INSTITUTO NACIONAL DE ESTATÍSTICA
PORTUGAL

REVSTAT
Statistical Journal

Special Issue on
"Statistical Applications in Bioinformatics"

Guest Editors:
Simon Tavaré
M. Antónia Amaral-Turkman

Volume 4, No.1
March 2006

Catalogação Recomendada

REVSTAT. Lisboa, 2003-
Revstat : statistical journal / ed. Instituto Nacional de Estatística. - Vol. 1, 2003- . - Lisboa I.N.E., 2003- . - 30 cm
Semestral. - Continuação de : Revista de Estatística =
ISSN 0873-4275. - edição exclusivamente em inglês
ISSN 1645-6726

CREDITS

- **PUBLISHER**
  - *Instituto Nacional de Estatística (INE)*
    Av. António José de Almeida, 2
    1000-043 LISBOA
    PORTUGAL
    Tel.: (0351) 21 842 61 00
    Fax: (0351) 21 842 63 64
    Web site: http://www.ine.pt
- **COVER DESIGN**
  - *Mário Bouçadas, designed on the stain glass window at INE by the painter Abel Manta*
- **LAYOUT AND GRAPHIC DESIGN**
  - *Carlos Perpétuo*
- **PRINTING**
  - *Instituto Nacional de Estatística*
- **EDITION**
  - *450 copies*
- **LEGAL DEPOSIT REGISTRATION**
  - *N.º 191915/03*

PRICE

[VAT 5% included]

- Single issue ..... **€ 12**

FOREWORD

This special issue of REVSTAT *Statistical Review* contains a selection of papers (invited and contributed) presented at the *Workshop on Statistics in Genomics and Proteomics* that took place in Monte Estoril, Portugal, from 5–8 October 2005.

The workshop, organized under the auspices of the *International Mathematical Center* (CIM, `http://www.cim.pt`) and the *Center of Statistics and its Applications* (CEAUL, `http://www.ceaul.fc.ul.pt`), brought together leading researchers in the areas of statistics in genomics and proteomics, who described the state of the art and presented several challenging problems for researchers in Biostatistics and Bioinformatics. Thanks are due to Wolfgang Urfer and Terry Speed who helped to organize the invited programme. We also express our gratitude to all the speakers for their contribution to the high scientific standards of the *Workshop on Statistics in Genomics and Proteomics*.

Due to the lack of space, it was not possible to condense all the outstanding contributions to the workshop into this single special issue. However, a further selection of invited and contributed papers will be published by CIM in their edition of Proceedings of Conferences.

The four papers in this volume illustrate some of the statistical problems currently of interest in bioinformatics. Dunning et al. describe methods for the low-level analysis of expression data coming from Illumina bead-based microarray platforms. Microarray data are often used to infer regulatory networks. Matias et al. discuss a probabilistic model for the occurrence of motifs in such networks, allowing investigators to identify significant motifs. Opgen-Rhein and Strimmer develop a graphical model for uncovering regulatory interactions using gene expression data observed over time. The longitudinal data theme continues into the last paper in the series, by Jung et al. They analyse protein expression data obtained from a cancer cell line using difference gel electrophoresis.

Finally we would like to express our thanks to Lisete Sousa and Luzia Gonçalves for all the work they put in the organization of this event.

Simon Tavaré
M.A. Amaral Turkman

INDEX

REVSTAT – Statistical Journal
Volume 4, Number 1, March 2006, 1–30

QUALITY CONTROL AND LOW-LEVEL STATISTICAL ANALYSIS OF ILLUMINA BEADARRAYS

Authors: Mark J. Dunning
– Department of Oncology, University of Cambridge, England
md392@cam.ac.uk

Natalie P. Thorne
– Department of Oncology, University of Cambridge, England
npt22@cam.ac.uk

Isabelle Camilier
– Ecole Polytechnique, 91128 Palaiseau, France
camilier@poly.polytechnique.fr

Michael L. Smith
– Department of Oncology, University of Cambridge, England
mls40@cam.ac.uk

Simon Tavaré
– Department of Oncology, University of Cambridge, England and
Department of Biological Sciences, University of Southern California, USA
s.tavare@damtp.cam.ac.uk

Abstract:

- The Illumina BeadArray™ platform is a novel microarray technology based on randomly assembled arrays of beads. Each bead on the array carries copies of a single gene-specific probe with, on average, about 30 replicates of each bead type on an array. Given the encouraging results regarding the reproducibility of BeadArray™ data and high profile studies already being carried out using the BeadArray™ technology, there is likely to be an increase in the volume of BeadArray™ data available. A major advantage of BeadArray™ technology is the high degree of replication of beads of a given type. However, current analysis methods give summarised information for each bead type as output rather than information for each individual bead on the array. The *beadarray* R package is able to recreate individual bead information for arrays using raw images as input. Here, we use a particular experiment to illustrate the image processing steps used by Illumina and corresponding methods available in *beadarray*. Our investigations into BeadArray™ data have demonstrated a high degree of reproducibility both within and between arrays. However, we identified some aspects of the low-level analysis that could be improved.

Key-Words:

- *Illumina; BeadArray; beadarray; Bioconductor; microarray.*

AMS Subject Classification:

- 62P10, 92C40, 92-08.

1. BACKGROUND

A BeadArray™ is an array of randomly positioned, three micron diameter, silica beads. Around 10^5 copies of a particular DNA sequence of interest are covalently attached to each bead ([15]). The position and identity of each bead on the array is determined using an automatic registration algorithm ([5]) and a molecular address ([7]). The DNA sequences attached to the beads are 75 base pairs in length, with 25 base pairs used for decoding and 50 base pairs for target hybridisation. This long-oligonucleotide approach has been shown to agree well with the popular short-oligonucleotide technology used by Affymetrix ([2]).

A pool of different bead types is created, beads of the same type having the same probe sequence attached. Separately, a fibre-optic bundle is treated with acid to create wells for individual beads to fit in ([10]). The fibre-optic bundle is exposed to the bead pool, causing the beads to be randomly sampled and assembled in the wells on the surface of the bundle.

Illumina have developed two different platforms which combine multiple BeadArrays. A Sentrix™ Array Matrix (SAM) contains 96 arrays, each of which has approximately 50,000 beads and around 1500 distinct bead types. A BeadChip™ allows either 24,000 bead types to be interrogated on eight samples simultaneously or 48,000 bead types across six samples. These multiple array technologies make the BeadArray™ platform especially suitable for high throughput experiments ([3], [1], [8]). The distribution of bead types on an array is effectively Poisson due to the random sampling of beads from the very large bead pool. Each bead type is represented about 30 times on average with extremely low probability of any bead type being represented less than five times ([9], [7]).

Large volumes of data can be generated using a single BeadArray™. Given these various new array technologies, there is clearly a need for statistical tools to analyse such data. There is already a wealth of software for statistical analysis of microarray data available in R packages on *Bioconductor* ([6], `www.bioconductor.org`). One of the most commonly used packages is *limma* (Linear Models for Microarray Analysis, [12]), which is an analysis package for two-colour microarrays. *beadarray* uses a similar programming style to *limma* and has recently been submitted as a development package to Bioconductor. The source for the package can be obtained at

`http://www.bioconductor.org/packages/bioc/1.8/html/beadarray.html`.

We used the data described in [16] to demonstrate the functionality of *beadarray* and to investigate the low-level analysis, processing and quality of BeadArray™ data. [16] studied the expression levels of some 700 genes measured in cell lines from 60 CEU individuals used in the Hapmap project ([3], [1]). The experiment comprises five SAMs with each of the 60 individuals replicated 6–8 times. Each array on the SAM had 1471 bead types with multiple (usually two) bead types for each gene under investigation and included various control probes.

2. PROCESSING OF BEADARRAY™ DATA BY ILLUMINA

In this section we describe the steps involved in the analysis of BeadArray™ data.

2.1. Image processing

The image processing steps used by Illumina to calculate bead intensities from raw images are described in [9]. These are given below:

(**i**) All pixel intensities are altered using a sharpening transformation. The intensity of a particular pixel is made higher / lower if its intensity is high / low in comparison to the intensities of the pixels surrounding it.

(**ii**) Foreground intensities are calculated as a weighted average of signals obtained using the four pixels nearest to each bead centre as a virtual bead centre. Sharpened pixel intensities are used in the calculation.

(**iii**) The local background, an average of the five dimmest pixels (unsharpened intensities) within the 17×17 pixel area around each bead centre, is subtracted.

Currently, raw TIFF images (see Figure 1) are read by an Illumina BeadArray Reader, giving bead-level data which can be read into the software package BeadStudio for analysis. At present, these bead-level data are encrypted and cannot be viewed directly. Bead-summary data can be output from BeadStudio with an unlogged, averaged intensity given for each bead type along with the number of beads used to calculate the average and the standard deviation of unlogged bead intensities.

2.2. Outlier removal and creation of bead-summary data

Outliers for each bead type are detected using the unlogged intensities of all beads of the same type. Any beads with intensity more than three median absolute deviations (MAD) from the mean are classed as outliers and excluded. The background measure reported for all beads is a single value, the mean of the negative controls on an array, rather than the local background values that are subtracted from each bead intensity.

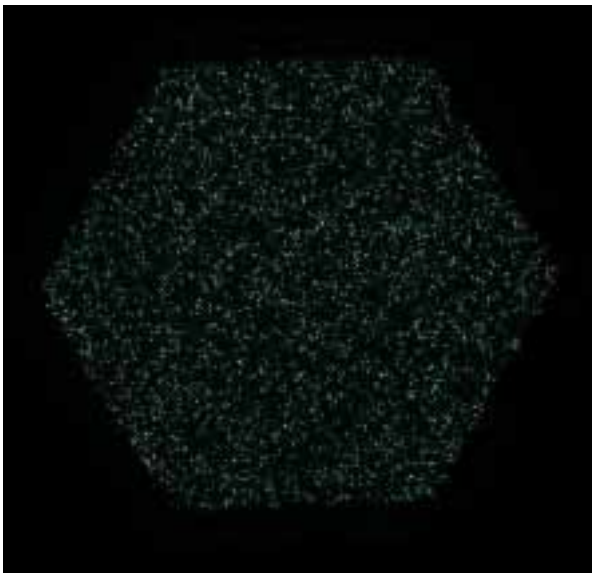

Figure 1: A raw image scanned from a BeadArray™ and viewed through BeadStudio. As with images scanned from conventional microarrays, each pixel intensity on the image represents the amount of hybridisation detected at each point on the array. Each three micron bead on the array is represented by nine pixels in a 3×3 square and are located roughly six microns apart.

2.3. Quality control of bead-summary data

Visualisation tools provided by BeadStudio include:

(**i**) plots of the unlogged intensities of control probes across all arrays;

(**ii**) scatter plots for comparing bead-summary data between two arrays;

(**iii**) interactive image plots of raw images with information about the intensity of each pixel (rather than bead intensity) on the image;

(**iv**) agglomerative clustering (average linkage method) of genes or samples using various distance and similarity measures.

The statistical analysis incorporated in BeadStudio for assessing differential expression between samples includes a Mann-Whitney test, Illumina custom (iterative robust least squares fit) or standard t-test. Normalisation choices include scaling by array averages, qspline [17] or scaling based on controls or rank invariant genes. Intensities may also have an additional background correction applied. This is based on the average of negative control genes and is called the method of background normalisation by Illumina.

3. PROCESSING OF BEADARRAY DATA using beadarray

The *beadarray* package was written to implement the analysis of BeadArray™ data in R in the same manner as two-colour microarrays or Affymetrix data, and to investigate image processing. An important feature of *beadarray* is the ability to access full bead-level detail for arrays rather than the bead-summary data given by BeadStudio. *beadarray* can also be used to analyse pre-processed bead-summary data created by BeadStudio.

3.1. Image processing

beadarray is able to create bead-level data by using the raw images scanned from BeadArrays, the locations of the bead centres and by implementing the steps described in Section 2.1. However, local background correction and sharpening are optional within *beadarray*, as is the background normalisation using negative controls. *beadarray* supports some of the background correction methods available in the *limma* R package, along with a selection of standard normalisation methods ([14]) found in *limma* and the *affy* packages.

3.2. Spatial plots

beadarray includes methods to identify automatically and to investigate any spatially dependent problems which may occur on BeadArrays. Users are able to screen all arrays in an ad-hoc manner to identify any arrays with unusual distributions of beads and these arrays can then be viewed in more detail.

We have implemented a test statistic to investigate the spatial randomness of a set of bead coordinates on an array. For this we divide the hexagonal array into eight sections and use a χ^2 goodness-of-fit test to assess the randomness of the number of beads found in each section (see Figure 2).

We use the χ^2 statistic to identify bead types on an array with apparent non-randomness and investigate these further with a spatial plot function. We also find it useful to apply similar χ^2 tests on the positions of the outliers so that arrays with spatial clustering of outliers can be quickly and easily identified. The raw images corresponding to the arrays can then be viewed through *beadarray* and the location of outliers in a region can be highlighted (see Figure 3). The function for displaying images can also be run interactively; clicking on a particular bead displays information such as the local background level, unsharpened intensity and identity of the bead.

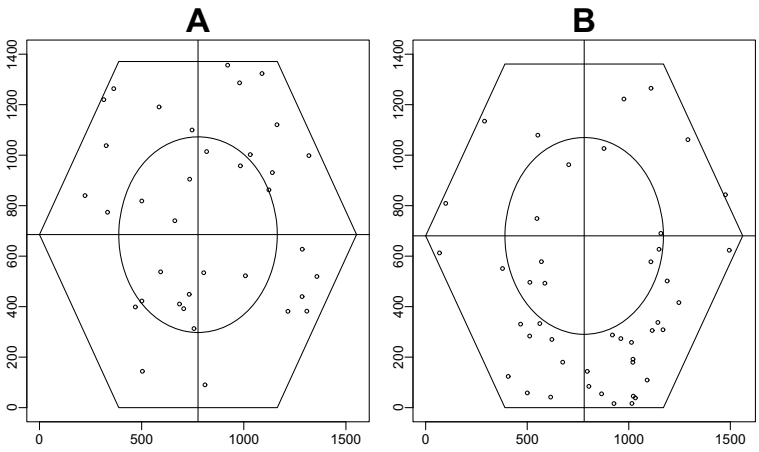

Figure 2: Using *beadarray* to assess randomness of bead positioning.
(**A**) The array is divided into eight sections of roughly equal area. The coordinates for a particular bead type are used to find the number of beads located in each of the sections. These are then compared to the expected number of beads if the beads were randomly distributed among all sections. The beads in this example are uniformly spread. (**B**) For this bead type there is a tendency for beads to be located in the lower half of the array.

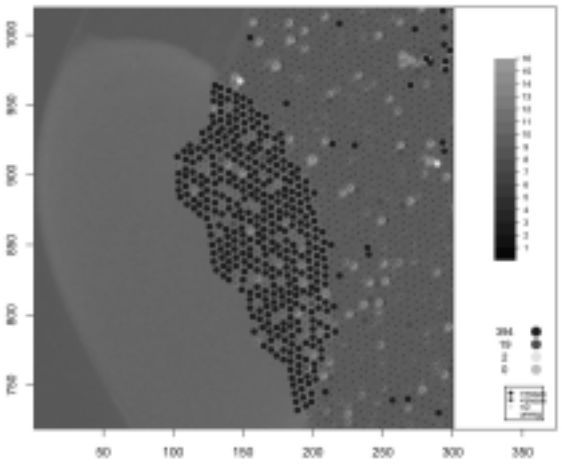

Figure 3: Viewing TIFF Images. Figure produced by *beadarray*.
beadarray allows regions on the original TIFF images to be viewed in more detail. The intensity of each pixel is given on the $\log_2$ scale with a brighter shade of green indicating a larger intensity. Bead centres are indicated by black crosses. This screenshot shows a region on an array with many outliers. Beads which are outliers are indicated by blue or red dots if their intensities are higher or lower than the mean for their bead type. Any beads which failed the decoding process can also be highlighted if desired. An interactive mode is also available whereby clicking on a particular bead gives information about that bead, such as the identity of the bead type and the foreground and background intensities. For colour picture see Supplementary Figures available at `http://www.damtp.cam.ac.uk/user/jcm68/beadarray.html`

3.3. Outlier removal

To visualise the variability of beads of the same type on the same array we plot the distance of each bead from the centre of the array against the $\log_2$ or unlogged intensity (see Figure 12). *beadarray* allows outliers to be identified using either unlogged or $\log_2$ intensities and using arbitrary numbers of MADs from the mean (the default is the Illumina setting of three MADs from the mean).

3.4. Quality Control

The average intensities of each bead type can be compared between multiple arrays using MA plots and scatter plots [4]. We can also compare the average intensity for any given bead type (not just control probes as in BeadStudio) between different arrays in the experiment and relate this information to the position of each array on the SAM using a "SAM Summary plot" (see Figure 4). The function used to create this SAM Summary plot can use any set of 96 values as input rather than just the values of control probes. For instance, one could plot the number of outliers found on each array to summarise the number of outliers on arrays over the SAM.

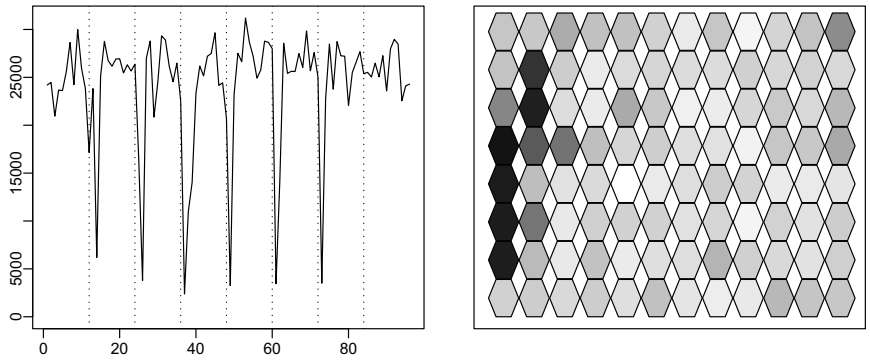

Figure 4: The SAM summary plot available in *beadarray*.
In this plot we show the bead average values for a particular control across all arrays on a SAM. In the left-hand plot the (unlogged) bead average value is plotted against array index. On the right-hand plot, 96 hexagons are shown in the same arrangement they appear on the SAM. The colour of each hexagon is related to the value of the bead average on that array; darker shades of grey indicate lower values. For this case we can see that the lowest bead average values occur on the left side of the SAM.

At present, *beadarray* is a package for quality control and low-level analysis only and does not provide any methods for determining differentially expressed genes. However, since *beadarray* was developed in the same programming style as

limma it should be straightforward to adapt existing methods for linear modeling ([11]). Accessibility to full bead-level data also makes it easier to combine the data from different arrays in a more flexible way. In particular, it is possible to produce weighted averages for each bead type over different replicates on either the $\log_2$ or unlogged scales. In BeadStudio, only unlogged values are available and therefore averages can only be combined on the unlogged scale.

4. RESULTS

In this section we show the results of our investigation into the methods used by Illumina for low-level analysis and image processing. This section also demonstrates some of the functionality available within *beadarray*.

4.1. Numbers and positioning of beads

The random sampling used in the construction of BeadArrays gives a random placement of beads and an average of approximately 30 of each bead type on an array. This randomness minimises the influence of spatially localised effects and lends robustness to the calculation of bead-summary data. The diagnostic tests described in Section 3 can be used to confirm the random nature of BeadArrays (see Figure 5). For every array under investigation, the mean number of beads of each bead type was found to be approximately 30 as expected.

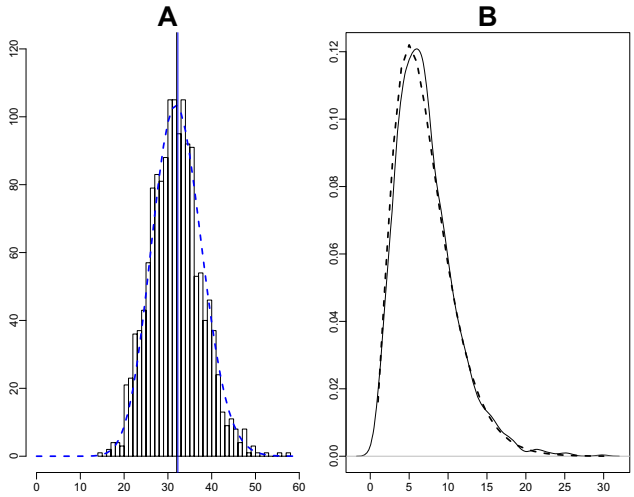

Figure 5: Random properties of BeadArrays. Figures produced by *beadarray*. (**A**) Histogram of the number of times each bead type is found on an array before outlier removal. The dotted line indicates the expected Poisson frequencies. (**B**) Distribution of the χ^2 statistics calculated for each bead type on an array. The dotted line indicates the expected χ^2 distribution.

The Poisson distribution of counts of each bead type shows that the probability of a bead type being represented less than five times is extremely low. Moreover, Illumina will not release an array where this has occurred. Even after outlier removal, no bead type on the five SAMs was found to have less than 11 beads on any array. Calculating the χ^2 statistic for all bead types on an array and repeating for all arrays confirms the random distribution of the beads.

4.2. Image processing

We used one array at a time from one SAM to look closely at the image processing steps used by Illumina. For each of these arrays, we took the raw image for the array and bead centre information provided by Illumina to calculate the foreground and local background intensities for each bead using the *beadarray* implementation of the steps described in Section 2.1. We did not perform background correction on the foreground intensities so that we could analyse the foreground and local background levels separately. Foreground intensities were also calculated without using the sharpening mask prior to averaging pixels for each bead (unsharpened intensities).

Figure 6 shows the effect of image processing on five arrays from the same SAM. Similar results were obtained for all arrays on the same SAM. From boxplot 6A we see that the unsharpened $\log_2$ intensities have a very low dynamic range and most of the bead intensities are concentrated between 10 and 11. As we are reading 16-bit images the maximum value for unsharpened $\log_2$ intensities is 16. However, if we apply sharpening to all pixels in the image and then calculate bead intensities, we affect the range of bead intensities produced. From the boxplots in 6B we see that the maximum values of bead intensities are now greater than 16 and the minimum values in the boxplots are lower than the minimum of boxplots in 6A. This implies that sharpening is increasing some bead intensities while decreasing others. The inter-quartile range of boxplots in 6B appear to be the same as for boxplots in 6A. In the boxplots in 6C we can see that the local background levels calculated for individual beads are virtually constant. Performing a local background correction on the sharpened intensities gives the intensities seen in boxplots in 6D. Similar results were produced using a global background correction (median of all local background measures). We notice that the dynamic range of boxplots in 6D are much higher than both boxplots in 6A and 6B.

Figure 7A confirms the results seen in boxplots in 6B. For most beads we see a positive difference between the sharpened and unsharpened intensities. However, it is also possible for sharpening to cause a decrease in bead intensities. This is most likely to happen for beads with low unsharpened intensities. For some beads with low intensity the effect of sharpening is very dramatic,

making the beads have very low intensity. Note from Figure 6A that most unsharpened beads lie within the range 10–11 on the $\log_2$ scale so could be highly altered by sharpening. Figure 7B also shows that background correction can have a very different effect depending on the unsharpened intensity. Local background correction is done using unlogged intensities. Therefore, it is not surprising that the higher intensity beads are less affected by local background correction. Lower intensities are affected much more dramatically by local background correction. For beads less than $\log_2$ intensity 11 the effect is nearly always negative, counteracting the increased intensity generally caused by sharpening. It seems that sharpening and local background correction have a non-linear effect on the data even on the $\log_2$ scale. This phenomenon is consistent on all arrays we investigated.

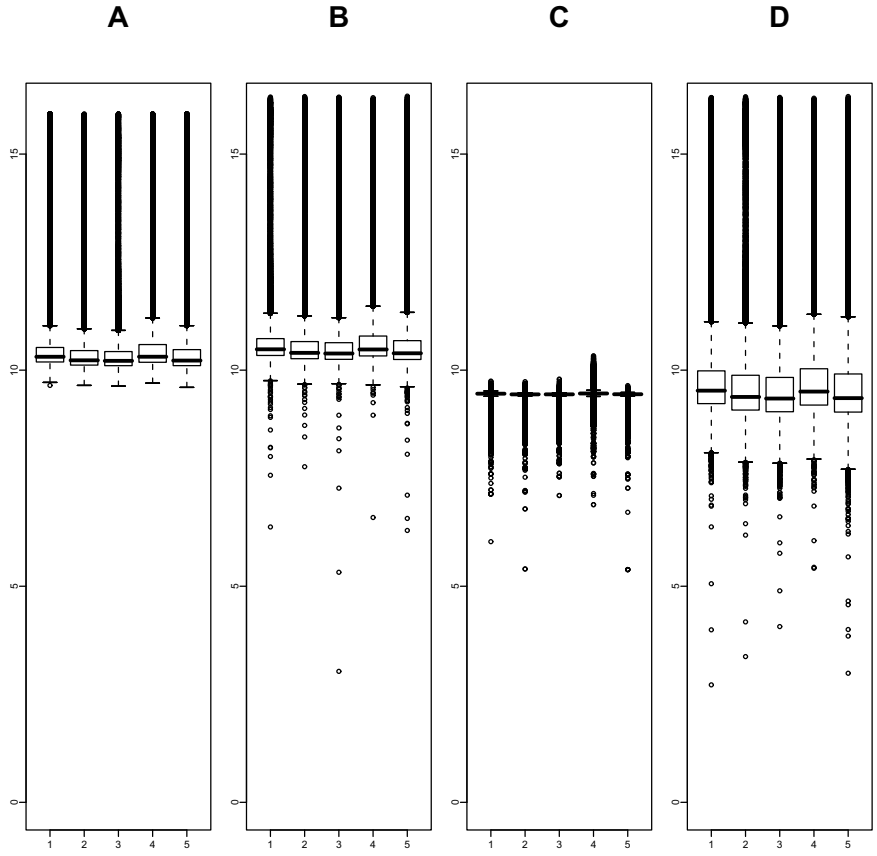

Figure 6: Effects of sharpening and background correction on raw intensities. Figures produced by *beadarray*.
(**A**) Boxplot of the unsharpened foreground intensities of all beads on the array. (**B**) Boxplot of foreground intensities of all beads on the array calculated using the sharpening mask. (**C**) Local background levels calculated for all beads on the array. (**D**) Boxplots of sharpened foreground intensities which have been background corrected by subtracting the local background. (These are the foreground intensities calculated by Illumina.)

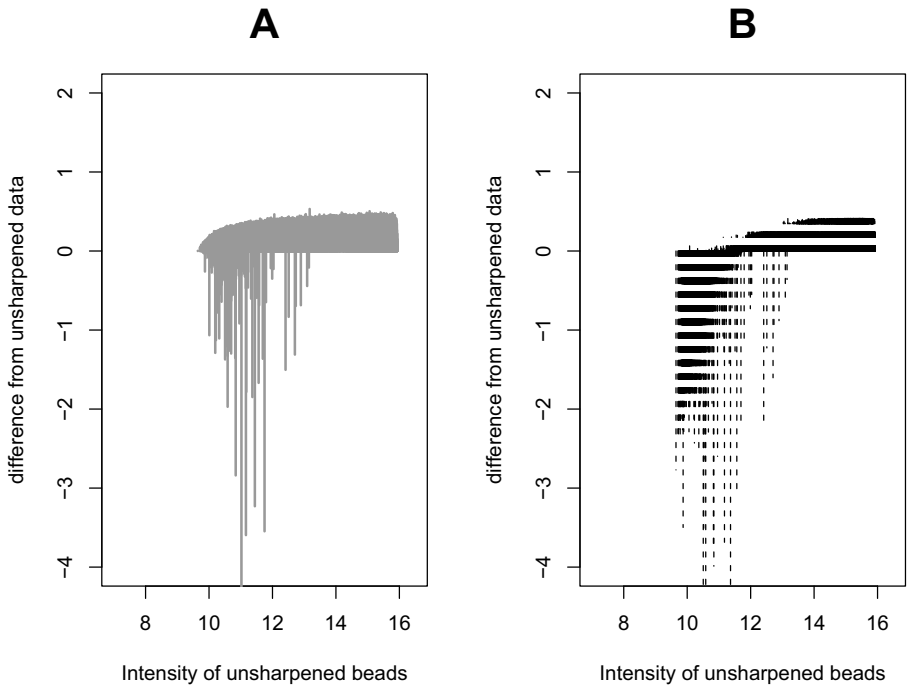

Figure 7: The effect of sharpening and local background correction on raw intensities for a particular array.
(**A**) The x axis represents the unsharpened intensities of all beads on an array and the y axis represents the difference in unsharpened and sharpened intensities for each bead. (**B**) Here, the y axis represents the difference between the unsharpened intensities of each bead and the intensity after sharpening and background correction. See Supplementary Figures at `http://www.damtp.cam.ac.uk/user/jcm68/beadarray.html` for a larger version of this figure.

4.3. Variability within bead types

Figures 6 and 7 demonstrate the global effect of sharpening and background correction on all beads on the array, but it is also of interest to know the effect on beads of the same type. In Figure 8 we show the standard deviation (SD) for all replicates of the first 50 bead types on an array both with and without sharpening and on the unlogged scale. Note that no outliers have been removed at this stage. The lowest SD is seen when foreground intensities are calculated without using sharpening or background correction. If we apply sharpening to all pixels in the image prior to calculating foreground intensities we see a slight increase in SD on the $\log_2$ scale. Background correction on the foreground intensities calculated using the sharpening mask results in a marked increase in SD.

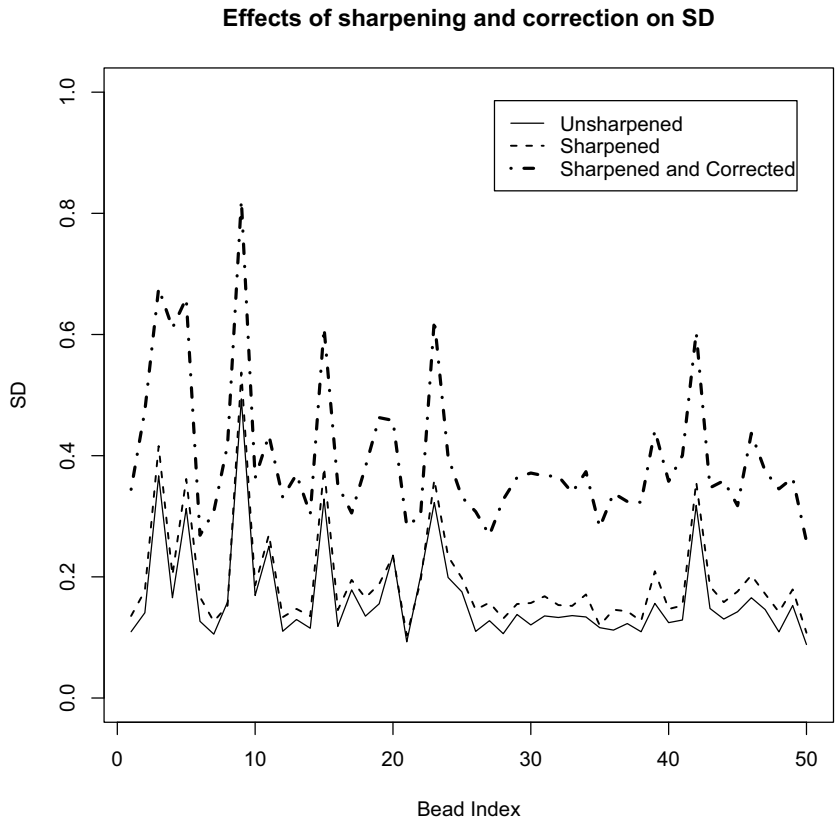

Figure 8: Effect of sharpening and local background correction on standard deviation of beads.
For the first 50 bead types on an array we show the standard deviation of all beads using $\log_2$ intensities which are unsharpened, sharpened or sharpened and background corrected.

4.4. Variability within and between arrays

We used one-way ANOVA on the unlogged foreground intensities on three replicate arrays to calculate the mean square error (MS) of each bead type within and between arrays. We repeated this using both sharpened and unsharpened intensities (see Figure 9). We again see that sharpening has the effect of increasing the variability between beads of the same type, both within and between arrays. The result is the same even when we use CV to measure the variability relative to the overall increased intensity due to sharpening. As would be expected, the variability between arrays is higher than the variability within arrays.

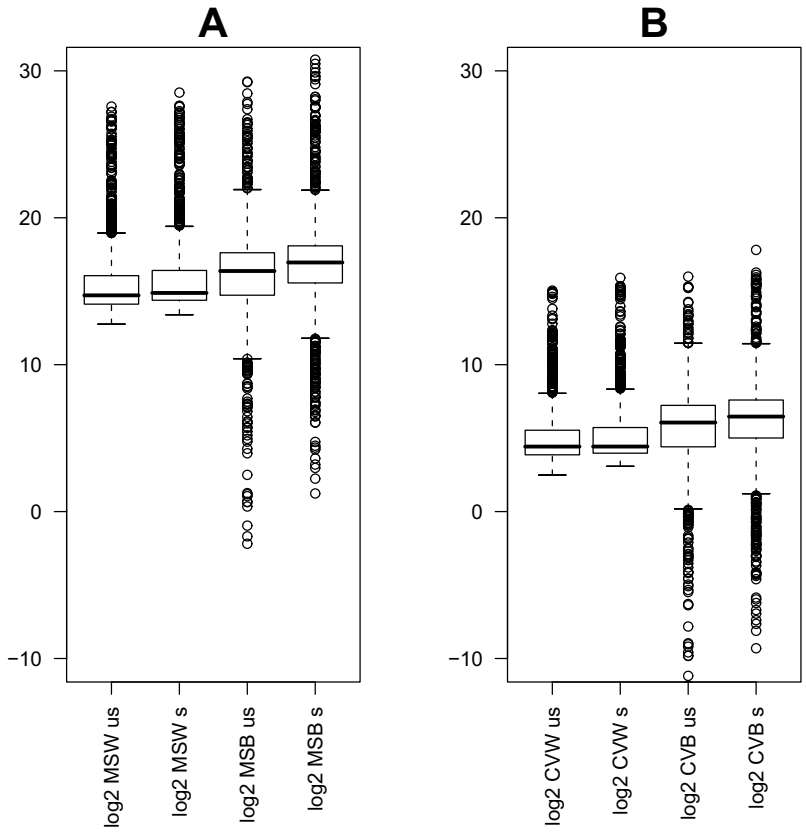

Figure 9: Within and between (replicate) array variability.
(**A**) log (base 2) of mean square error (MS) of bead types. (left to right) We show the $\log_2$ values of MS within arrays using unsharpened intensities, MS within arrays using sharpened intensities, MS between arrays using unsharpened intensities and MS between arrays using sharpened intensities. (**B**) Coefficient of variation (CV) of bead types. (left to right) We show the CV within arrays using unsharpened intensities, CV between arrays using sharpened intensities, CV between arrays using unsharpened intensities and CV between arrays using sharpened intensities. Calculations were made on unlogged data, then the MS and CV statistics were transformed to the $\log_2$ scale.

4.5. Effect of sharpening on outliers

Since sharpening has been shown to increase the variability of beads of the same type, we might also expect sharpening to have an effect on the outliers on an array. To investigate this in more detail we first used the unsharpened ($\log_2$) bead intensities and calculated the MAD for each bead. This was repeated using sharpened intensities and the MADs before and after sharpening were plotted

(see Figure 10). Most beads are seen to have MADs < 3 both with and without sharpening. However, some beads which have MADs > 3 without sharpening have MADs < 3 with sharpening (dark grey). In other words these beads were outliers without sharpening and the process of sharpening has made them no longer outliers. Similarly, some outliers are created by sharpening (light grey). The number of outliers created and removed by sharpening seems to be about the same. In general, we observe that beads have a lower MAD after sharpening. Further investigation revealed that out of the 1420 outliers detected on this array using sharpened intensities, 366 were removed by sharpening and 255 outliers were created by sharpening.

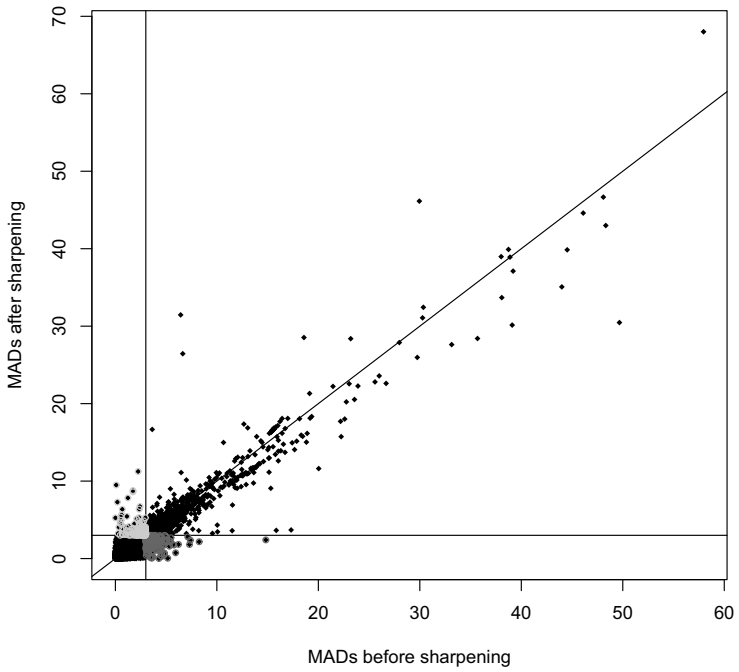

Figure 10: The effect of sharpening on outliers.
For all beads on an array we show the MAD of the bead calculated with and without using sharpening. The horizontal and vertical lines indicate 3 MADs (i.e. the cut-off that Illumina use to determine outlier beads). Dark grey spots indicate beads that are outliers without using sharpening but not outliers after sharpening whilst light grey spots indicate beads that are not outliers before sharpening but are outliers after sharpening.

In Figure 11 we show the total number of outliers on 12 arrays and how many outliers are created or removed by sharpening. It can be seen that the number of outliers created by sharpening is slightly higher than the number removed by sharpening. The majority of outliers for a particular array are unaffected by sharpening.

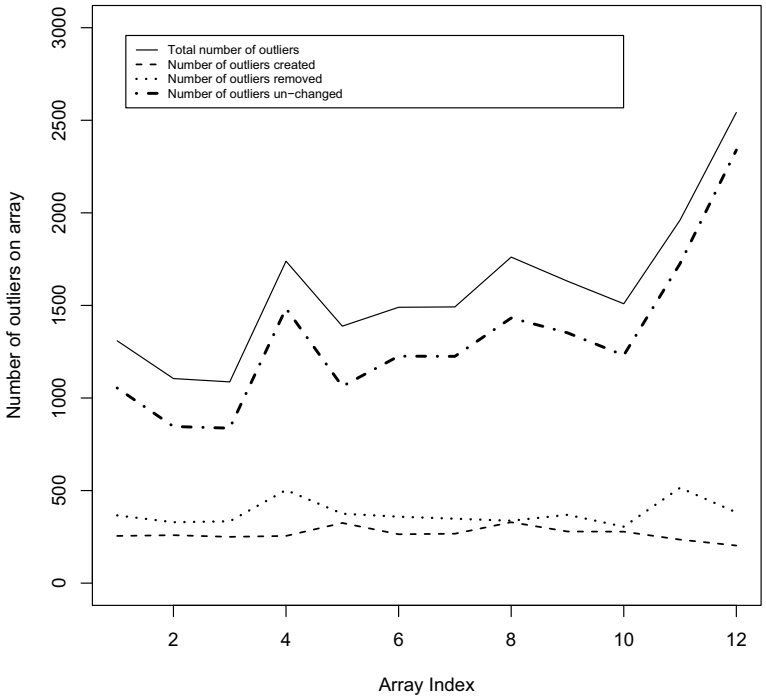

Figure 11: The effect of sharpening on outliers.
For 12 arrays we show the number of outliers which are removed and created by sharpening along with the total number of outliers on the array and outliers unchanged by sharpening.

4.6. The number of outliers on arrays

The number of outliers on each array is on the order of 1000–3000 beads; roughly one to two outliers per bead type or 1%–6% of total beads. These numbers are consistent across all arrays in the 96 array SAM and for all SAMs. In rare cases, as many as six or seven outliers were found for some bead types. However, only 0.5% of bead types on the 96 arrays had more than five outliers detected.

4.7. Outlier detection

As described in Section 3, *beadarray* offers a more flexible method for determining the outliers for a given bead type. We now use the sharpened, non-background corrected intensities for a particular bead type on an array to

demonstrate this flexibility (see Figure 12). The number of outliers detected for a particular bead type is dependent on the choice of scale used (unlogged or $\log_2$). In Figure 12 it can be seen that this choice has little effect on the total number of outliers that are detected on an array.

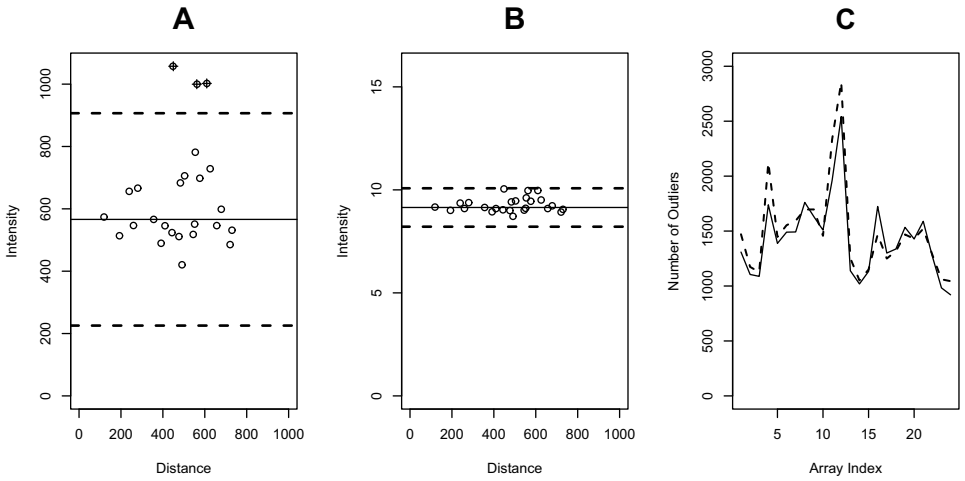

Figure 12: Outlier detection using *beadarray* and the effect of the choice of scale. (**A**) Unlogged intensities of all beads of a particular type on the same array are plotted on the y axis. The dotted horizontal lines represent a shift of 3 MADs from the mean (solid horizontal line). Beads outside the dotted lines are outliers for this bead type. Plotting the distance of each bead from the centre along the x axis allows for the possibility of identifying spatial effects on the array. (**B**) Intensities for the same bead type shown on the $\log_2$ scale. As before dotted lines represent a shift of 3 MADs from the mean. (**C**) The number of outliers detected in the sharpened, non-background corrected intensities on 24 arrays using either the $\log_2$ or unlogged scale.

4.8. Spatial plots of outliers

It is important to know where outliers are located on arrays as this can indicate possible spatial artifacts. Figure 3 shows a region on the far left tip of one of the arrays under investigation. It is quite apparent that this part of the array contains a significant spatial artifact. The consequence of this artifact is that beads lying within the affected area show an increased hybridisation level and are subsequently classified as outliers. Note that Figure 3 serves as an extreme example and spatial artifacts were only detected on a small number of arrays. Spatial artifacts were found more often around the edges of arrays and can be automatically identified through the *beadarray* package using the χ^2 statistic without systematically viewing each array in the experiment.

4.9. Effect of image processing on bead averages

We have previously shown that sharpening and background correction have the effect of increasing the SD of replicate beads. We now ask if this increase in SD among replicates has an effect on the resultant bead averages. In Figure 13 we plot the average $\log_2$ values (with outliers on the $\log_2$ scale removed) for four control probes across 12 arrays without sharpening, with sharpening and with both sharpening and background correction. Although the averages calculated using sharpened intensities and averages calculated using sharpened and background corrected intensities are lower, we can clearly see the same trend in all sets of averages. Controls A, B and C are housekeeping controls so show high intensity across all arrays. The effect of background correction on these averages seems minimal. The final control is a negative control and therefore we would expect it to have low intensity across all arrays. The averages calculated using background corrected intensities are much lower. As we saw previously, the effect of background correction is much greater on lower intensity beads.

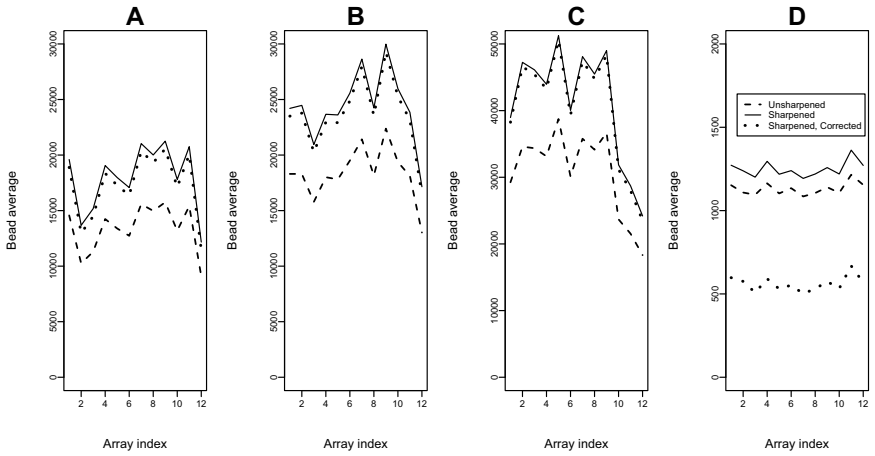

Figure 13: Effect of sharpening on bead summaries for particular control probes. Figures produced by *beadarray*.
The (unlogged) variation in bead-summary values calculated using unsharpened, sharpened and sharpened background corrected bead intensities for 12 arrays (x axes) in the same experiment. A, B and C show the results for three hybridisation controls respectively and D shows the results for a negative control across the 12 arrays.

For each bead type on each array we calculated the difference between the bead averages obtained using unsharpened and sharpened intensities (see Figure 14). This shows that the difference between the averages calculated using sharpened or unsharpened intensities is around 0.2. Furthermore, for particular bead types, the difference between averages is fairly constant across arrays (the SD is 0.01).

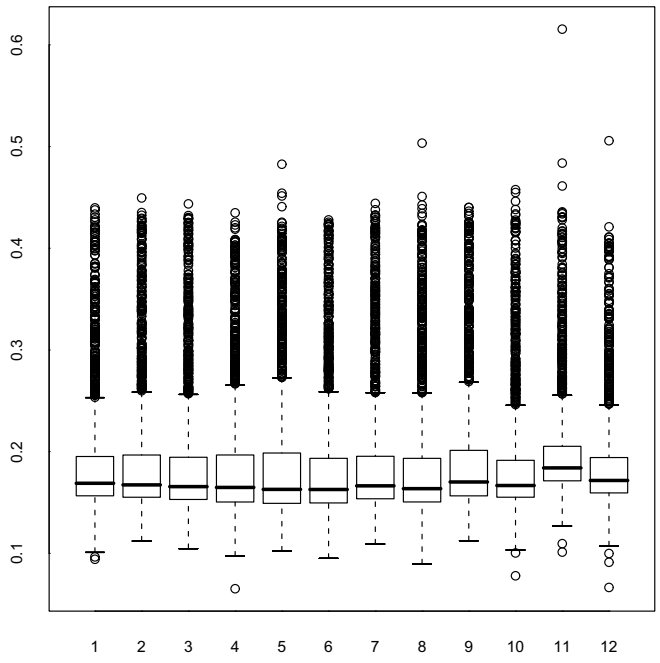

Figure 14: Effect of sharpening on all bead averages on an array. For 12 arrays we show the difference between the $\log_2$ bead averages calculated using unsharpened or sharpened intensities for every bead type on the array.

4.10. Variability between arrays

The generation of bead-summary data allows for conventional plots to be used to compare probe intensities between arrays. We used the bead-summary data from the first four arrays on the SAM (the first three of which are replicate arrays) to make MA and scatter plots (Figure 15) and density plots (Figure 16). Comparing replicate arrays allows us to observe both random and systematic sources of variation. In Figure 15, the majority of points lie along the diagonal for the scatter plots and along the central line for the MA plots. There is little noise in the MA plots between replicate arrays and only some intensity dependent bias is apparent (i.e. between Arrays 1 and 2). Comparisons involving the fourth array show more variation as this array is a different sample to the other three. In Figure 16 we see that the distribution of bead-summary intensities for these arrays are similarly shaped. Apart from the obvious need for location normalisation between these arrays, non-linear effects in the bead-summary intensities between them are minimal. Similar observations were made between arrays across the whole SAM. Comparing the average intensities of beads on the first array to all other 95 arrays gave a median correlation coefficient of 0.98.

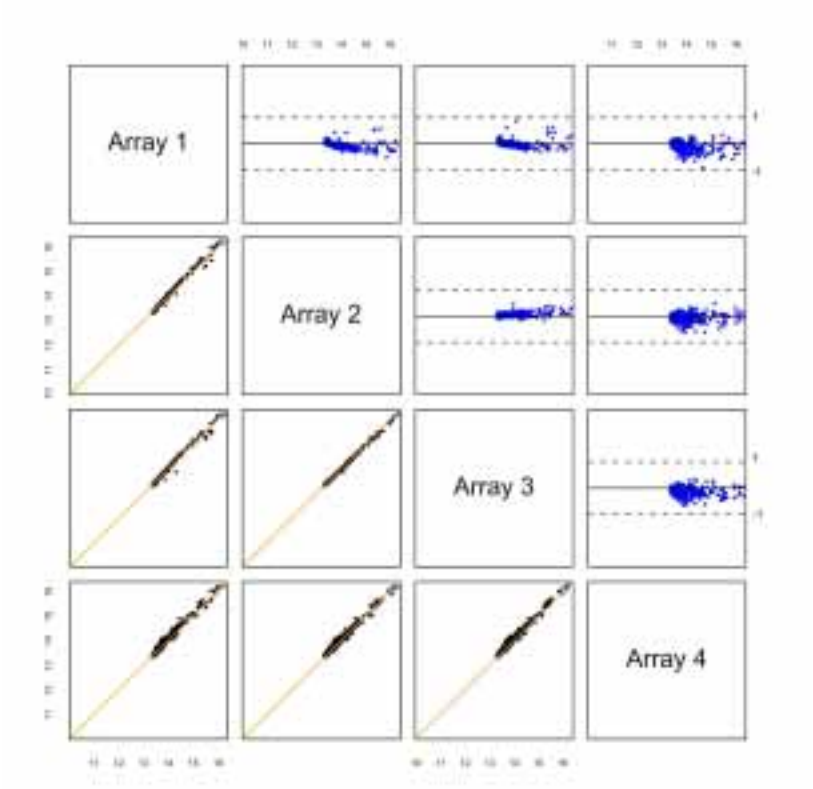

Figure 15: Comparing average bead-type intensities. Figures produced by *beadarray*. Scatter and MA-plots for the first four arrays on a SAM are shown. The intensities shown have been sharpened but not normalised or background corrected. The first three arrays are replicates of the same sample, hence display less variation.

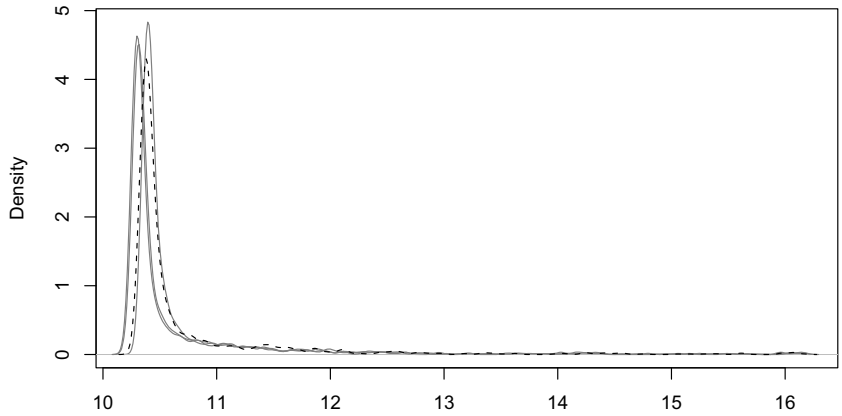

Figure 16: Comparing density plots.
We show the density plots of the bead averages for the four arrays shown in Figure 15.

5. DISCUSSION

Our preliminary investigations suggest a high degree of reproducibility with BeadArray™ data. Arrays are seen to exhibit highly similar distributions even before any normalisation has taken place. The precision of replicates of the same bead type is high both within and between arrays. Given the low variability of BeadArrays there might be a danger of over-normalising the data and removing important biological information. Preliminary investigations suggest that a quantile or qspline [17] normalisation is sufficient (data not shown [16]) for bead-summary data. However, we feel that in general such normalisation options should be investigated and performed on the bead-level rather than bead-summary data.

Our main finding concerning image processing was that the sharpening transformation used prior to calculating foreground intensities causes an increase in variability. The transformation is designed in such a way that high intensity pixels (with respect to their neighbours) are made even higher and low intensity pixels are made lower. Therefore, it is not surprising to find intensities greater than 16 on the $\log_2$ scale after sharpening. However, it appears that the intensities of beads of the same type are being altered independently of each other and this causes an increase in variance. As a result of sharpening, we observed that the bead type averages increase by around 0.2 and their SD's increase by 0.01.

The background values for beads were found to be virtually constant within arrays and also across arrays. Correcting using the local background measure is effectively equivalent to using a global value. Background corrected data show much more variability among beads of the same type. Automatic background correction on BeadArray™ data cannot therefore be recommended. *beadarray* does not perform correction automatically, thereby allowing foreground and background levels to be analysed separately. It should be emphasised that the bead-summary data produced by BeadStudio is automatically background corrected using the local background measures. Therefore any attempts to correct pre-processed bead-summary data (as given by the normalisation methods supplied by BeadStudio) may have an adverse effect.

The *beadarray* package can be used to highlight and understand problems that can occur with BeadArrays. We found random numbers and positioning of beads on all arrays. The random positioning minimises the effects of spatial artifacts on bead-summary data; such artifacts were rare. Due to the high replication of beads, any beads which occur inside such regions of unusual intensity are declared as outliers and can be removed from analysis without affecting the bead average values too much. The distribution of beads gives, on average, about 30 beads of each type. Depending on the scale of intensities used to detect outliers (either unlogged or $\log_2$) we might expect one or two of these beads to be detected as outliers. For the five SAMs in the investigation, no bead type on any array was found to have less than 11 replicates after outlier removal.

The number of outliers for each bead type can be detected by using either unlogged or $\log_2$ intensities. If we apply a $\log_2$ transformation to the data we decrease the range of the intensities. The purpose of such a transformation is to make changes in intensity comparable across the whole intensity range. Converting to the $\log_2$ scale also tends to make the variability more constant ([13]). Outliers which appear extreme on the unlogged scale will be much closer to the mean on the $\log_2$ scale. Therefore, it might be more consistent to use $\log_2$ intensities to calculate outliers if the intention is to use $\log_2$ intensities in analysis. In practice, the decision to use unlogged or $\log_2$ intensities to determine outliers had very little effect on the bead averages produced. Bead averages show very low variability across replicate arrays of the same sample.

We believe that *beadarray* offers a very flexible platform for the analysis of BeadArray™ data. By recreating bead-level data from scratch, users are given access to more information about each individual bead on an array. Making sharpening and background correction optional gives the opportunity to use diagnostic checks and make an informed choice about how data should be pre-processed. As the amount of BeadArray™ data available is relatively small we can only make recommendations for how data should be pre-processed based on our experience whilst developing the package. Analysing bead-summary data using *beadarray* offers a greater range of plotting tools than existing methods. Most importantly, all functions can deal with intensities on the $\log_2$ scale as is common for microarray analysis. Whilst *beadarray* does not currently provide any methods for detecting differential expression, these will be implemented in future versions. *beadarray* can also provide a more flexible analysis of bead-summary data pre-processed by Illumina.

In our study we have demonstrated how *beadarray* can be used for quality control and low-level analysis. We have presented some findings about the impact of the image processing steps used by Illumina on a particular experiment. The conclusions we reach in this paper may not indeed be valid in all cases and subsequent experiments will need to be analysed in a similar manner before more general conclusions can be reached. At the time of developing the package, only SAM data were available to us. We are currently expanding the package to include the analysis of BeadChip™ data. An investigation into normalisation methods will be linked with an implementation of methods for assessing differential expression. It will be of interest to see how sharpening and background correction affect the genes that are selected as differentially expressed.

ACKNOWLEDGMENTS

We thank Brenda Kahl and Semyon Kruglyak (Illumina), Barbara Stranger, Matthew Forrest and Manolis Dermitzakis (Wellcome Trust Sanger Institute) and Andrew Lynch (University of Cambridge) for many helpful discussions during the devlopment of this work. We also thank James Brenton (University of Cambridge) for useful advice on reading images into R. The authors were supported by grants from the US National Institutes of Health, Cancer Research UK and the Medical Research Council. Simon Tavaré is a Royal Society / Wolfson Research Merit Award holder.

REFERENCES

[1] Altshuler, D.; Brooks, L.D.; Chakravarti, A.; Collins, F.S.; Daly, M.J.; Donnelly, P. and the International HapMap Consortium (2005). A haplotype map of the human genome, *Nature*, **437**, 1299–1320.

[2] Barnes, M.; Freudenberg, J.; Thompson, S.; Aronow, B. and Pavlidis, P. (2005). Experimental comparison and cross-validation of the Affymetrix and Illumina gene expression analysis platforms, *Nucleic Acids Res.*, **33**, 5914–5923.

[3] The International HapMap Consortium (2003). The International HapMap Project, *Nature*, **426**, 789–796.

[4] Dudoit, S.; Yang, YH.; Speed, T.P. and Callow, M.J. (2002). Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiments, *Statistica Sinica*, **12**, 111–140.

[5] Galinsky, V.L. (2003). Automatic registration of microarray images. II. Hexagonal grid, *Bioinformatics*, **19**, 1832–1836.

[6] Gentleman, R.C.; Carey, V.J.; Bates, D.M.; Bolstad, B.; Dettling, M.; Dudoit, S.; Ellis, B.; Gautier, L.; Ge, Y.; Gentry, J.; Hornik, K.; Hothorn, T.; Huber, W.; Iacus, S.; Irizarry, R.; Leisch, F.; Li, C.; Maechler, M.; Rossini, A.J.; Sawitzki, G.; Smith, C.; Smyth, G.; Tierney, L.; Yang, J.Y. and Zhang, J. (2004). Bioconductor: open software development for computational biology and bioinformatics, *Genome Biol.*, **5**, R80.

[7] 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.; Haas, J.; Zhou, L.; Oliphant, A.; Fan, J.B.; Barnard, S. and Chee, M.S. (2004). Decoding randomly ordered DNA arrays, *Genome Res.*, **14**, 870–877.

[8] Gunderson, K.L.; Steemers, F.J.; Lee, G.; Mendoza, L.G. and Chee, M.S. (2005). A genome-wide scalable SNP genotyping assay using microarray technology, *Nat. Genet.*, **37**, 549–554.

[9] Kuhn, K.; Baker, S.C.; Chudin, E.; Lieu, M.H.; Oeser, S.; Bennett, H.; Rigault, P.; Barker, D.; McDaniel, T.K. and Chee, M.S. (2004). A novel, high-performance random array platform for quantitative gene expression profiling, *Genome Res.*, **14**, 2347–2356.

[10] Oliphant, A.; Barker, D.L.; Stuelpnagel, J.R. and Chee, M.S. (2002). BeadArray technology: enabling an accurate, cost-effective approach to high-throughput genotyping, *Biotechniques*, **Suppl.**, 56–58, 60-61.

[11] Smyth, G.K. (2004). Linear models and empirical Bayes methods for assessing differential expression in microarray experiments, *Statistical Applications in Genetics and Molecular Biology*, **3**, 113–136.

[12] Smyth, G.K. (2005). *Limma: linear models for microarray data*. In "Bioinformatics and Computational Biology Solutions using R and Bioconductor" (R. Gentleman, V. Carey, W. Huber, R. Irizarry and S. Dudoit, Eds.), Springer, New York, 397–420.

[13] Smyth, G.K.; Michaud, J. and Scott, H.S. (2005). Use of within-array replicate spots for assessing differential expression in microarray experiments, *Bioinformatics*, **21**, 2067–2075.

[14] Smyth, G.K. and Speed, T. (2003). Normalization of cDNA microarray data, *Methods*, **31**, 265–273.

[15] Steinberg, G.; Stromsborg, K.; Thomas, L.; Barker, D. and Zhao, C. (2004). Strategies for covalent attachment of DNA to beads, *Biopolymers*, **73**, 597–605.

[16] Stranger, B.E.; Forrest, M.S.; Clark, A.G.; Minichiello, M.J.; Deutsch, S.; Lyle, R.; Hunt, S.; Kahl, B.; Antonarakis, S.E.; Tavaré, S.; Deloukas, P. and Dermitzakis, E.T. (2005). Genome-wide associations of gene expression variation in humans, *PLoS Genet.*, **1**(6), e26.

[17] Workman, C.; Jensen, L.J.; Jarmer, H.; Berka, R.; Gautier, L.; Nielser, H.B.; Saxild, H.H.; Nielsen, C.; Brunak, S. and Knudsen, S. (2002). A new non-linear normalization method for reducing variability in DNA microarray experiments, *Genome Biol.*, **3**, research0048.

6. APPENDIX

The purpose of this appendix is to give an outline of the R functions for analysing bead-level data. Descriptions of how to read and analyse bead-summary data are provided in the Vignette distributed with *beadarray*. Those who are familiar with the R statistical language, and in particular the *limma* package, should be able to adapt easily to our new methods of analysis. Wherever possible we used objects that are similar to those used by *limma*. Example files to read bead-level data are provided at

`http://www.damtp.cam.ac.uk/user/jcm68/beadarray.html`

6.1. Reading bead-level data

There are two sets of files that are required by our package in order to create bead-level data.

- TIFF images — These are the raw images scanned directly from each individual array on a 96-well SAM. These are provided by Illumina.
- csv files — These define the location and bead type of each individual bead on a particular array on a 96-well SAM.

Before these files can be read into R, we first convert the TIFF files into PGM files. This can be done be using the *ImageMagick* utility [1]. A batch file is included with this library to convert automatically all the TIFF files in a directory.

If the correct csv and pgm files are available we can read these data into R using `readBeadImages`. This function requires a `beadTargets` object that can be read directly from a beadTargets.txt file. This `beadTargets` object specifies the filename of each image and csv file to be read.

In our example dataset we have two single channel BeadArray™ hybridisations. Since the arrays are hybridised with one target only (one-colour), we need only to specify one image file for each array. The beadTargets.txt file, pgm and csv files are provided at `http://www.damtp.cam.ac.uk/user/jcm68/beadarray.html`. Once downloaded, these files can be read into R. The R working directory can be set to the folder containing these files using the `setwd` function or the file menu (GUI implementation only). Alternatively the *path* argument in `readBeadImages` can be set to the current directory. The commands to read the data into R are then as follows:

```
> library(limma)
> library(beadarray)
> beadTargets = readBeadTargets()
> beadTargets

               Image1                  xyInfo SAMPLE
1269941_R001_C001.pgm 1269941_R001_C001.csv      6
1269941_R001_C002.pgm 1269941_R001_C002.csv      6

> BLData = readBeadImages(beadTargets)

Calculating foreground intensities for 1269941_R001_C001.pgm
Calculating background intensities.
Calculating foreground intensities for 1269941_R001_C002.pgm
Calculating background intensities.
```

[1] available from `www.imagemagick.org` — version 6.2.2 or later is required.

The default setting for `readBeadImages` is to recreate the foreground and background intensities for each bead in the same way in which they are calculated by Illumina. However, the use of sharpening and local background correction are optional (see section 3). To create unsharpened bead intensities one would use:

```
> BLData.ns = readBeadImages(beadTargets, sharpen = FALSE)
```

6.2. The BLData Object

The data object (`BLData`) is in fact a list object but behaves like a complex sort of matrix. It can be subsetted or treated like a matrix in lots of ways. We can use the `names` command to see what items can be found in the list. `BLData` is an `BeadLevelList` object and like the `RGList` object in *limma* can contain `R`, `Rb`, `G` and `Gb` objects (i.e. foreground and background intensities of two colour data).

```
> is(BLData)

 [1] "BeadLevelList"          "list"           "LargeDataObject" "vector"

> names(BLData)

 [1] "R"                   "Rb"                   "x"
 [4] "y"                   "probeID"              "targets"
 [7] "sharpened"           "backgroundSize"       "normalised"
[10] "backgroundCorrected"
```

Individual items in the list can then be accessed by using the $ operator in R. In our example we have the matrices `R` and `Rb` which are the foreground and background intensities for each bead (row) and each array (column). The example shown here is for a single channel experiment, hence we only have a foreground intensity value in the red channel and the green channel is not used. If we had two channel data then `BLData$R` and `BLData$G` would be the red and green channels respectively. The number of rows in the matrix is the same as the number of beads present on the array and the number of columns is the same as the number of arrays. In this example we only read in two arrays, so we only have two columns. In other words, each column of the matrix represents intensities of all beads on the same array. However, due to the random placement of beads on the array, each row of the matrix does not relate to intensities of a bead of the same type (as one might expect having dealt with conventional microarray data).

Since BeadArray™ technology uses randomly assembled beads it is important to know the location and identity of every bead on the array. Therefore the `BeadLevelList` we are using in this library also contains the x and y co-ordinates for each bead and an identifier (ProbeID) for the bead type of each bead.

6.3. Background correction and normalisation

Using the `boxplot` function in R allows boxplots of foreground and background intensities to be compared (see Figure 6).

```
> boxplot(log2(BLData$R)~col(BLData$R))
```

Background correction can be be performed on the data by:

```
> BLData.c = backgroundCorrectBeads(BLData)
```

By default, the `backgroundCorrectBeads` function subtracts the values in `BLData$Rb` from `BLData$R` and stores the result in the `R` matrix of the resulting *BeadLevelList* object. Other methods are available such as *minimum* which ensures that no negative values are produced. The only normalisation methods currently supported for bead-level data are median and quantile normalisation.

```
> BLData.med = medianNormalise(BLData)
> BLData.q = quantileNormalise(BLData)
```

6.4. Numbers of beads

Figure 5A can reproduced using the command

```
> histBeadCounts(BLData, array=1)
```

Additonally we can also see which bead types are represented less than 24 times (the 5th percentile for the appropriate Poisson distribution) on the array using `findLowestCounts`. For the first array that we use:

```
> findLowestCounts(BLData, 1)[1:10]

 [1]   10   23   30   42   87  119  182  185  585  607
```

For clarity only the first 10 results returned by the function are shown.

6.5. Outliers for each bead type

The `plotBeadIntensities` function was used to produce Figure 12. This function shows the intensity of every bead of a particular type against the distance of the bead from the centre of the array. Any outliers which exist for the bead

type are marked on the plot by a red cross. As an example we can plot the intensities of all beads with ProbeID 2 on array 1 and determine outliers using unlogged or $\log_2$ intensities. This function also has the option of changing the number of MADs from the mean used to determine outliers by changing the n parameter.

```
> par(mfrow = c(1, 2))
> plotBeadIntensities(BLData, probe=2, array=1)
> plotBeadIntensities(BLData, probe=2, array=1, log = TRUE)
```

The function `findOutliers` is used within `plotBeadIntensities` to determine the outliers for a particular bead type on an array. The function `findMostOutliers` can be used to find which bead types have more than a set number of outliers (the default is 5 outliers).

```
> findMostOutliers(BLData, array=1)

 [1]  807 1702 2458 5244 5917 6015 6117
```

We can find all the beads on an array which are outliers for their bead type by using the `findAllOutliers` function. The output of the function is an index between 1 and 49777 which refers to a particular bead on the array (beadID).

```
> o = findAllOutliers(BLData, array=1)
> o[1:10]

 [1] 44634  1263  8245   342 23176  6270  8898 31023  4273 15610
```

The length of the list can be easily found (`length`) and compared between different arrays as a diagnostic measure for the quality of the array. Additionally, the location of all the outliers on an array can be plotted using the `plotBeadLocations` function.

6.6. Spatial plots

The `plotBeadLocations` function can be used to plot the location of a set of beads on an array. Beads can be specified by a list of ProbeIDs or beadIDs. Figure 2A can be generated by using:

```
> plotBeadLocations(BLData, probeIDs=2, array=1)
```

The function plots all beads on the first array with ProbeID 2. By using the `o` object created above we can also plot the location of all outliers on the first array.

```
> plotBeadLocations(BLData, beadIDs=o, array=1)
```

The `plotBeadLocations` provides a quick diagnostic check for the distribution of a set of beads. As described in Section 3.2, we have also implemented a χ^2 statistic to quantify the non-randomness of bead distributions. This χ^2 test can be applied to all bead types on an array and the ProbeID of those with the highest value can be returned by:

```
> findHighestChis(BLData, array=1)[1:10]

 [1]  213  278  606  658  791  800  936  960  961 1071
```

Shown above are the ProbeIDs for the first 10 bead types with a χ^2 statistic greater than 14 (chosen because this is the 5th percentile of the appropriate χ^2 distribution). Any of these bead types can be investigated further by using the `plotBeadLocations` function.

Any regions on an array found to have a high proportion of outliers can be investigated further by the `displayTIFFImage` function.

```
> displayTIFFImage(BLData, array=1, a = 1000:1400, b=1200:1400)
```

The example above loads the original image for array 1 (the name of which is stored in the targets object) and displays the intensities of pixels with x in the range from 1000 : 1400 and y in the range 1200 : 1400.

The intensity of every pixel in the plot is represented by a shade of green, with brighter colours indicating a higher value. The blue and red spots indicate the position of outliers in the particular region with blue indicating beads with intensity higher than the mean for that bead type and red being beads with intensity lower than the average for their bead type. Yellow spots on the picture represent beads which have been calculated to have a negative foreground intensity. The black crosses show where the bead centres are located. Any beads which failed the decoding process can also be highlighted by setting the *showUnregistered* parameter.

The plot can also be made interactive by setting the *locateBeads* parameter to TRUE. We can then click on any bead centre and display the foreground and background intensities for this bead as well as a measure of the raw intensity.

We feel that `displayTIFFImage` gives more useful information about the raw images than the equivalent function included in BeadStudio. In BeadStudio, the user can explore the TIFF images and see the intensity of each individual pixel. However, the identity of each bead on the image is not given and there is no information about outliers.

6.7. Creating bead-summary data

Bead-summary data can be created using the bead-level data. In producing these summaries we must first remove outliers for each bead type as described in Section 3.3. This averaging is done by the `createBeadSummaryData` function and the method of detecting outliers can be specified by changing the *log* (for unlogged or logged parameters) and n (number of MADs) parameters.

```
> BSData = createBeadSummaryData(BLData)
```

The structure of the resulting object is described in greater detail in the Vignette supplied with *beadarray*.

REVSTAT – Statistical Journal
Volume 4, Number 1, March 2006, 31–51

NETWORK MOTIFS: MEAN AND VARIANCE FOR THE COUNT

Authors: C. Matias
– UMR CNRS 8071, Laboratoire Statistique et Génome,
91000 Evry, France
matias@genopole.cnrs.fr

S. Schbath
– INRA, Unité Mathématique, Informatique et Génome,
78352 Jouy-en-Josas, France
Sophie.Schbath@jouy.inra.fr

E. Birmelé
– UMR CNRS 8071, Laboratoire Statistique et Génome,
91000 Evry, France
birmele@genopole.cnrs.fr

J.-J. Daudin
– UMR ENGREF/INAPG/INRA,
Mathématiques et Informatique Appliquées, INA P-G, 75231 Paris, France
daudin@inapg.inra.fr

S. Robin
– UMR ENGREF/INAPG/INRA,
Mathématiques et Informatique Appliquées, INA P-G, 75231 Paris, France
robin@inapg.inra.fr

Abstract:

- Network motifs are at the core of modern studies on biological networks, trying to encompass global features such as small-world or scale-free properties. Detection of significant motifs may be based on two different approaches: either a comparison with randomized networks (requiring the simulation of a large number of networks), or the comparison with expected quantities in some well-chosen probabilistic model. This second approach has been investigated here. We first provide a simple and efficient probabilistic model for the distribution of the edges in undirected networks. Then, we give exact formulas for the expectation and the variance of the number of occurrences of a motif. Generalization to directed networks is discussed in the conclusion.

Key-Words:

- *network motif; motif count; random graph; sequence of degrees.*

AMS Subject Classification:

- 62P10; 62E15; 05C80.

1. INTRODUCTION

A cellular system can be described by a web of relationships between proteins, genes or more generally metabolites. Studying its basic structural elements, also called **motifs**, is a first step in the understanding of these networks that goes beyond global features (such as the small world or scale-free properties, see [2, 12]). For instance, motifs that occur more frequently than expected in random networks may reveal particular structures corresponding to biological phenomena. Several definitions exist for a network motif. Here we consider the most commonly used: a simple pattern of interconnection in a graph. Detection of significant motifs [7] may be based on two different approaches: either by comparing the observed network with appropriately randomized networks (this requires the simulation of a large number of networks), or by the comparison with expected quantities in some well-chosen probabilistic model. Up to now, only the first approach has been explored ([8], [11], [13]) because no satisfactory probabilistic model has yet been proposed for an analytical approach. The simplest model is the well-known Erdös model, where the probability of appearance of an edge between two different vertices is equal to some fixed $p \in (0, 1)$. This model only concerns undirected networks. Its major drawback lies in the fact that the numbers of edges per vertex, so-called vertex degrees, are distributed according to a Binomial distribution, generally approximated by a Poisson distribution, whereas biological networks appear to be scale-free, meaning a power law for the number of edges per vertex [1] (for more details on random graphs, we refer to [4, 6, 5]). Randomized networks (obtained by simulation, see [10] for instance) rely on the knowledge of the number of (incoming and outcoming, when dealing with directed graphs) edges for each vertex. In the same spirit, we provide a probabilistic model that fits these vertex degrees. Depending on the specified sequence of edges per vertex, our model may describe scale-free networks. This probabilistic model enables us to derive exact formulas for the mean and variance of the number of occurrences of a motif, in a graph specified by a sequence of degrees. One of the advantages of this approach is that we do not need computationally expensive simulations of a large number of graphs, for each fixed sequence of numbers of edges per vertex.

Let us mention another approach developped in [3] where "groups of motifs" are detected using an heuristic algorithm based on a probabilistic model. The main difference between this approach and our work lies in the definition of a motif. Berg and Lässig's motifs are groups of vertices which are highly interconnected in a sparse graph, whereas we consider sets of inter-connected vertices with a given topology.

Section 2 presents the definitions of motifs and their occurrences. To decide whether a given motif $\mathbf{m}$ has an unexpected frequency in a given observed graph, one has first to consider random graphs having some similar properties with the observed graph (Section 3), and then to calculate the expected count of $\mathbf{m}$ in such

random graphs, and eventually its variance (Section 4). Since the derivation of the exact distribution of a motif count is still an open problem, its exact mean and variance can be used to calculate a z-score directly. This avoids heavy simulations used in the literature to evaluate the significance of motif counts [9]. Indeed, from our knowledge, current methods to assess significance of motif counts are based on a large number of simulations *for each* type of graph (namely, a fixed sequence of degrees). Our approach is simple to implement and leads to a generic procedure (valid for any type of graph).

2. MOTIFS AND OCCURRENCES

Recall that, in this paper, a motif $\mathbf{m}$ of size k is simply a connected subgraph with k vertices. We will essentially focus on undirected graphs and motifs, but the generalization to a directed framework will be discussed in the conclusion. Therefore, there are only two motifs of size 3 (triangle and "V") and six motifs of size 4 (see Figure 1).

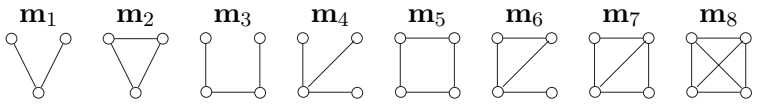

Figure 1: Motifs of size 3 and 4.

Let us fix an undirected graph G with N vertices labelled by $\{1, 2, ..., N\}$. I_k denotes the set of positions $\{i_1, i_2, ..., i_k\}$ in graph G where a motif of size k may occur. Namely, I_k is the set of all subsets of $\{1, 2, ..., N\}$ with cardinality k:

$$I_k = \left\{ \{i_1, i_2, ..., i_k\} \subset \{1, ..., N\}^k \text{ such that } i_j \neq i_\ell, \ \forall 1 \leqslant j \neq \ell \leqslant k \right\}.$$

In the same way, for any subset $J \subset \{1, ..., N\}$, define the sets of positions among the restricted number of vertices $\{1, ..., N\} \setminus J$,

$$I_k(J) = \left\{ \{i_1, i_2, ..., i_k\} \subset \left(\{1, ..., N\} \setminus J\right)^k \text{ such that } i_j \neq i_\ell, \ \forall j \neq \ell \right\}.$$

We say that a given motif $\mathbf{m}$ occurs at position $\alpha = \{i_1, i_2, ..., i_k\} \in I_k$ in G if and only if the sub-graph with vertices $\{i_1, i_2, ..., i_k\}$ in G either has the same topology as $\mathbf{m}$, or contains a subgraph with the same topology as $\mathbf{m}$. For instance, the triangle (motif $\mathbf{m}_2$ from Figure 1) occurs once in the graph in Figure 2 (position $\{2, 3, 4\}$), and the "V" motif ($\mathbf{m}_1$ from Figure 1) occurs 5 times (3 times at position $\{2, 3, 4\}$, once at position $\{1, 2, 3\}$ and once at position $\{1, 2, 4\}$).

To define $N(\mathbf{m})$ the number of occurrences of $\mathbf{m}$ in a graph G, we introduce variables $Y_\alpha(\mathbf{m})$, $\alpha \in I_k$, defined as the number of occurrences of motif $\mathbf{m}$ in the sub-

graph with vertices α. Thus, for any motif of size k, we have $N(\mathbf{m}) = \sum_{\alpha \in I_k} Y_\alpha(\mathbf{m})$. If $\alpha = \{i_1, i_2, ..., i_k\}$, the variable $Y_\alpha(\mathbf{m})$ can be reformulated as $Y_{i_1,i_2,...,i_k}(\mathbf{m})$.

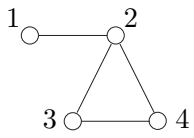

Figure 2: A graph containing 5 occurrences of motif $\mathbf{m}_1$ and 1 occurrence of motif $\mathbf{m}_2$.

3. RANDOM GRAPH MODEL

Undirected graphs are quite properly described by the sequence of the number of edges per node. Let us consider a graph G with N vertices labelled by $\{1, ..., N\}$ and a sequence of integers $(d_1, ..., d_N)$ such that $0 \leqslant d_i \leqslant N-1$. In practice, when analyzing a given graph, d_i is chosen as the observed degree of vertex i. We consider the following probabilistic model for graph G. Random variables Z_{ij} indicating presence/absence of an edge between vertices i and j ($i \neq j$) are independent Bernoulli variables with mean π_{ij} (they are not identically distributed). Moreover, this probability π_{ij} of appearance of an edge between vertices i and j is related to the observed number of edges at node i and the observed number of edges at node j:

$$\pi_{ij} = \pi_{ji} = \frac{d_i d_j}{C} \qquad \text{and} \qquad \pi_{ii} = 0 \ .$$

C is a normalizing constant such that $\pi_{ij} \in [0, 1]$. For instance, $C = \max_{i \neq j} d_i d_j$. If the degrees are not too large with respect to the total number N of vertices, one may use $C_0 = \sum_{j=1}^{N} d_j(d_+ - d_j)/d_+$ with $d_+ = \sum_i d_i$. With such a choice, the expected number of edges is equal to the observed total number of edges. Moreover, the expected number of edges at node i is almost equal to d_i. Note that we do not allow direct loops from an edge to itself ($\pi_{ii} = 0$).

The advantage of this model is that its parameters are easy to choose from an observed graph, contrary to more general π_{ij}'s, and it almost fits the observed sequence of degrees when choosing C_0 as the normalizing constant. It relies on the same idea of preserving the sequence degrees as the commonly used simulation approach [8]. Our probabilistic model appears as a rigorous formalization of the simulation method. [8] suggest generating graphs that preserve the number of occurrences of all $(k-1)$-node sub-graphs when studying motifs of size k. Taking into account the counts of the $(k-1)$-node sub-graphs would be better than only preserving the sequence of degrees but such a generalization appears to be difficult at this stage.

4. FIRST AND SECOND MOMENTS FOR THE COUNT

Motifs of size 1 or 2 are of no interest here because they are the vertices and the edges, respectively, and their frequencies are set by the graph model. Let $\mathbf{m}$ be a motif of size $k \geqslant 3$. Since the variance of $N(\mathbf{m})$ is equal to $\mathbb{E}N^2(\mathbf{m}) - (\mathbb{E}N(\mathbf{m}))^2$, we will calculate the first and second moments of the count, i.e. $\mathbb{E}N(\mathbf{m})$ and $\mathbb{E}N^2(\mathbf{m})$. As we will see, these moments depend on $\mathbf{m}$, both through its size and its topology. No general formula is provided but we propose a general methodology that can be applied to any topological motifs without theoretical difficulties. Because of technical reasons, we will restrict ourselves to motifs of size 3 and 4. More precisely, for each motif $\mathbf{m}$, we provide a simple description of variable $Y_\alpha(\mathbf{m})$ using indicator random variables (RVs). This description enables us to derive explicit formulas for the moments $\mathbb{E}N(\mathbf{m})$ and $\mathbb{E}N^2(\mathbf{m})$. Before detailing the different cases, we state a common framework that will point out the basic quantities to calculate systematically.

Getting the expected count just requires the calculation of $\mathbb{E}Y_\alpha(\mathbf{m})$ for $\alpha \in I_k$ since we have

$$\mathbb{E}N(\mathbf{m}) = \sum_{\alpha \in I_k} \mathbb{E}Y_\alpha(\mathbf{m}) \ .$$

Getting the second moment is a little more involved. By definition,

$$\mathbb{E}N^2(\mathbf{m}) = \mathbb{E}\left(\sum_{\alpha \in I_k} Y_\alpha(\mathbf{m}) \times \sum_{\beta \in I_k} Y_\beta(\mathbf{m}) \right) = \sum_{\alpha \in I_k} \sum_{\beta \in I_k} \mathbb{E}\big(Y_\alpha(\mathbf{m})\, Y_\beta(\mathbf{m})\big) \ .$$

Let us break down the sums over α and β into $(k+1)$ sums depending on the cardinality of the intersection $\alpha \cap \beta$, denoted by $|\alpha \cap \beta|$. Note that

(i) when $|\alpha \cap \beta| = k$, then $\alpha = \beta$ and $\mathbb{E}(Y_\alpha(\mathbf{m})Y_\beta(\mathbf{m})) = \mathbb{E}Y_\alpha^2(\mathbf{m})$,

(ii) when $|\alpha \cap \beta| \leqslant 1$ (disjoint occurrences or a unique vertex in common), then $Y_\alpha(\mathbf{m})$ and $Y_\beta(\mathbf{m})$ are independent random variables, leading to $\mathbb{E}(Y_\alpha(\mathbf{m})\, Y_\beta(\mathbf{m})) = \mathbb{E}Y_\alpha(\mathbf{m})\, \mathbb{E}Y_\beta(\mathbf{m})$.

It gives

$$\begin{aligned} \mathbb{E}N^2(\mathbf{m}) \ = \ & \sum_{|\alpha \cap \beta| = 0} \mathbb{E}\big(Y_\alpha(\mathbf{m})\big)\, \mathbb{E}\big(Y_\beta(\mathbf{m})\big) + \sum_{|\alpha \cap \beta| = 1} \mathbb{E}Y_\alpha(\mathbf{m})\, \mathbb{E}Y_\beta(\mathbf{m}) \\ & + \sum_{2 \leqslant n \leqslant k-1} \ \sum_{|\alpha \cap \beta| = n} \mathbb{E}\big(Y_\alpha(\mathbf{m})\, Y_\beta(\mathbf{m})\big) + \sum_{\alpha \in I_k} \mathbb{E}Y_\alpha^2(\mathbf{m}) \ . \end{aligned} \tag{4.1}$$

Additionally to quantities $\mathbb{E}Y_\alpha(\mathbf{m})$, we have to calculate terms in the form $\mathbb{E}(Y_\alpha(\mathbf{m})\, Y_\beta(\mathbf{m}))$ when α and β share between 2 and k elements. The next two subsections provide explicit formulas for motifs of size 3 and 4. The generic method is to write $Y_\alpha(\mathbf{m})$ as a sum of Bernoulli RVs whose expectations are straightforward to calculate.

4.1. Motifs of size 3

When $k = 3$, Equation (4.1) reduces to

$$
\begin{aligned}
\mathbb{E}N^2(\mathbf{m}) \;=\; & \sum_{\{i,j,k\}\in I_3} \; \sum_{\{\ell,u,v\}\in I_3(ijk)} \mathbb{E}Y_{i,j,k}(\mathbf{m})\,\mathbb{E}Y_{\ell,u,v}(\mathbf{m}) \\
& + \sum_{1\leqslant i\leqslant N} \; \sum_{\{j,k\}\in I_2(i)} \; \sum_{\{\ell,u\}\in I_2(ijk)} \mathbb{E}Y_{i,j,k}(\mathbf{m})\,\mathbb{E}Y_{i,\ell,u}(\mathbf{m}) \\
& + \sum_{\{i,j\}\in I_2} \; \sum_{k\in I_1(ij)} \; \sum_{\ell\in I_1(ijk)} \mathbb{E}\big(Y_{i,j,k}(\mathbf{m})\,Y_{i,j,\ell}(\mathbf{m})\big) \;+\; \sum_{\{i,j,k\}\in I_3} \mathbb{E}Y^2_{i,j,k}(\mathbf{m})\,.
\end{aligned}
\tag{4.2}
$$

Motif $\mathbf{m}_1$ ("V")

Our approach is based on the split of variable $Y_{i,j,k}(\mathbf{m}_1)$ into the sum of three Bernoulli RVs

$$
Y_{i,j,k}(\mathbf{m}_1) = Z_{ij,ik} + Z_{ij,jk} + Z_{ik,jk} \;, \qquad \forall\, i,j,k \in \{1,...,N\} \;,
$$

where $Z_{ij,ik} = 1$ if both edges ij and ik occur, and 0 otherwise. The expectation $\mathbb{E}Z_{ij,ik}$ is the probability $\pi_{ij}\pi_{ik}$. Thus we obtain

$$
\begin{aligned}
\mathbb{E}Y_{i,j,k}(\mathbf{m}_1) \;&=\; \pi_{ij}\pi_{ik} + \pi_{ij}\pi_{jk} + \pi_{ik}\pi_{jk} \;=\; \frac{d_i d_j d_k}{C^2}\,(d_i + d_j + d_k) \;, \\
\mathbb{E}N(\mathbf{m}_1) \;&=\; \sum_{\{i,j,k\}\in I_3} \frac{d_i d_j d_k}{C^2}\,(d_i + d_j + d_k) \;=\; \sum_{1\leqslant i\leqslant N} \; \sum_{\{j,k\}\in I_2(i)} \frac{d_i^2 d_j d_k}{C^2} \;.
\end{aligned}
\tag{4.3}
$$

Similarly, we denote by $Z_{ij,ik,jk}$ the indicator RV of the presence of edges ij, jk and ik (note that $Z_{ij,ik}\,Z_{ij,jk} = Z_{ij,ik,jk}$). To calculate $\mathbb{E}(Y_{i,j,k}(\mathbf{m}_1)\,Y_{i,j,\ell}(\mathbf{m}_1))$, we write

$$
\begin{aligned}
&\mathbb{E}\big(Y_{i,j,k}(\mathbf{m}_1)\,Y_{i,j,\ell}(\mathbf{m}_1)\big) \;= \\
&\qquad =\; \mathbb{E}\Big\{\big[Z_{ij,ik} + Z_{ij,jk} + Z_{ik,jk}\big]\,\big[Z_{ij,i\ell} + Z_{ij,j\ell} + Z_{i\ell,j\ell}\big]\Big\} \\
&\qquad =\; \pi_{ij}(\pi_{ik} + \pi_{jk})\,(\pi_{i\ell} + \pi_{j\ell} + \pi_{i\ell}\pi_{j\ell}) \;+\; \pi_{ik}\pi_{jk}(\pi_{ij}\pi_{i\ell} + \pi_{ij}\pi_{j\ell} + \pi_{i\ell}\pi_{j\ell}) \\
&\qquad =\; \frac{d_i d_j d_k d_\ell}{C^3}(d_i + d_j)^2 \;+\; \frac{d_i^2 d_j^2 d_k d_\ell}{C^4}\Big\{(d_i + d_j)(d_k + d_\ell) + d_k d_\ell\Big\} \;.
\end{aligned}
\tag{4.4}
$$

Now, we focus on the term $\mathbb{E}Y^2_{i,j,k}(\mathbf{m}_1)$. We get

$$\begin{aligned}\mathbb{E}Y^2_{i,j,k}(\mathbf{m}_1) &= \mathbb{E}Z_{ij,ik} + \mathbb{E}Z_{ij,jk} + \mathbb{E}Z_{ik,jk} + 6\,\mathbb{E}Z_{ij,ik,jk} \\ &= \mathbb{E}Y_{i,j,k}(\mathbf{m}_1) + 6\,\pi_{ij}\pi_{ik}\pi_{jk} \\ &= \frac{d_i d_j d_k}{C^2}\,(d_i+d_j+d_k) + 6\,\frac{d_i^2 d_j^2 d_k^2}{C^3}\ .\end{aligned} \tag{4.5}$$

Finally, by using Equations (4.2), (4.3), (4.4) and (4.5), we obtain

$$\begin{aligned}\mathbb{E}N^2(\mathbf{m}_1) &= \\ &= \sum_{\{i,j,k,\ell,u,v\}\in I_6} \frac{d_i d_j d_k d_\ell d_u d_v}{C^4}\,(d_i+d_j+d_k)\,(d_\ell+d_u+d_v) \\ &\quad + \sum_{1\leqslant i\leqslant N}\ \sum_{\{j,k\}\in I_2(i)}\ \sum_{\{\ell,u\}\in I_2(ijk)} \frac{d_i^2 d_j d_k d_\ell d_u}{C^4}\,(d_i+d_j+d_k)\,(d_i+d_\ell+d_u) \\ &\quad + \sum_{\{i,j\}\in I_2}\ \sum_{k\in I_1(ij)}\ \sum_{\ell\in I_1(ijk)} \frac{d_i d_j d_k d_\ell}{C^3}(d_i+d_j)^2 + \frac{d_i^2 d_j^2 d_k d_\ell}{C^4}\big\{(d_i+d_j)(d_k+d_\ell) + d_k d_\ell\big\} \\ &\quad + \sum_{\{i,j,k\}\in I_3} \frac{d_i d_j d_k}{C^2}(d_i+d_j+d_k) + 6\,\frac{d_i^2 d_j^2 d_k^2}{C^3}\ .\end{aligned}$$

Motif $\mathbf{m}_2$ (triangle)

Calculations are simpler for triangles. Motif $\mathbf{m}_2$ occurs at position $\{i,j,k\}$ if and only if the 3 edges ij, jk and ik are present, and $Y_{i,j,k}(\mathbf{m}_2)$ reduces to the indicator RV $Z_{ij,ik,jk}$. Thus we have

$$\mathbb{E}Y_{i,j,k}(\mathbf{m}_2) = \pi_{ij}\pi_{jk}\pi_{ik} = \frac{d_i^2 d_j^2 d_k^2}{C^3}\,; \qquad \mathbb{E}N(\mathbf{m}_2) = \sum_{\{i,j,k\}\in I_3} \frac{d_i^2 d_j^2 d_k^2}{C^3}\ . \tag{4.6}$$

Moreover, the product $Y_{i,j,k}(\mathbf{m}_2)\,Y_{i,j,\ell}(\mathbf{m}_2)$ is equal to the indicator RV $Z_{ij,jk,ik,i\ell,j\ell}$ of presence of the 5 edges ij, jk, ik, $i\ell$ and $j\ell$. Therefore,

$$\mathbb{E}\big(Y_{i,j,k}(\mathbf{m}_2)\,Y_{i,j,\ell}(\mathbf{m}_2)\big) \;=\; \pi_{ij}\pi_{jk}\pi_{ik}\pi_{j\ell}\pi_{i\ell} \;=\; \frac{d_i^3 d_j^3 d_k^2 d_\ell^2}{C^5}\ . \tag{4.7}$$

Since $Y_{i,j,k}(\mathbf{m}_2)$ is an indicator RV, we have $Y^2_{i,j,k}(\mathbf{m}_2) = Y_{i,j,k}(\mathbf{m}_2)$ and $\sum_{\{i,j,k\}\in I_3}\mathbb{E}Y^2_{i,j,k}(\mathbf{m}_2) = \mathbb{E}N(\mathbf{m}_2)$.

By plugging the formulas given by (4.6) and (4.7) in Equation (4.2), we obtain the result.

4.2. Motifs of size 4

When $k = 4$, Equation (4.1) reduces to

$$
\begin{aligned}
\mathbb{E}N^2(\mathbf{m}) \;=\; & \sum_{\{i,j,k,\ell\}\in I_4} \; \sum_{\{u,v,w,x\}\in I_4(ijk\ell)} \mathbb{E}Y_{i,j,k,\ell}(\mathbf{m})\,\mathbb{E}Y_{u,v,w,x}(\mathbf{m}) \\
& + \sum_{1\leqslant i\leqslant N} \; \sum_{\{j,k,\ell\}\in I_3(i)} \; \sum_{\{u,v,w\}\in I_3(ijk\ell)} \mathbb{E}Y_{i,j,k,\ell}(\mathbf{m})\,\mathbb{E}Y_{i,u,v,w}(\mathbf{m}) \\
& + \sum_{\{i,j\}\in I_2} \; \sum_{\{k,\ell\}\in I_2(ij)} \; \sum_{\{u,v\}\in I_2(ijk\ell)} \mathbb{E}\big(Y_{i,j,k,\ell}(\mathbf{m})\,Y_{i,j,u,v}(\mathbf{m})\big) \\
& + \sum_{\{i,j,k\}\in I_3} \; \sum_{\ell\in I_1(ijk)} \; \sum_{u\in I_1(ijk\ell)} \mathbb{E}\big(Y_{i,j,k,\ell}(\mathbf{m})\,Y_{i,j,k,u}(\mathbf{m})\big) \\
& + \sum_{\{i,j,k,\ell\}\in I_4} \mathbb{E}Y^2_{i,j,k,\ell}(\mathbf{m})\;.
\end{aligned}
\tag{4.8}
$$

Following the approach used for motifs of size 3, we detail how to calculate terms in the form $\mathbb{E}Y_{i,j,k,\ell}(\mathbf{m})$, $\mathbb{E}(Y_{i,j,k,\ell}(\mathbf{m})\,Y_{i,j,u,v}(\mathbf{m}))$, $\mathbb{E}(Y_{i,j,k,\ell}(\mathbf{m})\,Y_{i,j,k,u}(\mathbf{m}))$ and $\mathbb{E}Y^2_{i,j,k,\ell}(\mathbf{m})$, but only for motif $\mathbf{m}_4$. However, all final formulas are gathered in Tables 1, 2, 3 and 4. Before, we give the split of variables $Y_\alpha(\mathbf{m}_i)$ for $3\leqslant i\leqslant 8$, as sums of indicator RVs (see Equations (4.9) to (4.14)). These splits directly derive from the topology of the motif under consideration. Combined with Equation (4.8), they are the basis for obtaining the final formulas presented in the tables.

There are 12 different occurrences of motif $\mathbf{m}_3$ at position $\{i,j,k,\ell\}$, which correspond to different orders of the nodes:

$$
\begin{aligned}
Y_{i,j,k,\ell}(\mathbf{m}_3) = {} & Z_{ij,jk,k\ell} + Z_{jk,k\ell,\ell i} + Z_{k\ell,\ell i,ij} + Z_{\ell i,ij,jk} + Z_{ik,k\ell,\ell j} + Z_{ij,j\ell,\ell k} \\
& + Z_{\ell j,ji,ik} + Z_{\ell k,ki,ij} + Z_{i\ell,\ell j,jk} + Z_{\ell i,ik,kj} + Z_{ki,i\ell,\ell j} + Z_{ik,kj,j\ell}\,.
\end{aligned}
\tag{4.9}
$$

Different occurrences of motif $\mathbf{m}_4$ appear depending on the *central* node (bottom left node in Fig. 1, motif $\mathbf{m}_4$):

$$
Y_{i,j,k,\ell}(\mathbf{m}_4) \;=\; Z_{ij,ik,i\ell} + Z_{ji,jk,j\ell} + Z_{ki,kj,k\ell} + Z_{\ell i,\ell j,\ell k}\;. \tag{4.10}
$$

There are only 3 different ways for motif $\mathbf{m}_5$ to occur:

$$
Y_{i,j,k,\ell}(\mathbf{m}_5) \;=\; Z_{ij,jk,k\ell,\ell i} + Z_{ij,j\ell,\ell k,ki} + Z_{ik,kj,j\ell,\ell i}\;. \tag{4.11}
$$

Occurrences of motif $\mathbf{m}_6$ are obtained through occurrences of motif $\mathbf{m}_4$. When motif $\mathbf{m}_4$ occurs, there are 3 different ways of adding a vertex in order to obtain motif $\mathbf{m}_6$. This leads to a total of 12 different possible occurrences of motif $\mathbf{m}_6$ at $\{i,j,k,\ell\}$:

$$
\begin{aligned}
Y_{i,j,k,\ell}(\mathbf{m}_6) \;=\; & Z_{ij,ik,i\ell,jk} + Z_{ij,ik,i\ell,j\ell} + Z_{ij,ik,i\ell,k\ell} + Z_{ji,jk,j\ell,ik} \\
& + Z_{ji,jk,j\ell,k\ell} + Z_{ji,jk,j\ell,i\ell} + Z_{ki,kj,k\ell,ij} + Z_{ki,kj,k\ell,i\ell} \\
& + Z_{ki,kj,k\ell,j\ell} + Z_{\ell i,\ell j,\ell k,ij} + Z_{\ell i,\ell j,\ell k,ik} + Z_{\ell i,\ell j,\ell k,jk}\;.
\end{aligned}
\tag{4.12}
$$

Motif $\mathbf{m}_7$ is obtained from motif $\mathbf{m}_5$ by adding a diagonal:

$$Y_{i,j,k,\ell}(\mathbf{m}_7) = Z_{ij,jk,k\ell,\ell i,j\ell} + Z_{ij,jk,k\ell,\ell i,ik} + Z_{ij,j\ell,\ell k,ki,jk} + Z_{ij,j\ell,\ell k,ki,i\ell} + Z_{ik,kj,j\ell,\ell i,ij} + Z_{ik,kj,j\ell,\ell i,k\ell} \,. \tag{4.13}$$

Finally, motif $\mathbf{m}_8$ corresponds to a complete sub-graph on vertices $\{i,j,k,\ell\}$ and is thus equal to an indicator RV:

$$Y_{i,j,k,\ell}(\mathbf{m}_8) = Z_{ij,jk,k\ell,i\ell,ik,j\ell} \,. \tag{4.14}$$

Detailed calculations for motif $\mathbf{m}_4$ (star)

Let us start by calculating the expectation $\mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_4)$. We use Equation (4.10) and the fact that $\mathbb{E}Z_{ij,ik,i\ell}$ equals $\pi_{ij}\pi_{ik}\pi_{i\ell} = d_i^3 d_j d_k d_\ell / C^3$. Thus,

$$\mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_4) = \frac{d_i d_j d_k d_\ell}{C^3}\,(d_i^2 + d_j^2 + d_k^2 + d_\ell^2) \tag{4.15}$$

$$\text{and} \qquad \mathbb{E}N(\mathbf{m}_4) = \sum_{1 \leqslant i \leqslant N} \sum_{\{j,k,\ell\} \in I_3(i)} \frac{d_i^3 d_j d_k d_\ell}{C^3} \,. \tag{4.16}$$

We now calculate $\mathbb{E}(Y_{i,j,k,\ell}(\mathbf{m}_4)\, Y_{i,j,u,v}(\mathbf{m}_4))$ by using the product of the sums of indicator RVs:

$$\begin{aligned}
\mathbb{E}\big(Y_{i,j,k,\ell}(\mathbf{m}_4)\, Y_{i,j,u,v}(\mathbf{m}_4)\big) &= \\
&= \pi_{ij}\,(\pi_{ik}\pi_{i\ell} + \pi_{jk}\pi_{j\ell})\,(\pi_{iu}\pi_{iv} + \pi_{ju}\pi_{jv} + \pi_{iu}\pi_{ju}\pi_{uv} + \pi_{iv}\pi_{jv}\pi_{uv}) \\
&\quad + \pi_{k\ell}\,(\pi_{ik}\pi_{jk} + \pi_{i\ell}\pi_{j\ell})\,(\pi_{ij}\pi_{iu}\pi_{iv} + \pi_{ij}\pi_{ju}\pi_{jv} + \pi_{iu}\pi_{ju}\pi_{uv} + \pi_{iv}\pi_{jv}\pi_{uv}) \\
&= \frac{d_i d_j d_k d_\ell d_u d_v}{C^5} \\
&\quad \times \left\{ (d_i^2 + d_j^2)^2 + \frac{d_i d_j}{C}\Big((d_k^2 + d_\ell^2)\,(d_u^2 + d_v^2) + (d_i^2 + d_j^2)\,(d_u^2 + d_v^2 + d_k^2 + d_\ell^2) \Big) \right\} .
\end{aligned} \tag{4.17}$$

In the same way, we have

$$\begin{aligned}
\mathbb{E}\big(Y_{i,j,k,\ell}(\mathbf{m}_4)\, Y_{i,j,k,u}(\mathbf{m}_4)\big) &= \frac{d_i d_j d_k d_\ell d_u}{C^4} \left\{ d_i^2 \left(d_i + \frac{d_j^2 d_k}{C} + \frac{d_j d_k^2}{C} \right) \right. \\
&\quad + d_j^2 \left(\frac{d_i^2 d_k}{C} + d_j + \frac{d_i d_k^2}{C} \right) + d_k^2 \left(\frac{d_i^2 d_j}{C} + \frac{d_i d_j^2}{C} + d_k \right) \Big\} \\
&\quad + \frac{d_i^2 d_j^2 d_k^2 d_\ell d_u}{C^6} \left\{ (d_i^2 + d_j^2 + d_k^2)\,(d_u^2 + d_\ell^2) + d_\ell^2 d_u^2 \right\} .
\end{aligned} \tag{4.18}$$

We finally compute expectation $\mathbb{E}Y^2_{i,j,k,\ell}(\mathbf{m}_4)$:

$$\begin{aligned}\mathbb{E}Y^2_{i,j,k,\ell}(\mathbf{m}_4) \;&=\; \mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_4)\\ &\quad + 2\,\frac{d_i^2 d_j^2 d_k^2 d_\ell^2}{C^5}\left(d_i d_j + d_i d_k + d_i d_\ell + d_j d_k + d_j d_\ell + d_k d_\ell\right)\,, \qquad (4.19)\end{aligned}$$

$$\sum_{\{i,j,k,\ell\}\in I_4} \mathbb{E}Y^2_{i,j,k,\ell}(\mathbf{m}_4) \;=\; \mathbb{E}N(\mathbf{m}_4) \;+\; 2\sum_{\{i,j\}\in I_2}\;\sum_{\{k,\ell\}\in I_2(ij)} \frac{d_i^3 d_j^3 d_k^2 d_\ell^2}{C^5}\,.$$

Finally, the second moment $\mathbb{E}N^2(\mathbf{m}_4)$ is obtained by plugging the expressions given by (4.15), (4.16), (4.17), (4.18) and (4.19) in Equation (4.8).

5. CONCLUSION

We provide a rigorous probabilistic model for undirected graphs which fits the vertex degrees of an observed graph and thus partially describes real-world networks. This model allows us to derive explicit formulas for the mean and variance of the number of occurrences of the 2 motifs of length 3 and the 6 motifs of length 4. Here, a motif is a simple pattern of interconnexion in a graph. Our methodology can be extended to longer motifs through straightforward calculations. Indeed, one just needs to describe the motif as a sum of indicator variables of Z-type (see decomposition (4.9)–(4.14) for instance). Then the second moment $\mathbb{E}N^2(\mathbf{m})$ given in equation (4.1) reduces to sums of products of expectations of independent Binomial random variables (the Z_{ij}'s for single edges (ij)), easy to compute. Heavy simulations are usually done so far to study over-representation of motifs. Thus, our formulas are of great interest in practice.

We think that no general formula depending only on the total numbers of edges and vertices of the motif exists; additional topological information on the motif is required ($\mathbf{m}_3$ and $\mathbf{m}_4$ both have 4 vertices and 3 edges, but they clearly have different expected counts).

Our methodology can also be generalized to directed motifs and directed graphs. This is an important issue when analyzing biological networks where the orientation of the edges may be known (direction of a reaction in metabolic networks or activation/regulation in gene interaction networks). This will be the matter of a forthcoming paper. Briefly, the probability π_{ij} that an edge goes from i toward j is proportional to the product $\epsilon_i\,\rho_j$ where ϵ_i is chosen as the observed outcoming degree of vertex i and ρ_j is chosen as the observed incoming degree of vertex j. Therefore, this model fits to the incoming and outcoming vertex degrees. Note that this expression for π_{ij} has already been considered by [3] as part of a more general model to detect groups of highly inter-connected vertices which share some similarity.

Finally, one may be interested in counting exact occurrences of a motif $\mathbf{m}$ in graph G. For instance, no "V" motif is counted in a triangle. Our results can be easily extended by defining new indicator RV $X_{i_1,\ldots,i_k}(\mathbf{m})$ which is equal to 1 if the sub-graph with vertices $\{i_1,\ldots,i_k\}$ has exactly the same topology as $\mathbf{m}$ and 0 otherwise. We then write $X_{i_1,\ldots,i_k}(\mathbf{m})$ as a linear combination of ad-hoc edge indicators Z. For instance, if $\mathbf{m}$ is the "V"motif, we just write $X_{i,j,k}(\mathbf{m}_1) = Z_{ij,ik}(1-Z_{jk}) + Z_{ij,jk}(1-Z_{ik}) + Z_{ik,jk}(1-Z_{ij})$.

Table 1: Mean count $\mathbb{E}N(\mathbf{m})$ for non oriented motifs of size 4.

$\mathbf{m}$	$\mathbb{E}N(\mathbf{m})$
$\mathbf{m}_3$ 	$\mathbb{E}N(\mathbf{m}_3) = 2\,C^{-3} \sum_{\{i,j\}\in I_2} \sum_{\{k,\ell\}\in I_2(ij)} d_i^2 d_j^2 d_k d_\ell$
$\mathbf{m}_4$ 	$\mathbb{E}N(\mathbf{m}_4) = C^{-3} \sum_{i=1}^{N} \sum_{\{j,k,\ell\}\in I_3(i)} d_i^3 d_j d_k d_\ell$
$\mathbf{m}_5$ 	$\mathbb{E}N(\mathbf{m}_5) = 3\,C^{-4} \sum_{\{i,j,k,\ell\}\in I_4} d_i^2 d_j^2 d_k^2 d_\ell^2$
$\mathbf{m}_6$ 	$\mathbb{E}N(\mathbf{m}_6) = C^{-4} \sum_{1\leqslant i\leqslant N} \sum_{j\neq i} \sum_{\{k,\ell\}\in I_2(ij)} d_i^3 d_j d_k^2 d_\ell^2$
$\mathbf{m}_7$ 	$\mathbb{E}N(\mathbf{m}_7) = C^{-5} \sum_{\{i,j\}\in I_2} \sum_{\{k,\ell\}\in I_2(ij)} d_i^3 d_j^3 d_k^2 d_\ell^2$
$\mathbf{m}_8$ 	$\mathbb{E}N(\mathbf{m}_8) = C^{-6} \sum_{\{i,j,k,\ell\}\in I_4} d_i^3 d_j^3 d_k^3 d_\ell^3$

Table 2: Formulas giving $\mathbb{E}(Y_{ijk\ell}(\mathbf{m})\, Y_{ijuv}(\mathbf{m}))$ for non oriented motifs of size 4.

$\mathbf{m}$	$\mathbb{E}\big(Y_{ijk\ell}(\mathbf{m})\, Y_{ijuv}(\mathbf{m})\big)$
$\mathbf{m}_3$ 	$C^{-5} d_i d_j d_k d_\ell d_u d_v \Big[(d_i+d_j)^2\,(d_k+d_\ell)\,(d_u+d_v)\,\{1+3\,C^{-1} d_i d_j\}$ $+\, 2\, d_i d_j (d_i+d_j) \Big\{ (d_u+d_v)\,\big(2+C^{-1}(d_i d_j+2\, d_k d_\ell)\big)$ $+\, (d_k+d_\ell)\,\big(2+C^{-1}(d_i d_j+2\, d_u d_v)\big) \Big\}$ $+\, 4\, d_i^2 d_j^2\,\big(1+C^{-1}(d_u d_v+d_k d_\ell)\big)+4\, C^{-1} d_i d_j d_u d_v d_k d_\ell \Big]$
$\mathbf{m}_4$ 	see formula (4.17)
$\mathbf{m}_5$ 	$4\, C^{-7} d_i^3 d_j^3 d_k^2 d_\ell^2 d_u^2 d_v^2 + 5\, C^{-8} d_i^4 d_j^4 d_k^2 d_\ell^2 d_u^2 d_v^2$
$\mathbf{m}_6$ 	$\frac{d_i d_j d_k d_\ell d_u d_v}{C^7} \Big[d_i^2 d_j^2 (d_i+d_j)^2\,(d_k+d_\ell)\,(d_u+d_v)\,\Big(1+\frac{d_k d_\ell}{C}+\frac{d_u d_v}{C}\Big)$ $+\, d_i^2 d_j^2 (d_i+d_j) \Big\{ (d_k^2+d_\ell^2)\,(d_u+d_v)\,\Big(1+\frac{d_u d_v}{C}\Big)$ $+\, (d_u^2+d_v^2)\,(d_k+d_\ell)\,\Big(1+\frac{d_k d_\ell}{C}\Big) \Big\}$ $+\, d_i d_j (d_i+d_j)\,(d_i^2+d_j^2) \Big\{ d_k d_\ell (d_u+d_v)\,\Big(1+\frac{d_u d_v}{C}\Big)$ $+\, d_u d_v (d_k+d_\ell)\,\Big(1+\frac{d_k d_\ell}{C}\Big) \Big\}$ $+\, d_i d_j (d_i^2+d_j^2)\,\{d_k d_\ell (d_u^2+d_v^2)+d_u d_v (d_k^2+d_\ell^2)\}+d_i^2 d_j^2 (d_k^2+d_\ell^2)(d_u^2+d_v^2)$ $+\, (d_i^2+d_j^2)^2 d_k d_\ell d_u d_v + d_i d_j (d_i+d_j)^2 d_k d_\ell d_u d_v (d_k+d_\ell)(d_u+d_v)\,/\,C \Big]$
$\mathbf{m}_7$ 	$C^{-7} d_i^3 d_j^3 d_k^2 d_\ell^2 d_u^2 d_v^2 \Big\{ C^{-2} (d_i+d_j)^2\,(d_u+d_v)\,(d_k+d_\ell)$ $+\, C^{-2} d_i d_j (d_i+d_j) \Big[\Big(1+\frac{d_u d_v}{C}\Big)(d_k+d_\ell)+\Big(1+\frac{d_k d_\ell}{C}\Big)(d_u+d_v) \Big]$ $+\, C^{-3} d_i^2 d_j^2 (d_u d_v+d_k d_\ell)+C^{-3} d_i d_j d_k d_\ell d_u d_v \Big\}$
$\mathbf{m}_8$ 	$C^{-11} d_i^5 d_j^5 d_k^3 d_\ell^3 d_u^3 d_v^3$

Table 3: Formulas giving $\mathbb{E}(Y_{ijk\ell}(\mathbf{m})\, Y_{ijku}(\mathbf{m}))$ for non oriented motifs of size 4.

$\mathbf{m}$	$\mathbb{E}\big(Y_{ijk\ell}(\mathbf{m})\, Y_{ijku}(\mathbf{m})\big)$
$\mathbf{m}_3$ 	$\frac{d_i d_j d_k d_\ell d_u}{C^4}\Big[6\, d_i d_j d_k \Big\{1 + (d_i+d_j+d_k)\,(d_\ell+d_u)\,/\,C + (d_i d_j+d_i d_k+d_j d_k+d_\ell d_u)\,/\,C + \frac{d_i d_j d_k}{C^2}(d_\ell+d_u)\Big\} + (d_i^2 d_j+d_i d_j^2+d_i^2 d_k+d_i d_k^2+d_j d_k^2+d_j^2 d_k)\Big(1+\frac{d_\ell d_u}{C}+\frac{d_i d_j d_k}{C^2}(d_\ell+d_u)\Big) + 2\,\frac{d_i^2 d_j^2+d_i^2 d_k^2+d_j^2 d_k^2}{C}\,(d_u+d_\ell) + 2(d_i^2+d_j^2+d_k^2)\,\frac{d_i d_j d_k}{C}\Big(1+\frac{d_\ell d_u}{C}\Big) + 6\, d_i d_j d_k d_\ell d_u / C^2 (d_i d_k+d_i d_j+d_j d_k)\Big]$
$\mathbf{m}_4$ 	see formula (4.18)
$\mathbf{m}_5$ 	$C^{-6} d_i^2 d_j^2 d_k^2 d_\ell^2 d_u^2 \Big\{d_i d_j + d_i d_k + d_j d_k + 2\, C^{-1} d_i d_j d_k (d_i+d_j+d_k)\Big\}$
$\mathbf{m}_6$ 	$\frac{d_i^2 d_j^2 d_k^2 d_\ell d_u}{C^5}\Big[(d_i+d_j+d_k)^2 + 2\,\frac{d_u d_\ell}{C^2}\,(d_i^2 d_j^2+d_i^2 d_k^2+d_j^2 d_k^2) + 2(d_i+d_j+d_k)\,(d_i d_j+d_i d_k+d_j d_k)\,\frac{(d_u+d_\ell)}{C} + 2\,\frac{d_u d_\ell}{C^2}\,(d_u+d_\ell)\Big\{d_i^2(d_j+d_k)+d_j^2(d_i+d_k)+d_k^2(d_i+d_j)\Big\}\Big] + \frac{d_i^3 d_j^3 d_k^3 d_\ell d_u}{C^7}\Big[3(d_i+d_j+d_k)\,(d_u+d_\ell)^2 + 2\,\frac{d_\ell d_u}{C}\,(d_u+d_\ell)\,(d_i d_j+d_j d_k+d_i d_k)\Big] + \frac{d_i d_j d_k d_\ell^2 d_u^2}{C^6}\Big[d_i^3(d_j+d_k)^2 + d_j^3(d_i+d_k)^2 + d_k^3(d_i+d_j)^2\Big]$
$\mathbf{m}_7$ 	$C^{-6} d_i^2 d_j^2 d_k^2 d_\ell^2 d_u^2 \Big\{3C^{-2}(d_i d_j+d_i d_k+d_j d_k)\, d_i d_j d_k\,(d_\ell+d_u) + C^{-2}(d_i+d_j+d_k)\, d_i d_j d_k d_u d_\ell + 6\, C^{-3} d_i^2 d_j^2 d_k^2 d_\ell d_u + C^{-1}(d_i d_j+d_i d_k+d_j d_k)^2\Big\}$
$\mathbf{m}_8$ 	$C^{-9} d_i^4 d_j^4 d_k^4 d_\ell^3 d_u^3$

Table 4: Formulas giving $\mathbb{E}Y^2_{ijk\ell}(\mathbf{m})$ for non oriented motifs of size 4.

$\mathbf{m}$	$\mathbb{E}Y^2_{ijk\ell}(\mathbf{m})$
$\mathbf{m}_3$ 	$\mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_3) + C^{-4} d_i^2 d_j^2 d_k^2 d_\ell^2 \Big[12\,(3 + C^{-2} d_i d_j d_k d_\ell)$ $+ 10\, C^{-1} \big(d_i d_j + d_i d_k + d_i d_\ell + d_j d_k + d_j d_\ell + d_k d_\ell \big) \Big]$ $+ 2\, d_i d_j d_k d_\ell C^{-4} \Big\{ d_i d_j d_k (d_i + d_j + d_k) + d_i d_j d_\ell (d_i + d_j + d_\ell)$ $+ d_i d_k d_\ell (d_i + d_k + d_\ell) + d_j d_k d_\ell (d_j + d_k + d_\ell) \Big\}$
$\mathbf{m}_4$ 	see formula (4.19)
$\mathbf{m}_5$ 	$\mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_5) + 6\, C^{-6} d_i^3 d_j^3 d_k^3 d_\ell^3$
$\mathbf{m}_6$ 	$\mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_6) + 12\, C^{-5} d_i^2 d_j^2 d_k^2 d_\ell^2 \Big\{ 5\, C^{-1} d_i d_j d_k d_\ell$ $+ d_i d_j + d_i d_k + d_i d_\ell + d_j d_k + d_j d_\ell + d_k d_\ell \Big\}$
$\mathbf{m}_7$ 	$\mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_7) + 30\, \mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_8)$
$\mathbf{m}_8$ 	$C^{-6} d_i^3 d_j^3 d_k^3 d_\ell^3$

ACKNOWLEDGMENTS

This work has been supported by the French Action Concertée Incitative *Nouvelles Interfaces des Mathématiques*.

REFERENCES

[1] Barabási, A.-L. and Bonabeau, E. (2003). Scale-free networks, *Scientific American*, 50–59.

[2] Barbour, A.D. and Reinert, G. (2001). Small worlds, *Random Struct. Alg.*, **19**, 54–74.

[3] Berg, J. and Lässig (2004). Local graph alignment and motif search in biological networks, *PNAS*, **101**, 14689–14694.

[4] Bollobas, B. (2001). *Random Graphs*, Cambridge University Press.

[5] Durrett, R. (2006). *Random Graph Dynamics*, Cambridge University Press.

[6] Janson, S.; Rucinski, A. and Luczak, T. (2000). *Random Graphs*, Wiley.

[7] Koyutürk, M.; Grama, A. and Szpankowski, W. (2004). An efficient algorithm for detecting frequent subgraphs in biological networks, *Bioinformatics*, **20**, i200–i207.

[8] Milo, R.; Shen-Orr, S.; Itzkovitz, S.; Kashtan, N.; Chklovskii, D. and Alon, U. (2002). Networks motifs: simple building blocks of complex networks, *Science*, **298**, 824–827.

[9] Milo, R.; Itzkovitz, S.; Kashtan, N.; Levitt, R.; Shen-Orr, S.; Ayzenshtat, I.; Sheffer, M. and Alon, U. (2004). Superfamilies of evolved and designed networks, *Science*, **303**, 1538–1542.

[10] Newman, M.E.J.; Strogatz, S.H. and Watts, D.J. (2001). Random graphs with arbitrary degree distributions and their applications, *Phys. Rev. E*, **64**, 026118.

[11] Shen-Orr, S.; Milo, R.; Mangan, S. and Alon, U. (2002). Network motifs in the transcriptional regulation network of Eschericha Coli, *Nature Genetics*, **31**, 64–68.

[12] Stumpf, M.P.; Wiuf, C. and May, R.M. (2005). Subnets of scale-free networks are not scale-free: Sampling properties of networks, *PNAS*, **102**, 4221–4224.

[13] Zhang, L.; King, O.; Wong, S.; Goldberg, D.; Tong, A.; Lesage, G.; Andrews, B.; Bussey, H.; Boone, C. and Roth, F. (2005). Motifs, themes and thematic maps of an integrated *Saccharomyces cerevisiae* interaction network, *Journal of Biology*, **4**, 6.

APPENDIX

This appendix contains some indications in order to prove the formulas put in Tables 1, 2, 3 and 4 for the motifs $\mathbf{m}_3$, $\mathbf{m}_5$, $\mathbf{m}_6$, $\mathbf{m}_7$ and $\mathbf{m}_8$ of length 4.

Motif $\mathbf{m}_3$

We use the split of $Y_{i,j,k,\ell}(\mathbf{m}_3)$ into the sum of 12 different terms. Some symmetrical terms appear and we obtain

$$\mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_3) \;=\; 2\,\frac{d_i d_j d_k d_\ell}{C^3}\left(d_i d_j + d_i d_k + d_i d_\ell + d_j d_k + d_j d_\ell + d_k d_\ell\right)$$

$$\text{and} \qquad \mathbb{E}N(\mathbf{m}_3) \;=\; 2\sum_{\{i,j\}\in I_2}\;\sum_{\{k,\ell\}\in I_2(i,j)}\frac{d_i^2 d_j^2 d_k d_\ell}{C^3}\;.$$

Let us now compute $\mathbb{E}(Y_{i,j,k,\ell}(\mathbf{m}_3)Y_{i,j,u,v}(\mathbf{m}_3))$. This is a big product but a large number of terms may be grouped together and we have

$$\begin{aligned}
\mathbb{E}\big(Y_{i,j,k,\ell}(\mathbf{m}_3)\,Y_{i,j,u,v}(\mathbf{m}_3)\big) \;&=\\
&= \frac{d_i d_j d_k d_\ell d_u d_v}{C^5}\left\{(d_i+d_j)\,(d_u+d_v)\left(1+\frac{d_i d_j}{C}\right)+2\,d_i d_j\left(1+\frac{d_u d_v}{C}\right)\right\}\\
&\quad\times\left\{(d_i+d_j)(d_k+d_\ell)+2\,d_i d_j\right\}\\
&\quad+\,2\,\frac{d_i^2 d_j^2 d_k d_\ell d_u d_v}{C^6}\left\{(d_i+d_j)\,(d_u+d_v)+d_u d_v+d_i d_j\right\}\\
&\quad\times\left\{(d_i+d_j)(d_k+d_\ell)+2\,d_k d_\ell\right\}\;.
\end{aligned}$$

After some simplifications, we obtain,

$$\begin{aligned}
\mathbb{E}\big(Y_{i,j,k,\ell}(\mathbf{m}_3)\,Y_{i,j,u,v}(\mathbf{m}_3)\big) \;&=\\
&= \frac{d_i d_j d_k d_\ell d_u d_v}{C^5}\left[(d_i+d_j)^2\,(d_k+d_\ell)\,(d_u+d_v)\left\{1+3\frac{d_i d_j}{C}\right\}+2d_i d_j(d_i+d_j)\right.\\
&\quad\times\left\{(d_u+d_v)\left(2+\frac{d_i d_j}{C}+2\,\frac{d_k d_\ell}{C}\right)+(d_k+d_\ell)\left(2+\frac{d_i d_j}{C}+2\,\frac{d_u d_v}{C}\right)\right\}\\
&\quad\left.+\,4\,d_i^2 d_j^2\left(1+\frac{d_u d_v}{C}+\frac{d_k d_\ell}{C}\right)+4\,\frac{d_i d_j d_u d_v d_k d_\ell}{C}\right].
\end{aligned}$$

To get $\mathbb{E}Y^2_{i,j,k,\ell}(\mathbf{m}_3)$, we write

$$\begin{aligned}
Y^2_{i,j,k,\ell}(\mathbf{m}_3) \;=\; & Y_{i,j,k,\ell}(\mathbf{m}_3) \\
& + 2\Big\{ 6\, Z_{ij,jk,k\ell,\ell i} + 6\, Y_{i,j,k,\ell}(\mathbf{m}_8) + 6\, Z_{ij,j\ell,\ell k,ki} + 6\, Z_{i\ell,\ell j,jk,ki} \\
& + 5\, Z_{ik,i\ell,jk,j\ell,k\ell} + 5\, Z_{ij,i\ell,jk,j\ell,k\ell} + 5\, Z_{ij,ik,jk,j\ell,k\ell} + 5\, Z_{ij,ik,i\ell,j\ell,k\ell} \\
& + 5\, Z_{ij,ik,i\ell,jk,k\ell} + 5\, Z_{ij,ik,i\ell,jk,j\ell} + Z_{i\ell,jk,j\ell,k\ell} + Z_{ik,i\ell,jk,k\ell} \\
& + Z_{ij,i\ell,jk,j\ell} + Z_{ij,ik,i\ell,jk} + Z_{ij,ik,i\ell,k\ell} + Z_{ij,ik,jk,k\ell} + Z_{ij,i\ell,j\ell,k\ell} \\
& + Z_{ij,ik,jk,k\ell} + Z_{ik,i\ell,j\ell,k\ell} + Z_{ik,jk,j\ell,k\ell} + Z_{ij,ik,i\ell,j\ell} + Z_{ij,ik,jk,j\ell} \Big\} \,.
\end{aligned}$$

This leads to

$$\begin{aligned}
\mathbb{E}Y^2_{i,j,k,\ell}(\mathbf{m}_3) \;=\; & \mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_3) \\
& + \frac{d_i^2 d_j^2 d_k^2 d_\ell^2}{C^4} \Big[12 \Big(3 + \frac{d_i d_j d_k d_\ell}{C^2} \Big) \\
& + \frac{10}{C} \left(d_i d_j + d_i d_k + d_i d_\ell + d_j d_k + d_j d_\ell + d_k d_\ell \right) \Big] \\
& + 2\, \frac{d_i d_j d_k d_\ell}{C^4} \Big\{ d_i d_j d_k (d_i + d_j + d_k) + d_i d_j d_\ell (d_i + d_j + d_\ell) \\
& + d_i d_k d_\ell (d_i + d_k + d_\ell) + d_j d_k d_\ell (d_j + d_k + d_\ell) \Big\} \,.
\end{aligned}$$

Motif $\mathbf{m}_5$ (square)

First, let us calculate the probability $\mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_5)$ that the motif $\mathbf{m}_5$ occurs at position $\{i,j,k,\ell\}$. Write $Y_{i,j,k,\ell}(\mathbf{m}_5) = Z_{ij,jk,k\ell,\ell i} + Z_{ij,j\ell,\ell k,ki} + Z_{ik,kj,j\ell,\ell i}$. Each one of these indicator RVs has same expectation equal to $d_i^2 d_j^2 d_k^2 d_\ell^2 / C^4$. Therefore, we have

$$\mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_5) = 3\, \frac{d_i^2 d_j^2 d_k^2 d_\ell^2}{C^4} \qquad \text{and} \qquad \mathbb{E}N(\mathbf{m}_5) = 3 \sum_{\{i,j,k,\ell\} \in I_4} \frac{d_i^2 d_j^2 d_k^2 d_\ell^2}{C^4} \,.$$

We now calculate $\mathbb{E}(Y_{i,j,k,\ell}(\mathbf{m}_5)\, Y_{i,j,u,v}(\mathbf{m}_5))$ like $\mathbb{E}\{(Z_{ij,jk,k\ell,\ell i} + Z_{ij,j\ell,\ell k,ki} + Z_{ik,kj,j\ell,\ell i})\, (Z_{ij,ju,uv,vi} + Z_{ij,jv,vu,ui} + Z_{iu,uj,jv,vi})\}$. We get

$$\mathbb{E}\big(Y_{i,j,k,\ell}(\mathbf{m}_5)\, Y_{i,j,u,v}(\mathbf{m}_5)\big) \;=\; 4\, \frac{d_i^3 d_j^3 d_k^2 d_\ell^2 d_u^2 d_v^2}{C^7} + 5\, \frac{d_i^4 d_j^4 d_k^2 d_\ell^2 d_u^2 d_v^2}{C^8} \,. \tag{5.1}$$

Now we provide the calculation of $\mathbb{E}(Y_{i,j,k,\ell}(\mathbf{m}_5)\, Y_{i,j,k,u}(\mathbf{m}_5))$.

$$\begin{aligned}\mathbb{E}\big(Y_{i,j,k,\ell}(\mathbf{m}_5)\, Y_{i,j,k,u}(\mathbf{m}_5)\big) \;&=\\ &= \pi_{ij}\pi_{jk}\pi_{k\ell}\pi_{i\ell}\left\{\pi_{ku}\pi_{iu}+\pi_{ju}\pi_{ku}\pi_{ik}+\pi_{ik}\pi_{ju}\pi_{iu}\right\}\\ &\quad+\pi_{ij}\pi_{j\ell}\pi_{k\ell}\pi_{ik}\left\{\pi_{jk}\pi_{ku}\pi_{iu}+\pi_{ju}\pi_{ku}+\pi_{jk}\pi_{ju}\pi_{iu}\right\}\\ &\quad+\pi_{ik}\pi_{jk}\pi_{j\ell}\pi_{i\ell}\left\{\pi_{ij}\pi_{ku}\pi_{iu}+\pi_{ij}\pi_{ju}\pi_{ku}+\pi_{ju}\pi_{iu}\right\}\\ &= \frac{d_i^2d_j^2d_k^2d_\ell^2d_u^2}{C^6}\left\{d_id_j+d_id_k+d_jd_k+2\,\frac{d_id_jd_k}{C}(d_i+d_j+d_k)\right\}.\end{aligned} \tag{5.2}$$

Easy computation of $\mathbb{E}Y^2_{i,j,k,\ell}(\mathbf{m}_5)$ is allowed since all 3 products of two different indicator RVs appearing in $Y_{i,j,k,\ell}(\mathbf{m}_5)$ are equal to $Z_{ij,jk,k\ell,\ell i,j\ell,ik}$ (indicator RV of the complete graph with vertices $\{i,j,k,\ell\}$), whose expectation equals $d_i^3d_j^3d_k^3d_\ell^3/C^6$

$$\mathbb{E}Y^2_{i,j,k,\ell}(\mathbf{m}_5) \;=\; \mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_5)+6\,\frac{d_i^3d_j^3d_k^3d_\ell^3}{C^6}\;. \tag{5.3}$$

Motif $\mathbf{m}_6$

According to the split of $Y_{i,j,k,\ell}(\mathbf{m}_6)$ into the sum of 12 terms with symetrical expectations in the form $d_i^3d_jd_k^2d_\ell^2/C^4$, we have,

$$\sum_{\{i,j,k,\ell\}\in I_4}\mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_6) \;=\; C^{-4}\sum_{i=1}^{N}\sum_{\{i,j\}\in I_2}\sum_{\{k,\ell\}\in I_2(ij)} d_i^3d_jd_k^2d_\ell^2\;.$$

Concerning $\mathbb{E}Y^2_{i,j,k,\ell}(\mathbf{m}_6)$, we have

$$\begin{aligned}Y^2_{i,j,k,\ell}(\mathbf{m}_6) \;=\; Y_{i,j,k,\ell}(\mathbf{m}_6)+2\Big\{&30\,Y_{ijk\ell}(\mathbf{m}_8)+6\Big(Z_{ik,i\ell,jk,j\ell,k\ell}+Z_{ij,i\ell,jk,j\ell,k\ell}\\ &+Z_{ij,ik,jk,j\ell,k\ell}+Z_{ij,ik,i\ell,j\ell,k\ell}+Z_{ij,ik,i\ell,jk,k\ell}+Z_{ij,ik,i\ell,jk,j\ell}\Big)\Big\}\;.\end{aligned}$$

Finally,

$$\begin{aligned}\mathbb{E}Y^2_{i,j,k,\ell}(\mathbf{m}_6) \;=\; \mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_6)+12\,\frac{d_i^2d_j^2d_k^2d_\ell^2}{C^5}\Big\{&5\,\frac{d_id_jd_kd_\ell}{C}\\ &+d_id_j+d_id_k+d_id_\ell+d_jd_k+d_jd_\ell+d_kd_\ell\Big\}\;.\end{aligned}$$

Motif $\mathbf{m}_7$

Using the split $Y_{i,j,k,\ell}(\mathbf{m}_7) = Z_{ij,ik,i\ell,jk,j\ell} + Z_{ij,ik,i\ell,kj,k\ell} + Z_{ij,ik,i\ell,\ell j,\ell k} + Z_{ji,jk,j\ell,ik,i\ell}$, we obtain

$$\mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_7) = \frac{d_i^2 d_j^2 d_k^2 d_\ell^2}{C^5}\left(d_i d_j + d_i d_k + d_i d_\ell + d_j d_k + d_j d_\ell + d_k d_\ell\right),$$

$$\mathbb{E}N(\mathbf{m}_7) = \sum_{\{i,j\}\in I_2}\ \sum_{\{k,\ell\}\in I_2(ij)} \frac{d_i^3 d_j^3 d_k^2 d_\ell^2}{C^6}.$$

$$\mathbb{E}\big(Y_{i,j,k,\ell}(\mathbf{m}_7)\, Y_{i,j,u,v}(\mathbf{m}_7)\big) =$$

$$\begin{aligned}
&= \frac{d_i^2 d_j^2 d_k^2 d_\ell^2}{C^4}\left(\frac{d_j d_\ell}{C} + \frac{d_i d_k}{C} + \frac{d_j d_k}{C} + \frac{d_i d_\ell}{C} + \frac{d_i d_j}{C}\right)\\
&\quad\times \frac{d_i d_j d_u^2 d_v^2}{C^3}\left(\frac{d_j d_v}{C} + \frac{d_i d_u}{C} + \frac{d_j d_u}{C} + \frac{d_i d_v}{C} + \frac{d_i d_j}{C} + \frac{d_i d_j d_u d_v}{C^2}\right)\\
&\quad+ \frac{d_i^2 d_j^2 d_k^2 d_\ell^2}{C^4}\times\frac{d_k d_\ell}{C}\times\frac{d_i^2 d_j^2 d_u^2 d_v^2}{C^4}\left(\frac{d_j d_v}{C}+\frac{d_i d_u}{C}+\frac{d_j d_u}{C}+\frac{d_i d_v}{C}+\frac{d_i d_j}{C}+\frac{d_u d_v}{C}\right).
\end{aligned}$$

After simplifications, we have

$$\mathbb{E}\big(Y_{i,j,k,\ell}(\mathbf{m}_7)\, Y_{i,j,u,v}(\mathbf{m}_7)\big) =$$

$$\begin{aligned}
&= \frac{d_i^3 d_j^3 d_k^2 d_\ell^2 d_u^2 d_v^2}{C^7}\left\{(d_i+d_j)^2\,\frac{(d_u+d_v)\,(d_k+d_\ell)}{C^2}\right.\\
&\quad+ \frac{d_i d_j}{C^2}\,(d_i+d_j)\left[\left(1+\frac{d_u d_v}{C}\right)(d_k+d_\ell) + \left(1+\frac{d_k d_\ell}{C}\right)(d_u+d_v)\right]\\
&\quad\left.+ \frac{d_i^2 d_j^2}{C^3}\,(d_u d_v + d_k d_\ell) + \frac{d_i d_j d_k d_\ell d_u d_v}{C^3}\right\}.
\end{aligned}$$

Now we focus on $\mathbb{E}(Y_{i,j,k,\ell}(\mathbf{m}_7)\, Y_{i,j,k,u}(\mathbf{m}_7))$.

$$\mathbb{E}\big(Y_{i,j,k,\ell}(\mathbf{m}_7)\, Y_{i,j,k,u}(\mathbf{m}_7)\big) =$$

$$\begin{aligned}
&= \frac{d_i^2 d_j^2 d_k^2 d_\ell^2}{C^5}\left\{\frac{d_j d_\ell d_i d_k d_u^2}{C^3}\left[d_j d_u + d_i d_k + d_j d_k + \frac{d_i d_j d_k d_u}{C} + d_i d_j + \frac{d_i d_j d_k d_u}{C}\right]\right.\\
&\quad+ \frac{d_i d_\ell d_j d_k d_u^2}{C^3}\left[\frac{d_i d_j d_k d_u}{C} + d_i d_k + d_j d_k + d_i d_u + d_i d_j + \frac{d_i d_j d_k d_u}{C}\right]\\
&\quad+ \frac{d_k d_\ell d_i d_j d_u^2}{C^3}\left[\frac{d_i d_j d_k d_u}{C} + d_i d_k + d_j d_k + \frac{d_i d_j d_k d_u}{C} + d_i d_j + d_k d_u\right]\\
&\quad+ \left(d_i d_k + d_j d_k + d_i d_j\right)\frac{d_u^2}{C^2}\\
&\quad\left.\times\left[\frac{d_i d_j d_k d_u}{C} + d_i d_k + d_j d_k + \frac{d_i d_j d_k d_u}{C} + d_i d_j + \frac{d_i d_j d_k d_u}{C}\right]\right\}.
\end{aligned}$$

After simplifications, we have

$$\mathbb{E}\big(Y_{i,j,k,\ell}(\mathbf{m}_7)\, Y_{i,j,k,u}(\mathbf{m}_7)\big) \;=$$
$$= \frac{d_i^2 d_j^2 d_k^2 d_\ell^2 d_u^2}{C^6} \left\{ 3\,(d_i d_j + d_i d_k + d_j d_k)\, \frac{d_i d_j d_k}{C^2}\,(d_\ell + d_u) \right.$$
$$\left. + \,(d_i + d_j + d_k)\, \frac{d_i d_j d_k d_u d_\ell}{C^2} + 6\, \frac{d_i^2 d_j^2 d_k^2 d_\ell d_u}{C^3} + \frac{(d_i d_j + d_i d_k + d_j d_k)^2}{C} \right\}.$$

Now, we compute $\mathbb{E}Y^2_{i,j,k,\ell}(\mathbf{m}_7)$. Any product of two different indicator RVns appearing in $\mathbf{m}_7$ is equal to indicator RV of the complete graph on $\{i,j,k,\ell\}$. Thus,

$$\mathbb{E}Y^2_{i,j,k,\ell}(\mathbf{m}_7) \;=\; \mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_7) + 30\,\mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_8)\;,$$

where $\mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_8)$ is given below.

Motif $\mathbf{m}_8$

Motif $\mathbf{m}_8$ corresponds to a totally connected subgraph. In particular, $Y_{i,j,k,\ell}(\mathbf{m}_8)$ is an indicator RV, which simplifies calculations. We have

$$\mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_8) \;=\; \frac{d_i^3 d_j^3 d_k^3 d_\ell^3}{C^6}\;,$$

$$\mathbb{E}\big(Y_{i,j,k,\ell}(\mathbf{m}_8)\, Y_{i,j,u,v}(\mathbf{m}_8)\big) \;=\; \frac{d_i^5 d_j^5 d_k^3 d_\ell^3 d_u^3 d_v^3}{C^{11}}\;,$$

$$\mathbb{E}\big(Y_{i,j,k,\ell}(\mathbf{m}_8)\, Y_{i,j,k,u}(\mathbf{m}_8)\big) \;=\; \frac{d_i^4 d_j^4 d_k^4 d_\ell^3 d_u^3}{C^9}\;,$$

$$\mathbb{E}Y^2_{i,j,k,\ell}(\mathbf{m}_8) \;=\; \mathbb{E}Y_{i,j,k,\ell}(\mathbf{m}_8)\;.$$

REVSTAT – Statistical Journal
Volume 4, Number 1, March 2006, 53–65

INFERRING GENE DEPENDENCY NETWORKS FROM GENOMIC LONGITUDINAL DATA: A FUNCTIONAL DATA APPROACH

Authors: Rainer Opgen-Rhein
– Department of Statistics, University of Munich, Ludwigstrasse 33, D-80539 Munich, Germany
opgen-rhein@stat.uni-muenchen.de

Korbinian Strimmer
– Department of Statistics, University of Munich, Ludwigstrasse 33, D-80539 Munich, Germany
korbinian.strimmer@lmu.de

Abstract:

- A key aim of systems biology is to unravel the regulatory interactions among genes and gene products in a cell. Here we investigate a graphical model that treats the observed gene expression over time as realizations of random curves. This approach is centered around an estimator of dynamical pairwise correlation that takes account of the functional nature of the observed data. This allows to extend the graphical Gaussian modeling framework from i.i.d. data to analyze longitudinal genomic data. The new method is illustrated by analyzing highly replicated data from a genome experiment concerning the expression response of human T-cells to PMA and ionomicin treatment.

Key-Words:

- *graphical model; longitudinal data; dynamical correlation; gene dependency networks.*

AMS Subject Classification:

- 37N25, 62M10, 92B15, 92D10.

1. INTRODUCTION

The identification of networked genetic interdependencies that form the basis of cellular regulation is one of the key issues in systems biology. Consequently, many authors have investigated statistical approaches such as graphical models to estimate genetic networks from high-throughput data [e.g., 8, 7, 11].

A graphical model is a representation of stochastic conditional dependencies between the investigated variables. Among the simplest graphical models is the class of graphical Gaussian models (GGMs) — see, e.g., Whittaker [13]. In this framework gene network may be constructed as follows. First, a positive definite and well-conditioned estimate $\boldsymbol{R} = (r_{kl})$ of the linear correlation matrix $\boldsymbol{P} = (\rho_{kl})$ is inferred from the data. Second, the standardized inverse of this matrix gives an estimate $\tilde{\boldsymbol{R}} = (\tilde{r}_{kl})$ of the *partial* correlations $\tilde{\boldsymbol{P}} = (\tilde{\rho}_{kl})$. The strength of these coefficients indicate the presence or absence of a direct association between each pair of genes. For large sample size computation of covariances and GGM selection can be conducted using classical estimation and testing theory as outlined in Whittaker [13]. However, the small sample size relative to the large number of genes typically considered in genome experiments requires the additional application of shrinkage and other regularization techniques [2, 12].

A drawback shared by the GGM approach and other graphical models such as Bayesian networks is that these methods rely on the assumption of identically and independently distributed (i.i.d.) data. However, an increasing proportion of microarray expression experiments are concerned with *longitudinal* measurements of mRNA and protein concentrations. For instance, stress response and cell cycle experiments by design produce time course data. A further characteristic of these data is that the time points at which the experiments are conducted are almost always not equidistant but irregularly spaced.

In order to avoid these issues, in this paper we investigate GGM network inference from the perspective of functional data analysis [9]. Specifically, we describe a graphical model that treats the observed gene expression over time as realizations of random curves, rather than to describe the individual time points separately. This approach is based on the notion of *dynamical correlation* which provides a similarity score for pairs of groups of randomly sampled curves. Subsequently, it allows computation of partial dynamical correlations and the identification of the associated network structure.

The remainder of the paper is organized as follows. In the next section we summarize the basic notation for functional data analysis and also introduce the functional inner product. Next, we discuss the concept of dynamical correlation of which we describe two different variants, one introduced in this paper and one by Dubin and Müller [3]. Subsequently, the dynamical correlation is employed for GGM network selection. Finally, in order to compare the traditional GGM method with the present approach we reanalyze data from a human T-cell experiment with 58 genes, 10 time points, and 44 replications [10], and compare the networks resulting from dynamical correlation with those from static correlation.

2. METHODS

2.1. Setup and notation

We consider data from a typical gene expression time course experiment. For p genes (variables) and n subjects (replications) mRNA concentrations are measured over a time interval $[A, B]$. This results in functional observations $f_{ik}(t)$ where $1 \leq i \leq n$ and $1 \leq k, l \leq p$. We assume all functions $f_{ik}(t)$ to be square-integrable so that the functional inner product

$$\big\langle g(t), h(t) \big\rangle = \frac{1}{B-A} \int_A^B g(t)\, h(t)\, dt \tag{2.1}$$

exists, where $g(t)$ and $h(t)$ are any of the observed functions. The time average of $f_{ik}(t)$ may then be conveniently expressed by $\langle f_{ik}(t), 1 \rangle$. The average over the n replicates gives the empirical mean function $\bar{f}_k(t) = \frac{1}{n} \sum_{i=1}^n f_{ik}(t)$.

In practice, however, the functions $f_{ik}(t)$ are not continuously measured but rather obtained by experiments at discrete time points t_j, with $1 \leq j \leq m$ and $A = t_1 < t_2 < \ldots < t_{m-1} < t_m = B$. Note that the time points need not be equidistant. If one assumes a linear approximation of $g(t)$ and $h(t)$ the inner product of Eq. 2.1 turns into the weighted sum

$$\big\langle g(t), h(t) \big\rangle \approx \sum_{j=1}^{m} g(t_j)\, h(t_j)\, \frac{\delta_j + \delta_{j+1}}{2(B-A)} \tag{2.2}$$

where the $\delta_j = t_j - t_{j-1}$ are the time differences between subsequent measurements (with $\delta_1 = \delta_{m+1} = 0$).

In the random effects representation of Dubin and Müller [3] each observed $f_{ik}(t)$ is a realization of the random function

$$f_k(t) = \mu_k(t) + \mu_{0k} + \epsilon_{0k} + \sum_{u=1}^{\infty} \epsilon_{uk}\, \eta_u(t)\ , \tag{2.3}$$

where ϵ_{0k} and ϵ_{uk} are random variables with $E(\epsilon_{0k}) = 0$ and $E(\epsilon_{uk}) = 0$, $\mu_k(t)$ is the fixed time dependent mean function with zero time average $\langle \mu_k(t), 1 \rangle = 0$, $\mu_{0k} + \epsilon_{0k}$ represents the static random part and the remaining terms describe the dynamic random part. In Eq. 2.3 the $\eta_u(t)$ are orthonormal basis functions with zero time average $\langle \eta_u(t), 1 \rangle = 0$.

In this notation the empirical mean function $\bar{f}_k(t)$ is an estimate of $E(f_k(t)) = \mu_k(t) + \mu_{0k}$. As $\mu_k(t)$ has time average zero we are also able to identify the two components of $E(f_k(t))$ by using $\hat{\mu}_{0k} = \langle \bar{f}_k(t), 1 \rangle$ and $\hat{\mu}_k(t) = \bar{f}_k(t) - \hat{\mu}_{0k}$.

2.2. Dynamical correlation

2.2.1. Measuring similarity between two exactly known curves

Suppose for a moment that we have sufficient data to estimate the expression levels through time of two genes k and l *exactly*, i.e. that we know the mean functions $E(f_k(t))$ and $E(f_l(t))$. In order to understand the functional connection between these two variables a measure of similarity between the two curves is required. Dubin and Müller [3] suggest to introduce the notion of *dynamical correlation* with the informal proposition that "if both trajectories tend to be mostly on the same side of their time average (a constant) then the dynamical correlation is positive; if the opposite occurs, then dynamical correlation is negative".

This immediately leads to the following straightforward definition of dynamical correlation between two curves $g(t)$ and $h(t)$. First, compute the time-centered functions $g^C(t) = g(t) - \langle g(t), 1\rangle$ and $h^C(t) = h(t) - \langle h(t), 1\rangle$. Then define the variances as

$$\mathrm{Var}\big(g(t)\big) = \big\langle g^C(t), g^C(t)\big\rangle$$

and

$$\mathrm{Var}\big(h(t)\big) = \big\langle h^C(t), h^C(t)\big\rangle \ .$$

Finally, compute the the standardized functions $g^S(t) = g^C(t)/\sqrt{\mathrm{Var}(g(t))}$ and $h^S(t) = h^C(t)/\sqrt{\mathrm{Var}(h(t))}$, and obtain the correlation by

$$\mathrm{Cor}\big(g(t), h(t)\big) = \big\langle g^S(t), h^S(t)\big\rangle \ .$$

2.2.2. The general case including sampling error

The above definition of dynamical correlation for a single curve extends in a straightforward fashion to the case where each observed time course f_{ik} represents a noisy realization of the mean function $E(f_k)$.

In order to estimate the correlation between two variables k and l we first define the simultaneously time- *and* space-centered functions according to $f_{ik}^C(t) = f_{ik}(t) - \langle \bar{f}_k(t), 1\rangle$. Note that here the inner product is computed over the mean function $\bar{f}_k(t)$. Based on the $f_{ik}^C(t)$ an estimate of the variance of variable k is then given by

$$\widehat{\mathrm{Var}}_k \ = \ \hat{\sigma}_{kk} \ = \ s_{kk} \ = \ \frac{1}{n-1}\sum_{i=1}^{n}\big\langle f_{ik}^C(t), f_{ik}^C(t)\big\rangle \ . \tag{2.4}$$

This allows to compute standardized residual functions $f_{ik}^S(t) = f_{ik}^C / \sqrt{s_{kk}}$ that form the basis for the estimate of dynamical correlation

$$\widehat{\mathrm{Cor}}_{kl} \;=\; \hat{\rho}_{kl} \;=\; r_{kl} \;=\; \frac{1}{n-1} \sum_{i=1}^{n} \left\langle f_{ik}^S(t), f_{il}^S(t) \right\rangle \;. \tag{2.5}$$

Correspondingly, the estimated dynamical covariance between variables k and l is simply

$$\widehat{\mathrm{Cov}}_{kl} \;=\; \hat{\sigma}_{kl} \;=\; s_{kl} \;=\; r_{kl} \sqrt{s_{kk} s_{ll}} \;. \tag{2.6}$$

This simple estimator of dynamical correlation exhibits several attractive properties. In particular, it is a generalization of the standard correlation for cross-sectional data. Specifically, if $m = 1$ and $n > 1$ then it reduces to the usual maximum-likelihood estimator of correlation. Furthermore, it is also applicable if there is only a single realization of each time series available ($n = 1$, $m > 1$).

2.2.3. The Dubin–Müller definition of dynamical correlation

Another related but different definition of dynamical correlation is given by Dubin and Müller [3]. They propose to compute the standardized residual functions according to

$$f_{ik}^S(t) \;=\; q_{ik}(t) / \sqrt{\left\langle q_{ik}(t), q_{ik}(t) \right\rangle} \tag{2.7}$$

using

$$q_{ik}(t) \;=\; f_{ik}(t) - \bar{f}_{ik}(t) - \left\langle f_{ik}(t), 1 \right\rangle + \left\langle \bar{f}_{ik}(t), 1 \right\rangle \;. \tag{2.8}$$

This definition has the drawback that it is only defined if both $m > 1$ and $n > 1$. As we will exemplify below, it also produces counter-intuitive correlations.

2.3. Estimating gene association networks using dynamical correlation

The basic idea to infer a network from the pairwise dynamical correlation is to refer to the genes as the nodes and to the correlations as the connectivity strengths assigned to the edges of the network. However, we cannot use the correlations directly, because they represent only marginal dependencies and also include indirect interactions between two variables. Instead, we need to rely on the concept of *partial* correlation which describe the correlation between any two variables i and j conditioned on all the other variables. It is straightforward to

compute the matrix of partial dynamical correlations $\tilde{\boldsymbol{P}} = (\tilde{\rho}_{kl})$ from the correlation coefficients $\boldsymbol{P} = (\rho_{kl})$ via the inverse relationship

$$\boldsymbol{\Omega} \;=\; \boldsymbol{P}^{-1} \;=\; (\omega_{ij}) \tag{2.9}$$

$$\tilde{\rho}_{kl} \;=\; -\frac{\omega_{kl}}{\sqrt{\omega_{kk}\,\omega_{ll}}} \tag{2.10}$$

[4]. Applying these equations to estimates $\boldsymbol{R} = (r_{kl})$ of (dynamical) correlations allows to obtain estimates $\tilde{\boldsymbol{R}} = (\tilde{r}_{kl})$ of the associated partial (dynamical) correlations.

In order to test the significance of the correlations and to decide which of the possible edges to include in the resulting gene association network statistical tests are needed. In this paper we employ the "local fdr" network search as proposed by Schäfer and Strimmer [11, 12]. The false discovery rate (fdr) is the expected proportion of false positives among the proposed edges. The local fdr is an empirical Bayes estimator of the false discovery rate proposed by Efron [5, 6]. This method computes the posterior probability for an edge to be present or absent, and takes account of the multiplicity in the simultaneous testing of edges. The final network is obtained by visualizing all significant edges in an undirected graph.

3. RESULTS

In the following section we first apply our method of computing dynamical correlation to example data to clarify our definition and to compare it with the related concept of Dubin and Müller [3]. Subsequently, we infer the gene association network for a longitudinal gene expression data set described in Rangel et al. [10].

3.1. Illustrative example

In order to understand the concept of dynamical correlation and to illustrate the difference between our definition (Eq. 2.5) and that of Dubin and Müller [3] we first consider a set of artificial examples. These are shown in Fig. 1 where two negatively dependent variables are depicted. For instance, this may represent the case where one gene is up-regulated and the other is correspondingly down-regulated. For each gene there are two measured curves, and there are three slightly different ways in which the sampled curves relate to each other (Fig. 1a, b, and c). The exact definition of the curves can be found in Tab. 1. Note that the two realizations are paired, i.e. the upper lines belong to individual 1 and the lower ones to individual 2.

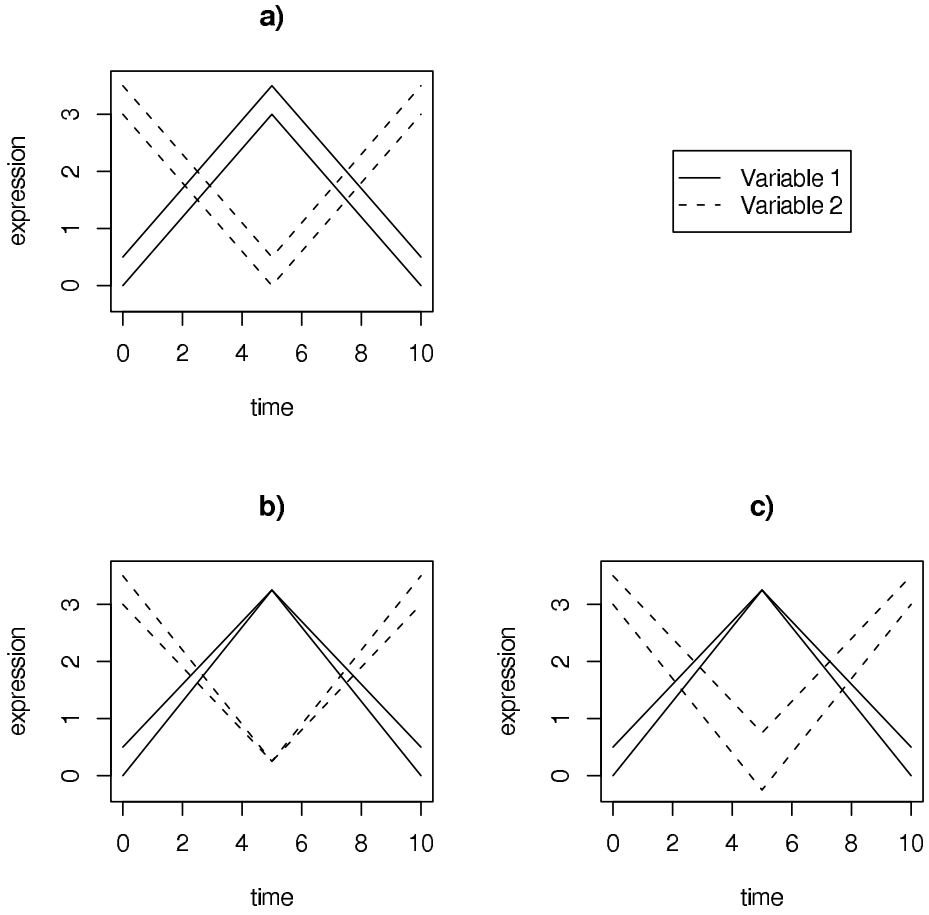

Figure 1: Toy example to illustrate the concept of dynamical correlation between two variables ("genes"). In all three cases a), b) and c) there are two realizations ("individuals"). See main text for details, and Tab. 1 for the underlying data.

Table 1: Data points of the toy examples in Fig. 1.

Data		**Variable 1**			**Variable 2**		
Time points		*0*	*5*	*10*	*0*	*5*	*10*
Fig. 1a	*Realization 1*	0	3	0	3	0	3
	Realization 2	0.5	3.5	0.5	3.5	0.5	3.5
Fig. 1b	*Realization 1*	0	3.25	0	3	0.25	3
	Realization 2	0.5	3.25	0.5	3.5	0.25	3.5
Fig. 1c	*Realization 1*	0	3.25	0	3	−0.25	3
	Realization 2	0.5	3.25	0.5	3.5	0.75	3.5

Intuitively, one would expect that the dynamical correlation between the two variables is strongly negative in all three cases. For our definition of dynamical correlation according to Eq. 2.5 this is indeed the case: the correlations for the three examples cases Fig. 1a, b, and c are -0.946, -0.982, and -0.947, respectively. In contrast, the dynamical correlation of Dubin and Müller [3] behaves in a completely different fashion. For Fig. 1a it is not defined, for case b) it is equal to $+1$ and for case c) it is equal to -1.

Therefore, it is easy to see that the Dubin and Müller [3] estimator is *not* suited for detecting functional dependencies in genomic longitudinal data. This is because that estimator is geared towards detecting changes in the relative trends of the individual realizations, rather than between the common trend. However, note that this is generally not the effect one wants to identify when looking for gene interaction. In addition, the Dubin and Müller [3] definition of dynamical correlation has the additional disadvantage over that of Eq. 2.5 that it is not defined if there is only a single time course per gene available. In contrast, the above toy examples show that our definition of dynamical correlation is able to detect the main trend of positive or negative dependency between two variable, and is not susceptible to the small changes in the sampled curves.

3.2. Gene expression time course data

We now employ our method of estimation of the (partial) dynamical correlation to a real world example and compare it with the results of the traditional GGM method. Specifically, we reanalyzed a microarray time series data set described in detail in Rangel et al. [10]. These data characterize the response of a human T-cell line (Jirkat) to a treatment with PMA and ioconomin. After preprocessing the time course data consist of 58 genes measured across 10 time points with 44 replications. Rangel et al. [10] used a state space model to estimate the influence between genes and measured a genetic network by combining direct effects and indirect effects via hidden states. This approach is generally very time-consuming due to the necessity of using of the EM algorithm for optimization. A peculiarity of the Rangel et al. [10] data is also that the measurements in the experiment were taken at unequally spaced time points, i.e. after 0, 2, 4, 6, 8, 18, 24, 32, 48, and 72 hours after treatment. This was neglected in the original state-space analysis which assumed equally spaced data. In contrast, note that the present functional data approach allows the incorporation of arbitrary time distances between subsequent measurements.

As approximation of the temporal expression of the 58 genes we used a linear spline and employed Eq. 2.2 for the functional inner product. After estimating the dynamical correlations with Eq. 2.5 we computed the associated partial correlation coefficients employing Eq. 2.9 and Eq. 2.10. Fig. 2 shows the histogram of the estimated partial correlation coefficients after Fisher's normaliz-

ing z-transformation. Also depicted in this plot are the fitted overall distribution (fat line) and the null (dashed line) and alternative distribution (filled histogram) as estimated by the locfdr algorithm [5, 6]. The 0.2 local fdr cut-off values for the partial correlations are indicated by the black triangles. As expected, the distribution of the partial correlations is centered around zero and most of the coefficients are not significant. Consequently, the resulting network is sparse and there are only 54 significant edges. The network itself is displayed in Fig. 3b.

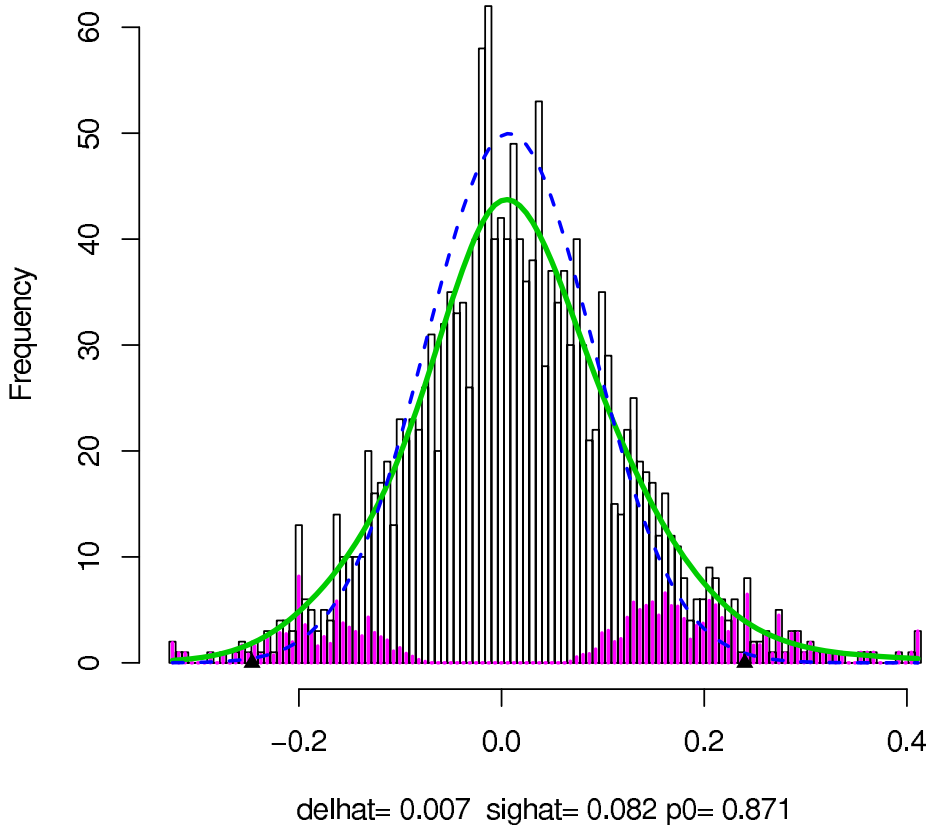

Figure 2: Histogramm of the Fisher z-transformed estimated partial dynamical correlations. Values left and right the two black triangles are considered significantly different from zero, and thus correspond to edges in a gene dependency network.

It is instructive to compare the genetic network inferred with dynamical correlation to the gene association network obtained by the classic GGM approach. For this analysis we ignored the dynamic aspects of the data and assumed that all measurements were taken at the same time point, which leads to 440 observations (44 replications times 10 time points) for each of the 58 genes. As this number of observations is not small in comparison to the number of the genes no regularization is needed (cf. Schäfer and Strimmer [12] for the opposite case).

From the empirical correlation matrix we proceeded as above, obtaining estimates of partial correlation and a static GGM network. This is displayed on the left side of figure 3. For comparison, the network estimated with dynamical correlation in shown on its right side. For clarity only the nodes which have at least one connection are displayed.

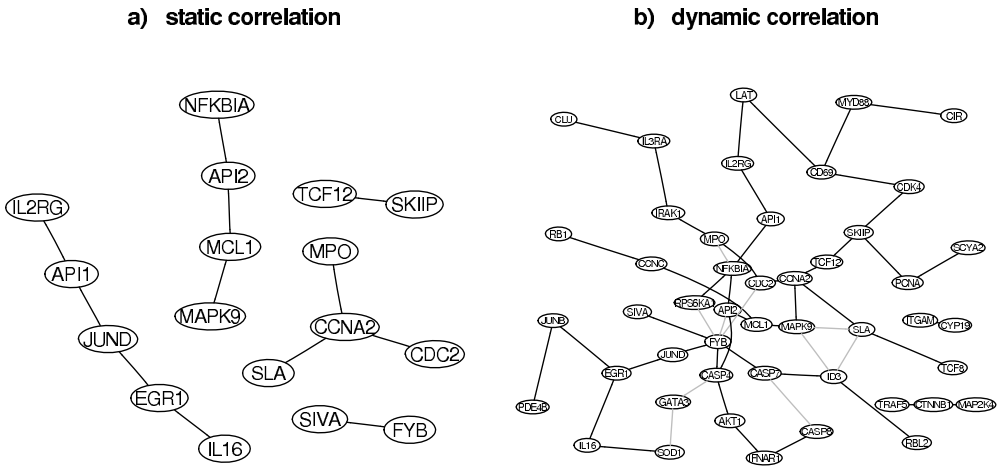

Figure 3: Gene dependency networks inferred from human T-cell data [10] using (a) static correlation and (b) dynamical correlation.

The network calculated with static correlation consists of 17 nodes with 12 edges, a smaller network than the one based on dynamical correlation. This indicates that our dynamical estimator is able to identify additional time-varying components of the interaction between the investigated genes.

4. DISCUSSION

A growing interest in genetics lies in observing and inferring the gene interactions over time. Here, we introduced a method to infer a gene dependency network from functional data. In this approach time course experiments are seen as a realization of random curves. The method described generalizes the widely used static GGM approach (see the corresponding references in [11]) and is able to unravel the dependency structure of longitudinal data across the whole time series rather than at single time points. Furthermore, unlike many other time series method the functional approach does not require equally spaced measurements. In addition, our algorithm is easily implemented and computationally inexpensive (the calculation of the above gene dependency network takes only a fraction of a second).

In order to further develop our approach many extensions are conceivable. For instance, in the above analysis of human T-cells the data was highly replicated. In genomics, however, it is more typical that the sample size is very small compared to the number of genes (this is the so-called "small n, large p" paradigm). In this case, the empirical covariance is a highly inefficient estimator, and needs to be regularized [12]. For small n this will also be the case with our estimate of dynamical correlation (Eq. 2.5). Thus, shrinkage techniques similar to those of Schäfer and Strimmer [12] are needed.

A further important extension is the inclusion of autoregressive aspects [1]. While our method covers the dynamical correlation through time it is not able to account, e.g., for a time shift between any two variables. This is illustrated in Fig. 4 which is a variation of the toy examples presented in section 3. For this data the Dubin and Müller [3] estimate is (again) not defined and our suggested dynamical estimator results in very small correlation close to zero, even though it is clear by inspection that the two depicted variables are strongly connected. These dependencies and the associated time shifts could be accounted for by modeling the temporal mean via a system of differential equation (or in the discrete case by some autoregressive process). We also note that for this reason we have also refrained here from a comparison of the gene association network inferred from dynamical correlation (Fig. 3b) with the state space network presented by Rangel et al. [10]. Future work should regard for these aspects.

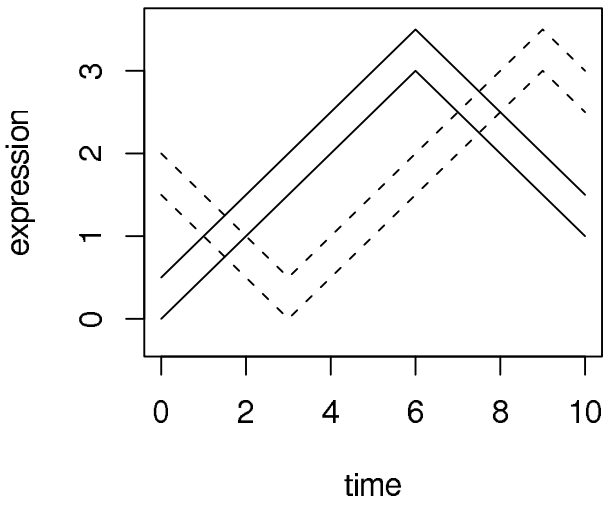

Figure 4: Example with a fixed time lag between the two variables.

ACKNOWLEDGMENTS

K.S. thanks the organizers of the "Workshop on Statistics in Genomics and Proteomics (WSGP 2005)" at Monte Estoril, Portugal (5–8 October 2005) for a stimulating meeting. This work was supported by Deutsche Forschungsgemeinschaft (DFG) Emmy-Noether research award to K.S.

REFERENCES

[1] Diggle, P.J.; Heagerty P.J.; Liang K.-Y. and Zeger, S. L. (2002). *Analysis of Longitudinal Data, 2nd Edition*, Oxford: Oxford University Press.

[2] Dobra, A.; Hans, C.; Jones, B.; Nevins, J. R.; Yao, G. and West M. (2004). Sparse graphical models for exploring gene expression data, *J. Multiv. Anal.*, **90**, 196–212.

[3] Dubin, J. A. and Müller, H.-G. (2005). Dynamical correlation for multivariate longitudinal data, *J. Amer. Statist. Assoc.*, **100**, 872–881.

[4] Edwards, D. (1995). *Introduction to Graphical Modelling*, New York: Springer.

[5] Efron, B. (2004). Large-scale simultaneous hypothesis testing: the choice of a null hypothesis, *J. Amer. Statist. Assoc.*, **99**, 96–104.

[6] Efron, B. (2005). *Local false discovery rates*, Technical Report, Dept. of Statistics, Stanford University.

[7] Friedman, N. (2004). Inferring cellular networks using probabilistics graphical models, *Science*, **303**, 799–805.

[8] Hartemink, A.J.; Gifford, D.K.; Jaakkola, T.S. and Young, R.A. (2002). Bayesian methods for elucidating genetic regulatory networks, *IEEE Intell. Systems*, **17**, 37–43.

[9] Ramsay, J.O. and Silverman B.W. (2005). *Functional Data Analysis, 2nd Edition*, New York: Springer Verlag.

[10] Rangel, C.; Angus, J.; Ghahramani, Z.; Lioumi, M.; Sotheran, E.; Gaiba, A.; Wild, D.L. and Falciani, F. (2004). Modeling T-cell activation using gene expression profiling and state space modeling, *Bioinformatics*, **20**, 1361–1372.

[11] Schäfer, J. and Strimmer, K. (2005a). An empirical Bayes approach to inferring large-scale gene association networks, *Bioinformatics*, **21**, 754–764.

[12] Schäfer, J. and Strimmer, K. (2005b). A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics, *Statist. Appl. Genet. Mol. Biol.*, **4**, 32.

[13] Whittaker, J. (1990). *Graphical Models in Applied Multivariate Statistics*, New York: Wiley.

REVSTAT – Statistical Journal
Volume 4, Number 1, March 2006, 67–80

STATISTICAL EVALUATION OF METHODS FOR THE ANALYSIS OF DYNAMIC PROTEIN EXPRESSION DATA FROM A TUMOR STUDY

Authors: Klaus Jung
– Department of Statistics, University of Dortmund, Germany
klaus.jung@uni-dortmund.de

Ali Gannoun
– Equipe de Probabilités et Statistique, Université Montpellier, France

Barbara Sitek
– Medical Proteom-Center, Ruhr-University Bochum, Germany

Ognjan Apostolov
– Medical Proteom-Center, Ruhr-University Bochum, Germany

Alexander Schramm
– University Children's Hospital of Essen,
Division of Hematology, Oncology and Endocrinology, Germany

Helmut E. Meyer
– Medical Proteom-Center, Ruhr-University Bochum, Germany

Kai Stühler
– Medical Proteom-Center, Ruhr-University Bochum, Germany

Wolfgang Urfer
– Department of Statistics, University of Dortmund, Germany

Abstract:

- In this article, we analyze time dependent protein expression data obtained from a proteome study of a neuroblastoma cell line. Neuroblastoma are common solid tumors which occur in early childhood. The expression data was obtained by difference gel electrophoresis (DIGE). It is known that the clinical outcome of neuroblastoma depends on the activation of different neurotrophin receptors by their ligands. Here, we are looking for proteome changes resulting from the activation of Tyrosine Kinase (TrkA) receptors by their ligand NGF (nerve growth factor). Before analyzing the data by longitudinal data analysis we do data preprocessing and apply a method for the imputation of missing values.

Key-Words:

- *protein expression data; proteome; data preprocessing; missing values imputation; longitudinal data analysis; neuroblastoma.*

1. INTRODUCTION

The term 'proteome' stands for all proteins, which are coded by a genome at specific time points and under certain conditions. It is known that in addition to the analysis of the genome investigation of the highly complex and dynamic proteome will provide a far more detailed description of biological processes. To this end a number of proteomic techniques have been developed which allow the analysis of complex protein mixtures. Currently the two-dimensional electrophoresis (2-DE) is the separation method with highest resolution power for protein samples. Up to 10,000 proteins can be separated in one gel and therefore are accessible for quantitative analysis (cf. Klose and Kobalz ([11])). Statistical methods for the analysis of protein expression data from 2-DE comprise data preprocessing, multiple hypothesis testing and nearly the whole spectrum of multivariate techniques. In Jung et al. ([8]) we reviewed and presented some methods for data preprocessing, missing values imputation and longitudinal data analysis. Here, we apply and evaluate these techniques by analyzing protein expression data from a proteome study of the neuroblastoma cell line SY5Y. Neuroblastoma are common solid tumors which occur in early childhood. The proteome of neuroblastoma depends on the activation of different neurotrophin receptors (TrkA and TrkB) by their ligands (cf. Nakagaware et al. ([13])). In this article, we compare proteome samples of the SY5Y cell line when the TrkA receptors are activated by their ligand NGF (nerve growth factor) and when they are not activated. Hence, we have a treatment and a control group. The experiment is detailed in Sitek et al. ([16]). The protein expression in the two groups was measured at 5 time points (0, 0.5, 1, 6 and 24h) with 4 biological replicates at time 0 and 5 biological replicates at each of the other time points. The data was obtained using the latest improvement of 2-DE, the so called Difference Gel Electrophoresis (DIGE). This technique allows one to put up to three different samples on the same gel. These samples (usually treatment, control and an internal standard) are tagged by different fluorophores (Cy2, Cy3 and Cy5). The internal standard is used to standardize all gels to the same level. 2-DE separates the proteins of a mixture by their isoelectric point (pI) and molecular size to distinct spots. After separation the proteins are detected using a confocal fluorescence scanner where fluorescence intensity of a spot can be regarded as a measure of expression for its respective protein. For quantitative proteome analysis image analysis software (DeCyder V5.0, Amersham Biosciences ([3])) automatically determines the boundaries and sizes of the spots.

Our article is organized as follows. In section 2 we analyze the performance of the preprocessing methods like calibration, normalization and standardization. In section 3 we evaluate the k nearest neighbour method for the estimation of missing values with respect to an estimation error. Furthermore, we apply an analysis of variance model for longitudinal data to the neuroblastoma data in section 4 and discuss the biological implications. Finally, we will mention future challenges in statistical proteomics.

2. DATA PREPROCESSING

Before starting the actual statistical analysis of expression values from 2-D fluorescence difference gel electrophoresis (DIGE) several preprocessing steps are required. In this chapter we examine procedures for calibration, normalization and standardization of such expression values. In particular, we evaluate the performance of the preprocessing methods that were proposed by Karp et al. ([10]). The figures in this chapter are based on the measurements taken from the 'master gel' of the TrkA experiment, i.e. the gel with the greatest number of detected spots (3562, here). Nevertheless, we obtained the same results from all other gels of the experiment.

2.1. Calibration

An impression of the necessity of calibration can be obtained from figure 1 were the raw background subtracted spot volumes (that have been obtained from the DeCyder software) of the Cy2, Cy3 and Cy5 labelled samples are plotted against each other. The plots show linear dependencies between the different

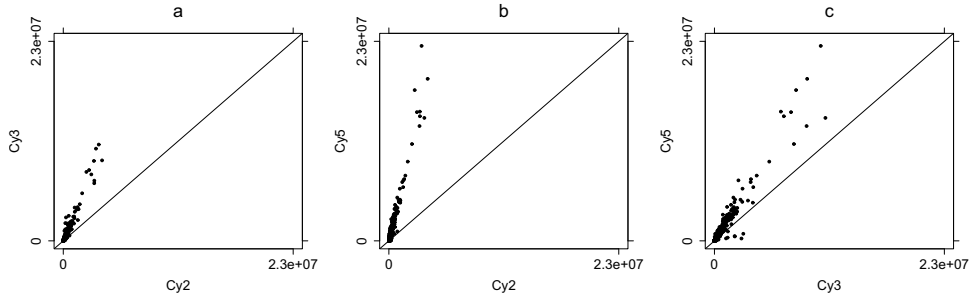

Figure 1: Raw background subtracted spot volumes spot volumes of the Cy2, Cy3 and Cy5 labelled samples plotted against each other.

labelled samples. However, the point clouds appear not on the line of gradient unity, so it can be assumed that the scatter is not only due to biological variation but also to some dye effect. To remove this technical variation given by these dye effects Karp et al. ([10]) and Kreil et al. ([12]) proposed to use the calibration model

$$y_{ij} \;=\; a_j + b_j\,\tilde{y}_{ij}\;, \tag{2.1}$$

separately for each gel, with $i=1,...,n$ and $j=1,2,3$, where $\tilde{y}_{ij}$ is the measured background subtracted spot volume of the ith spot from the sample that has

been labelled with the jth dye. The calibrated value of this spot volume is y_{ij}. The dye effects are adjusted by the scaling factors b_j and the additive offsets a_j compensate for any constant additive bias present after background subtraction. This calibration model was developed by Huber et al. ([7]) for the calibration of DNA microarrays. A corresponding software package, called 'vsn', for the open source statistic software R (available at `http://cran.r-project.org`) uses a robust version of maximum likelihood estimation for the estimation of the model parameters. We will call this preprocessing method the 'vsn-method', here. After calibration the spot volumes scatter around the bisecting line (figure 2) and the scatter should now represent only the biological variation. This calibration

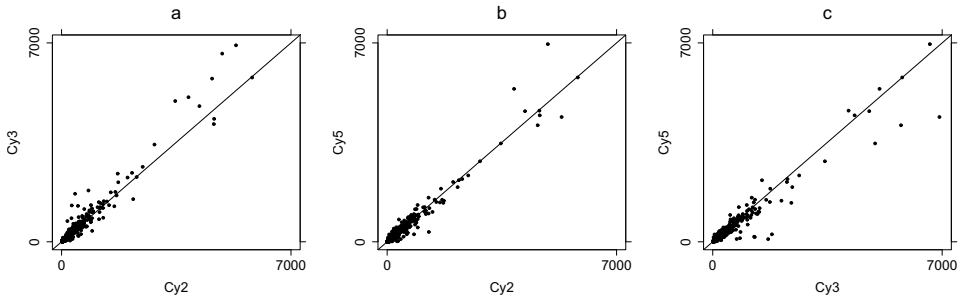

Figure 2: Spot volumes, calibrated by the vsn-method, of the Cy2, Cy3 and Cy5 labelled samples plotted against each other.

method raises the question whether the dye effects were the same for all gels, so we compared the estimated parameters when calibrating each gel of the TrkA experiment. Table 1 shows the mean and its percentage deviation of the calibration factors and offsets for all gels of the experiment. As we can see the percentage deviations from the means are higher than 100%, so there are obviously different dye effects from gel to gel. Hence, the calibration has to be done separately for each gel.

Table 1: The mean and its percentage deviation of the calibration factors and offsets, respectively, when using calibration model (2.1) for each gel of the TrkA experiment.

j	$\mu_1 = \text{mean}(a_j)$	deviation(μ_1)	$\mu_2 = \text{mean}(b_j)$	deviation(μ_2)
1	0.0006	128.0%	4.45	166.7%
2	0.0003	134.3%	6.67	155.1%
3	0.0001	125.8%	7.34	154.9%

2.2. Variance stabilization and normalization

In figure 2 it can be seen that the deviations of the spot volumes from the different labelled samples calibrated by the two methods is bigger for big values than the deviation for small values. For all five gels that have been prepared with the samples taken at time five (24h) we calibrated the expression values by the above method. From these values we calculated the mean and the variance of each spot. We analyzed only those spots which have been detected on at least three gels of time five, i.e. 1910 spots. The ranks of the means are plotted against the variances in figure 3a. Here, it can also be seen that the variance for big values

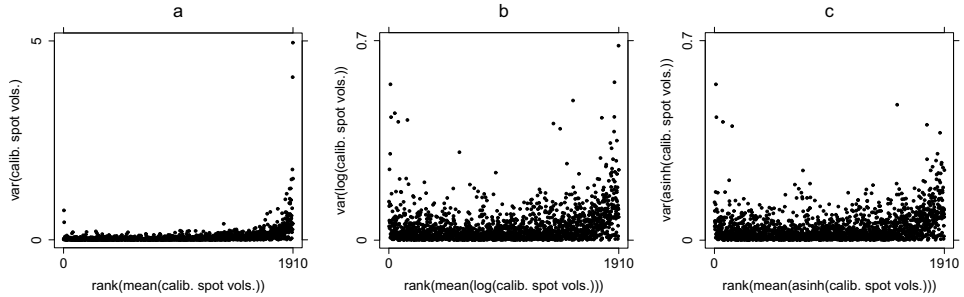

Figure 3: Rank of the mean versus the variance of a) the calibrated spot volumes, b) the calibrated and log-transformed spot volumes and c) the calibrated and asinh-transformed spot volumes.

is larger than the variance for small values. For this reason, in the standardisation process (cf. next section) where the internal standard is subtracted from the treatment and from the control, respectively, we also apply a transformation to stabilise the variance. One can either apply the logarithm or the asinh on the calibrated values to get a uniformly distributed variance. Figure 4 shows the calibrated spot volumes with the logarithm applied on them. However,

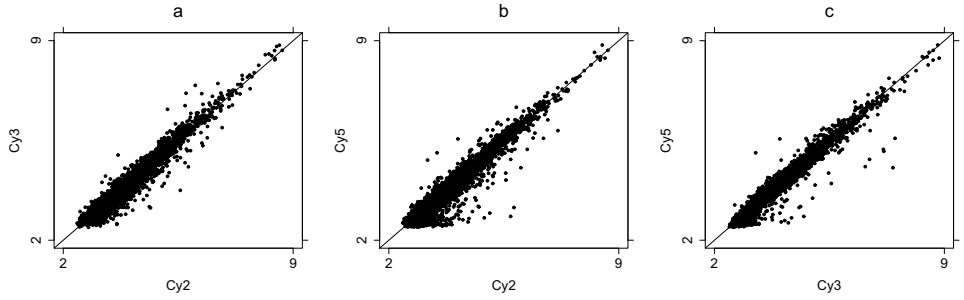

Figure 4: Calibrated and log-transformed spot volumes.

the logarithm goes very fast to $-\infty$ for small values and can thus causes a bias for small values. Instead of the logarithm one can also use the asinh. This is a function that is similar to the logarithm but smoother for small values. The calibrated and asinh-transformed values are plotted in figure 5. The effect

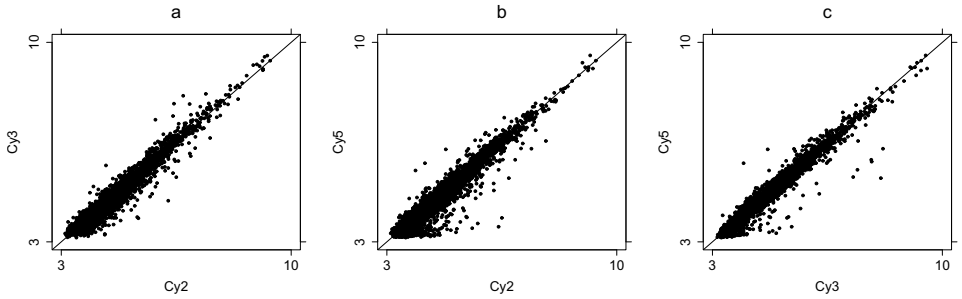

Figure 5: Calibrated and asinh-transformed spot volumes.

of these transformations on the variance-mean-dependencies can be seen in figure 3. Fig. 3b and c show that after applying the logarithm or the asinh transformation to the calibrated values the variance is stabilised with respect to the mean.

2.3. Standardization

The benefit of the DIGE method is to have an internal standard on each gel. The internal standard is a sample consisting of aliquots from all other samples of the experiment. Subtracting the values of the internal standard from values from the treated and untreated samples brings all gels on the same level and thus reduces the gel-to-gel variance. The complete preprocessing for the treatment values is thus given by either

$$\log(a_2 + b_2\,\tilde{y}_{i2}) - \log(a_1 + b_1\,\tilde{y}_{i1})\ , \tag{2.2}$$

or by

$$\operatorname{asinh}(a_2 + b_2\,\tilde{y}_{i2}) - \operatorname{asinh}(a_1 + b_1\,\tilde{y}_{i1})\ , \tag{2.3}$$

and similarly for the control values. In figure 6 the density histogram of the vsn-processed and standardized values for the treatment values is given. This distribution is symmetric and nearly normally distributed as can also be seen in the QQ-plot.

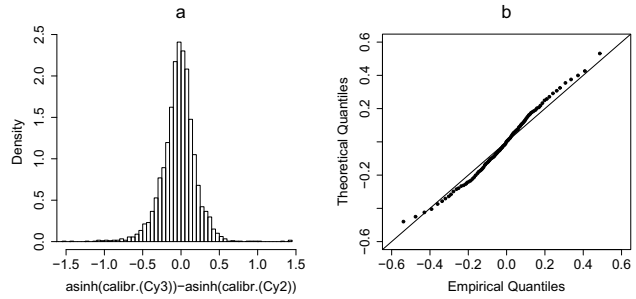

Figure 6: a) Density histogram of the preprocessed spot volumes from the treatment sample. b) QQ-plot from these values.

3. ESTIMATION OF MISSING VALUES

Many statistical methods, especially those for multivariate data, are based on the assumption that the data set to be analyzed is complete. However, in 2-D DIGE studies with gel replicates between 10 to 30 % of the values are missing. This is due to the fact that not all spots are detected or matched on each gel. In this section we compare two methods for the estimation of missing values, the 'row-mean method' and the 'k-nearest neighbour ($k$nn) method'. The latter one has already been successfully applied to microarray data (cf. Troyanskaya et al. ([17])). To illustrate these two methods we consider a simple example with artificial data of six spots on four gels (cf. table 2). In this example spot 2 on gel 3

Table 2: Artificial gel data with a missing value for spot 2 on gel 3.

	gel 1	gel 2	gel 3	gel 4
spot 1	24.21	28.87	21.59	22.79
spot 2	26.43	18.07		23.84
spot 3	238.42	270.97	258.74	233.63
spot 4	27.53	30.05	25.35	28.50
spot 5	132.58	152.61	144.09	148.82
spot 6	250.41	277.93	273.65	264.53

has not been detected, so the value is missing. The row mean method simply uses the average of all available measurements in the row where the value is missing as estimator for this missing value. For the example in table 2 the estimated value for spot 2 on gel 3 is then $(26.43 + 18.07 + 23.84) \,/\, 3 = 22.78$. The underlying idea of the $k$nn method is that there is a relationship between the expression profiles of some proteins. So, if a value for spot x is missing, the

method uses the values from those spots which are strongly related to this spot x. To determine the relationships of spot x to all other spots of the data set one can use a distance measure like the Euclidean, the Mahalanobis or the Chebyshev distance (cf. Jung et al. ([8])). If the value for spot x is missing on gel y, these distances are calculated by using only the values from the gels other than gel y. In the example, we use the values from gel 1, 2 and 4 to calculate the distances of spot 2 to all other spots. Spots 1 and 4 have a very short distance to spot 2, here, so we take the values from spot 1 and 4 on gel 3 to estimate the missing value, for example by taking the average of these values: $(21.59 + 25.35) \,/\, 2 = 23.47$. Other possible estimators are the median or some weighted mean. An important question that appears when using the $k$nn method is, how many neighbours should be used for the estimation. To determine the estimation error we used the five gels from the fifth time point in the TrkA experiment, removed all rows with missing values, so that a complete data set A with 526 rows and 5 columns remained. From this data set we generated 4 incomplete data sets B1 to B4 with 5, 10, 20 and 30 % of randomly chosen missing values, respectively. Then we applied the $k$nn method using different numbers k of neighbours. The resulting filled up data sets C1 to C4 were then compared to the original complete data set A by calculating the normalized root mean square (RMS) error:

$$\text{(3.1)} \qquad \textit{normalized RMS error} \;=\; \frac{\sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m}(A_{ij}-C_{ij})^2/(n*m)}}{\sum_{i=1}^{m}\sum_{j=1}^{m}A_{ij}/(n*m)}\,,$$

where n is the number of spots and m is the number of replicates. A plot of this error is given in figure 7. Plot 7a shows the error when using the $k$nn method

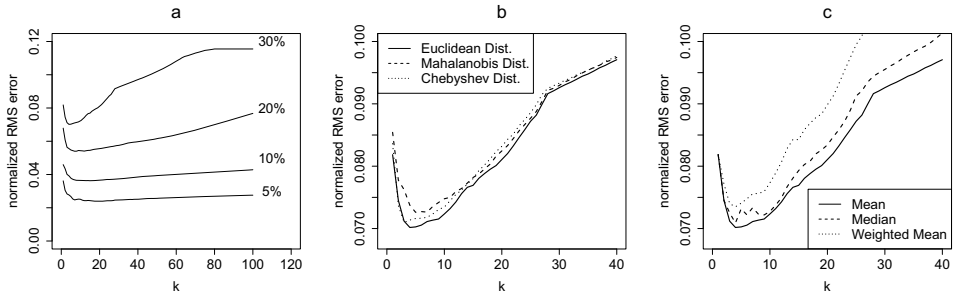

Figure 7: a) Normalized RMS error in dependence of k. The $k$nn method applied a) to data with different proportions of missing values, b) with different distance measures and c) with different missing values estimators.

with the Euclidean distance and the mean applied to data sets with different proportions of missing values. The error increases with increasing proportion of missing values and the minimum of the curves is between 5 and 20 neighbours. We compared also the performance of the difference measures (figure 7b) and

of the estimators (figure 7c). For both plots a data set with 30% of missing values was analyzed. Furthermore, for figure 7b the mean was used as missing values estimator and for figure 7c the Euclidean distance was used as distance measure. These plots show that using the Euclidean distance is slightly better than using the Mahalanobis or the Chebyshev distance and that the mean is a better estimator than the median or a weighted mean. We obtained the same results from plots with other combinations of difference measures and estimators. Compared to the row-mean method the $k$nn method results in smaller errors. The minima of the error-curves of figure 7a and the errors of the row-mean method are given in table 3. As further research activity it would also be of interest to compare the $k$nn method to other imputation methods given in Nguyen et al. ([14]). They use for example Partial Least Square (PLS) regression to impute missing values.

Table 3: Comparison of the normalized RMS error when using the row mean method and the $k$nn method, respectively.

proportion of missing values	5%	10%	20%	30%
row mean error	0.13	0.19	0.26	0.32
min ($k$nn error)	0.02	0.04	0.05	0.07

4. LONGITUDINAL DATA ANALYSIS

4.1. Analysis of variance

Before doing the statistical analysis we preprocessed the data by the vsn-method described in chapter 2. We also filled up the missing values by the $k$nn method described in chapter 3. The interest of the statistical analysis was to find those proteins for which the expression profiles over the time were different in the treated and untreated sample, respectively. In order to find differences in the temporal course of the treated and untreated samples we used an analysis of variance model for longitudinal data (cf. Jung et al. ([8]) and Diggle et al. ([5])). Such a model should take the time dependence of the measurements into account. Using F-tests one can detect time/treatment-interactions of spots. For our analysis we used only those spots for which at least three values were available at each time point. The detected significant spots are presented in table 4. The p-values in this table are not corrected for multiple testing (cf. Dudoit et al. ([6])), because the number n of spots is not clearly fixed in a 2-DE experiment. Biochemist often decide to exclude a great number of spots still after the statistical analysis.

Table 4: Spots with a time/treatment interaction.

rank	spot-no.	p-value
1	1136	0.0023
2	910	0.0055
3	988	0.0075
4	1669	0.0255
5	1054	0.0428

The expression profiles of the two most significant spots are plotted in figure 8. Both spots have a similar expression at the beginning of the experiment and the profiles drift at the end.

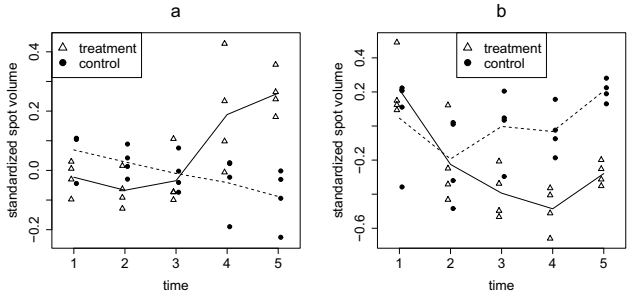

Figure 8: a) Temporal courses of the expression values of the spot numbers 910 (a) and 1136 (b) in the treated (solid line) and untreated (dashed line) samples.

4.2. Biological implications

This biological experiment was performed to identify candidate proteins contributing to neuroblastoma clinical outcome. We identified 5 proteins with a time/treatment-interaction upon addition of neurotrophin in SY5Y-TrkA cells (table 4). These proteins were identified using MALDI (matrix assisted laser desorption and ionization) mass spectrometry. For instance protein 910 with a changed temporal course after neurotrophin receptor activation consists to a family of heat shock proteins known to be involved in a number of cellular processes. Regarding cancer research the increased expression of heat shock protein 70 (Hsp 70) has been reported in a variety of tumor tissues. Hsp 70 has also been detected in plasma and therefore could potentially be used as a biomarker for diagnosis. It has been demonstrated, that patients suffering from prostate cancer have an increased level of Hsp 70 in the blood plasma (cf. Abe et al. ([1])). Based on this knowledge Hsp 70 could be a candidate tumor marker for neuroblastoma. To test this hypothesis further experiments have to be performed.

5. FUTURE CHALLENGES IN STATISTICAL PROTEOMICS

The statistical analysis of protein expression data is similar to the analysis of gene expression data from DNA microarrays. A future challenge for statisticians is the adaption of the methods for the analysis of gene expression data to be applicable to protein expression data. An important question of genomics was to find genes with differential expression in samples from different tissue types (cf. Jung et al. ([9])). Statistical tools for this purpose can also be applied to protein expression data from 2-DE when having estimated the missing values before. Of interest are also protein expression data from mass spectrometry (cf. Aebersold and Goodlett ([2]) and Pusch et al. ([15])). Statistical applications for those data span the whole range of multivariate methods like classification problems or multivariate outlier detection.

Furthermore, interactions between biomolecules are important in many important processes, such as cell proliferation and cell signalling. When pathogens (e.g. bacteria) attack our body, it responds by producing many antibodies. They bind to a part of pathogen, called antigen. Biochemists have studied how and where a given antibody binds to an antigen by investigation of a single point mutant of the antibody. Andersson ([4]) describes a different strategy for such mutation experiments. Instead of mutating each antibody at one position only, several modifications are made in the same antibody. Using statistical tools like Partial Least Squares regression he could find out which modification was relevant for establishing the binding. Also the investigation of the impact of environmental changes on the binding strength of an antibody-antigen interaction is important for antibodies used in diagnostic tests for cancer.

In both situations the binding properties of the interaction of biomolecules can be characterized by association and dissociation rates. These parameters can be measured by surface plasmon resonance detectors. New multivariate methods should be developed to analyze the relationships between these kinetic parameters and all the factors that influence these measures and to predict the kinetics of biomolecular interactions for new combinations of explanatory variables. There is also need for new statistical tools which allow the inclusion of structural and sequence information from nuclear magnetic resonance spectra and Fourier transform ion cyclotron resonance mass spectra for generating new biological and clinical knowledge. Such extensions of available methods could be of considerable importance in drug development for improving the binding of a drug to the desired target and for decreasing unwanted side reactions.

ACKNOWLEDGMENTS

Klaus Jung is supported by a predoctoral fellowship of the German Research Foundation (DFG). This work was also sponsored by the 'Optimization Fund' within the German National Genome Research Network (NGFN) and the Ministry of Innovation, Science and Research and Technology in North Rhine-Westfalia.

REFERENCES

[1] Abe, M.; Manola, J.B.; Oh, W.K.; Parslow, D.L.; George, D.J.; Austin, C.L. and Kantoff, P.W. (2004). Plasma levels of heat shock protein 70 in patients with prostate cancer: a potential biomarker for prostate cancer, *Clinical Prostate Cancer*, **3**, 49–53.

[2] Aebersold, R. and Goodlett, D.R. (2001). Mass Spectrometry in Proteomics, *Chemical Reviews*, **1**, 269–295.

[3] Amersham Biosciences (2003). *DeCyder Differential Analysis Software, Version 5.0, User Manual*, Amersham Biosciences, Sweden.

[4] Andersson, K. (2004). Characterization of Biomolecular Interactions Using a Multivariate Approach, *Acta Universitatis Upsaliensis, Comprehensive Summaries of Uppsala Dissertations from the Faculty of Medicine 1363*, Uppsala.

[5] Diggle, P.J.; Liang, K.Y. and Zeger, S.L. (1994). *Analysis of Longitudinal Data*, Clarendon Press, Oxford.

[6] Dudoit, S.; Shaffer, J.P. and Boldrick, J.C. (2003). Multiple hypothesis testing in microarray experiments, *Statistical Science*, **18**, 71–103.

[7] Huber, W.; Heydebreck, A. von; Sueltmann, H.; Poustka, A. and Vingron, M. (2002). Variance stabilization applied to microarray data calibration and to the quantification of differential expression, *Bioinformatics*, **18**, S96–S104.

[8] Jung, K.; Gannoun, A.; Stühler, K.; Sitek, B.; Meyer, H.E. and Urfer, W. (2005). Analysis of dynamic protein expression data, *RevStat-Statistical Journal*, **3**, 99–111.

[9] Jung, K.; Quast, K.; Gannoun, A. and Urfer, W. (2006). A renewed approach to the nonparametric analysis of replicated microarray experiments, *Biometrical Journal*, to appear.

[10] Karp, A.N.; Kreil, D.P. and Lilley, K.S. (2004). Determining a significant change in protein expression with DeCyder during a pairwise comparison and to the quantification of differential expression, *Proteomics*, **4**, 1421–1432.

[11] Klose, J. and Kobalz, U. (1995). Two-dimensional electrophoresis of proteins: an updated protocol and implications for a functional analysis of the genome, *Electrophoresis*, **16**, 1034–1059.

[12] Kreil, D.P.; Karp, A.N. and Lilley, K.S. (2004). DNA microarray normalization methods can remove bias from differential protein expression analysis of 2D difference gel electrophoresis results, *Bioinformatics*, **20**, 2026–2034.

[13] Nakagawara, A.; Azar, C.G.; Scavarda, N.J. and Brodeur, G.M. (1994). Expression and function of Trk-B and BDNF in human neuroblastomas, *Molecular and Cellular Biology*, **14**, 759–767.

[14] Nguyen, D.V.; Wang, N. and Carrol, R.J. (2004). Evaluation of missing value estimation for microarray data, *Journal of Data Science*, **2**, 347–370.

[15] Pusch, W.; Flocco, M.T.; Leung, S.-M.; Thiele, H. and Kostrzewa, M. (2003). Mass spectrometry-based clinical proteomics, *Pharmacogenomics*, **4**, 463–476.

[16] Sitek, B.; Apostolov, O.; Stühler, K.; Pfeiffer, K.; Meyer, H.E.; Eggert, A. and Schramm, A. (2005). Identification of dynamic proteome changes upon ligand-activation of Trk-receptors using two-dimensional fluorescence difference gel electrophoresis and mass spectrometry, *Molecular and Cellular Proteomics*, **4**, 291–299.

[17] Troyanskaya, O.; Cantor, M.; Sherlock, G.; Brown, P.; Hastie, T.; Tibshirani, R.; Botstein, D. and Altman, R.B. (2001). Missing value estimation methods for DNA microarrays, *Bioinformatics*, **17**, 6, 520–525.

REVSTAT – STATISTICAL JOURNAL

Background

Statistical Institute of Portugal (INE), well aware of how vital a statistical culture is in understanding most phenomena in the present-day world, and of its responsibility in disseminating statistical knowledge, started the publication of the scientific statistical journal *Revista de Estatística*, in Portuguese, publishing three times a year papers containing original research results, and application studies, namely in the economic, social and demographic fields.

In 1998 it was decided to publish papers also in English. This step has been taken to achieve a larger diffusion, and to encourage foreign contributors to submit their work.

At the time, the Editorial Board was mainly composed by Portuguese university professors, being now composed by national and international university professors, and this has been the first step aimed at changing the character of *Revista de Estatística* from a national to an international scientific journal.

In 2001, the *Revista de Estatística* published three volumes special issue containing extended abstracts of the invited contributed papers presented at the 23rd European Meeting of Statisticians.

The name of the Journal has been changed to REVSTAT – STATISTICAL JOURNAL, published in English, with a prestigious international editorial board, hoping to become one more place where scientists may feel proud of publishing their research results.

- The editorial policy will focus on publishing research articles at the highest level in the domains of Probability and Statistics with emphasis on the originality and importance of the research.
- All research articles will be refereed by at least two persons, one from the Editorial Board and another, external.
- The only working language allowed will be English.
- Two volumes are scheduled for publication, one in June and the other in November.
- On average, four articles will be published per issue.

Aims and Scope

The aim of REVSTAT is to publish articles of high scientific content, in English, developing innovative statistical scientific methods and introducing original research, grounded in substantive problems.

REVSTAT covers all branches of Probability and Statistics. Surveys of important areas of research in the field are also welcome.

Abstract/indexed in

REVSTAT is expected to be abstracted/indexed at least in *Current Index to Statistics*, *Mathematical Reviews*, *Statistical Theory and Method Abstracts*, and *Zentralblatt für Mathematic*.

Instructions to Authors, special-issue editors and publishers

Papers may be submitted in two different ways:

— By sending a paper copy to the Executive Editor and one copy to one of the two Editors or Associate Editors whose opinion the author(s) would like to be taken into account, together with a postscript or a PDF file of the paper to the e-mail: revstat@fc.ul.pt.

— By sending a paper copy to the Executive Editor, together with a postscript or a PDF file of the paper to the e-mail: revstat@fc.ul.pt.

Submission of a paper means that it contains original work that has not been nor is about to be published elsewhere in any form.

Submitted manuscripts (text, tables and figures) should be typed only in black, on one side, in double spacing, with a left margin of at least 3 cm and not have more than 30 pages.

The first page should include the name, affiliation and address of the author(s) and a short abstract with the maximum of 100 words, followed by the key words up to the limit of 6, and the AMS 2000 subject classification.

Authors are obliged to write the final version of accepted papers using LaTeX, in the REVSTAT style.

This style (REVSTAT.sty), and examples file (REVSTAT.tex), which may be download to *PC Windows System* (Zip format), *Mackintosh, Linux* and *Solaris Systems* (StuffIt format), and *Mackintosh System* (BinHex Format), are available in the REVSTAT link of the National Statistical Institute's Website: http://www.ine.pt/revstat.html

Additional information for the authors may be obtained in the above link.

Accepted papers

Authors of accepted papers are requested to provide the LaTeX files and also a postscript (PS) or an acrobat (PDF) file of the paper to the Secretary of REVSTAT: liliana.martins@ine.pt.

Such e-mail message should include the author(s)'s name, mentioning that it has been accepted by REVSTAT.

The authors should also mention if encapsulated postscript figure files were included, and submit electronics figures separately in .tiff, .gif, .eps or .ps format. Figures must be a minimum of 300 dpi.

Also send always the final paper version to:

Maria José Carrilho
Executive Editor, REVSTAT – STATISTICAL JOURNAL
Instituto Nacional de Estatística
Av. António José de Almeida
1000-043 LISBOA
PORTUGAL

Copyright and Reprints

Upon acceptance of an article, the author(s) will be asked to transfer copyright of the article to the publisher, the INE, in order to ensure the widest possible dissemination of information, namely through the National Statistical Institute's Website (http://www.ine.pt).

After assigning the transfer copyright form, authors may use their own material in other publications provided that the REVSTAT is acknowledged as the original place of publication. The Executive Editor of the Journal must be notified in writing in advance.

Authors of articles published in the REVSTAT will be entitled to one free copy of the respective issue of the Journal and twenty-five reprints of the paper are provided free. Additional reprints may be ordered at expenses of the author(s), and prior to publication.