Correspondence Analysis and Data Warehousing with Microarray Data

Practical, firstly held Sat 7-12-02, all links checked 23-6-04

Kurt Fellenberg, Christian Busold

Contents

Data Warehousing

Data analysis is intimately linked with data storage. Terms like 'data warehousing' reflect this. Data warehousing aims at providing data in a format suitable for analysis. In a classical data warehouse solution, data are held in one or several so-called 'operational' databases. A warehouse then collects data from these databases mainly used for storage and makes them fit into a unified data model [1, 5]. Typically, a warehouse will collect only a few, 'important' attributes from each dataset. Operations such as extractions and transformations are recorded as meta data. Its structure may be denormalized, i.e. it allows for redundancy in order to avoid frequent joining from distinct tables. A warehouse is designed to assist analytical tasks rather than pure data storage, integrating different data sources and data formats to gain unified access for analysis algorithms.

Handling a DBMS

PostgreSQL - like Oracle and other DBMS but unlike e.g. MySQL - is a transaction based DBMS. A transaction-based DBMSs is capable of the administration of more than one version of a database at the same time to protect integrity of the stored data by transactions. Transactions give databases an all-or-nothing capability when making modifications. A transaction can comprise one or multiple queries with every of the performed changes becoming valid upon successful execution of the whole transaction and none of them in case of an error. At the same time all other users are insulated from seeing the partially committed transaction until the very moment of commitment, preventing database consistency from being damaged by simultaneous write access. Although transaction-based database management slows down access performance, we recommend to use a transaction based DBMS.

Task:

Create some tables and get used to altering, filling and querying them.

Please find some basic SQL commands at http://www.postgresql.org/docs/aw_pgsql_book/node22.html

Designing a DB structure

Task:

Prepare a structure that is to take one or several microarray experiments.

An experiment may comprise several experimental conditions, each recorded by several repeatedly performed hybrdizations. Please keep in mind, that the intensities should be annotated both by information about the genes and about experimental procedures and sample biology. Try to comply to the Minimal Information About Microarray Experiments (MIAME) standard. Include GO IDs linking to standardized information about the gene products.

Surfing for information

Question:

What is MIAME? Why should one comply with MIAME (benefits)?
Does it solve all the problems (drawbacks, limitations)?

Question:

What actually is an ontology?:-)
What is GO? Why should one use GO (benefits)?
Does it solve all the problems (drawbacks, limitations)?
What relationships are modeled?
Assess the GO DB structure in terms of flexibility and statistical accessibility.

Question:

Check UML structure representations (if not available the attribute listings) of several well-known microarray data repositories for biological or protocol-related keywords (table- or attribute names). Put it in context to what we've learned about BD implementation "by structure" and "by content".

Question:

Which is the best known public European microarray data repository?
How many hybridizations does it contain?

Analysis

Own implementation

Task:

Choose Java, Perl, Matlab, or other programming language you feel comfortable with. If in doubt, please use Java.
Please change seats to work together in groups each consisting of one member who is familiar with a programming language and non-experienced members.

Task:

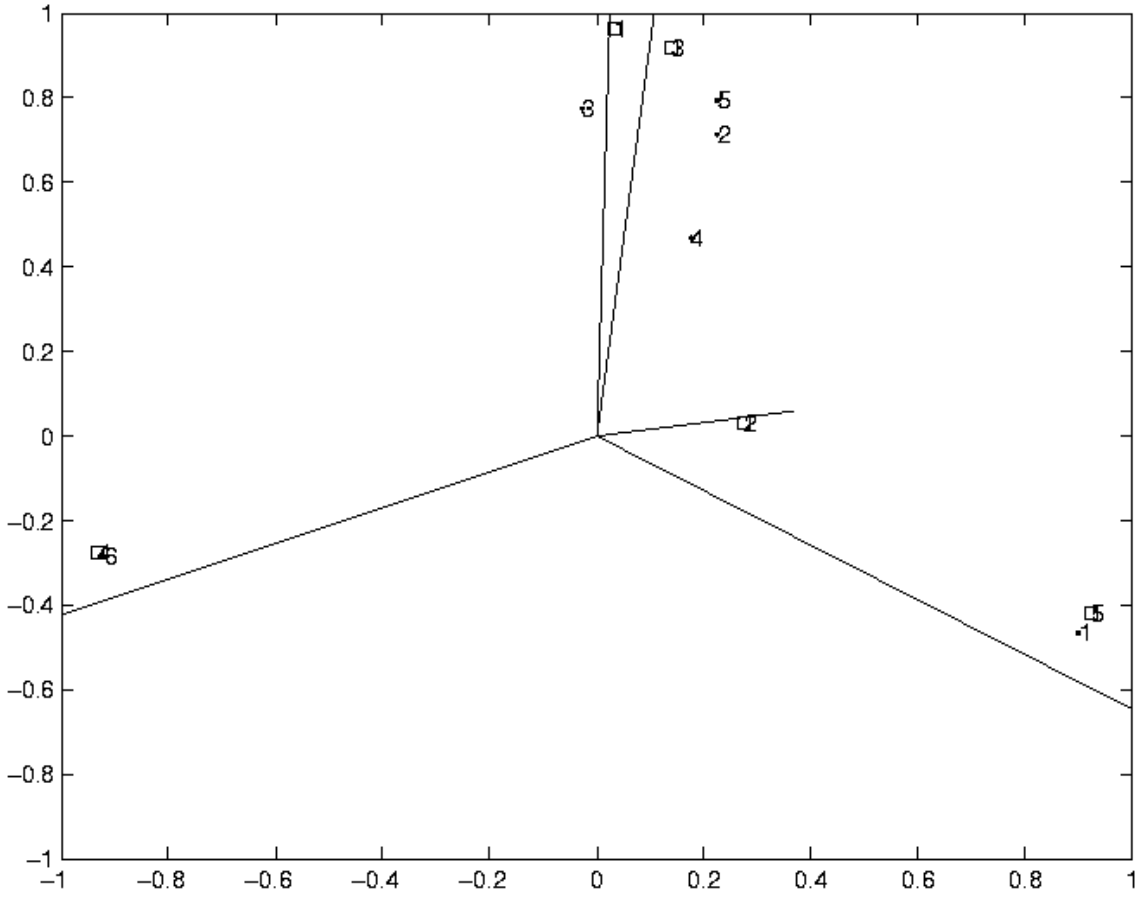
Import normalized signal intensities (representing at least 3 experimental conditions) from your DB into your program.

Task:

Implement a basic CA (see pp. 10-12). Use the following small reference matrix for verification:

2	16	4	4	65
5	4	8	3	5
9	4	6	6	3
2	5	3	2	2
6	3	13	4	7
4	9	5	84	3

(at http://www.m-chips.org/akad_fuer_weiterbildung/m)



Task:

Try to run your implementation on the imported data set.

Task:

Expand your implementation by standard-coordinate representations of the hybridizations (preferably plotted as lines).

Task:

... by HMS.

Task:

... by a bar-plot representing the shares of total variance explained by the axes of the plot (and the axes not being plotted).

Task:

Compute frequencies of either gene- or experiment annotation values to characterize gene or experiment clusters in your dataset.

Data analysis (without hacking ;-)

Task:

Connect to dkfz as testex. Make sure that you see graphics by typing 'xclock' (use <Ctrl>-c to escape). Type 'matlab', then 'mchips'. Now you can browse the database "eurofan" (see http://www.dkfz-heidelberg.de/tbi/services/mchips/#public, first dataset). Please find information about how to do this at http://www.dkfz.de/tbi/services/mchips/mchips_help.html under "Select a dataset".

Task:

What was done in experiment 29 ("heatshock_timecourse")?

Task:

Retrieve experiment 29, normalize, filter by minmax-separation (threshold 0) according to above manual.

Task:

Perform CA type E. Is there any need to look at the third dimension? How much of the total variance in the dataset does it explain/visualize? Are there experimental conditions, the difference of which is visible in the 3rd dimension only?

Task:

Quit Matlab ('quit'), type 'setenv MATLABDB yeast2', type 'cd Praktikum', type 'netscape &', reenter Matlab ('matlab').
Type 'mchips', retrieve experiment 1 ("CDC14_induction") or
alternatively (faster) type 'load cdc14_complete'.

Task:

Normalize , discard hybs with correlation coefficant below 0.2 (i.e. none), filter by minmax-separation (threshold 0), perform CA type D.

Task:

Are there any hybridizations that behave strange, i.e. that do have an unexpected position? Which?
Repeat analysis without them (either by browsing/selecting or 'clear; load cdc14'). Use first CA type D, then E.

Task:

Identify genes showing high association with transgene+Gal and other conditions.

Discussion

Question:

Implementing an analysis system by first designing a DB structure to get data stored before setting up analysis algorithms will produce other results than doing it vice versa. Why? Put it in context to DB / data warehouse.

Question:

Do you think a European microarray data repository should be an operational DB or more like a datawarehouse (serving data analysis)? If the latter, would you think it is feasible?

More information / references

In the theory session, the question came up how to actually grasp the CA procedure, especially the projection of columns and rows into the very same subspace. Here is a nice book about that: Greenacre, M. J. (1993) *Correspondence Analysis in Practice* (Academic, London). There are also much shorter books such as Clausen, S. E. (1998) *Applied Correspondence Analysis* (SAGE, London) but I find them less intuitive and harder to read. Please find more books incl. more 'theoretical' ones referenced in the CA paper.

Webrecources

Microarray-related:

- About Microarray Exeriments
  http://www.bio.davidson.edu/courses/genomics/chip/chip.html !:-)
  http://www.ebi.ac.uk/microarray/biology_intro.html
- MIAME:
  http://www.mged.org/Workgroups/MIAME/miame.html
- Literature:
  http://linkage.rockefeller.edu/wli/microarray
- Databases:
  http://ihome.cuhk.edu.hk/~b400559/arraysoft_public.html

  http://www.ebi.ac.uk/microarray/ArrayExpress/arrayexpress.html
  http://www.ncbi.nlm.nih.gov/geo/
  http://genome-www5.stanford.edu/MicroArray/SMD/
  http://www.ncgr.org/research/genex/
  http://www.cbil.upenn.edu/RAD2/
  http://arep.med.harvard.edu/ExpressDB/
  http://bioinf.man.ac.uk/microarray/maxd/maxdSQL/index.html

http://genome.nhgri.nih.gov/arraydb/
http://staffa.wi.mit.edu/chipdb/public/index.html
http://www.microarrays.org/AMADFaq.html

- Software:
  http:www.nslij-genetics.org/microarray/soft.html

  http://www.stat.uni-muenchen.de/~strimmer/rexpress.html
  http://www.ii.uib.no/~bjarted/jexpress/index.html
  http://www.microarrays.org/software.html

- Groups:
  http://www.stat.Berkeley.EDU/users/terry/zarray/Html/index.html
  http://www-stat.stanford.edu/~tibs/lab/index.html
  http://research.nhgri.nih.gov/microarray/index.html
  http://derisilab.ucsf.edu/
  http://aretha.jax.org/jax-cgi/churchill/index.cgi

- Public Data:
  http://www.nslij-genetics.org/microarray/data.html
  http://genome-www.stanford.edu/cellcycle/data/rawdata/

Others:

- Gene Databases
  http://www.geneontology.org/
  http://www.mpiem.gwdg.de/Forschung/Biol/database.html

- Ontology
  http://mged.sourceforge.net/ontologies/index.php
  http://www.cs.man.ac.uk/~stevensr/ontology.html

- Info on CA
  http://www.statsoftinc.com/textbook/stcoran.html

- Info on other Methods
  http://odur.let.rug.nl/~kleiweg/kohonen/kohonen.html
  http://www.cis.hut.fi/aapo/papers/IJCNN99_tutorialweb/
  http://www.cis.hut.fi/projects/ica/fastica/index.shtml
  http://www.cs.huji.ac.il/labs/compbio/expression/

- Hacking
  http://stein.cshl.org/WWW/software/GD (Perl graphics)
  http://www.R-project.org/ (R)
  http://www.mathworks.com/access/helpdesk/help/techdoc/matlab.shtml (Matlab)
  http://developer.netscape.com/docs/manuals/
  http://www.cryst.bbk.ac.uk/classlib/BTL2/
  http://www.mysql.com/information/crash-me.php (compare different DBMSs)
  http://www.postgresql.org/docs/awbook.html (PostgreSQL)
  http://www.postgresql.org/users-lounge/index.html (PostgreSQL)
  http://www.oracle.com/ip/deploy/database/oracle9i/index.html (Oracle)
  http://home.netscape.com/eng/mozilla/3.0/handbook/javascript/ (Javascript)
  http://java.sun.com/products/archive/jdk/1.1/index.html (Java)
  http://math.nist.gov/javanumerics/

Our stuff is at `http://www.m-chips.org`.

CA HOWTO

Correspondence Analysis

I provide here a concise summary of the technique, see refs. [3] and [4] for a thorough exposition. An informal, intuitive description will be given below. The aim is to embed both rows (genes) and columns (hybridizations) of a matrix in the same space, the first two or three coordinates of which contain most of the information. Let I genes and J hybridizations be collected into the $I \times J$ matrix $\mathbf{N}$ with elements n_{ij} . Let n_{i+} and n_{+j} denote the sum of the ith row and jth column, respectively. By n_{++} I denote the grand total of $\mathbf{N}$. The mass of the jth column is defined as $c_j = n_{+j}/n_{++}$, and likewise the mass of the ith row is $r_i = n_{i+}/n_{++}$. Basis for the calculation is the *correspondence matrix* $\mathbf{P}$ with elements $p_{ij} = n_{ij}/n_{++}$ from which the matrix $\mathbf{S}$ with elements $s_{ij} = (p_{ij} - r_i c_j)/\sqrt{r_i c_j}$ is derived. $\mathbf{S}$ is submitted to singular value decomposition [2], i.e. it is decomposed into the product of three matrices: $\mathbf{S}$=$\mathbf{U}\Lambda\mathbf{V}^T$. Λ is a diagonal matrix, and its diagonal elements are referred to as the singular values of $\mathbf{S}$. I think of them as sorted from the largest to the smallest and denote them by λ_k. The coordinates for gene i in the new space are then given by $f_{ik} = \lambda_k u_{ik}/\sqrt{r_i}$, for $k = 1, ..., J$. Hybridizations are viewed in the same space with hybridization j given coordinates $g_{jk} = \lambda_k v_{jk}/\sqrt{c_j}$, for $k = 1, ..., J$. These coordinates are called principal coordinates.

To reduce dimensionality, only the first two or three coordinates of the new space are plotted. The loss of information associated with this dimension reduction is quantified in terms of the

proportion of the so-called total inertia $\sum_k \lambda_k^2$ that is explained by the axis displayed. Total inertia is proportional to the value of the χ^2 statistic, and thus the amount of information represented in, e.g., a planar embedding $(\lambda_1^2 + \lambda_2^2)/\sum_k \lambda_k^2$, corresponds to the proportion of the χ^2 statistic explained by the embedding.

The above summary is aimed to provide all the information needed to implement a simple CA algorithm. This can be easily done by using nested *for* loops. A much shorter implementation without loops can be achieved in any programming language supporting matrix multiplication and providing a routine for singular value decomposition, e.g. in MATLAB (Appendix B).

Standard coordinates as an aid in visualization

Correspondence analysis attempts to separate dissimilar objects (genes or hybridizations) from each other; similar objects are clustered together resulting in small distances. In contrast, the distance between a gene and a hybridization cannot be directly interpreted. For visualization of between-variable association in the plot one includes virtual genes which have all their intensity focused in one hybridization [4]. The coordinates of such a gene are called standard coordinates of the hybridization where this gene is expressed. Likewise, one could introduce standard coordinates for genes. The standard coordinates for the genes are computed as $u_{ik}/\sqrt{r_i}$ and for the hybridizations as $v_{jk}/\sqrt{c_j}$. In practice, the spread of the set of real genes and hybridizations is much smaller than the spread introduced when including these virtual genes and hybridizations via their standard coordinates. As a consequence, the real points would shrink to a tiny area, so I rather depict the direction from the centroid of the data to the standard coordinates instead of the standard coordinates themselves.

Medians and replicate hybridizations in correspondence analysis

Typically, replicate hybridizations are performed for each condition under study leading to several values for one gene/condition pair. The number of such repeated hybridizations is often small. I therefore represent these values by their gene-wise median rather than their gene-wise average because the median is less sensitive to outliers. The need remains, though, to visualize also the original data and not only the median since they contain valuable information about experimental variance and quality of individual hybridizations. In fact, CA offers the possibility to reflect both aspects. To this end, CA is first effected by using the gene-wise medians, determining the coordinate system to embed the original hybridization intensities. These data points are then referred to as supplementary points or points without mass. Thus the share of noise belonging to an experimental condition is shown by the spread of its hybridizations around the median. As the dimensions of the data are reduced by using medians

of hybridizations per experimental condition, I refer to this strategy as hybridization-median determined scaling (HMS).

The embedding for hybridizations without mass is computed as follows. Let the matrix $\mathbf{N}$ contain only the hybridization medians and let $\mathbf{N}^\star$ of elements $n^\star_{ij'}$ be the original data matrix containing all the hybridizations. $\mathbf{N}$ is submitted to CA. Let $\mathbf{P}^\star$ have elements $p^\star_{ij'} = n^\star_{ij'}/n^\star_{++}$. The principal coordinates for the supplementary hybridizations from correspondence matrix $\mathbf{P}^\star$ are then calculated as

$$g^\star_{j'k} = \frac{1}{\sum_i p^\star_{ij'}} \sum_i \frac{p^\star_{ij'} f_{ik}}{\lambda_k}.$$

In our own data sets, a single hybridization consists of two corresponding spot sets because each cDNA had been spotted twice on the array. I refer to these spot sets as *primary* and *secondary spots*. They tend to show a higher correlation than hybridizations belonging to the same experimental condition. Plotting them separately (duplicating the number of supplementary points) provides an atomic unit of distance in the biplot, where no units are assigned to the axes. The intensity unit cancels out when calculating the correspondence matrix $\mathbf{P}$.

Bibliography

[1] C. Ballard, D. Herreman, D. Schau, R. Bell, E. Kim, and A. Valencic. *Data modeling techniques for data warehousing*. IBM International Technical Support Organization, www.redbooks.ibm.com, San Jose, CA, 1998.

[2] G. H. Golub and C. Reinsch. Singular value decomposition and least squares solutions. *Numer. Math.*, 14:403–420, 1970.

[3] M. J. Greenacre. *Theory and Applications of Correspondence Analysis*, page 223. Academic Press, London, 1st edition, 1984.

[4] M. J. Greenacre. *Correspondence Analysis in Practice*, pages 181–183 and 36. Academic Press, London, 1st edition, 1993.

[5] C. Schönbach, P. Kowalski-Saunders, and V. Brusic. Data warehousing in molecular biology. *Briefings in Bioinformatics*, 1:190–198, 2000.