Normalization

• **Normalization** is needed to ensure that differences in intensities are indeed due to differential expression, and not some printing, hybridization, or scanning artifact.

• Normalization is necessary before any analysis which involves within or between slides comparisons of intensities, e.g., clustering, testing.

• Somewhat different approaches are used in two-color and one-color technologies

---

Example of Replicate Data

Here different scanners were used

---

Example of Replicate Data

---
Most Common Problem

Intensity dependent effect: Different background level most likely culprit

Scatter Plot

Demonstrates importance of MA plot

Two-color platforms

- Platforms that use printing robots are prone to many systematic effects:
  - Dye
  - Print-tip
  - Plates
  - Print order
  - Spatial
- Some examples follow
Print-tip Effect

Plate effect
Spatial Effect

spotted cDNA arrays, Stanford-type

Spatial Effects

Batches: array to array differences $d_{ij} = \text{mad}(h_i - h_j)$

arrays $i=1..63$; roughly sorted by time
What can we do?

- Throw away the data and start again? Maybe.
- Statistics offers hope:
  - Use control genes to adjust
  - Assume most genes are not differentially expressed
  - Assume distribution of expression are the same

Simplest Idea

- Assume all arrays have the same median log expression or relative log expression
- Subtract median from each array
- In two-color platforms, we typically correct the Ms. Median correction forces the median log ratio to be 0
  - Note: We assume there are as many over-expressed as under-expressed genes
- For Affymetrix arrays we usually add a constant that takes us back to the original range.
  - It is common to use the median of the medians
  - Typically, we subtract in the log-scale
- Usually this is not enough, e.g. it will not account for intensity dependent bias

House Keeping Genes

I rarely find house keeping genes useful
More Elaborate Solutions

- Proposed solutions
  - Force distributions (not just medians) to be the same:
    - Amaratunga and Cabrera (2001)
    - Bolstad et al. (2003)
  - Use curve estimators, e.g. loess, to adjust for the effect:
    - Li and Wong (2001) Note: they also use a rank invariant set
    - Colantuoni et al (2002)
    - Dudoit et al (2002)
  - Use adjustments based on additive/multiplicative model:
    - Rocke and Durbin (2003)

Quantile normalization

- All these non-linear methods perform similarly
- Quantiles is my favorite because its fast and conceptually simple
- Basic idea:
  - order value in each array
  - take average across probes
  - Substitute probe intensity with average
  - Put in original order

Example of quantile normalization

<table>
<thead>
<tr>
<th>Original</th>
<th>Ordered</th>
<th>Averaged</th>
<th>Re-ordered</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 4 14</td>
<td>3 1 8</td>
<td>5 5 5</td>
<td>5 5 8</td>
</tr>
<tr>
<td>4 6 8</td>
<td>1 4 8</td>
<td>4 5 5</td>
<td>6 8 5</td>
</tr>
<tr>
<td>3 5 8</td>
<td>4 5 3</td>
<td>6 6 6</td>
<td>5 6 5</td>
</tr>
<tr>
<td>3 3 3</td>
<td>5 6 14</td>
<td>8 8 8</td>
<td>5 5 6</td>
</tr>
</tbody>
</table>
A worry is that it overcorrects

Two-color Platforms

• Quantile normalization is popular with high-density one channel arrays

• With two-color platforms we have many effects to worry about and seems we should take advantage of the paired structure
ANOVA

- One of the first approaches was to fit ANOVA models to log intensities with a global effect for each Dye
- This does not correct for the non-linear dependence on intensity
- Recent implementations subtract a constant from the original scale to remove the non-linear effect

For references look at papers by Gary Churchill

Different Background

Above is MA for R=50+S, G=100+S

Correcting M approaches

- Most popular approach is to correct M directly
- We assume that we observer M + Bias and that Bias depends on Intensity (A), print-tip, plate, spatial location, etc...
- Idea: Estimate bias and remove it
- For continuous variables we assume the dependence is smooth and use loess to estimate them
- The normalized M is M - estimated Bias
- Most versatile method

For details look for papers by Terry Speed and Gordon Smyth
Example: Intensity Effect

• The most common problem is intensity dependent effects
  – Probably due to different background
• Loess is used to estimate and remove this effects

Loess
Error model approaches

- Error model approaches describe the need for normalization with an additive background plus stochastic multiplicative error model.
- From this model an variance stabilizing transformation is obtained.
- Log ratios are no longer the measure of differential expression.

For details see papers by Wolfgang Huber and David Rocke.
Error models

Describe the possible outcomes of a set of measurements

Outcomes depend on:
- true value of the measured quantity
  (abundances of specific molecules in biological sample)
- measurement apparatus
  (cascade of biochemical reactions, optical detection system with laser scanner or CCD camera)

The two component model

measured intensity = offset + gain × true abundance

\[ y_{ik} = a_{ik} + b_{ik} x_k \]

- \( a_{ik} = a_i + e_{ik} \)
  - per-sample offset
  - \( e_k \sim N(0, b_k^2 s^2) \)
  - “additive noise”
- \( b_{ik} = b_i b_k \exp(h_{ik}) \)
  - per-sample normalization factor
  - \( h_k \sim N(0, s^2) \)
  - “multiplicative noise”
The two-component model

![Graph showing two types of noise: additive and multiplicative.](image)

B. Durbin, D. Rocke, JCB 2001

Parameterization

\[
y = a + \varepsilon + b \cdot x \cdot (1 + \eta)
\]

Two practically equivalent forms \((h \ll 1)\):

\[
y = a + \varepsilon + b \cdot x \cdot e^\eta
\]

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a)</td>
<td>systematic background with the same values for all probes ((\text{per array} \times \text{color})) per array \times \text{color} \times \text{print-tip group}</td>
</tr>
<tr>
<td>(\varepsilon)</td>
<td>random background (\text{iid} ) in whole experiment (\text{iid per array})</td>
</tr>
<tr>
<td>(b)</td>
<td>systematic gain factor (\text{per array} \times \text{color} \times \text{print-tip group})</td>
</tr>
<tr>
<td>(h)</td>
<td>random gain fluctuations (\text{iid} ) in whole experiment (\text{iid per array})</td>
</tr>
</tbody>
</table>

Important issues for model fitting

Parameterization

- variance vs bias
- "Heteroskedasticity" (unequal variances)
  - weighted regression or variance stabilizing transformation
Outliers
  - use a robust method
Algorithm
  - If likelihood is not quadratic, need non-linear optimization. Local minima / concavity of likelihood?
\( X_u \) a family of random variables with 
\( E X_u = u, \) \( \text{Var} X_u = v(u) \). Define 
\[
 f(x) = \int x \frac{1}{\sqrt{v(u)}} \, du 
\]

derivation: linear approximation 
\( \Rightarrow \text{var} \ f(X_u) \sim \text{independent of } u \)
the “glog” transformation

\[ f(x) = \log(x) \]
\[ h_s(x) = \text{asinh}(x/s) \]

\[ \text{arsinh}(x) = \log\left(x + \sqrt{x^2 + 1}\right) \]

\[ \lim_{s \to \infty} (\text{arsinh} x - \log x - \log 2) = 0 \]

---

the transformed model

\[ \text{arsinh} \left( \frac{Y_{ki}}{b_{si}} - a_{si} \right) = \mu_k + \varepsilon_{ki} \]

\[ \varepsilon_{ki} \sim \mathcal{N}(0, \sigma^2) \]

- i: arrays
- k: probes
- s: probe strata (e.g. print-tip, region)
**profile log-likelihood**

\[ pll(a, b) = \sup_{c, \mu} ll(a, b, c, \mu) \]

Here:

\[ pll(a_1, b_1, \ldots, a_d, b_d) = \]

\[ -n d \log \sigma + \sum_{k=1}^{d} \sum_{i=1}^{n} \log h(y_{ik}) \]

\[ = -\frac{nd}{2} \log \left( \sum_{k=1}^{d} \sum_{i=1}^{n} (h(y_{ik}) - \mu_k)^2 \right) + \sum_{k=1}^{d} \sum_{i=1}^{n} \log h(y_{ik}) \]

---

**Least trimmed sum of squares regression**

Least trimmed sum of squares regression

- least sum of squares
- least trimmed sum of squares

\[ \text{minimize} \quad \sum_{i=1}^{n/2} (y_{ik} - f(x_{ik}))^2 \]

---

**“usual” log-ratio**

\[ \log \frac{x_1}{x_2} \]

**’glog’ 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**

Estimated log-fold-change vs Signal intensity

**Variance-bias trade-off and shrinkage estimators**

Shrinkage estimators:
- 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:
- a shrinkage estimator for fold change

There are many possible choices, we chose “variance-stabilization”:
- interpretable even in cases where genes are off in some conditions
- can subsequently use standard statistical methods (hypothesis testing, ANOVA, clustering, classification...) without the worries about low-level variability that are often warranted on the log-scale

**“Single color normalization”**

n red-green arrays \((R_1, G_1, R_2, G_2, \ldots R_n, G_n)\) within/between slides

for \((i=1:n)\)
- calculate \(M_i = \log(R_i/G_i), A_i = \frac{1}{2} \log(R_i*G_i)\)
- normalize \(M_i\) vs \(A_i\)
- normalize \(M_1, M_2, \ldots M_n\) all at once
- normalize the matrix of \((R, G)\)
- then calculate log-ratios or any other contrast you like
Back to you Rafa!

Concluding Remarks

- Notice Normalization and background correction are related
- Current procedures are based on assumptions
- Many new problems clearly violate these assumptions
- We will discuss this problem in another lecture