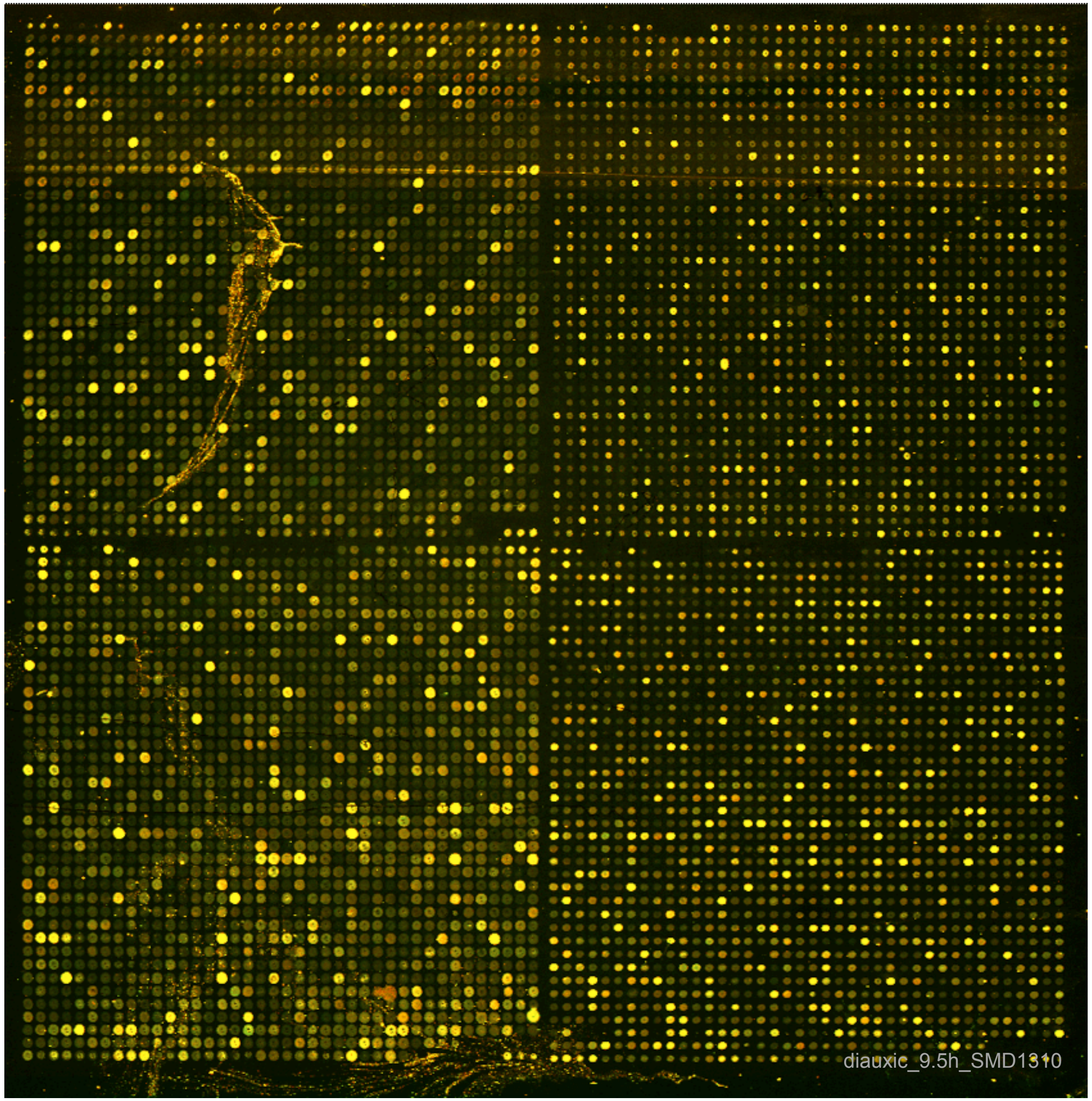
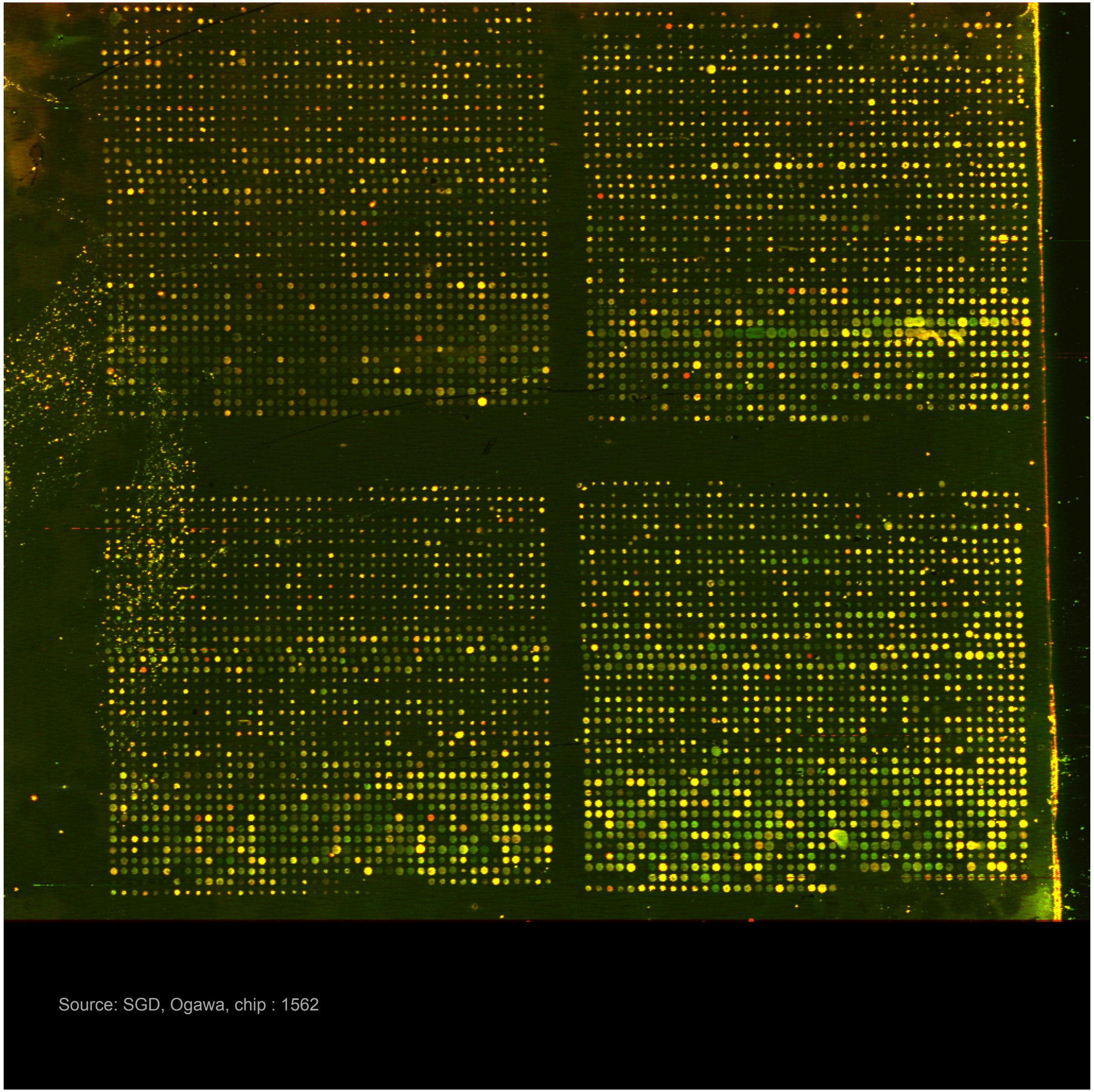
Normalization of microarray data

2-channel microarrays

Why do we need to normalize ?

- Microarrays can suffer from various experimental artifacts.
  - Spotting effects (block effect): the spot size (and thus DNA concentration) may be affected by the spotting head -> vary beteen printing blocks.
  - Position effects: some regions of the slide are more intense, both for the signal and for the background.
  - Dust : dust is highly fluoresent.
  - Dirt: the slide hereby is clearly contains some dirty zone.
  - Scratches: scratches on the glass create autofluorescent zones.
  - Undetectable spots: for some spots, the signal is so small that it vcan be lower than the surrounding background.
  - Intensity effect: the Intensity=f ([RNA]) curve is not perfectly linear, and is different for the green and red dyes.





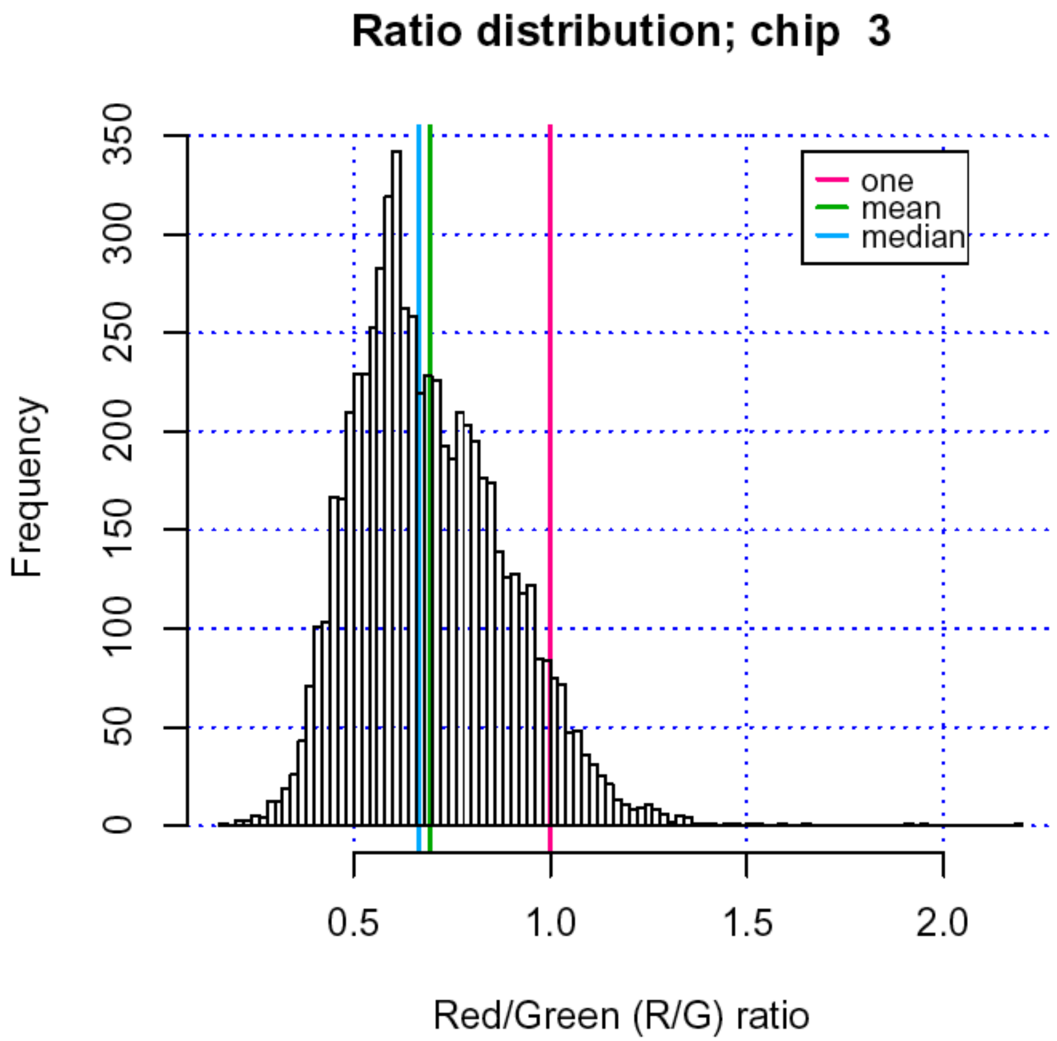
Source: SGD, Ogawa, chip : 1562

The raw measurements

Raw measurements

- Raw measurements are provided as mean and background intensities for the red and green channels.
- For each spot on the slide :
  - $R=R_m-R_b$
    - R_m red mean
    - R_b red background
  - $G=G_m-G_b$
    - G_m green mean
    - G_b green background

Never use ratios



- $r=R/G$
- The ratio is a very poor statistics.
- It reflects very badly the regulation :
  - A 10-fold up-regulation is represented by a value of 10, its distance to the random expectation is 9.
  - A 10-fold down-regulation is represented by a ratio of 0.1. Its distance to the random expectation is 0.9.
- Raw ratios will thus emphasize up-regulation, and ignore down-regulation.

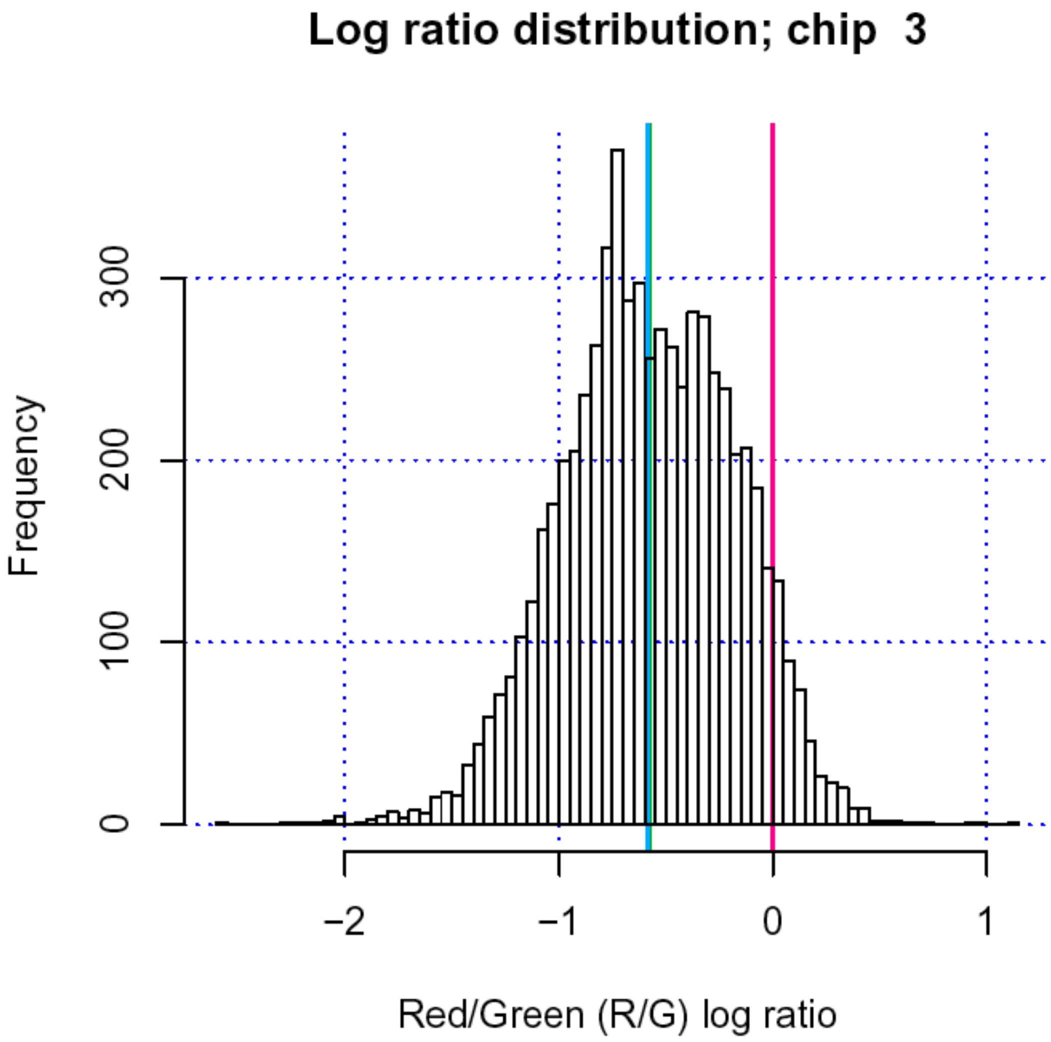
Log-ratios

- Using log-ratios has a normalizing effect.
- Usually, a base 2 is is used for the log, because it is more intuitive and easy to convert.
- Some examples:

R & G	ratio	log(ratio)	regulation
R=G	1	0	random expectation
R=G*2	2	1	2-fold up-regulation
R=G*4	4	2	4-fold up-regulation
R=G/4	0.25	-2	4-fold down-regulation

- The statistic is symmetric: up- and down-regulated genes are at the same distance from random expectation (0)

Log ratio distribution

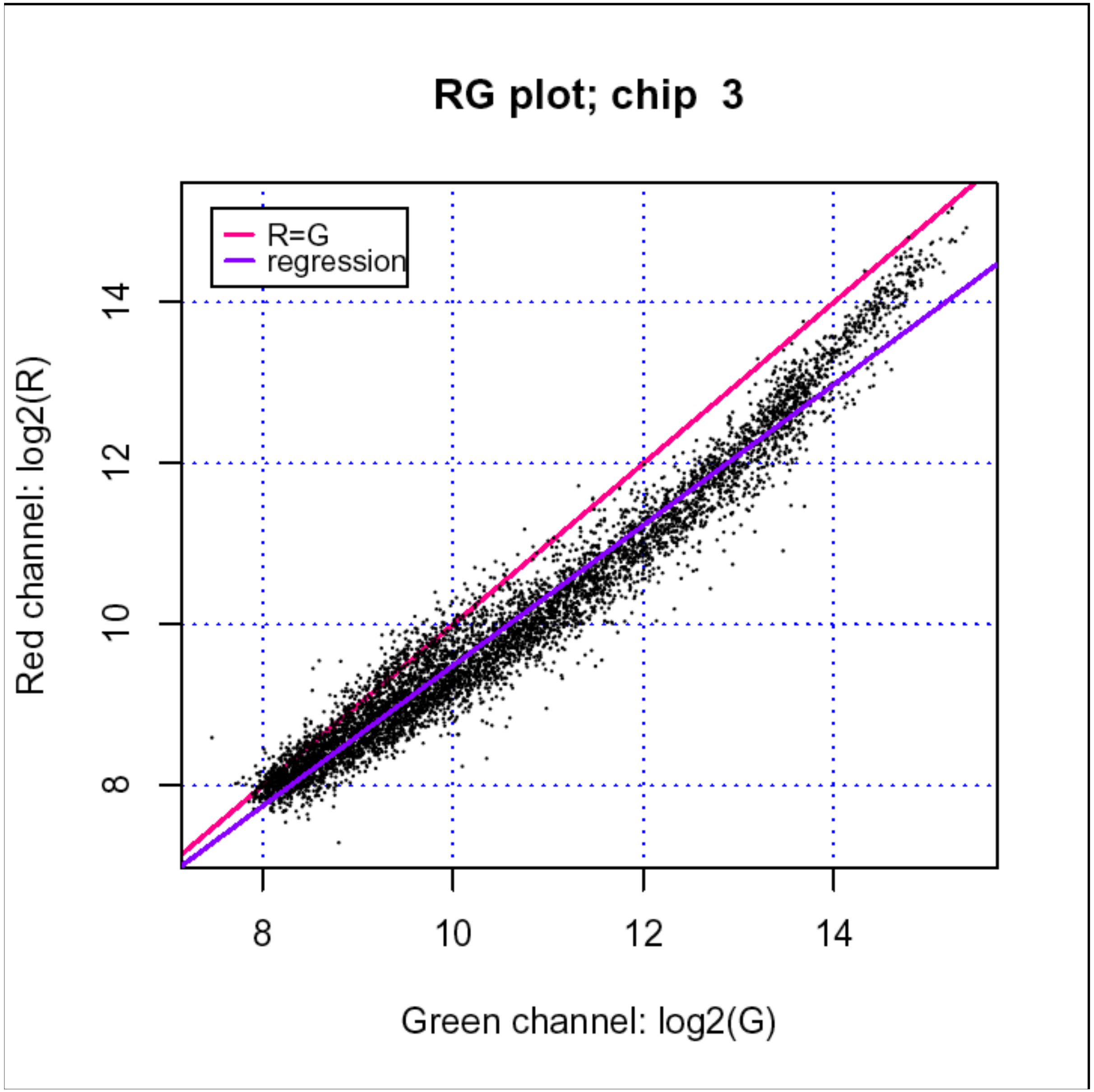


- Channel bias

  This chip is visibly not centred around zero. The negative trends suggests a bias towards green channel.

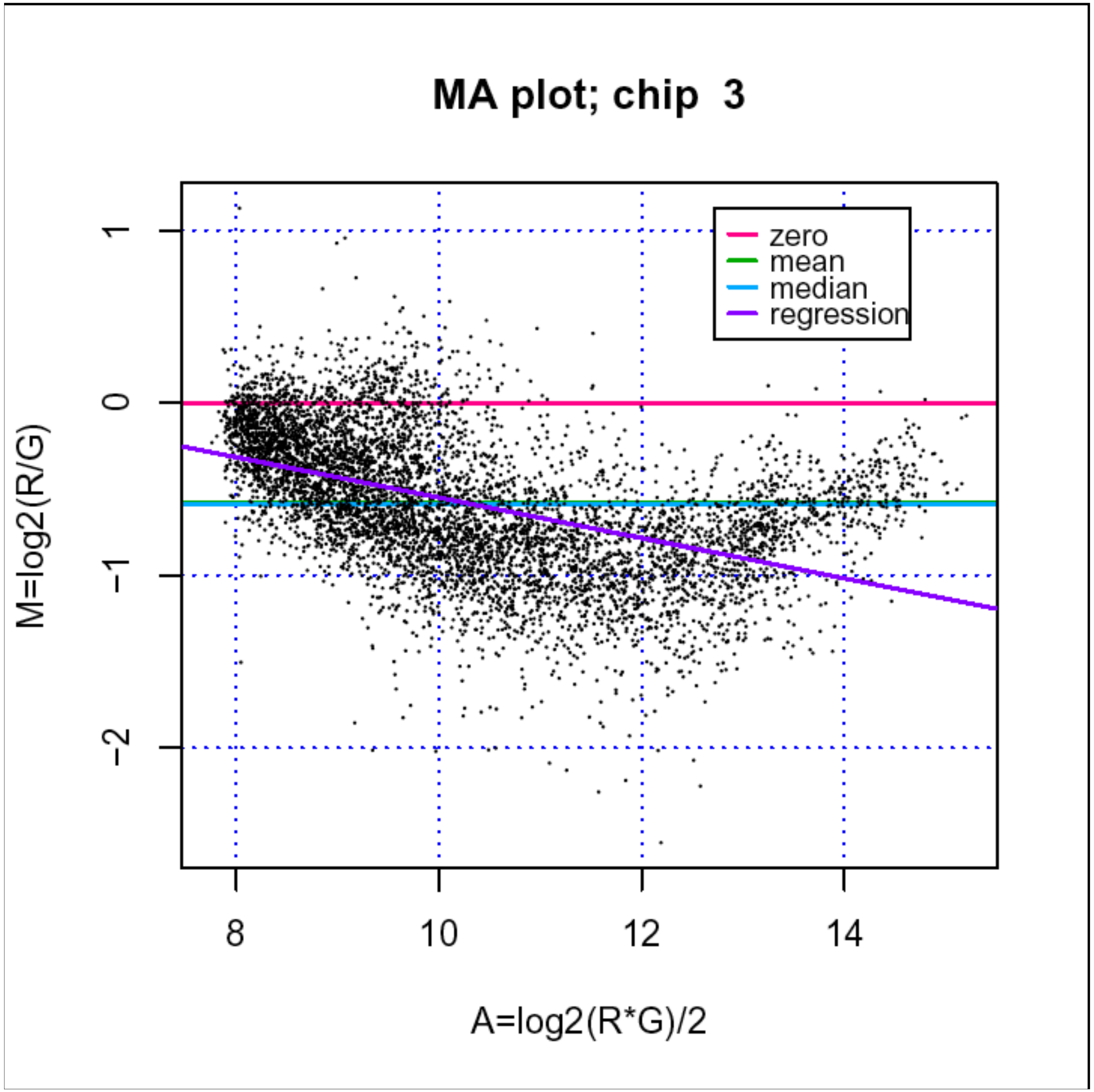
Biases in microarray samples

RG (Red-Green) plot



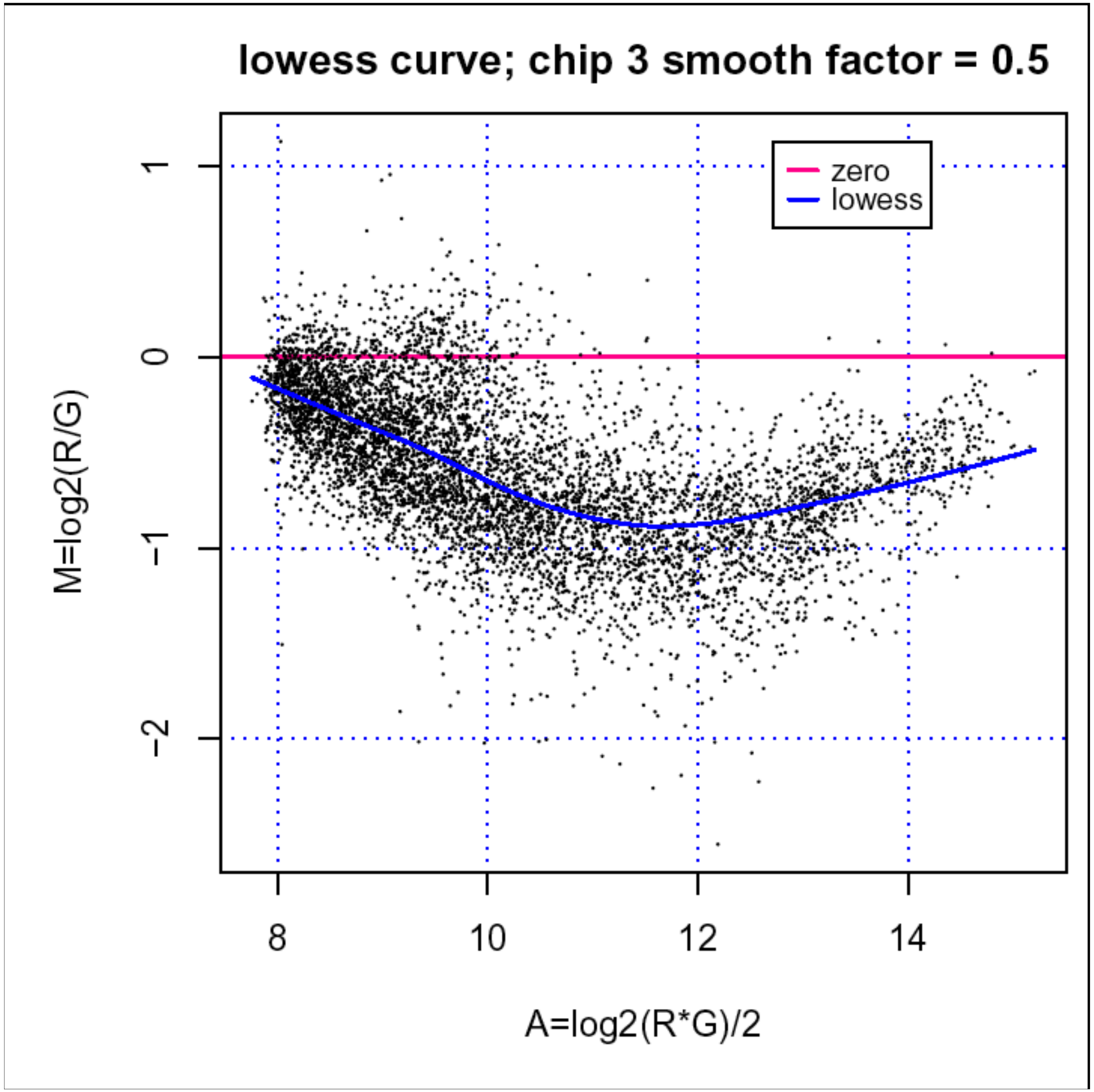
- Channel bias
  - The majority of the dots are below the diagonal.
- Intensity effect
  - The cloud is curved, suggesting a non-linear response of red and/or green channels.
  - A linear regression does not fit well the cloud.

MA plot = RI (Ratio-Intensity) plot



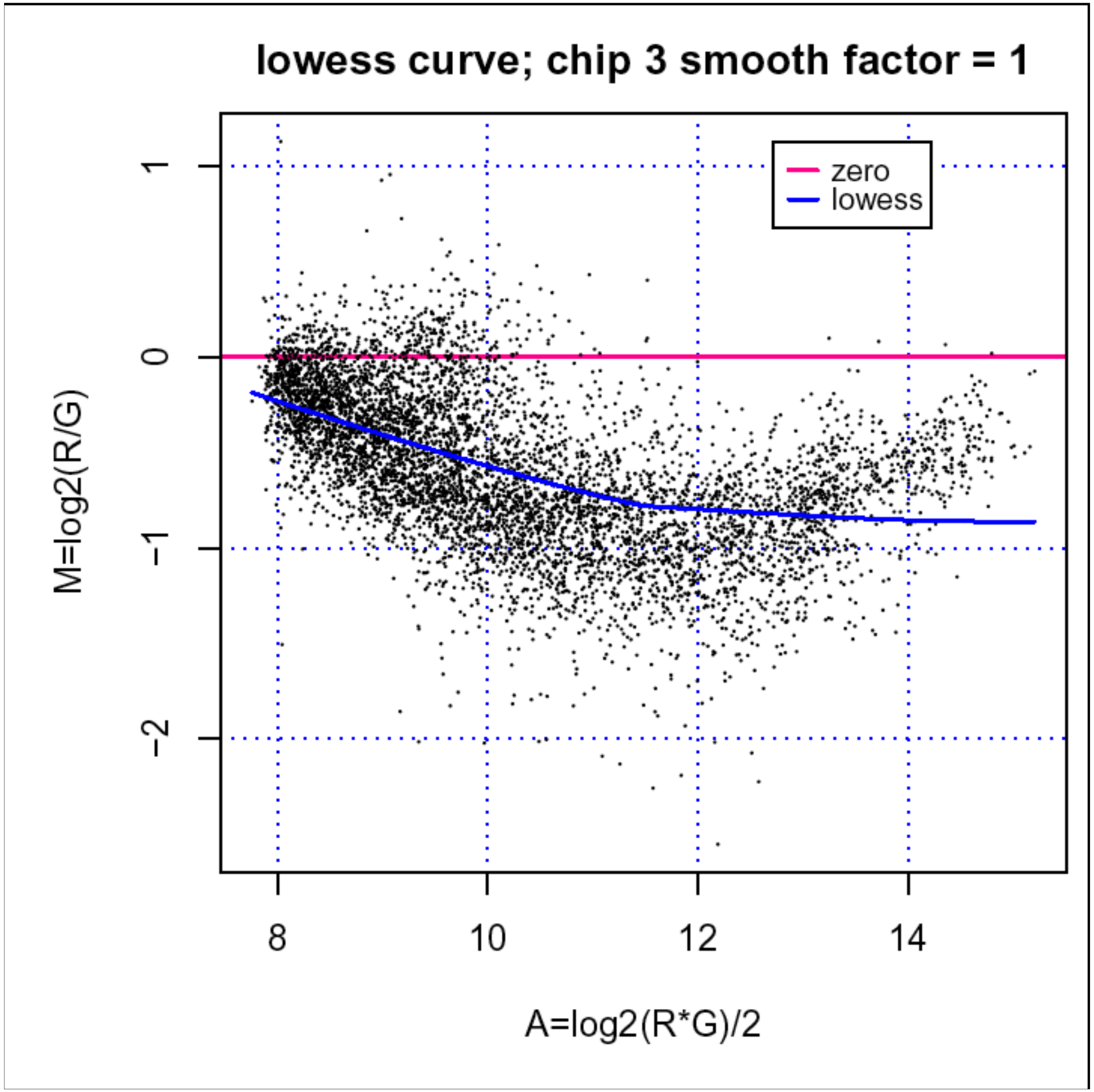
- M is the log-ratio
  - *M = log(R/G)*
- A is the average log intensity
  - *A = log(R*G)/2 =[log(R)+log(G)]/2*
- The MA plot emphasizes the bad centring and the intensity bias.
- Channel bias
  - The mean ratio and median log-ratio differ from 0.
- Intensity effect
  - The cloud is visibly curved
  - The regression line does not fit the cloud.

Locally weighted linear regression (lowess)



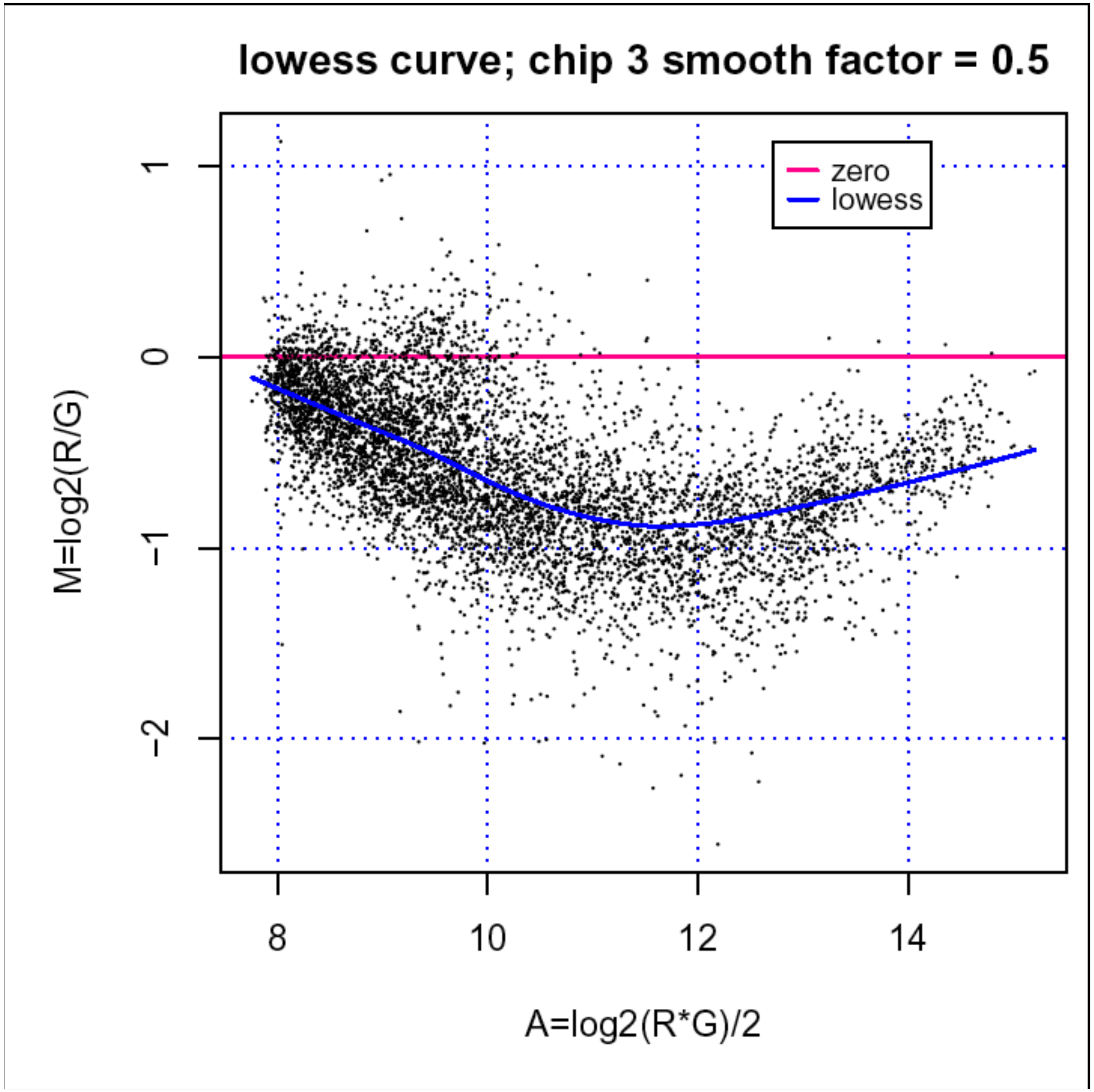
- Locally weighted regression (**LOWESS**) consists in calculating, for each X value, the regression line on the basis of a subset of points around this X value.

Lowess - smooth factor



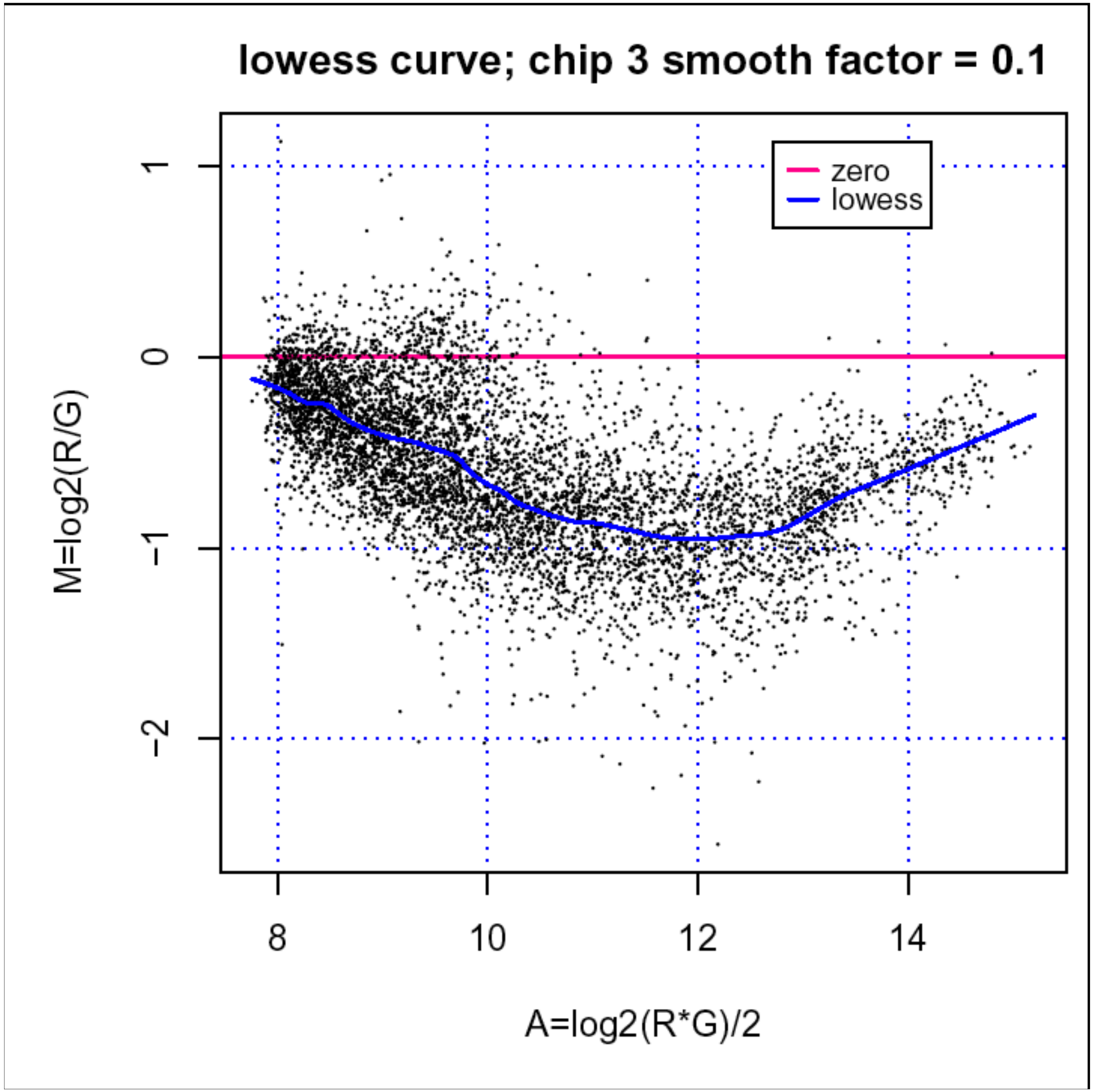
- The main parameter for lowess is the ***smooth factor***, which gives the proportion of points in the plot which influence the smooth at each value. Smaller smooth factor values give a closer fit.
- When the smooth factor is 1, all points influence the regression at each value of the X axis.
- The regression is however not linear, because the influence of a point on another one diminishes with the distance (gaussian kernel).

Lowess - smooth factor



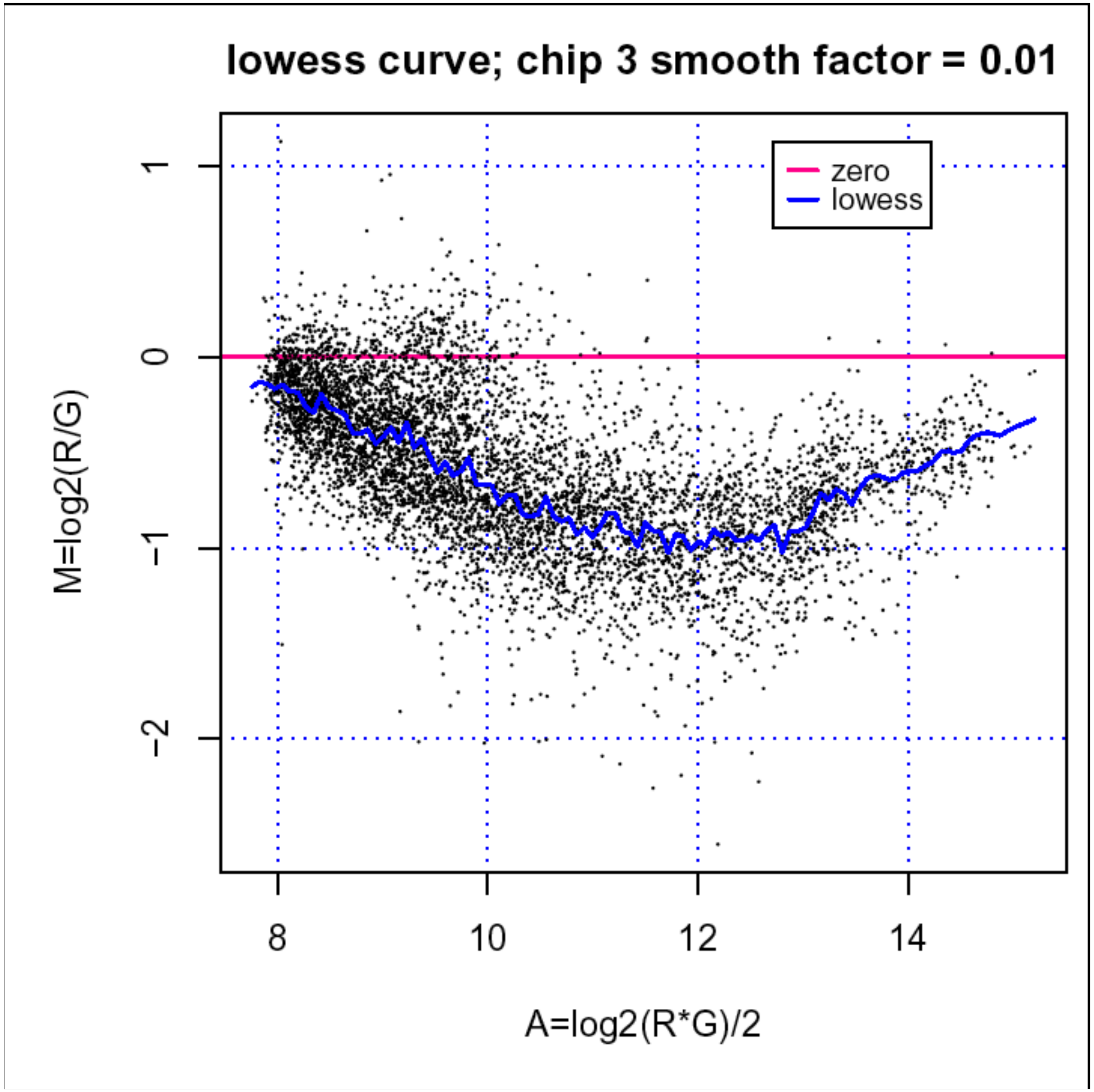
- A smooth factor of 0.5 fits quite well the curve.

Lowess - smooth factor



- With a smooth factor of 0.1, the regression line shows irregularities.

Lowess - smooth factor

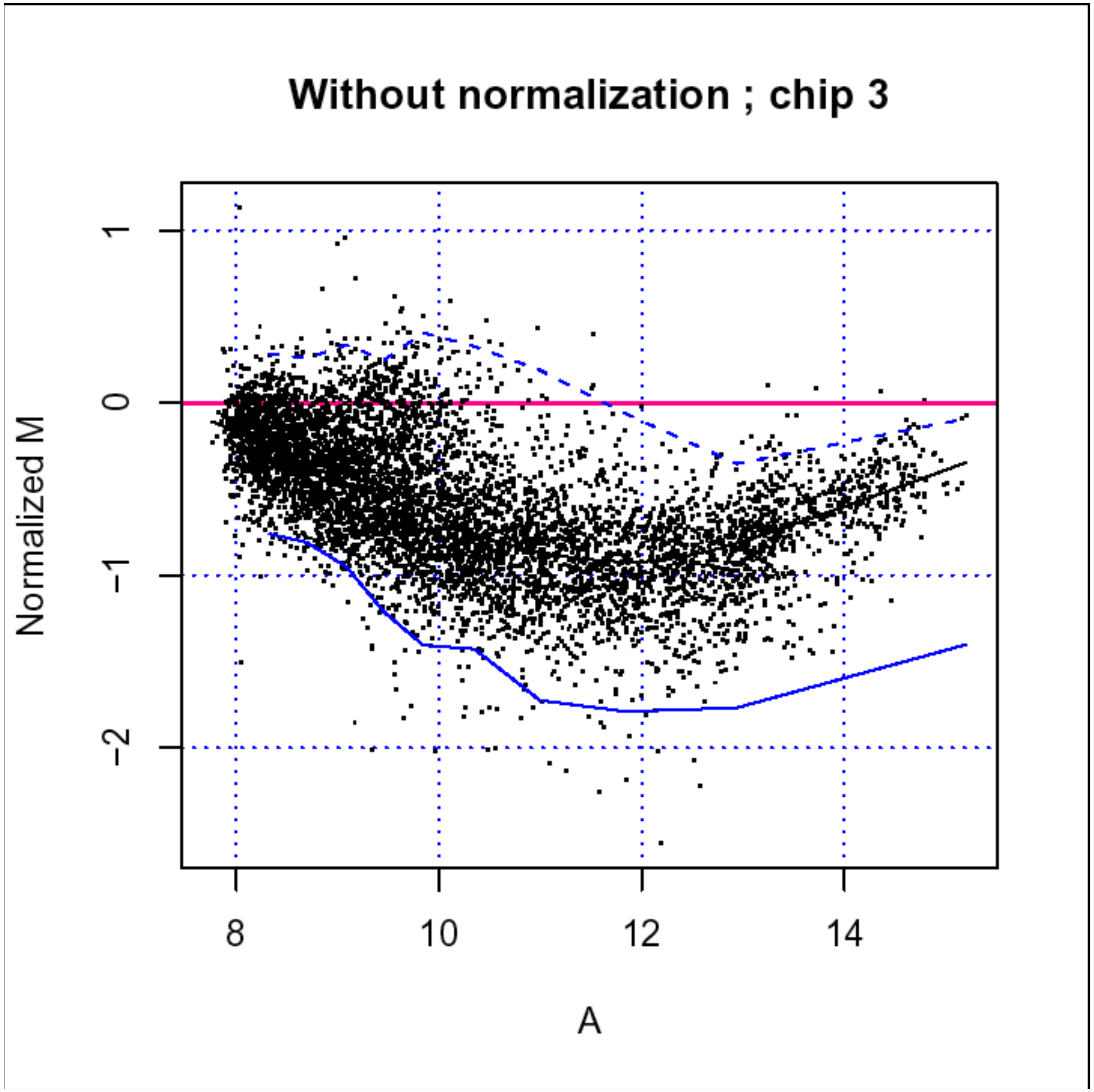


- With a smooth factor of 0.01, the regression line is very irregular.
- At each position, the regression is calculated with a small number of neighbours, and is thus strongly influenced by local fluctuations.
- The curve follows local fluctuations of the cloud, which are likely to reflect random effects rather than intrinsic variations.
- This is a problem of ***over-fitting.***

Normalization methods

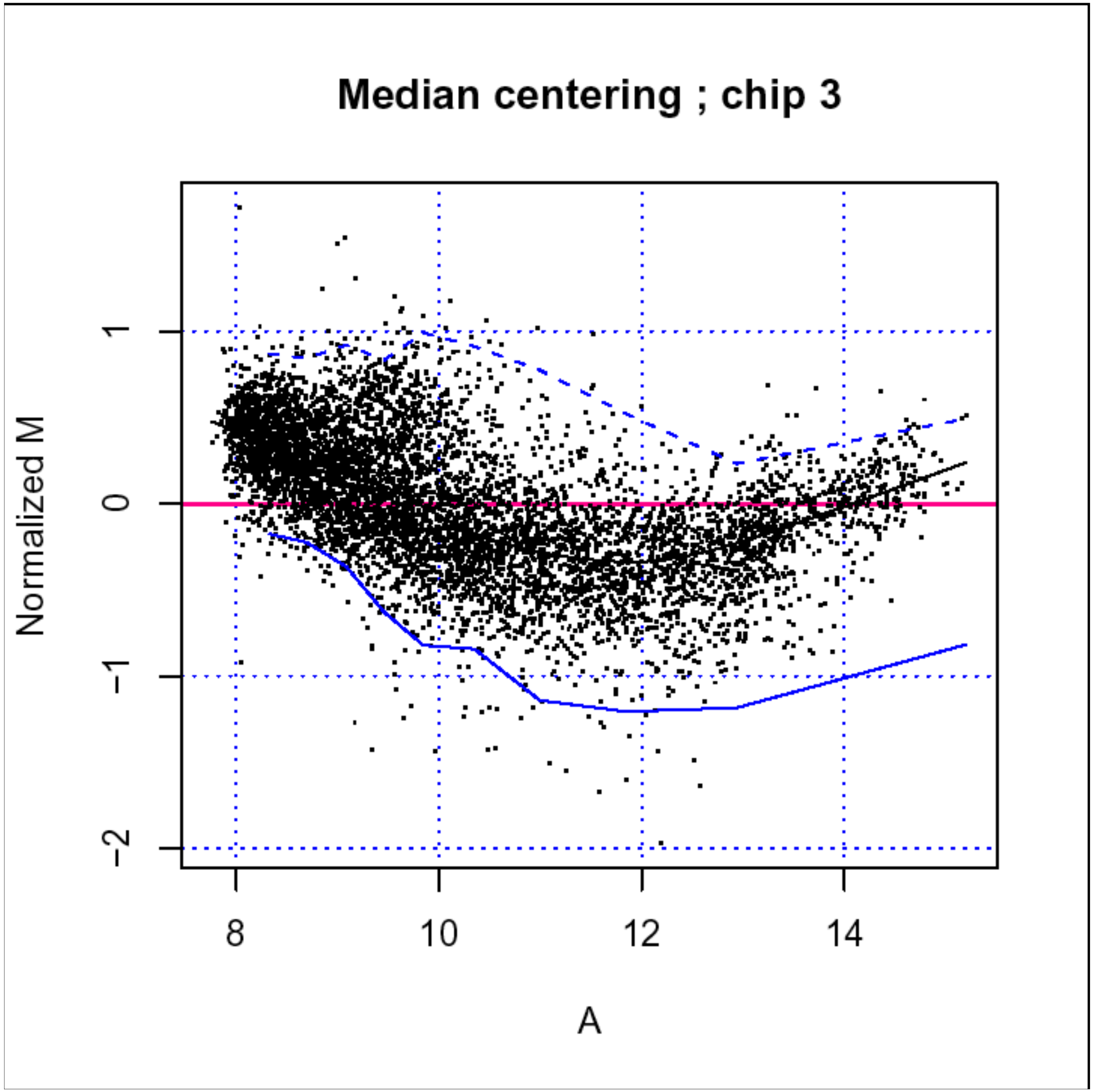
- The BioConductor library sma contains various methods of normalization.
  - Median centring
  - Global (chip-wise) LOWESS
  - Block-wise LOWESS
  - Block-wise LOWESS with scaling
- The function plot.mva() performs either of these normalizations and draws the MA plot of the normalized data.

Without normalization



- In the next slides, we will apply the different normalization methods to the same chip, and comment the result.
- The original chip is
  - biased towards the green channel
  - Intensity-biased

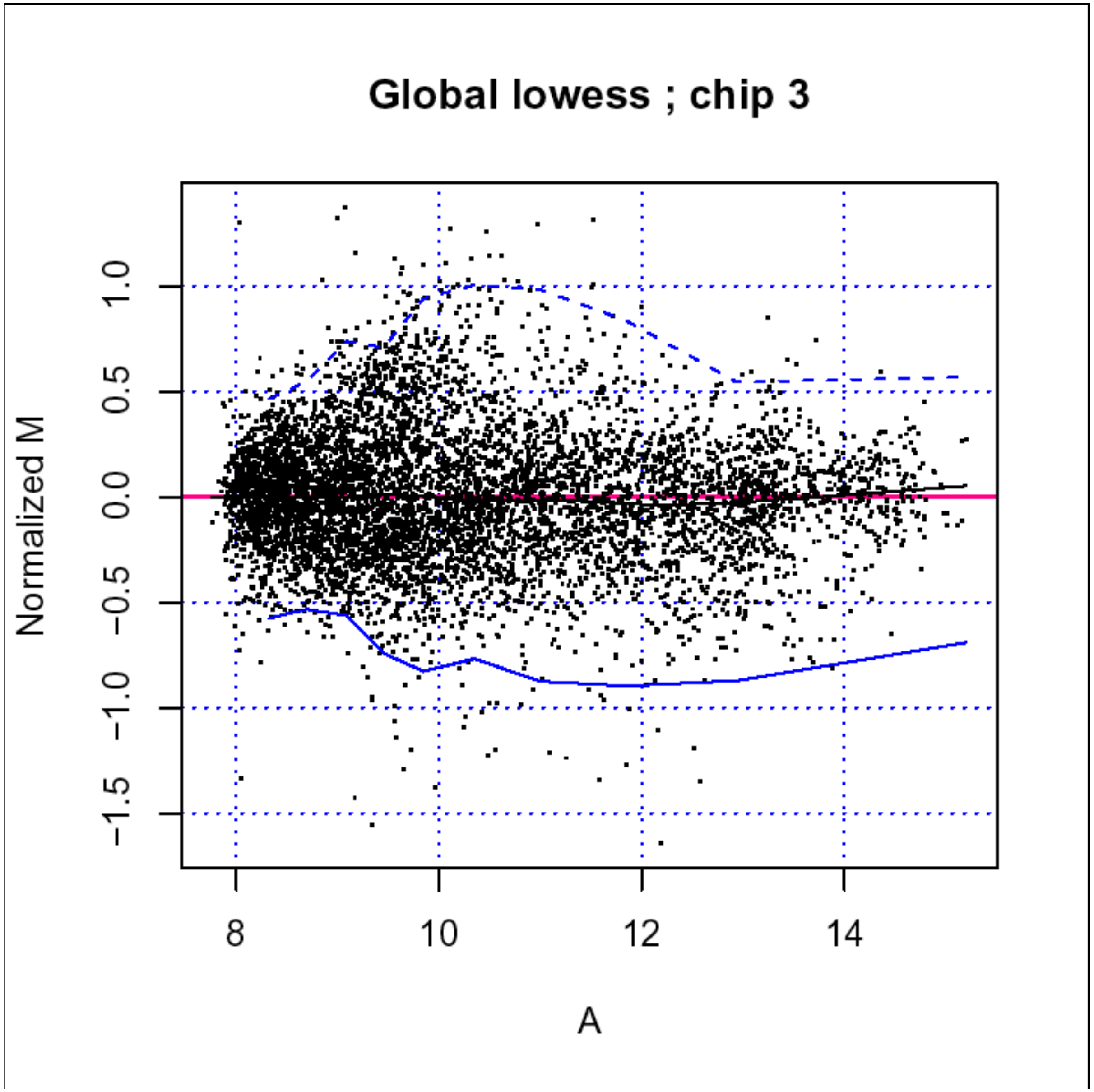
Median centring



- The median of the log-ratios is substracted from each log-ratio value.

$$M_{norm} = M - median(M)$$

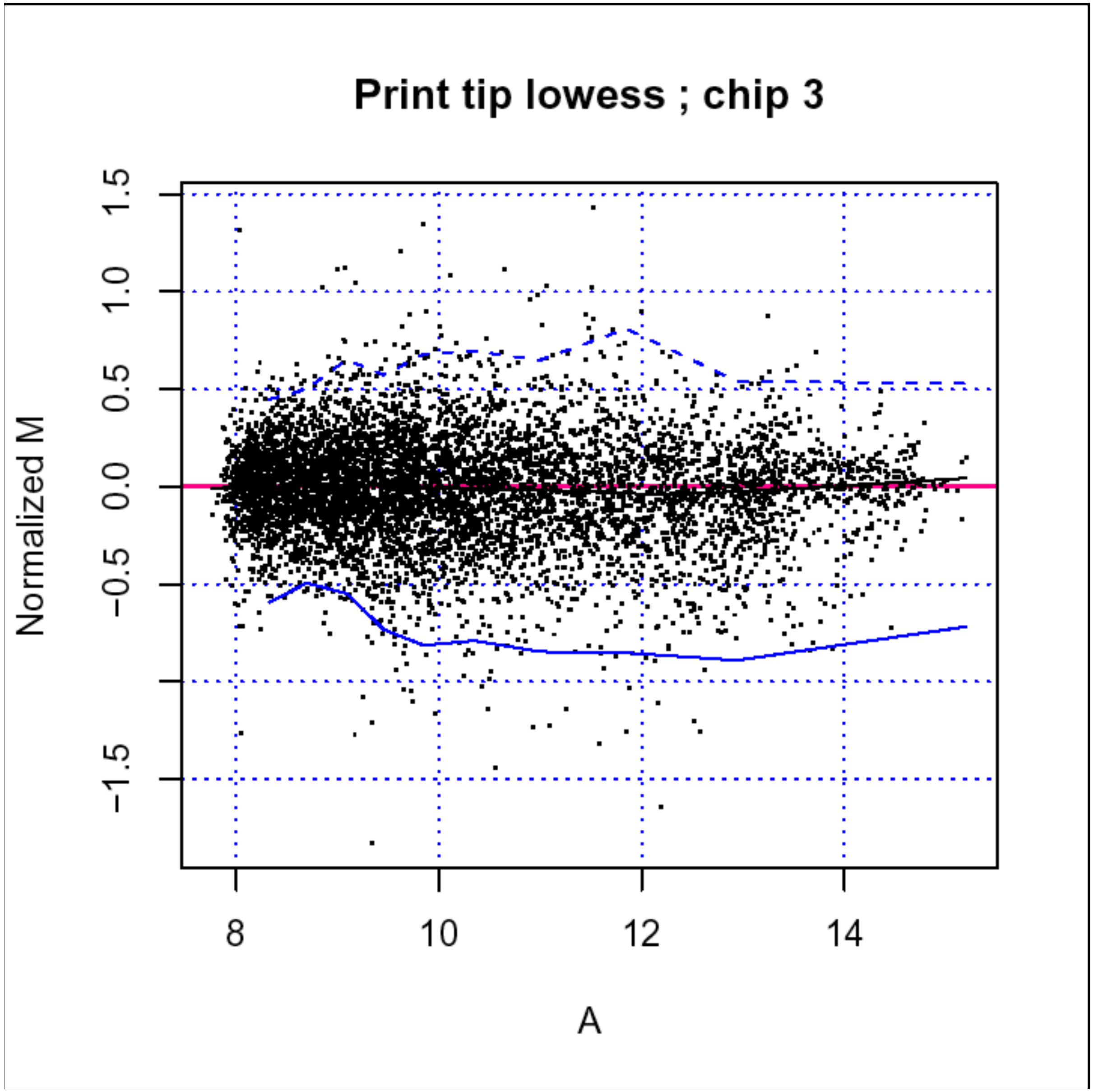
Global (chip-wise) lowess normalization



- For chip-wise LOWESS, a regression curve *y(A)* is calculated with all the spots of the chip.
- The values are normalized by substracting the regression curve from the M value.

$$\boxed{M_{norm} = M - y(A)}$$

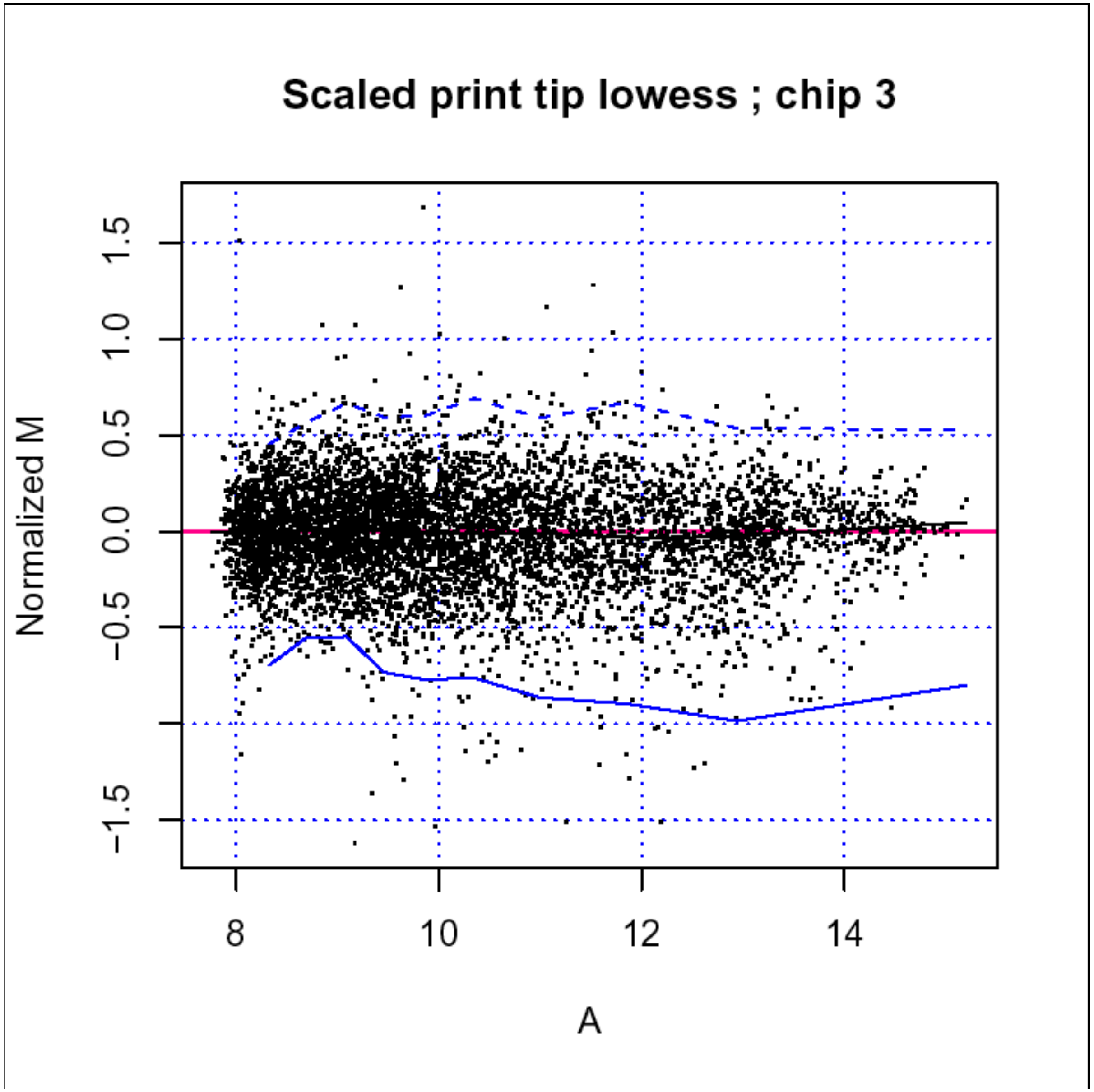
Block-wise lowess (one lowess per print tip)



- For block-wise LOWESS, a regression curve $y_{block}(A)$ is calculated for each block separately.
- The values are normalized by substracting the blcok-specific regression curve from the M value.

$$M_{norm} = M - y_{block}(A)$$

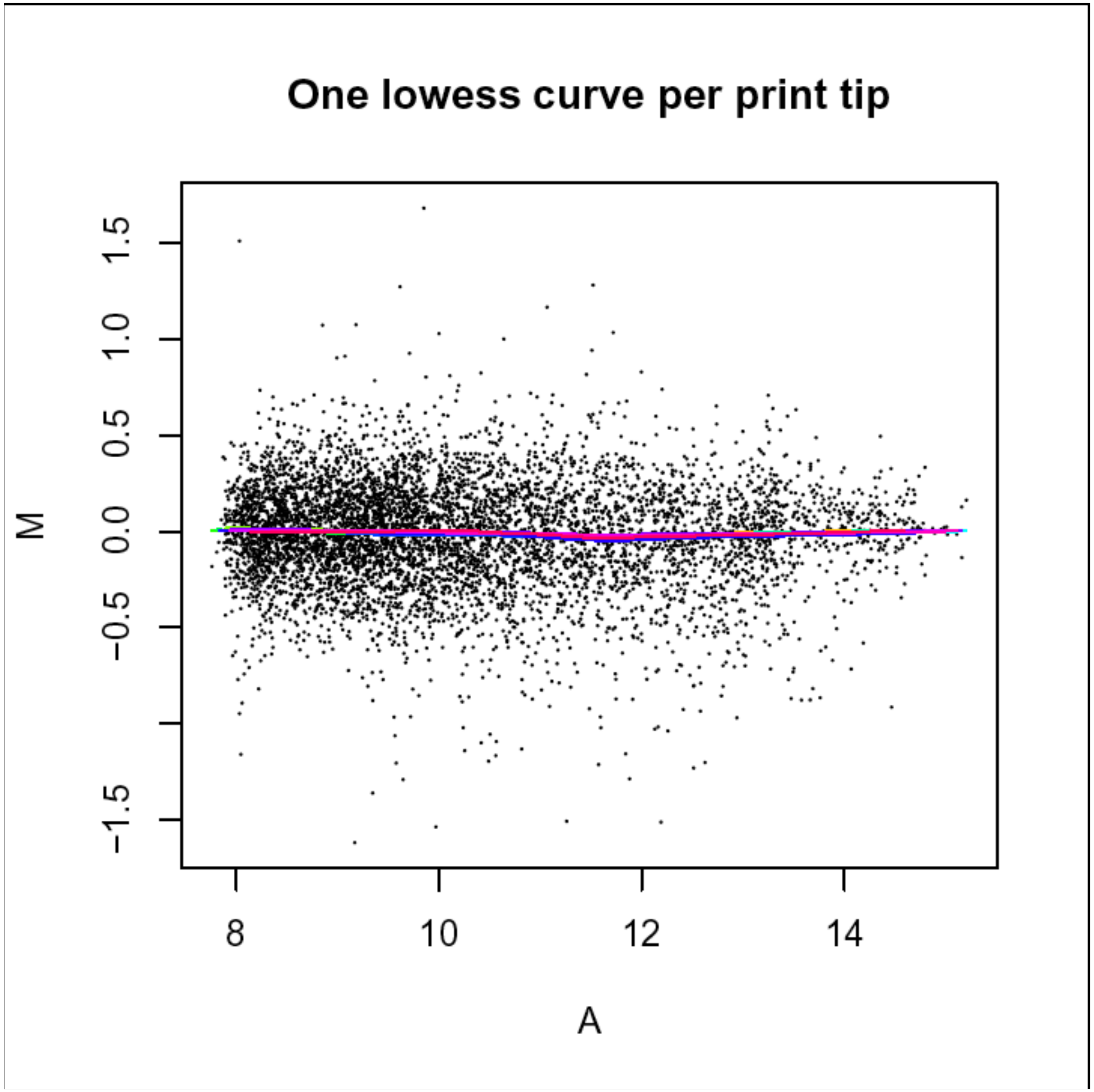
Block-wise lowess with scaling



- The block-wise LOWESS can be combined to a scaling operation: each value is divided by a block-specific estimator of dispersion s_{block}
- Yang et al. (Technical report 589) recommend to use the median absolute deviation (MAD) to estimate the block-specific dispersion.

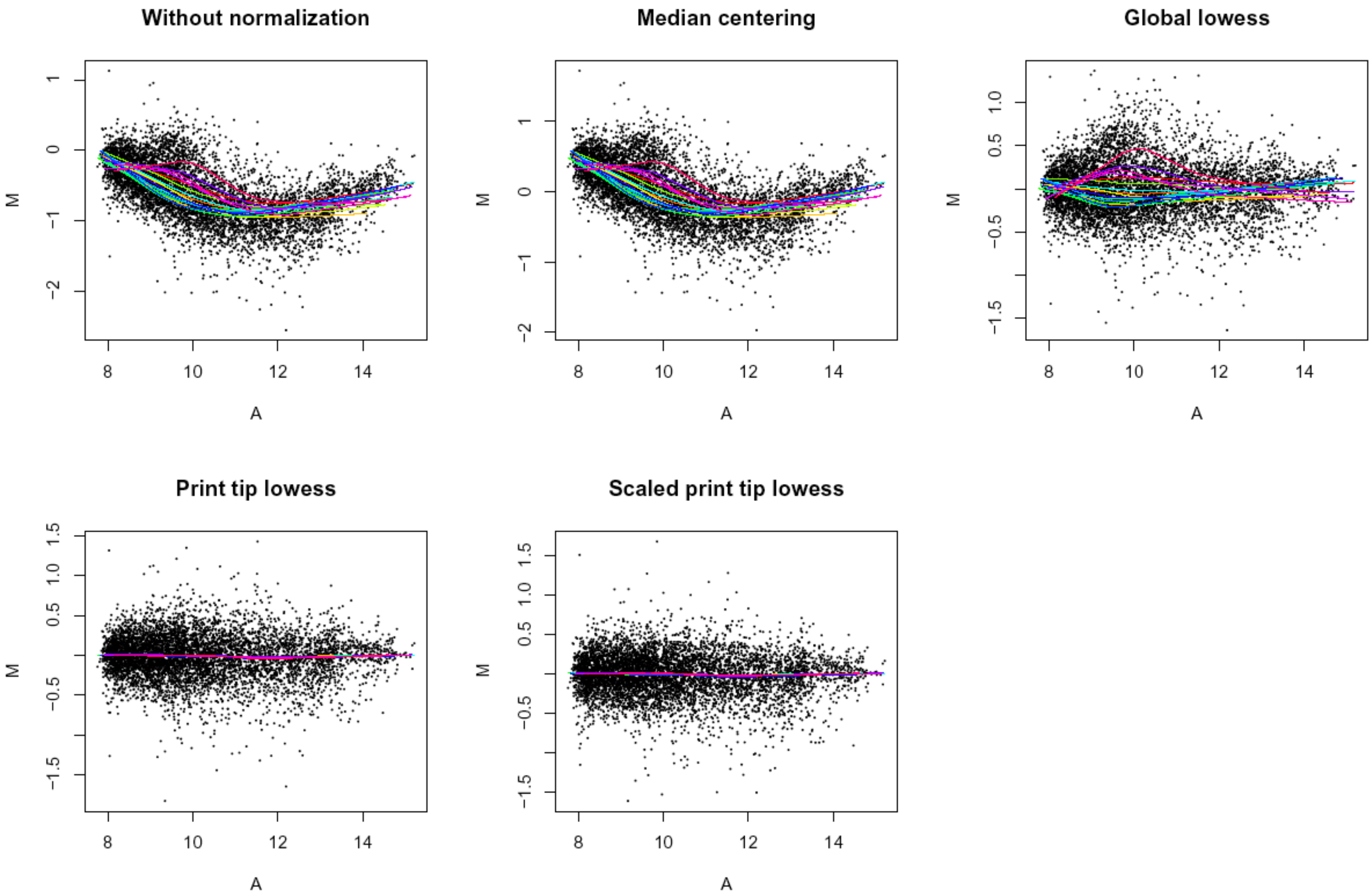
$$M_{norm} = \frac{M - y_{block}(A)}{s_{block}(A)}$$

One lowess curve per print tip



- BIoConductor contains a function plot.print.tip.lowess() which draws a MA plot with one regression curve per print tip
- This function is convenient to check the result of the normalization, and to compare the different normalization methods.

Comparison between normalization methods



Chip-wise selection of significant genes in 2-channel microarrays

Filtering genes on the basis of their log-ratio

- In the first publications on microarray analysis, genes were filtered on the basis of a threshold on the log-ratio. Typically, papers from Stanford were considering as significantly regulated all genes with

R/G	log2(R/G)	regulation
≥ 2	≥ 1	up-regulated
≤ 1/2	≤ -1	down-regulated

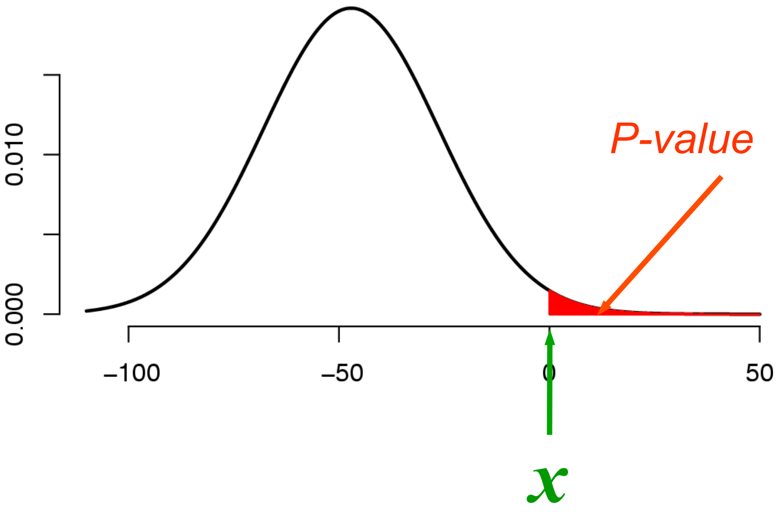
- These thresholds were based on an empirical observation (a control chip). They however suffer from several drawbacks
  - They do not rely on any statistical or probabilistic criterion.
  - They do not take into account the bias in centring.
    - This can be circumvented by first centring each chip independently.
  - They do not take into account the chip-specific dispersion. Among a series, some chips may have a wider dispersion than others, due to experimental bias (scanner setting, problems with dye, ...).
  - A scaling is thus required, but after scaling, the values do not directly represent expression ratios anymore.

Significance testing

- We can evaluate the significance of each observation, by calculating its P-value.

$$\boxed{Pvalue = P\left(X \geq x\right)}$$

- Under the assumption of normality, the P-value can be obtained from z-scores. Z-scores represent the number of standard deviations from the mean.



$$\boxed{\begin{aligned} z &= \left(x - m\right)/s \\ Pvalue &= P\left(Z \geq z\right) \end{aligned}}$$

Bonferoni rule

- Multi-testing
  - Assessing the significance of each gene on a chip represents thousands of simultaneous tests. Let N be the number of genes.
  - The risk of error (P-value) associated to each gene will thus be challenged N times.
  - The significance thresholds used for single testing (0.01, 0.001) are thus likely to return many false positive.
- Bonferoni rule
  - Adapt the threshold to the number of simultaneous tests.

$$\boxed{\alpha \le \frac{1}{N}}$$

E-value

- An alternative but equivalent way t treat the problem of multi-testing is to calculate the expected value for each observation.
- One can then select a threshold on E-value according to the number of false positive considered as acceptable.

$$\boxed{Evalue = Pvalue * N}$$

References

- Normalization
  - Yang, Y. H., Dudoit, S., Luu, P., Lin, D. M., Peng, V., Ngai, J. & Speed, T. P. (2002). Normalization for cDNA microarray data: a robust composite method addressing single and multiple slide systematic variation. Nucleic Acids Res 30(4), e15.
- Data sources
  - Callow, M. J., Dudoit, S., Gong, E. L., Speed, T. P. & Rubin, E. M. (2000). Microarray expression profiling identifies genes with altered expression in HDL-deficient mice. Genome Res 10(12), 2022-9.