RPA: analysis of probe reliability and differential gene expression on short oligonucleotide arrays.

Leo Lahti
Helsinki Institute for Information Technology HIIT
and Adaptive Informatics Research Centre,
Department of Information and Computer Science,
Aalto University School of Science and Technology,
P.O. Box 15400, FI-00076 Aalto, Finland
leo.lahti@iki.fi

October 18, 2010

1 Introduction

RPA (Robust Probabilistic Averaging) package[1] provides tools for analyzing probe reliability and differential gene expression on Affymetrix short oligonucleotide arrays.

RPA provides estimates of probe reliability, and a probeset-level estimate of differential gene expression between a user-specified reference array and the other arrays. Probabilistic formulation allows incorporation of prior information concerning probe reliability into gene expression analysis within a principled framework. The underlying probabilistic model for probe-level observations is described in Lahti et al..

Your comments and contributions are welcome.

1.1 Relation to other probe-level models

Probe-level calculation of differential expression helps to avoid the modeling of unidentifiable probe affinities, which are the key probe-specific parameter in many preprocessing methods. In contrast, RPA estimates the overall reliability of each probe in terms of a probe-specific variance. This distinguishes RPA from probe-level preprocessing methods such as dChip's MBEI Li and Wong (2001), RMA Irizarry et al. (2003a), or FARMS Hochreiter et al. (2006), that provide probeset-level summaries but have not been used to investigate probe performance. However, in addition to probe reliability analysis RPA

[1] http://www.cis.hut.fi/projects/mi/software/RPA/

can be used for preprocessing in differential gene expression studies, where it has been shown to outperform other popular preprocessing methods. For details, see Lahti et al..

1.2 Summary of RPA model

RPA assumes a Gaussian model for probe effects. Consider a probe set targeted at measuring the expression level of target transcript g. Probe-level observation s_{ij} of probe j on array i is modeled as a sum of the true expression signal (common for all probes in the probeset), and probe-specific Gaussian noise: $s_{ij} = g_i + \mu_j + \varepsilon_{ij}$. Importantly, the stochastic noise component is probe-specific in RPA, and distributed as $\varepsilon_{ij} \sim N(0, \tau_j^2)$. The variance parameters $\{\tau_j^2\}$ are of interest in probe reliability analysis; the inverse variance $1/\tau_j^2$ can be used to measure of probe reliability (see get.probe.noise.estimates function).

The mean parameter μ_j of the noise model describes systematic probe affinity effect but it is unidentifiable. In RPA, these parameters cancel out when the signal log-ratio between a user-specified 'reference' array and the remaining arrays is computed for each probe: the differential expression signal between arrays $t = \{1, \ldots, T\}$ and the reference array c for probe j is given by $m_{tj} = s_{tj} - s_{cj} = g_t - g_c + \varepsilon_{tj} - \varepsilon_{cj} = d_t + \varepsilon_{tj} - \varepsilon_{cj}$. In vector notation the differential gene expression profile of probe j across the arrays can be written as $\mathbf{m}_j = \mathbf{d} + \boldsymbol{\varepsilon}_j$. In practice, $\mathbf{d}$ and the probe-specific variances $\{\tau_j\}_{j=1}^P$ for the P probes within the probeset are estimated simultaneously based on the probabilistic model. With large sample sizes the solution will converge to estimating the mean of the probe-level observations weighted by probe reliability.

The probe-level data is background corrected, normalized, and log2-transformed before the analysis. By default, RPA uses the background correction model of RMA Irizarry et al. (2003b) and quantile normalization Bolstad et al. (2003). Our implementation utilizes the *affy* package Gautier et al. (2004) to handle probe-level data. For details about short oligonucleotide arrays and the design of the Affymetrix GeneChip arrays, see the Affymetrix MAS manual Affymetrix (2001).

2 Probe reliability analysis

RPA operates on affybatch objects. An affybatch can be created from Affymetrix CEL files using the ReadAffy function of the BioConductor *affy* package Gautier et al. (2004). An affybatch contains the probe-level data of Affymetrix arrays. Our toy examples use the `Dilution` dataset provided by *affydata* package. Load example data (the 'Dilution' affybatch):

```
> require(affy)
> require(affydata)
> data(Dilution)
```

RPA.pointestimate is the main function. Let us perform the analysis for particular probesets in the Dilution data using the first array ($cind = 1$) as the reference for calculating differential expression values of the other arrays.

```
> require(RPA)
> sets <- geneNames(Dilution)[1:2]
> rpa.results <- RPA.pointestimate(Dilution, sets, cind = 1)
```

Probe reliability and differential gene expression analysis is performed on the whole data set as follows (potentially slow).

```
> rpa.results <- RPA.pointestimate(Dilution, cind = 1)
```

The 'rpa2eset' function coerces the probeset-level differential expression estimates (d) into an ExpressionSet object that can be analyzed using standard R/BioC tools for gene expression. The results for a particular probeset are visualized with

```
> rpa.plot("1000_at", rpa.results)
```

The output is shown in Figure 1. See help('rpa.plot') for details.

2.1 Estimating probe-specific noise and probe reliability

RPA estimates the noise level of each individual probe through the probe-specific variance parameter (τ_j^2). These can be obtained with

```
> noise <- get.probe.noise.estimates(rpa.results)
```

The higher the variance, the more noisy the probe. Inverse of the variance, $\frac{1}{\tau_j^2}$, can be used to quantitate probe reliability. Note that the relative weight of a probe within probeset is determined by the relative noise of the probe with respect to the other probes in the same probeset. Comparison of probe-specific variances across probesets may benefit from normalization of this effect. The get.probe.noise.estimates function can optionally provide normalized versions of the noise estimates.

2.2 Manual analysis of an individual probe set

Preprocess data before the analysis:

```
> Smat <- RPA.preprocess(Dilution, cind = 1)
```

Pick probe-level data for a probe set (arrays x probes matrix):

```
> S <- t(Smat$fcmat[Smat$set.inds[["1000_at"]], ])
```

Estimate probeset-level signal and probe-specific variances:

```
> res <- RPA.iteration(S)
```

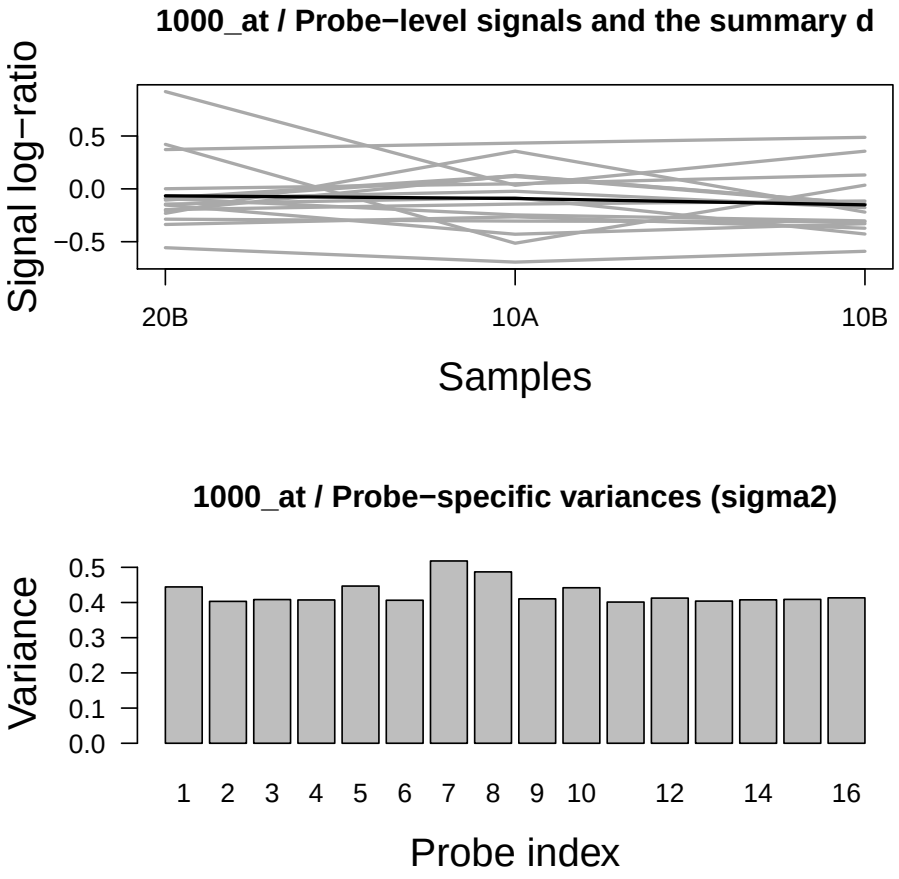

Figure 1: Estimated probe-specific variances and differential gene expression for an example probe set.

2.3 Setting probe-specific priors

Prior information of probe reliability can be set by tuning the shape (alpha) and scale (beta) parameters of the model. These are inverse Gamma distribution parameters, which is the conjugate prior for the variances. Set priors for a particular probeset. If the 'priors' parameter is not given, non-informative priors will be given for the other probesets:

```
> alpha <- beta <- rep(1, 16)
> probe.index <- 5
> alpha[[probe.index]] <- 3
> beta[[probe.index]] <- 1
> priors <- set.priors(Dilution, set = "1000_at", alpha, beta)
```

Run RPA with priors:

```
> rpa.results <- RPA.pointestimate(Dilution, sets, priors = priors)
```

3 Differential gene expression analysis

RPA provides a wrapper ('rpa') for robust preprocessing of gene expression data in differential gene expression studies. One of the arrays is used as the reference against which the differential expression values are calculated; RPA expression values are $\log_2$-ratios related to the reference array. Choice of the control array does not affect probe reliability estimates. RPA supports also alternative CDF environments (see function documentation for details).

```
> eset <- rpa(Dilution, cind = 1)
```

The output is an ExpressionSet object, which allows downstream analysis of the results using standard R/BioC tools for gene expression data.

4 Citing RPA

Please cite Lahti et al. when using the package.

5 Details

This document was written using:

```
> sessionInfo()
```

```
R version 2.12.0 RC (2010-10-11 r53293)
Platform: i386-pc-mingw32/i386 (32-bit)

locale:
[1] LC_COLLATE=C
[2] LC_CTYPE=English_United States.1252
[3] LC_MONETARY=English_United States.1252
[4] LC_NUMERIC=C
[5] LC_TIME=English_United States.1252

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
[1] hgu95av2cdf_2.7.0 RPA_1.6.0         affydata_1.11.10  affy_1.28.0
[5] Biobase_2.10.0

loaded via a namespace (and not attached):
[1] affyio_1.18.0         preprocessCore_1.12.0 tools_2.12.0
```

References

Affymetrix. *Affymetrix Microarray Suite User Guide*. Affymetrix, Santa Clara, CA, version 5 edition, 2001.

Benjamin M. Bolstad, Rafael A. Irizarry, M. Astrand, and Terence P. Speed. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, 19(2):185–193, 2003.

Laurent Gautier, Leslie Cope, Benjamin M. Bolstad, and Rafael A. Irizarry. affy–analysis of Affymetrix GeneChip data at the probe level. *Bioinformatics*, 20(3):307–315, 2004.

Sepp Hochreiter, Djork-Arne Clevert, and Klaus Obermayer. A new summarization method for affymetrix probe level data. *Bioinformatics*, 22(8):943–949, 2006.

Rafael A. Irizarry, Benjamin M. Bolstad, Francois Collin, Leslie M. Cope, Bridget Hobbs, and Terence P. Speed. Summaries of Affymetrix GeneChip probe level data. *Nucl. Acids Res.*, 31(4):e15, 2003a.

Rafael A. Irizarry, Bridget Hobbs, Francois Collin, Yasmin D. Beazer-Barclay, Kristen J. Antonellis, Uwe Scherf, and Terence P. Speed. Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics*, 4(2): 249–264, 2003b.

Leo Lahti, Laura L. Elo, Tero Aittokallio, and Samuel Kaski. Probabilistic analysis of probe reliability in differential gene expression studies with short oligonucleotide arrays. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*. URL `http://doi.ieeecomputersociety.org/10.1109/TCBB.2009.38`.

Cheng Li and Wing Hung Wong. Model-based analysis of oligonucleotide arrays: Expression index computation and outlier detection. *Proc. Natl. Acad. Sci.*, 98:31–36, 2001.