Low-Level Analysis of High-Density Oligonucleotide Microarray Data

Ben Bolstad

http://www.stat.berkeley.edu/~bolstad

Biostatistics, University of California, Berkeley

University of Missouri
Feb 9, 2004
Outline

• What is Low-Level Analysis?
• Affymetrix GeneChip Technology
• Two topics in Low-level analysis
  – Constructing a gene expression measure
  – Probe level models for detecting differential expression
Low-Level Analysis

• What is low-level analysis?
  – Analysis and manipulation of probe intensity data
    • Expression calculation: Background, Normalization, Summarization
    • Determining presence/absence
    • Quality control diagnostics

• Why do we do it?
  – Hopefully it will allow us to produce better, more biologically meaningful gene expression values
  – We want accurate (low bias) and precise (low variance) gene expression estimates
  – Is there additional information at the probe-level that we might otherwise throw away?
High-Level Analysis

- Clustering/Classification
- Pathway Analysis
- Cell Cycle
- Gene function
- Anything where a more biological interpretation is desired

Such matters will not be discussed further in today's talk.
From Chip To Data
Brief Technology Overview

• High density oligonucleotide array technology as developed by Affymetrix
  http://www.affymetrix.com

• Known as the GeneChip

Overview images courtesy of Affymetrix unless otherwise specified
Probes and Probesets

Target Sequence

25 mer probe

probeset
Two Probe Types

PM: the Perfect Match
MM: the Mismatch differing from the Perfect Match only at the central base

PM: CAGACATAGTGTG\textcolor{red}{C}TGTTTTCTTCTCT
MM: CAGACATAGTGTG\textcolor{red}{G}TGTTTTCTTCTCT
Constructing the Chip

Focusing on a Single GeneChip Cell Location

- Actual size of GeneChip™
- Millions of DNA strands built up in each cell
- 500,000 cells on each GeneChip™ array
- Actual strand = 25 base pairs
Sample Preparation
Hybridization to the Chip
The Chip is Scanned

Shining a laser light at GeneChip causes tagged DNA fragments that hybridized to glow

Non-hybridized DNA

Hybridized DNA
Constructing a gene expression measure
Computing Expression Measures: A Three Step Procedure

1. Background/Signal adjustment (B)
2. Normalization (N)
3. Summarization (S)

Let $X$ be cel file data from multiple arrays then
Expression values = $S(N(B(X)))$
Background/Signal Adjustment

• A method which does some or all of the following
  – Corrects for background noise, processing effects
  – Adjusts for cross hybridization
  – Adjust estimated expression values to fall on proper scale

• Probe intensities are used in background adjustment to compute correction (unlike cDNA arrays where area surrounding spot might be used)
Background Methods

- **Affymetrix**
  - Location dependent
  - Ideal mismatch

- **RMA**
  - Convolution model

- **Other**
  - Standard curve adjustment
  - GCRMA (Wu et al 2003)
Normalization

“Non-biological factors can contribute to the variability of data ... In order to reliably compare data from multiple probe arrays, differences of non-biological origin must be minimized.”

• Normalization is a process of reducing unwanted variation across chips. It may use information from multiple chips

1 GeneChip 3.1 Expression Analysis Algorithm Tutorial, Affymetrix technical support
Normalization Methods

- Complete data (no reference chip, information from all arrays used)
  - Quantile normalization (Bolstad et al 2003)
  - Contrast (Åstrand)
  - Cyclic Loess
- Baseline (normalized using reference chip)
  - Scaling (Affymetrix)
  - Non linear (Li-Wong)
Summarization

- Reduce the 11-20 probe intensities for each probeset on each array to a single number for gene expression

- Main Approaches
  - Single chip
    - AvDiff (Affymetrix) – no longer recommended for use due to many flaws
    - Mas 5.0 (Affymetrix) – use a 1 step Tukey biweight to combine the probe intensities in log scale
  - Multiple Chip
    - MBEI (Li-Wong dChip) – a multiplicative model
    - RMA – a robust multi-chip linear model fit on the log scale
Parallel Behaviour Suggests Multi-chip Model
Affymetrix Spike-in Data

- 59 chips. All but 1 of the rows are done as triplicates

<table>
<thead>
<tr>
<th></th>
<th>37777</th>
<th>684</th>
<th>1597</th>
<th>38734</th>
<th>39058</th>
<th>36311</th>
<th>36889</th>
<th>1024</th>
<th>36202</th>
<th>36085</th>
<th>40322</th>
<th>407</th>
<th>1091</th>
<th>1708</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>32</td>
<td>64</td>
<td>128</td>
<td>0</td>
<td>512</td>
<td>1024</td>
</tr>
<tr>
<td>B</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>32</td>
<td>64</td>
<td>128</td>
<td>256</td>
<td>0.25</td>
<td>1024</td>
<td>0</td>
</tr>
<tr>
<td>C</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>32</td>
<td>64</td>
<td>128</td>
<td>256</td>
<td>512</td>
<td>0.5</td>
<td>0</td>
<td>0.25</td>
</tr>
<tr>
<td>D</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>32</td>
<td>64</td>
<td>128</td>
<td>256</td>
<td>512</td>
<td>1024</td>
<td>1</td>
<td>0.25</td>
<td>0.5</td>
</tr>
<tr>
<td>E</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>32</td>
<td>64</td>
<td>128</td>
<td>256</td>
<td>512</td>
<td>1024</td>
<td>0</td>
<td>2</td>
<td>0.5</td>
<td>1</td>
</tr>
<tr>
<td>F</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>32</td>
<td>64</td>
<td>128</td>
<td>256</td>
<td>512</td>
<td>1024</td>
<td>0</td>
<td>0.25</td>
<td>4</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>G</td>
<td>8</td>
<td>16</td>
<td>32</td>
<td>64</td>
<td>128</td>
<td>256</td>
<td>512</td>
<td>1024</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>8</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>H</td>
<td>16</td>
<td>32</td>
<td>64</td>
<td>128</td>
<td>256</td>
<td>512</td>
<td>1024</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>16</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td>I</td>
<td>32</td>
<td>64</td>
<td>128</td>
<td>256</td>
<td>512</td>
<td>1024</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>32</td>
<td>8</td>
<td>16</td>
</tr>
<tr>
<td>J</td>
<td>64</td>
<td>128</td>
<td>256</td>
<td>512</td>
<td>1024</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>64</td>
<td>16</td>
<td>32</td>
</tr>
<tr>
<td>K</td>
<td>128</td>
<td>256</td>
<td>512</td>
<td>1024</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>128</td>
<td>32</td>
<td>64</td>
</tr>
<tr>
<td>L</td>
<td>256</td>
<td>512</td>
<td>1024</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>256</td>
<td>64</td>
<td>128</td>
</tr>
<tr>
<td>M</td>
<td>512</td>
<td>1024</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>32</td>
<td>512</td>
<td>128</td>
<td>256</td>
</tr>
<tr>
<td>N</td>
<td>512</td>
<td>1024</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>32</td>
<td>512</td>
<td>128</td>
<td>256</td>
</tr>
<tr>
<td>O</td>
<td>512</td>
<td>1024</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>32</td>
<td>512</td>
<td>128</td>
<td>256</td>
</tr>
<tr>
<td>P</td>
<td>512</td>
<td>1024</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>32</td>
<td>512</td>
<td>128</td>
<td>256</td>
</tr>
<tr>
<td>Q</td>
<td>1024</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>32</td>
<td>64</td>
<td>1024</td>
<td>256</td>
<td>512</td>
</tr>
<tr>
<td>R</td>
<td>1024</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>32</td>
<td>64</td>
<td>1024</td>
<td>256</td>
<td>512</td>
</tr>
<tr>
<td>S</td>
<td>1024</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>32</td>
<td>64</td>
<td>1024</td>
<td>256</td>
<td>512</td>
</tr>
<tr>
<td>T</td>
<td>1024</td>
<td>0</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>32</td>
<td>64</td>
<td>1024</td>
<td>256</td>
<td>512</td>
</tr>
</tbody>
</table>
Comparing the background methods

- Using an Affymetrix spike-in experiment we shall examine
  - Observed vs spike-in concentration
  - Observed vs expected fold change
  - Composite M vs A plots
  - ROC curves
- In each case we will compute expression values use standard RMA methodology. (i.e., quantile normalization, median polish summarization)
## Assessing Bias: Observed Expression vs Spike-in Concentration

<table>
<thead>
<tr>
<th>Slope</th>
<th>None</th>
<th>RMA</th>
<th>MAS 5</th>
<th>IMM</th>
<th>MAS5/IMM</th>
<th>S.C.A.</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>0.493</td>
<td>0.63</td>
<td>0.589</td>
<td>0.69</td>
<td>0.695</td>
<td>0.856</td>
</tr>
<tr>
<td>Mid</td>
<td>0.665</td>
<td>0.784</td>
<td>0.751</td>
<td>0.82</td>
<td>0.82</td>
<td>1.041</td>
</tr>
<tr>
<td>Low</td>
<td>0.184</td>
<td>0.376</td>
<td>0.318</td>
<td>0.52</td>
<td>0.563</td>
<td>0.631</td>
</tr>
<tr>
<td>High</td>
<td>0.329</td>
<td>0.33</td>
<td>0.327</td>
<td>0.295</td>
<td>0.291</td>
<td>0.256</td>
</tr>
</tbody>
</table>
Assessing Bias: Observed Fold-change versus Expected Fold-change

<table>
<thead>
<tr>
<th>Slope</th>
<th>None</th>
<th>RMA</th>
<th>MAS 5</th>
<th>IMM</th>
<th>MAS5/IMM</th>
<th>S.C.A.</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>0.484</td>
<td>0.624</td>
<td>0.583</td>
<td>0.683</td>
<td>0.692</td>
<td>0.847</td>
</tr>
</tbody>
</table>
Assessing Variability: M vs A plots

- Vertical Axis is M a log2 fold-change.
- Horizontal Axis is A an average absolute expression value.
- Ideally non differential genes tight about M=0

![Diagram](image)
No background
RMA convolution background
Detecting Differential Expression: ROC Curves

ROC curves based upon Fold Change

- No Background
- Convolution
- Max 2.0 Background
- Ideal Mismatch
- Max 4.0 + Ideal Mismatch
- Standard Curve adjustment
## Summary of Trade-offs

<table>
<thead>
<tr>
<th>Background Method</th>
<th>Detect Differential Genes</th>
<th>Accurate estimates of Fold change</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Background</td>
<td>Good</td>
<td>Poor</td>
</tr>
<tr>
<td>RMA</td>
<td>Good</td>
<td>Poor</td>
</tr>
<tr>
<td>MAS 5.0</td>
<td>Good</td>
<td>Poor</td>
</tr>
<tr>
<td>Ideal Mismatch</td>
<td>Poor</td>
<td>Good</td>
</tr>
<tr>
<td>MAS5.0/IdealMM</td>
<td>Poor</td>
<td>Good</td>
</tr>
<tr>
<td>Standard Curve Adjustment</td>
<td>Good</td>
<td>Good</td>
</tr>
</tbody>
</table>
Comparing the Normalization Methods

• Want to reduce variation but at the same time we do not want to introduce any bias
• First a quick examination of the expression values by array
• Using same spike-in experiment as before, this time no background correction, only normalization and median polish summarization.
Scaling is Not Sufficient
Variability of Non-Differential Genes is Reduced
Little effect on Spike-ins

<table>
<thead>
<tr>
<th>Method</th>
<th>All</th>
<th>Low</th>
<th>Mid</th>
<th>High</th>
<th>FC</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Normalization</td>
<td>0.493 (0.845)</td>
<td>0.185 (0.148)</td>
<td>0.664 (0.733)</td>
<td>0.328 (0.207)</td>
<td>0.484 (0.952)</td>
</tr>
<tr>
<td>Quantile</td>
<td>0.493 (0.851)</td>
<td>0.184 (0.153)</td>
<td>0.665 (0.741)</td>
<td>0.329 (0.224)</td>
<td>0.484 (0.955)</td>
</tr>
<tr>
<td>Scaling</td>
<td>0.493 (0.852)</td>
<td>0.186 (0.156)</td>
<td>0.663 (0.742)</td>
<td>0.33 (0.225)</td>
<td>0.484 (0.954)</td>
</tr>
</tbody>
</table>
ROC Curves

ROC curve based on fold-change

True Positives

0.6 0.7 0.8 0.9 1.0

False Positives

0.00 0.02 0.04 0.06 0.08 0.10

- Red: No normalization
- Green: Quantile
- Blue: Scaling
Comparing Established Expression Measures
Probe Level Models for Detection of Differential Expression
General Probe Level Model

\[ y_{ij} = f(X) + \epsilon_{ij} \]

- Where \( f(X) \) is a linear function of factor (and possibly covariate) variables
- Assume that \( \mathbb{E}[\epsilon_{ij}] = 0 \)
  \[ \text{Var}[\epsilon_{ij}] = \sigma^2 \]
  \[ y_{ij} = \log_2 N(B(PM_{ij})) \]
We Will Focus on Two Particular PLM

- **Array effect model**
  \[ y_{ij} = \alpha_i + \beta_j + \varepsilon_{ij} \]

- **Treatment effect model**
  \[ y_{ij} = \alpha_i + \tau_{lj} + \varepsilon_{ij} \]

In both cases \( \sum_{i=1}^{I} \alpha_i = 0 \)
Fitting the PLM

- Robust regression using M-estimation
- By default, we will use Huber’s $\psi$
- Fitting algorithm is IRLS with weights $\psi(r)$
- Software for fitting such models is part of affyPLM package of Bioconductor
Suppose model is $Y = X\beta + \varepsilon$

Huber (1981) gives three forms for estimating variance covariance matrix

1. $\kappa^{-2} \left[ 1/(n-p) \sum_i \psi(r_i)^2 \right] \left( X^T X \right)^{-1}$

2. $\kappa \left[ 1/(n-p) \sum_i \psi'(r_i) \right] W^{-1}$

3. $\frac{1}{\kappa} \left( 1/(n-p) \sum_i \psi(r_i)^2 W^{-1} (X^T X) W^{-1} \right)$

We will use this form $W = X^T \Psi' X$
Fold Change

\[ FC = \bar{X}_l - \bar{X}_m \]

Where

\[ \bar{X}_l = \frac{\sum \beta_j \text{Ind}(j \in \text{group } l)}{\sum \text{Ind}(j \in \text{group } l)} \]
Simple t-statistic

\[ t = \frac{\bar{X}_l - \bar{X}_m}{\sqrt{\frac{s_l^2}{n_l} + \frac{s_m^2}{n_m}}} \]
“Robust” t-statistic

\[
  t = \frac{\hat{X}_l - \hat{X}_m}{\sqrt{\frac{\hat{S}_l^2}{n_l} + \frac{\hat{S}_m^2}{n_m}}}
\]

• Use medians in place of means
• Use MAD in place of standard deviation
Simple Moderated t-Statistic

\[ t = \frac{\bar{X}_l - \bar{X}_m}{\sqrt{\frac{s^2_l}{n_l} + \frac{s^2_m}{n_m} + S_{\text{med}}}} \]

- \( S_{\text{med}} \) is median across all genes
Limma “ebayes” t-statistic

• Generalization of Bayesian method of Lonnstadt and Speed (2002) to the general linear model case
• An alternative and much more sophisticated moderated t-statistic
Probe Level Model test statistics

- Suppose that $\sum$ is component of the variance-covariance matrix related to $\beta$

- Let $c$ be the contrast vector defined such that the $j$th element of $c$ is $1/n_l$ if array $j$ is in group $l$ and $-1/n_m$ if array $j$ is in group $m$, 0 otherwise
Probe Level Model test statistics

\[
t_{\text{PLM.1}} = \frac{c^T \beta}{\sqrt{\sum_{j=1}^{J} c_j^2 \Sigma_{jj}}} \quad t_{\text{PLM.2}} = \frac{c^T \beta}{\sqrt{c^T \Sigma c}}
\]
A First Comparison

- 8 chips from Affymetrix HG-U95A spike-in dataset
  - 4 arrays for each of two concentration profiles
- Fit an array effect model to all 8 chips
  - Compare the performance of the different methods by looking at all comparisons
    - 1 vs 1
    - 2 vs 2
    - 3 vs 3
    - 4 vs 4
Affy Spikein: 3 on 3

True Positives vs False Positives

Legend:
- FC
- T.std
- T.PLM.1
- T.PLM.2
- T.robust
- T.mcd
- T.ebayes
What Happens as the Number of Arrays Increases?

- Expand comparison to all 24 Arrays with same concentration profiles from Affymetrix HG-U95A spike-in dataset
- Fit an array effect model to all 24 arrays
- Look at comparisons between equal number of chips
True positives when FP = 1%

Number of arrays in each group

True Positives

- FC
- T.std
- T.PLM.1
- T.PLM.2
- T.robust
- T.mod
- T.ebayes
A Larger Comparison

• Look at the entire 59 chips for Affymetrix HG-U95A spike-in dataset
• Examine two cases. After standard preprocessing
  – Fit a model to all 59 chips
  – Fit models for each pairwise comparison
• There are 91 pairwise comparisons
## Results

<table>
<thead>
<tr>
<th>Method</th>
<th>Individual Models</th>
<th></th>
<th></th>
<th></th>
<th>Single Model</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0% FP</td>
<td>5% FP</td>
<td>AUC</td>
<td>0% FP</td>
<td>5% FP</td>
<td>AUC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FC</td>
<td>0.451</td>
<td>0.985</td>
<td>0.975</td>
<td>0.444</td>
<td>0.982</td>
<td>0.971</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Std</td>
<td>0.323</td>
<td>0.982</td>
<td>0.956</td>
<td>0.301</td>
<td>0.975</td>
<td>0.952</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Robust</td>
<td>0.16</td>
<td>0.939</td>
<td>0.857</td>
<td>0.144</td>
<td>0.935</td>
<td>0.852</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mod</td>
<td>0.437</td>
<td>0.987</td>
<td>0.975</td>
<td>0.413</td>
<td>0.98</td>
<td>0.97</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLM.1</td>
<td>0.653</td>
<td>0.991</td>
<td>0.979</td>
<td>0.54</td>
<td>0.951</td>
<td>0.93</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLM.2</td>
<td>0.657</td>
<td>0.991</td>
<td>0.979</td>
<td>0.539</td>
<td>0.951</td>
<td>0.93</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ebayes</td>
<td>0.514</td>
<td>0.988</td>
<td>0.978</td>
<td>0.45</td>
<td>0.986</td>
<td>0.974</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
More Spike-in Datasets

• Two GeneLogic Spike-in datasets
  – AML dataset (34 arrays)
  – Tonsil dataset (36 arrays)
• In each case use single models fitted to all arrays
What is going on here?

• Examine residuals stratified by concentration group
  – Spike-ins
  – Randomly chosen non-differential probesets at low, medium and high average expression
Affymetrix Spike-ins
Low Non-Differential
Middle Non-Differential

39638_at

31406_at

39199_at

41233_at

1248_at

38988_at

36264_at

37417_at

40110_at

630_at

34147_g_at

38592_s_at

38950_r_at

1650_g_at

40404_s_at

38609_at
High Non-Differential

40887_g_at
39088_at
39473_r_at
31538_at
256_s_at
32438_at
35905_s_at
33660_at
34085_at
33677_at
31623_f_at
37746_r_at
38061_at
36546_r_at
41210_at
882_at
GeneLogic AML Spike-ins
GeneLogic Tonsil dataset
How About With More “Real” Data?

• Previous comparisons were for Spike-in data where only 11 or 14 probesets were expected to show any change between conditions. Consider GeneLogic Dilution/Mixture study. Using the 30 Liver and 30 CNS arrays to give a “truth”

• Use the 75:25 (5 arrays) and 25:75 (5 arrays) mixture arrays to test

• Choose 400 probesets with most extreme t-statistics from Dilution set to define “truth”
## Results

<table>
<thead>
<tr>
<th>Method</th>
<th>3 vs 3</th>
<th></th>
<th>4 vs 4</th>
<th></th>
<th>5 vs 5</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0% FP</td>
<td>5% FP</td>
<td>AUC</td>
<td>0% FP</td>
<td>5% FP</td>
<td>AUC</td>
</tr>
<tr>
<td>FC</td>
<td>0.007</td>
<td>0.886</td>
<td>0.697</td>
<td>0.008</td>
<td>0.888</td>
<td>0.703</td>
</tr>
<tr>
<td>Std</td>
<td>0.004</td>
<td>0.793</td>
<td>0.53</td>
<td>0.008</td>
<td>0.872</td>
<td>0.626</td>
</tr>
<tr>
<td>Robust</td>
<td>0.002</td>
<td>0.485</td>
<td>0.271</td>
<td>0.005</td>
<td>0.747</td>
<td>0.49</td>
</tr>
<tr>
<td>Mod</td>
<td>0.007</td>
<td>0.908</td>
<td>0.697</td>
<td>0.002</td>
<td>0.932</td>
<td>0.735</td>
</tr>
<tr>
<td>PLM.1</td>
<td>0.056</td>
<td>0.943</td>
<td>0.751</td>
<td>0.057</td>
<td>0.947</td>
<td>0.756</td>
</tr>
<tr>
<td>PLM.2</td>
<td>0.057</td>
<td>0.943</td>
<td>0.752</td>
<td>0.057</td>
<td>0.948</td>
<td>0.758</td>
</tr>
<tr>
<td>Ebayes</td>
<td>0.001</td>
<td>0.918</td>
<td>0.744</td>
<td>0</td>
<td>0.933</td>
<td>0.761</td>
</tr>
</tbody>
</table>
What about the treatment effect model?

• The limma ebayes test statistic seems to be outperforming the PLM test statistics in the AUC statistic. Closer examination of ROC curve shows it exceeding all other methods between 0.25% and 2.5% false positives.

• Try the Treatment effect model with
  – PLM.2
  – PLM.2 with a simple moderation
Mixture data: 5 on 5
Ongoing work in this area

- Technology changes: what still works?  
  What doesn’t?
- Better moderation for the PLM test statistic
- Other probe-level models
Acknowledgements

- Terry Speed (UC Berkeley)
- Francois Colin (UC Berkeley)
- Rafael Irizarry (Johns Hopkins)

- Bioconductor Core
  http://www.bioconductor.org
Additional Slides
Background Signal Methods

• Affymetrix
  – Location dependent background based on grids
    • I will refer to this as the MAS 5 background
  – Originally proposed subtracting MM from PM but this is problematic because as many as a third of MM’s are greater than the respective PM
    • No longer used
  – Now uses what they refer to as the Ideal Mismatch which is MM when possible and something else when not possible (designed so that there is now no negatives)
    • Call this IMM
Original RMA Background

- Convolution model is suggested by looking at density of observed empirical distributions
**Convolution Model**

- **O = S + N**
  - O is observed PM, S is signal (assumed exponential), N is noise (assumed normal, truncated at zero)

- **Correction is then**

\[
E(S \mid O = o) = a + b \frac{\phi\left(\frac{a}{b}\right) - \phi\left(\frac{o - a}{b}\right)}{\Phi\left(\frac{a}{b}\right) - \Phi\left(\frac{o - a}{b}\right)} - 1
\]

\[
a = o - \mu - \sigma^2 \alpha, \quad b = \sigma
\]
A Standard Curve Adjustment Based on Spike-in Information

- Observes that there is a curve that relates observed expression and spike-in concentration. The ideal would be to have a linear relationship between concentration and computed expression. The curve gives us a concentration dependent adjustment.
What About Non Spike-ins?

- We don’t know a concentration for most probesets. If we did, or if we had a variable that related to concentration, the adjustment would be easy to perform.

- Fit the following model:

  \[ y^{(k)}_{1i} = \alpha^{(k)}_i + \varepsilon^{(k)}_i \]

  \[ y^{(k)}_{2i} = \alpha^{(k)}_i + \gamma^{(k)}_i + \varepsilon^{(k)}_i \]

- Where

  \[ y^{(k)}_{1i} = \log_2 PM^{(k)}_i \]

  \[ y^{(k)}_{2i} = \log_2 MM^{(k)}_i \]
Relates to Concentration

Expression levels vs Gamma

RNA expression vs background
Establishing a Relationship Between $\gamma$ and Concentration
The Two Curves Yield an Adjustment Curve
Quantile Normalization

• Normalize so that the quantiles of each chip are equal. Simple and fast algorithm. Goal is to give same distribution to each chip.

• We will illustrate the algorithm with an example.
Sort columns of original matrix

Take averages across rows

Set average as value for All elements in the row

Unsort columns of matrix to original order

\[
\begin{bmatrix}
1 & 5 & 3 & 5 \\
2 & 1 & 6 & 7 \\
3 & 2 & 2 & 6 \\
4 & 6 & 1 & 8 \\
\end{bmatrix}
\rightarrow
\begin{bmatrix}
1 & 1 & 1 & 5 \\
2 & 2 & 2 & 6 \\
3 & 5 & 3 & 7 \\
4 & 6 & 6 & 8 \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
1 & 1 & 1 & 5 \\
2 & 2 & 2 & 6 \\
3 & 5 & 3 & 7 \\
4 & 6 & 6 & 8 \\
\end{bmatrix}
\rightarrow
\begin{bmatrix}
2 \\
3 \\
4.5 \\
6 \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
2 \\
3 \\
4.5 \\
6 \\
\end{bmatrix}
\rightarrow
\begin{bmatrix}
2 & 2 & 2 & 2 \\
3 & 3 & 3 & 3 \\
4.5 & 4.5 & 4.5 & 4.5 \\
6 & 6 & 6 & 6 \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
2 & 2 & 2 & 2 \\
3 & 3 & 3 & 3 \\
4.5 & 4.5 & 4.5 & 4.5 \\
6 & 6 & 6 & 6 \\
\end{bmatrix}
\rightarrow
\begin{bmatrix}
2 & 4.5 & 4.5 & 2 \\
3 & 2 & 6 & 4.5 \\
4.5 & 3 & 3 & 3 \\
6 & 6 & 2 & 6 \\
\end{bmatrix}
\]
Why Quantile Normalization?

- Quantile normalization found to perform acceptably in reducing variance without drastic bias effects
- Quantile normalization is fast
RMA Model

• To each probeset \((k)\), with \(i\) being number of probes and \(j\) being number of chips, fit the model:

\[
y_{ij}^{(k)} = \alpha_i^{(k)} + \beta_j^{(k)} + \epsilon_{ij}^{(k)}
\]

where \(\alpha_i^{(k)}\) is a probe effect and \(\beta_j^{(k)}\) is the log gene expression. \(y_{ij}^{(k)}\) is the log2 background adjusted and normalized PM intensity

• Different ways to fit this model
  – Median polish – quick
  – Robust linear model – yields some good quality diagnostic tools
Probe Level Models are based on RMA

- RMA method
  - Convolution Model Background
  - Quantile Normalization
  - Summarization using a robust multi-chip model on the log scale. Model is fitted using the median polish algorithm on a probeset by probeset basis
Basic RMA model

Let

\[ y_{ij} = \log_2 N(B(\text{PM}_{ij})) \]

then

\[ y_{ij} = m + \alpha_i + \beta_j + \varepsilon_{ij} \]

where \( \alpha_i \) is probe-effect

\( \beta_j \) is chip-effect (\( m + \beta_j \) is log2 gene expression on array \( j \))

Median-polish imposes constraints

\( \text{median} \alpha_i = \text{median} \beta_j = 0 \)

\( \text{median}_i \varepsilon_{ij} = \text{median}_j \varepsilon_{ij} = 0 \)
Advantages/Disadvantages of RMA/Median polish

• Advantages
  – Fast
  – Gene expression measures perform favorably when compared with MAS 5.0, Li-Wong MBEI
  – Robust against outliers

• Disadvantages
  – No standard error estimates
  – No algorithmic flexibility to fit alternative models
<table>
<thead>
<tr>
<th>Slope</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>0.63</td>
</tr>
<tr>
<td>Mid</td>
<td>0.784</td>
</tr>
<tr>
<td>Low</td>
<td>0.376</td>
</tr>
<tr>
<td>High</td>
<td>0.33</td>
</tr>
</tbody>
</table>

Convolution background

Slope Low End: 0.376
R-Sq Low End: 0.203
Slope High End: 0.33
R-Sq High End: 0.201

Slope Middle Range: 0.784
R-Sq Middle Range: 0.73
Slope Overall: 0.63
R-Sq Overall: 0.65

Log2(Conc)
<table>
<thead>
<tr>
<th>Slope</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>0.589</td>
</tr>
<tr>
<td>Mid</td>
<td>0.751</td>
</tr>
<tr>
<td>Low</td>
<td>0.318</td>
</tr>
<tr>
<td>High</td>
<td>0.327</td>
</tr>
<tr>
<td>Slope</td>
<td>Value</td>
</tr>
<tr>
<td>---------</td>
<td>---------</td>
</tr>
<tr>
<td>All</td>
<td>0.69</td>
</tr>
<tr>
<td>Mid</td>
<td>0.82</td>
</tr>
<tr>
<td>Low</td>
<td>0.52</td>
</tr>
<tr>
<td>High</td>
<td>0.295</td>
</tr>
<tr>
<td>Slope</td>
<td>Value</td>
</tr>
<tr>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>All</td>
<td>0.856</td>
</tr>
<tr>
<td>Mid</td>
<td>1.041</td>
</tr>
<tr>
<td>Low</td>
<td>0.631</td>
</tr>
<tr>
<td>High</td>
<td>0.256</td>
</tr>
</tbody>
</table>
Slope: 0.484
Slope: 0.624
Slope: 0.583
Slope: 0.683
Slope: 0.692
Slope: 0.847