beadarray, BASH and HULK - tools to increase the value of Illumina BeadArray experiments.

Andy G. Lynch*, Mike L. Smith, Mark J. Dunning, Jonathan M. Cairns, Nuno L. Barbosa-Morais and Simon Tavaré

Cambridge Research Institute, University of Cambridge/Cancer Research UK
http://www.compbio.group.cam.ac.uk

1 Introduction

The Illumina BeadArray is an increasingly popular microarray technology with applications to RNA expression, genotyping, DNA copy number and methylation. Here we focus on the RNA expression arrays, but many of the principles carry over to other BeadArray technologies.

The defining characteristics of the BeadArray are its random construction (Gunderson *et al.* (2004)), and the coexistence of multiple arrays on a single chip. The random construction extends to both the contents and the layout of the array, so that while a typical probe will have about 20 replicates scattered across the array surface, the number of replicates for a probe can range from 0 (entirely absent) to over 40. Thus no two arrays have the same layout.

The most recent RNA expression BeadArray contains approximately 1,000,000 beads representing 50,000 bead-types (i.e., beads carrying the same probe), and 12 BeadArrays share each BeadChip. The 'raw' data from such a chip will consist of the 12 scanned images, and 12 accompanying text files giving for each bead the identity of the probe attached, the location of the bead, and a default intensity value calculated for the bead.

However, the raw data are not usually available. Instead, Illumina's BeadStudio system summarizes the data so that means, standard errors and numbers of beads are reported for the 50,000 bead-types, rather than data for the 1,000,000 beads (the so-called "bead-level" data). Raw data can, with effort, be obtained, but while there are benefits to starting from the raw data, there is little encouragement to do so from journals, data repositories, or Illumina themselves.

To address this matter, we have developed *beadarray* (Dunning *et al.* (2007)), a BioConductor package that provides a framework for the analysis of bead-level Illumina data and a set of tools that enhance the information that can be extracted from such data.

2 The Analysis

We present the benefits offered by our tools for the component stems of a typical analysis.

2.1 Converting pixel intensities to bead intensities

Default intensity measures for each bead are available in the raw data text files. These follow an image-sharpening procedure, calculation of foreground and background intensities for the bead, and a basic background subtraction step. For those wishing to optimize the process, *beadarray* offers flexibility in terms of the background intensity calculation, and makes the sharpening step optional. Foreground calculation options will be implemented in the near future, and through BioConductor there are a large number of alternatives to the standard background subtraction step. However, it is the case that Illumina's default values perform well enough that the extra storage, memory, and processing requirements necessary for re-extraction of the values do not often seem to be prices worth paying.

2.2 Transformation of bead-intensities

A distinct drawback of the default Illumina process is that the values are retained on the scale in which they are extracted rather than being transformed for the steps that follow. It is more common to analyse such data after a log-transformation or some other variance stabilizing transformation (Lin *et al.* (2008)), although note that the Illumina-specific VST has been proposed only for summarized data. *beadarray* allows for log-transformation at the bead level, and other transformations can be implemented if demand is encountered.

2.3 Masking of spatial artefacts

At no point in their preprocessing do Illumina make use of the spatial information included in the raw data. While they do include an outlier removal step at a later stage, this has proven inadequate to address the problems of spatial artefacts (Figure 1.). Harshlight (Suárez-Fariñas *et al.* (2005)) is an algorithm for the identification of such spatial artefacts on Affymetrix chips, but makes use of multiple chips and the consistent chip layout.

We have previously presented BASH (BeadArray Subversion of Harshlight) (Cairns *et al.* (2008)), which takes the concepts of Harshlight but implements them in an iterative manner within a single BeadArray. Artefacts are identified from unusually dense clusters of outlying beads, where outliers are defined amongst the replicate beads of the same type. The gaps in clusters are then filled in to remove artefacts more completely. This relies on having an efficient algorithm for identifying and manipulating the network of neighbouring beads. This we have implemented in BASH and use also in the next step.

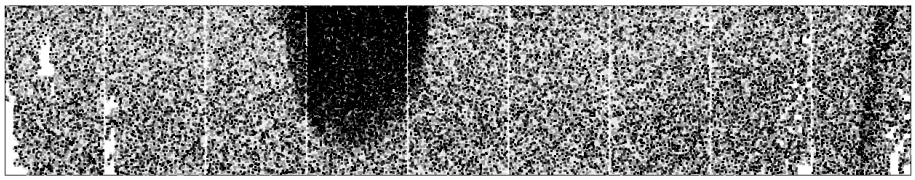

Figure 1: Spatial artefacts on an Illumina array. Beads with standardized residual values of greater than 2 are depicted, with darker shades representing larger absolute values. Removing outliers in the standard manner will only remove a fraction of the beads within the affected areas. Distinct white areas indicate the absence of beads rather than the absence of outliers.

2.4 Filters for low intensity spatial trends

Within *beadarray*, HULK (BeadArray Normalization by NEighbourhood Residuals) uses the network functions offered by BASH to adjust log-intensities by the weighted average of residual values within a local neighbourhood. This adjustment addresses gradients (and similar phenomena) across chips, which are beyond the scope of BASH. While BASH removes beads from the analysis, HULK adjusts the values of the beads left after BASHing, and combination of the two tools requires care. The total log-intensity is not preserved by HULK, which may be undesirable, but not largely detrimental compared to the removal of large gradient effects.

2.5 Summarization

Ultimately, we wish to summarize the bead-level data so that we have intensities (and other properties) for each bead-type. There are conflicting reports of Illumina's summarization procedure, but it is clear that there is an outlier removal step (outliers being defined as more than three median absolute deviations from the median within the bead-type), and possibly, when there are more than 30 replicates, a trimming step (with the top and bottom percentage of beads removed). Recall that Illumina do this on the original, non-logged, values. The positively-skewed raw intensities, combined with a symmetric rule for calling outliers, often makes it impossible to identify unusually low values by this method.

beadarray offers a choice of scale for summarization, with optional trimming or outlier removal steps. More flexibility of outlier definition is possible with *beadarray* than with Illumina's software. Variance calculations, though, do not acknowledge this distributional censoring, which may be undesirable. This is a concern both for Illumina's and our tools.

2.6 Filtering Bead-Types

One of the most recognised problems of microarray analysis is that of multiple testing; the number of probe-centric models usually exceeds the number of data points in each model by several orders of magnitude. There are two obvious approaches to reducing the number of models that need to be fitted: either reduce the number of probe-centric models by filtering out unlikely bead-types, or analyse the data at a higher level (e.g. genes, gene networks).

'Uninteresting' bead-types are often filtered out of an analysis using the observed data. There is a concern that this must bias, or at least complicate, downstream analyses. An alternative is to filter probes ahead of the experiment, based on a careful reannotation of the probe sequences. Such an annotation is also required to perform a higher-level analysis, and is therefore crucial for acceptably reducing the number of tests being performed. Thus we have reannotated the more popular Illumina platforms (www.compbio.group.cam.ac.uk/Resources/Annotation/).

2.7 Filtering/Down-weighting Arrays

As well as filtering bead-types out of the analysis, we may need to filter out (or down-weight) some arrays. We can judge individual arrays on the basis of quality control plots. *beadarray* knows the identities of the control probes for the popular Illumina platforms, and can construct a range of such plots at the array, chip or experiment level. We have identified failings in a number of Illumina's control probes and can exclude these in any analysis, improving our ability to detect flawed arrays. While Illumina's BeadStudio software produces similar plots, they suffer from Illumina's lack of a log-transformation. Additionally, specific Illumina plots are compromised (e.g. where the pairing of perfect and mismatch probes is lost in the presentation).

2.8 Normalization

Other quality control procedures are available through BioConductor, and similarly a large number of normalization steps can also be accessed, as well as the ones in *beadarray*. Normalization is the process of removing technical artefacts such as chip-to-chip variance, so that the values obtained from one array are comparable to those from another. There are, however, drawbacks. A common theme in an Illumina analysis is that the random construction leads to arrays that provide differing levels of evidence Yet, while great consideration has been given to the issue of normalizing the mean intensities, the propagation of the second moment estimates through the normalization procedure has been neglected. There are obvious first steps that one might try, and these will be incorporated into a future version of *beadarray*.

2.9 Statistical modelling

A similar problem occurs when feeding BeadArray data through the popular analysis package *limma*. This is easy for *beadarray* as both are housed within BioConductor. A typical *limma* analysis involves a variance moderation step "eBayes", wherein the often unstable variance estimates (from each of the probe models) are strengthened by sharing information across models. This is sensible, but for Illumina there is a good deal of known structure for those variances that is ignored in this step. Improvements can thus be made.

We have shown (Dunning *et al.* (2007)) that weighting the linear models by the inverse squares of the standard errors, calculated from the replicate beads, is beneficial. However for two-colour Illumina BeadArrays (such as the GoldenGate and Infinium arrays - and here we depart from our expression focus) it is not possible to calculate standard errors adequately for functions of the two-channels (such as log-ratios) unless using the bead-level data (Lynch *et al.* (2009)). This is because the summarized Illumina data do not include the covariances of the two channels, and these are necessary for a decent approximation to be made.

We acknowledge the support of The University of Cambridge, Cancer Research UK and Hutchison Whampoa Limited.

References

Cairns J.M., Dunning M.J., Ritchie M.E., Russell R. and Lynch A.G. (2008). BASH: a tool for managing BeadArray spatial artefacts *Bioinformatics*, **24**, 2921-2922

Dunning M.J., Smith M.L., Ritchie M.E. and Tavaré S. (2007). beadarray: R classes and methods for Illumina bead-based data. *Bioinformatics*, **23**, 2183-2184

Dunning M.J., Barbosa-Morais N.L., Lynch A.G., Tavaré S. and Ritchie M.E. (2008). Statistical issues in the analysis of Illumina data. *BMC Bioinformatics*, **9**, 85

Gunderson K.L., Kruglyak S., Graige M.S., Garcia F., Kermani B.G., Zhao C., Che D., Dickinson T., Wickham E., Bierle J., Doucet D., Milewski M., Yang R., Siegmund C., Hass J., Zhou L., Oliphant A., Fan J.B., Barnard S., and Chee M.S. (2004). Decoding randomly ordered DNA arrays. *Genome Research*, **14**, 870-877

Lin S.M., Du P., Huber W. and Kibbe W.A. (2008). Model-based variance-stabilizing transformation for Illumina microarray data. *NAR*, **36**, e11

Lynch A.G., Dunning M.J, Iddawela M., Barbosa-Morais N.L. and Ritchie M.E. (2009). Considerations for the processing and analysis of GoldenGate-based two-colour Illumina platforms *SMMR*, online ahead of print

Suárez-Fariñas M., Haider A. and Wittkowski K.M. (2005). "Harshlighting" small blemishes on microarrays. *BMC Bioinformatics*, **6**, 65