μToss Quick Start Guide

μToss Team

February 12, 2010

1 Introduction

The *μToss* packages allow the user to discover, apply and compare multiple testing procedures and multiple interval estimation procedures.

They include a corpus of functions implementing and integrating these procedures and a GUI. These are found in the *mutoss* and *mutossGUI* packages respectively.

2 *μToss* Rationale

The rationale behind the *μToss* packages is two-fold.

It is aimed at allowing statisticians to discover, apply and compare standard and custom multiplicity controlling procedures. This is achieved by the mutoss package which includes:

- Mutoss Class: An S4 class object designed to hold the input and output of various multiple testing procedures and to serve as a standard input and output of future procedures.
- Elementary Functions: A mass of functions either implementing a procedure or calling an existing implementation. These are used by both the GUI and the command line interface.
- Simulation Platform: *simulation( )* and *gatherStatistics( )* functions allowing the mass generation of data, application of different procedures and analysis of results (documented separately).

μToss is also aimed at the researcher wishing to analyze new data or to reproduce published results. This is accomplished by the mutossGUI package.

> At the time of release, the package has only undergone basic testing. This being the case, we recommend new data to be analyzed with standard software alongside *μToss*. This is planned to change in future releases.

3 System Requirements

3.1 mutoss Package

The package will run on any machine running R with recommended version 2.8 and above.

3.2 mutossGUI package

On top of the mutoss package requirements, Java Run time Environment ver 5 and above is needed.

4 GUI Work flow

Download and install the *mutossGUI* package. The GUI should start automatically. Otherwise load it with

```
>mutossGUI()
```

4.1 Testing of Hypotheses

If you have already a vector of p-values start at step (5).

1. Load the raw data (assumed to be a *data.frame* object) using the **Data** button.
2. Specify the model type and explanatory variables using the **Model** button.
   For linear contrasts choose **Single endpoint in k groups**.
   For applying the one single model to many response variables choose **Multiple endpoints in k groups.**
3. Define model by choosing response and explanatory variables.
4. Define the hypotheses of interest by specifying the contrasts using the **Hypotheses** button.
5. Insert p-values using the **p-Value** button or calculate them following the previous steps.
6. Choose the error type to control using the **Error Rate** button.
7. Use the **Adjusted p-Values** to calculate the procedure specific adjusted p-values (you will be prompted for additional options when necessary) or choose **Rejected** to apply the procedure and reject hypotheses.
8. Visualize results by choosing the **Info** option in the **Adjusted p-Values** or **Rejected** buttons.

9. Save the output as an R object using the **File->Export MuToss Object to R** option.

Further analysis is now possible using the compareMutoss functions or other R functionality.

4.2 Interval Estimations

Steps 1-4 are identical to the hypothesis testing work flow.

1. Load the raw data (assumed to be a *data.frame* object) using the **Data** button.
2. Specify the model type and explanatory variables using the **Model** button.
   For linear contrasts choose **Single endpoint in k groups**.
   For applying the one single model to many response variables choose **Multiple endpoints in k groups.**
3. Define model by choosing response and explanatory variables.
4. Define the contrasts of interest by specifying the contrasts using the **Hypotheses** button.
5. Choose the error type to control using the **Error Rate** button.
6. Use the **Confidence Intervals** to compute confidence intervals on parameters of interest.
7. Visualize results by choosing the **Info** option in the **Confidence Intervals** button.
8. Save the output as an R object using the **File->Export MuToss Object to R** option.

Further analysis is now possible using R functionality.

5 Command Line Work Flow

Download and install the *mutoss* package to access the different procedures in the package (note mutossGUI is not needed for this purpose). A list can presented using

```
>help(package='mutoss')
```

To work with these elementary functions, just use them as any other R function. See in line help for further details.

To use these functions to read and write into Mutoss S4 class objects use the *mutoss.apply( )* function. See the in line help of the function for further details.

6 Glossary

Hypotheses-Testing-Procedures The corpus of procedures for testing multiple statistical hypotheses.

Interval-Estimating-Procedures The corpus of procedures for constructing interval estimates for multiple parameters.

p-Value-Procedures The corpus of (multiple) hypotheses testing procedures which rely on the marginal p-values of each hypothesis (and do not require the original data and model). This procedures might possibly require additional information such as logical relations between procedures, a qualitative description of the probabilistic relation between test statistics etc.

Data-Procedures The corpus of (multiple) testing procedures which require the original response variables, the explanatory variables (model) and the parameters of interest (contrasts).
These procedures can be seen as p-value-procedures with a specific relation between test-statistics which is derived from the model and the contrasts.

Error-Type The type of error a procedure aims to control. This can be a hypothesis testing error rate (FWER. FDR,...) or an interval estimation error rate (simultaneous coverage, false coverage rate,...).

Error-Rate The allowed rate of the *Error Type*.