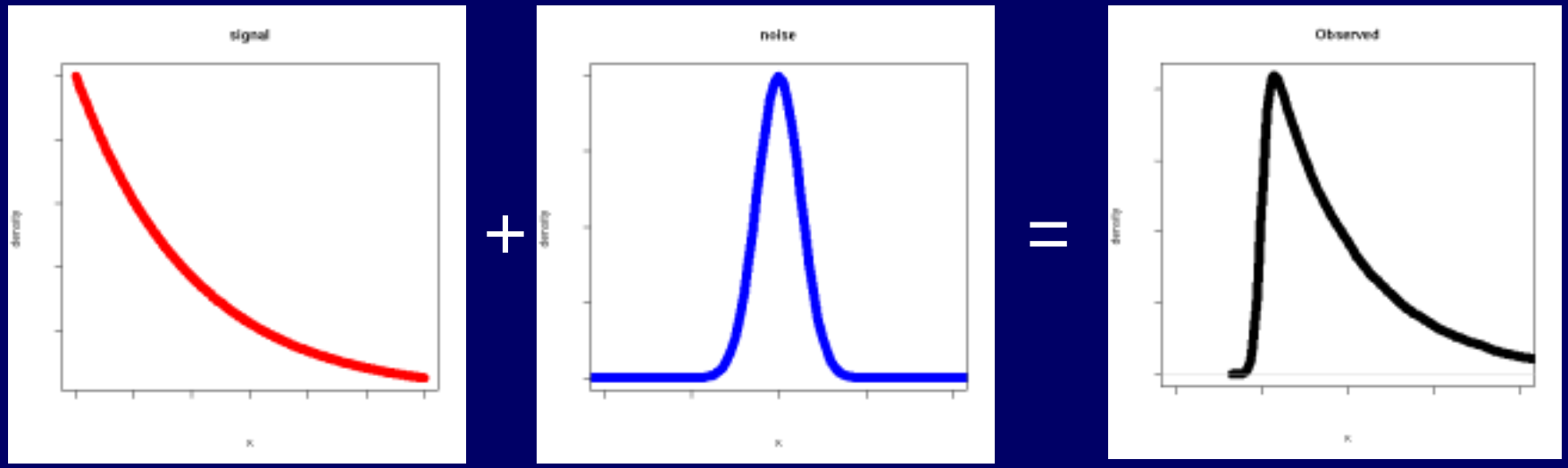
- Adjust for **background** on the raw intensity scale
- Take **$\log_2$** of background adjusted PM
- Carry out quantile **normalization** of $\log_2$(PM-BG), with chips in suitable sets
- Conduct a **robust multi-chip** analysis (RMA) of these quantities

**We call our approach RMA**

# Why remove background?

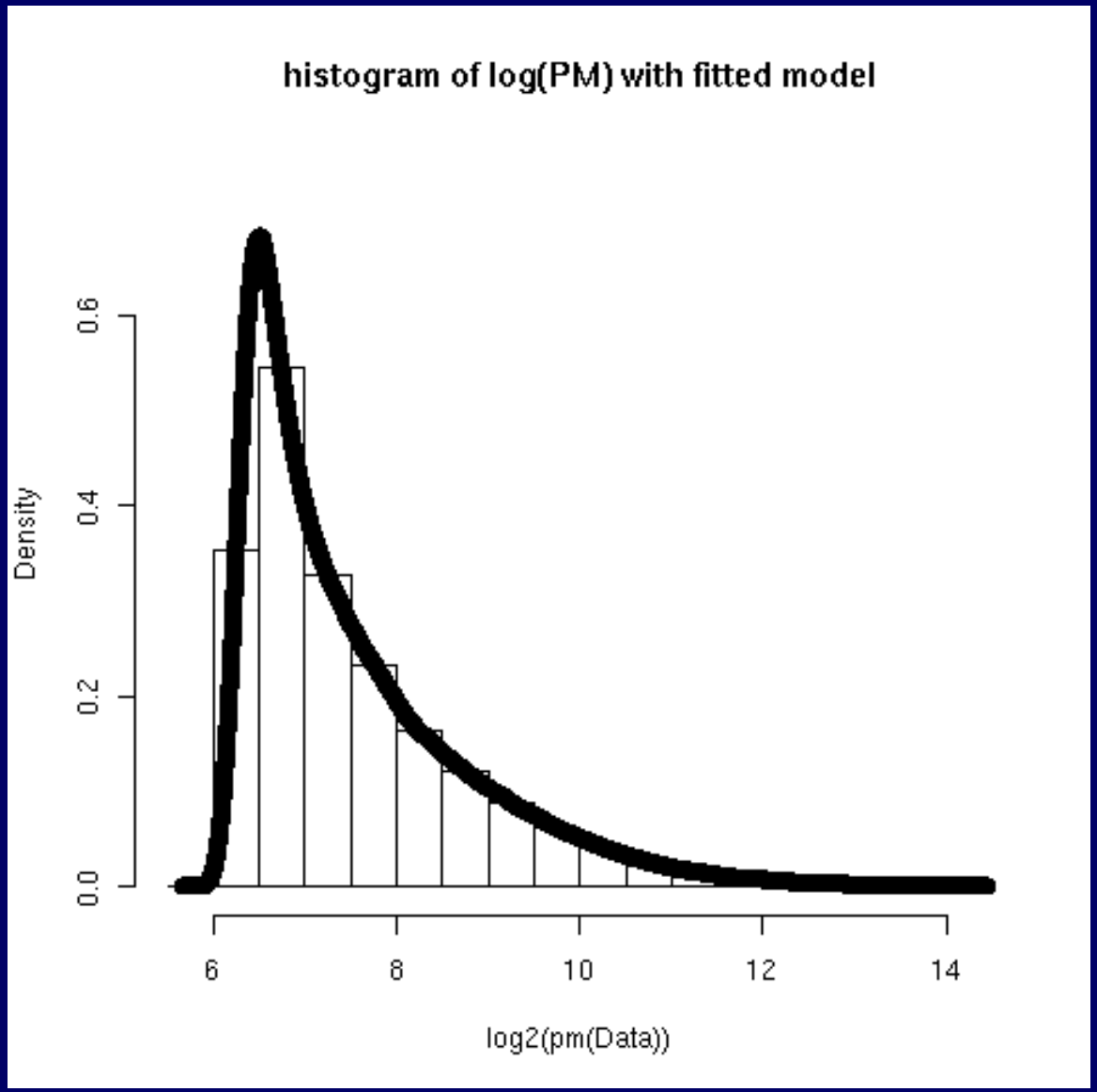


# Background model: pictorially




Signal + Noise = Observed

# PM data on $\log_2$ scale: raw and fitted model



# Background model: formulae

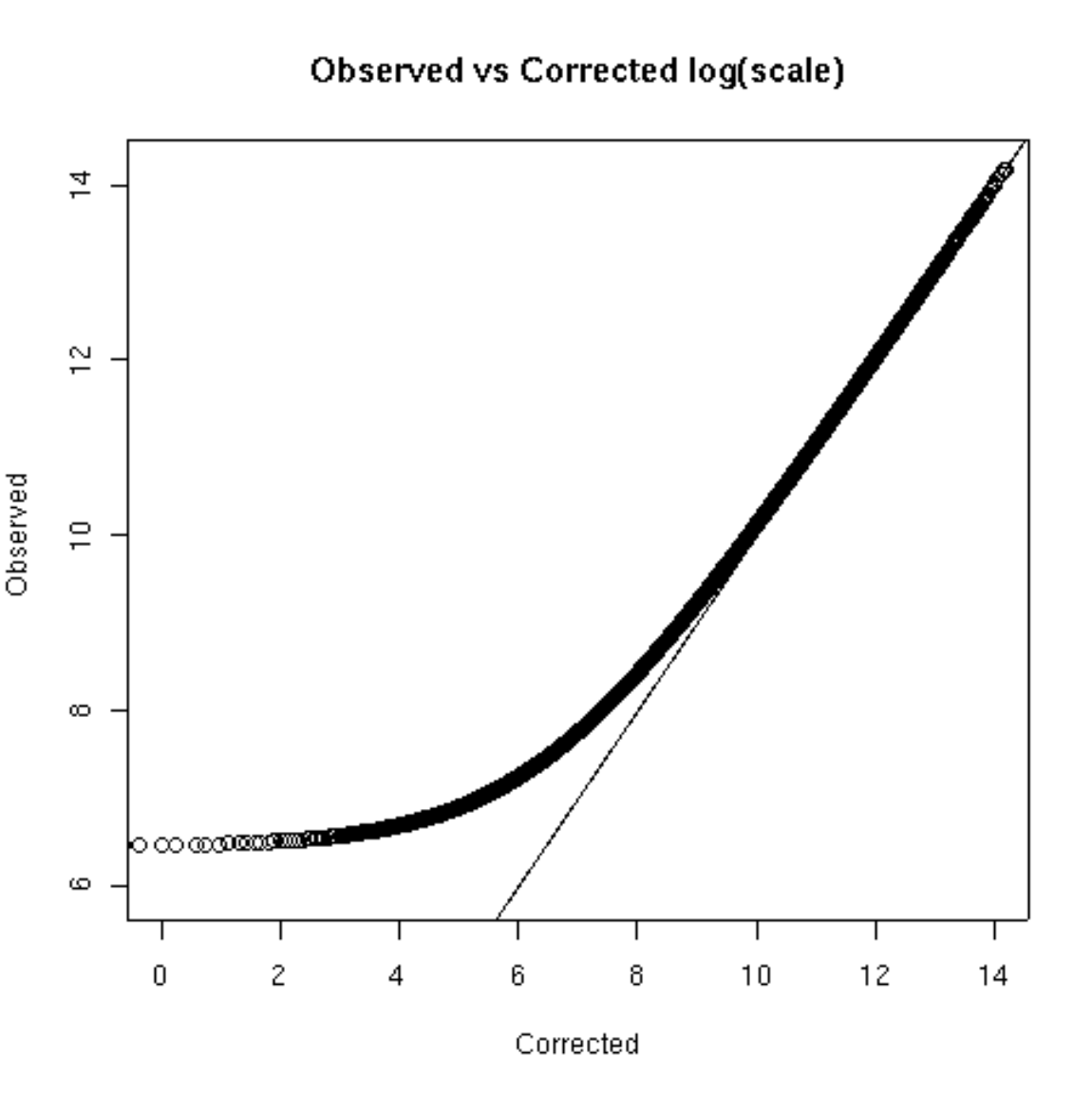
- Observed PM intensity denoted by S.
- Model S as the sum of a signal X and a background Y, S=X+Y, where we assume X is exponential ($\alpha$) and Y is Normal ($\mu$, $\sigma^2$), X, Y independent random variables.
- Background adjusted values are then E(X|S=s), which is

$$a + b[\phi(a/b) - \phi((s-a)/b)]/[\Phi(a/b) - \Phi((s-a)/b) - 1],$$

where $a = s - \mu - \sigma^2 \alpha$, $b = \sigma$, and $\phi$ and $\Phi$ are the normal density and cumulative density, respectively.

This is our model and formula for background correction.

# Observed PM vs Corrected PM




As s increases, the background correction asymptotes to

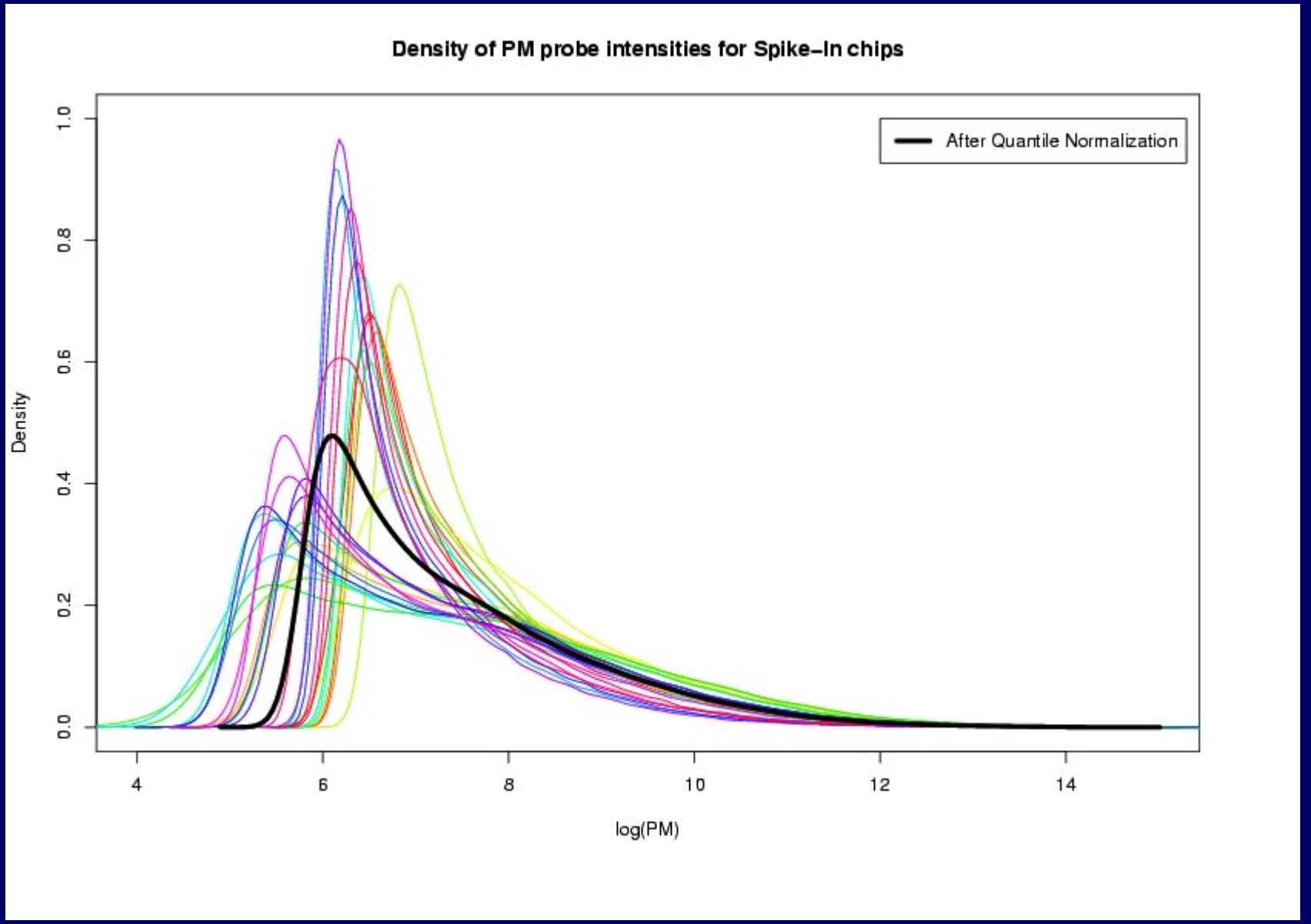
$s - \mu - \alpha\sigma^2$ .

In practice, $\mu >> \alpha\sigma^2$, so this is ~ $s - \mu$ .

# Quantile normalization

- Quantile normalization is a method to make the distribution of probe intensities the same for every chip.
- The normalization distribution is chosen by ***averaging each quantile*** across chips.
- The diagram that follows illustrates the transformation.

# Quantile normalization: pictorially



## Quantile normalization: in words

- The two distribution functions are effectively estimated by the sample quantiles.
- Quantile normalization is fast
- After normalization, variability of expression measures across chips reduced
- Looking at post-normalization PM vs pre-normalization PM (natural and log scales), you can see transformation is non linear.

# Quantile normalization: formulae

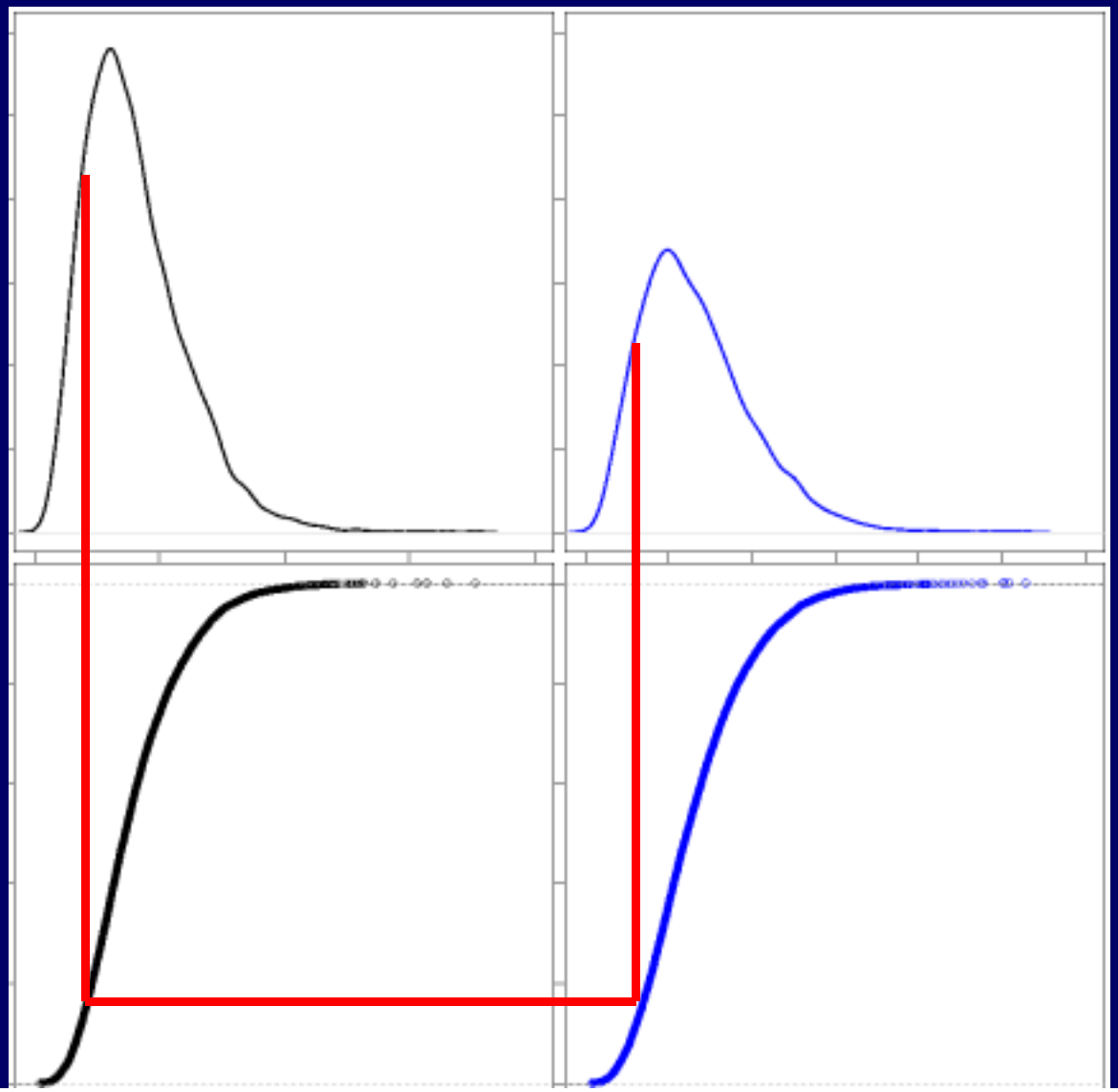
$x_{norm} = F_2^{-1}(F_1(x))$

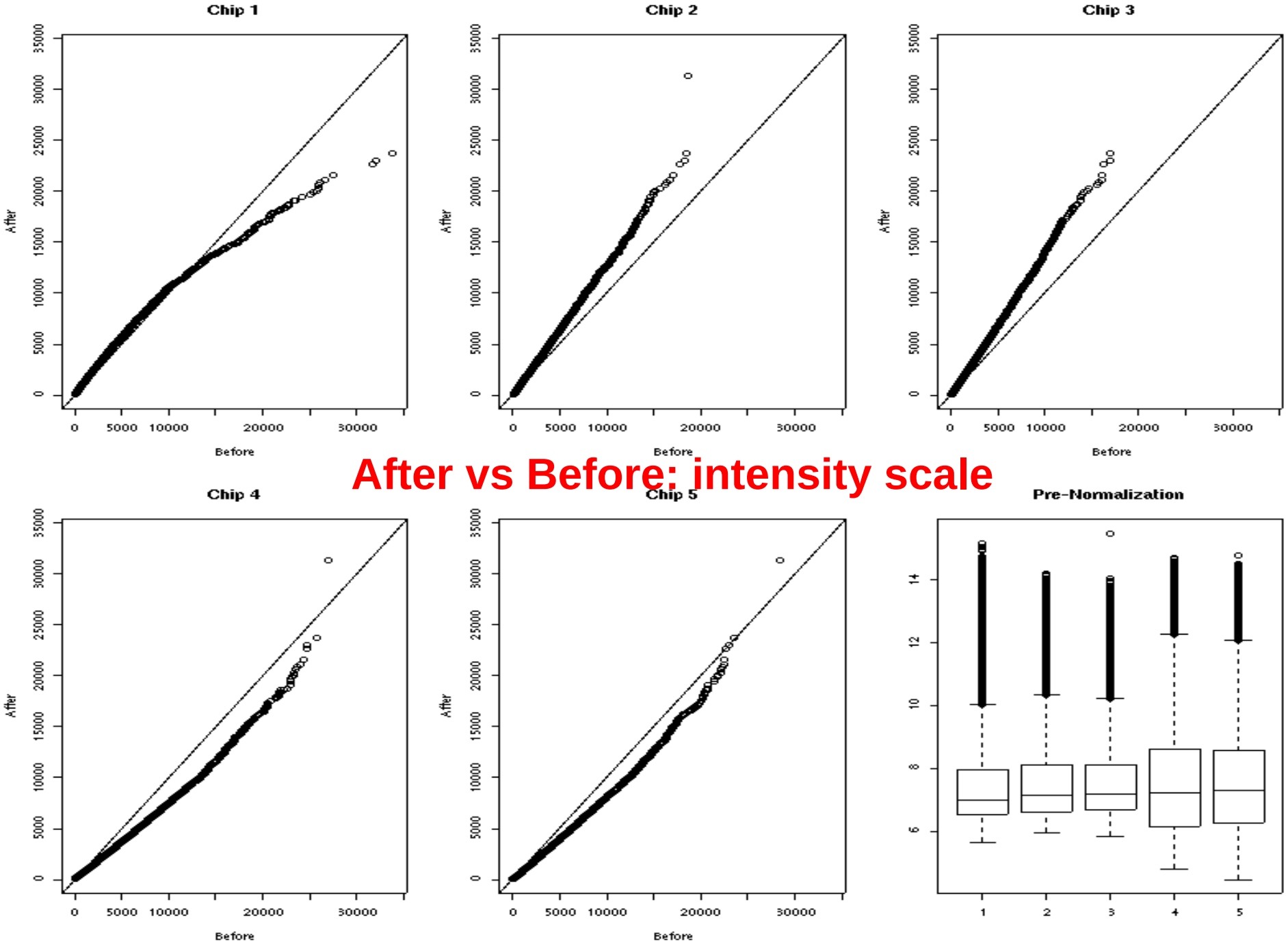
Raw data

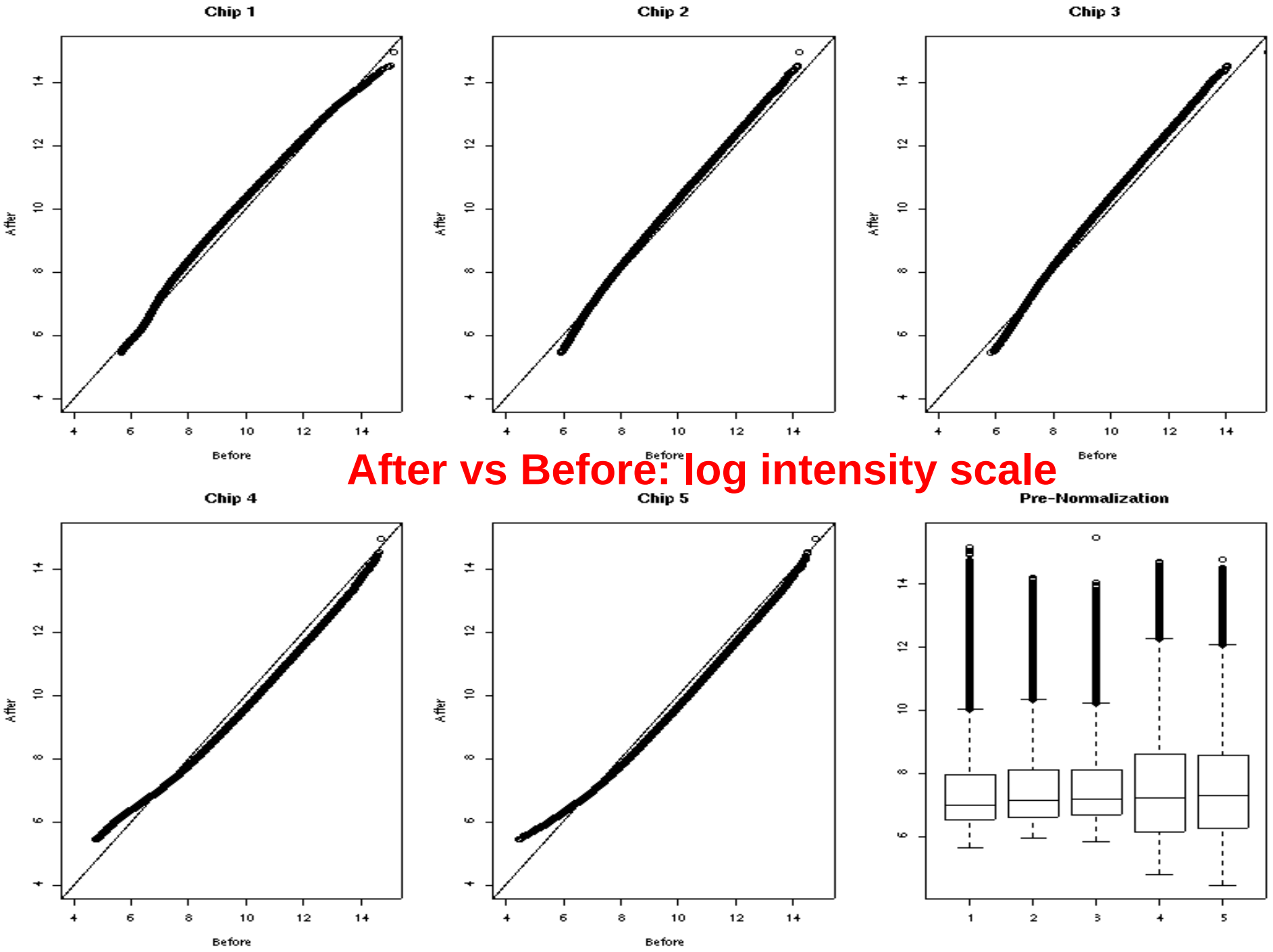
Normalization distribution $F_2(x)$

Density function

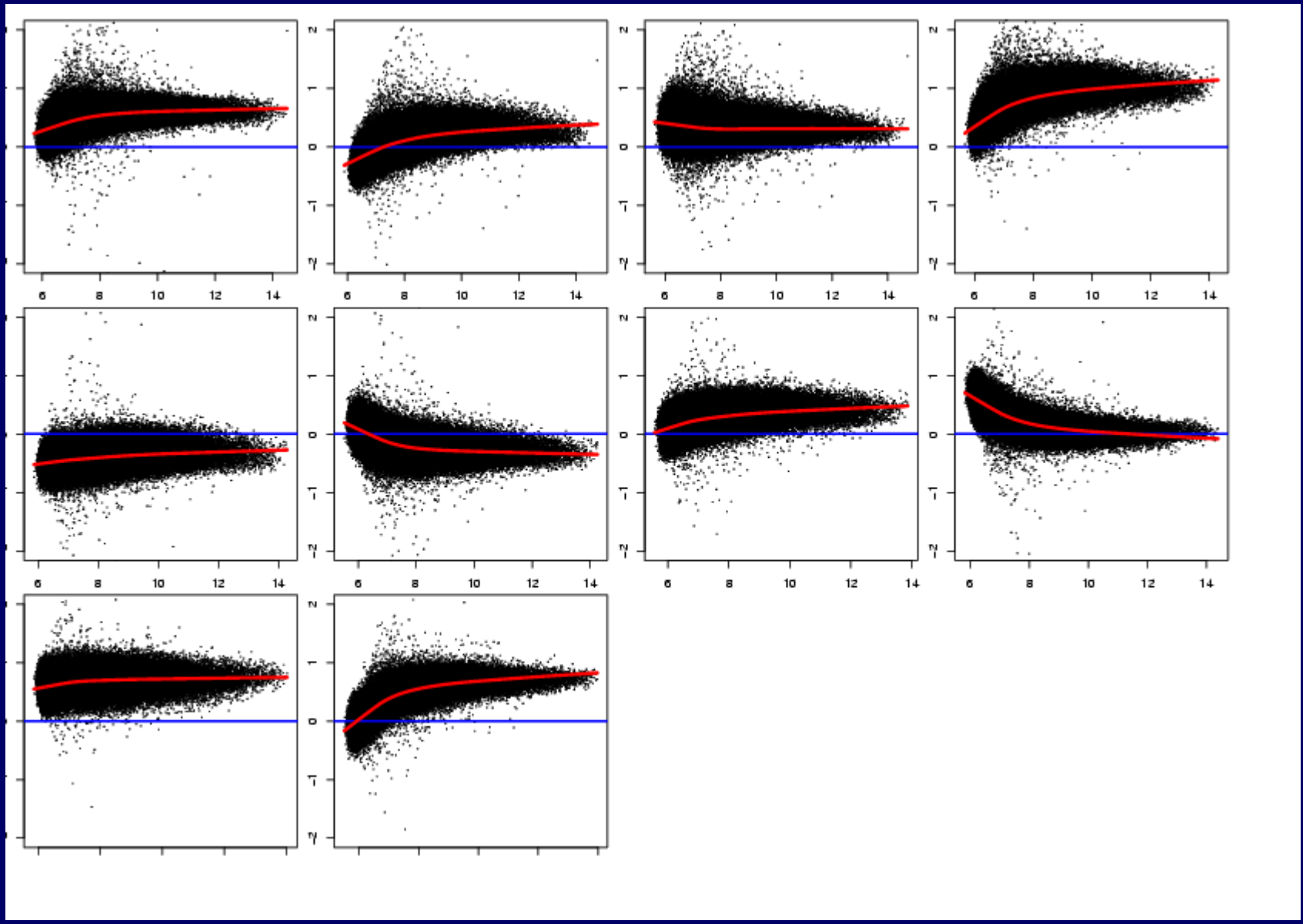
Distribution function $F_1(x)$



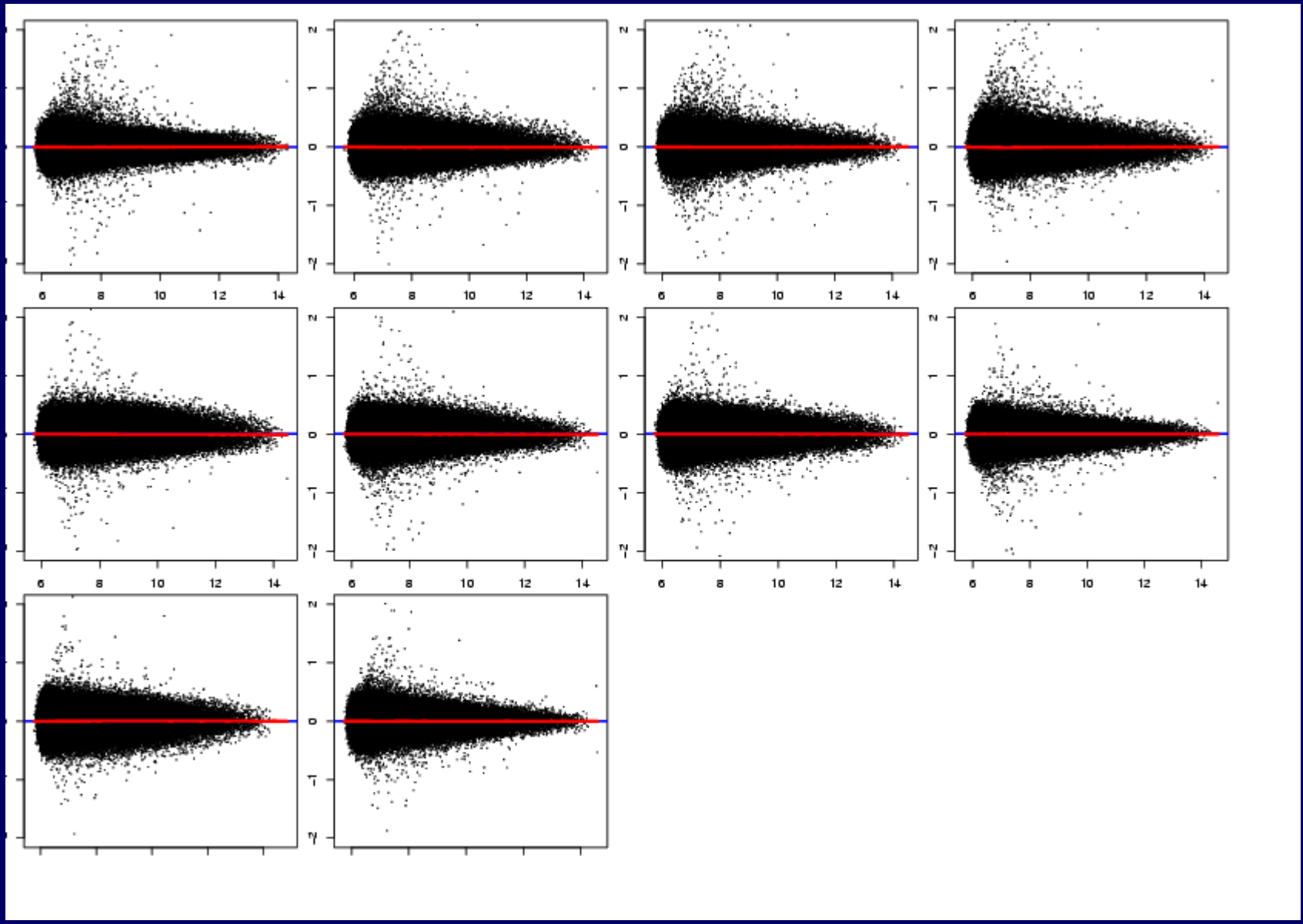

Chip 1
Chip 2
Chip 3
After vs Before: intensity scale
Chip 4
Chip 5
Pre-Normalization
After
Before
0
5000
10000
20000
30000
1
2
3
4
5
6
8
10
12
14


Chip 1
Chip 2
Chip 3
After vs Before: log intensity scale
Chip 4
Chip 5
Pre-Normalization
After
Before
4
6
8
10
12
14
1
2
3
4
5

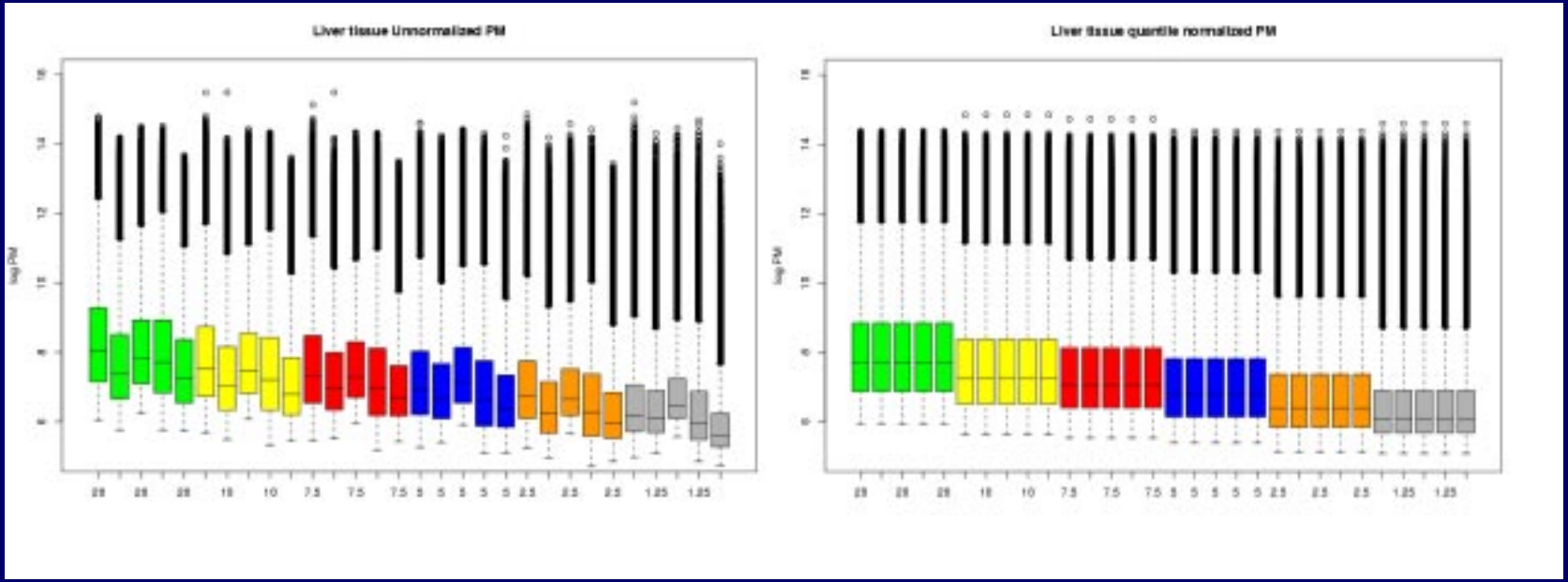
# M v A plots of chip pairs: before normalization



# M v A plots of chip pairs: after quantile normalization



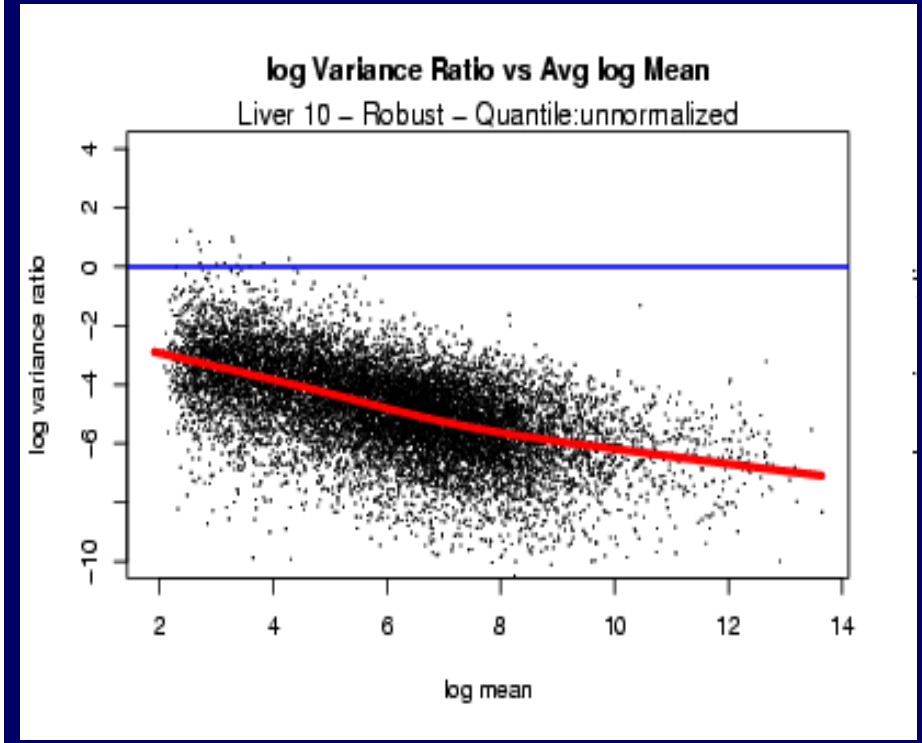
# Dilution series: before and after quantile normalization in groups of 5



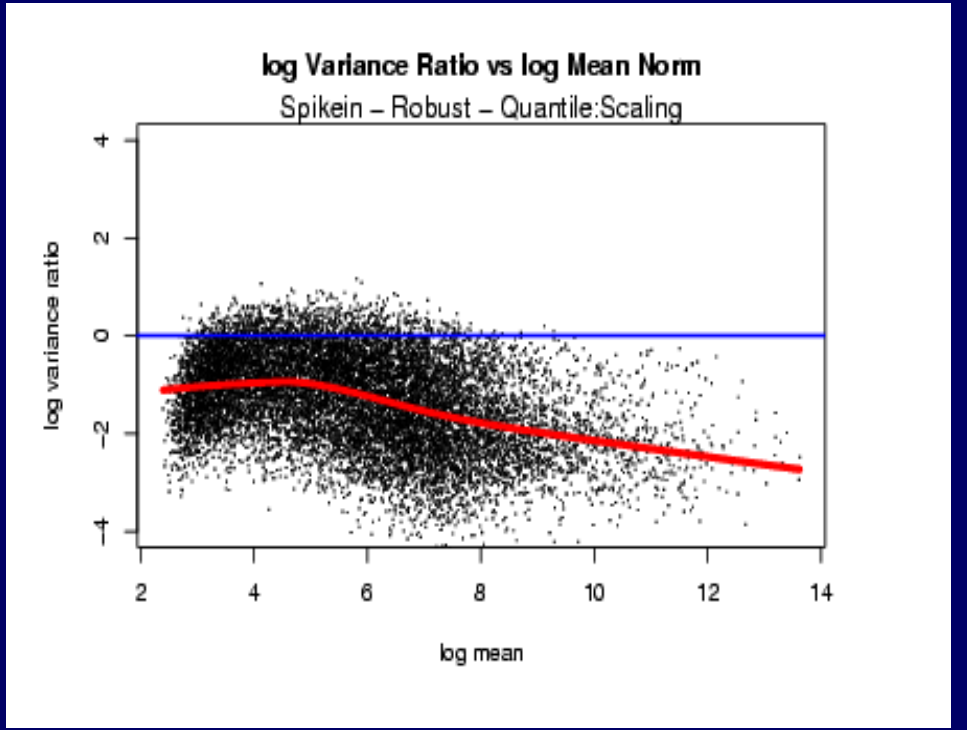

Note systematic effects of scanners 1,…,5 in before boxplots

# Normalization reduces variability in comparison with

## Quantile vs Un-normalized




## Quantile vs Affymet. normalized



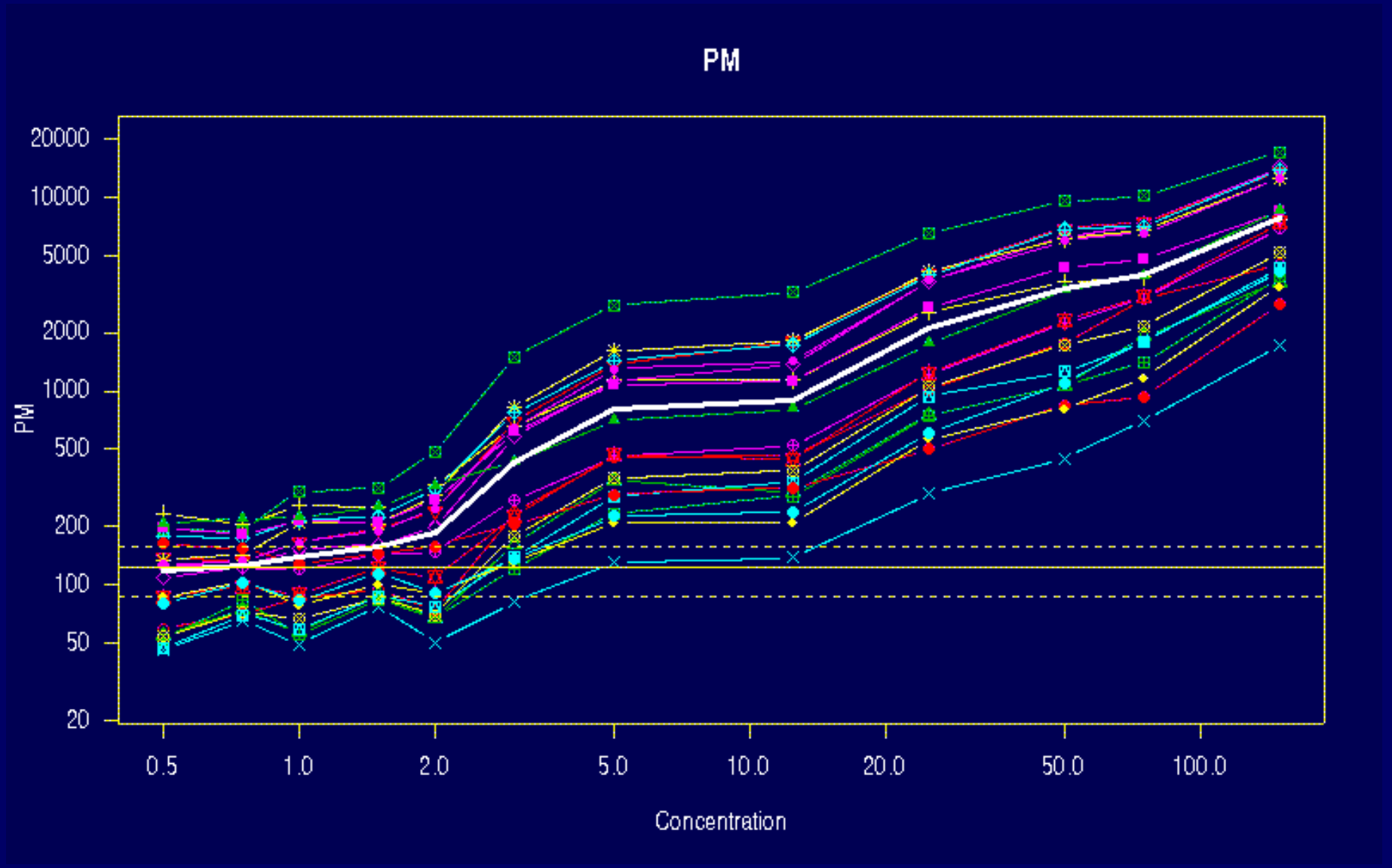

**Vertical: log[var q. norm/var other]; Horizontal: Aver. log mean**

**Note differences in vertical scales**

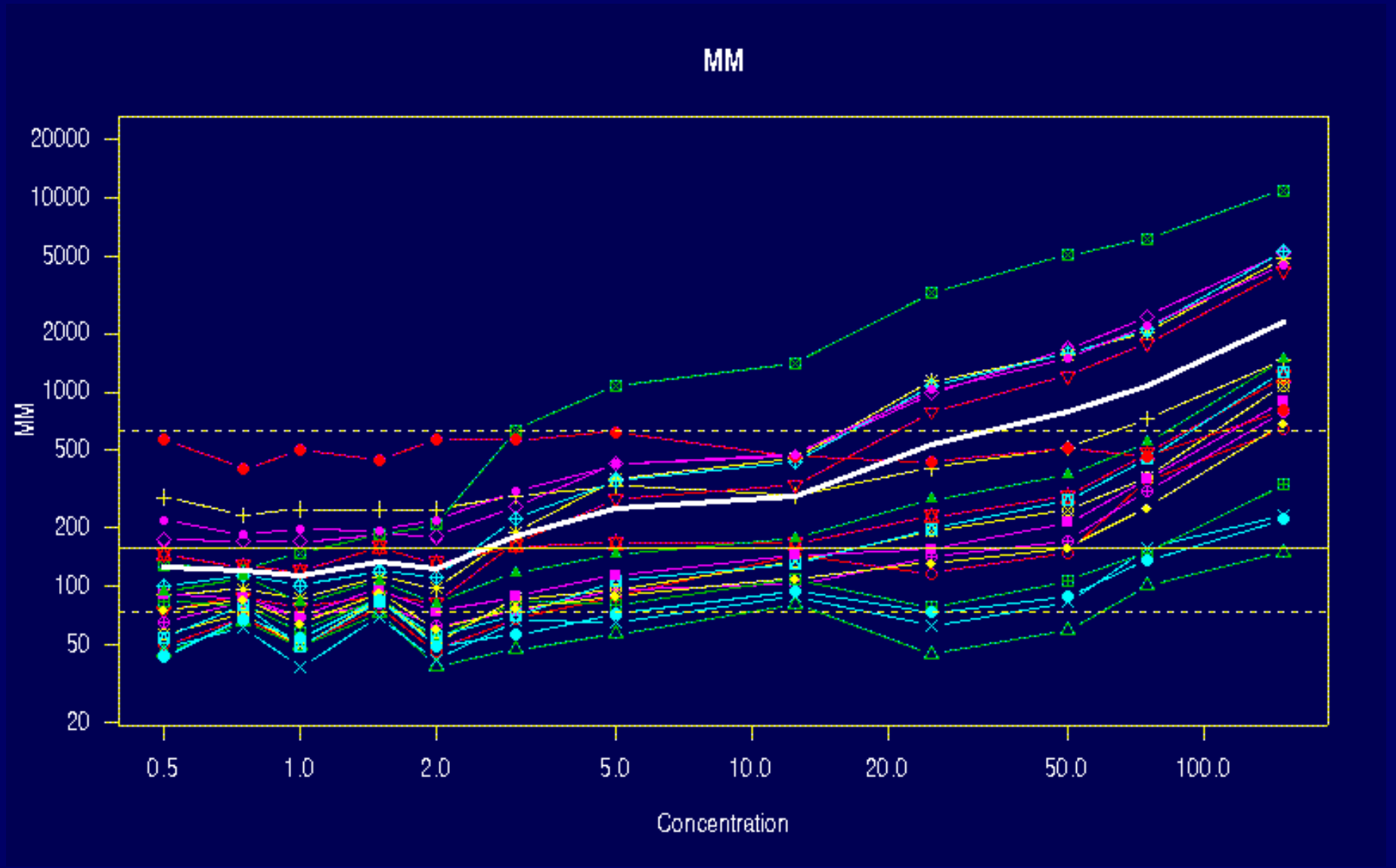
# Probe effects: spike-in experiments

- Set A: 11 control cRNAs were spiked in, all at the same concentration, which varied across chips.
- Set B: 11 control cRNAs were spiked in, all at different concentrations, which varied across chips. The concentrations were arranged in 12x12 cyclic Latin square (with 3 replicates)

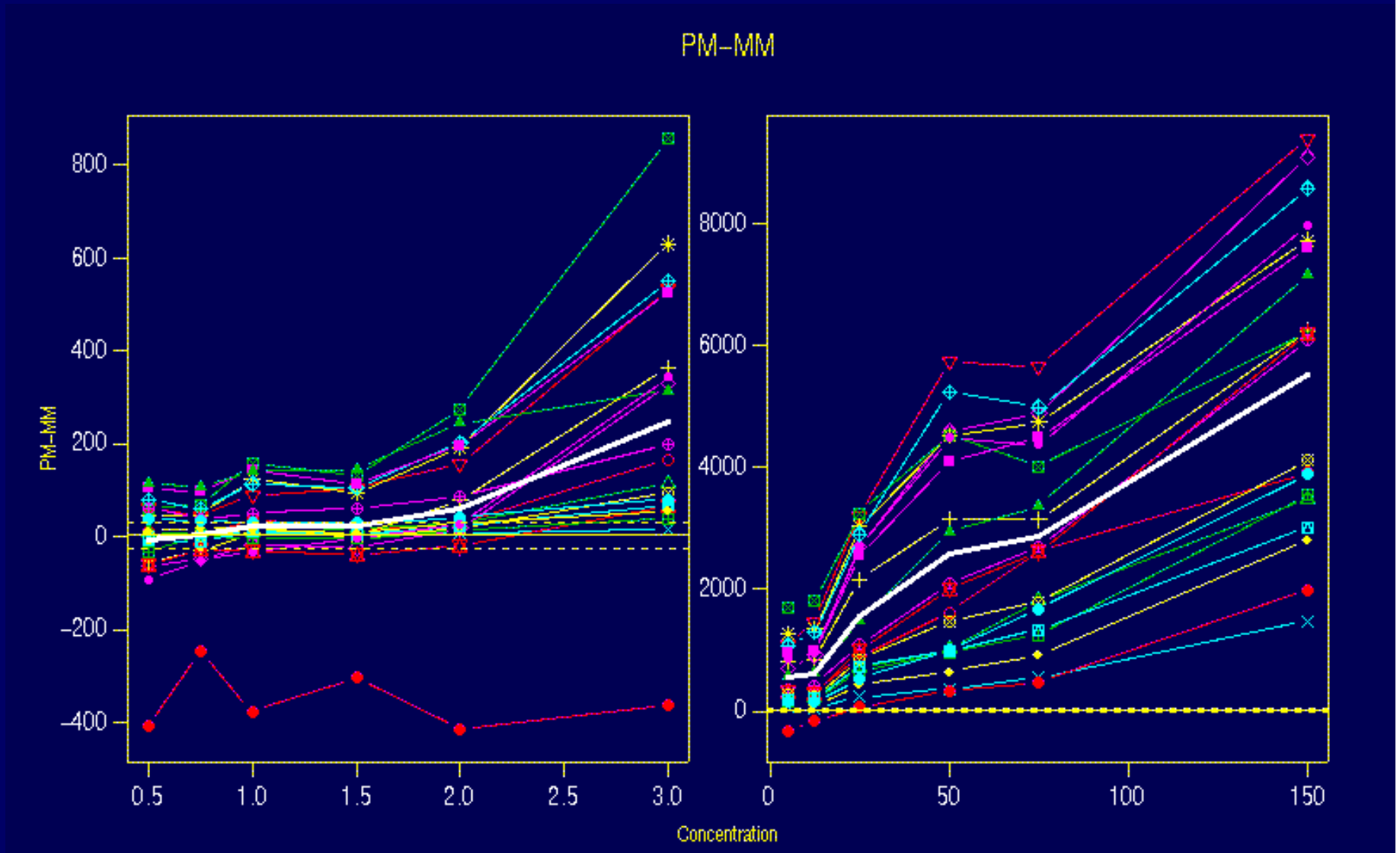
# Set A: Probe level data



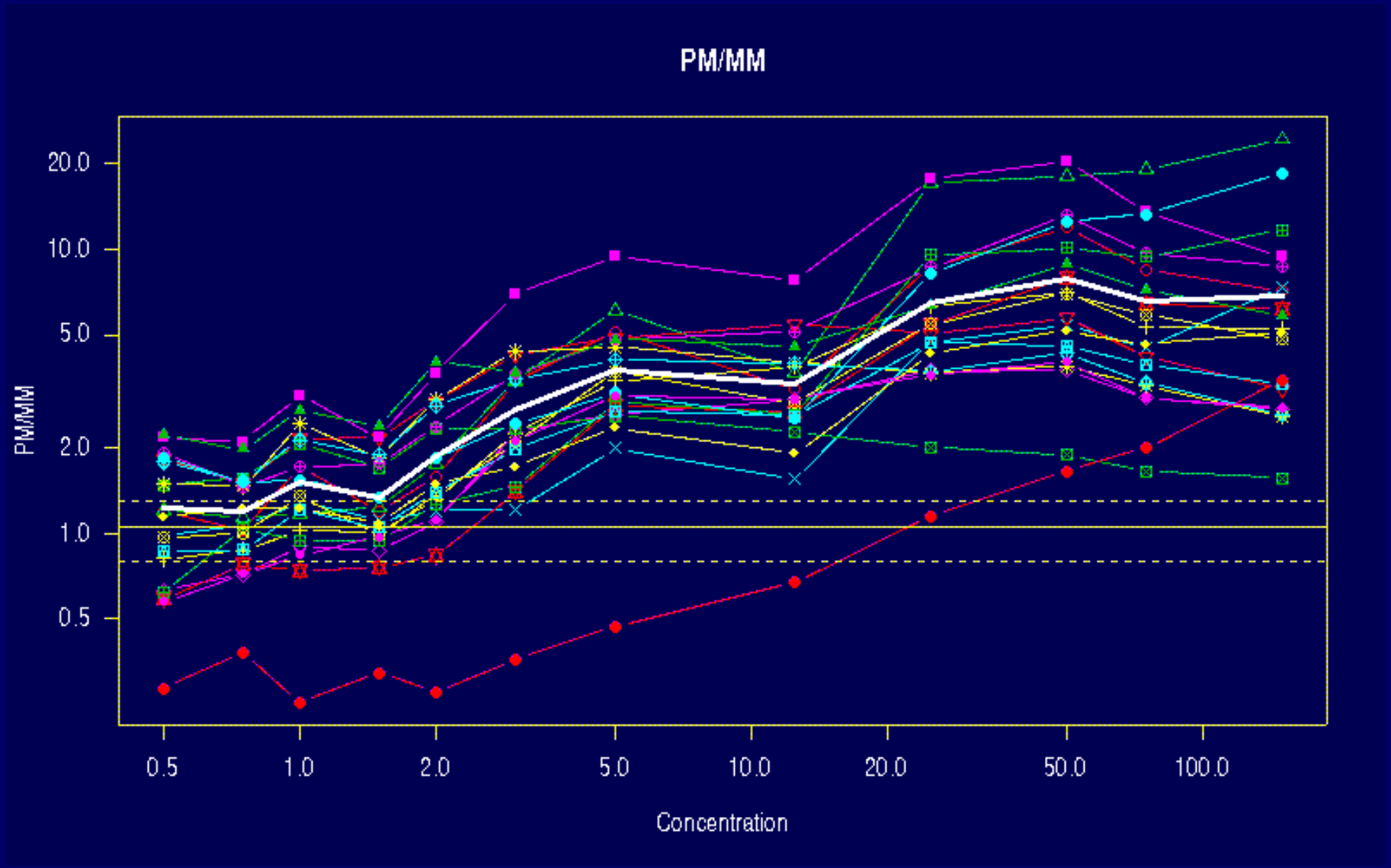
# Set A: Probe level data



# Set A: Probe level data



# Set A: Probe level data



# RMA = Robust multi-chip analysis

- Background correct PM
- Normalize (quantile normalization)
- Assume additive model:

$$\log(PM_{ij} - BG) = a_i + b_j + \varepsilon_{ij}$$

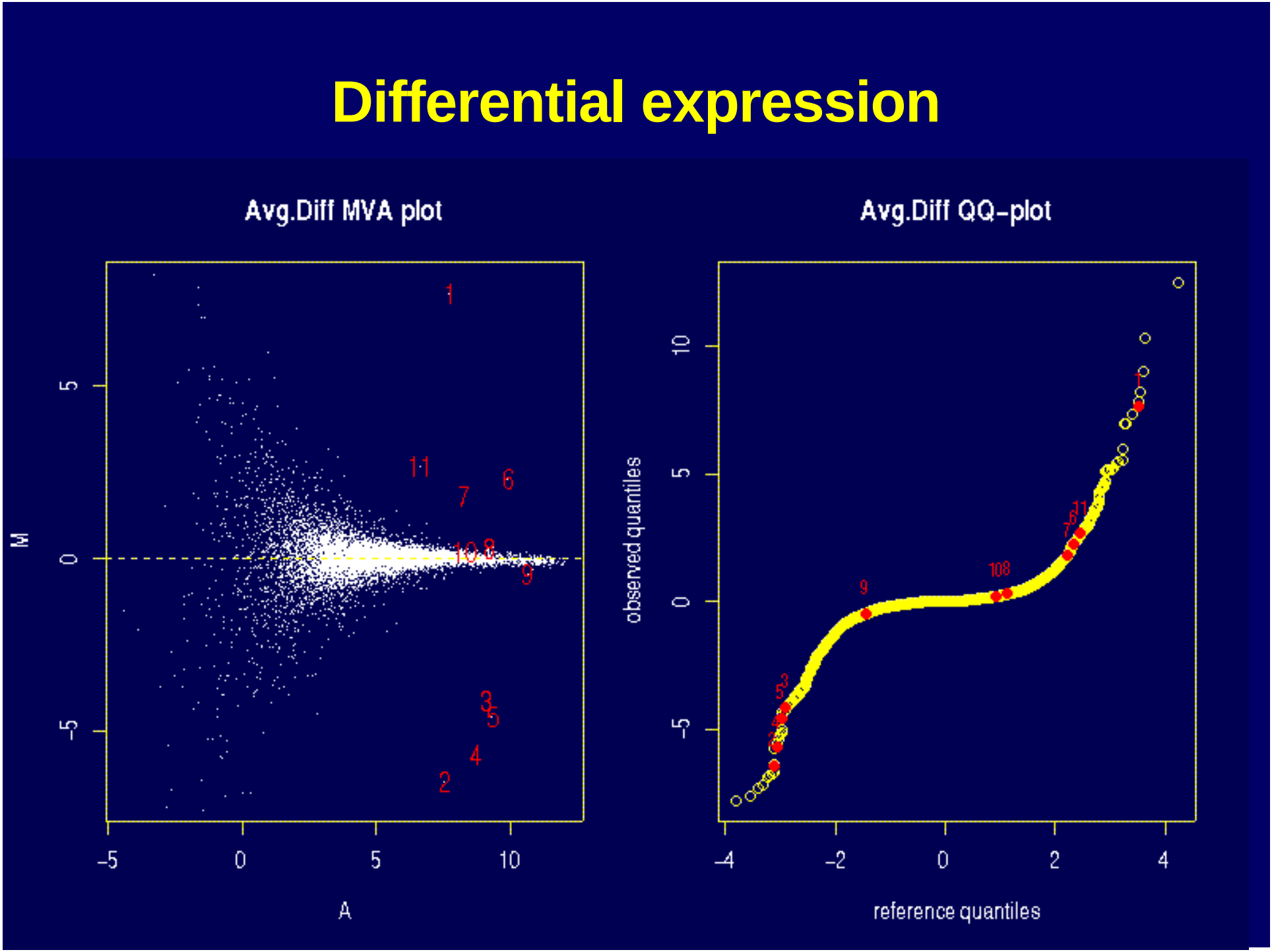
- Estimate chip effects $a_i$ and probe effects $b_j$ using a robust method

# Comparing expression summaries using spike-in data

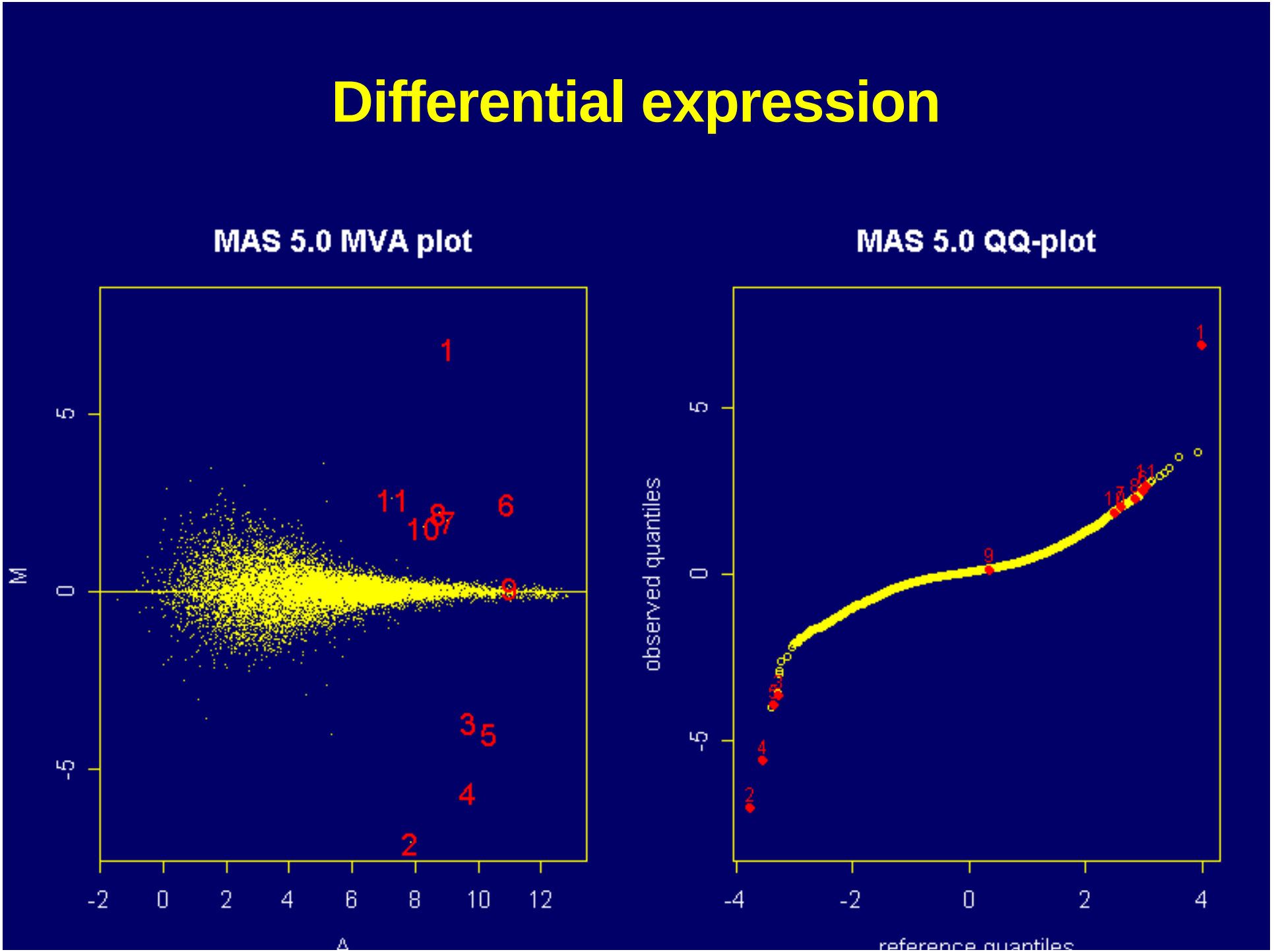
| Probe Set | Conc 1 | Conc 2 | Rank |
|---|---|---|---|
| BioB-5 | 100 | 0.5 | 1 |
| BioB-3 | 0.5 | 25.0 | 2 |
| BioC-5 | 2.0 | 75.0 | 4 |
| BioB-M | 1.0 | 37.5 | 4 |
| BioDn-3 | 1.5 | 50.0 | 5 |
| DapX-3 | 35.7 | 3.0 | 6 |
| CreX-3 | 50.0 | 5.0 | 7 |
| CreX-5 | 12.5 | 2.0 | 8 |
| BioC-3 | 25.0 | 100 | 9 |
| DapX-5 | 5.0 | 1.5 | 10 |
| DapX-M | 3.0 | 1.0 | 11 |

Later we consider 23 different combinations of concentrations

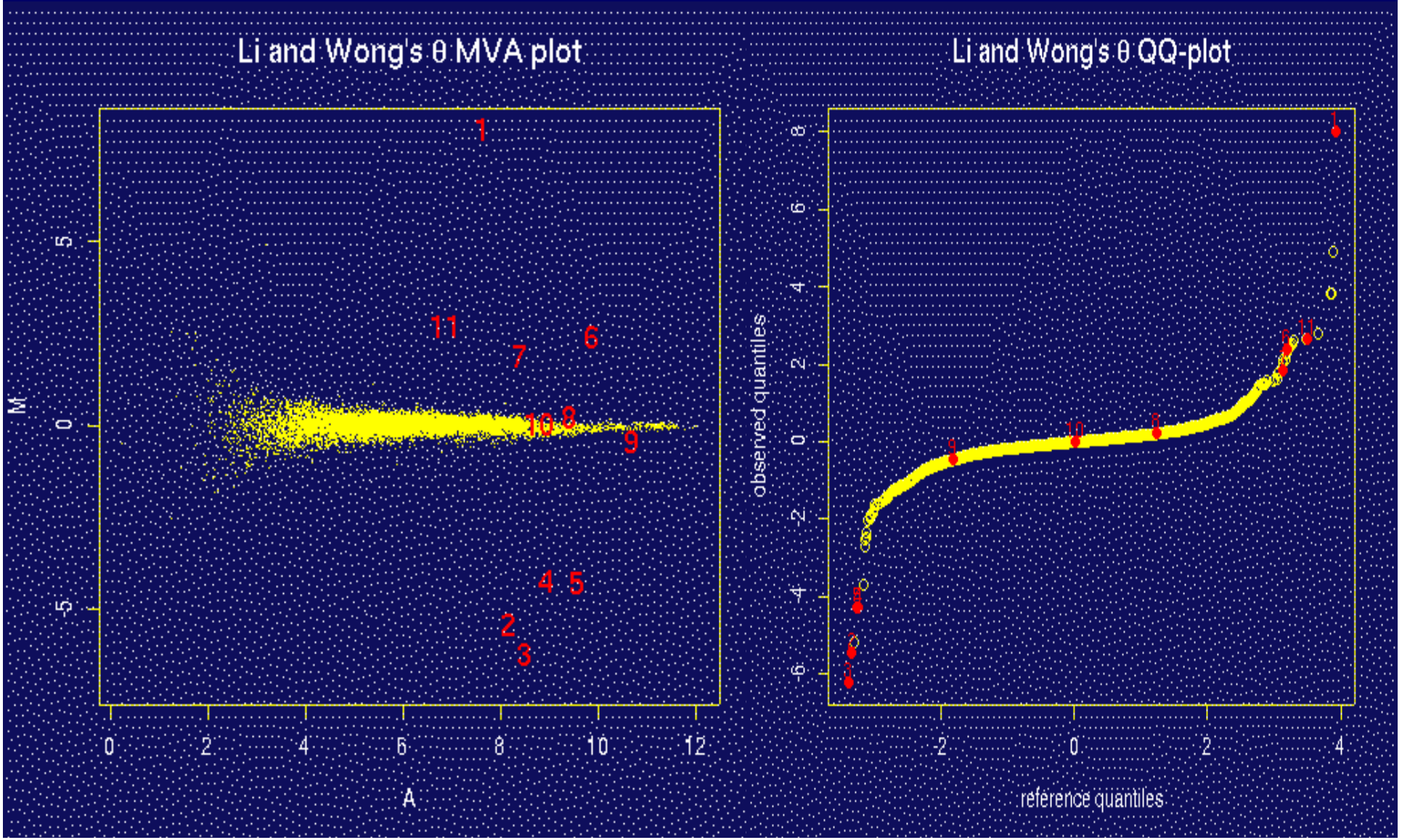
# Differential expression



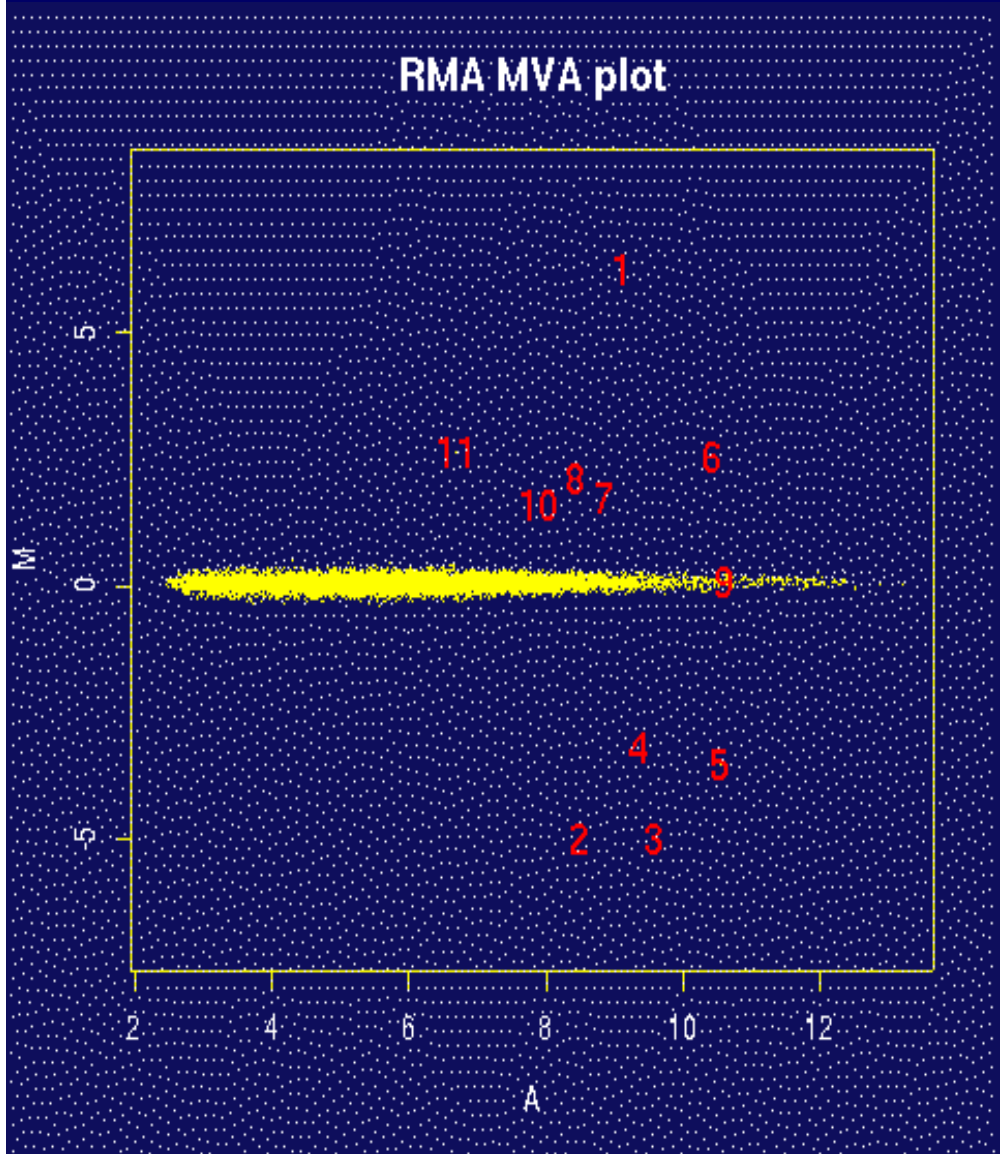
# Differential expression

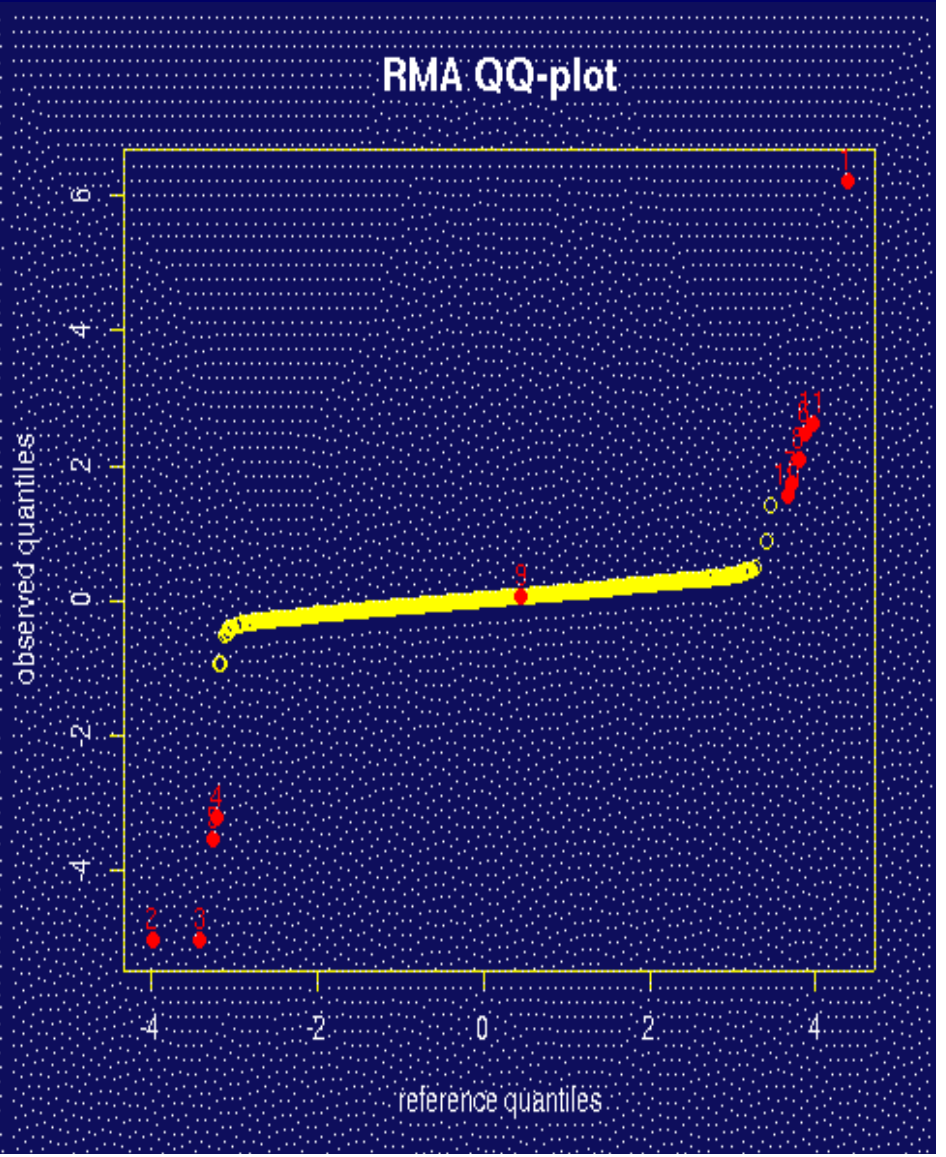


# Differential expression



# Differential expression

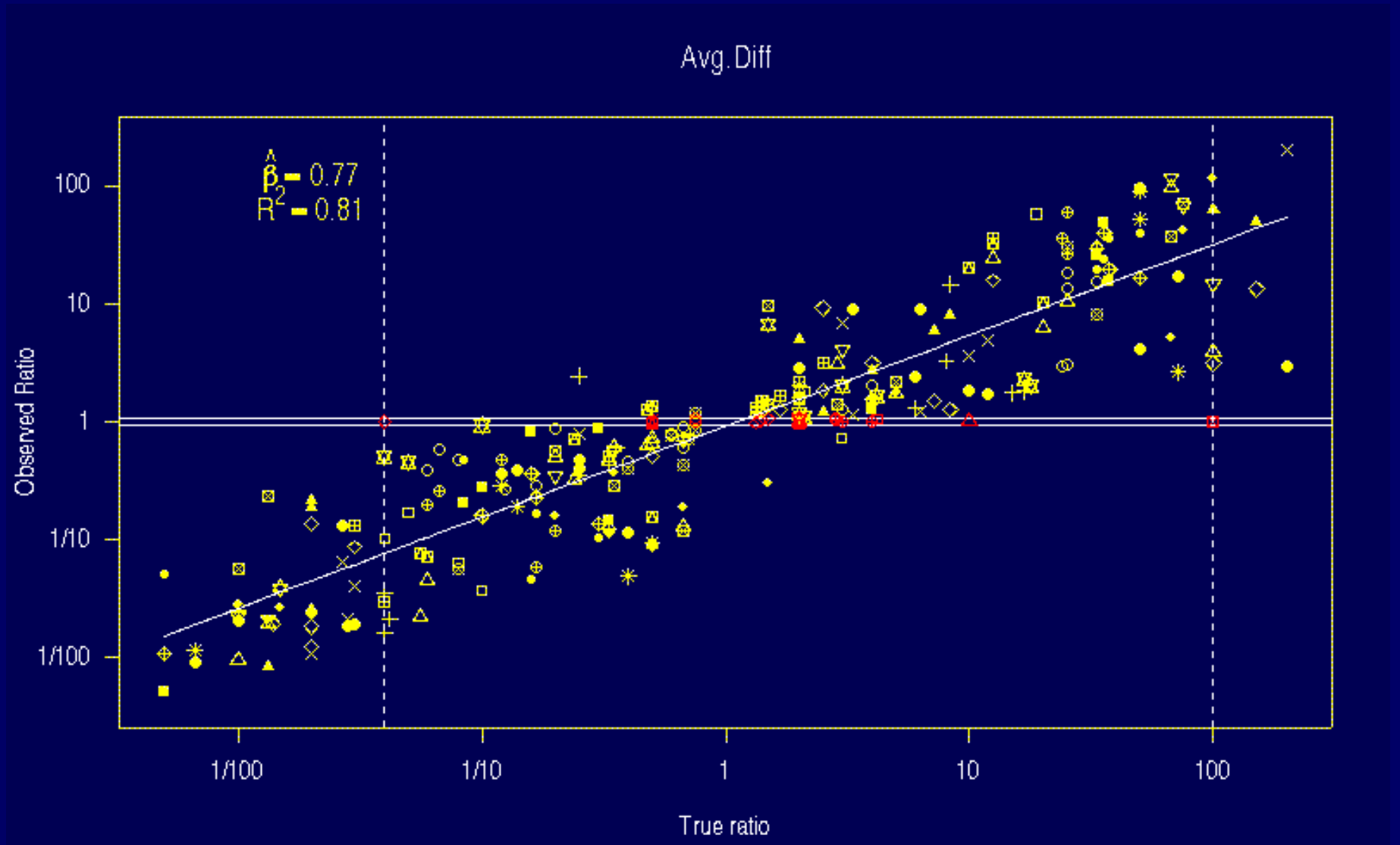


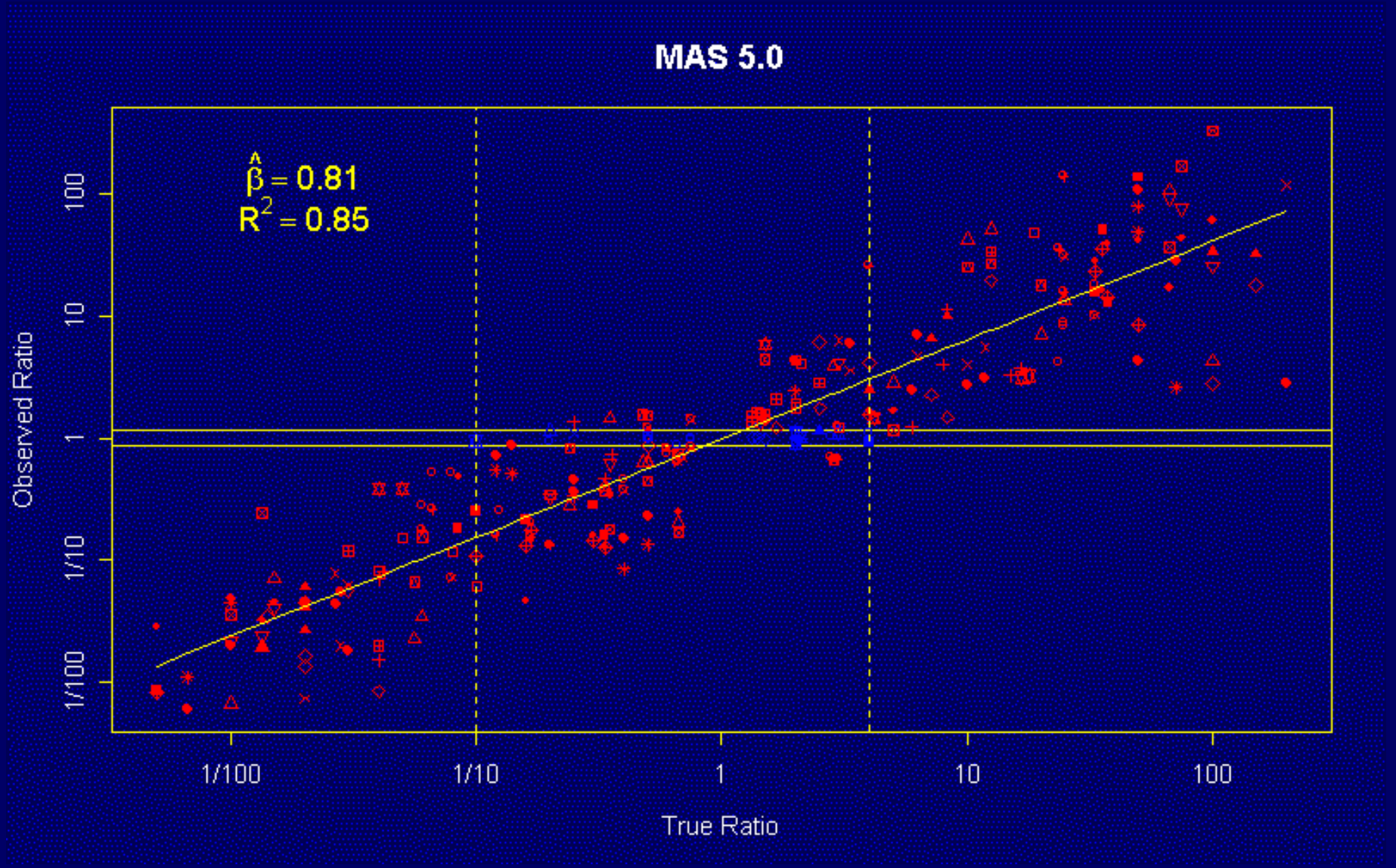
# Observed ranks

| Gene | AvDiff | MAS 5.0 | Li&Wong | AvLog(PM-BG) |
|---|---|---|---|---|
| BioB-5 | 6 | 2 | 1 | 1 |
| BioB-3 | 16 | 1 | 3 | 2 |
| BioC-5 | 74 | 6 | 2 | 5 |
| BioB-M | 30 | 3 | 7 | 3 |
| BioDn-3 | 44 | 5 | 6 | 4 |
| DapX-3 | 239 | 24 | 24 | 7 |
| CreX-3 | 333 | 73 | 36 | 9 |
| CreX-5 | 3276 | 33 | 3128 | 8 |
| BioC-3 | 2709 | 8572 | 681 | 6431 |
| DapX-5 | 2709 | 102 | 12203 | 10 |
| DapX-M | 165 | 19 | 13 | 6 |
| Top 15 | 1 | 5 | 6 | 10 |

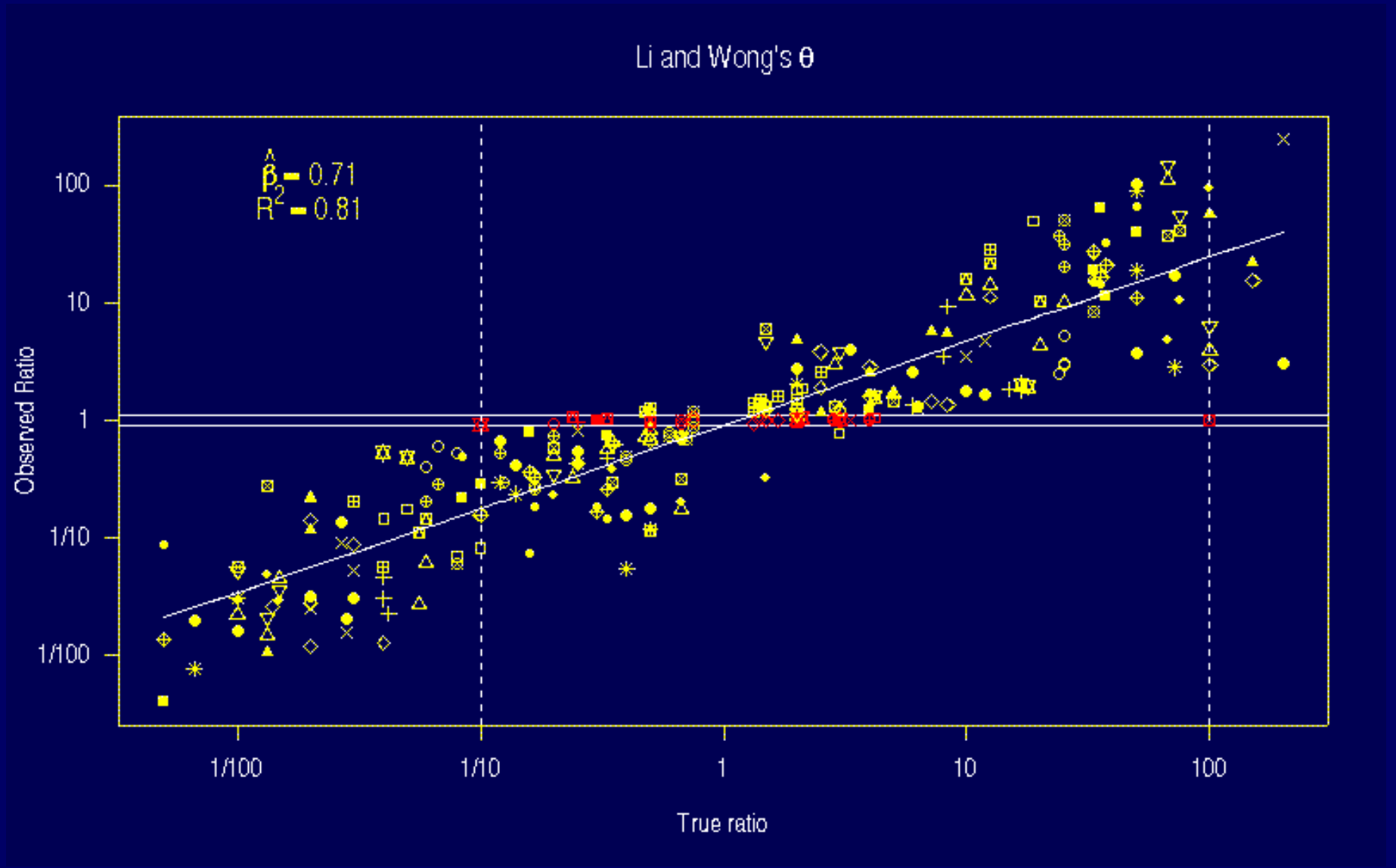
# Observed vs true ratios



# Observed vs true ratios

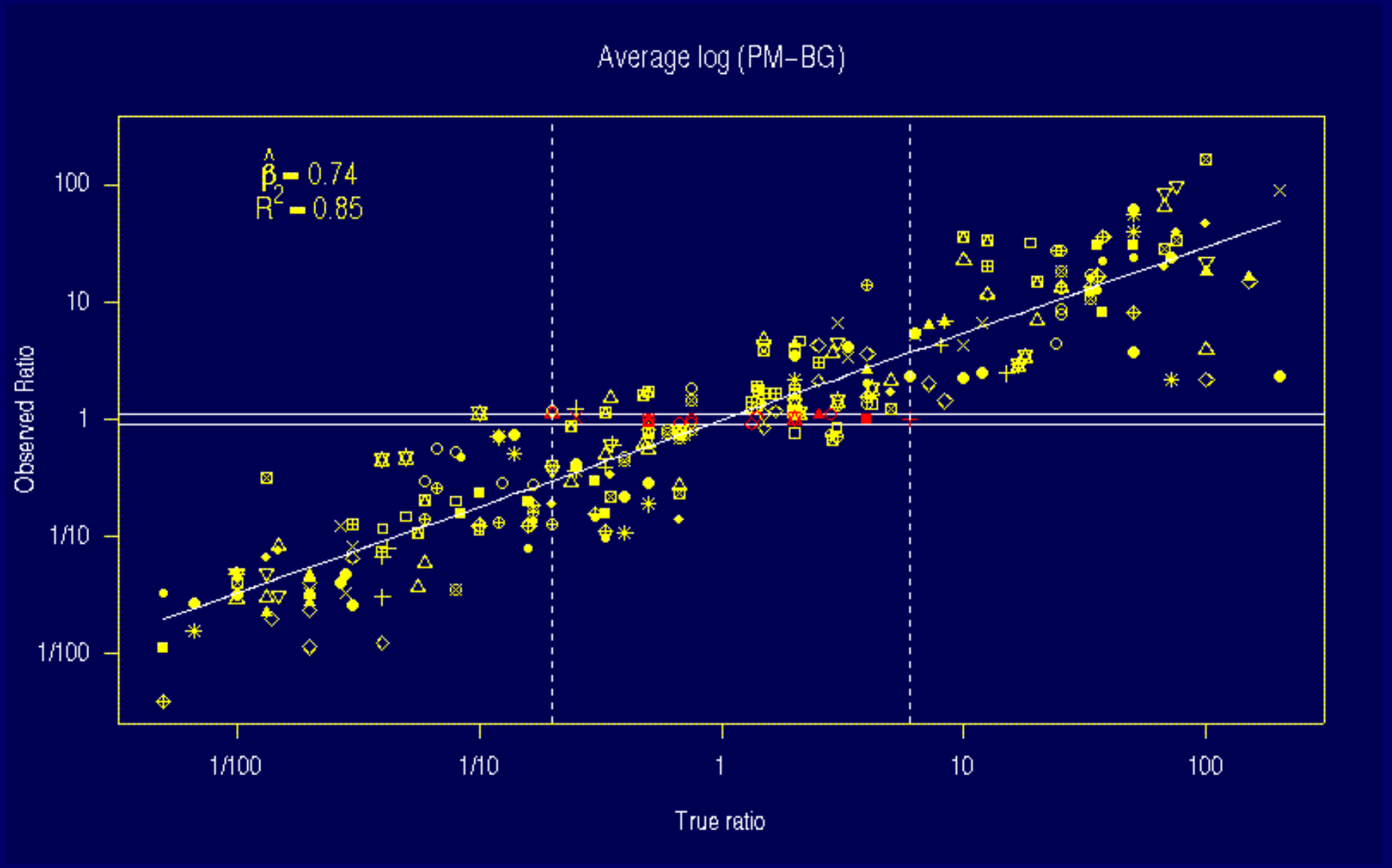


# Observed vs true ratios



# Observed vs true ratios

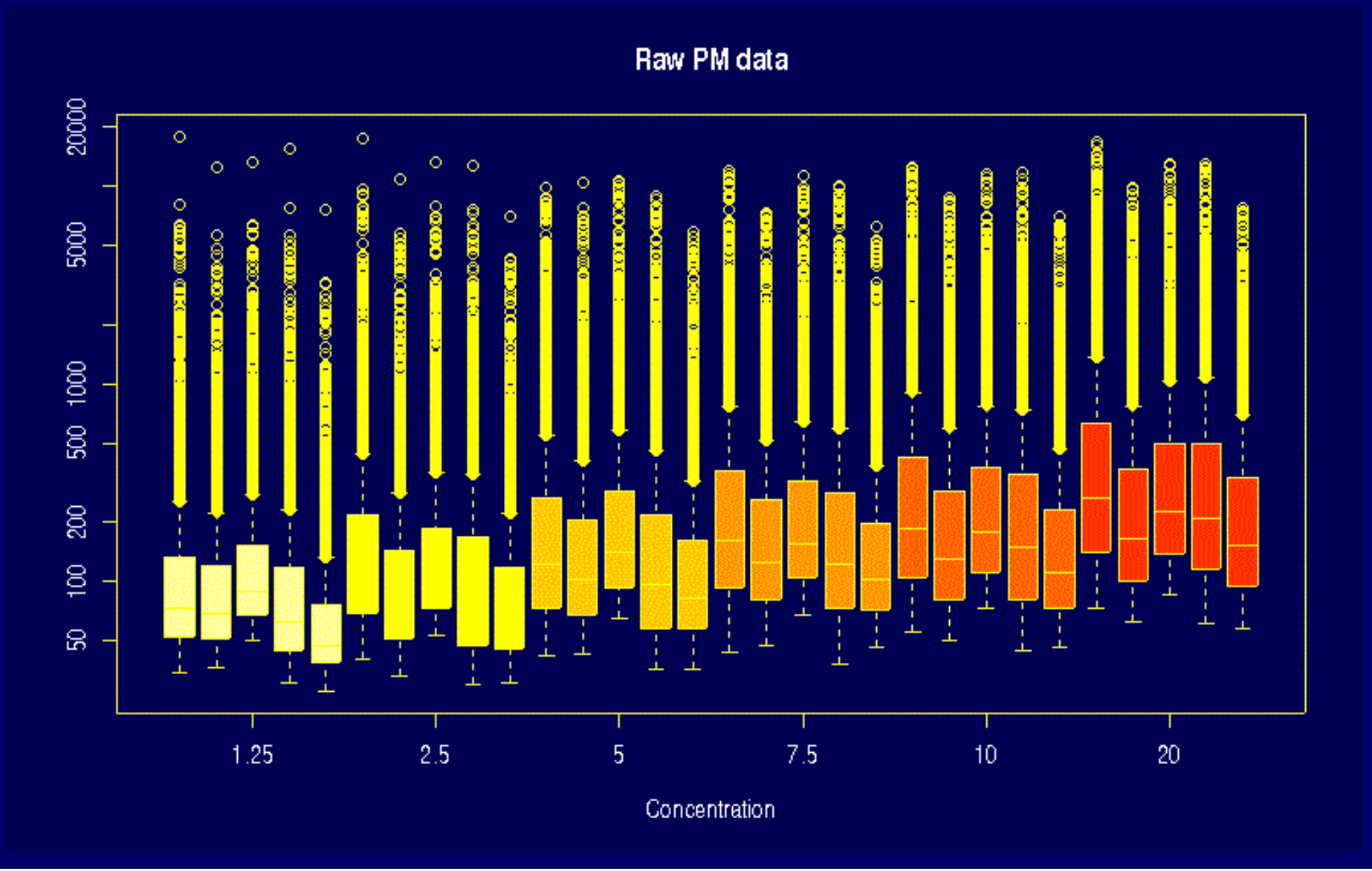
## AvLog(PM-BG) a precursor to RMA



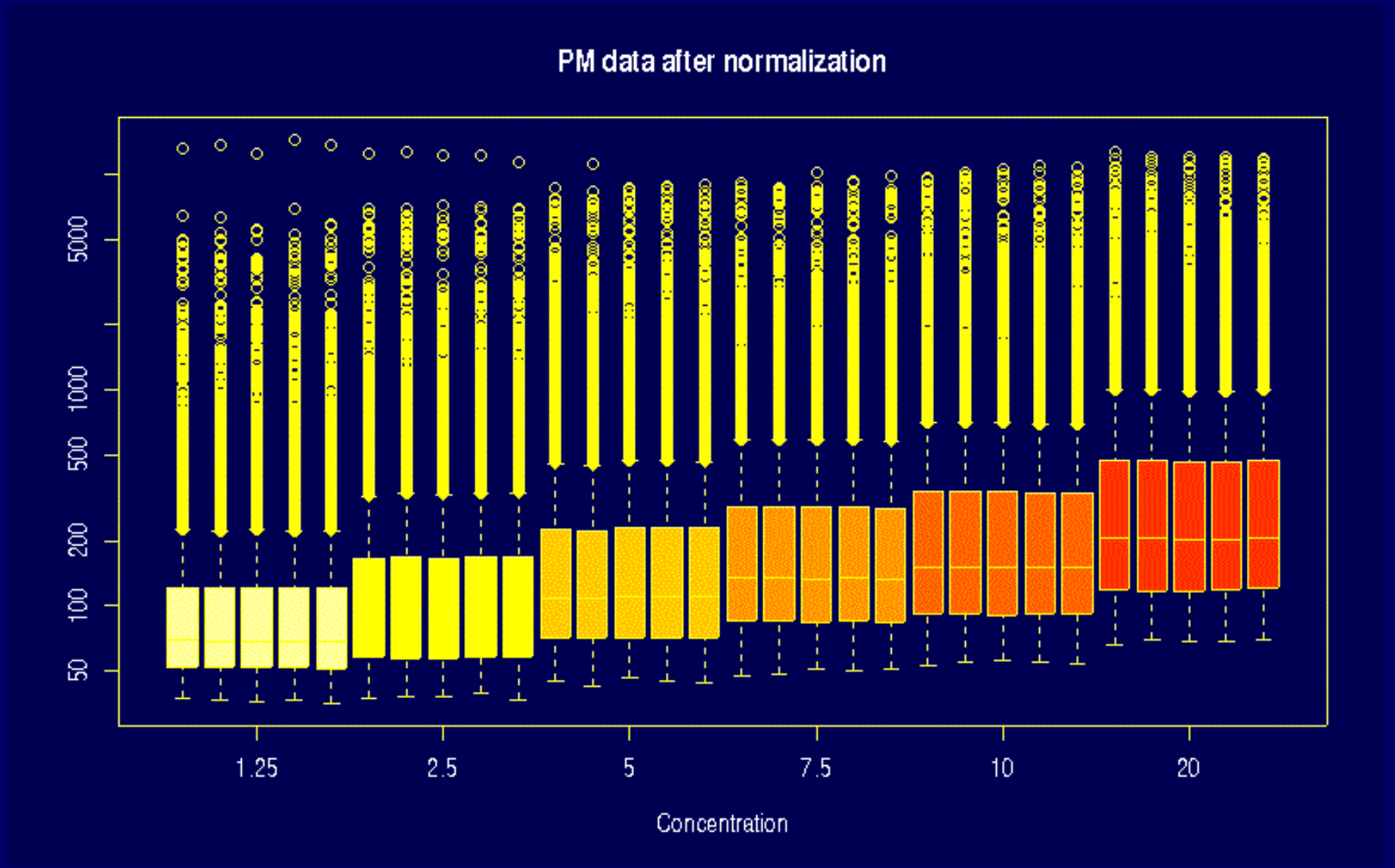
# Dilution experiment

- cRNA hybridized to human chip (HGU95) in range of proportions and dilutions
- Dilution series begins at 1.25 μg cRNA per GeneChip array, and rises through 2.5, 5.0, 7.5, 10.0, to 20.0 μg per array. 5 replicate chips were used at each dilution
- Normalize just within each set of 5 replicates
- For each probe set compute expression, average and SD over replicates

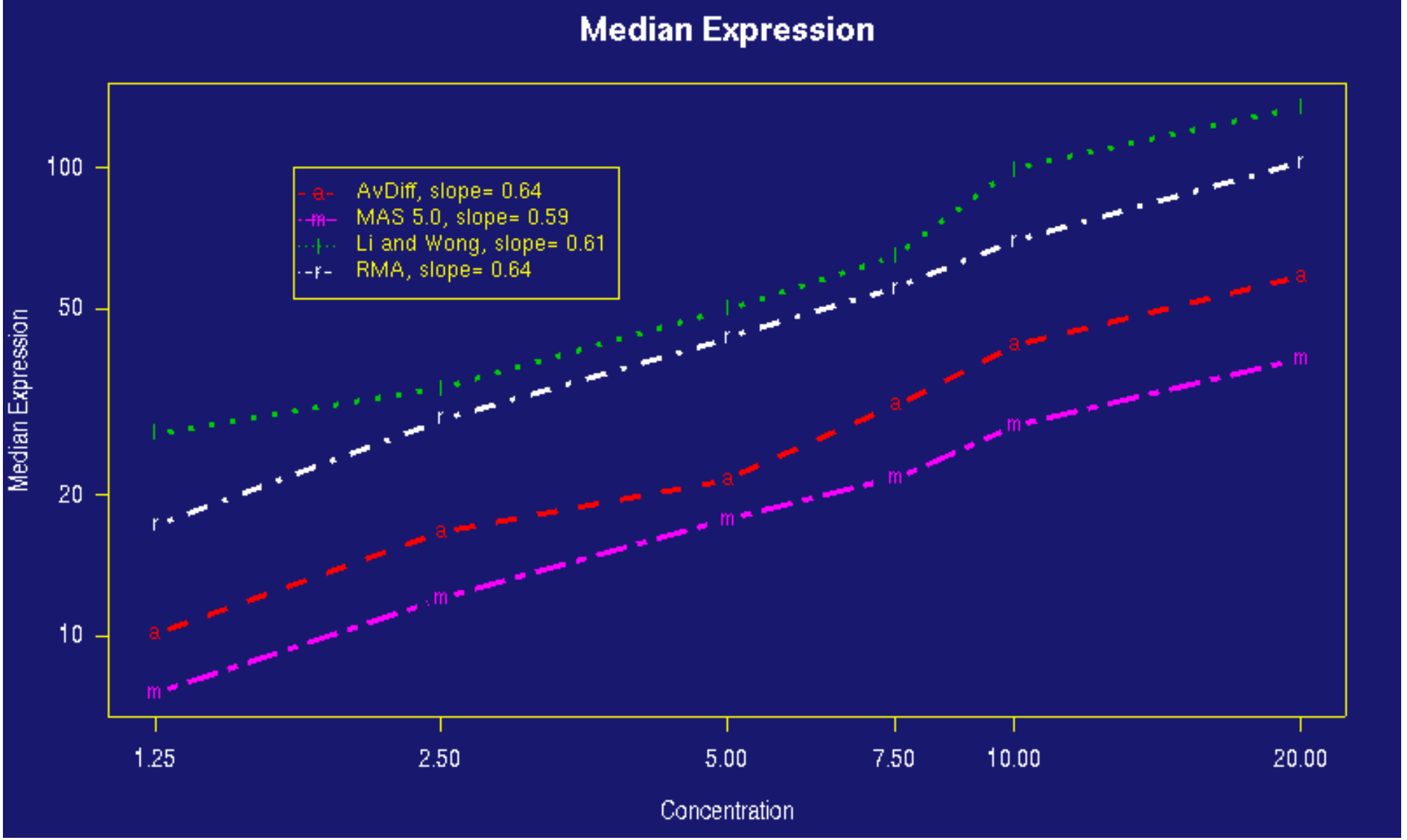
# Dilution experiment data



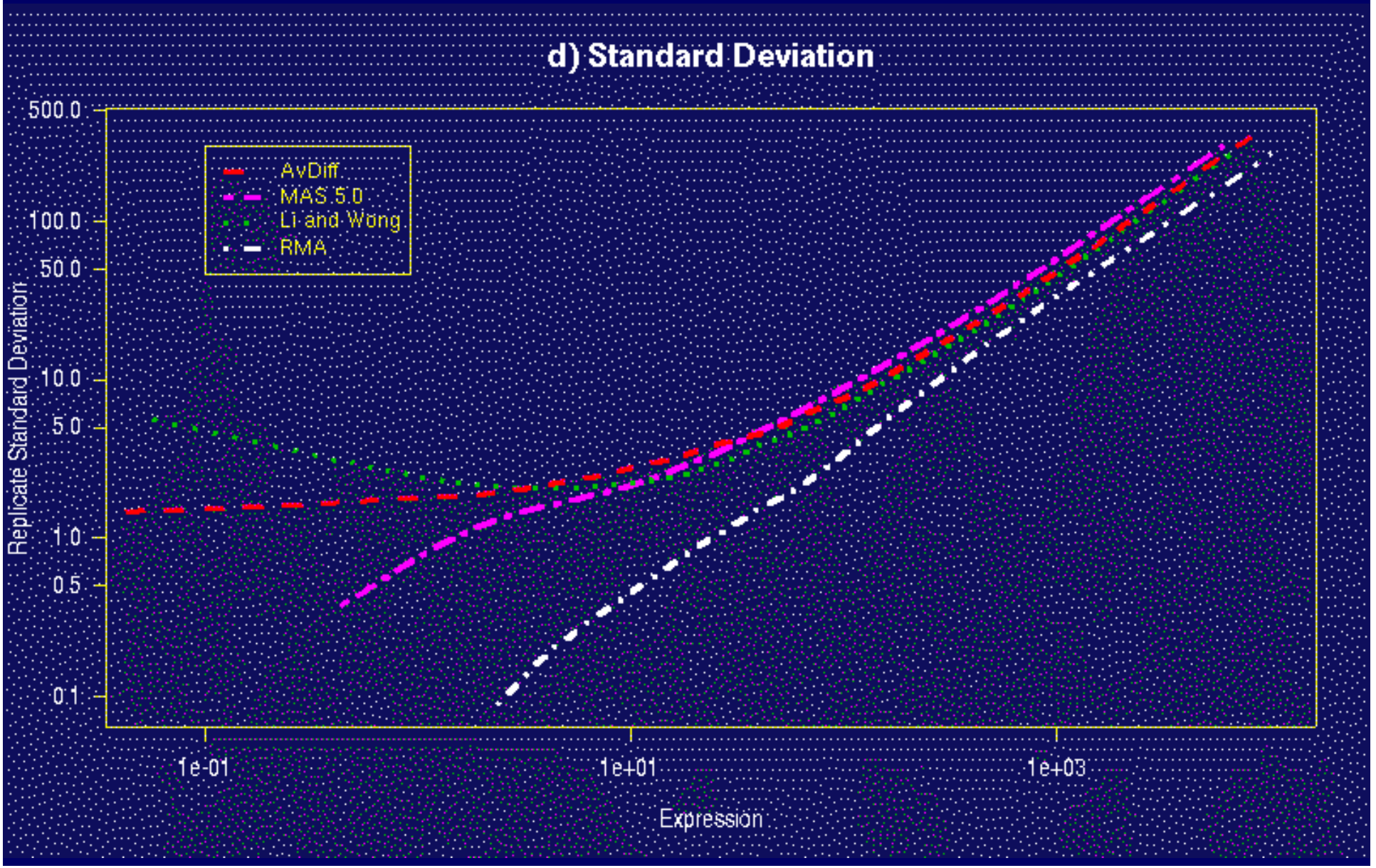
# Dilution experiment data



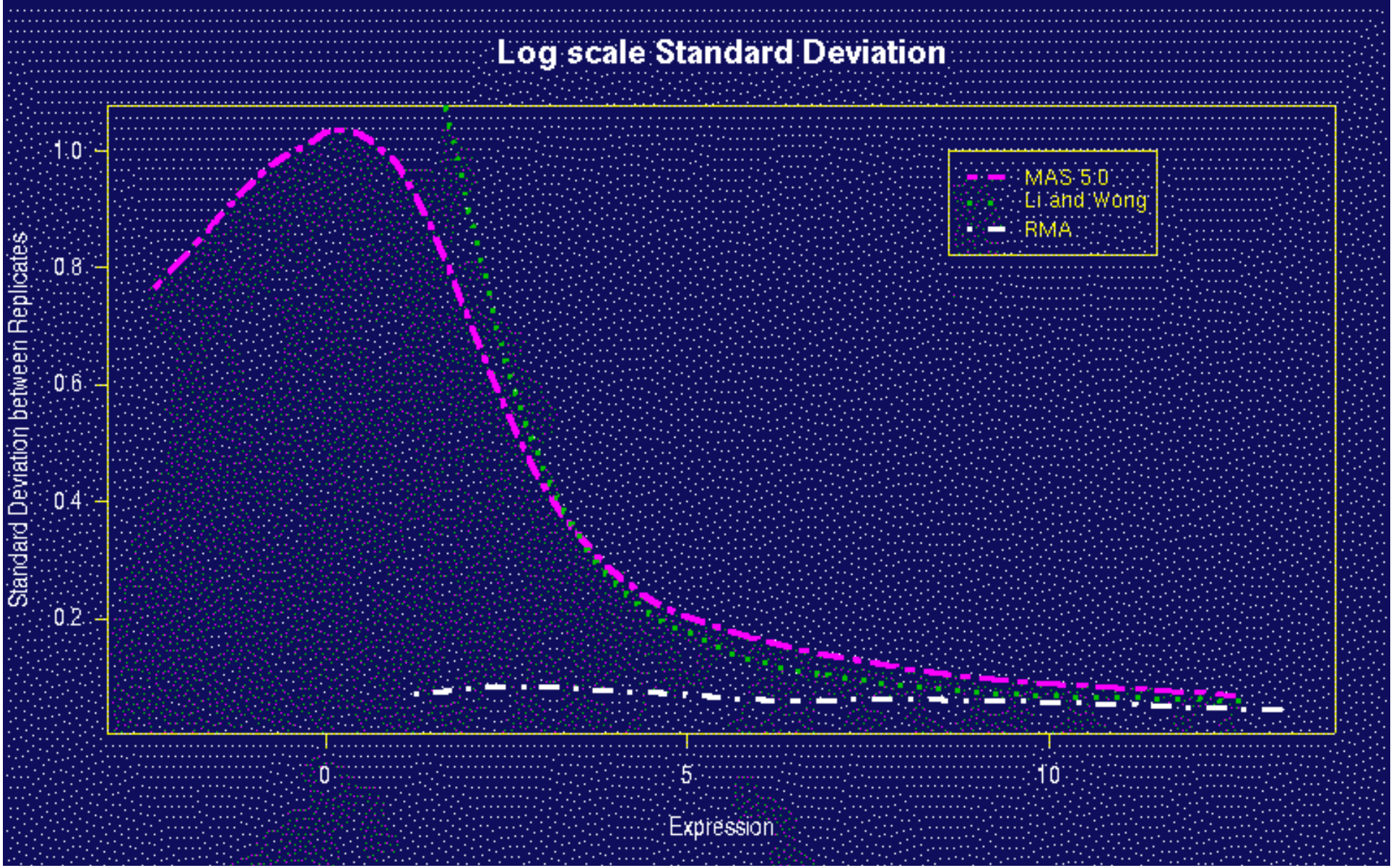
# Expression


Median Expression
AvDiff, slope= 0.64
MAS 5.0, slope= 0.59
Li and Wong, slope= 0.61
RMA, slope= 0.64
Median Expression
100
50
20
10
1.25
2.50
5.00
7.50
10.00
20.00
Concentration

# Standard deviation of expression



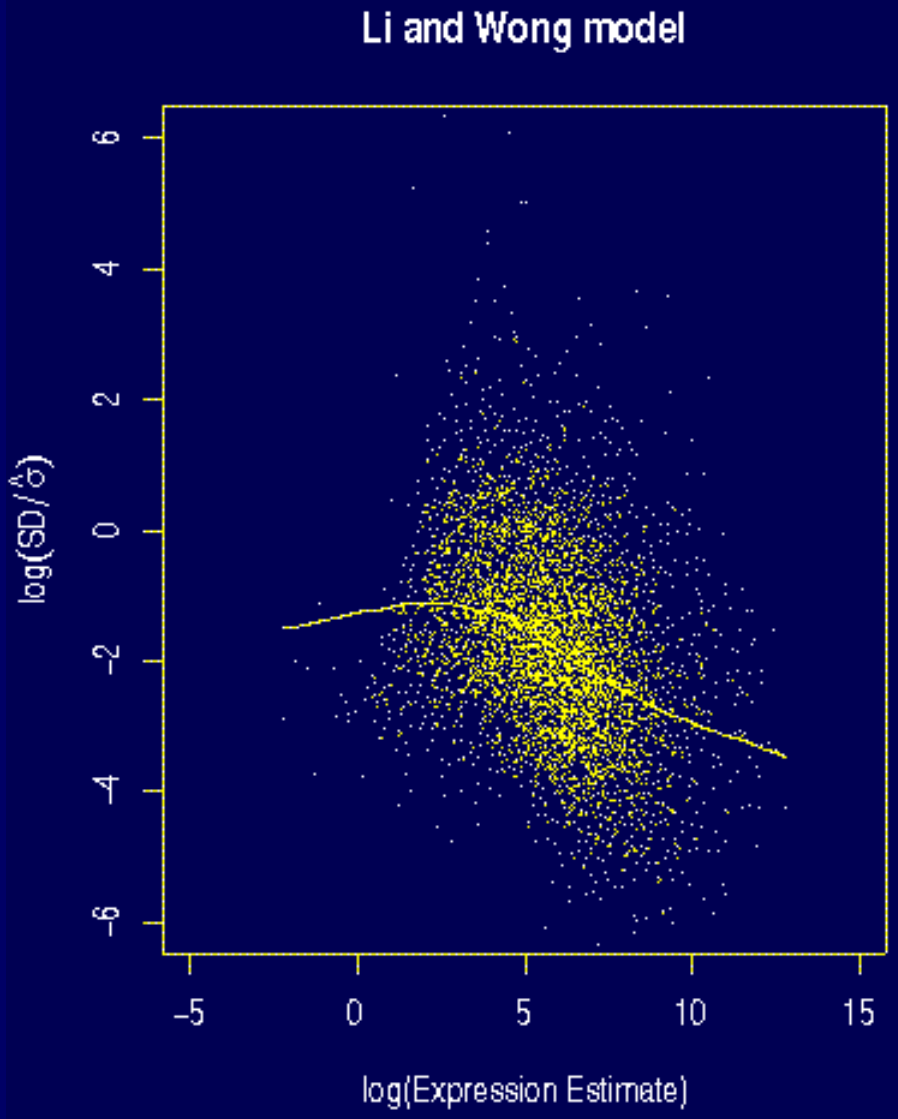
# Standard deviation of expression: log scale

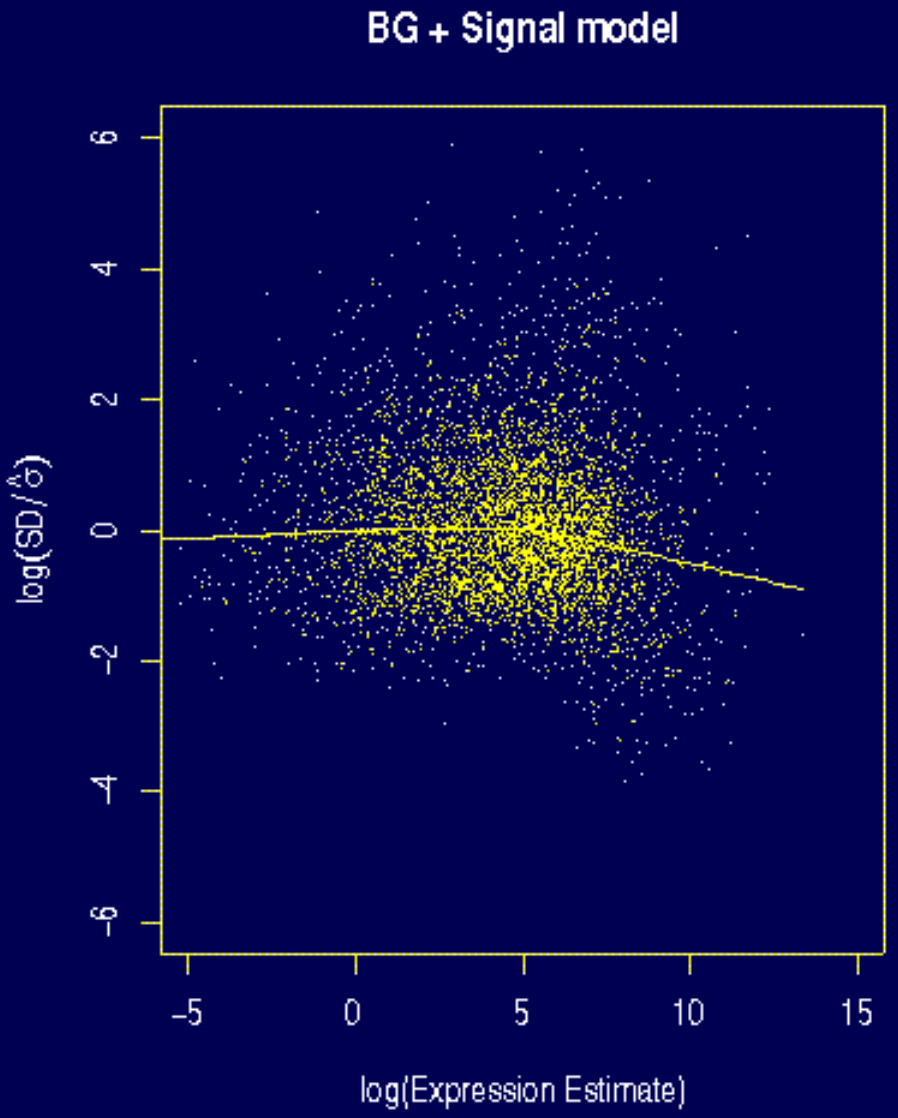


## Model check

- Compute observed SD of 5 replicate expression estimates
- Compute RMS of 5 nominal SDs
- Compare by taking the log ratio
- Closeness of observed and nominal SD taken as a measure of goodness of fit of the model

# Observed vs. Model SE



# The way robustness works

We look at the parameter estimates from a robust two-way analysis, mp = median polish, and a standard linear model = lm, from 37 control probe sets across 6 chips. Overall we have 20x37 probe effects, 6x37 chip effects, and 20x6x38 residuals.

This was repeated for 37 randomly chosen probe sets, to check that the control probe sets were not atypical. They weren't.

After observing certain patterns, an explanation is offered of the way robustness works.

# The two methods

Each fits the same model to probe level data:

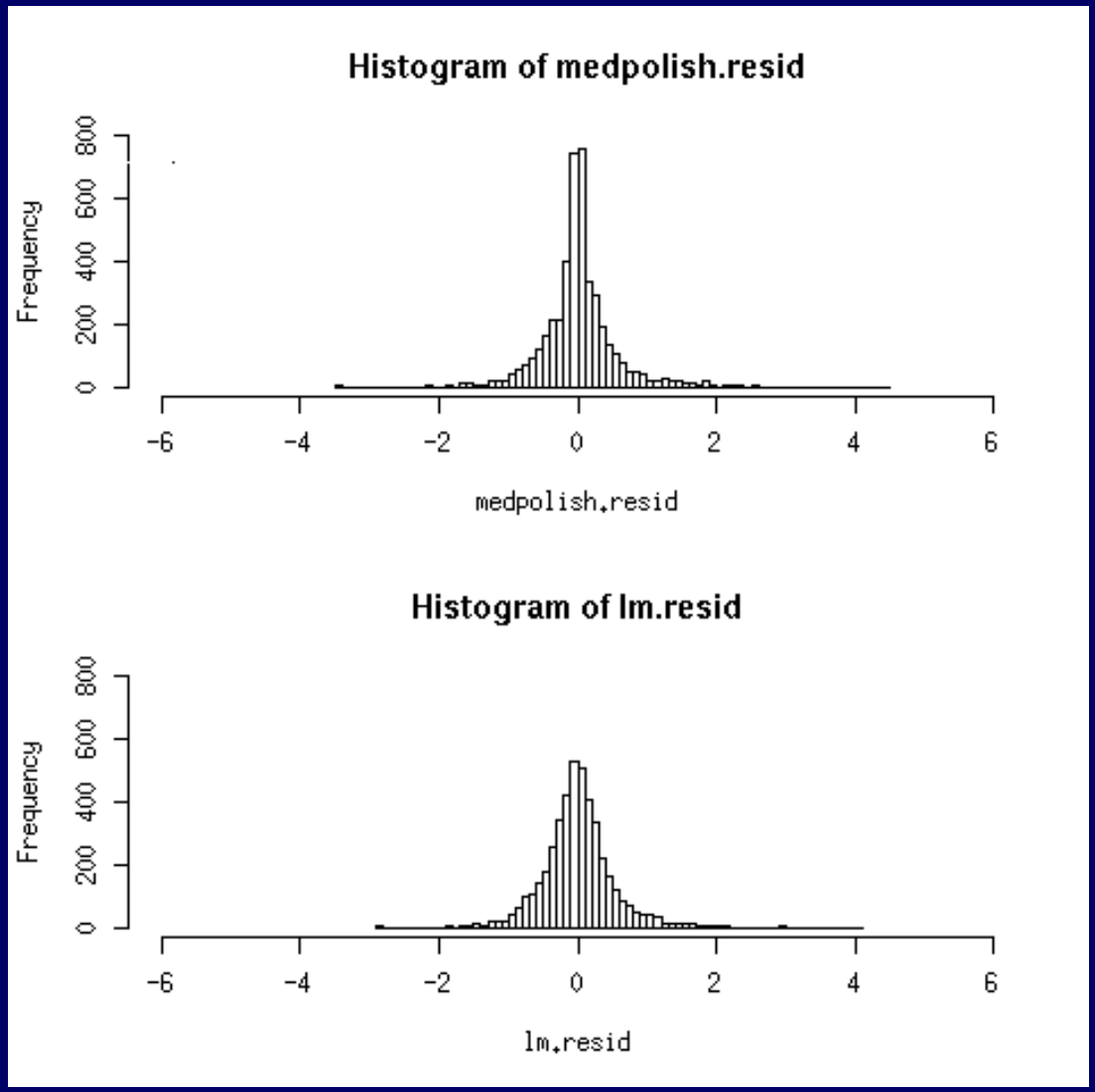
$$\log(PM_{ij} - BG) = chipeffect_i + probeffect_j + residual_{ij}$$

For simplicity, we suppress the probe set index, denoted earlier by n .

Median polish (mp) fits iteratively, successively removing row and column medians, and accumulating the terms, until the process stabilizes. The residuals are what is left at the end.

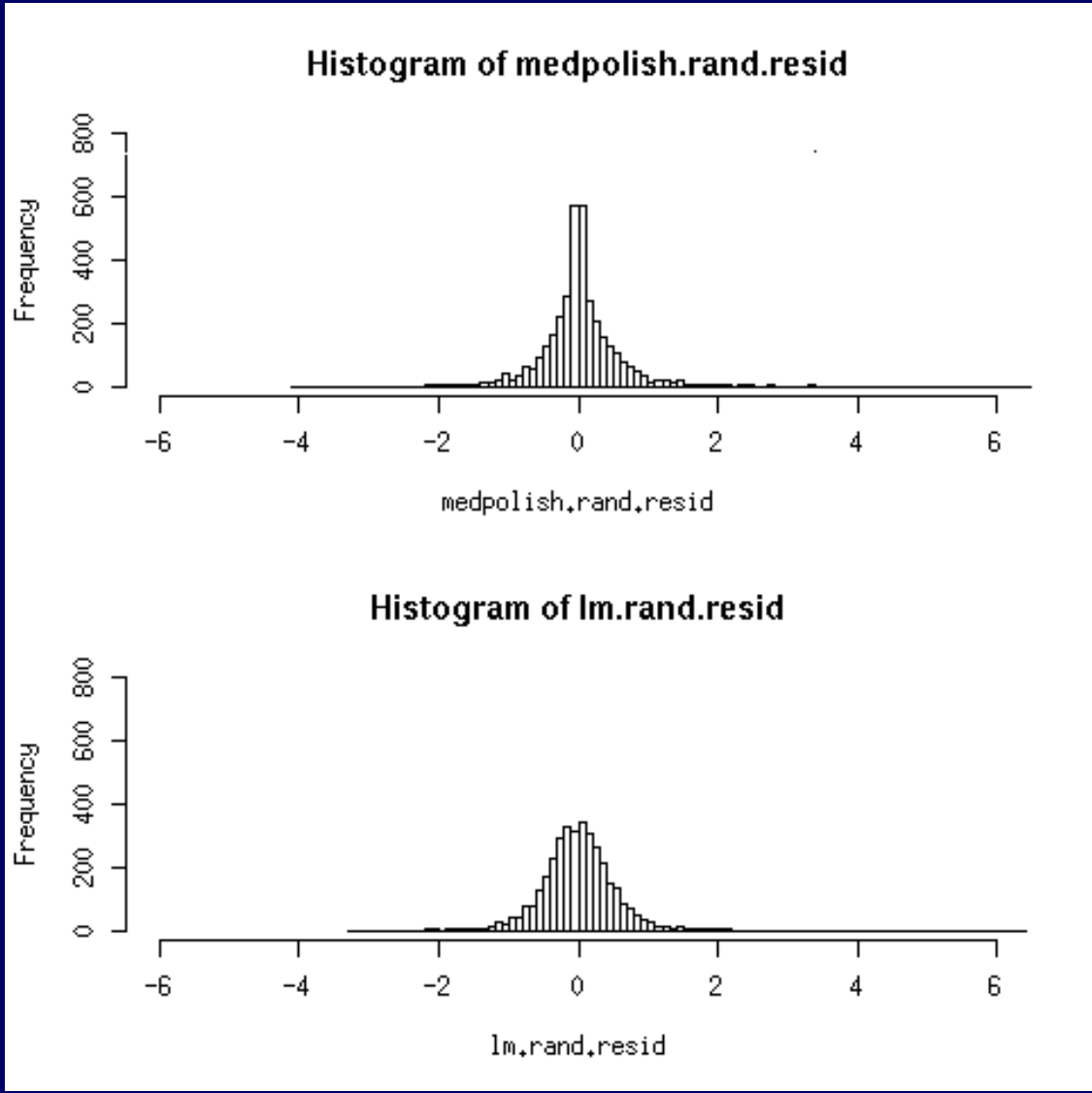
The standard linear model uses the familiar closed-form estimates of the parameters a and b (what are they?), and again the residuals are what is left after subtracting them from the observations.

# Two sets of residuals (control probes)

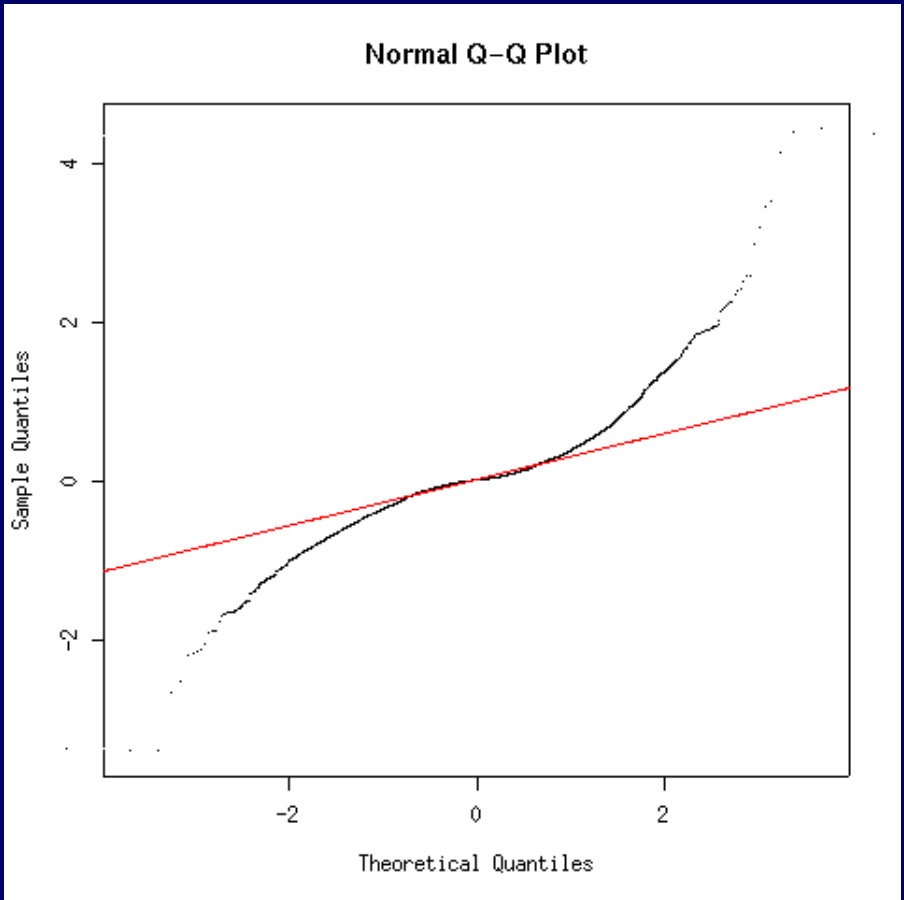


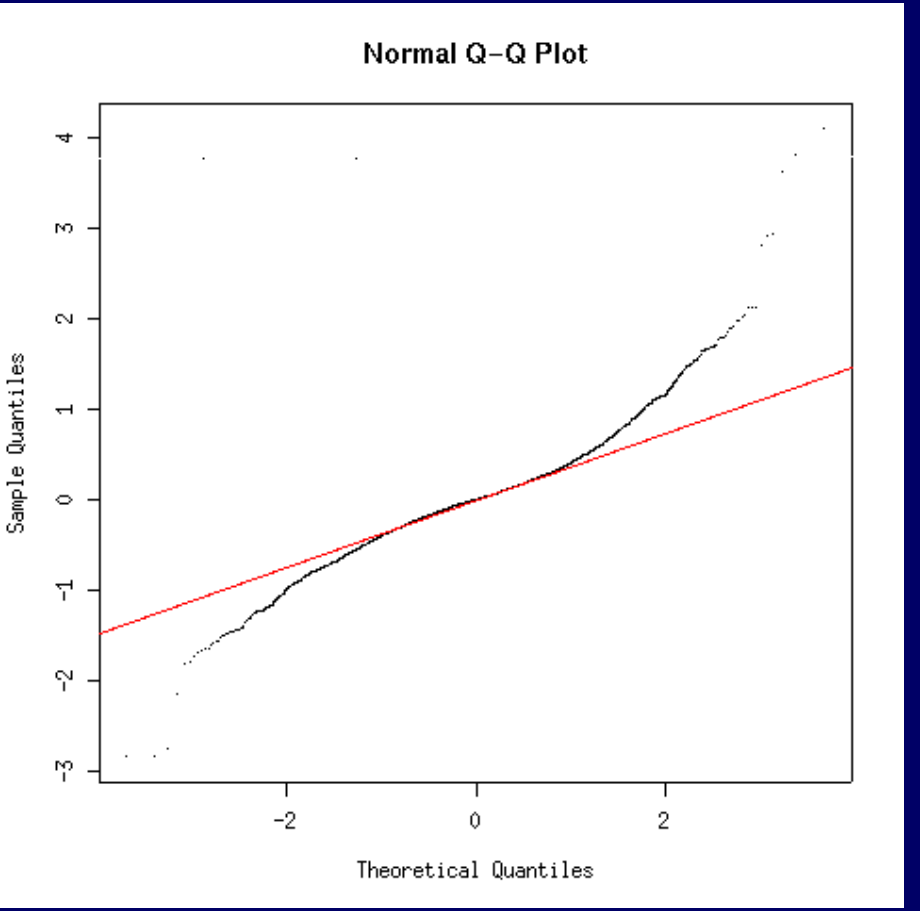

**Note the  slightly different shapes of their distributions**

# Similarly with random probe sets

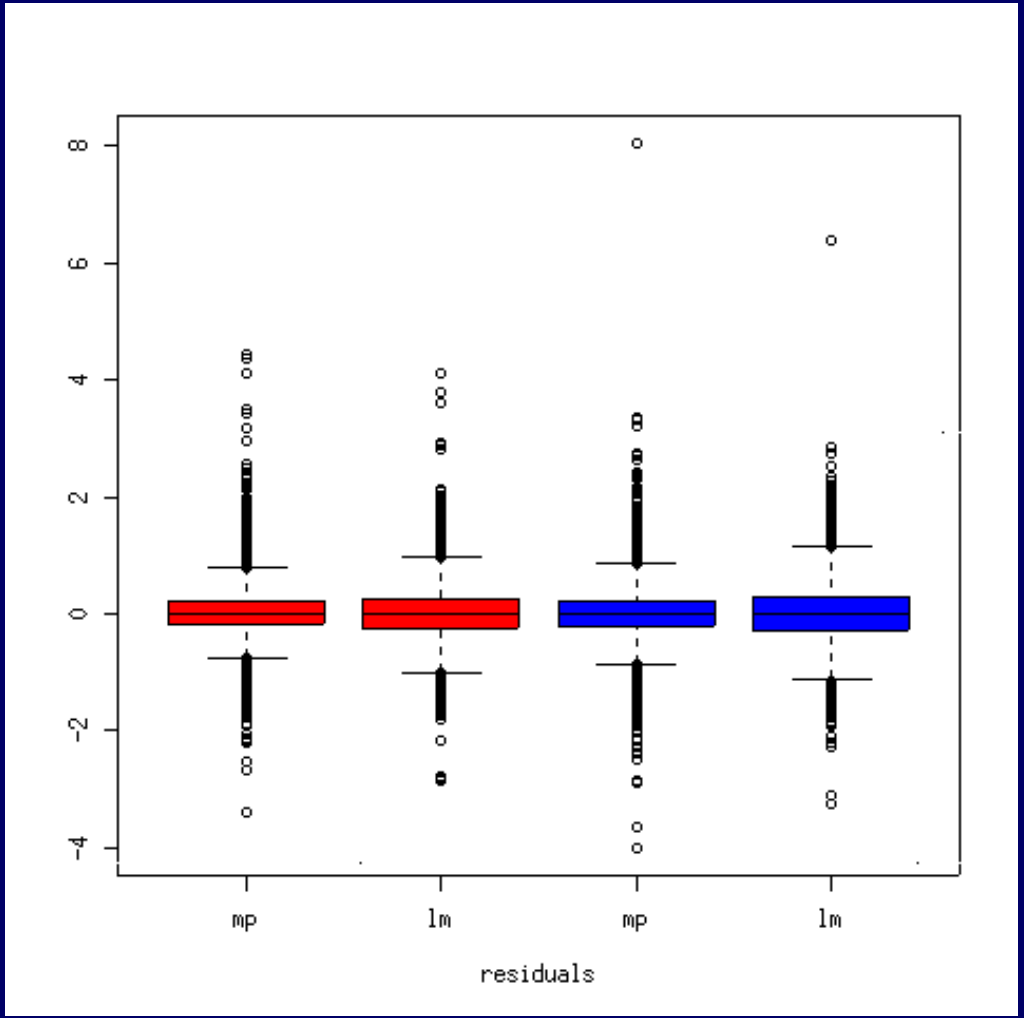


# Normal qq plots of mp and lm residuals



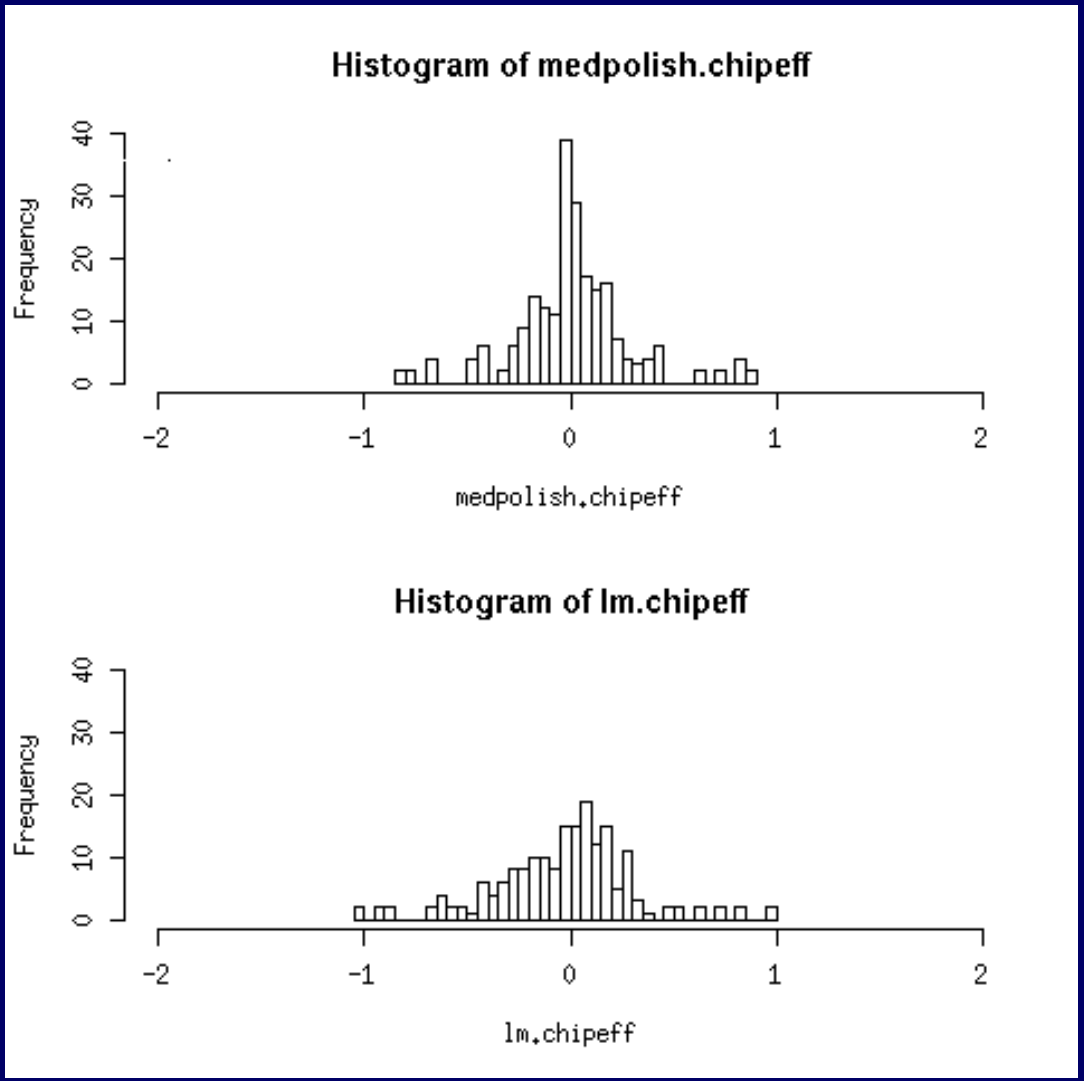




**Which has (slightly) fatter tails?**



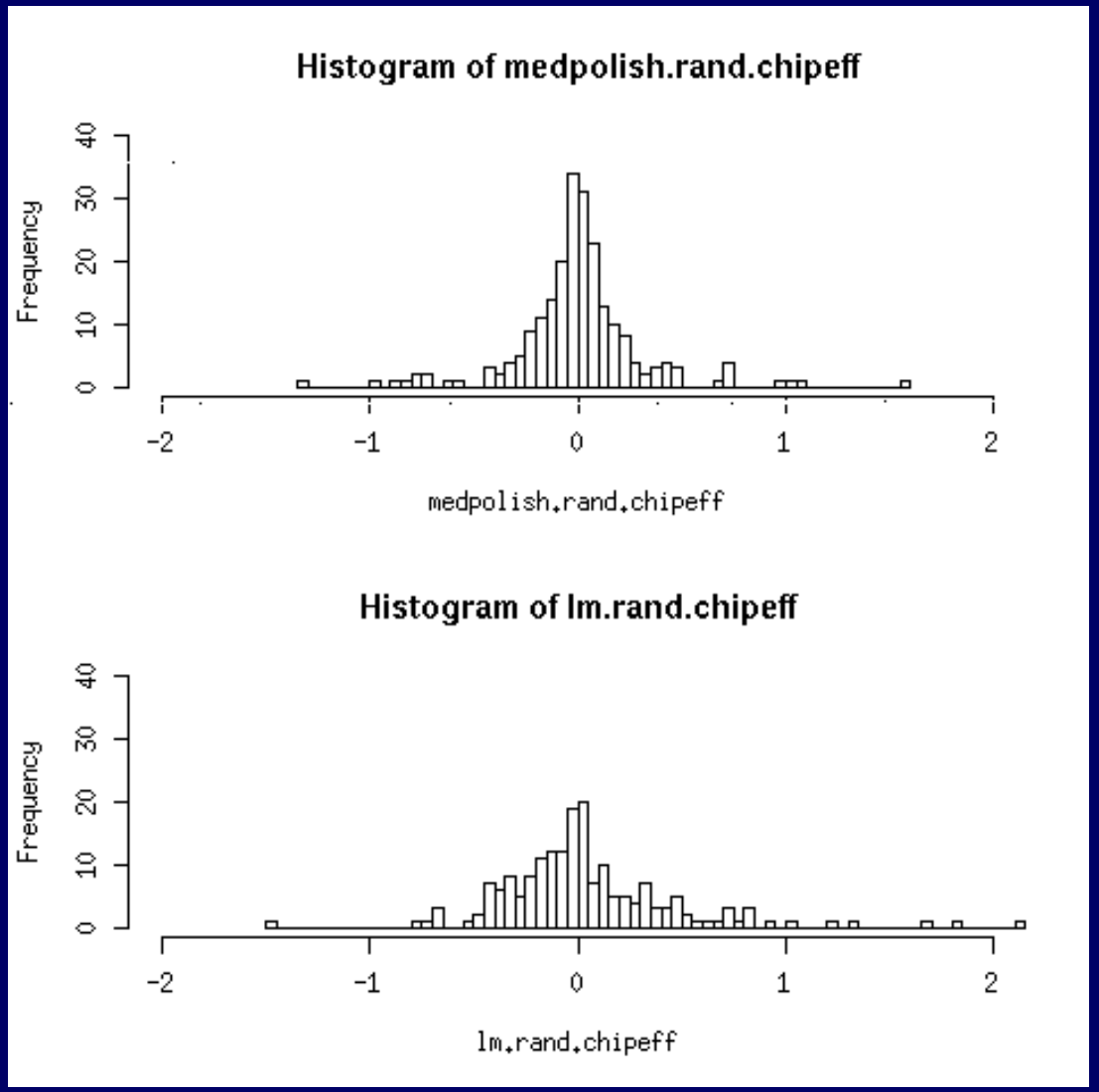

**mp residuals have fatter tails than lm, but smaller IQRs**

# Chip effects: 37 control probe sets, 6 chips



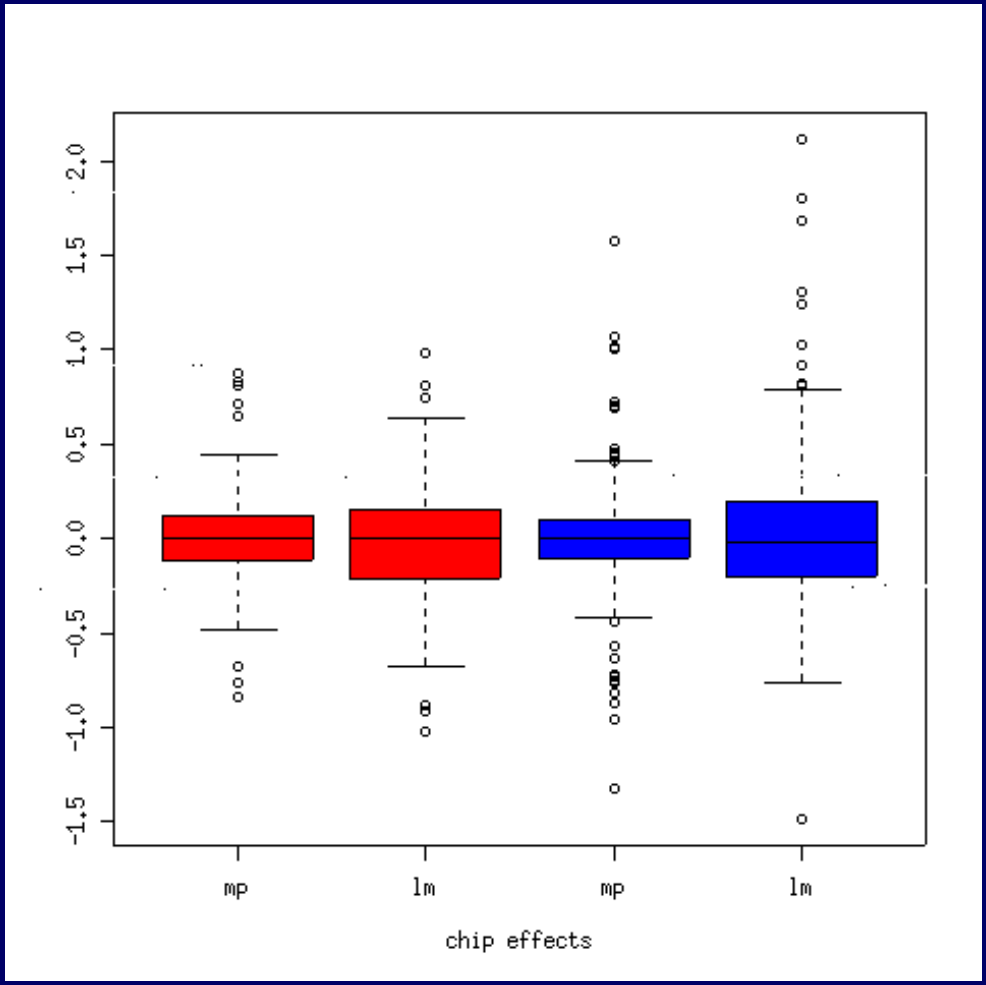

mp chip effects somewhat more concentrated than lm's

# Chip effects: 37 random probe sets, 6 chips



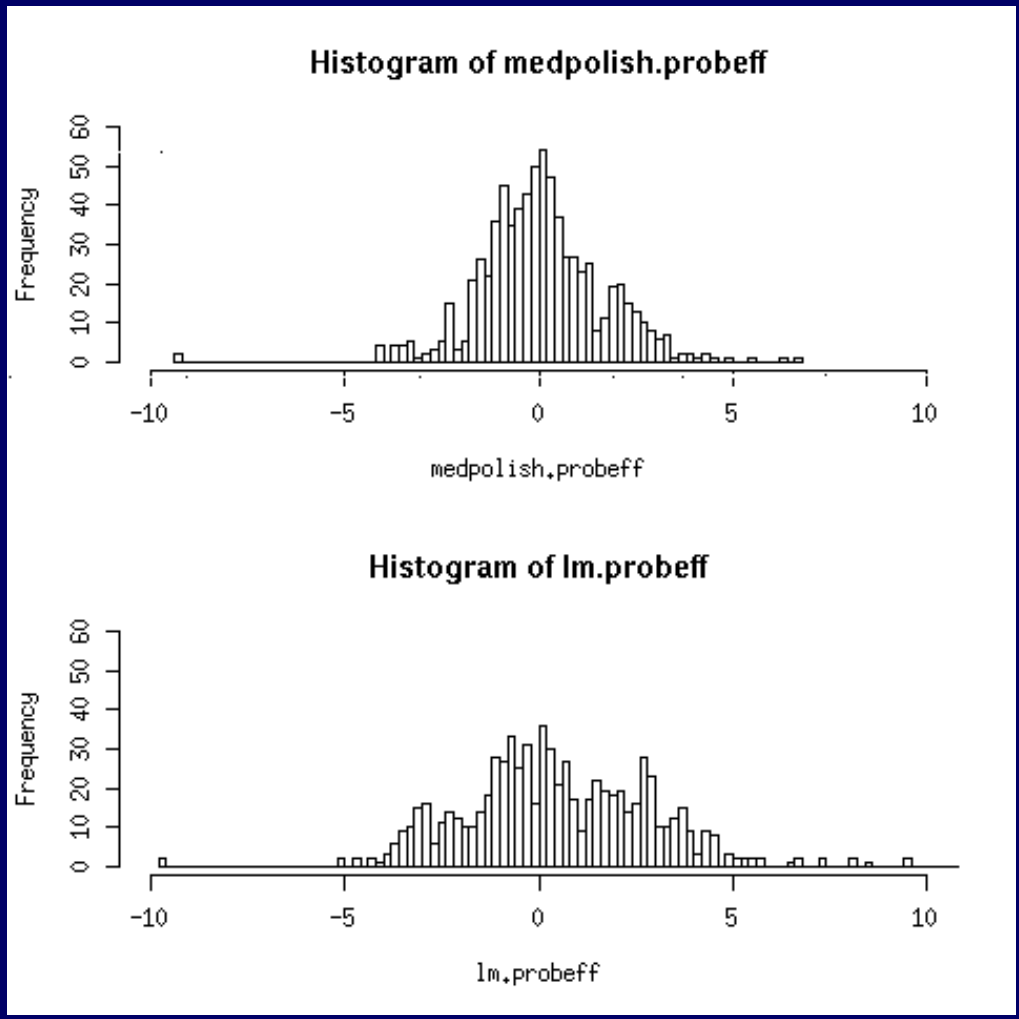

**The same holds with random probe sets**

## Chip effects: control probe sets left, random right median polish (mp) first, linear model (lm) second



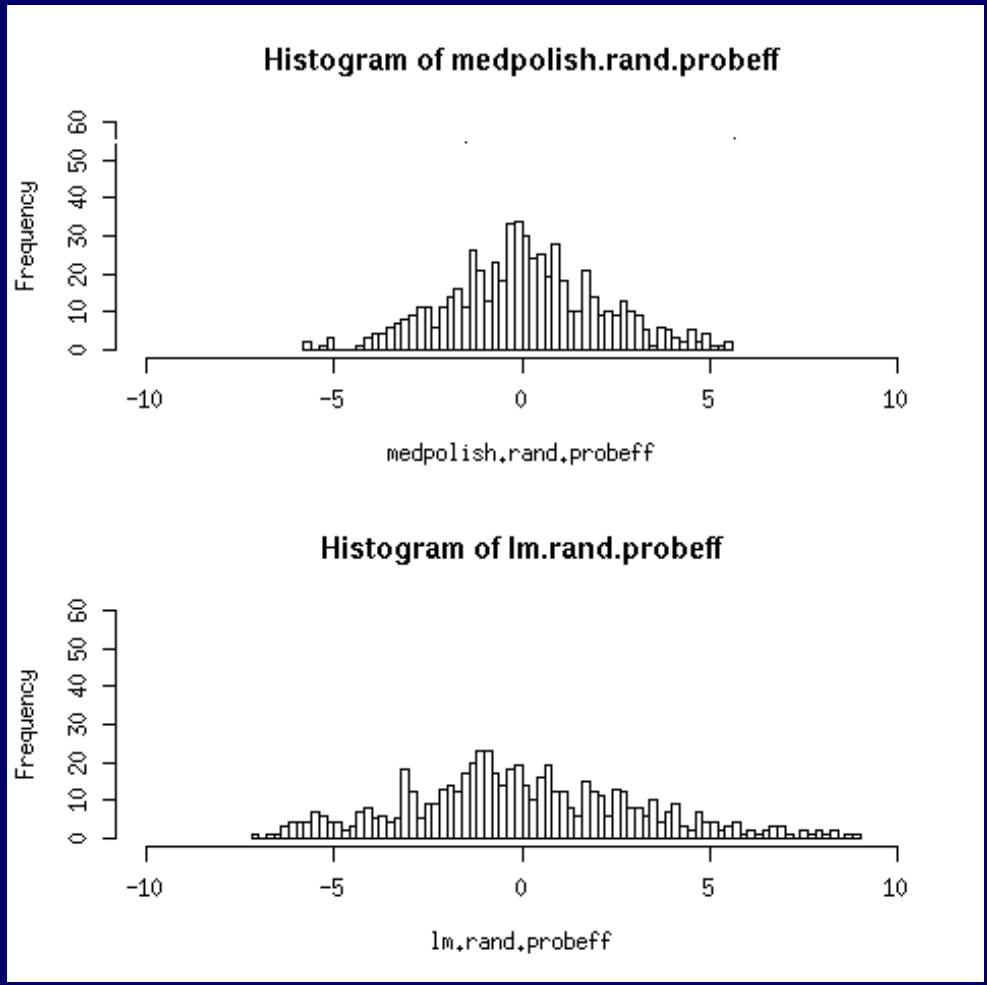

## lm has more outlier chip effects than mp, and larger IQR

# Probe effects: 37x20 control probes



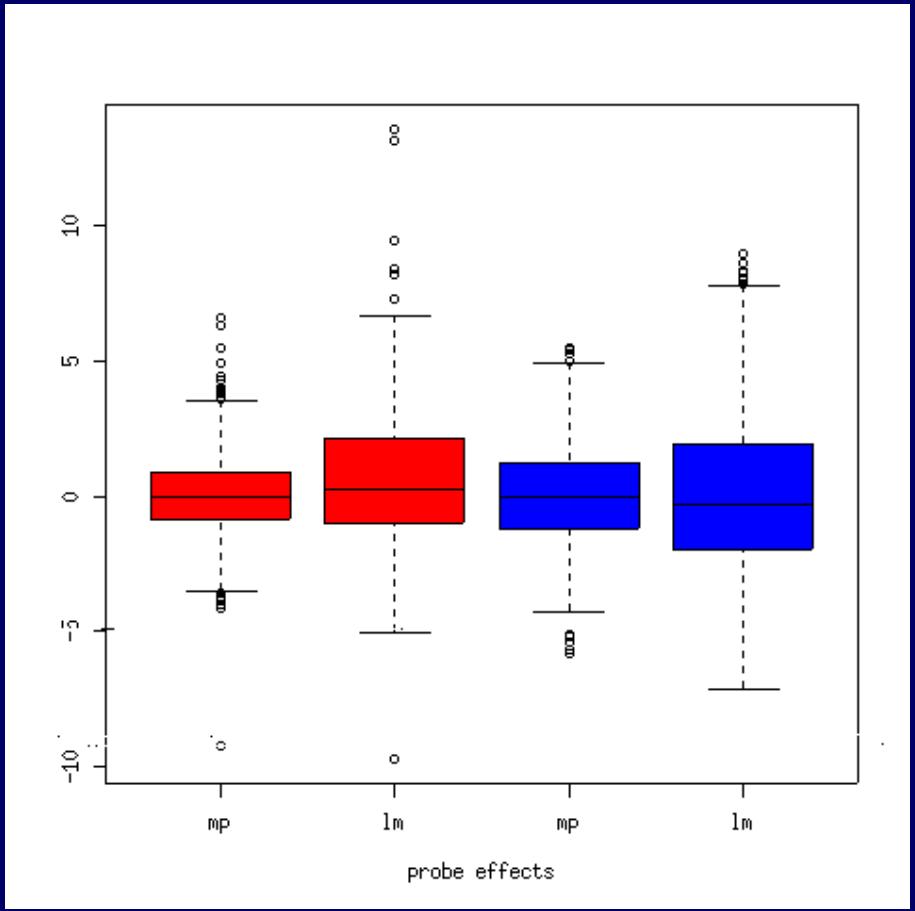

**Median polish probe effects also more concentrated**

# Probe effects: 37x20 random probes




**And again, with random probe sets**

## Probe effects: control probe sets left, random right median polish (mp) first, linear model (lm) second




## Here too lm has more outlier probe effects than mp, and a larger IQR

# A possible explanation of the better performance of mp over lm

- mp does not *adapt* to but essentially *ignores* outlier probe-level values. Part evidence for this can be found its distribution of residuals (fatter tails, smaller IQR).
- By contrast, lm tries to *accommodate* outlier probe level values. We see a more normal distribution of residuals.
- By ignoring outlier probe level values, mp does not let them affect its estimates of chip and probe parameters.
- By accomodating outlier probe level values, lm changes its chip and probe parameter estimates.
- We see the difference in the shapes of the distributions of chip and probe parameters.
- Which is better? Well, mp performs better. Is this why?

## What did we learn?

- Do not subtract or divide by MM
- Take logs
- Correct for background, never going <0
- Normalize, in groups
- Probe effect is additive on log scale
- Be robust, in groups

# Conclusion

- Use normalized $\log_2$(PM-BG)
- Using global background improves on use of probe-specific MM
- Gene Logic spike-in and dilution study show technology works well
- RMA is arguably the best summary in terms of bias, variance and model fit
- Future: What statistic should we use to rank?

# Acknowledgements


- Gene Brown's group at Wyeth/Genetics Institute, and Uwe Scherf's Genomics Research & Development Group at Gene Logic, for generating the spike-in and dilution data
- Gene Logic for permission to use these data
- Francois Collin, GeneLogic
- Rafael Irizarry and colleagues, JHU
- Bridget Hobbs, WEHI
- Magnus Åstrand (Astra Zeneca Mölndal)