

Banco de Portugal's Microdata Research Laboratory

BPLIM: Internal Server Manual

September 12, 2021

Contents

1	Access to the Internal Server	2
2	Important guidelines	4
2.1	Keep your home area tidy	5
2.2	Using the Terminal	5
3	Statistical software	6
3.1	Stata	6
3.2	'batch' mode: an example using Stata	9
3.3	R	12
3.4	Julia	13
3.5	Updates to the commands and packages list	13
4	Allowed outputs	13
5	Remove outputs	13
6	User's Home folder	14
7	Scientific Support	14
8	Appendix	14
8.1	Basic 'shell' commands on Linux	14
8.2	Using the 'vi' file editor	16
8.3	Add information to the PATH	17

8.4	Frequently Asked Questions	17
8.5	Git	17
8.6	Singularity	18
8.7	Jupyter Lab	18

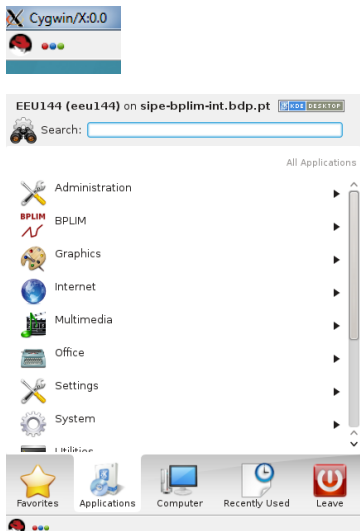
1 Access to the Internal Server

Upon access approval, a user will be able to connect to the Internal Server through remote desktop connection, which is installed on the desktop with a shortcut icon “**bplim-int**”. One can also obtain direct access to the project folder under “**pxxx_name**”, where “name” is the name of the user(s).

1. To access the server, click on the shortcut “**bplim-int**”, then enter the password.¹



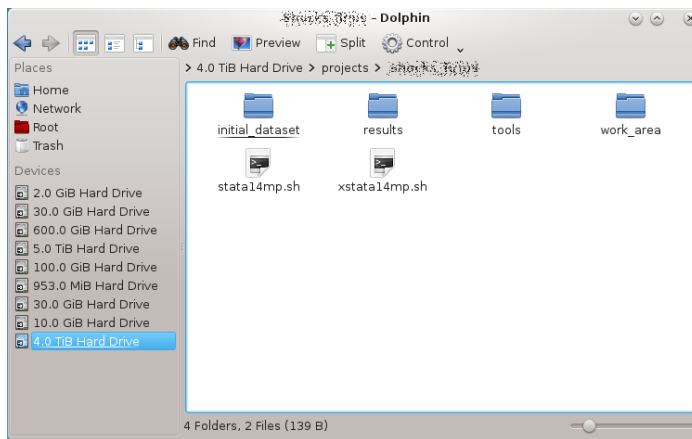
2. Select the “**Kickoff Application Launcher**” menu (in the lower, or upper, left corner):



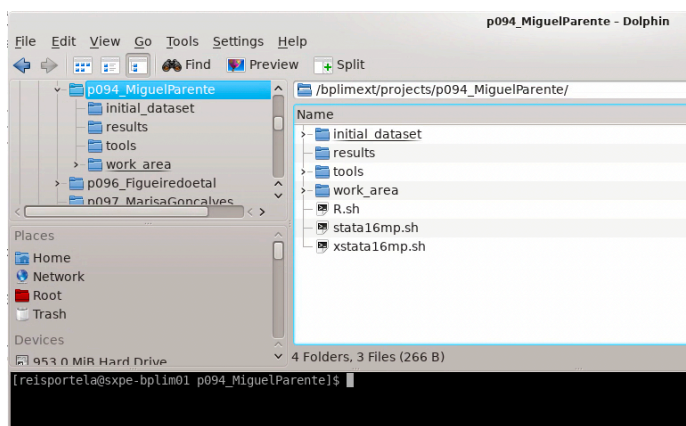
3. Then you should:
 - a. Click on the “**Applications**” button
 - b. Select “**BPLIM**” and click on your project (*i.e.*, “pxxx_name”). At this stage, you should see a graphical environment (Dolphin application²) like this:

¹The password is the same as the one that you use to access the internal network at the Banco de Portugal. External researchers accessing the data in the Internal Server will be assigned a username and password to access the project folder.

²Dolphin is an intuitive and easy-to-use file manager. You can use it, for example, to browse the directory, to cre-



You can see the command prompt line using the keyboard shortcut 'F4'.



- c. Files with the ".sh" extension allow you to send commands to your operating system or to enter your operating system for interactive use (for example, the file `xstata16mp.sh` will launch the graphical version of Stata 16). You can start the application by double-clicking the file name in 'Dolphin'³ or by typing in the Terminal `xstata16mp.sh`.

4. Move to the project's folder "`cd /bplim/projects/pxxx_name`" using the Terminal. The directories that you have access to within the folder include:

initial_dataset

Data sources provided by BPLIM.

You have read-only access to this directory.

results

Output files that researchers wish to generate and extract from the server.

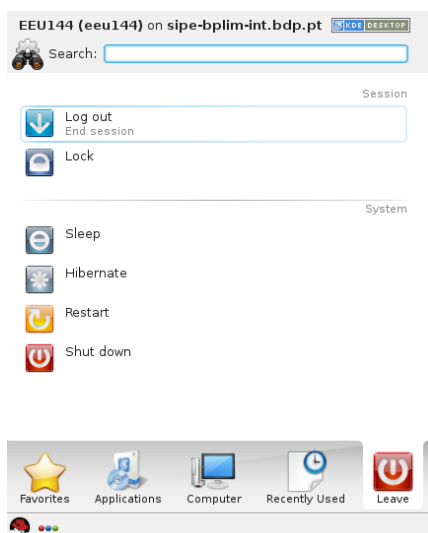
You have read-write access to this directory.

ate or to delete files/directories (by using the right mouse button). For more information about Dolphin, please visit: <https://userbase.kde.org/Dolphin>

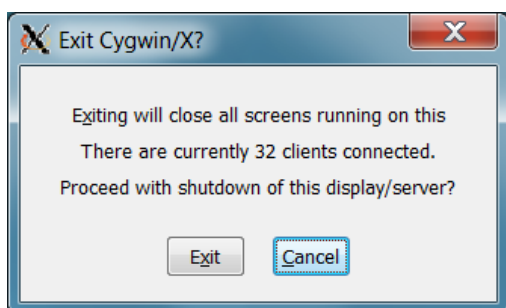
³In case 'xstata16mp.sh' does not launch Stata please see 'Section 3'.

tools	Specific analysis tools. <i>You have read-only access to this directory.</i>
work_area	Temporary work directory. <i>You have read-write access to this directory.</i>
/bplim/doc/Manuals	Manuals and auxiliary files are available here.

- You will have in your **work_area** folder templates for both Stata and R. By default the template file is read only.
5. To reset and disconnect the remote desktop connection or session, you can simply log out your remote session, as shown on the screenshot below. After you log out, close the window.⁴



Confirm before exiting by clicking on the **"Exit"** button to close the window⁵



2 Important guidelines

⁴Click on the cross button at the upper right corner to close.

⁵Note that before exiting the server, you need to make sure that all active programs have been closed (unless they have been launched in *batch* mode). Running programs in *batch* mode is justified for procedures that require high computational resources, intense calculation and / or long processing time.

2.1 Keep your home area tidy

- *Do not save files in your home area /home/USER_LOGIN. In case you exceed its size you will not be able to log in.*
- Check regularly the size of your project in the harddrive. Open a Terminal and apply the following steps:
 1. Move to the project folder: `cd /bplim/projects/p000_xxx_yyy/`
 2. List project size: `du -h`
 3. Check size by folder and list folders with at least 1Gb: `du --max-depth 1 -h | sort -n | grep G`
 4. Move to the folder 'work_area': `cd work_area`
 5. Repeat the check in this folder: `du --max-depth 1 -h | sort -n | grep G`
 6. Identify duplicated and temporary files and delete them: use the command `rm`
 7. Compress big files/folders you are not using at the moment:

Compress folders: `tar -zcvf YOUR_FOLDER.tar.gz YOUR_FOLDER`

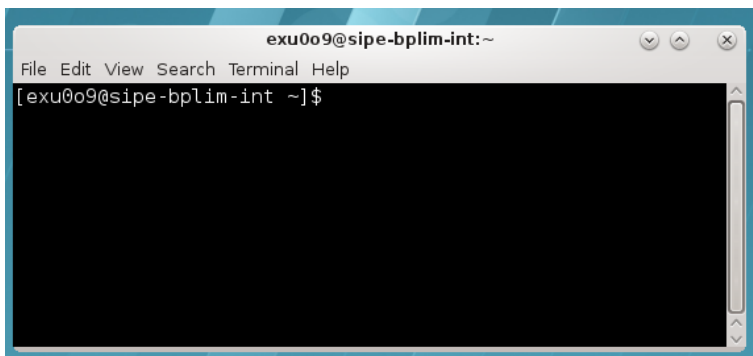
Compress individual files: `gzip YOUR_FILE`

2.2 Using the Terminal

Linux's Terminal is a command-line interpreter. You can use the 'shell' for a wide range of tasks, including searching files and files' contents, organizing your working space, and, most importantly, run your programs in batch mode.

1. Linux's Terminal can be accessed from⁶

RedHat > Applications > System > Terminal



⁶The *shell* supports the commands in Linux operating system (some are disabled).

2. See [Section 8.1](#) for a list of some of the most used commands.
3. In case you are using a non-English keyboard, the ‘true’ keyboard might be different from the one you see. The changes apply mostly to the symbols, not letters or numbers. For example, in case you have a Portuguese keyboard on your computer the ‘+’ is now in key ‘?’, or the ‘*’ is in SHIFT + ?. This issue is specific to the Operating System of your computer
4. Remember that Linux is case-sensitive: e.g., “LS” and “ls” are treated as different commands.
5. You can use the arrow keys to scroll up and down through the commands you have entered.
6. You can use the “Tab” key to complete the command line automatically.
7. *e.g.*, type the following line to list elements within a folder in a ‘human-readable’ format, *h*, long list format, *l*, in reverse order, *r*, sort by modification time, *t*, and almost all files, *A*,

```
ls -lArth
```

3 Statistical software

3.1 Stata

Stata versions available in the server: 14, 15, 16 and 17 (adjust the following lines to the Stata version you want to use)

1. Stata can be accessed in interactive graphical or non-graphical modes.⁷
 - Interactive non-graphical mode

Move to the desired folder, e.g.,

```
cd /bplim/projects/I001_jdoe/
```

and type

```
/opt/bplim/stata16/stata-mp
```

⁷The version of Stata on the server has the same features as the Stata on Windows or Mac. By default when Stata starts in this way the active “working directory” active becomes your folder “work_area”.

```

exu009 : stata-mp - Konsole
File Edit View Bookmarks Settings Help
. sysuse auto
(1978 Automobile Data)
r; t=0.00 17:36:05

. sum

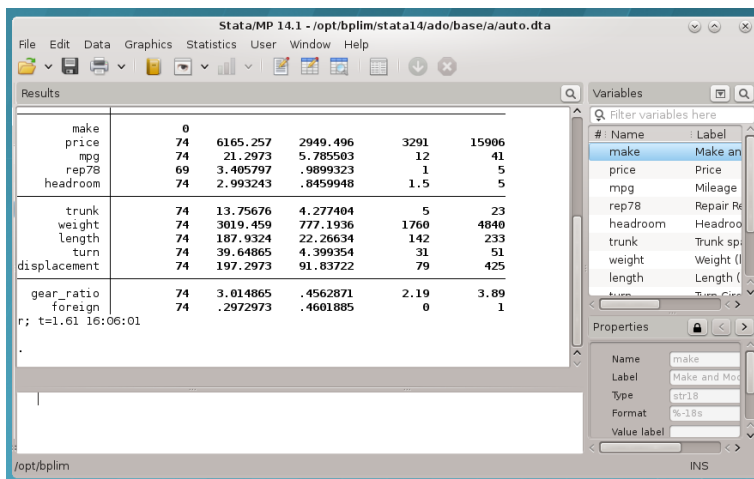
Variable |      Obs      Mean   Std. Dev.   Min    Max
-----+-----
make      0
price     74   6165.257   2949.496   3291   15906
mpg       74    21.2973    5.785503    12     41
rep78     69    3.405797    .9899323     1     5
headroom  74    2.993243    .8459948    1.5     5
trunk     74    13.75676    4.277404     5     23
weight    74   3019.459    777.1936   1760   4840
length    74   187.9324    22.26634    142    233
turn      74   39.64865    4.399354     31     51
displacement 74   197.2973    91.83722     79    425
gear_ratio 74    3.014865    .4562871    2.19    3.89
foreign   74    .2972973    .4601885     0     1
r; t=0.00 17:36:08

```

- You may add a ‘PATH’ to your system folder by typing, for the example on Stata 16, the following command in the shell “`vi ~/.bash_profile`” and adapt the following line

`PATH=$PATH:$HOME/.local/bin:$HOME/bin:/opt/bplim/stata16`

- For the interactive graphical mode click on the icons “**xstata16mp.sh**” (Stata 16) located in the ‘desktop’, depending on the desired Stata version,



- You can use the ‘Do-file Editor’ in Stata to create your own “do-files” and “ado-files”, or you can use *KWrite* editor (or ‘gedit’)
- You can open it from **RedHat > Applications > Utilities > KWrite**. You can also launch ‘KWrite’ from the ‘shell’ by typing ‘kwrite’
- In case the icon is not in your desktop, use Dolphin, move to folder ‘/bplim/scripts/wrappers/’, and drag and drop the file ‘xstata-mp’ corresponding to the Stata version you want into the

desktop

2. To look for “**ado-files**”:

“Ado-files” are text files containing the Stata program. It is advisable that one create and save his/her “ado-files” so the results can be replicated later by running the saved “ado-files” on the BPLIM’s datasets.

Stata looks for “ado-files” in several places. When it comes to personal ado-directories, they can be categorized in four ways:

- (SITE), the directory for “ado-files” your site might have installed;
- (PLUS), the directory for “ado-files” you might have installed;
- (PERSONAL), the directory for “ado-files” you might have written;
- (OLDPLACE), the directory where Stata users used to save their personally written ado-files.

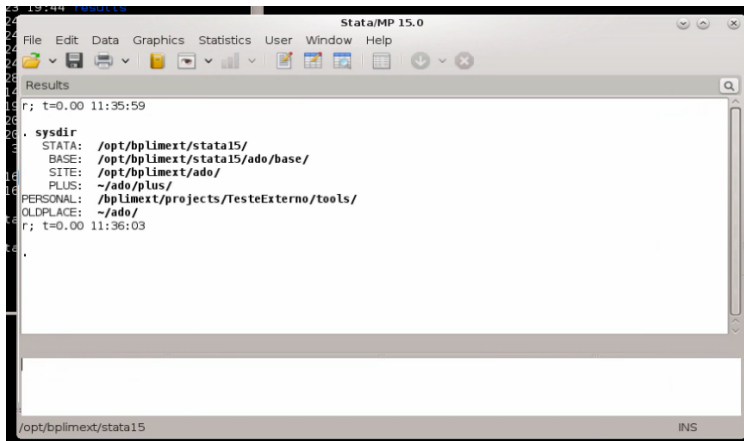
The ado-files you have just written or those created for this project can be found in the current directory (.).

Specific ‘ado-files’ you may ask to be made available in the server will be placed in your folder ‘/bplim/projects/YOURPROJECTID/tools’. You should add this folder to your Stata ‘ado-files’ folder by executing the following command within Stata,

```
sysdir set PERSONAL "/bplim/projects/YOURPROJECTID/tools"
```

You may also edit your ‘profile.do’ file, located in your root folder, “/home/YOURPROJECTID/”, and add key instructions you may want to be executed every time you start Stata. The above instruction is one of such cases. You can create, or edit, the file ‘profile.do’ using ‘Do-file Editor’ within Stata (‘vi profile.do’ or KWrite are also a possibility).

The ***sysdir*** command within Stata will tell you where they are on your computer:



3. Stata's temporary files:

Check the location of the folder with temporary files

- Open a Terminal and type

```
tempfile junk
```

```
display "'junk'"
```

In the work_area of your project add a folder named tmp

Using the Terminal, go to your home folder, `cd ~`, and open the file `.bashrc` and add the following lines:

```
STATATMP="/bplim/projects/YOUR_PROJECT_ID/work_area/tmp"
```

```
export STATATMP
```

In the Terminal type `source .bashrc` and call Stata from the Terminal (`xstata-mp`).

3.2 'batch' mode: an example using Stata

1. Start a '*shell*' in Linux and navigate to the directory of the "do-file" file that you want to run (ex: `progl.do`)

```
cd /bplim/projects/I001_jdoe/work_area/
```

2. You might find it easier to use 'Dolphin' (= File Manager) to move over your folder structure.

In this case, we recommend activating the 'shell' (= 'Terminal') associated with 'Dolphin'

- use Dolphin/File Manager

- click 'F4' to activate the shell with Dolphin. Benefit: fast transition within folders and, at the same time, the ability to run shell commands

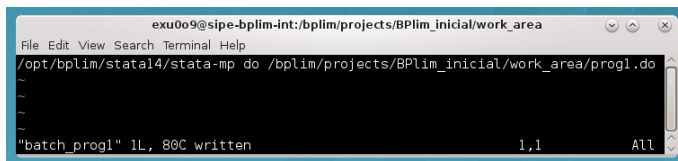
3. Create an ASCII file named, e.g., 'batch_prog1'

4. Inside the file write just a line with the execution command you would type in the 'shell'; e.g.,

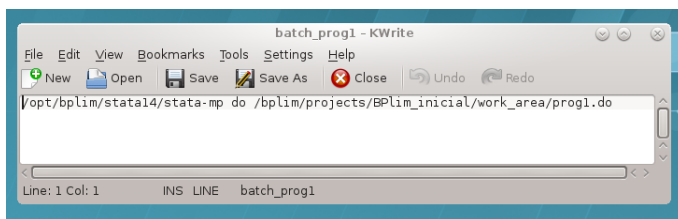
```
/opt/bplim/stata16/stata-mp do
```

```
/bplim/projects/I001_jdoe/work_area/prog1.do
```

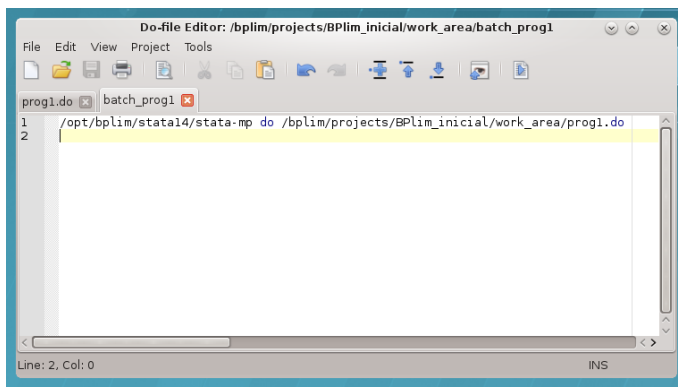
5. You can use, for example, the command line app 'vi' to create the batch file



6. The batch file can also be created using apps like 'kwrite' or Stata 'do file editor'



OR



7. You may add the extension '.txt' to the name of the batch file, as sometimes Stata *doeditor* does not 'see' the file 'batch', while it 'sees' 'batch.txt'

8. Once the batch file is created one runs the .do file in batch mode by typing in the 'Terminal':

```
at now -f batch.txt
```

9. Type 'man at' to see a further option of the command 'at'; e.g., one could type

```
at now + 5 hours -f batch.txt
```

or

```
`at now + 4 minutes -f batch_prog1`
```

to run the Stata program within 5 hours or 4 minutes from now, respectively. ‘man’ is the help function in Linux

10. Type ‘top’ in the shell/Terminal to confirm the program is running
11. Under ‘top’ type ‘i’ to hide irrelevant processes (show less output)
12. To kill a running process with ‘top’ press ‘k’, for ‘kill’, write > the process number and then type ‘9’. The process number is > identified in the first column as PID
13. To get out of the top, type ‘q’
14. Useful features of the command ‘at’:
 - ‘atq’: use it to see programs in the batch queue (an ‘=’ sign indicates the program is running; an ‘a’ indicates it is in the queue and we see the time when it will be executed)
 - ‘atrm #’: remove a batch from the batch queue
 - one can see how the batch is running by typing

```
tail --f logcrc_may21.log
```

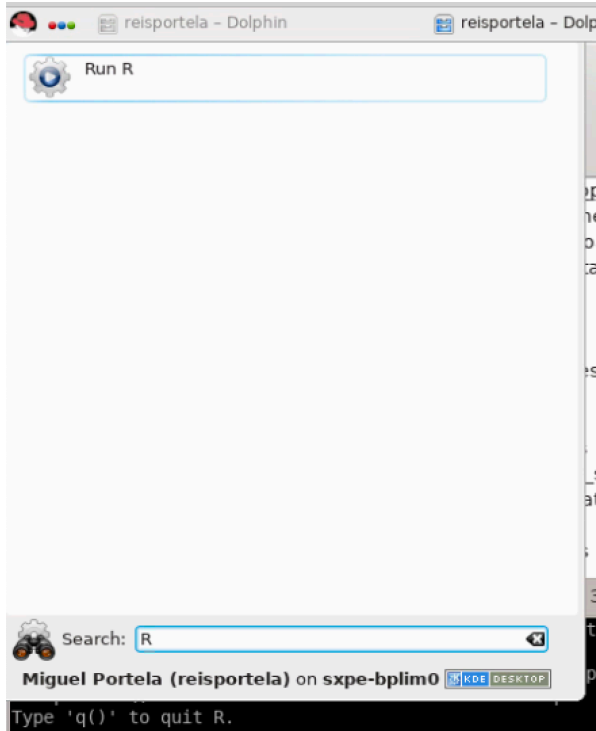
It allows you to see an updated version of the last lines of the log; *i.e.*, it updates each time the log is changed by Stata. A key advantage of tail is that it does not interfere with the log file, namely, it does not write over it.

15. Another way to run a program in the background is by using the command ‘screen’
 - ‘screen’ is useful when one wants to run Stata in interactive mode and still guarantee that if the network connection goes down one does not lose the session. We can simply kill the ‘NoMachine’ session and recover it later by typing ‘screen --r’
 - We can run several instances of screen. If this is the case, after opening a new NoMachine session we need to type in the Terminal shell ‘screen -d’ to identify the running background sessions. We can retrieve a particular session by knowing the ‘pid’ number and typing ‘screen -r 34176’

3.3 R

1. As with Stata, R can be accessed in interactive graphical or non-graphical modes.

- Interactive non-graphical mode: go the RedHat symbol and type 'R' in the Search box



- Alternatively, you can open a Terminal and type

R

- Please make sure R is in your PATH; type \$PATH in the Terminal. If this is not the case, type
PATH=\$PATH:/usr/bin/

2. Using RStudio.

- Open a Terminal and type

rstudio

- Please make sure RStudio is in your PATH; type \$PATH in the Terminal. If this is not the case, type

PATH=\$PATH:/opt/bplim/R/usr/lib64/rstudio/bin/

- In case you face difficulties opening/saving files in RStudio, please open a Terminal and type

/bplim/scripts/wrappers/R.sh ## Python

1. Open a Terminal and type

```
python3
```

3.4 Julia

1. Open a Terminal and type (julia is located in /opt/bplim/julia/lib/, you can add it to your PATH)

```
julia
```

2. Use Atom: open a Terminal and type

```
atom
```

3.5 Updates to the commands and packages list

Additional commands/packages or updates to the existing ones have to be requested to BPLIM's Team.

4 Allowed outputs

Stata results can be extracted to a file on disk using one of the following formats:

1. ASCII files: e.g., log files
2. graphs: as .PNG
3. csv: CSV (Comma Separated Value format), e.g., for use with MS Excel
4. rtf: Rich Text Format for use with word processors
5. xls or xlsx: Excel files with output tables
6. tex: Latex format

5 Remove outputs

Place in the “results” folder all the outputs you want to remove from the server.⁸

1. Send an email with the title “**project p001_jdoe**: request for result extraction” to “**bplim@bportugal**”.

⁸You may only remove text files that do not contain data or results that allow identification of individual units. For all the graphs you request as an output you must provide the corresponding Table to replicate it. You may only export graphs in .PNG format (no vector graph is allowed).

2. Upon validation, the results will be sent to you via email.

6 User's Home folder

1. Do not save files in your Home folder: "/home/USER_ID/".
2. Regularly clean your Trash folder. If your disk use goes over the quota you will be prevented to login. In the Terminal type:

```
rm -rf ~/.local/share/Trash/*
```

7 Scientific Support

Researchers will be provided with the necessary scientific and computational support (*i.e.*, advises on programming, computational resources, micro econometrics, and econometrics of panel data for research undertaken with the selected microdata).

8 Appendix

8.1 Basic 'shell' commands on Linux

- top: List the procedures that are being executed on the server
 - press 'i' option to omit background processes;
 - clicar press 'h' para **help on top options** ; 'h' > option to obtain the **top command help**.

- pwd: Show current working directory

- cd: Change directory

```
cd /bplim/projects/I001_jdoe/work_area/
```

'cd ~' moves to your home folder

- cp: Copy file(s) to a given path

```
cp prog1.do /bplim/projects/I001_jdoe/results
```

- mv: Move file(s) or rename a file from a given path

```
mv prog1.do /bplim/projects/I001_jdoe/results
```

- `rm`: Delete a file

```
rm /bplim/projects/I001_jdoe/results/prog1.do
```

- `mkdir`: Creates a directory

```
mkdir programs
```

- `rmdir`: Deletes a directory

```
rmdir programs
```

- `screen`: Switch between screen

```
screen top
```

- `man`: Show the manual page for the given command

```
man ls
```

- `du -h`: Check the information of disk usage of files and directories.

The “-h” option with “du” command provides results in “Human Readable Format”.

Ex: `du /bplim/projects/I001_jdoe/work_area/`

- `df -h`: Check disk space utilization and show the disk space > statistics in “human readable” format.

- `vi`: View ‘ASCII’ files; e.g., log files

- `ghostscript`: Preview files with the extensions of *.eps* and *.pdf*

```
ghostscript /bplim/projects/I001_jdoe/results/file_name.pdf
```

- `okular`: View ‘PDF’

- `find`: Find files

Structure: `find /path option filename`

```
find . -name "*.do"
```

Send the ‘find’ output to a file:

```
find . -name "\*.do" > find_results.txt
```

Look for a particular string within the ‘find’ output:

```
find . -name "\*.do" | grep "analysis"
```

Identify files with extension '.do' that **contain** the word 'graph':

```
find . -name "\*.do" -exec grep "graph export" '{}' \; -print
```

- passwd: Change your password
- **To exit** a program, type **CTRL + C** ('CTRL + C' kills a particular execution in the shell)

8.2 Using the 'vi' file editor

1. In the shell type 'vi batch1.txt'
2. These are the main shortcut keys
 - a. 'i' insert text
 - b. 'ESC' key get out of the 'insert' mode
 - c. 'x' delete specific characters
 - d. 'dd' delete a full line
 - e. '10 dd' delete 10 lines
 - f. 'yy' copy lines
 - g. 'p' paste lines
 - h. 'SHIFT + G' go to the last line
 - i. 'gg' goes to the first line
 - j. 'ESC + q!' exit 'vi' without writing
 - k. 'w!' write and replace the file
 - l. 'ESC + q' exit the 'vi' session
 - m. Check, for example, <https://www.cs.colostate.edu/helpdocs/vi.html>
3. Much easier solution: call 'gedit' file editor
4. Linux commands I have to add to the manual
5. 'CTRL + R': allows me to recover a previous command

6. `vi .bash_history`

8.3 Add information to the PATH

1. Open a Terminal
2. Type `'vi .bashrc'`
3. Press `'i'`
4. Add the following line at the end of the script:

```
export PATH=$PATH:/opt/bplim/stata16/: /opt/bplim/R/bin/:/opt/bplim/R/usr/lib64/rstudio/bin/  
:/opt/bplim/julia/bin/:/opt/bplim/R/
```

5. You can now run, for example, R by typing in the Terminal `'R'`
6. Likewise for `'julia'`, `'stata-mp'`, `'xstata-mp'`, `'rstudio'` or `'python'`

8.4 Frequently Asked Questions

1. Visualizing \LaTeX tables

- In case you want to see the pdf of tables you have exported to \LaTeX you can create a generic tex file, `main.tex`, with the following content:

```
\documentclass{article}  
  
\begin{document}  
  
\input{your_table.tex}  
  
\end{document}
```

where your table is `'your_table.tex'`. The tex file can be compiled in the Terminal typing `pdflatex main.tex`.

8.5 Git

You can use version-control. The server runs [Git](#) version 1.8.3.1.

[Wikipedia:](#)

“Git is a distributed version-control system for tracking changes in any set of files, originally designed for coordinating work among programmers cooperating on source code

during software development. Its goals include speed, data integrity, and support for distributed, non-linear workflows”

First steps

1. move to a specific folder; *e.g.*, `cd /bplim/projects/your_project_ID/work_area/`
2. `git init`
3. create a `.gitignore` file (check [toptal](#) for some examples)
4. `git add *.do`
5. `git commit -a -m First`
6. `git show first_do_file.do`

8.6 Singularity

You can use Singularity to run the Anaconda environment inside the server. The image for Singularity is placed in:

```
/opt/bplim/singularity-images
```

Explore the following example. Type these sequence of commands:

```
cd /bplim/projects/your_project_ID/work_area
```

```
mkdir containers
```

```
cd containers
```

```
singularity build --sandbox Anaconda /opt/bplim/singularity-images/bplimAnaconda.sif
```

```
singularity shell --writable Anaconda
```

Now you are inside the container and can run commands such as:

```
anaconda-navigator
```

```
jupyter lab
```

```
spyder
```

8.7 Jupyter Lab

Explore [Jupyter lab](#):

“JupyterLab is a web-based interactive development environment for Jupyter notebooks, code, and data. JupyterLab is flexible: configure and arrange the user interface to support a wide range of workflows in data science, scientific computing, and machine learning. JupyterLab is extensible and modular: write plugins that add new components and integrate with existing ones.”

Start Jupyter Lab by typing:

```
jupyter lab
```

Sample session:

