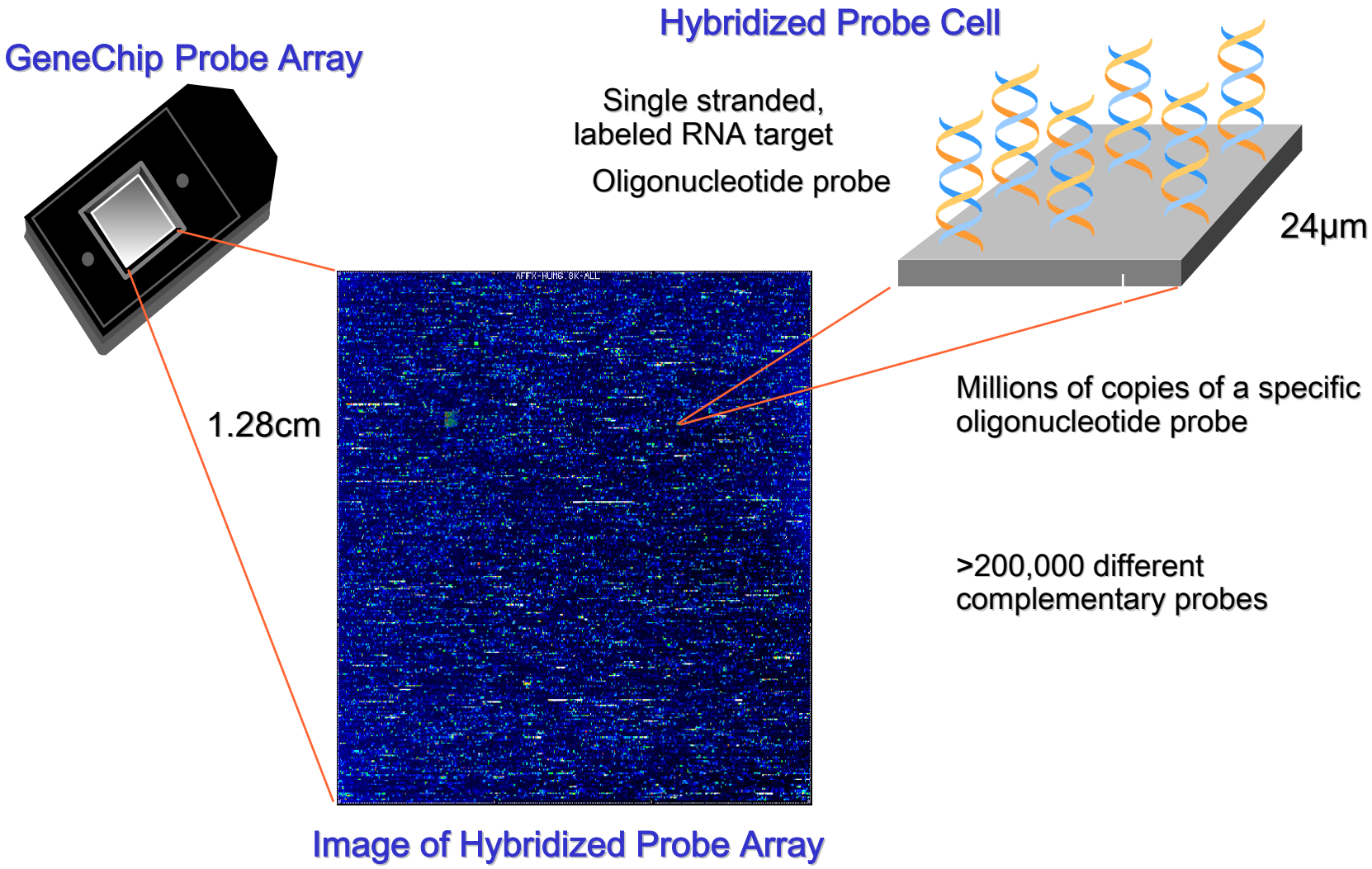
Part II. Pre-processing: Affymetrix GeneChip arrays

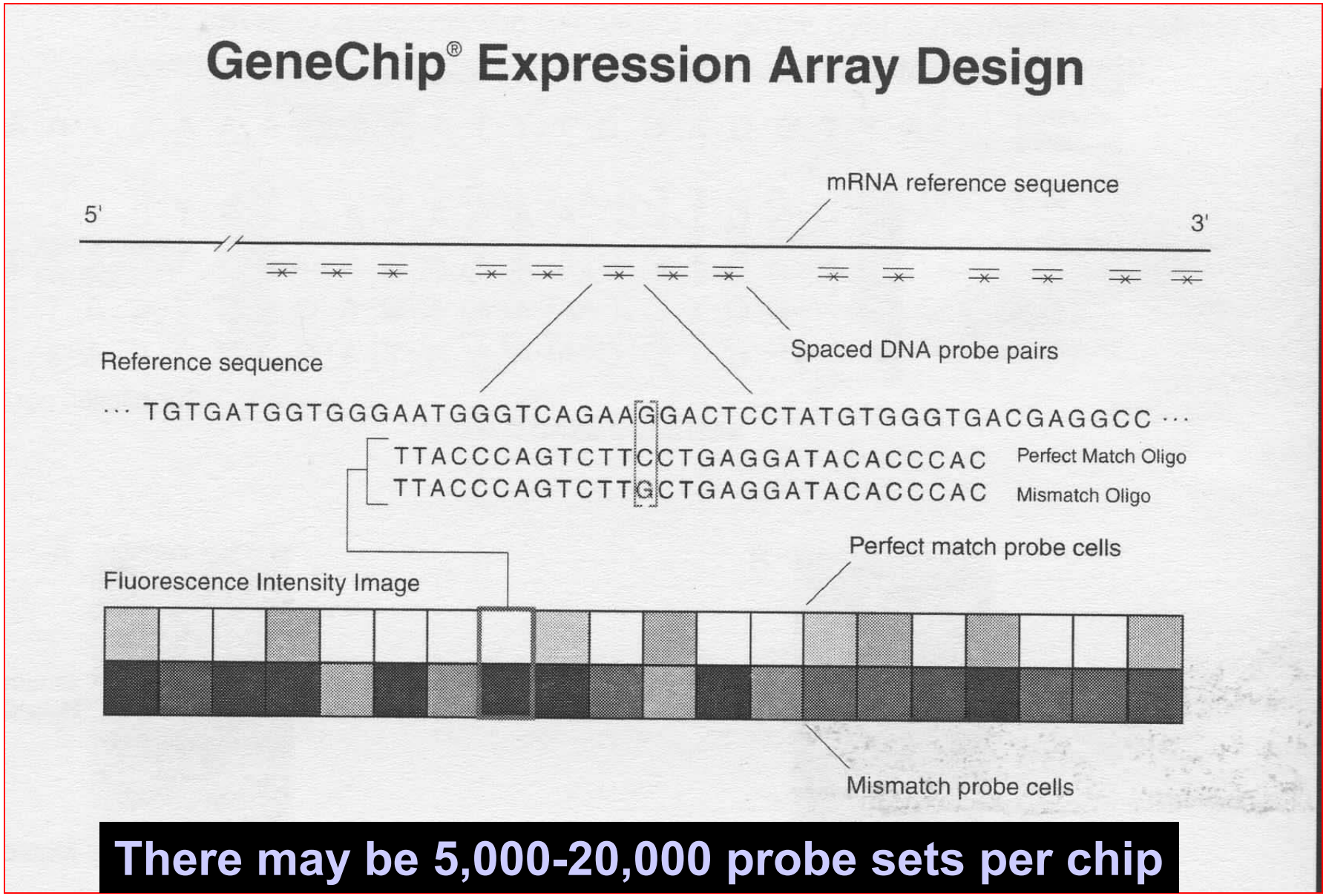
Rafael Irizarry

Affymetrix GeneChip Arrays



Compliments of D. Gerhold

A probe set = 11-20 PM,MM pairs



There may be 5,000-20,000 probe sets per chip

Affymetrix files

- Main software from Affymetrix company MAS - MicroArray Suite, now version 5.
- **DAT** file: Image file, ~10^7 pixels, ~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 (genes, gene fragments, ESTs).

Image analysis

- Raw data, **DAT image files → CEL files**
- Each probe cell: 10x10 pixels.
- **Gridding**: estimate location of probe cell centers.
- **Signal**:
  - Remove outer 36 pixels → 8x8 pixels.
  - The probe cell signal, PM or MM, is the 75th percentile of the 8x8 pixel values.
- **Background**: Average of the lowest 2% probe cell values is taken as the background value and subtracted.
- Compute also quality measures.

Data and notation

- PM_{ijg}, MM_{ijg} = Intensity for perfect match and mismatch probe in cell *j* for gene *g* in chip *i*.
  - $i = 1, \ldots, n$ from one to hundreds of chips,
  - $j = 1, \ldots, J$ usually 11 or 20 probe pairs,
  - $g = 1, \ldots, G$ between 8,000 and 20,000 probe sets.
- Task: summarize for each probe set the probe level data, i.e., 20 PM and MM pairs, into a single expression measure.
- Expression measures may then be compared within or between chips for detecting differential expression.

Summarizing 11-20 probe intensity pairs to give a measure of expression for a probe set

There are many possible low-level summaries, but they fall into four main classes defined by combinations of two categories

1. Single or multi chip
2. Linear or log scale

Competing measures of expression, 1

- The original GeneChip® software used *AvDiff*

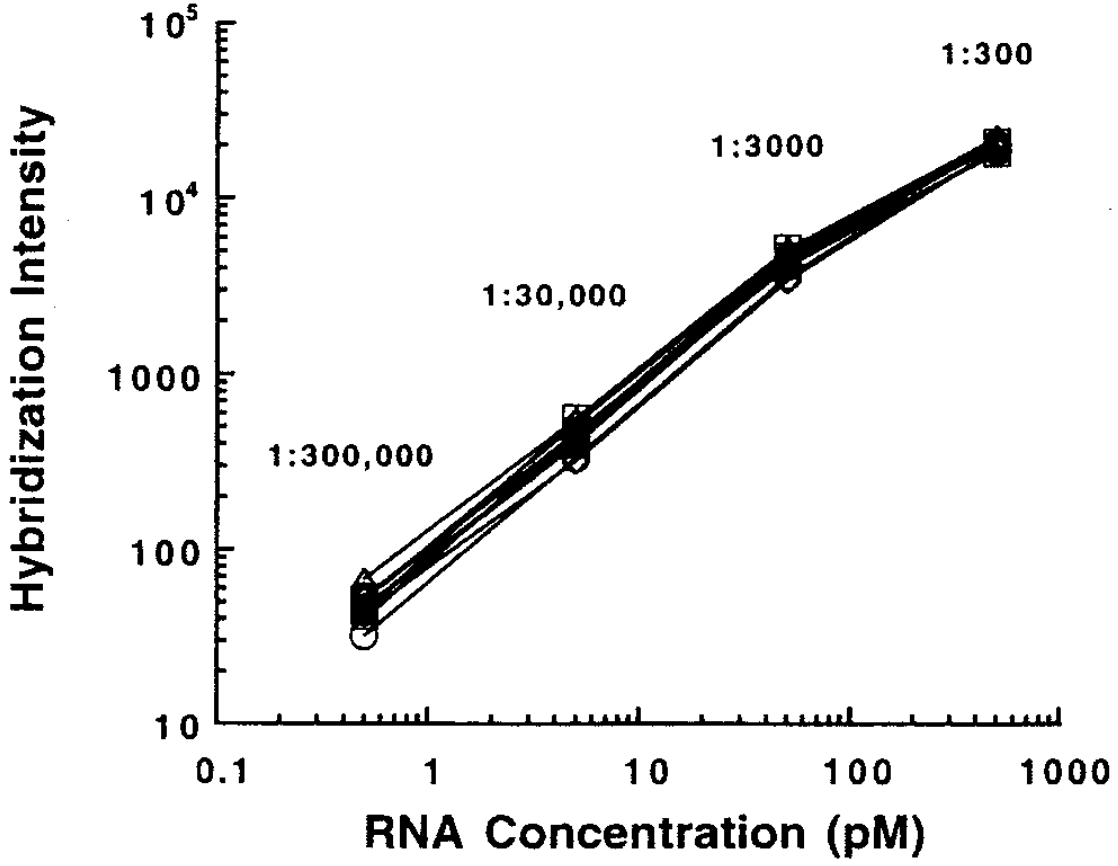
$$AvDiff = |A|^{-1} \sum_{\{j \in A\}} (PM_j - MM_j)$$

where A is a suitable set of pairs chosen by the software. Here 30%-40-% could be <0, which was a major irritant.

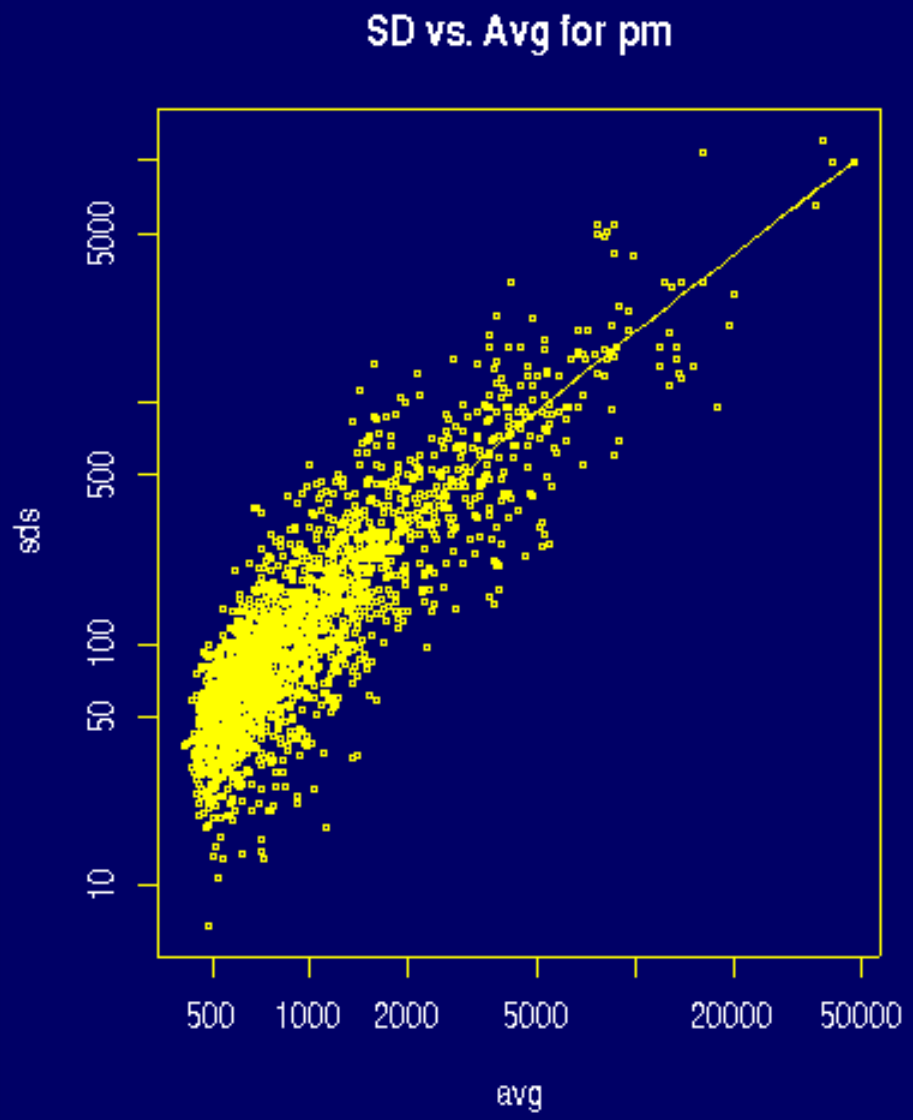
- *Log* PM_j / MM_j was also used in the above.

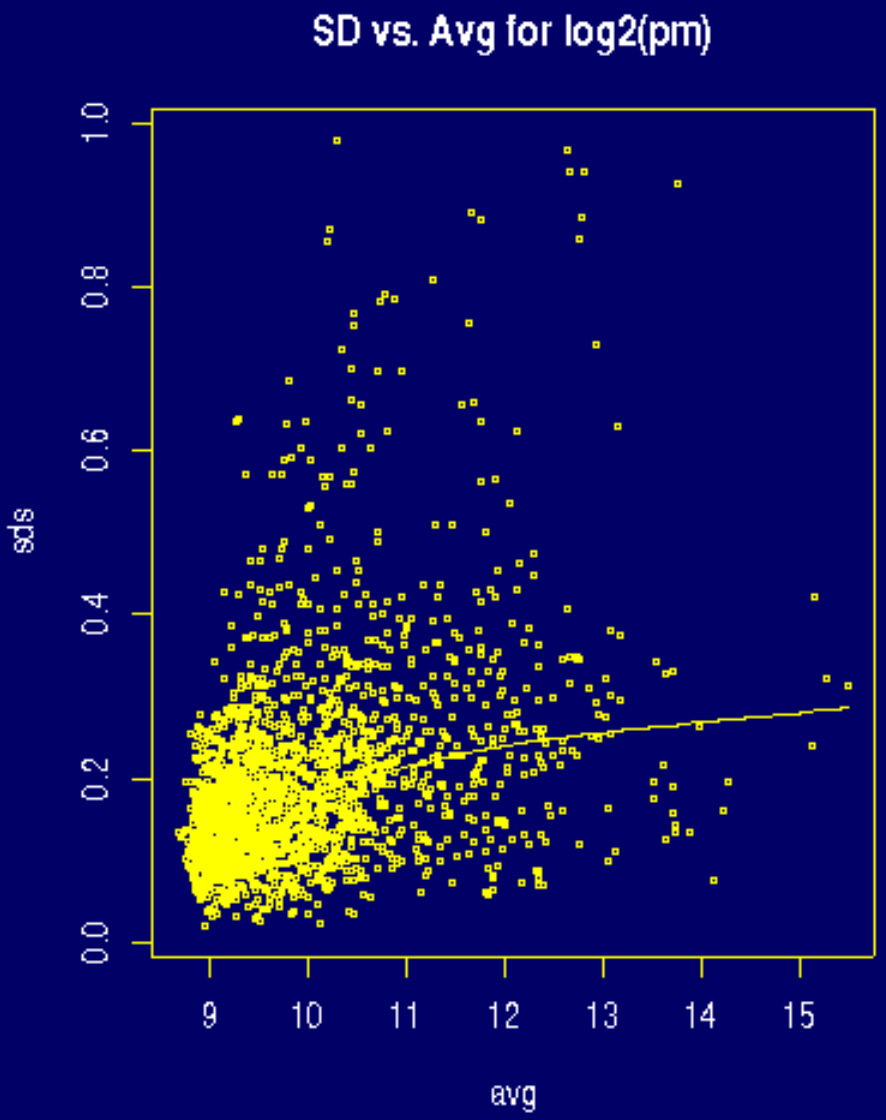
What is the evidence?

Lockhart et. al. Nature Biotechnology 14 (1996)



Why we take $\log_2$





Competing measures of expression, 2

- The latest version of GeneChip® uses something else, namely

$$Log\{Signal\ Intensity\} = TukeyBiweight\{log(PM_j - MM_j^*)\}$$

with MM_j^* a version of MM_j that is never bigger than PM_j. Here *TukeyBiweight* can be regarded as a kind of robust/resistant mean.

Competing measures of expression, 3

- Li and Wong (dChip) fit the following model to sets of chips

$$PM_{ij} - MM_{ij} = \theta_i \varphi_j + \varepsilon_{ij}$$

where $\varepsilon_{ij} \sim N(0, \sigma^2)$. They consider θ_i to be expression in chip *i*. Their model is also fitted to PM only, or to both PM and MM. Note that by taking logs, assuming the LHS is >0, this is close to an additive model.

Competing measures of expression, 4

What we do: four steps

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

- Adjust for **background** on the raw intensity scale;
- Carry out **quantile normalization** of PM-*BG with chips in suitable sets, and call the result n(PM-*BG);
- Take **$\log_2$** of normalized background adjusted PM;
- Carry out a **robust multi-chip** analysis (RMA) of the quantities $\log_2 n(\text{PM-*BG})$.

We call our approach RMA.

To obtain expression measures probe intensities (PM) are summarized after:

- Adjusting for background and non-specific binding. For example subtracting MM
- Normalization. For example, Affy's scaling
  Some expression measures normalize after summarizing

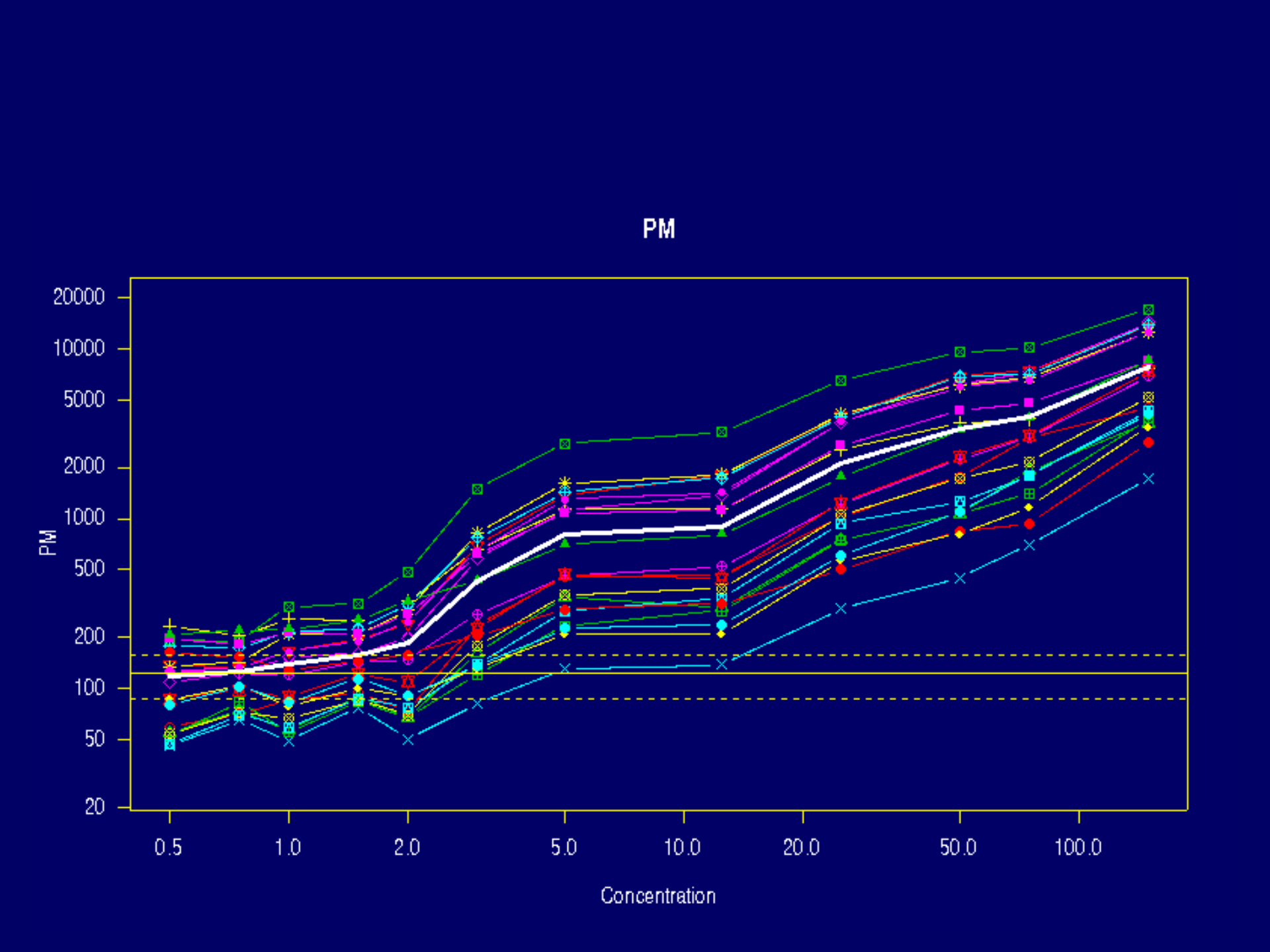
Why we ignore the MM values

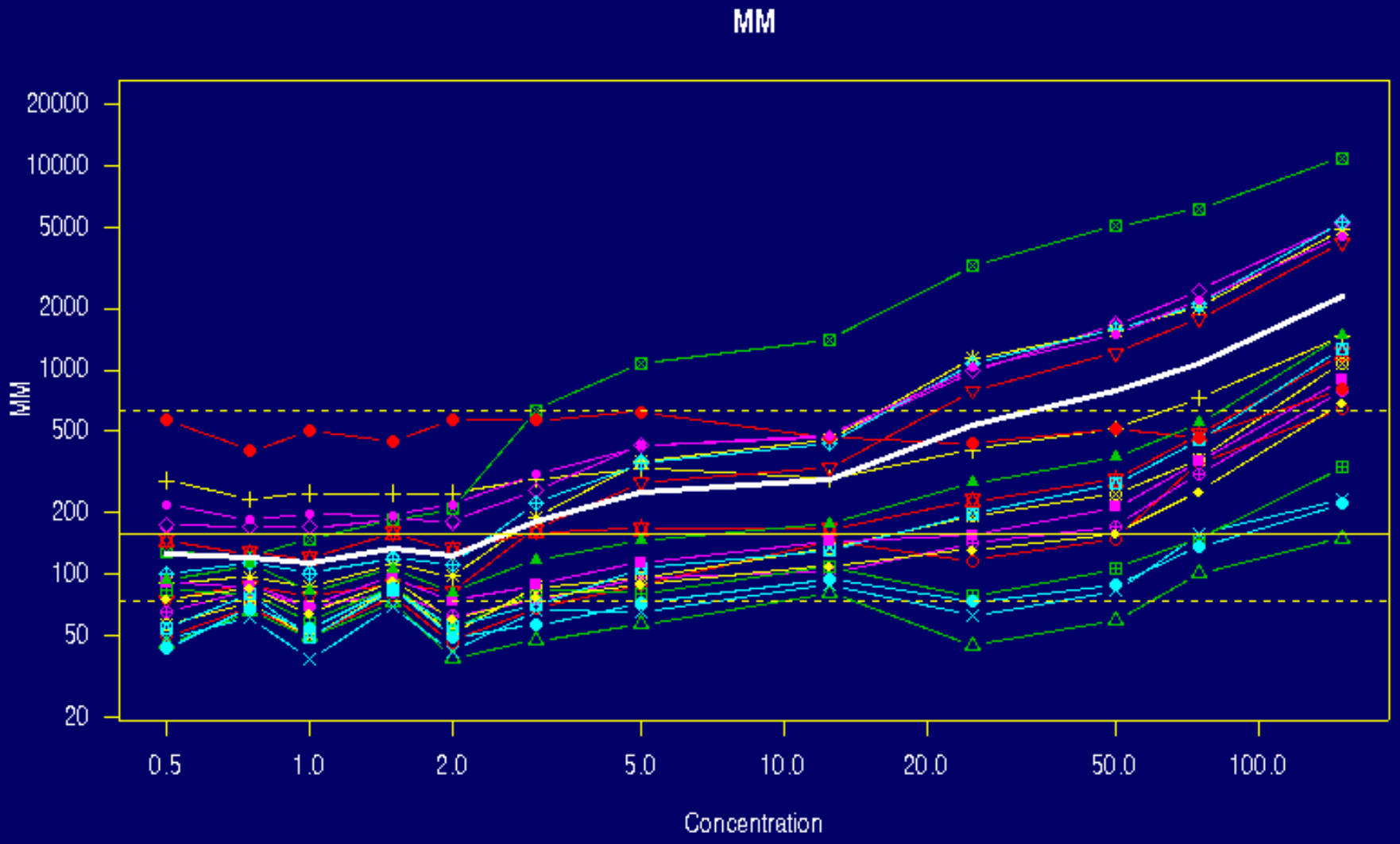
We haven't yet found a way to use them that is better than what we currently do. They definitely provide information, about both signal and noise, but using it without adding more noise (see below) seems to be a challenge.

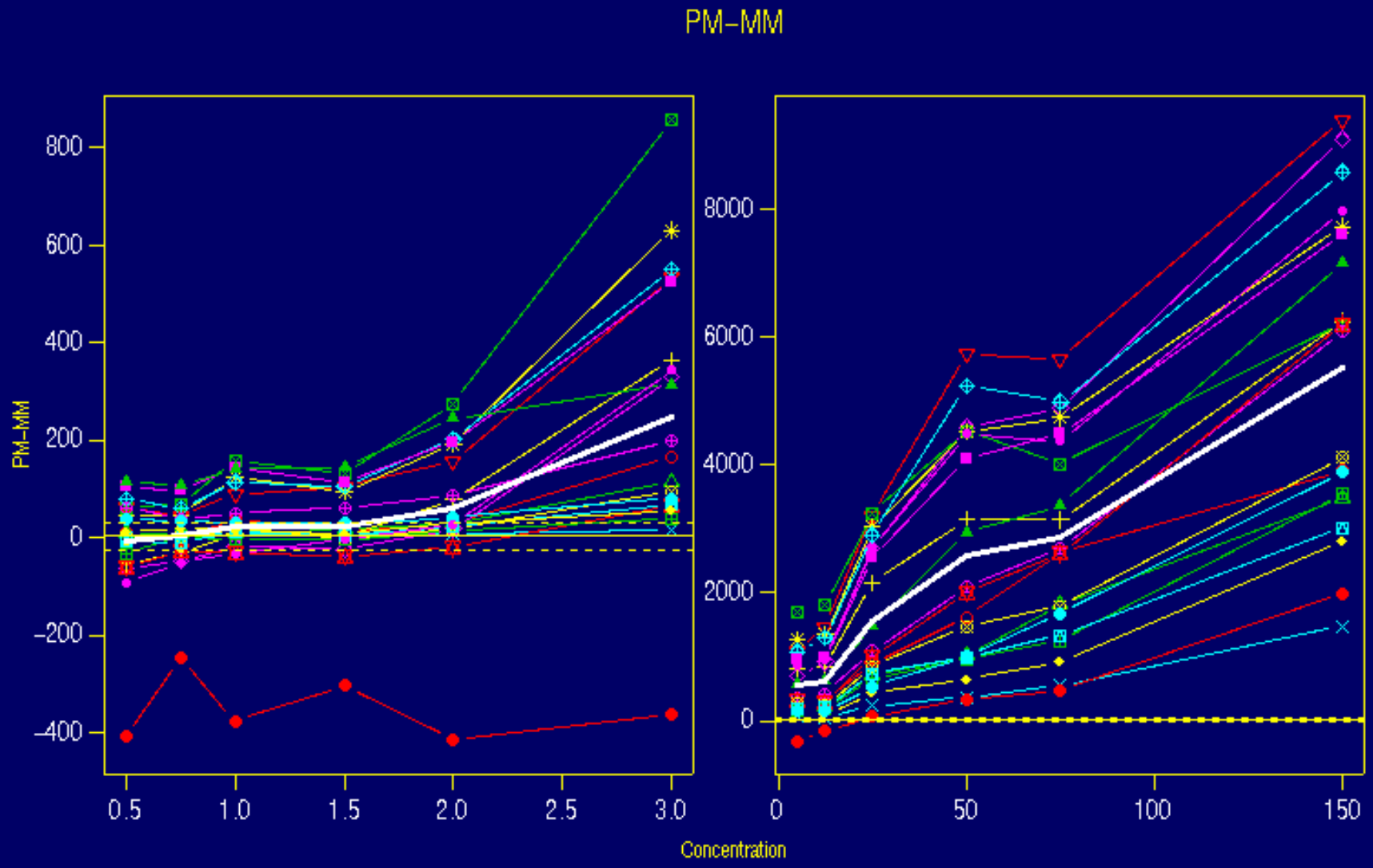
We should be able to improve the BG correction using MM, without having the noise level blow up: work in progress.

Data used for next few slides

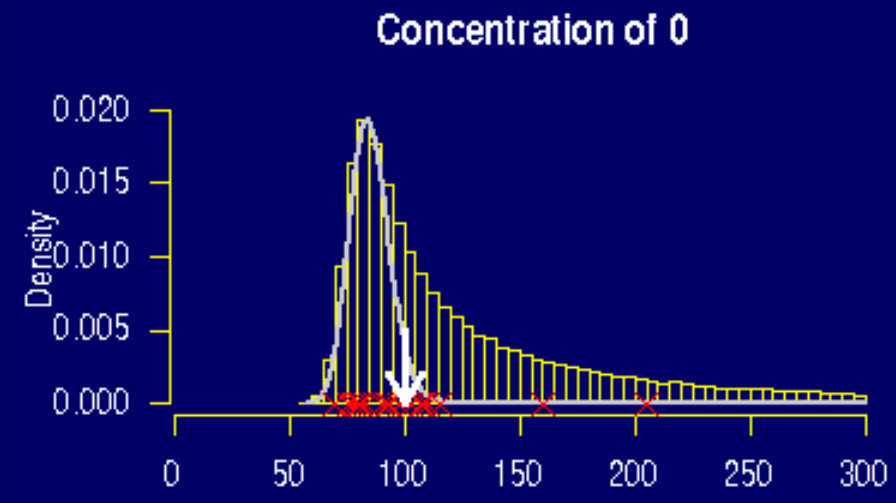
Spike in data set A: 11 control cRNAs spiked in, all at the same concentration, which varies across 12 chips.

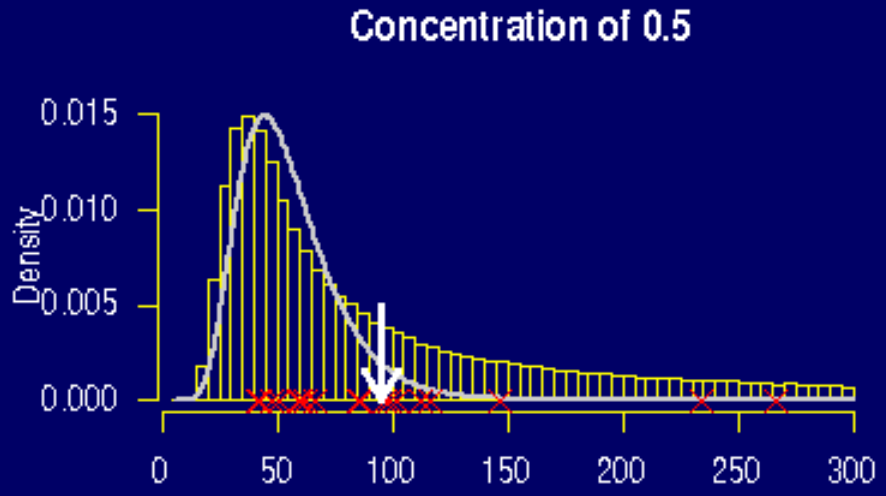

PM
20000
10000
5000
2000
1000
500
200
100
50
20
PM
0.5
1.0
2.0
5.0
10.0
20.0
50.0
100.0
Concentration

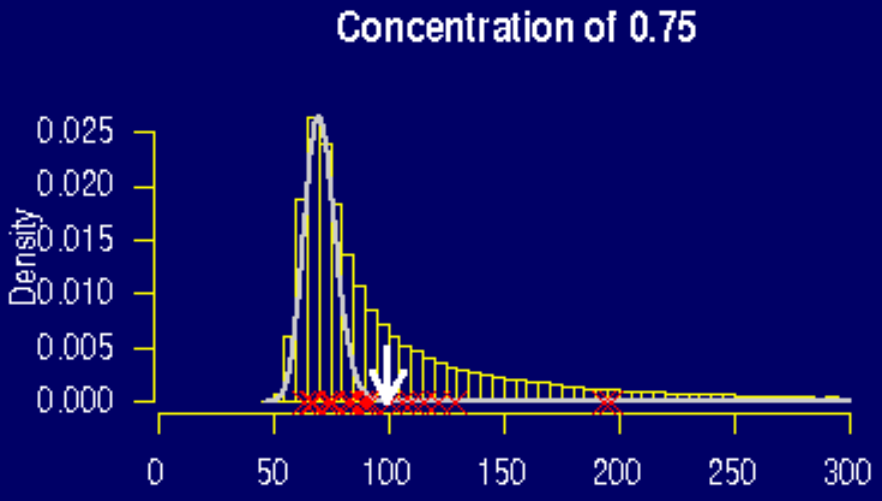

MM
20000
10000
5000
2000
1000
500
200
100
50
20
MM
0.5
1.0
2.0
5.0
10.0
20.0
50.0
100.0
Concentration

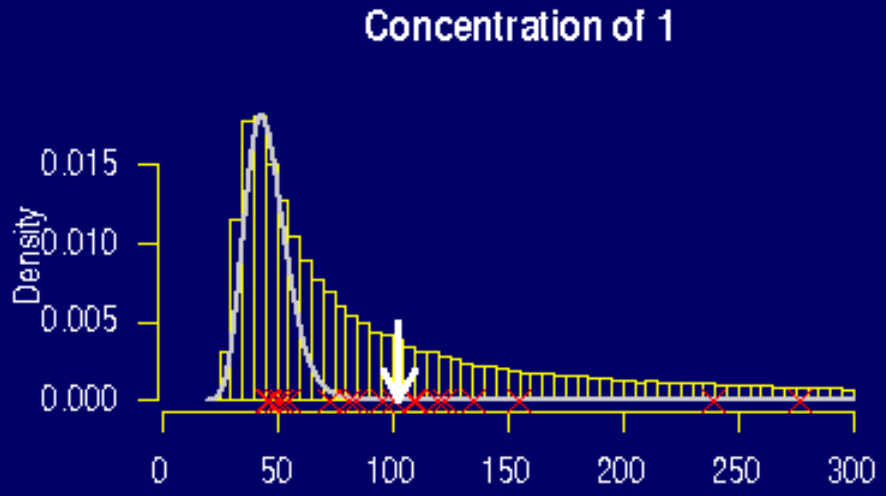

PM−MM
PM−MM
800
600
400
200
0
−200
−400
8000
6000
4000
2000
0
0.5
1.0
1.5
2.0
2.5
3.0
0
50
100
150
Concentration

Why and how we remove background









White arrows mark the means[MM]

RMA background

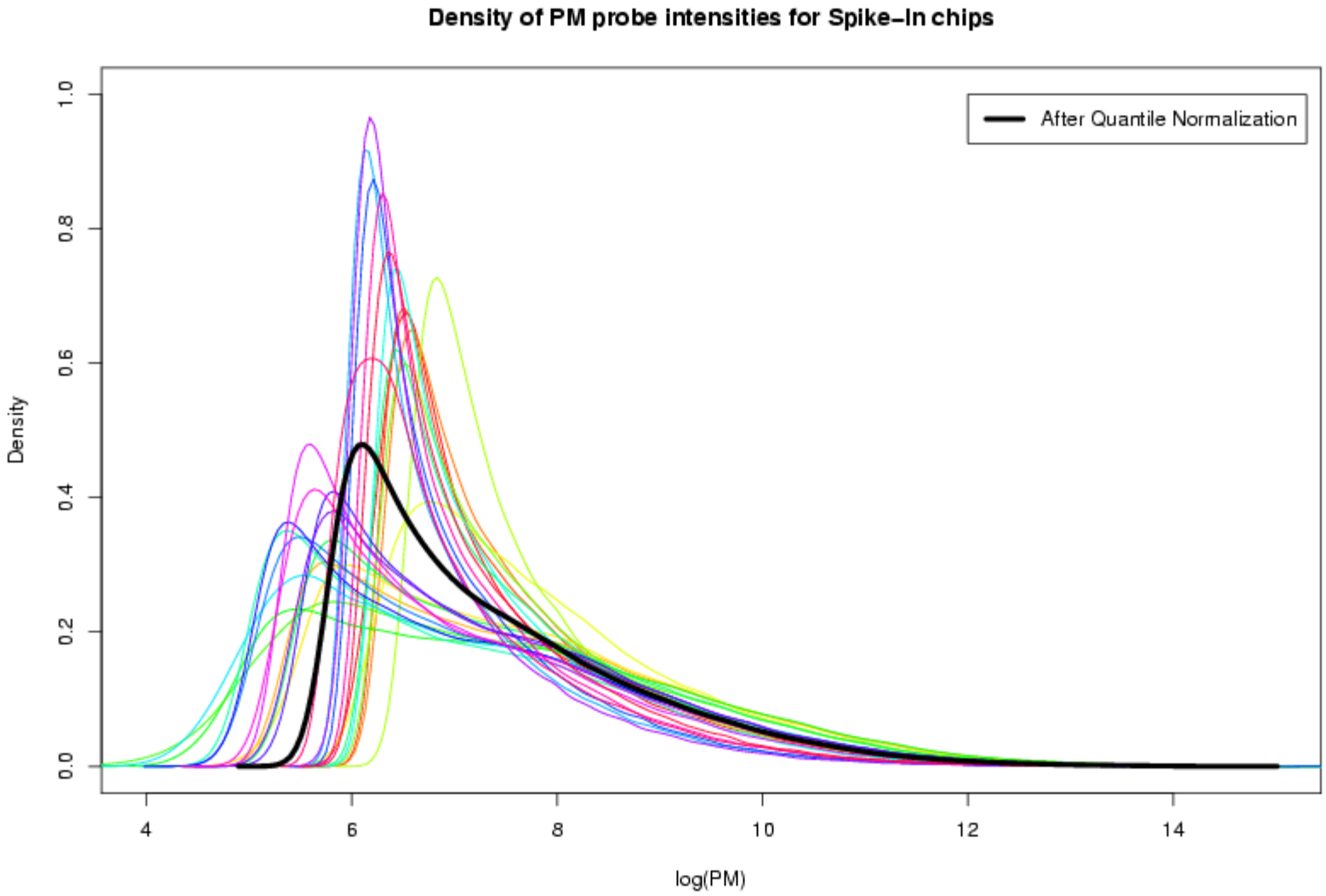
Our current background estimation

- Model observed PM as the sum of a signal intensity SG and a background intensity BG

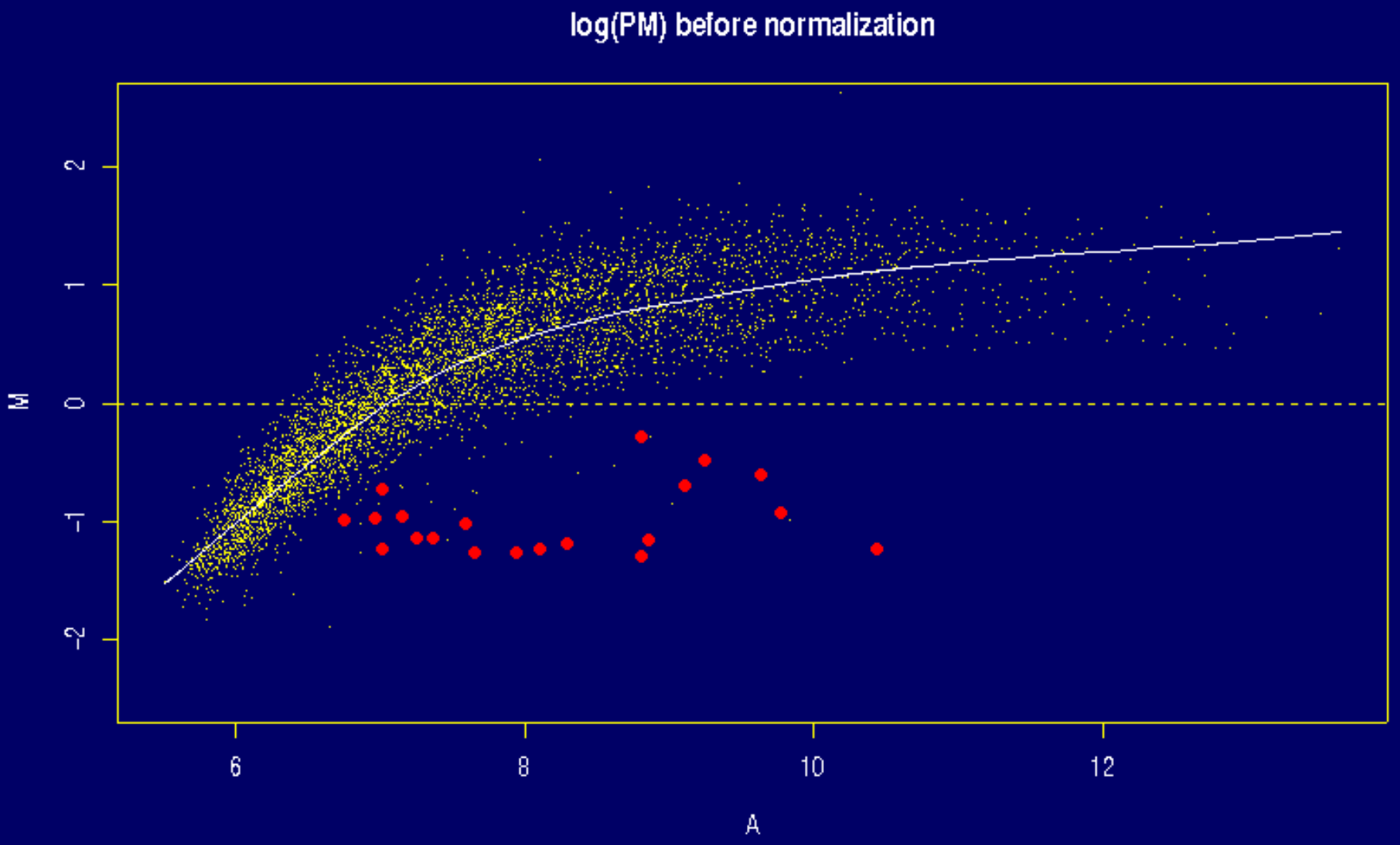
$$PM = SG + BG,$$

  where it is assumed that SG is *Exponential* (α), BG is *Normal* (μ, σ^2), and SG and BG are independent.
- Background adjusted PM values are then **E(SG|PM)**.

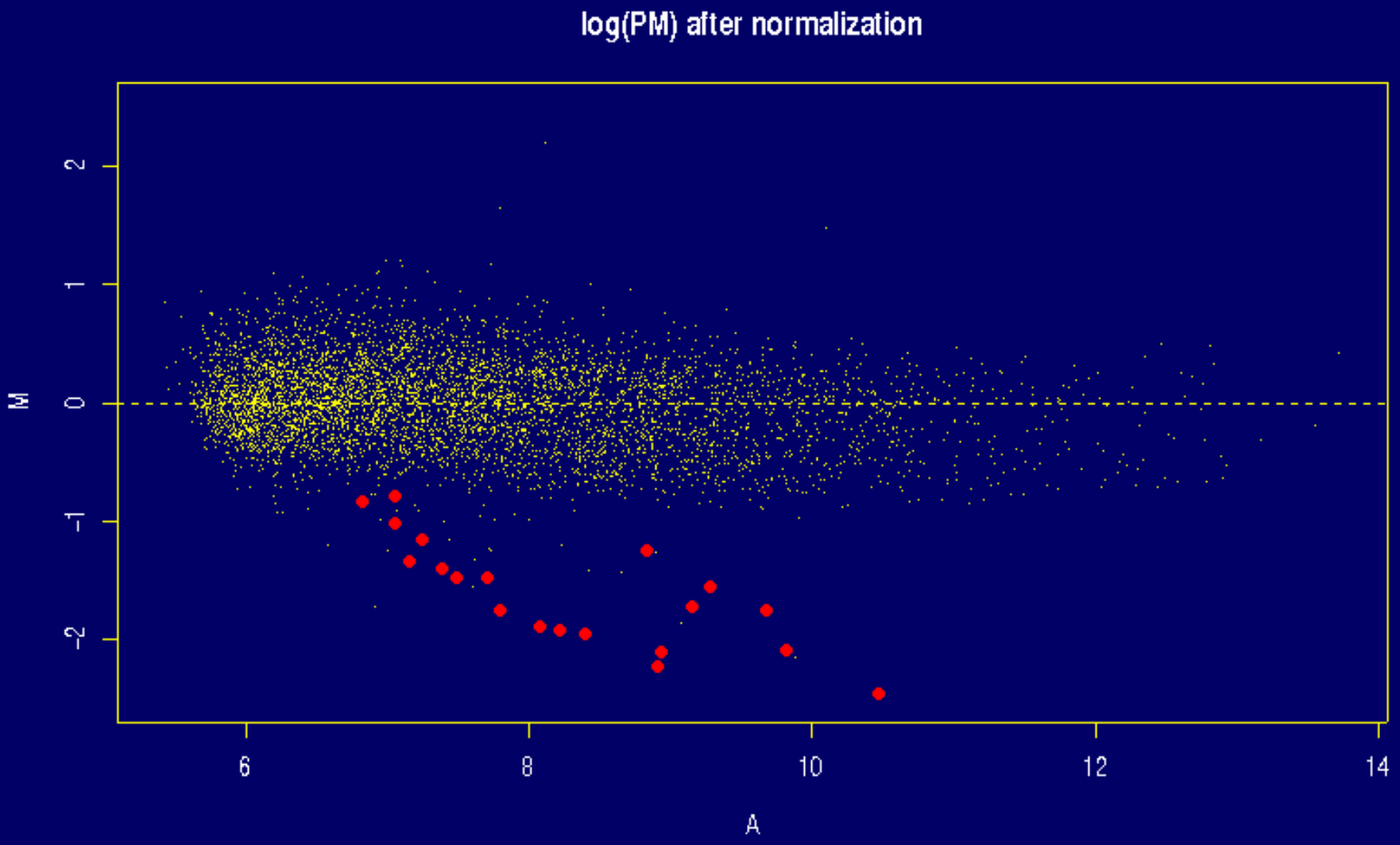
Why and how we normalize



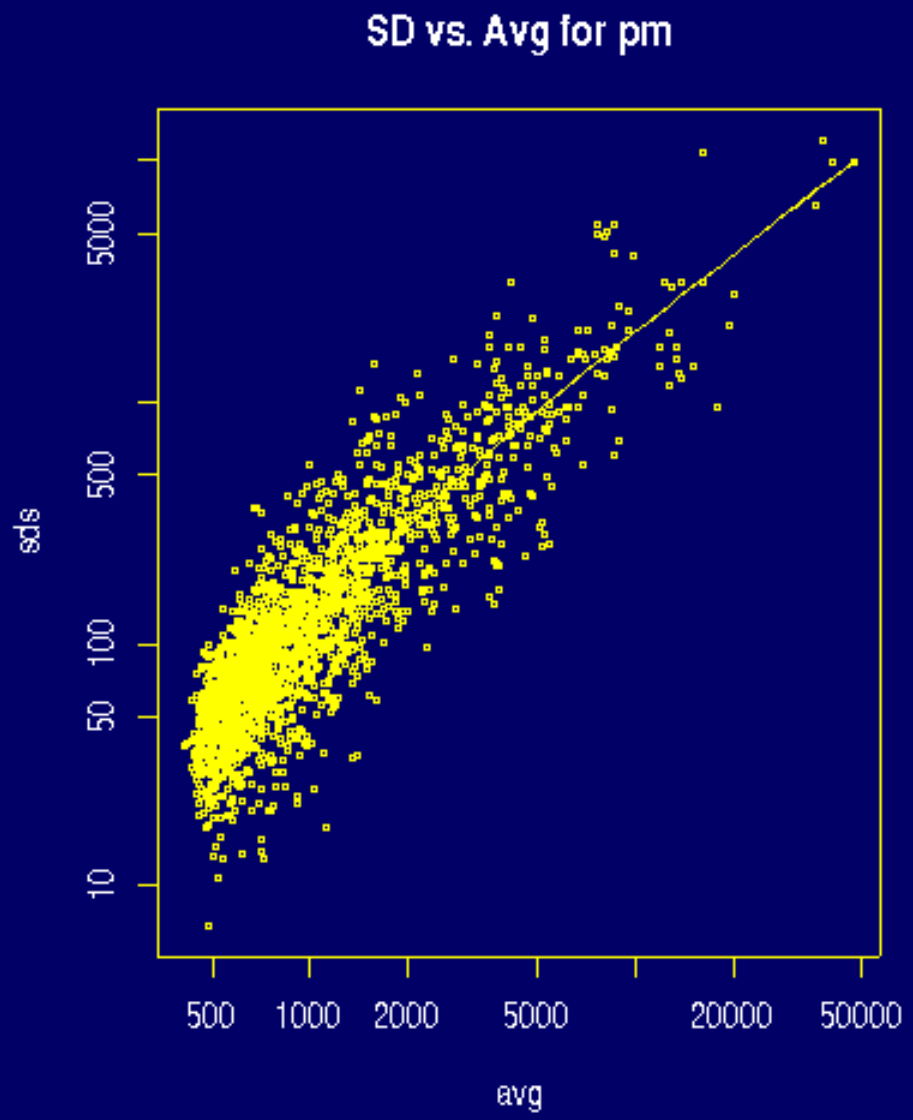
Normalization at Probe Level

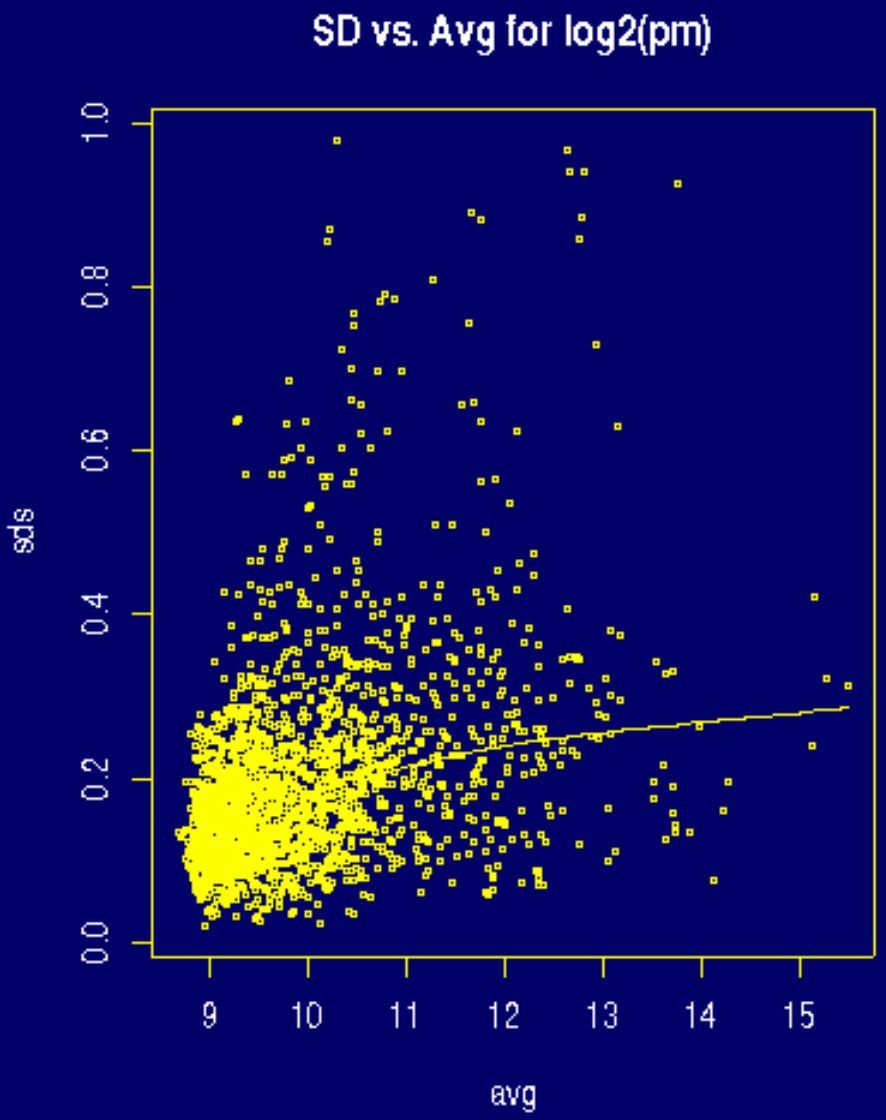


Normalization at Probe Level



Why we take $\log_2$





Why we write

$\log_2 n(PM-*BG) = \text{chip effect} + \text{probe effect}$

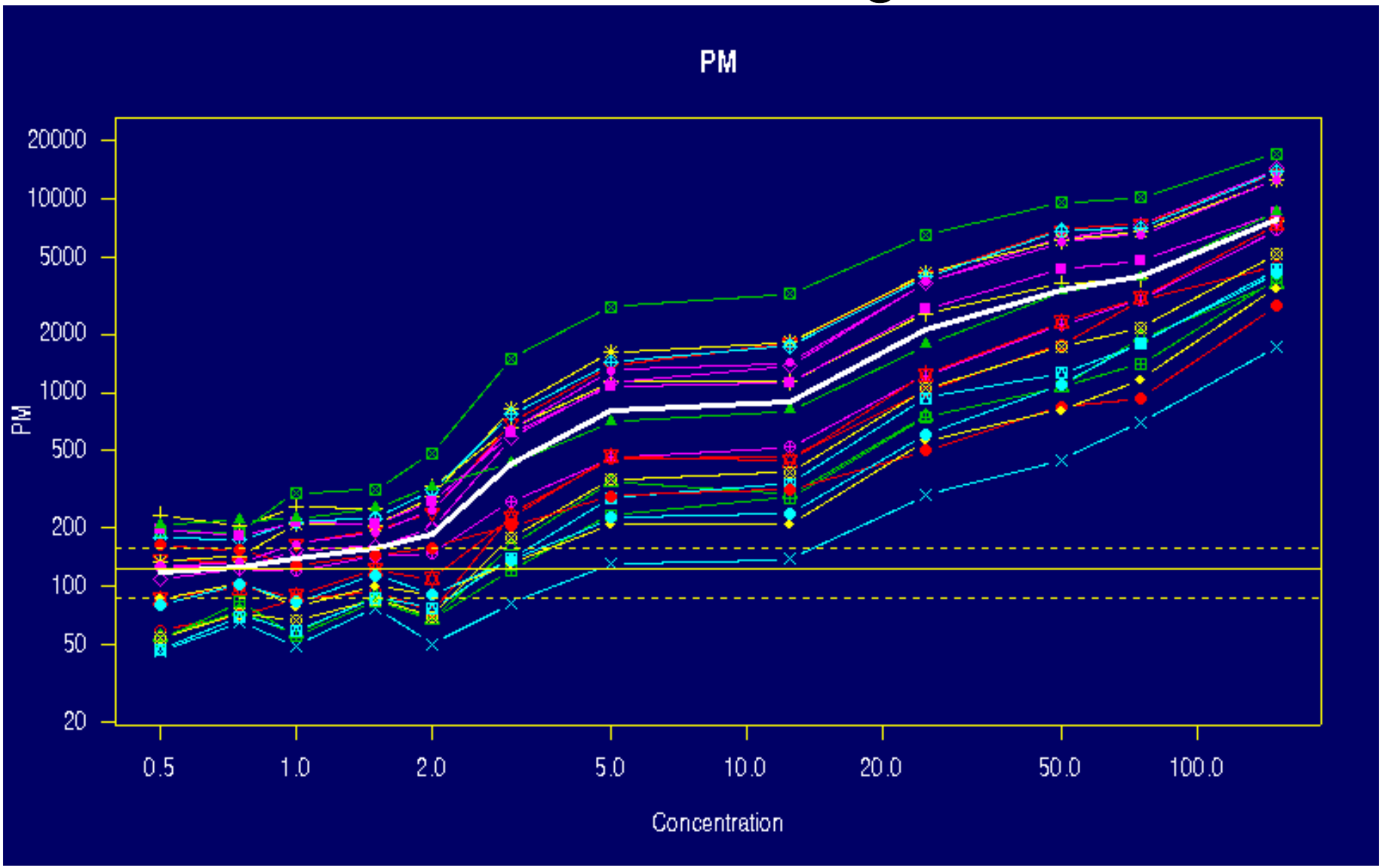
Because:

probe effects are additive on the log scale

The example on the next slide is typical of the set of 11.

Every set of experiments should exhibit this parallel behaviour across probes

Probe level data exhibiting parallel behaviour on the log scale



Why we carry out a Robust Multi-chip Analysis

Why multi-chip?

To put each chip's values in the context of a set of similar values.

This helps even if we do not do so robustly.

Why robust?

To get even more out of our multi-chip analysis.

In the old human and mouse series, perhaps 10%-15% of probe level values are "outliers".

Robust summaries really improve over the standard ones, by down weighting outliers and leaving their effects visible in residuals.

How we carry out our Robust Multi-chip Analysis

We base our analysis on the linear model embodying the parallel behaviour noted:

$$\log_2 n(PM_{ij} - *BG) = m + a_i + b_j + \varepsilon_{ij}$$

where *i* labels chips and *j* labels probes.

- Our current implementation (in R) uses median polish.
- Using robust linear model (***rlm)*** fitting procedure is an option
- This is an M-estimator with Huber's ψ.
- It is like Tukey's biweight, but in the 2-way array of chips by probes, and we adjust for probe affects
- Median polish and rlm are similar

RMA in summary

- We *background* correct PM on original scale
- We carry out *quantile* normalization
- We take log_2

Under the *additive* model

$$\log_2 n(PM_{ij} - *BG) = m + a_i + b_j + \varepsilon_{ij}$$

- We estimate chip effects a_i and probe effects b_j using a ***robust/resistant*** method.

Comparisons

We study the trade-off of

- Bias/variance (accuracy/precision), or
- False positives/true positives.

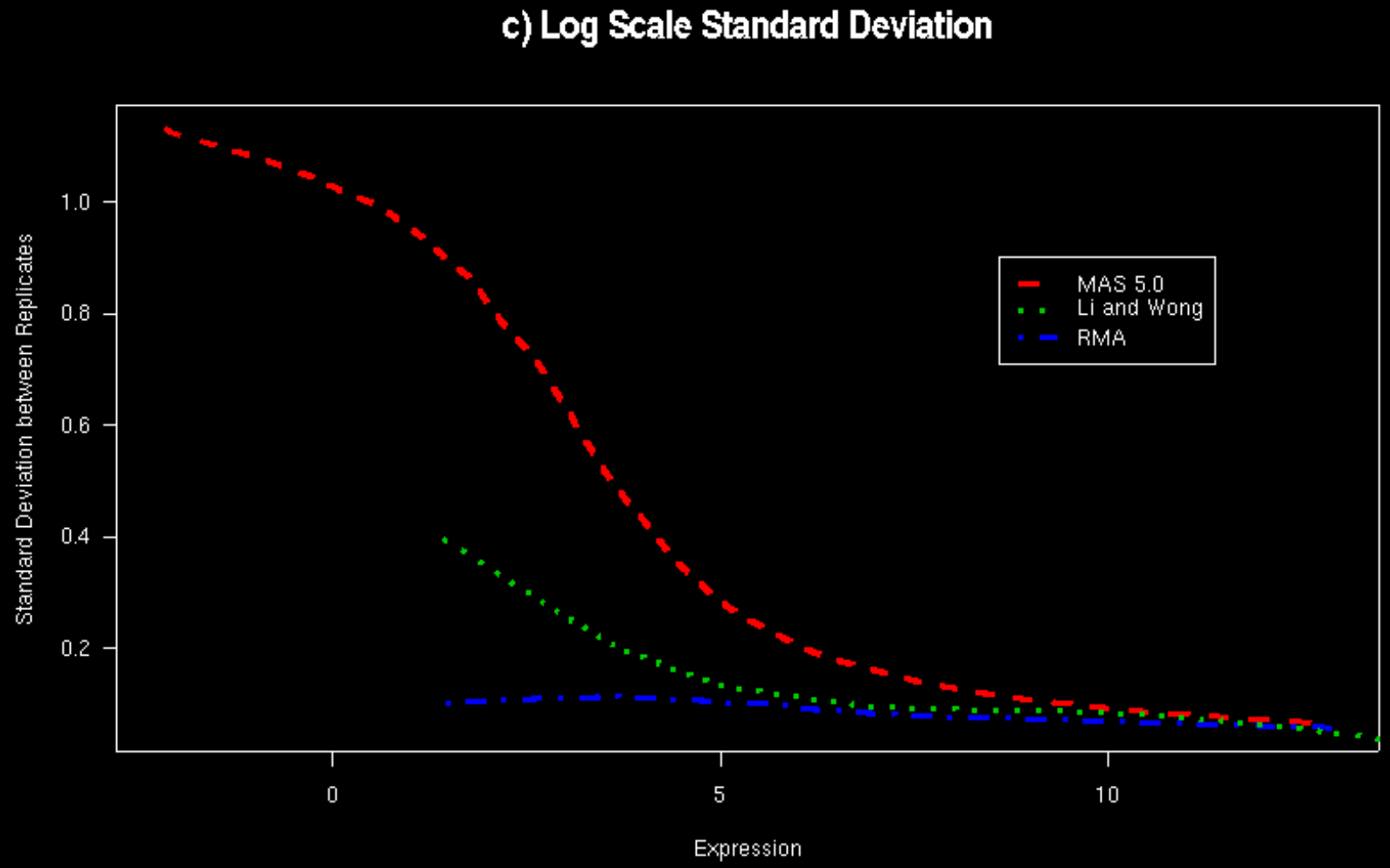
To place ourselves on the spectrum, we need some truth. Often hard to come by, but we have some special data sets from GeneLogic and Affymetrix.

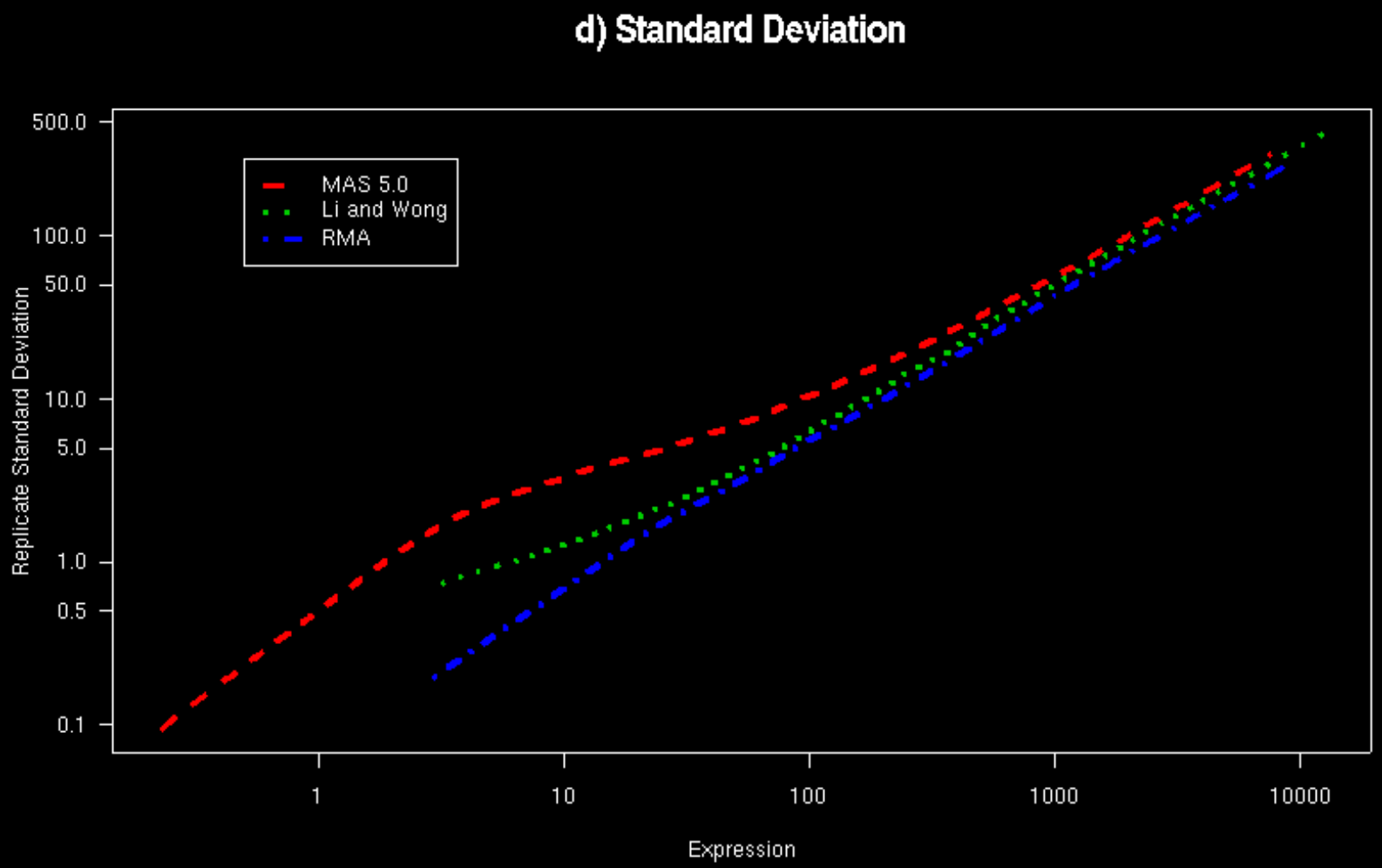
We begin looking at variability (SD) across replicates.

Dilution experiment

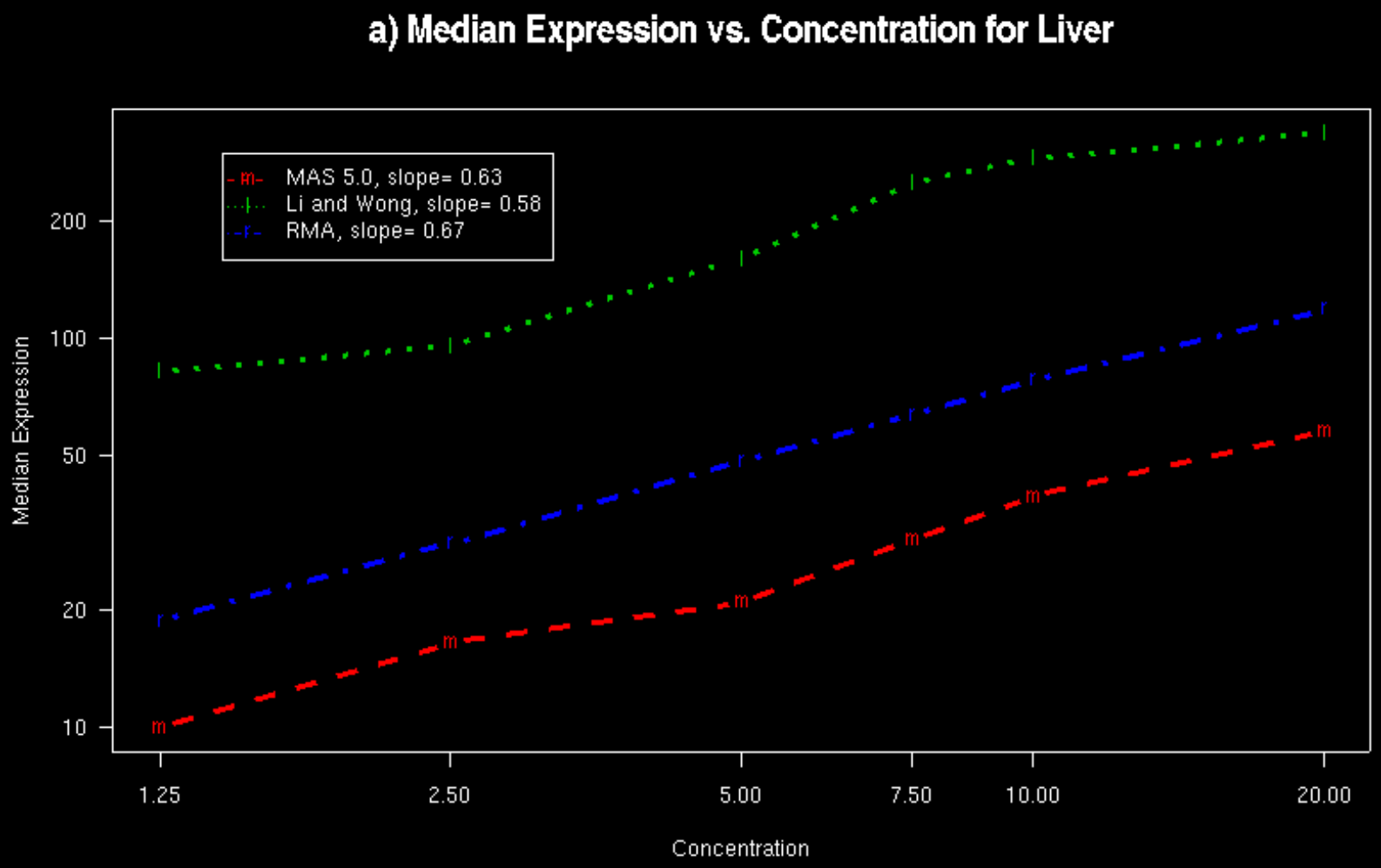
- cRNA hybridized to human chip (HGU95) in range of concentrations. Two different RNA sources were used each time.
- Dilution series begins at 1.25 μg cRNA per GeneChip array, and rises through 2.5, 5.0, 7.5, 10.0, to 20.0 μg per array. 5 replicate chips were used at each dilution
- Normalize just within each set of 5 replicates
- For each of 12,000 probe sets, we compute expression, average and SD over replicates

RMA has smaller SD
Especially for low intensities

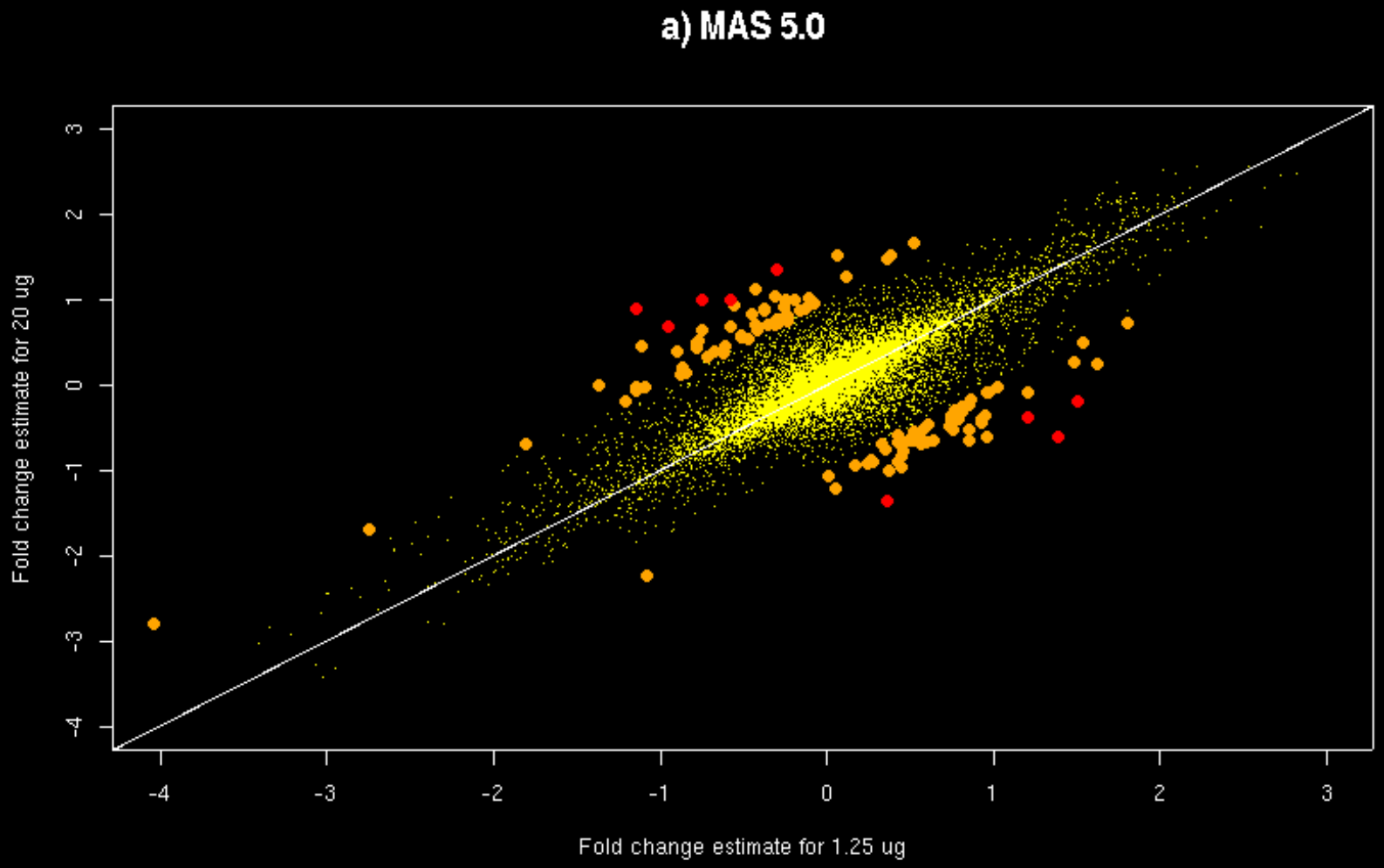



d) Standard Deviation
MAS 5.0
Li and Wong
RMA
Replicate Standard Deviation
500.0
100.0
50.0
10.0
5.0
1.0
0.5
0.1
1
10
100
1000
10000
Expression

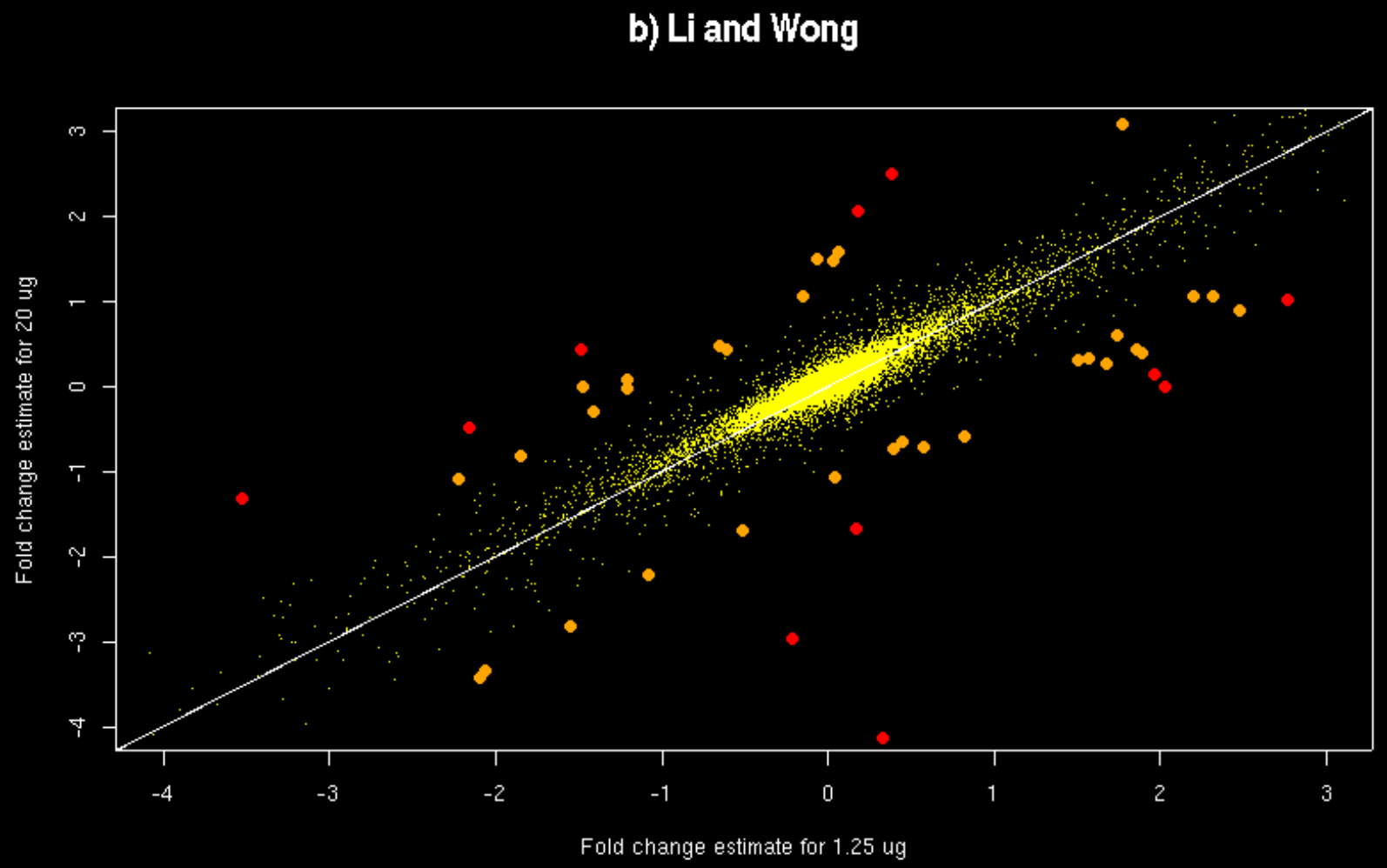
Do we sacrifice signal detection (bias)?



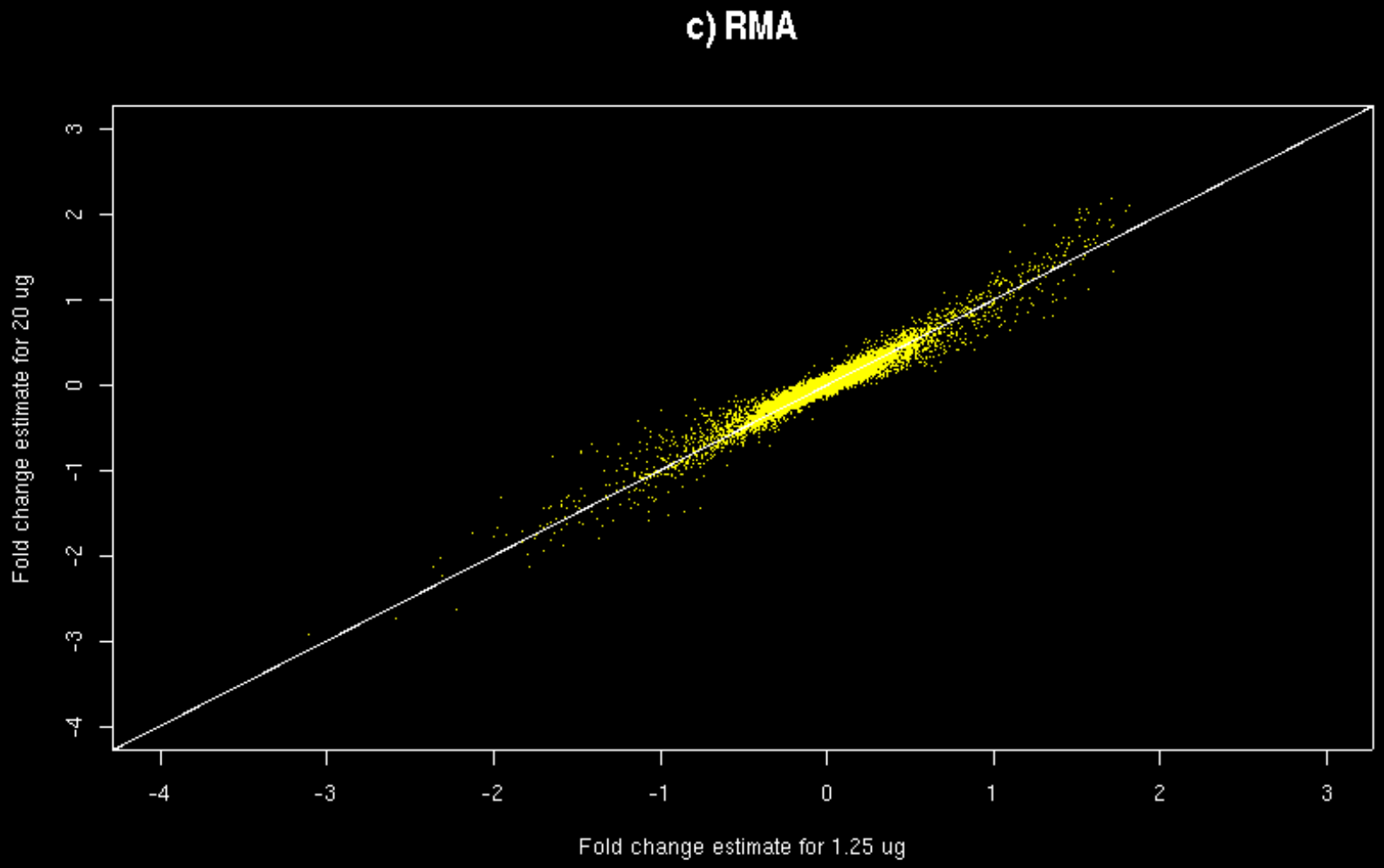
Consistency across dilution



Consistency across dilution



Consistency across dilution



Next comparison uses Part of Spike-in Data B

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 many different combinations of concentrations.

Displaying differential expression

In the following slides: for each gene we plot the log fold change **M** across two chips, given by

$$\log(\text{chip 1}/\text{chip 2}) = \log(\text{chip 1}) - \log(\text{chip 2}),$$

vertically, against overall abundance **A**, measured by

$$\log \sqrt{(\text{chip 1})(\text{chip 2})} = [\log(\text{chip 1}) + \log(\text{chip 2})]/2$$

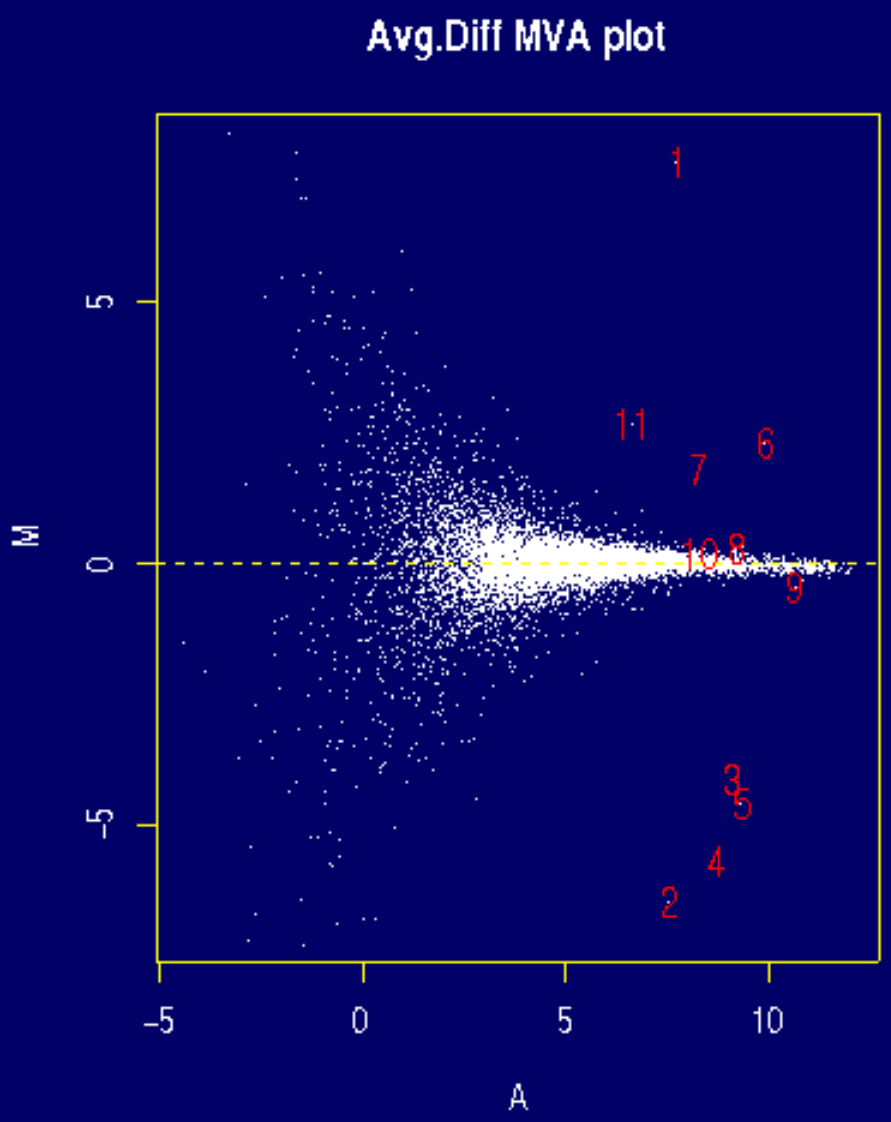
horizontally. This is just a rotated version of the plots everyone else uses.

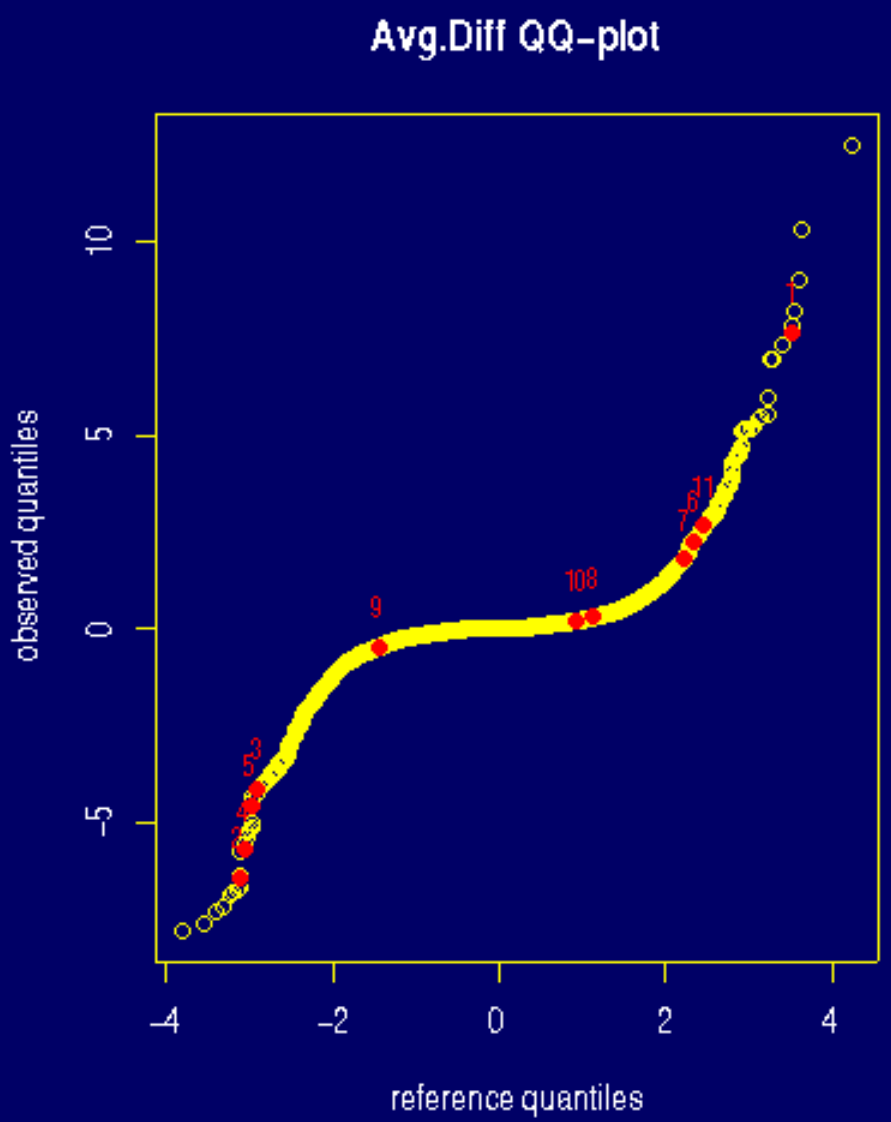
Quantile-quantile plots

These are used by statisticians to compare distributional shape, and to highlight extremes, relative to a reference distribution for the majority. In a sense they are just cumulative distributions with the axes rescaled.

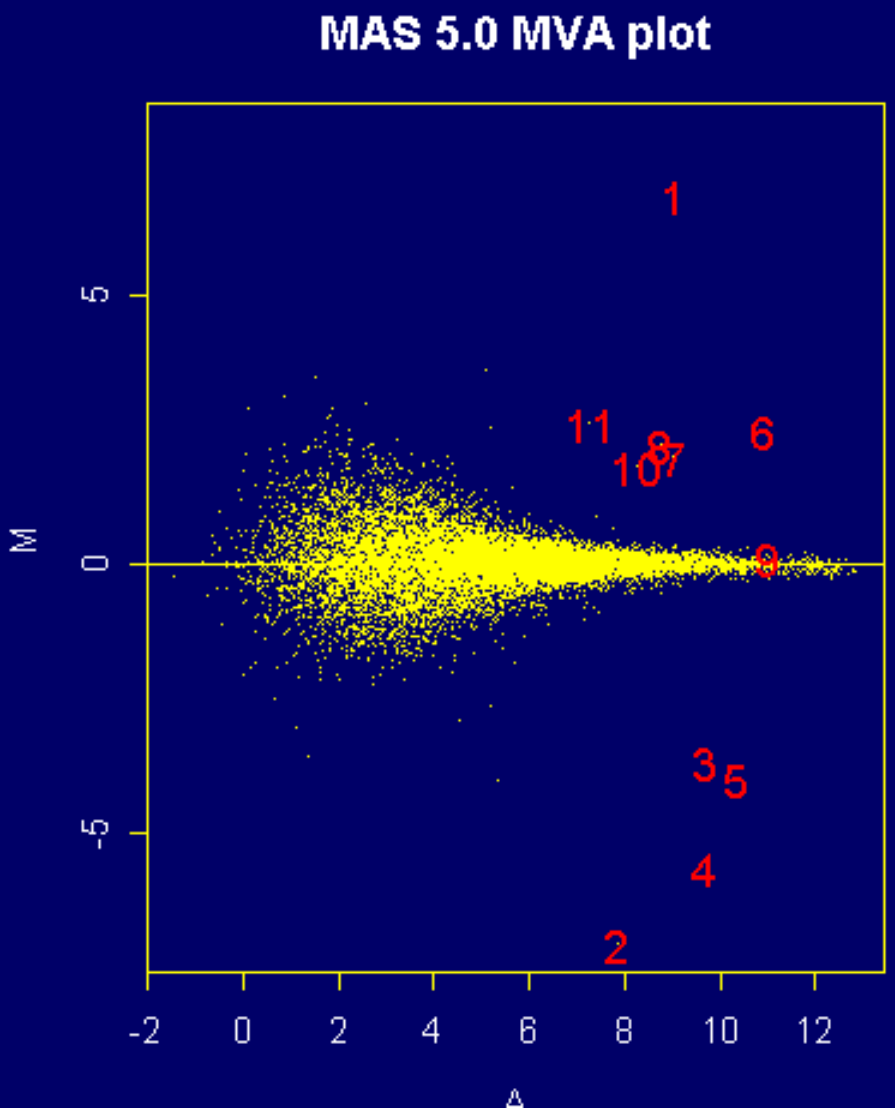
In the examples which follow, the reference distribution is normal (Gaussian) for the log fold change M. The better the fit of the majority, the straighter the line, and the more the extremes will stand out. Everything is on the log scale.

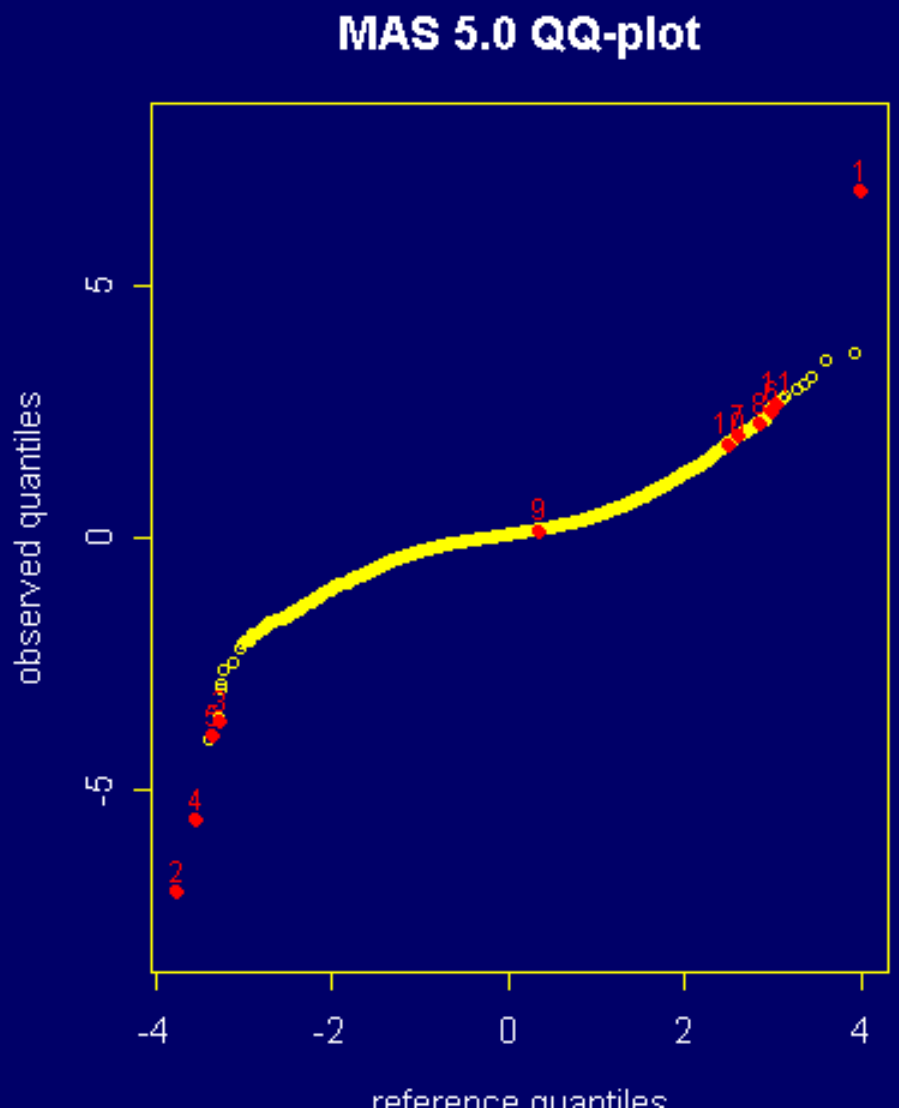
Differential Expression



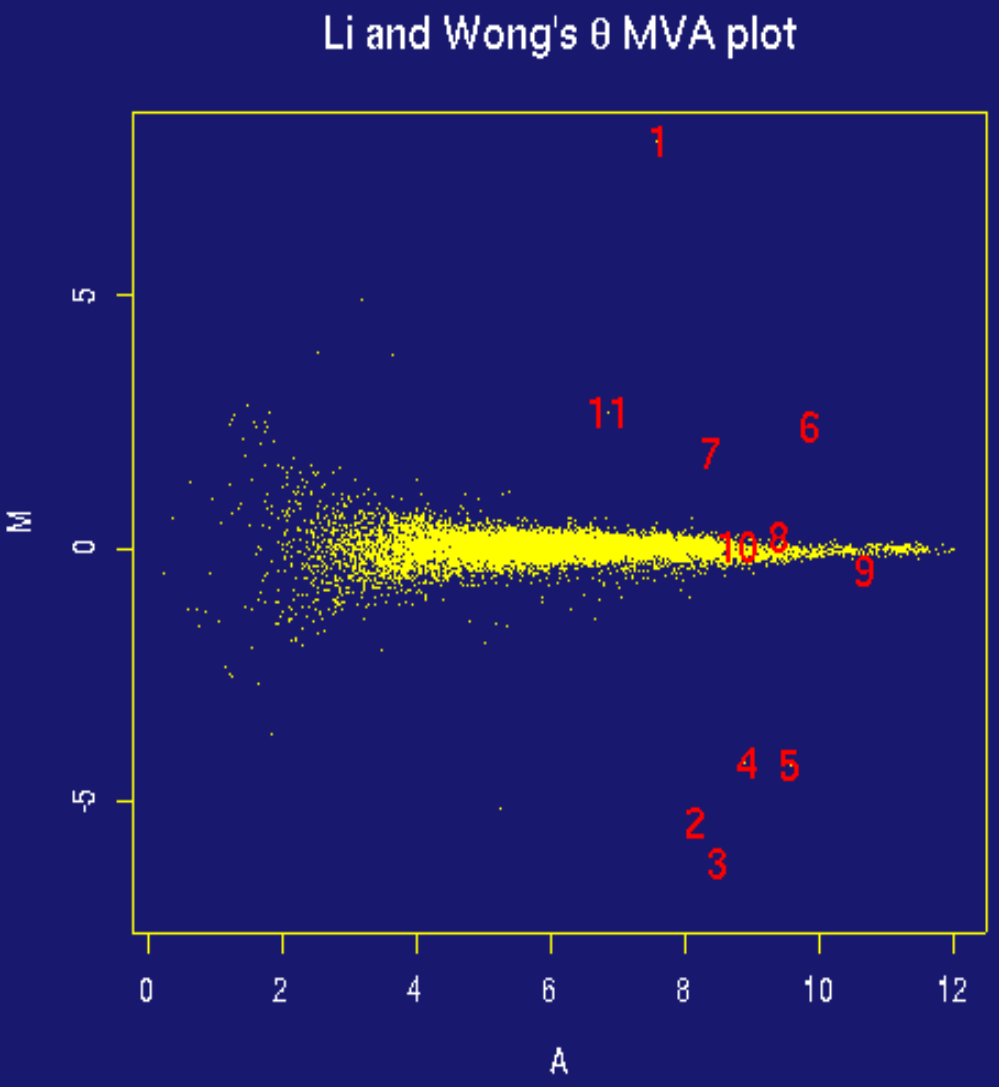


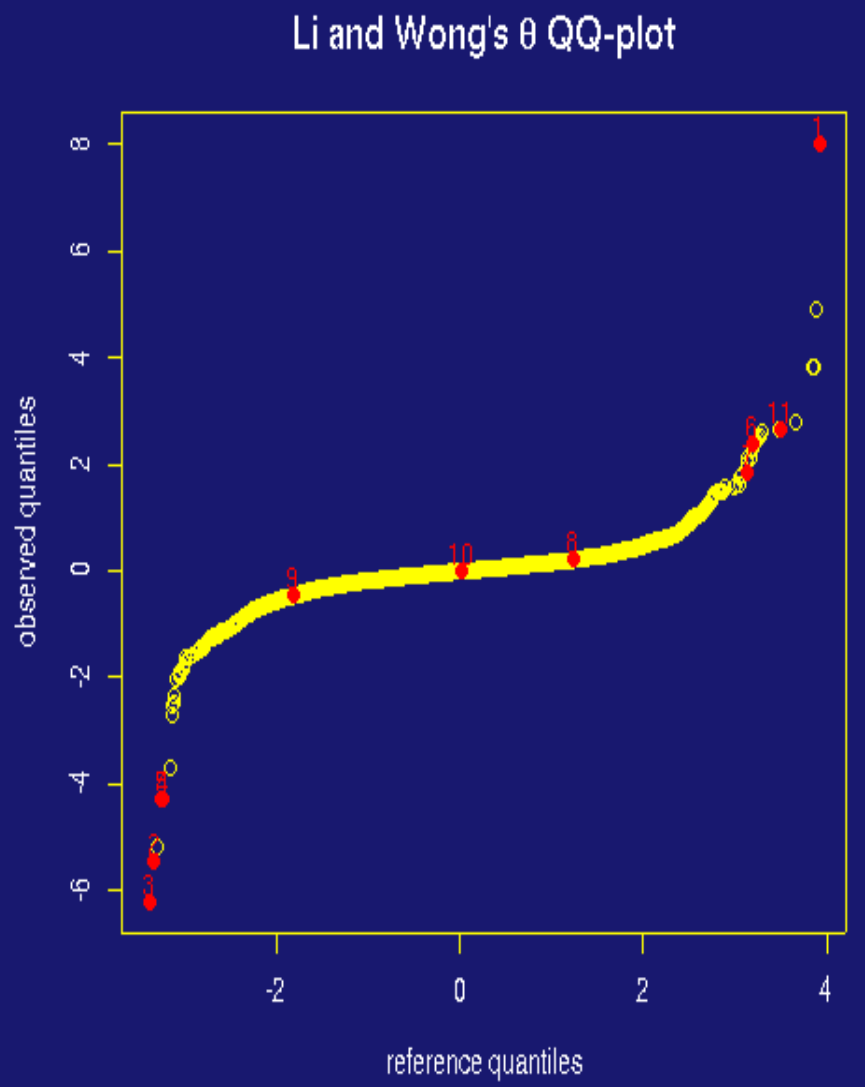
Differential expression



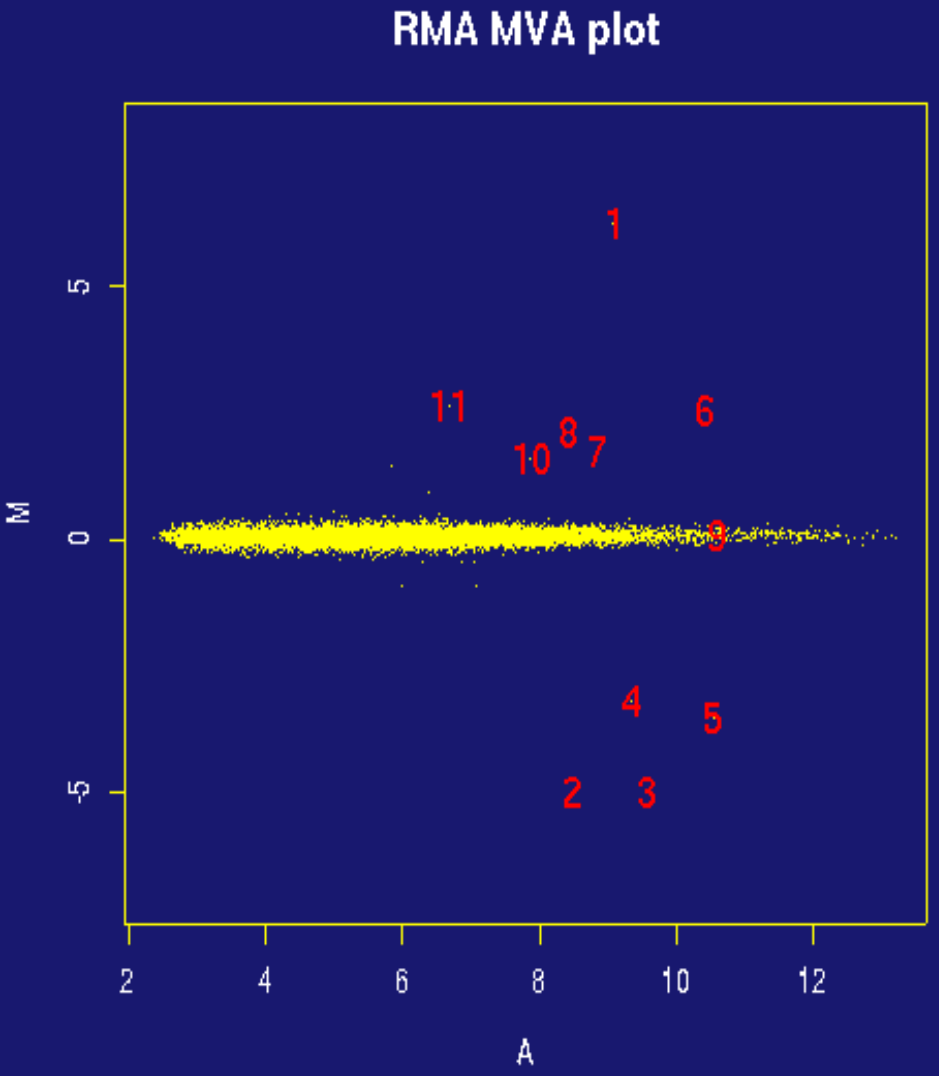


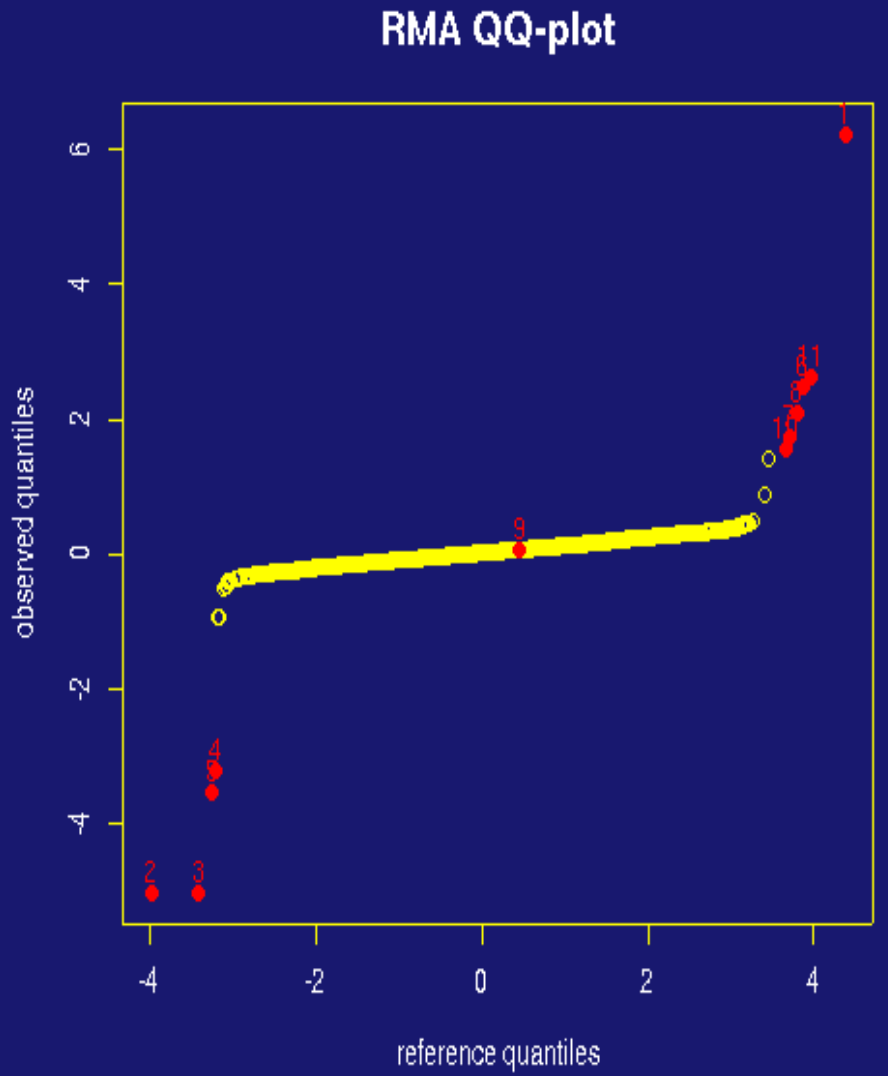
Differential expression





Differential expression





Observed ranks

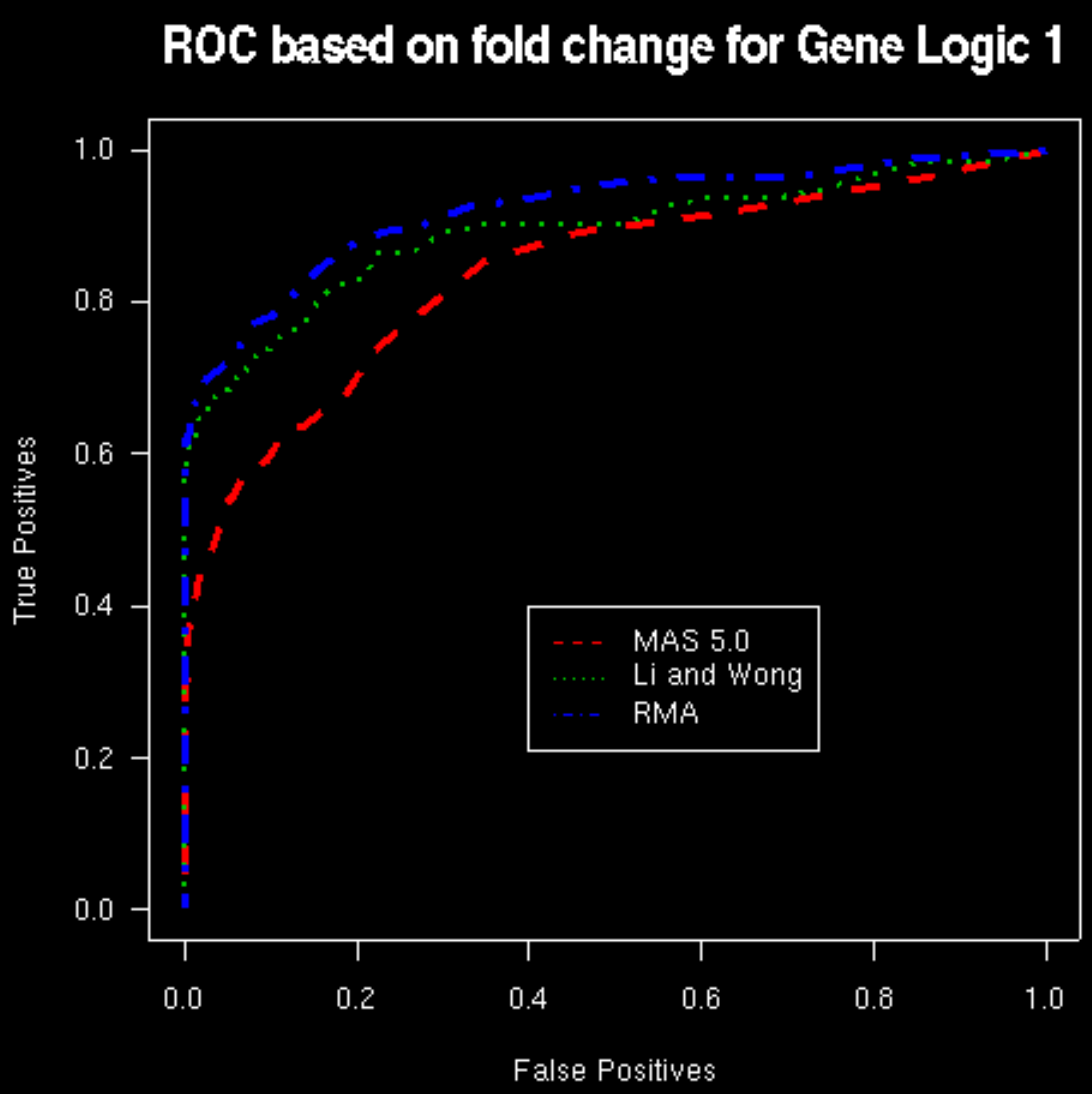
Gene	AvDiff	MAS 5.0	Li&Wong	AvLog(PM-BG)
BioB-5	6	2	1	1
BioB-3	16	1	3	2
BioC-5	74	6	2	5
BioB-M	30	3	7	3
BioDn-3	44	5	6	4
DapX-3	239	24	24	7
CreX-3	333	73	36	9
CreX-5	3276	33	3128	8
BioC-3	2709	8572	681	6431
DapX-5	2709	102	12203	10
DapX-M	165	19	13	6
Top 15	1	5	6	10

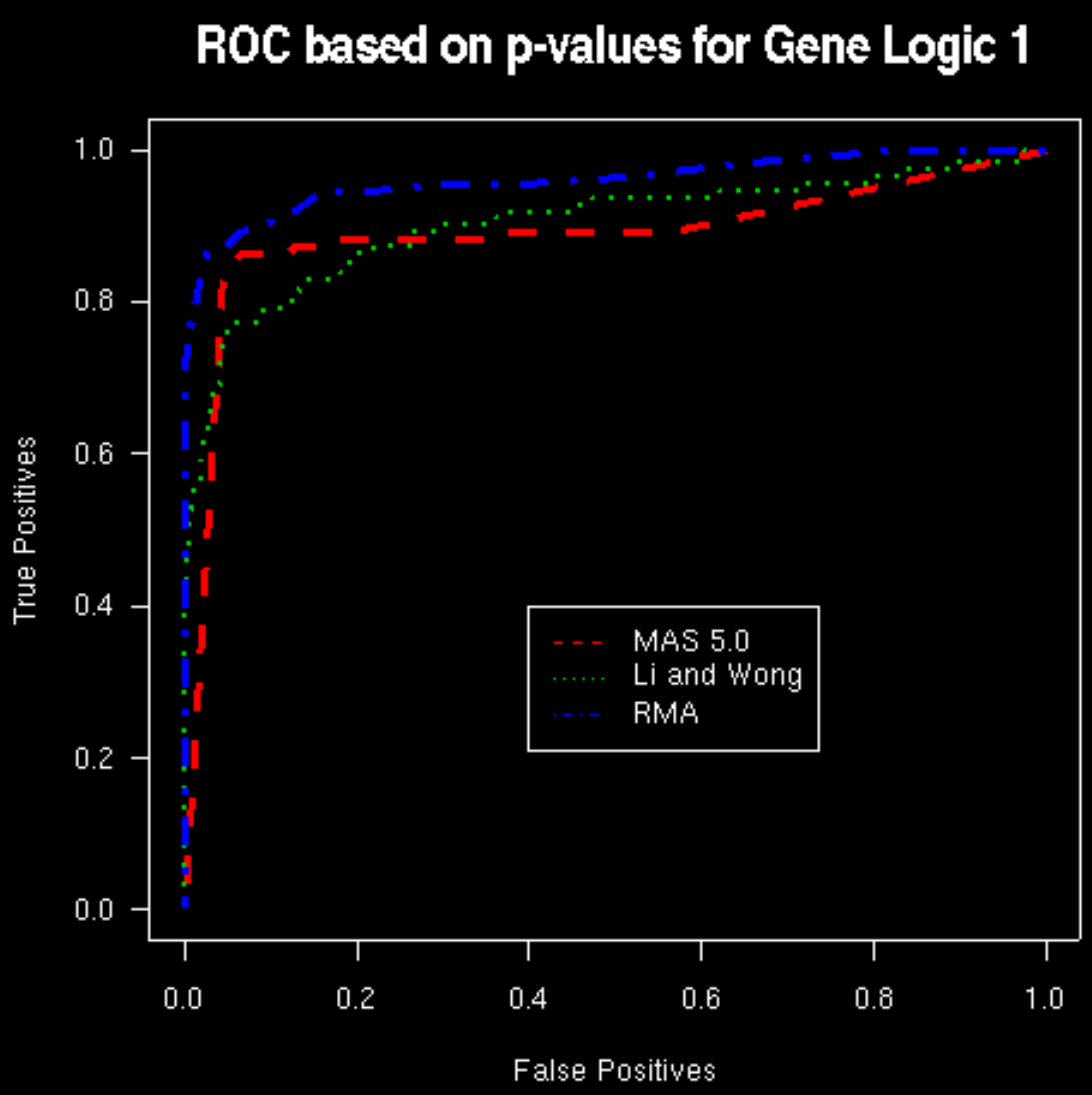
Receiver Operating Characteristic curves: single chip comparisons

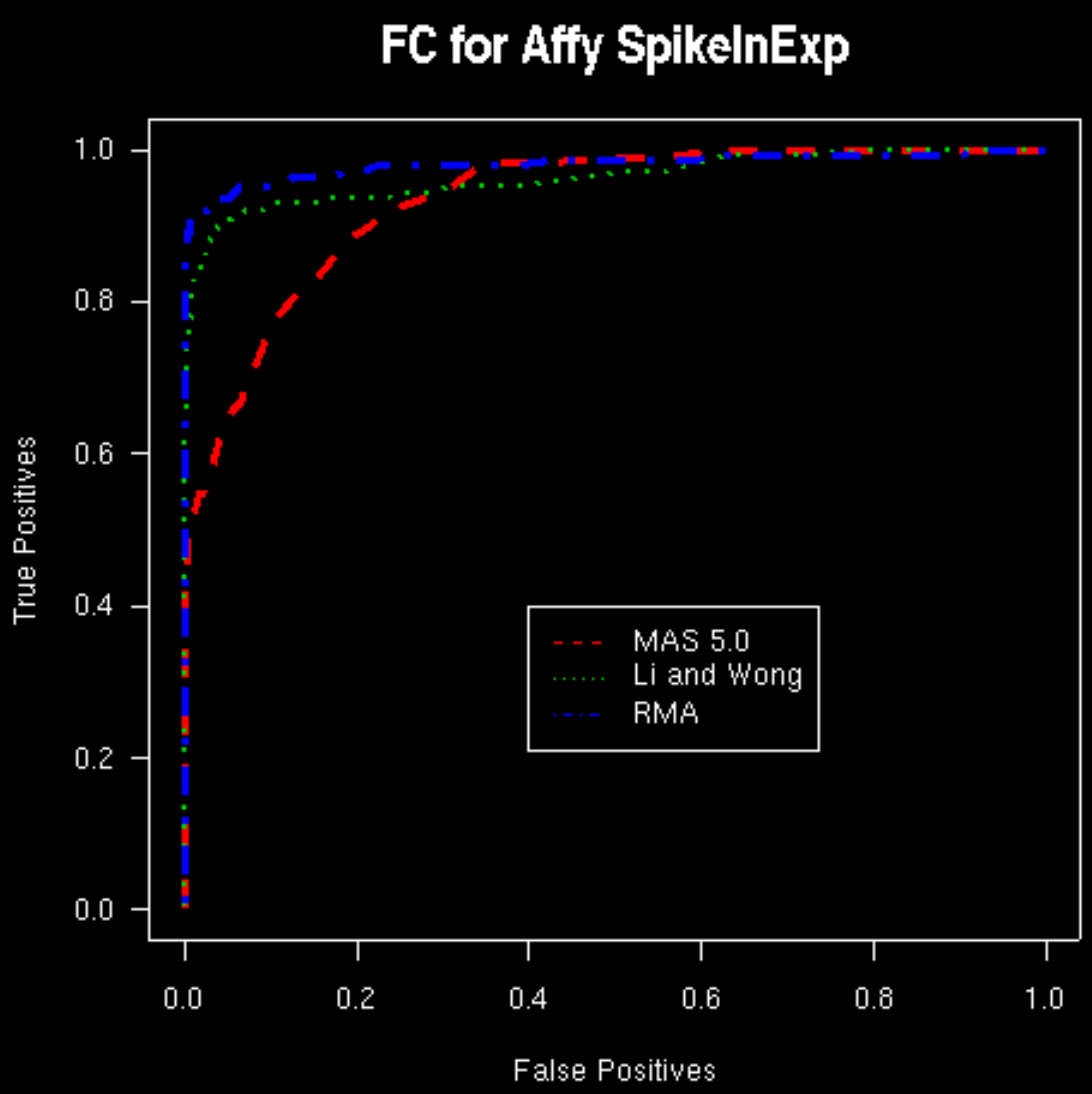
ROC curves compare the true and false positive rates at varying cut-off values for a stated criterion such as fold change.

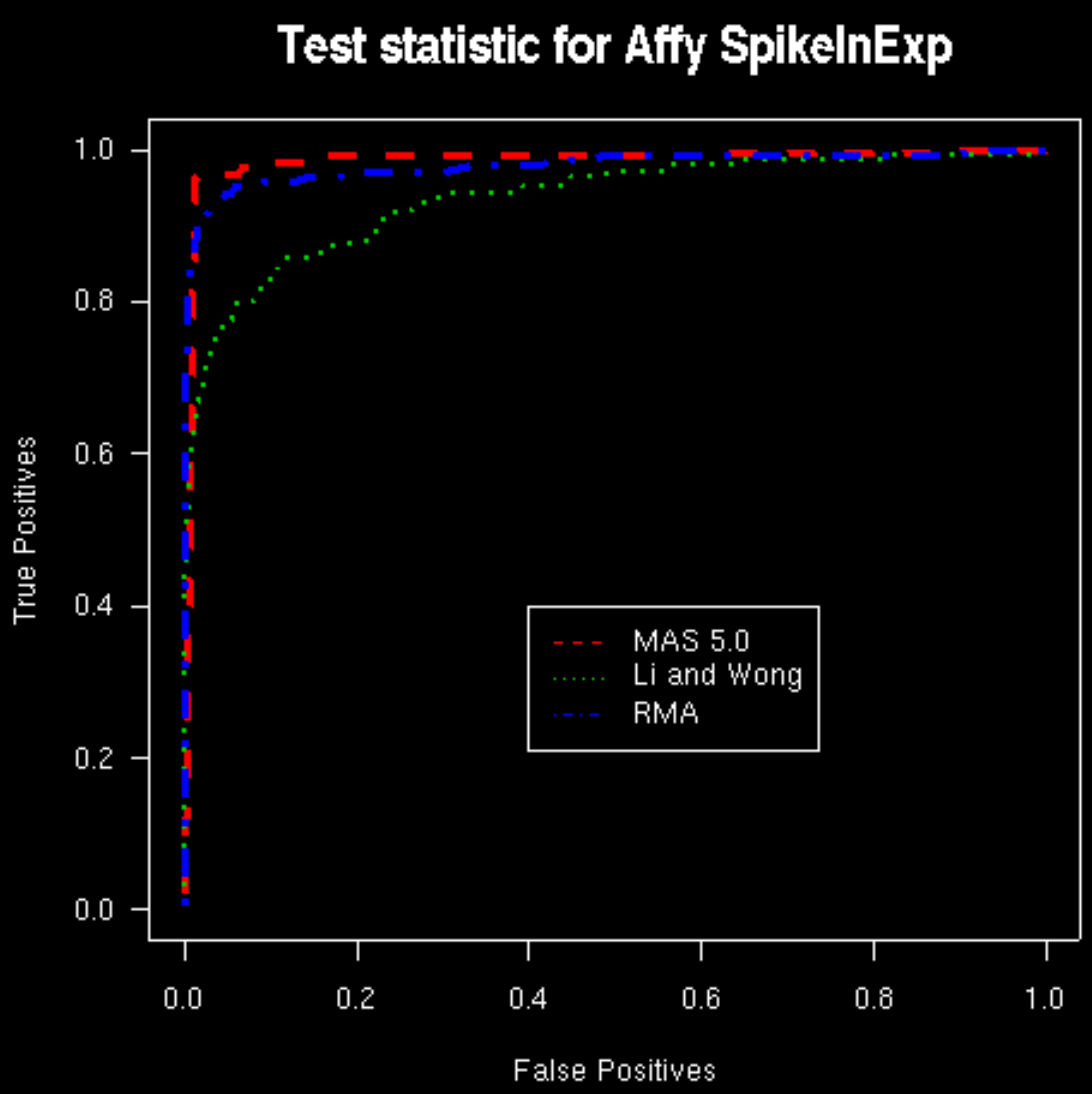
In the next four plots we compare ROC curves from MAS, LW (PM-only) and RMA using either fold change (FC) or the associated p-values, in single chip comparisons using spike-in data.

We have pooled the results from a number of 1 chip vs 1 chip comparisons, to get a smooth ROC curve.


ROC based on fold change for Gene Logic 1
True Positives
False Positives
0.0
0.2
0.4
0.6
0.8
1.0
MAS 5.0
Li and Wong
RMA


ROC based on p-values for Gene Logic 1
True Positives
False Positives
0.0
0.2
0.4
0.6
0.8
1.0
MAS 5.0
Li and Wong
RMA


FC for Affy SpikeInExp
True Positives
False Positives
0.0
0.2
0.4
0.6
0.8
1.0
MAS 5.0
Li and Wong
RMA


Test statistic for Affy SpikeInExp
True Positives
False Positives
0.0
0.2
0.4
0.6
0.8
1.0
MAS 5.0
Li and Wong
RMA

Conclusions from single chip comparison ROC curves

On the basis of the data just presented, and much more:

With FC, RMA is best, LW next. MAS does not do well here.

With p-values, RMA is as good as, and usually better than MAS, which is next. MAS does best on the Affymetrix spike-in data sets. LW (dChip) does not do so well here.

All judgements are comparative. Everyone does well in absolute terms, but some do better.

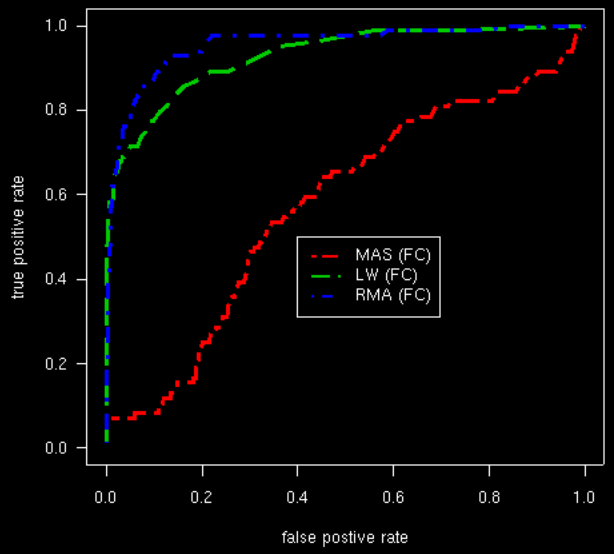
Comparing expression summaries and test statistics with replicated data

Here we display ROC curves to compare expression measures and test statistics with replicated spike-in data.

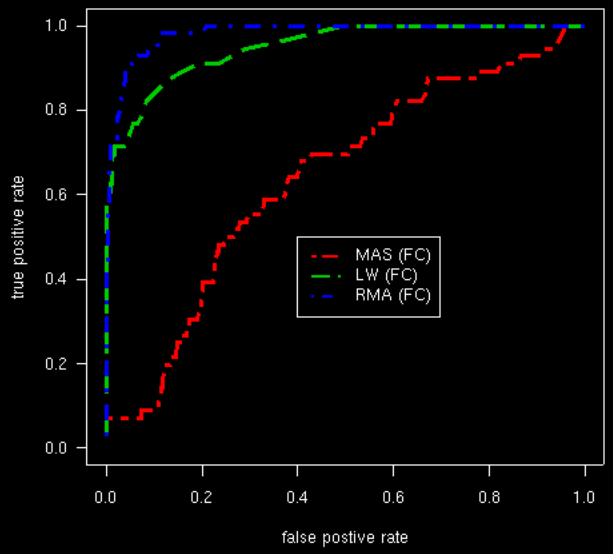
We have used a subset of 24 of the 59 chips from the Affymetrix spike-in study, which is 2 sets of 12 where all probe sets are at the same concentration. This gives us two populations, which for N = 2, 3, 4, 6 and 12 we divide into 12/N subgroups and use measures and test statistics to determine true and false positives. One ROC curve for all is calculated.

Comparisons using FC, N = 2, 3, 4, 6 and 12.

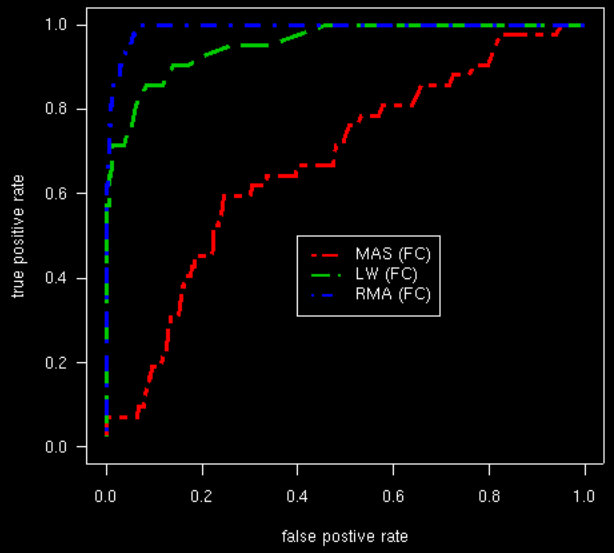
a) Comparison of measures (N=2) using Affy SpikeIn



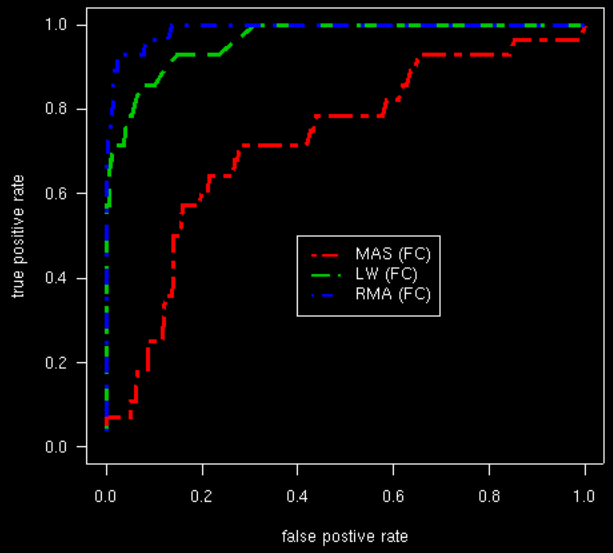
b) Comparison of measures (N=3) using Affy SpikeIn



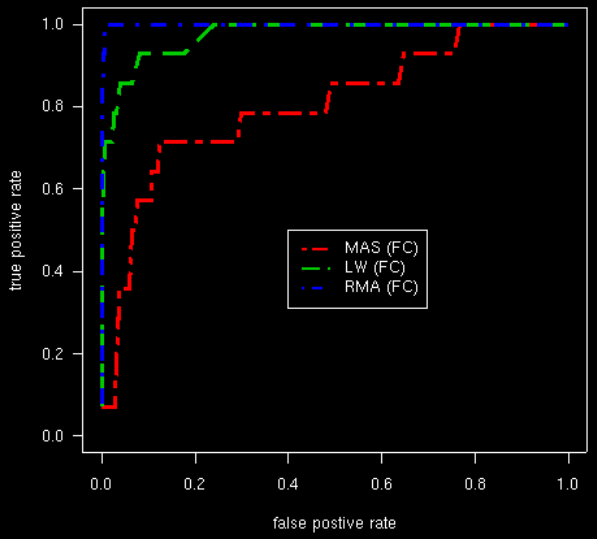
c) Comparison of measures (N=4) using Affy SpikeIn



d) Comparison of measures (N=6) using Affy SpikeI



e) Comparison of measures (N=12) using Affy SpikeIn



A few remaining questions

Now that the probe sequences are available, it is a challenge to make use of them to compute better expression summaries.

Use of RMA residuals in quality assessment - of parts or all of chips, of probes - work in progress. Very promising.

To pool or not to pool? How many replicates? What kind replicates? How can we adjust for the host of systematic effects that manifest themselves in GeneChip data?

Low level analysis never ends…as the technology evolves, we need to go with it, answering these questions as we go.