PREPROCESSING MICROARRAY DATA

Background correction
Normalization
Summarization
Transforms
Microarray studies life cycle

- Quality Measurement
- Experimental design
- Microarray experiment
- Image analysis
- Normalization
- Analysis
- Clustering
- Discrimination
- Biological verification and interpretation

Here we are
Objective

• Achieve a measurement scale such that
  – It has the same origin (zero or other) for all spots
  – It uses the same unit for all spots and microarrays
  – It has a linear relationship with the DNA/RNA biological
  – It has good statistical properties (good for later analyses)

• Deal with the particular characteristics of each platform and experiment
  – Color differences
  – Reference sample
  – Summarize information of each gene
  – Deal with platform characteristics (e.g. “probesets/probepairs”)
Hypotheses

• Most normalization methodologies make two major assumptions about the data.
  
  – When comparing different samples, only few genes are over-expressed or under-expressed in one array relative to the others.
  
  – The number of genes over-expressed in a condition is similar to the number of genes under-expressed.
  
• This assumptions should agree with your experimental context.
General Steps

• **Background correction** (correcting the scale origin for spots)

• Normalization (standardizing the scale unit - rescaling)

• Adjustments characteristics of each platform or experiment
  – Perfect-Match Mismatch Adjustment (Affymetrix)
  – Correcting for different dye properties (in two color arrays)
  – Adjustments depending on the DNA strands

• Summary of information from several spots into a single measure for each gene
  – Averaging Affymetrix ”probe sets”
  – Averaging duplicated spots
  – Calculating ratios
  – Taking logarithms
Preprocessing two color arrays

• Background correction
  • Scanners: separate Signal (Rs, Gs) and Background (Rb, Gb) estimates.
  • Background corrected estimates (Rc, Gc)
    • Rc = Rs – Rb,   Gc = Rs – Rb, OR (better)
    • Rc = max(Rs -Rb, 0), Gc = max(Gs -Gb, 0)

• Summarization & Transforms: log-Ratios
  • Estimate relative expression as
    log(Rc/Gc)
Global normalization

• Based on a global adjustment
  
  \[ \log_2 R/G \rightarrow \log_2 R/G - c = \log_2 R/(kg) \]

• Choices for \( k \) or \( c = \log_2 k \) are
  
  – \( c = \) median or mean of log ratios for
    
    – A particular gene set
    
    – All genes or control or housekeeping genes.
  
  – Total intensity normalization, where
    
    \[ k = \frac{\sum R_i}{\sum G_i}. \]
Example: (Callow et al 2002)
Global median normalization.
Intensity-dependent normalization

• Run a line through the middle of the MA plot, shifting the M value of the pair (A,M) by $c = c(A)$, i.e.
  \[ \log_2 \frac{R}{G} \rightarrow \log_2 \frac{R}{G} - c(A) \]
  \[ = \log_2 \frac{R}{(k(A)G)}. \]

• One estimate of $c(A)$ is made using the LOWESS function of Cleveland (1979): LOcally WEighted Scatterplot Smoothing.
Example: (Callow et al 2002) loess vs median normalization.
Effect of within-slide normalization

M box plot for all arrays

M box plot for all arrays. Normalization within slides only
Effect of between-slide normalization

M box plot for all arrays. Normalization within slides only

M box plot for all arrays. Normalization within & between slide
Preprocessing one color data

- Many methods have been developed to preprocess affymetrix arrays.
  - Current methods: GCRMA, PLIER
  - Popular methods: RMA and MAS5
  - Rudimentary methods: MAS4, LOESS
MAS 4: Averaging absolute differences

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

- Ignore pair deviating more than $3\sigma$ from $\mu$
- Many known problems
  - $1/3$ of MM are bigger than PM
  - There may appear negative MM
  - Using MMs adds noise
- Been substituted by other (→ MAS5 → PLIER)
MicroArray Suite 5.0 (i)

- Relies on a robust statistic (Tukey's biweight) to:
  - Weight background effect and
  - Estimate signal

- Tukey's biweight weights each value according to its distance to the median
  - Central location estimate
  - Outliers adjustment
Problems with MAS4.0 (& MAS5.0)

• Loss of probe-level information.
• Background estimate may cause noise at low intensity levels due to subtraction of MM data.
Robust Multiarray Average (RMA)

Subtraction of MM data performed by MAS
  – corrects for NSB, but
  – introduces noise.

Need a method that gives positive intensity values.

Normalising at probe level avoids the loss of information.
Robust Multiarray Average (RMA)

1) **Background correction.**
2) Normalization (across arrays).
3) Probe level intensity calculation.
4) Probe set summarization.
Robust Multiarray Average (RMA)

Assumes PM data is combination of background and signal

- $PM = Signal + Background$, where
  - Signal: $S \sim \exp(\lambda)$ and
  - Background: $B \sim N(\mu, \sigma^2)$

By assuming strictly positive distribution for signal background corrected signal is also positively distributed.

Background correction performed on each array separately.
RMA: background correction (2)

\[ E(S \mid PM) = PM - \mu - \lambda \sigma^2 \]

\[ + \sigma \frac{\phi\left(\frac{PM - \mu - \lambda \sigma^2}{\sigma}\right) - \phi\left(\frac{\mu + \lambda \sigma^2}{\sigma}\right)}{\Phi\left(\frac{PM - \mu - \lambda \sigma^2}{\sigma}\right) - \Phi\left(\frac{\mu + \lambda \sigma^2}{\sigma}\right) - 1} \]

Probability density for a N(0,1)

Distribution function for a N(0,1)

Estimate \( \mu, \sigma, \) and \( \lambda \) separately in each chip using the observed distribution of PMs.

By introducing them in the above formula we obtain an estimate of \( E(S\mid PM) \) for each PM value. These will be the background-adjusted values.
Robust Multiarray Average (RMA)

1) Background correction.

2) **Normalization (across arrays).**

   3) Probe level intensity calculation.

   4) Probe set summarization.
Robust Multiarray Average (RMA)

- Normalises across all arrays to make all distributions the same.
- ‘Quantile Normalization’ used to correct for array biases.
- Compares expression levels between arrays for various quantiles.
- Can view this on quantile-quantile plot.
- Protects against outliers.
Quantile normalization outlined
### Background-Corrected and Log-Transformed Perfect-Match Intensity

<table>
<thead>
<tr>
<th>Probe Set</th>
<th>Probe</th>
<th>GeneChip 1</th>
<th>GeneChip 2</th>
<th>GeneChip 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>7</td>
<td>9</td>
<td>19</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>14</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>2</td>
<td>6</td>
<td>11</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>4</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>10</td>
<td>11</td>
<td>16</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>12</td>
<td>10</td>
<td>15</td>
</tr>
</tbody>
</table>

Replace the highest value on each chip with the average of the highest values.

\[(12+11+19)/3=14\]

### Background-Corrected and Log-Transformed Perfect-Match Intensity

<table>
<thead>
<tr>
<th>Probe Set</th>
<th>Probe</th>
<th>GeneChip 1</th>
<th>GeneChip 2</th>
<th>GeneChip 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>7</td>
<td>9</td>
<td><strong>14</strong></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td><strong>14</strong></td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>2</td>
<td>6</td>
<td>11</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>4</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>10</td>
<td><strong>14</strong></td>
<td>16</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td><strong>14</strong></td>
<td>10</td>
<td>15</td>
</tr>
</tbody>
</table>
Replace the second highest value on each chip with the average of the second highest values.

\[
(10+10+16)/3 = 12
\]

<table>
<thead>
<tr>
<th>Probe Set</th>
<th>Probe</th>
<th>GeneChip 1</th>
<th>GeneChip 2</th>
<th>GeneChip3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>7</td>
<td>9</td>
<td>14</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>14</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>2</td>
<td>6</td>
<td>11</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>4</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>12</td>
<td>14</td>
<td>12</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>14</td>
<td>12</td>
<td>15</td>
</tr>
</tbody>
</table>

Replace the third highest value on each chip with the average of the third highest values.

\[
(7+9+15)/3 = 10.33333
\]

<table>
<thead>
<tr>
<th>Probe Set</th>
<th>Probe</th>
<th>GeneChip 1</th>
<th>GeneChip 2</th>
<th>GeneChip3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>10.33333</td>
<td>10.33333</td>
<td>14</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>14</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>2</td>
<td>6</td>
<td>11</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>4</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>12</td>
<td>14</td>
<td>12</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>14</td>
<td>12</td>
<td>10.33333</td>
</tr>
</tbody>
</table>
Replace the fourth highest value on each chip with the average of the fourth highest values.
\[(4+8+14)/3=8.66667\]

<table>
<thead>
<tr>
<th>Probe Set</th>
<th>Probe</th>
<th>GeneChip 1</th>
<th>GeneChip 2</th>
<th>GeneChip 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>10.33333</td>
<td>10.33333</td>
<td>14</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>8.66667</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>2</td>
<td>6</td>
<td>11</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>8.66667</td>
<td>8.66667</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>12</td>
<td>14</td>
<td>12</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>14</td>
<td>12</td>
<td>10.33333</td>
</tr>
</tbody>
</table>

Replace the fifth highest value on each chip with the average of the fifth highest values.
\[(3+6+11)/3=6.66667\]

<table>
<thead>
<tr>
<th>Probe Set</th>
<th>Probe</th>
<th>GeneChip 1</th>
<th>GeneChip 2</th>
<th>GeneChip 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>10.33333</td>
<td>10.33333</td>
<td>14</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>6.66667</td>
<td>5</td>
<td>8.66667</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>2</td>
<td>6.66667</td>
<td>6.66667</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>8.66667</td>
<td>8.66667</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>12</td>
<td>14</td>
<td>12</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>14</td>
<td>12</td>
<td>10.33333</td>
</tr>
</tbody>
</table>
Robust Multiarray Average (RMA)

1) Background correction.
2) Normalization (across arrays).

3) **Probe level intensity calculation.**
4) Probe set summarization.
Robust Multiarray Average (RMA)

Linear model.
Uses background corrected, normalised, log transformed probe intensities ($Y_{ijn}$).

$$Y_{ijn} = \mu_{in} + \alpha_{jn} + \varepsilon_{ijn}$$

$\mu_{in}$ = Log scale expression level (RMA measure).

$\alpha_{jn}$ = Probe affinity affect.

$\varepsilon_{ijn}$ = Independent identically distributed error term (with mean 0).
Robust Multiarray Average (RMA)

1) Background correction.
2) Normalization (across arrays).
3) Probe level intensity calculation.
4) **Probe set summarization.**
Robust Multiarray Average (RMA)

• Combine intensity values from the probes in the probe set to get a single intensity value for each gene (probeset).

• Uses ‘Median Polishing’.
  • Each chip normalised to its median.
  • Each gene normalised to its median.
  • Repeated until medians converge.
  • Maximum of 5 iterations to prevent infinite loops.
RMA: Median polish

- Given a probe set with \( J \) probe pairs, let \( y_{ij} \) be the background corrected, logarithmically adjusted and quantile-normalized value of chip \( i \) y and probe \( j \).

- Let's assume that \( y_{ij} = \mu_i + \alpha_j + e_{ij} \) where \( \alpha_1 + \alpha_2 + \ldots + \alpha_n = 0 \).

The idea is to estimate the errors by “median polishing” and then subtract the estimated errors to obtain adjusted probe summaries.
RMA: Median polish

- Let $y_{ij}$ be the adjusted value that will be obtained after polishing medians.
- Let $\alpha_j = y_j - y_\cdot$ where $y_j = \Sigma_i y_{ij}$, $y_\cdot = \Sigma_i \Sigma_j y_{ij}$, ("I" is the number of chips).
- Sea $\mu_i = y_{i.} = \Sigma_j y_{ij} / J$
- $\mu_i$ is the expression measure for each probe from chip $i$. 
An Example

Suppose the following are background-adjusted, log$_2$-transformed, quantile-normalized PM intensities for a single probe set. Determine the final RMA expression measures for this probe set.

<table>
<thead>
<tr>
<th>GeneChip</th>
<th>Probe</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>3</td>
<td>6</td>
<td>4</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>1</td>
<td>10</td>
<td>5</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>2</td>
<td>7</td>
<td>8</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>9</td>
<td>4</td>
<td>12</td>
<td>9</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>7</td>
<td>5</td>
<td>9</td>
<td>6</td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>
An Example (continued)

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>3</td>
<td>6</td>
<td>4</td>
<td>7</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>10</td>
<td>5</td>
<td>11</td>
<td>8</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>7</td>
<td>8</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>9</td>
<td>4</td>
<td>12</td>
<td>9</td>
<td>12</td>
<td>9</td>
</tr>
<tr>
<td>7</td>
<td>5</td>
<td>9</td>
<td>6</td>
<td>10</td>
<td>7</td>
</tr>
</tbody>
</table>

0   -1    2    0    3
0   -7    2   -3    3
-1   -5    0    1    1
0   -5    3    0    3
0   -2    2   -1    3

matrix after removing row medians
An Example (continued)

\[
\begin{array}{cccccc}
0 & -1 & 2 & 0 & 3 & 0 \\
0 & -7 & 2 & -3 & 3 & 0 \\
-1 & -5 & 0 & 1 & 1 & -1 \\
0 & -5 & 3 & 0 & 3 & 0 \\
0 & -2 & 2 & -1 & 3 & 0 \\
0 & -5 & 2 & 0 & 3 & 0 \\
\end{array}
\]

\[
\begin{array}{cccccc}
0 & 4 & 0 & 0 & 0 & 0 \\
0 & -2 & 0 & -3 & 0 & 0 \\
-1 & 0 & -2 & 1 & -2 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 3 & 0 & -1 & 0 & 0 \\
0 & 3 & 0 & -1 & 0 & 0 \\
\end{array}
\]

\text{column medians}

\text{matrix after subtracting column medians}
An Example (continued)

\[
\begin{bmatrix}
0 & 4 & 0 & 0 & 0 \\
0 & -2 & 0 & -3 & 0 \\
-1 & 0 & -2 & 1 & -2 \\
0 & 0 & 1 & 0 & 0 \\
0 & 3 & 0 & -1 & 0
\end{bmatrix}
\]

row medians

\[
\begin{bmatrix}
0 & 4 & 0 & 0 & 0 \\
0 & -2 & 0 & -3 & 0 \\
0 & 1 & -1 & 2 & -1 \\
0 & 0 & 1 & 0 & 0 \\
0 & 3 & 0 & -1 & 0
\end{bmatrix}
\]

matrix after removing row medians
## An Example (continued)

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>4</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>-2</td>
<td>0</td>
<td>-3</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>-1</td>
<td>2</td>
<td>-1</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>3</td>
<td>0</td>
<td>-1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>3</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>-3</td>
<td>0</td>
<td>-3</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-1</td>
<td>2</td>
<td>-1</td>
</tr>
<tr>
<td>0</td>
<td>-1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>2</td>
<td>0</td>
<td>-1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**column medians**

**matrix after subtracting column medians**
An Example (continued)

\[
\begin{array}{ccccc}
0 & 3 & 0 & 0 & 0 \\
0 & -3 & 0 & -3 & 0 \\
0 & 0 & -1 & 2 & -1 \\
0 & -1 & 1 & 0 & 0 \\
0 & 2 & 0 & -1 & 0 \\
\end{array}
\]

All row medians and column medians are 0. Thus the median polish procedure has converged. This above is the residual matrix that we will subtract from the original matrix to obtain the fitted values.
### An Example (continued)

#### original matrix

<table>
<thead>
<tr>
<th></th>
<th>4</th>
<th>3</th>
<th>6</th>
<th>4</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>1</td>
<td>10</td>
<td>5</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>7</td>
<td>8</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>4</td>
<td>12</td>
<td>9</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>5</td>
<td>9</td>
<td>6</td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

#### residuals from median polish

-0 | 3 | 0 | 0 | 0 |
-3 | 0 | -3 | 0 | 0 |
0 | 0 | 0 | -1 | 2 | -1 |
-1 | 1 | 0 | 0 | 0 |
0 | 2 | 0 | -1 | 0 | 0 |

#### matrix of fitted values

<table>
<thead>
<tr>
<th></th>
<th>4</th>
<th>0</th>
<th>6</th>
<th>4</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>4</td>
<td>10</td>
<td>8</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>8</td>
<td>6</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>5</td>
<td>11</td>
<td>9</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>3</td>
<td>9</td>
<td>7</td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

#### row means

4.2 = \( \hat{\mu}_1 \)
8.2 = \( \hat{\mu}_2 \)
6.2 = \( \hat{\mu}_3 \)
9.2 = \( \hat{\mu}_4 \)
7.2 = \( \hat{\mu}_5 \)

RMA expression measures for the 5 GeneChips
R Commands for Obtaining RMA Expression Measures from Affymetrix .CEL Files

# load the affy package.
library(affy)

#Set the working directory to the directory containing all the .CEL files.
setwd("C:/z/Courses/Smicroarray/AffyCel")

#Read the .CEL file data.
Data<-ReadAffy()

#Compute the RMA measures of expression.
expr=rma(Data)

#Write the data to a tab-delimited text file.
write.exprs(expr, file="mydata.txt")
Robust Multiarray Average (RMA)

Pre-Normalisation
Robust Multiarray Average (RMA)

Post-Normalisation