A software for the analysis of nonlinear mixed effects models

*Maximum likelihood estimation*

*Model selection*

*Hypothesis testing*

*Graphical analysis*

*Data simulation*
# Contents

1 Downloading packages ................................................. 4

2 Installation ............................................................... 5
   2.1 Prerequisites ...................................................... 5
      2.1.1 Linux specifics ............................................. 5
      2.1.2 Windows 64bits specifics ............................... 5
   2.2 About Installer .................................................. 7
   2.3 Directory structure ............................................. 8
      2.3.1 Installation directory ................................. 8
      2.3.2 User directory ...................................... 9
   2.4 About Plugins .................................................. 9
   2.5 Running MONOLIX .............................................. 10
   2.6 Installation use cases ..................................... 10
      2.6.1 Desktop .................................................. 10
      2.6.2 Desktop with a shared MONOLIX installation ............ 10
      2.6.3 Application server with a shared MONOLIX installation .. 11
      2.6.4 Application server with a remote connection .......... 11
      2.6.5 Application server with a desktop installation .......... 11
      2.6.6 Cluster installation with a shared MONOLIX installation .. 12
      2.6.7 Cluster installation with MONOLIX installed on each node .. 12
   2.7 License .......................................................... 12
      2.7.1 Desktop license ....................................... 13
      2.7.2 Floating license .................................... 13
      2.7.3 Roaming license .................................... 22

3 Troubleshooting ....................................................... 23
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1 Downloading MONOLIX</td>
<td>23</td>
</tr>
<tr>
<td>3.2 Running MONOLIX</td>
<td>23</td>
</tr>
<tr>
<td>4 ChangeLog</td>
<td>25</td>
</tr>
</tbody>
</table>
1 Downloading packages

The MONOLIX packages can be downloaded through the download manager hosted at http://download.lixoft.com. The download manager is available for users provided with an access key. Different MONOLIX packages are available, depending on the MATLAB version and of the operating system. MONOLIX currently supports Windows XP/Vista/Seven 32bits, Linux (all common distributions) 32/64 bits. On Windows XP/Vista/Seven 64 bits, MONOLIX standalone version can run in 32 bits mode.

Choice of MONOLIX versions

- MATLAB versions:
  - Linux matlab-r2010b-r2011b-r2011b-r2012a (64 bits)
  - Linux matlab-r2009a-r2010a (32 bits)
  - Linux matlab-r2010b-r2011b (32 bits)
  - Linux matlab-r2009a-r2010a (64 bits)
  - Windows (seven and vista) matlab-r2009a-r2010a (32 bits)
  - Windows (seven and vista) matlab-r2010b (32 bits). Due to bugs in MATLAB 2010b, it is strongly recommended to use MATLAB 2010b-SP1.

- Standalone versions:
  - Linux (32 bits)
  - Linux (64 bits)
  - Windows (32 bits)
  - Windows (64 bits)\(^1\)

\(^1\)MONOLIX will run in 32 bits mode
2 Installation

2.1 Prerequisites

perl is required to run perlScripts and the validation suite; it is not required otherwise.

2.1.1 Linux specifics

- install sharutils: uudecode is required to uncompress the Monolix package;
- make sure you have gcc/g++/make installed or install them.

2.1.2 Windows 64bits specifics

The 32 bits standalone version of Monolix runs fine on Windows 7 64bits. You will need to install the 64 bits Windows version of Monolix in any of these situations:

- On other 64 bits versions of Windows (non Windows 7);
- If you wish to use a MatLab version of Monolix.
- If you simply prefer to use a 64bits version of standalone Monolix, although in practice this should not have an impact on the performance.

The installer of the 64 bits Windows version of Monolix executes the Windows SDK installer. This SDK embeds the C++ compiler required to generate Mlxtran modules. The installation process is as follows:

Welcome page of the SDK installer: click on ‘next’ button to continue;
2.1 Prerequisites

License agreement page: accept the agreement by selecting ‘I agree’, then click on ‘next’ button to continue;

Choose the installation path. The proposed directories are required by Monolix;

The component used by Monolix are the compiler only, therefore it is not necessary to install documentation and samples.

On ‘Begin installation’ page, click on ‘next’ button to continue;
On 'Installation complete' page click on ‘finish’ button to continue;

After completion of the SDK installation procedure, the Monolix installer resumes the Monolix installation procedure.

If the SDK was already installed on the computer, the SDK installer will propose a list of actions:

No action is necessary, click on 'Cancel';

Confirm the 'Cancel' choice by clicking on 'Yes' button.

2.2 About Installer

- Linux: the installer is a self-extractable archive.
  - run the following command (depending on your os version):
    #> sh Monolix-4.2.1-matlab2010a-linux32.bin
    or
    #> sh Monolix-4.2.1-matlab2010bSP1-linux32.bin
2.3 Directory structure

You can specify the target installation directory by giving the path as argument.
A directory containing Monolix will be created in the directory installation path.

- Windows
  - copy the installer on your Desktop or in your windows temporary directory
  - Double click on the executable and follow the instructions.

2.3 Directory structure

The Monolix directory structure is divided in two parts:

- the software directory containing the Monolix program,
- the personal user directory containing the Monolix workspace and documentation

2.3.1 Installation directory

```
Monolix ................................................................. Monolix root directory
  monolix413 ......................................................... Monolix version directory
    bin ............................................................... tools directory
    config .......................................................... configuration files
      Demos ......................... set of demos (copied in Monolix user directory)
      graphics ..................................................... graphics configurations
        listOfGraphics ............... graphics predefined configurations
        project ..................... graphics default configurations for Mlxtran
        settings .................... graphics default configurations
        scenario ...................... predefined scenarios
    session ................................................. Monolix session configuration
  factory ......................................................... Mlxtran C++ API
  jar ............................................................... Java library
  lib ................................................................. C++ library
  matlab ......................................................... Monolix main program
```
2.4 About Plugins

Monolix can embed the BiM plugin, a faster and more accurate ODE solver, not included by default due to its license restrictions. The plugin must therefore be downloaded and installed separately:

- **Linux**
  
  - libBim.tgz (using the command: `tar xzf libBim.tgz` or your graphical archiver)
  
  - copy the files stored in the directory libBim into the library directory of Monolix:
    * for the MATLAB version: `<install path>/lib`
    * for the standalone version: `<install path>/bin/Monolix_mcr/runtime/lib`

- **Windows**
  
  - copy libBim.dll into the library directory of Monolix:
    
    - for the matlab version: `<install path>/lib`

2.3.2 User directory

The user directory is created after the first launch of Monolix. This directory contains the basic configuration of Monolix, documentation, demos, log files, license file, ...

```
monolixData .................................................. Monolix root directory
monolix413 .................................................. Monolix version directory
  doc .................................................. Monolix documentation
  log .................................................. Log files
  script_modules ............................................. Compiled Mlxtran modules
  perlScripts ................................................ Perl scripts
  work .................................................. User working directory
  Demos .................................................. Modifiable demos
  config .................................................. Configuration files
  license .................................................. Tools directory
  tmp .................................................. Set of demos (copied in Monolix user directory)
```
2.5 Running MONOLIX

- for the standalone version: <install path>/bin/Monolix_mcr\runtime\lib

With the standalone version of MONOLIX the directory <install path> is located at:

* under Window XP or Windows Server 2003:
  c:\Documents And Settings\All Users\Application Data
* under Window Vista, Windows 7 or Windows Server 2008:
  c:\ProgramData

Important notice: these directories may be hidden by the operating system, thus you have to configure your file browser for access.

2.5 Running MONOLIX

- Linux
  - MATLAB version
    * start MATLAB
    * go to directory '<install path>/matlab' and type monolix.
  - Standalone version: go to '<install path>/bin' and type ./Monolix.sh.

- Windows
  - MATLAB version
    * start MATLAB
    * go to directory <install path>/matlab' and type monolix.
  - Standalone version: go to '<install path>/bin' and type Monolix.bat.

2.6 Installation use cases

2.6.1 Desktop

MONOLIX is installed on the computer of the user and the user has a personal activation key (see Section 2.7.1 Desktop license). After the installation or during the first startup of MONOLIX a popup titled 'Lixoft Activate' appears and asks the activation key. When the activation procedure is finished, MONOLIX will be configured (typically a directory monolixData is created in the user home directory) and launched.

2.6.2 Desktop with a shared MONOLIX installation

MONOLIX is installed on a remote server and the user accesses MONOLIX through a shared directory (via CIFS, Network drive, NFS, ...) and the user has a personal activation key (see Section 2.7.1 Desktop license).

During the first startup of MONOLIX a popup title 'Lixoft Activate' appears and asks the activation key. When the activation procedure is finished, MONOLIX will be configured (typically a directory monolixData is created in the user home directory) and launched.
2.6 Installation use cases

2.6.3 Application server with a shared Monolix installation

Monolix is installed on a remote server using the procedure described in Section 2.7.2 'Floating license'. The license file (obtained during activation procedure) is copied in the directory

- `<monolix user install path>/config/system/access` for the MATLAB version of Monolix
- or `<monolix install path>/bin/Monolix_mcr/runtime/config/system/access` for the standalone version of Monolix.

The user accesses Monolix through a shared directory (via CIFS, Network drive, NFS, ...). The user runs Monolix directly, no activation is required. Nevertheless, when a user runs Monolix a license token is taken. If all license tokens are used (too many users run Monolix in the same time), a popup titled 'Lixoft activate' appears and the user is supposed to wait until at least one token is released.

2.6.4 Application server with a remote connection

With a floating license Monolix is installed on a remote server using the procedure described in Section 2.7.2 'Floating license'. The license file (obtained during activation procedure) is copied in the directory

- `<monolix user install path>/config/system/access` for the MATLAB version of Monolix
- or `<monolix install path>/bin/Monolix_mcr/runtime/config/system/access` for the standalone version of Monolix.

The user accesses Monolix using a remote desktop application. The user runs Monolix directly, no activation is required. Nevertheless, when a user runs Monolix a license token is taken. If all license tokens are used (too many users run Monolix in the same time), a popup titled 'Lixoft activate' appears and the user is supposed to wait until at least one token is released.

With desktop licenses Monolix is installed on a remote server, the user accesses to Monolix using a remote desktop application and has a personal activation key (see Section 2.7.1 Desktop license).

During the first startup of Monolix a popup title 'Lixoft Activate' appears and asks the activation key. When the activation procedure is finished, Monolix will be configured (typically a directory `monolixData` is created in the user home directory) and launched.

2.6.5 Application server with a desktop installation

Monolix is installed on a remote server using the procedure described in Section 2.7.2 'Floating license'. Each Monolix user is supposed to have a copy of the license file obtained during the activation procedure. After the installation or during the first startup of Monolix
2.7 License

, a popup titled 'Lixoft Activate' appears. The tab 'With License file' has to be selected. The user is supposed to browse to the copy of the license file to activate Monolix. When a user runs Monolix, a license token is taken.

If all license tokens are used (too many users run Monolix in the same time), a popup titled 'Lixoft activate' appears and the user is supposed to wait until at least one token is released.

2.6.6 Cluster installation with a shared Monolix installation

Monolix is installed on a master server using the procedure described in Section 2.7.2 'Floating license'. The license file (obtained during activation procedure) is copied in the directory

- `<monolix user install path>/config/system/access` for the MATLAB version of Monolix
- or `<monolix install path>/bin/Monolix_mcr/runtime/config/system/access` for the standalone version of Monolix.

Each cluster node accesses to Monolix through a shared directory (via CIFS, Network drive, NFS, ...).

The user runs Monolix directly, no activation is required. Nevertheless, when a user runs Monolix a license token is taken (there is no limit of runs on cluster nodes).

If all license tokens are used (too many users run Monolix in the same time), a popup titled 'Lixoft activate' appears and the user is supposed to wait until at least one token is released.

2.6.7 Cluster installation with Monolix installed on each node

License server (RLM) has to be installed on a master server and the license file is downloaded using the procedure described in Section 2.7.2 'Floating license'. Monolix is installed on each cluster. During this installation it is not necessary to activate Monolix when the popup titled 'Lixoft activate' appears (just close the popup). The license file (obtained previously) is supposed copied in the directory

- `<monolix user install path>/config/system/access` for the MATLAB version of Monolix
- or `<monolix install path>/bin/Monolix_mcr/runtime/config/system/access` for the standalone version of Monolix

of each node.

2.7 License

Monolix licenses can be of the following types:

- Individual license - named user. The named user can install and run Monolix on a predetermined number of different computers.

- Floating license - concurrent access. The license is hosted by a license server, and Monolix can either run on a server or individual workstations.
2.7 License

Remark: the former license management tool uses a license file (license.ini); this type of license is deprecated since Monolix version 4.1.3.

2.7.1 Desktop license

The activation key (provided by Lixoft) must be entered in the dialog box titled 'Lixoft license activation' (‘With activation key’ tab). This dialog box only appears when no license is available on the user’s computer or when the license expires.

2.7.2 Floating license

The use of a floating license requires to set up a license server. In this case there are two installation strategies for Monolix users:

- install Monolix on a directory shared by all Monolix users,
- install Monolix on each user’s computer and copy the license file obtained as described below into the directory:

  - <monolix user install path>/config/system/access for the MATLAB version of Monolix,
  - or <monolix install path>/bin/Monolix_mcr/runtime/config/system/access for the standalone version of Monolix.

After the installation process, when the 'Lixoft activate window' appears just close the window (do not enter the activation key of the floating license). Then, start the RLM server, located at:

- <monolix install path>/tools/rlm/rlm.exe for the MATLAB version of Monolix,
- or <monolix install path>/bin/Monolix_mcr/runtime/tools/rlm/rlm.exe for the standalone version of Monolix.

At this step there is no license available yet; the IT manager should use the RLM web server to download the license by following the procedure below:
1. In the web browser, type `<IP>:5054`, where `<IP>` is the IP address of the computer hosting the RLM server (e.g. 192.168.46.248:5054).

   Notice that the RLM server opens two ports: 5053 and 5054. The first port (5053) is a service port used for the transactions of licenses. The second port (5054) is the RLM web server port used to access to the RLM configuration through a web browser.

   It is possible that one or both ports may have been used by another application.

   - If the web server port (5054) is not available you can launch RLM server with a new port by using the program option `-ws` (e.g: `rlm -ws 5055`) in this case, the access to RLM configuration through a web browser is done using the address `<IP>:<NEW PORT>` (e.g. 192.168.46.248:5055).

   - If the server port (5053) is not available, a file `config.conf` has to be created in the `rlm` directory and has to contain the following information:

     ```
     HOST `<IP>` `<MAC ADDRESS>` `<NEW PORT>`
     ```

     e.g.
     ```
     HOST 192.168.46.245 a8c0f82e 5060
     ```

2. Begin license activation:
3. Enter the RLM activation url: `activate.lixoft.net`. And click on Next button.

If the rlm server does not have Internet access, the license has to be created by LIXOFT. Send a mail to support@lixoft.com with the following informations:

- Mac address of the computer hosting the RLM server
- IP address of the computer hosting the RLM server
LIXOFT will send in return a `.lic` file which has to be copied in the directory

- `<monolix install path>/config/system/access` (MATLAB version of Monolix)
- `<monolix install path>/bin/Monolix_mcr/runtime/config/system/access` (standalone version of Monolix).

At this step, the installation of Monolix is complete.

4. Activate the license.

Fill the ISV input with the string 'lixoft' (without the quotes) and the License activation key with the activation key provided by LIXOFT (key format is xxxx-xxxx-xxxx-xxxx)
5. Enter (at maximum) the number of bought licenses, then click on **Next** button.

Notice, the number of licenses cannot exceed the number of bought licenses.
6. Select the license directory and file.

In the field named License file to create write the full path to license file
<monolix install path>/config/system/access/myfloat.lic for the MATLAB version of MONOLIX
or <monolix install path>/bin/Monolix_mcr/runtime/config/system/access for the standalone version.

e.g. if the MONOLIX (matlab version) installation directory is /media/share/monolix the input field name License file to create should contain
/media/share/monolix/config/access/myfloat.lic

This license file has to be copied on each installation of MONOLIX:

- If Monolix is installed on a shared space (i.e. each node of the cluster has an access to this directory), copy the license file into the directory
  <monolix install path>/config/system/access/ for the MATLAB version of MONOLIX
  or <monolix install path>/bin/Monolix_mcr/runtime/config/system/access for the standalone version.

Make sure that the MONOLIX directory is accessible from each cluster node.
Example (with a MATLAB version of Monolix)

- Monolix is installed on the computer master-computer in the directory:
  /usr/local/monolix/.
  The license is in the directory:
  /usr/local/monolix/config/access/
- The RLM server is run on the computer master-computer.
- Cluster computers mount the directory /usr/local/monolix/.
- Each monolix user runs Monolix from the previously mounted directory.
  
  • If Monolix is installed on each node of the cluster, copy the license file on each computer in the directory <monolix install path>/config/system/access for the MATLAB version of Monolix or <monolix install path>/bin/Monolix_mcr/runtime/config/system/access for the standalone version.

Example

- The RLM server is executed on the computer master-server.
- Monolix is installed on each cluster node of the cluster.
- The license file is copied on each cluster node in the directory <monolix install path>/config/system/access/ for the MATLAB version of Monolix or <monolix install path>/bin/Monolix_mcr/runtime/config/system/access for the standalone version.
- Each monolix user runs Monolix from the cluster node.
7. Stop the server manually and restart it from the directory (or use option -c)

- `<monolix install path>/config/system/access/` for the MATLAB version
- `<monolix install path>/bin/Monolix_mcr/runtime/config/system/access` for the standalone version of Monolix.

Now RLM is running with the provided license. This is verified in the web interface by clicking on status button.
8. RLM Server: server hostname and port considerations.
   If for any reason, the server port or the server hostname is not registered in a DNS, it is
   possible to change these informations directly on licence file.
   The line `HOST <hostname> <mac> <port>` can be changed by `HOST <rlm server ip> <mac>
   <new port>`.

9. RLM Server: firewall considerations.
   If the RLM server is behind a firewall, the port 5053, 5054 and the ISV port have to be
   opened.
   The ISV port can be set directly in license file by changing the ISV line as follow:

   ```
   ... ISV lixoft port=<your ISV port> ...
   ```

10. Managing RLM server:
    The documentation of the management of the RLM server provided by Reprise Software
    is available at
2.7.3 Roaming license

RLM has the ability to allow a floating license to roam to a system which will subsequently be disconnected from the network for a short period of time. The resulting license can be used for the number of days specified when the license was set to roam, and is checked back in automatically at the end of this. In addition the user can return the roamed license back to license pool early if this is desired.

See License activate tools (which can be launched from the Monolix interface, in tools menu)

This feature is enabled on demand. An extra activation key will be provided by LIxoft and the procedure to get the roaming license feature is identical to the installation of a floating license. To enable this feature, the file system.xml (stored in directory monolix install path>/config/ -Matlab version- or monolix install path>/bin/Monolix_mcr/runtime/config/ -standalone version of Monolix - must be modified by setting to "on" the roaming option:

```xml
<?xml version="1.0" encoding="utf-8"?>
<monolix>
 <preference>
  <session>
   <userPath windows="%USERPROFILE%" linux="$HOME"/>
   <license activation="http://activate.lixoft.net" roaming="on"/>
  </session>
 </preference>
</monolix>
```
3 Troubleshooting

3.1 Downloading MONOLIX

**Problem:** My web browser claims that the MONOLIX download website has insufficient reputation and suggests to stop the download.

**Solution:** Some browsers like Google Chrome and Internet Explorer may ask whether to keep or remove the MONOLIX archive just after download because of the insufficient reputation of the MONOLIX download website, simply because it is not referenced, as opposed to the LIXOFT website. Please ignore the warning and choose to keep the file. You can use an MD5 tool to verify that the downloaded file is not corrupted.

**Problem:** The MONOLIX archive is removed just after being downloaded.

**Solution:** Some antivirus may consider the MONOLIX archive as risky and put it in quarantine or remove it. This is due to the fact that MONOLIX embeds a compiler for the MLXTRAN language. Two solutions are available:

1. Deactivate your antivirus auto-protection process during download and installation, or
2. Restore the file from the quarantine.

To restore the file from quarantine, please refer to the documentation of your antivirus software. For the most common examples:

- **Norton Antivirus 2012:**
  - Start Norton Antivirus
  - Choose Advanced, then Quarantine

- **Avast Antivirus 7:**
  - Open Avast
  - Choose Maintenance, then Virus Chest

You should see the downloaded file among the quarantined files. Execute the Restore action; the archive will be restored into the directory used for downloading. Click on the archive (ignore a possible “malware” warning, again related to the fact that MONOLIX embeds a compiler.), and installation will start.

3.2 Running MONOLIX

**Problem:** When launching the standalone version, my antivirus tells me that the file mlxinitializer.exe is risky.
3.2 Running Monolix

**Solution:** If your antivirus apparently removed the file mlxinitializer.exe, check if it was actually put on Quarantine, or removed. If it is in Quarantine, please restore it by following the same instructions as provided above. If the file was removed you will need to reinstall Monolix.

You should be able to add this file to your antivirus **Trusted Zone or Trusted files**.

- **Norton Antivirus 2012:**
  - go to folder Monolix/monolix421s/bin in installation directory: for instance c:/ProgramData/Monolix/monolix421s/bin
  - right click on mlxinitializer.exe, click on Norton Antivirus, then Norton File Insight then look for ‘Unproven’, and click ‘Trust Now’.

- **Avast 7:** This software may start Monolix in a SandBox, i.e in a zone where the antivirus avoids any modification of the system or the files. He will ask you what to do at each run. Select Run normally.
  You can also add mlxinitializer.exe to the exclusions in its Auto-Sandbox settings: option Additional Protection/AutoSandbox and then click on Settings button.
Monolix 4.2.1 (2013-02-15)

Bugs Fixes:
- MLXTRAN Project : in STRUCTURAL_MODEL section resolved problem
  with path relative to %MLXPROJECT%
- mlxEditor, mlxPerlScript : under Suse Linux OS, conflict with
  libstdc++ and Qt libraries installed on the OS.
- Graphics : Kaplan Meier
  - mean normalization
- survival curve: case of censored data
- simulations where wrong in presence of correlation between
  individual parameters
- MLXTRAN Model :
  - Events could be close at a numerical epsilon for the solver,
    but not for the solver driver
  Rarely, it resulted into an explicit integration failure, returning "NaN"
- For the simulation of RTTE models, the ordering of the output
  names had to be alphabetical
- Not declaring all regression variables that where selected
  from the data set crashed the application.
- Declaring some PK without actual doses within the data set
  raised an error.
- Using the deprecated syntax with several lagged compartments
  returned "NaN"
- Algorithms
- Error when some subjects had no doses in conditional mode
  computation
- GUI
  - "Display the data" button did not update the information when
    the dataset was changed after running algorithms
- Convergence assessment GUI failed when there were only one
  individual parameter
- structural models with several dots (.) were not compiled when
  clicking in the compile button in the Model selection GUI
- projects with more outputs in structural model than
  observations in dataset caused an error when it is loaded
- the editor was not saved in the preference file

Enhancements:
- add possibility to configure the compiler (used to create
  Structural Model plugins) through the file 'system.xmlx'
- user API:
  - it is possible now to use matlab function "ver" to know Monolix
    version and Monolix API version
- mlxEditor:
  - allow multiple files selection on open file dialog box
- add 'Find and replace'
- set tabs movable
- MLXTRAN Model:
  - Continuous observations can be declared within the model.
  - Macro for a depot absorption, with a target ODE component.
- Permutation kernel for mcmc included

Other:
- Licensing system: '.ini' files desactivated (only the '.lic' files are allowed)
- residual error models in main interface are shown now with their full name (those used in MLXTRAN project and model)

----------

Monolix 4.2.0 (2012-11-26)

Bugs Fixes:
- MLXTRAN Project: in OBSERVATION section when a prediction has the same name as an individual parameter the project parses fail
- PerlScript: bug with parameter '--use-matlab=false' was taken as 'true'
- Identity line works in observations vs predictions graphic
- Prediction distribution: percentiles are correctly displayed
- Color when stratify in covariates graphic
- Problem with prior (by default prior is Variance and not Standard Deviation, this implies a syntax error (standardDeviation <-> variance)
- Wrong data file for the demonstration project rttseWeibullCount_project.mlxtran
- "Display the data" button did not work
- bug when unchecking and checking "random effects" variability in simulation interface

Enhancements:
- Interval censoring for continuous data
- Extended priors on fixed effects
- Mlxtran model and Mlxtran project editor
- Perl script HMI
- Autosave
- Multiple covariate definitions
- Add batch-mode demo
- Add a doc package and a rlm server package (floating license server)
- Graphic
  - BLQ graphic: possibility to choose his own interval of censored data
  - Reorganisation of panel for list of graphics
  - Background color for each graphic in preferences
- When split, limits are the same for all axes
- Obs. vs Pred., observations can be relied by individual
- Optimal bandwidth setting for parameter distribution
- CvSaem graphic : choice of axes number
- Interval-censored data and maximum number of events for time-to-event and drop-out data models
- Markov chain for categorical data
- Continuous-time Markov process for categorical data
- Probit and normal cdf for MLXtran model
- New user API including simulation-estimation, convergence assessment and simulations tools
- Possibility to define new covariates as transformation of already defined ones

New graphics:
- Posterior and prior functions for bayesian
- Individual contribution for the LL
- Transition probabilities
- Kaplan-Meier survival function

New tables:
- Individual contribution to log-likelihood
- Covariates summary

---------

Monolix 4.1.4 (2012-07-16)

Bugs Fixes:
- Saving preferences from tools menu failed.
- Display remaining time (license) correctly.
- Problem with license activation file path.
- Add license agreement into Linux installer.
- The horizontal slider in "Check initial fixed effects" interface did not appear for some number of individual parameters.

Enhancements:
- Windows 64 RC.
- Management of the maximum number of threads for MLXTran models (can be set from the preference tools: MonolixGUI->Tools->Preference)
- License activate: inform user to not set activation key if the license is a floating license.
- Documentation :
  * Installation guide : Windows 64 bits.
  * User Guide : Cluster section revised.
  * Model MLxTRAN : list of keywords of the language.
Monolix 4.1.3-sp2 (2012-05-29)

Enhancements:
- system.xml : possibility to not display Lixoft Activate.
- Lixoft Activate : add the possibility to send an email with encoded computer information to create license @Lixoft.
- Lixoft Activate : manage "cannot connect to url" error by asking user to go on a web site or send an email.

Bugs Fixes:
- IOV Problem with R2010bSP1
- perlScripts : bug in the management of the configuration file for [program-execute-options] and run on a cluster.
- add 'rlmutil.exe' for windows packages (forgotten in previous packages).
- problem floating license.
- warnings for occasions without dose were removed.
- when the last Individual/Occasion had no dose, Monolix crashed.
- When there were syntax errors in the structural model, monolix said that it could not find the file instead giving the MLXTRAN message
- NaN observations are now mentionned as error when algorithms are launched.
- Update documentation : in batch mode section, there is a bad path.

--------------

Monolix 4.1.3-sp1 (2012-05-21)

Bug Fixes:
- GUI:
  * Check Initial Fixed Effects interface crashed when creating covariate and parameter's sliders for some sizes

--------------

Monolix 4.1.3 (2012-05-02)

New Features:
- MLXTran model: allows negative categories
- License management: uses RLM as license provider
- Compiler manager: adds the possibility to choose the embedded compiler
- The Monolix and Matlab versions are now stored in the algorithm result files
Bug Fixes:

- **MLXTRAN project:**
  - continuous transformation can take a mathematical expression
  - problem with structural model path

- **MLXTRAN model:**
  - Under Linux 64 bits, due to library conflicts with Matlab R2010b and better, the multi-threaded loading of the model description for the project occasionally fails
  - Only the last table variable is recorded, overwriting the first one

- **Graphics:**
  - log / linear works on all graphics
  - when log-log scale is set for "observed versus predicted", the diagonal line isn’t displayed anymore

- **GUI:**
  - editor call did not work

- **Algorithms:**
  * bug for individuals without some type of observations and with IOV computing conditional mode
  * bug when there were continuous outputs after discrete outputs
  * Fisher Information Matrix by Stochastic Approximation does now handle better the case when there are no parameters to estimate in the residual error

- **Session:**
  * when the directory monolixData/monolix<version> is renamed during an active Monolix session, stopping Monolix caused an exit of Matlab.

---

**Monolix 4.1.2: (2012-03-05)**

---

**New Features:**

- PerlScripts : possibility to save the results in the project directory instead of the output directory

- In system.xml : automatically creates a directory hierarchy for ‘monolixData’ path

---

**Enhancements:**

- MLXTran (structural model) multi-threading processing enhancement

---

**Bug Fixes:**

- Batch Processing failed when a very large number of projects were
- MDV column: when MDV=2 only the regression variables were taken into account
- Fixed a bug in graphics saving
- Fixed error when an empty result folder was timestamped
- Simulation of categorical data, whenever no category 0 is defined
- Fixed take into account UserPath defined in 'system.xmlx' for the preference file saving

----------

Monolix 4.1.1: (2012-02-13)

----------

New Features:
- timestamped backup
- preferences interface
- tools menu for activating license and preferences
- option for locking structural model modifications
- Project-MLXTRAN grammar modification: initialization of parameter
  is written now as beta\{pi,cov\}, pop\{pi\}, omega\{pi\}, ...
- save graphics as png / ps / jpg / bmp or tiff
- selection of graphics/tables to be saved

Bug Fixes:
- Project-MLXtran: user can define the result folder
- LoQ difference between 3.2 and 4.1
- statistical test for error model and covariate model
- xmlx loading from 3.2 to 4.1
- correlation (levelName consistency with IOV) + parser error
- observation model (prediction = observation name)
- path for MONOLIX user profile can include special characters

----------

Monolix 4.1.0: (2012-01-23)

----------

psmlx:
- compatible with the mlxtran format of projects
- available on Windows OS

mlxtran:
- new syntax
- PK macros
- RTTE models

license:
- interface for installing the license file

Interface:
- setting for axes' limits
- information for the observation model
- shortcut for model libraries

File system:
- improved handling of special characters for filepaths

Demos:
- updated for the new mlxtran syntax
- dispatch of the model library for demos

Known Bugs:
- under Windows OS, user directory cannot contain special characters other than spaces

--------------

Monolix 4.0.1: (2011-10-27)

--------------

psmlx:
- use-matlab option didn't work in command line mode
- multi-threading: multhreading didn't work
- take account the p-coded files

mlxtran
- problem with FIM options: both linearization and stochasticApproximation appeared after a save with stochasticApproximation option set
- avoid the unloading of project when settings files does not exist: default settings are loaded

license:
- multi write database didn't work well in multi-threading mode

Interface:
- save lists
- configuration panel
- launching some graphics alone is now possible
- the graphics were closed when "Use last estimates" were used
- when monolix was launched twice without loading or creating a project, two toolbars were created

Algorithm and simulations:
- simulation works now with dataset without EVID column and with MDV column

Results:
- the graphics fit now to the paper in .ps files
- xLabels were wrong for some graphics when several regression variables were present
- some graphics crashed when launched after some hypotheses tests were done
- Visual studio redistribuable problem