# POMELO II (THE RETURN OF THE POMELO)

## Help index for Pomelo II



## Introduction to Pomelo II

**(Or, for old-time Pomelo users: Why you should switch to Pomelo II)**

Pomelo II is a new and much improved incarnation of Pomelo, a tool for finding differentially expressed genes, and genes that are of potential interest because they are related to an outcome of interest (e.g., type of cancer, survival).

The main features that differentiate Pomelo II from its ancestor, Pomelo, are:

- Much faster than Pomelo. The underlying C++ code has now been parallelized for the permutation computations. For the longer problems you can expect up to a 40 to 60 (or 90 to 110, depending on the hardware where we are running Pomelo II) fold increase in speed. Whenever you run Pomelo II you will be using between about 60 (or 120) CPUs simultaneously.
- Customizable heatmaps for the results that include a dendrogram of the displayed genes. This allows "similar genes" to be grouped together in the figures, thus simplifying interpretation. You select the genes to plot based on different criteria (unadjusted p-value, FDR-adjusted p-value, ...).
- If you use any of the currently standard identifiers for your gene IDs for either human, mouse, or rat genomes, you can obtain additional information by clicking on the names in the output tables and figures. This information is based on that provided by our IDConverter tool.
- Tables of results are now sortable according to different criteria (unadjusted p-value, FDR-adjusted p-value, ...).
- New non-permutation methods have been added (t-limma, anova-limma and paired t-limma).
- It is now possible to use additional covariables (with anova-limma). That is, you can now add more information, such as: subject age, sex, country, or any other characteristic that might vary expression data.
- We now only provide FDR-adjusted p-values (see below).

## FDR and adjusted p-values

So as to control multiple testing effects, we provide FDR-adjusted p-values, which we calculate using the approach of **Benjamini & Hochberg** (1995; J. Royal Statistical Society B, 57:289-300). FWER-correction is no longer provided. We think that control of the FDR (False Discovery Rate) is probably more relevant for most genomic and proteomics research than control of the FWER (Family-Wise Error Rate); in addition, use of the maxT FWER control procedure requires that the subset pivotality assumption holds, which is not the case with some of the tests available in Pomelo.

Finally, even if the assumptions of the Benjamini and Hochberg's FDR procedure are not satistified, the error is often small (Reiner, A., D. Yekutieli, and Y. Benjamini, 2003, "Identifying differentially expressed genes using false discovery rate controlling procedures", Bioinformatics 19, 368-375) and, overall, this method is competitive to other alternatives such as Benjamini & Yekutieli's or permutation-based approaches.

More extensive details on hypothesis testing and multiple testing correction are provided in the help for Pomelo I.

## Tests available

### t-test (Permutations)

The all-famous t-test. Used to compare an interval variable (one where distances between levels make sense) between two groups. For example, we could compare gene expression data between two types of patients. We use here the test statistic for the case where we do not assume equal variances in the two groups. We compute the usual t-statistic, but p-values are obtained by permutation testing.

## Anova (Permutations)

Analysis of variance. We compare between more than two groups the value of an interval data, such as the gene expression among five types of cancer. The statistic is your usual ANOVA F-ratio and p-values are obtained using a permutation test.

## Regression (Permutations)

Linear regression. We try to predict the (interval scaled) values of a dependent variable based on the values of an independent, interval scaled variable. A typical example is predicting the expression levels of a protein using gene expression data. (Thus, note here that the "independent" variable is gene expression). We compute the usual ordinary least-squares regression coefficient and obtain the p-value using a permutation test.

## FisherIxJ

Non permutation method for contingency tables. It obtains the unadjusted p-value using Fisher's exact test. This test would be the one to use if we have, say, different classes of patients (e.g., six types of cancers) and for a set of 1000 markers we can have either presence/absence of each marker in each patient (this would yield 1000 contingency tables of dimensions 6x2 ---each marker by each cancer type), or we can have several levels of (unordered) expression, say four types of expression (which would yield 1000 6x4 tables).

If the markers have more than two levels of expression and these are ordered (say, from less to more) other tests could be more powerful, such as the Cochran-Armitage test (which we have not implemented). Another alternative with categorical response data are logistic and multinomial models (which we have not implemented).

Please note that FisherIxJ is a test that is rarely used, almost never with gene-expression arrays. We implemented FisherIxJ for some people who were using tissue arrays (that yielded pressence/absence values).

## Cox model

A widely used model for survival data. With survival data we often have censored observations (e.g., a patient that is not yet dead, and all we know is that it lived for at least 100 days after initiation of the treatment).

Currently, the Cox model included here works with interval-scaled covariates; if you pass it a categorical covariate, it can ONLY have two possible values; otherwise, you will get meaningless results (it will be treated as an interval scaled covariate).

The Cox model implemented only works with right-censored observations, not left-censored or doubly-censored ones. As well, it is of course your

responsibility to make sure that assumptions about censoring (e.g., lifetimes and censoring times are independent, etc) are reasonable. Statistics and p-values are computed in the "usual way"; the statistic we return is the Wald statistic and the p-value is from a Wald test.

## t-test (limma)

The limma t-test is a non-permutation method used to compare gene expression data between two groups. When performing a test on one particular gene, this test borrows information from all the other genes to obtain better estimates of the variance, using an empirical bayes method.

Please note that all three limma-based tests make several assumptions; most of them are fairly reasonable and seem to hold in practice. However, notice that, for the empirical bayes procedure to work, it is assumed that the error variance of all genes comes from the same distribution; this assumption is clearly violated if the expression levels for different genes are the result of **merging (averaging) over different number of spots**. For instance, if you plan to use the limma-based tests, you **do not want to use merge in** preP if the number of replicates per gene (or clone) is different among genes (clones).

## Paired t-test (limma)

This test is the same test as the limma t-test, but it is to be used when samples are paired. If samples are paired, you will want to use a paired t-test (or a linear model, as explained below). Suppose we want to perform a t-test to compare tumor vs non-tumor samples from the same subjects. Each of the samples belonging to the same person should be similar, as they are from the same subject and, of course, the samples are naturally paired (each pair tumor/non-tumor comes from the same subject). By using a paired t-test, we take pairing into account decreasing the variance of our estimates (generally, taking pairing into account leads to smaller estimates of the error variance, and thus larger test statistics).

Pairing comes in many forms and flavors. In general, you have pairing whenever there is a feature of your design or sampling scheme that leads to pairs of your two conditions to share some (generally random) characteristic. This shared characteristic will lead to the pairs showing correlation in expression. For instance, in the two samples from the same subject example, each pair of samples shares that they come from the same subject, where the subject characteristics themselves are not something that interest us **per se**, but will induce a correlation between the two observations in the same subject. Similarly, if we sample families, and we choose pairs of siblings, one with cancer and one without, we have again pairing (brothers or sisters are likely to resemble each other by virtue of them being, well, brothers/sisters). Another example of pairing would arise if we take samples from two cell lines every hour. If there are any circadian patterns in gene expression, then the two samples (one from each cell line) at each hour will show correlated gene expression; thus, the two samples from every hour are paired, and the data ought to be analyzed with a paired t-test.

Please, see caveat about merging.

## Anova, linear models (limma)

Non permutation method used to compare gene expression data between

two or more groups. Linear models are a more general type of models that include, as particular cases, ANOVA and multiple regression. We use here the functionality provided by limma, and thus return empirical bayes adjusted estimates (i.e., we use information from all genes to obtain better estimates of variances).

Once the test results have been calculated (i.e., after obtaining the F-statistic and associated p-value for the test of global differences between groups), this method gives us the option to individually compare classes (t-test) and draw venn diagrams with the class comparisons.

Please, see caveat about merging.

## Limma tests vs permutation tests

With permutation tests we obtain the p-value by comparing the observed statistic (e.g., the observed t-statistic) from our data against a null distribution obtained from the actual data themselves (e.g., we randomly permute the class labels, and compute the t-statistic for the permuted data, and we repeat this, for instance, 50000 times, so that we can obtain the null distribution of the statistic under the hypothesis that the class labels were assigned randomly, regardless of gene expression).

In contrast, when using limma we use the standard distributions for computing the p-values (i.e., we compare the observed t-statistic agains the expected t-statistic under the null hypothesis by comparing with the t-distribution with a given degrees of freedom). As explained above, limma does not directly compute the t-statistic with the usual formula, but rather uses a "moderated statistic", where the estimate of the variance for every gene is "moderated" using information from all other genes. This is done using a specific model for the variances with an empirical bayes procedure as explained in 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, No. 1, Article 3. What you will notice is that extremely small variances are pulled towards the much more frequent larger values and extremely larger variances are pulled towards smaller values. Moderated statistics have been shown repeatedly to work better (in several senses) with microarray data. The differences are most noticeable with small sample sizes.

## Using additional covariables

What Pomelo tries to do is "detect" genes that are different using as a criteria gene expression data difference among two or more classes. Since **most likely our data is not homogeneous** (homogeneous meaning all subjects of the same age, country, ... ), we may find that the differences due to different classes are "blurred" by other differences. By using additional covariables, we can add this information to our study and thus enable a more complete analysis. (If there is a factor, say age, that accounts for

some relevant portion of the variation in the gene expression data, but you don't use it in your analyses, then what you are doing is adding that variation, that is explained by age, to the error term. And, among other consequences, the larger the error term, the smaller the statistic).

Although additional covariables can be very useful, if the covariable you try to add is the same as the class file, pomelo can do nothing with this. **Eg.** If you are comparing two classes and you add a covariable indicating that all subjects of one class are male and all subjects of the other are female, pomelo has no way of knowing which differences belong to the covariable and which to the class difference. In this case, sex and class are completely confounded. Any differences in expression could be explained either by differences in the class variable you are interested, or by differences in sex.

Additional covariables can only be used with "Anova, linear models (limma)".

**Program input**

## Necessary data files for each test

The files **"Gene expression data"** and **"Class labels or dependent variable"** are required for all tests. Files which are only necessary for certain tests are: **"Censored indicator file"** (Cox test), **"Paired indicator file"** (paired t-test) and **"Additional covariables file"** (optional for anova-limma).

Gene expression data
: The file with the expression data (e.g., you microarray data). In this file, rows represent variables (generally genes) and columns are subjects or arrays. We want to find those variables that are most distinctly expressed among groups (e.g., a t-test or Anova) or that are most related to, say, survival (e.g., Cox model).

Class labels or dependent variable
: These are generally the class labels (e.g., healthy or affected, or different types of cancer) that group the samples, or the survival times of patients, or another dependent continuous variable (if regression models). In our analyses we want to find which of the genes shows differential expression between the classes given here, or if a given dependent variable (e.g., survival time, expression of certain protein) is significantly related to the expression of some gene.

Censored indicator
: For survival data only. An observation is censored if the time of occurrence of the event (usually death) has not yet been observed. We will represent uncensored observations with a 1 (because we have observed the time of death) and censored observations with a 0 (meaning that the survival time given is only a lower bound).

Paired indicator
: This file is only necessary for paired t-test. It is used to indicate which subjects are paired. Two subjects are paired if they share some common characteristic that we think may imply similar gene expression traits (e.g., two samples from the same subject would be

paired).

Additional covariables
: You may use this file if you wish to add more information to your analysis (such as subject age, weight, ...). You will be offered this option after having launched a **limma anova test**.

## Data format for gene expression data

The file for the gene expression should be formated as:

- Data should conform to the "genes in rows, patients (or classes) in columns". In other words, each row of the data file is supposed to represent a different gene or variable and we will carry-out **one test for each row**.
- Use tab (\t) as the field separator within rows.
- Use newline (\n) between rows. It is also convenient to finish each file with one (\n).
- The first column is assumed to contain the ID information for genes, marker, or whatever. This will be used to label the output (but it also means that whatever is in the first column is not used in the analyses).
- You can have an arbitrary number of rows with comments. These rows must always start with an "#".
- Array names: if you want to name your arrays (useful for the output of the analyses) do as follows:
  1. Place a line that starts with "#";
  2. After the "#" put "Name" or "NAME" or "name" (don't say we don't give you choices);
  3. Write the array names (separated by tabs).
- There should be no rows with non-numeric data (except for the comments and name rows). In particular, eliminate initial rows that contain things such as headers with the IDs of patients.
- **Missing values** can be coded in three different ways: a) as "NA"; b) with one empty space (i.e., " "); c) as nothing (i.e, simply leave these places empty).
- This is a small gene expression data file using "NA" for missing values:

| | | | | | |
|---|---|---|---|---|---|
| gene1 | 23.4 | 45.6 | NA | 76 | 85.6 |
| genW@ | NA | 34 | 23 | NA | 13 |
| geneX# | 23 | 25.6 | 29.4 | 13.2 | NA |

- This is the same file using nothing for missing values:

| | | | | | |
|---|---|---|---|---|---|
| gene1 | 23.4 | 45.6 | | 76 | 85.6 |
| genW@ | | 34 | 23 | | 13 |
| geneX# | 23 | 25.6 | 29.4 | 13.2 | |

- This is the same file using nothing for missing values and a first row with array identifiers:

| | | | | | |
|---|---|---|---|---|---|
| #name | s1 | s2 | s3 | s4 | s5 |
| gene1 | 23.4 | 45.6 | | 76 | 85.6 |
| genW@ | | 34 | 23 | | 13 |
| geneX# | 23 | 25.6 | 29.4 | 13.2 | |

- Beware that **some spreadsheet programs (such as Excel)** will give you a lot of headaches if you use nothing as a missing value code when the missing values are at the end of the row. That is because for those

lines or rows those programs truncate the row at the last valid observation (i.e., they discard the last trailing tabulators that separate empty space), so you will have a file where different rows have different number of values. And Pomelo will rightfully complain and will not run. For example, the last file, if it came from Excel, would probably have five columns in the last row (and not six columns, as it should), and Pomelo would give you an error message and stop till you fix it. Solutions? Use NA for passing missing values to Pomelo, or do not pass those files through spreadsheets.

## Data format for class labels and censored indicators

Separate values by tab (\t), and finish the file with a carriage return or newline. No missing values are allowed here. Class labels can be anything you wish; they can be integers, they can be words, whatever. Of course, if you are using regression or Cox model, your dependent variable must be a number. And if you are using a t-test there can only be two classes.

This is a simple example of class labels file:

CL1 CL2 CL1 CL4 CL2 CL2 CL1 CL4

## Data format for Fisher's exact test on contingency tables

The **class labels**, as above, can be any arbitrary coding. The values in the **gene expression** data file should be consecutive integers that start at 0 (i.e., do **NOT** use negative intergers, or you'll get errors). This is an example file:

| #gene | c1 | c1 | c2 | c2 | c3 | c3 |
|---|---|---|---|---|---|---|
| gene1 | 0 | 0 | 1 | 1 | 1 | 0 |
| gene2 | NA | 2 | NA | 1 | 0 | 0 |
| gene3 | NA | 1 | 2 | NA | 0 | 0 |
| gene4 | 1 | 1 | 2 | 2 | 2 | 0 |
| gene5 | 1 | 0 | 0 | 2 | 1 | 2 |
| gene6 | 2 | 1 | 1 | 2 | 0 | 0 |
| gene7 | 2 | 1 | 1 | 0 | 2 | 0 |
| gene8 | 2 | 2 | 0 | 0 | NA | NA |

As you can see, most of these rows would yield (if we used the first row as the class labels) 3x2 tables or 3x3 tables (e.g., 3rd or 5th rows).

## Data format for paired indicator

The format should be the same as for class labels, except that we will place the same integer at the places where the two paired subjects are. We can choose any integer we want.

This is a simple example of a paired indicator file:

4 32 4 8 32 8 22 22

In the example file, the first and the third subject are paired (integer used 4, although we could have used any other), the second subject is associated to the fifth (integer used 32) and so on.

## Data format for additional covariables

Each row of the file represents a subject and each column an additional covariable. The format must be as follows: tab (\t) separated fields, a first row with the names of the columns (covariables) and the following rows with each subject's covariable value.

This is a simple example of a additional covariables file:

| Age | Country | Weight |
|---|---|---|
| 20 | Spain | 76 |
| 46 | Germany | 100 |
| 33 | England | 88 |
| 59 | Spain | 79 |
| 61 | England | 65 |
| 24 | Spain | 58 |
| 80 | Germany | 90 |
| 59 | England | 72 |

The order of the subjects **must be exactly the same** as the order in the expression data file. That means, the covariable values (age, country,...) of the first subject (i.e. row), must belong to the subject in the first column of the expression, etc.

The data can be of two types: numeric (age, weight, ...) or non-numeric (country, ...). When the data is read, you will be taken to a page where you will be able to see if it has been read correctly. Be careful, if your data does not have the correct format, it can be **incorrectly read**, meaning a non-numeric variable can be read as numeric or vice versa. If this happens all further analyses will be wrong!

## Output files

Once the program has finished running, you will get a table with the output from the run and a heatmap.

The results table contains a header indicating the test you have used, number of permutations and which covariables where used (if any).

The table shows an index corresponding to the original ordering in the data file, gene names, p-values (undajusted), FDR-adjusted p-values, and statistics (and the absolute value of the statistic). For Fisher's IxJ tests the columns names statistic and abs(statistic) really have no meaning.

The figure is a heatmap where you can filter how and which genes to plot. For now the color scale goes from yellow to blue (green is the mix of yellow

and blue); missing values are shown as white. Both tables and heatmap are clickable and will take you to a page with additional information (our IDConverter Light, based on IDConverter).

If you have run an "Anova, linear models (limma)" test, the output will also contain a Class compare section containing a button. By clicking on the button we will be taken to a class compare page, where we will be able to compare classes individually.

The observed test statistics are:

t-test (Permutations)
: The difference of means divided by the square root of the sum of the sample variances of each of the means. (So just the usual t for the unequal variance case).

Anova (Permutations)
: The usual F-statistic (mean squares model/mean squares error).

Regression (Permutations)
: The coefficient divided by its standard error (i.e., the typical t-statistic).

Cox model
: The Wald statistic (the estimate of the coefficient divided by its standard error).

t-test (limma)
: Moderated t-statistic.

Paired t-test (limma)
: Moderated t-statistic for paired subjects.

Anova, linear models (limma)
: Moderated F-statistic.

## Examples and data sets

In this page there are a few examples. Data sets to play with Pomelo II can be obtained from the Examples pages for SignS and GeneSrF.

## Code, authors and acknowledgements

The code underlying these tests is written in C++ and R. The C++ code was originally written by Ramón Díaz-Uriarte and has been parallelized with LAM/MPI by Edward Morrissey (substitute the "%%" by "@"). All of the multiple testing functions have been written from scratch, although some algorithms have been based on Westfall & Young (1993) or the documentation for the "multtest" package for R, some inspiration has been obtained from the above package (and a lot of testing has been done using multtest as a benchmark). Moreover, we have taken C code from R for Fisher's exact tests (fexact.c; the latter is based on Mehta and Patel's algorithm). The code for Cox model is from R, library survival.

All three limma tests have been written using limma R package [Smyth, G.

K. (2005). Limma: linear models for microarray data. In: 'Bioinformatics and Computational Biology Solutions using R and Bioconductor'. R. Gentleman, V. Carey, S. Dudoit, R. Irizarry, W. Huber (eds), Springer, New York, pages 397--420].

We have also used the GNU scientific library, GSL.

The CGI has been written by Edward Morrissey and Ramón Díaz-Uriarte using Python. The heatmaps use R and the R packages GDD by Simon Urbanek and imagemap by Barry Rowlingson.

The code from R and the survival, limma, GDD, and imagemap packages, and the GSL are all free software released under GNU's General Public License. Our C++ is thus also released under the GPL. The CGI code is released under the Affero GPL.

This application is running on a cluster of machines using Debian GNU/Linux as operating system, Apache as web server, Linux Virtual Server for web server load-balancing, with heartbeat and drbd for high-availability of both services and storage, and LAM/MPI for parallelization.

**Funding** partially provided by Project TIC2003-09331-C02-02 of the Spanish Ministry of Education and Science. This application is running on a cluster of machines purchased with funds from the RTICCC.

## Terms of use: NO WARRANTY

- You acknowledge that this Software is experimental in nature and is supplied "AS IS", without obligation by the authors or the CNIO to provide accompanying services or support. The entire risk as to the quality and performance of the Software is with you. The CNIO and the authors expressly disclaim any and all warranties regarding the software, whether express or implied, including but not limited to warranties pertaining to merchantability or fitness for a particular purpose.

## Terms of use: citation

If you use Pomelo II for any publication, we would appreciate if you could let us know and that you cite our program (you know, "credit where credit is due"). For now, you can give the main web site: http://pomelo2.bioinfo.cnio.es).

We appreciate if you give us feedback concerning bugs, errors or misconfigurations. Complaints or suggestions are welcome.

## Privacy and Security

Uploaded data set are saved in temporary directories in the server. These data are accessible from the internet until they are erased (five days now). The same as you do, anybody can access those directories, if they known their names, which are made of 20 random digits, so they should not be easy to guess.

In any case, you should keep in mind that communications between the client (your computer) and the server are not encripted at all, thus it is also possible for somebody else to look at your data while you are uploading or dowloading them.

Again, all responsibility for access to your data is yours. We do not offer any warranties that your data will be kept confidential or that nobody will break into our system. We suggest you rename the rows, columns, and other identifiers, so that they are meaningless to anybody but you.

## Copyright and license