Global testing of differential gene expression

Ulrich Mansmann Reinhard Meister Manuela Hummel

October 12, 2005

Contents

1 Abstract

In studies about differential gene expression between different clinical diagnoses the main interest may often not be in single genes but rather in groups of genes that are associated with a pathway or have a common location in the genome. In such cases it may be better to perform a global test because the problems of multiple testing can be avoided. The approach presented here is an ANCOVA global test on phenotype main effect and gene–phenotype interaction.

Testing many pathways simultaneously is also possible. This, of course, causes again need for correction for multiple testing. Besides the standard approaches for correction we introduce a closed testing procedure in which the experiment–wise error rate equals the required level of confidence of the overall test.

This document was created using R version 2.2.0 and versions 1.4.0 and 3.2.0 of the packages *GlobalAncova* and *globaltest* respectively.

2 Introduction

The ANCOVA global test is a simultaneous test on phenotype main effect and gene–phenotype interaction in a two–way layout linear model. If the mean expression level for at least one gene differs between groups the global null hypothesis, which is the intersection of all null hypotheses for the single genes, is violated. As our test is based on the sum of gene-wise reduction in sum of squares due to phenotype, all systematic differences in gene expression between phenotypes equally contribute to the power of the test.

Single genes are not, in general, the primary focus of gene expression experiments. The researcher might be more interested in relevant pathways, functional sets or genomic regions consisting of several genes. Most of the current methods for studying pathways analyse differential expression of single genes. In these methods pathways where many genes show minor changes in their expression values may not be identified. Goeman's global test and the ANCOVA global test were designed to address this issue.

Applying global tests for differential expression in pathways substantially reduces the number of tests compared to gene-wise multiple testing. The amount of correction for multiple testing decreases. Function (KEGG, GO) or location (chromosome, cytoband) could be used as grouping criteria, for example.

We want to compare our method with the global test of Goeman et al., 2004 [1]. Therefore text and examples in this document follow to a certain extent the vignette presented in the R-package *globaltest*. Our function `GlobalAncova` tests whether the expectation of expression levels differs between two biological entities for a given group of genes. This vignette has its focus on the practical use of the test. For more details about the mathematical background and the interpretation of results, we refer to the paper by Mansmann and Meister, 2005 [3].

This document shows the functionality of the R-package *GlobalAncova*. The datasets, all necessary R–packages and our package *GlobalAncova* are available from the Bioconductor website (*www.bioconductor.org*).

First we load the packages and data we will use.

```
> library(GlobalAncova)
> library(globaltest)
> library(golubEsets)
> library(hu6800)
> library(vsn)
> library(multtest)
> data(golubMerge)
> golubM <- update2MIAME(golubMerge)
> golubX <- vsn(golubM)
```

This creates a dataset `golubX`, which is of the format *exprSet*, the standard format for gene expression data in BioConductor. It consists of 7129 genes and 72 samples. We used *vsn* to normalize the data. Any other normalization method may be used instead. From several phenotype variables we use "ALL.AML" as the clinical diagnoses of interest. ALL and AML are two types of acute leukemia. There are 47 patients with ALL and 25 with AML.

3 Global Testing of a Single Pathway

3.1 Cell Cycle Pathway

Suppose we are interested in testing whether AML and ALL have different gene expression patterns for certain pathways, for example from the KEGG database.

All Genes
We start by applying our test to all genes in the Golub dataset so that differences in the overall gene-expression pattern can be demonstrated.

```
> gr <- as.numeric(golubX$ALL.AML == "ALL")
> ga.all <- GlobalAncova(xx = exprs(golubX), group = gr,
+     covars = NULL, perm = 100, test.genes = NULL)
```

The first input *xx* is a 7129 $\times$ 72 matrix that contains the expression values of all genes and samples. The second input *group* is a vector that defines the clinical diagnosis for the 72 patients. It must be coded as 0–1. More than two clinical groups or even continuous phenotype coding might be considered in future versions of the package.

To avoid alpha–inflation due to correlated data and effects of non–normality of the data tests for significance of the resulting F–ratios are performed using a permutation test approach. The argument *perm* defines the number of permutations. The number of permutations is 10,000 for default but in the current version the test for many genes can take quite a long time with 10,000 permutations. Here we set *perm* to just 100 or 1000 so that creating this vignette will not last too long. For getting more reliable results one should recompute the examples with more permutations. We are currently looking for a reliable asymptotic test as a chance to get around with too time consuming permutations.

The result is a typical ANOVA table with information only for the interaction term of interest. Besides the classical F–test p–values, there are also p–values from the permutation test.

```
> ga.all
```

```
$ANOVA.table
                         SS     DF          MS
Total          1142401.25 513287   2.2256579
Genes adjusted  945036.99   7128 132.5809463
GroupXGenes      14800.04   7129   2.0760336
Residual        182564.22 499030   0.3658382

$test.result.GroupXGenes

F.value      5.674732
p.value.perm 0.000000
p.value.theo 0.000000
```

From this result we conclude that the overall gene expression profile for all 7129 genes is associated with the clinical outcome. This means that samples with different AML/ALL status tend to have different expression profiles. We expect most pathways (especially the ones containing many genes) also to be associated with the phenotype groups.

If we apply Goeman's global test we get

```
> gt.all <- globaltest(golubX, "ALL.AML")
> gt.all

Global Test result:
Data: 72 samples with 7129 genes; 1 pathway tested
Model: logistic

  genes tested Statistic Q Expected Q sd of Q    p-value
1  7129   7129      53.992         10  1.9035 5.1616e-35
```

Both tests show that the data contain overwhelming evidence for differential gene expression between AML and ALL.

Cell Cycle Pathway

Now we ask the more specific question of whether there is evidence for differential gene expression between both diagnoses restricted to genes belonging to the cell cycle pathway. First we load all KEGG pathways.

```
> kegg <- as.list(hu6800PATH2PROBE)
```

The list `kegg` consists of 153 pathways. Each pathway is represented by a vector of gene names. We are mainly interested in the cell cycle pathway which has the identifier "04110" in the KEGG database. It corresponds to 92 probe sets on the hu6800 chip.

```
> cellcycle <- kegg[["04110"]]
```

We apply the global test to this pathway using the option *test.genes*.

```
> ga.cc <- GlobalAncova(exprs(golubX), gr, test.genes = cellcycle,
+     perm = 1000)
> ga.cc
```

```
$ANOVA.table
                        SS   DF         MS
Total          10688.1921 6623  1.6137992
Genes adjusted  8091.5805   91 88.9184671
GroupXGenes      244.1049   92  2.6533146
Residual        2352.5066 6440  0.3652961

$test.result.GroupXGenes

F.value      7.263464
p.value.perm 0.000000
p.value.theo 0.000000
```

Also with *globaltest* we get a very small p–value

```
> gt.cc <- globaltest(golubX, "ALL.AML", cellcycle)
> gt.cc

Global Test result:
Data: 72 samples with 7129 genes; 1 pathway tested
Model: logistic

  genes tested Statistic Q Expected Q sd of Q    p-value
1    92     92      69.005     10.195  3.2811 1.4669e-18
```

The test results clearly indicate that the expression pattern of the cell cycle pathway is different between the two clinical groups.

Adjusting for Covariates

Covariate information is incorporated by specifying the *covars* option. For example if we want to adjust for `Source`, the institution that provided the samples, we can do this by

```
> ga.cc.source <- GlobalAncova(exprs(golubX), gr, covars = golubX$Source,
+     test.genes = cellcycle, perm = 1000)
> ga.cc.source

$ANOVA.table
                        SS   DF         MS
Total          10688.1921 6623  1.6137992
Genes adjusted  8293.5090  183 45.3197210
GroupXGenes      121.3187   92  1.3186816
Residual        2273.3644 6348  0.3581229

$test.result.GroupXGenes

F.value      3.682204
p.value.perm 0.000000
p.value.theo 0.000000
```

The source of the samples apparently has some explanatory effect on the outcome resulting in a smaller F–ratio than in the model without adjusting. But the influence of the genes is still highly significant.

With the *globaltest* we get a higher p–value.

```
> gt.cc.source <- globaltest(golubX, ALL.AML ~ Source,
+     cellcycle)
> gt.cc.source

Global Test result:
Data: 72 samples with 7129 genes; 1 pathway tested
Model: logistic, ALL.AML ~ Source
Adjusted: 11.5 % of variance of Y remains after adjustment

  genes tested Statistic Q Expected Q sd of Q  p-value
1    92     92      17.983     10.924  3.7995 0.047077
```

Permutation based p–values can also be obtained with Goeman's test, however only when covariates are absent.

3.2 p53–Signalling Pathway

```
> data(p53.signalling)
> data(group.info)
> data(cov.info)
```

We present another example from a study on different stages of colon cancer. The data is available with the *GlobalAncova* package. This example illustrates the role of covariates in the context of global testing in more detail. The tumour suppressor protein p53 contributes as a transcription factor to cell cycle arrest and apoptosis induction. Therefore, the p53-signalling pathway was selected as a candidate, where differential expression between two relevant prognostic groups defined by UICC II and UICC III stage of colon carcinoma probes was expected. The dataset `p53.signalling` contains 45 genes of the pathway that are present on the Affymetrix chip HU133a for 36 samples, 18 for each stage of cancer. The group information for each sample is stored in `group.info`. In `cov.info` there is also information about the gender of the patients and the location of the tumors.

```
> data(p53.signalling)
> data(group.info)
> data(cov.info)
```

Adjusting for Covariates
First we compute the Global Ancova without adjusting for covariates and get a significant result.

```
> set.seed(123)
> ga.table.1 <- GlobalAncova(xx = p53.signalling, group = group.info,
+     perm = 1000)
> ga.table.1

$ANOVA.table
                       SS   DF          MS
Total          4740.37616 1619   2.9279655
Genes adjusted 4502.83829   44 102.3372338
```

```
GroupXGenes      14.72441   45   0.3272091
Residual        222.81346 1530   0.1456297

$test.result.GroupXGenes

F.value      2.246857e+00
p.value.perm 5.000000e-03
p.value.theo 5.778971e-06
```

Including the covariates improves the separation of expression values by UICC stage.

```
> ga.table.2 <- GlobalAncova(p53.signalling, group.info,
+     covars = cov.info, perm = 1000)
> ga.table.2

$ANOVA.table
                        SS   DF         MS
Total          4740.37616 1619  2.9279655
Genes adjusted 4516.40461  134 33.7045120
GroupXGenes      18.07929   45  0.4017619
Residual        205.89226 1440  0.1429807

$test.result.GroupXGenes

F.value      2.809902e+00
p.value.perm 1.000000e-03
p.value.theo 3.608313e-09
```

The test results illustrate that the theoretical p–values are probably over–optimistic.

The *globaltest* also reveals better separation by including covariate information. Regarding the p–values this test is more optimistic here.

```
> colnames(p53.signalling) <- seq(1:dim(p53.signalling)[2])
> names(group.info) <- colnames(p53.signalling)
> gt.table.1 <- globaltest(p53.signalling, group.info)
> gt.table.1

Global Test result:
Data: 36 samples with 45 genes; 1 pathway tested
Model: logistic

  genes tested Statistic Q Expected Q sd of Q   p-value
1    45     45      22.316         10   3.965 0.0069788

> gt.table.2 <- globaltest(p53.signalling, group.info,
+     adjust = as.data.frame(cov.info))
> gt.table.2

Global Test result:
Data: 36 samples with 45 genes; 1 pathway tested
```

```
Model: logistic, Y ~ sex + loc
Adjusted: 87 % of variance of Y remains after adjustment

  genes tested Statistic Q Expected Q sd of Q    p-value
1    45     45       28.71         10  4.1803 0.00073116
```

Sex or Location of the Tumor as Phenotype
In contrast to the results above using stage as the clinical outcome if we had used sex or location as phenotype there would be no evidence of differential gene expression between the respective groups. For Goeman's test we get again similar results with slightly smaller p–values.

```
> ga.table.sex <- GlobalAncova(p53.signalling, group = cov.info[,
+     "sex"], perm = 1000)
> ga.table.loc <- GlobalAncova(p53.signalling, group = cov.info[,
+     "loc"], perm = 1000)
> gt.table.sex <- globaltest(p53.signalling, cov.info[, "sex"])
> gt.table.loc <- globaltest(p53.signalling, cov.info[, "loc"])

> ga.table.sex

$ANOVA.table
                         SS   DF          MS
Total          4740.376159 1619   2.9279655
Genes adjusted 4502.838286   44 102.3372338
GroupXGenes       8.977898   45   0.1995089
Residual        228.559975 1530   0.1493856

$test.result.GroupXGenes

F.value      1.33552929
p.value.perm 0.18400000
p.value.theo 0.06917166

> ga.table.loc

$ANOVA.table
                         SS   DF          MS
Total          4740.376159 1619   2.9279655
Genes adjusted 4502.838286   44 102.3372338
GroupXGenes       5.740071   45   0.1275571
Residual        231.797802 1530   0.1515018

$test.result.GroupXGenes

F.value      0.8419512
p.value.perm 0.6040000
p.value.theo 0.7625854

> gt.table.sex
```

```
Global Test result:
Data: 36 samples with 45 genes; 1 pathway tested
Model: logistic

  genes tested Statistic Q Expected Q sd of Q p-value
1    45     45      13.606         10   3.965 0.17162
```

```
> gt.table.loc
```

```
Global Test result:
Data: 36 samples with 45 genes; 1 pathway tested
Model: logistic

  genes tested Statistic Q Expected Q sd of Q p-value
1    45     45      8.6994         10  4.0429 0.57929
```

4 Testing Several Pathways Simultaneously

Systems biology involves the study of mechanisms underlying complex biological processes as integrated systems of many diverse interacting components, often referred to as pathways.

We regard the possibility to investigate differential gene expression simultaneously for several of those pathways as a contribution towards understanding biological relevant relations.

The user can apply `GlobalAncova` to compute p–values for a couple of pathways with one call by specifying the *test.genes* option. The members of each pathway to be tested must belong to genes in the expression–matrix. Afterwards a suitable correction for multiple testing has to be applied. An alternative based on the closed testing approach is described later.

Suppose for example that we want to test the first five KEGG pathways with the Golub data. We proceed as follows.

```
> ga.kegg <- GlobalAncova(exprs(golubX), gr, test.genes = kegg[1:5],
+     perm = 1000)
> ga.kegg

      genes F.value p.value.perm p.value.theo
00970    23  4.4771        0.000       0.0000
00100    13  7.9829        0.000       0.0000
05020    12  1.2310        0.262       0.2565
01510    50  2.4545        0.001       0.0000
00430     8  6.3840        0.000       0.0000
```

The result is a matrix whose rows correspond to the KEGG pathways. Note that if a pathway consists of a single gene a squared t–statistic which is equivalent to a F–statistic is computed. Also in this case a permutation test is performed.

With the `globaltest` we get a similar matrix.

```
> gt.kegg <- globaltest(golubX, "ALL.AML", kegg[1:5])
```

```
> gt.kegg

Global Test result:
Data: 72 samples with 7129 genes; 5 pathways tested
Model: logistic

      genes tested Statistic Q Expected Q sd of Q    p-value
00970    23     23     40.5300     9.3642  5.9616 6.5427e-04
00100    13     13     76.0750    10.3220  6.2824 2.5998e-07
05020    12     12      9.3932     7.5489  3.5372 2.6168e-01
01510    50     50     21.4750     8.8044  2.6204 1.5510e-04
00430     8      8     42.4510     7.0544  5.6346 3.4005e-04
```

This test also works for a single gene.

4.1 Simultaneous Adjustment of p–values

Next we show how to extract p–values for correction for multiple testing. Note however that due to the extremely high correlations between these tests, many procedures that correct for multiple testing here are inappropriate. An appropriate way of adjusting would be for example the method of Holm, 1979 [2]. An alternative to such adjustments that is not affected by correlations between tests is a closed testing procedure. For this approach you need a family of null hypotheses that is closed under intersection. Then a single hypothesis can be rejected at level α if it is rejected along with all hypotheses included in it (Marcus et al. 1976).

For the adjustment according to Bonferroni and Holm we build a vector of the raw p–values. The function `mt.rawp2adjp` provides several adjusting methods. We here display only the raw and "Holm" adjusted p–values. To obtain the original order of the pathways we order the result of `mt.rawp2adjp` according to `index`.

```
> ga.kegg.raw <- ga.kegg[1:5, 3]
> ga.kegg.adj <- mt.rawp2adjp(ga.kegg.raw)
> ga.kegg.adj$adjp[order(ga.kegg.adj$index), c("rawp", "Holm")]

      rawp  Holm
[1,] 0.000 0.000
[2,] 0.000 0.000
[3,] 0.262 0.262
[4,] 0.001 0.002
[5,] 0.000 0.000
```

Besides pathway "05020" all other pathways are significant.

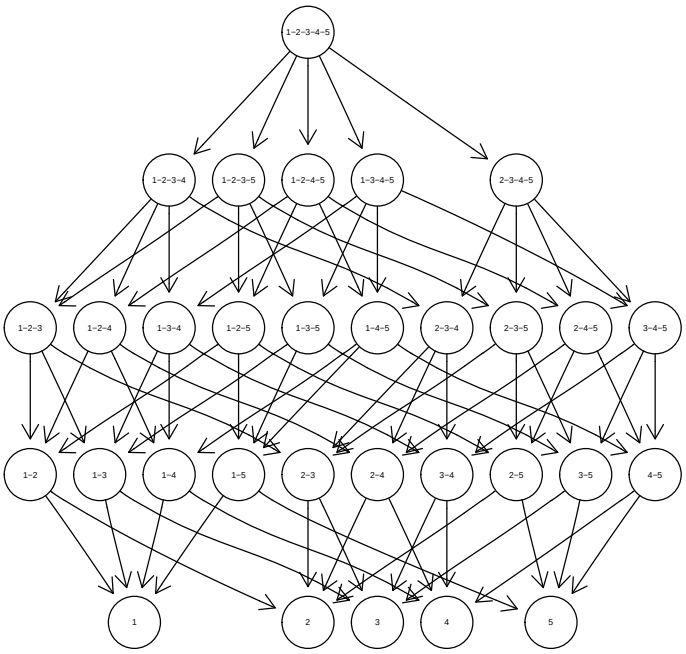
4.2 Closed Testing Procedure

Closed testing procedures (Marcus et al., 1976 [4]) offer a versatile and powerful approach to the multiple testing problem. Implementation is non–trivial, therefore, the program given in this version should be regarded as a prototype.

In order to apply the closed testing procedure we first have to create the required family of hypotheses by building all intersections between the four "natural" hypotheses tested above and all intersections of those new hypotheses and so on.

The resulting family of hypotheses can be illustrated in a directed graph. The node "1-2-3-4-5" for example stands for the global hypothesis that the genes of all five selected pathways are not differentially expressed. Now the interesting hypothesis "1" for example can be rejected if also the hypotheses "1-2-3-4-5", "1-2-3-4", ..., "1-3-4-5", "1-2-3", ..., "1-4-5", "1-2", ..., "1-5" are rejected. These relationships are represented by the edges of the graph.

```
Loading required package: Rgraphviz
Loading required package: graph
Loading required package: cluster
Loading required package: Ruuid
```



We can compute the closed testing procedure using the function

```
> ga.closed <- GlobalAncova.closed(xx = exprs(golubX), group = gr,
+     test.genes = kegg[1:5], previous.test = ga.kegg, level = 0.05,
+     perm = 100)
```

where *test.genes* is again a list of pathways. In order to shorten computing time we can provide the results of the previous application of `GlobalAncova` for the pathways of interest. The option *level* allows to manipulate the level of significance. *perm* again gives the desired number of permutations used in the permutation test.

The function `GlobalAncova.closed` provides the formed null hypotheses (this means lists of genes to be tested simultaneously), the test results for each pathway of interest and the names of significant and not significant pathways. If for a pathway one single hypothesis can not be rejected there is no need to test all the remaining hypotheses. That is why in test results of not significant pathways the lines are filled with NA's after a p–value $> \alpha$ occured.

```
> names(ga.closed)

[1] "new.data"        "test.results"    "significant"     "not.significant"

> ga.closed$test.results

$"00970"
                               genes F.value p.value.perm p.value.theo
00970                             23  4.4771            0            0
00970.00100                       36  5.7849            0            0
00970.05020                       35  3.4856            0            0
00970.01510                       73  3.1067            0            0
00970.00430                       31  4.8651            0            0
00100.00970.05020                 48  4.8007            0            0
00100.00970.01510                 86  3.8917            0            0
00100.00970.00430                 44  5.8676            0            0
05020.00970.01510                 82  2.9593            0            0
05020.00970.00430                 43  3.9224            0            0
01510.00970.00430                 81  3.3562            0            0
05020.00100.00970.01510           95  3.7099            0            0
05020.00100.00970.00430           56  4.9773            0            0
01510.00100.00970.00430           94  4.0528            0            0
01510.05020.00970.00430           90  3.1995            0            0
01510.05020.00100.00970.00430    103  3.8711            0            0

$"00100"
                               genes F.value p.value.perm p.value.theo
00100                             13  7.9829            0            0
00970.00100                       36  5.7849            0            0
00100.05020                       25  5.1134            0            0
00100.01510                       63  3.6746            0            0
00100.00430                       21  7.5026            0            0
00100.00970.05020                 48  4.8007            0            0
00100.00970.01510                 86  3.8917            0            0
00100.00970.00430                 44  5.8676            0            0
05020.00100.01510                 72  3.4526            0            0
05020.00100.00430                 33  5.3650            0            0
01510.00100.00430                 71  3.9091            0            0
05020.00100.00970.01510           95  3.7099            0            0
05020.00100.00970.00430           56  4.9773            0            0
01510.00100.00970.00430           94  4.0528            0            0
01510.05020.00100.00430           80  3.6839            0            0
01510.05020.00100.00970.00430    103  3.8711            0            0
```

$"05020"

	genes	F.value	p.value.perm	p.value.theo
05020	12	1.231	0.262	0.2565
00970.05020	NA	NA	NA	NA
00100.05020	NA	NA	NA	NA
05020.01510	NA	NA	NA	NA
05020.00430	NA	NA	NA	NA
00100.00970.05020	NA	NA	NA	NA
05020.00970.01510	NA	NA	NA	NA
05020.00970.00430	NA	NA	NA	NA
05020.00100.01510	NA	NA	NA	NA
05020.00100.00430	NA	NA	NA	NA
01510.05020.00430	NA	NA	NA	NA
05020.00100.00970.01510	NA	NA	NA	NA
05020.00100.00970.00430	NA	NA	NA	NA
01510.05020.00970.00430	NA	NA	NA	NA
01510.05020.00100.00430	NA	NA	NA	NA
01510.05020.00100.00970.00430	NA	NA	NA	NA

$"01510"

	genes	F.value	p.value.perm	p.value.theo
01510	50	2.4545	0.001	0
00970.01510	73	3.1067	0.000	0
00100.01510	63	3.6746	0.000	0
05020.01510	59	2.3235	0.000	0
01510.00430	58	2.8805	0.000	0
00100.00970.01510	86	3.8917	0.000	0
05020.00970.01510	82	2.9593	0.000	0
05020.00100.01510	72	3.4526	0.000	0
01510.00970.00430	81	3.3562	0.000	0
01510.00100.00430	71	3.9091	0.000	0
01510.05020.00430	67	2.7161	0.000	0
05020.00100.00970.01510	95	3.7099	0.000	0
01510.00100.00970.00430	94	4.0528	0.000	0
01510.05020.00970.00430	90	3.1995	0.000	0
01510.05020.00100.00430	80	3.6839	0.000	0
01510.05020.00100.00970.00430	103	3.8711	0.000	0

$"00430"

	genes	F.value	p.value.perm	p.value.theo
00430	8	6.3840	0	0
00970.00430	31	4.8651	0	0
00100.00430	21	7.5026	0	0
05020.00430	20	3.1246	0	0
01510.00430	58	2.8805	0	0
00100.00970.00430	44	5.8676	0	0
05020.00970.00430	43	3.9224	0	0
05020.00100.00430	33	5.3650	0	0
01510.00970.00430	81	3.3562	0	0
01510.00100.00430	71	3.9091	0	0

```
01510.05020.00430                67  2.7161           0           0
05020.00100.00970.00430          56  4.9773           0           0
01510.00100.00970.00430          94  4.0528           0           0
01510.05020.00970.00430          90  3.1995           0           0
01510.05020.00100.00430          80  3.6839           0           0
01510.05020.00100.00970.00430   103  3.8711           0           0
```

```
> ga.closed$significant
```

```
[1] "00970" "00100" "01510" "00430"
```

```
> ga.closed$not.significant
```

```
[1] "05020"
```

We get the same significant and not significant pathways as before.

5 Diagnostic Plots

There are two types of diagnostic plots available supporting communication and interpretation of results of the global ANCOVA. The `Plot.genes` visualizes the influence of individual genes on the test result while the `Plot.subjects` visualizes the influence of individual samples. Both plots are based on the decomposition of sums of squares.

We use again the Golub data for demonstration of the plot functions.

5.1 Gene Plot

The influence of each gene on the outcome of the test can be assessed and visualized with a diagnostic plot generated by our function `Plot.genes`. It corresponds to the function `geneplot` in the *globaltest* package. The function `Plot.genes` gives a graphical display of single gene-wise analysis for all genes. Bars are always positive as a reduction of sum of squares is always achieved in this case. The bar height indicates the influence of the respective gene on the test statistic. The added reference line is the residual mean square error per gene and corresponds to the expected height of the bars under the null hypothesis which says that the gene is not associated with the clinical outcome. Covariate information can be included in the same way as in the `GlobalAncova` function with the *covars* option. The bars are coloured in order to show in which of the phenotype groups the gene has higher expression values.

The commands for creating gene plots in the *GlobalAncova* and the *globaltest* are as follows. For facility of inspection it is useful not to plot the bars for all genes at one time but only for a few, for example 40.

The two approaches show almost the same results. We prefer plotting horizontal bars rather than vertical because we think it is easier to read off the bar heights this way. In the group variable *gr* 0 represents AML and 1 the ALL patients.

```
> Plot.genes(exprs(golubX)[cellcycle[1:40], ], gr)
> gp.cc <- geneplot(gt.cc)
> plot(gp.cc[1:40])
```

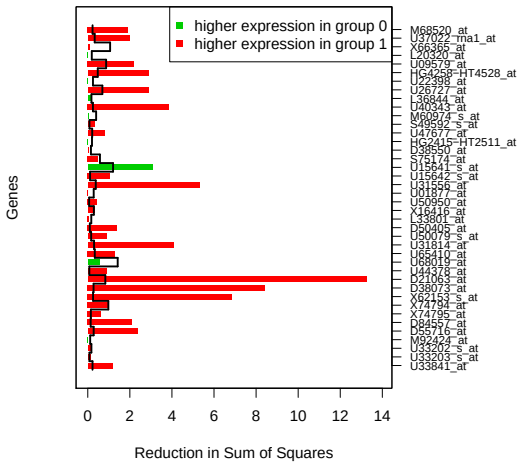


Figure 1: Gene Plot for the Golub data with *GlobalAncova*. Shown are the first 40 genes of the cell cycle pathway. The bar height indicates the influence of the respective gene on the test statistic. The colour shows in which of the phenotype groups the gene has higher expression values. The reference line is the residual mean square error per gene.

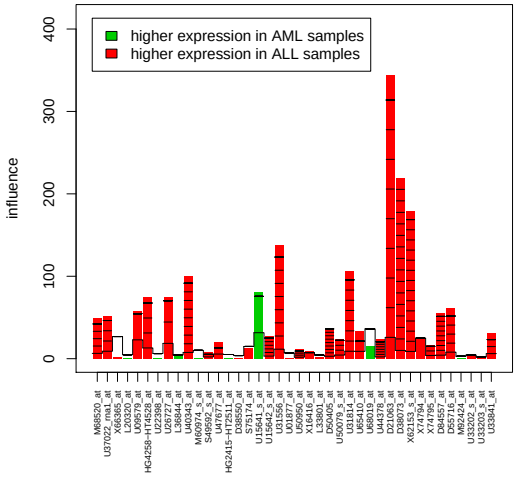


Figure 2: Gene Plot for the Golub data with *globaltest*. Shown are the first 40 genes of the cell cycle pathway. The bar height indicates the influence of the respective gene on the test statistic. The colour shows in which of the phenotype groups the gene has higher expression values. The reference line gives the expected height of the bar under the null hypothesis. Marks indicate with how many standard deviations the bar exceeds the reference line.

5.2 Subjects Plot

The function `Plot.subjects` visualizes the influence of the individual samples on the test result and corresponds to the `sampleplot` of Goeman. The function `Plot.subjects` gives information on the reduction of sum of squares per subject. Here we sum over genes. Large reduction demonstrates a good approximation of a subject's gene expressions by the corresponding group means. If an individual does not fit into the pattern of its phenotype, negative values can occur. A small p–value will therefore generally coincide with many positive bars. If there are still tall negative bars, these indicate deviating samples: removing a sample with a negative bar would result in a lower p-value. Again we can use *covars* for covariate adjustment. The bars are again coloured to distinguish the samples of the two different clinical diagnoses. With the option *sort* it is also possible to sort the bars with respect to the phenotype groups. Before plotting we add the sample names to the expression matrix. Otherwise with `Plot.subjects` samples would just be enumerated from 1 to 11.

We compare again the different approaches:

```
> colnames(exprs(golubX)) <- pData(golubX)[, 1]
> Plot.subjects(exprs(golubX), gr)
> sampleplot(gt.cc)
```

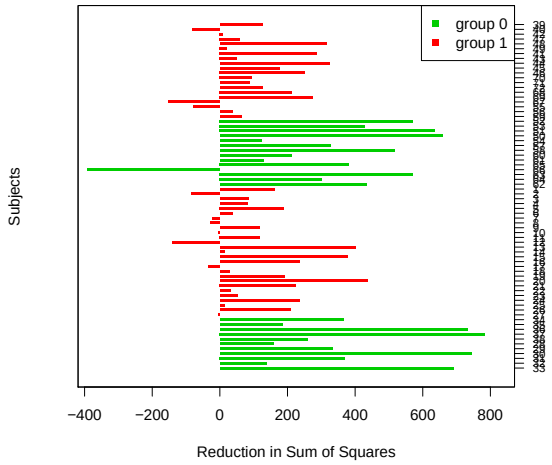


Figure 3: Subjects Plot for the Golub data with *GlobalAncova*. The bar height indicates the influence of the respective sample on the test result. If an individual does not fit into the pattern of its phenotype, negative values can occur. Bars are coloured corresponding to groups.

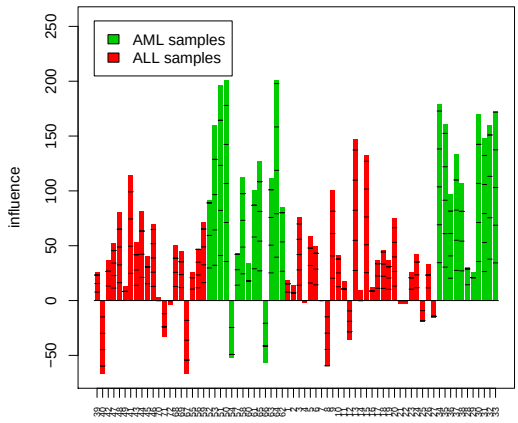


Figure 4: Subjects Plot for the Golub data with *globaltest*. The bar height indicates the influence of the respective sample on the test result. If an individual does not fit into the pattern of its phenotype, negative values can occur. Bars are coloured corresponding to groups. The reference line shows the expected influence of the samples under the null hypothesis. Marks on the bars indicate the standard deviation of the influence of the sample under the null hypothesis.

6 Acknowledgements

This work was supported by the NGFN project 01 GR 0459, BMBF, Germany.

References

[1] J. J. Goeman, F. de Kort, S. A. van de Geer, and J. C. van Houwelingen. A global test for groups of genes: testing association with a clinical outcome. *Bioinformatics*, 20 (1):93–99, 2004.

[2] S. Holm. A simple sequentially rejective multiple test procedure. *Scand. J. Statist.*, 6:65–70, 1979.

[3] U. Mansmann and R. Meister. Testing differential gene expression in functional groups. *Methods Inf Med*, 44 (3), 2005.

[4] R. Marcus, E. Peritz, and K. R. Gabriel. On closed testing procedures with special reference to ordered analysis of variance. *Biometrika*, 63 (3):655–660, 1976.