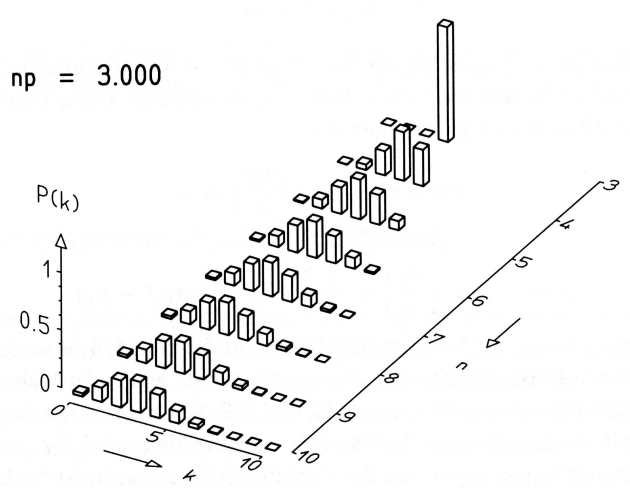
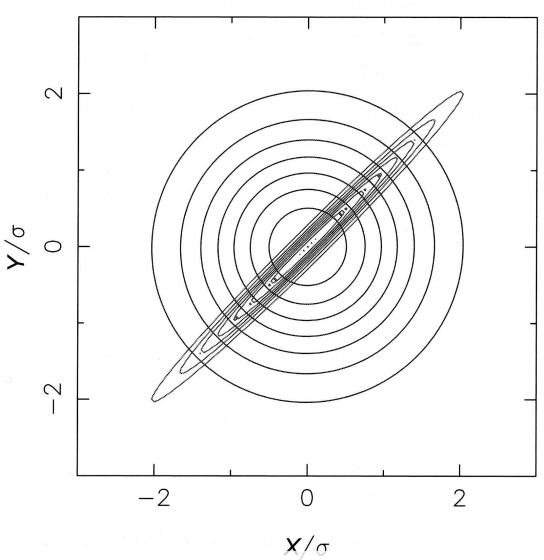
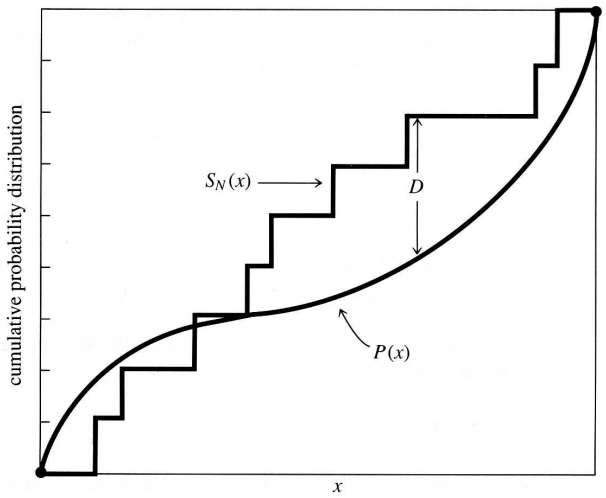






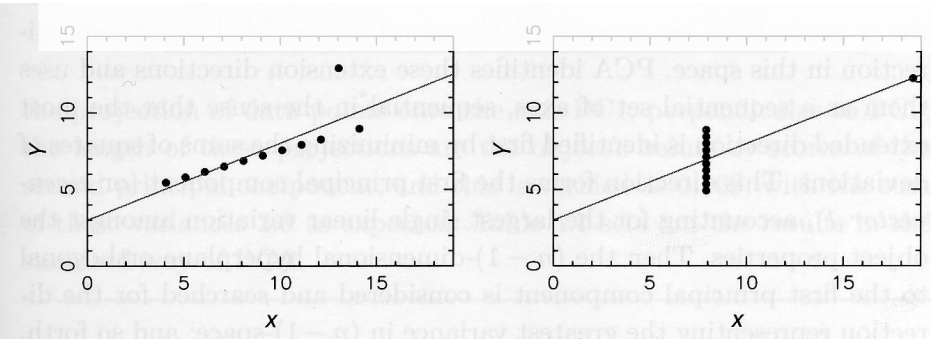
Statistical methods (UKSta)

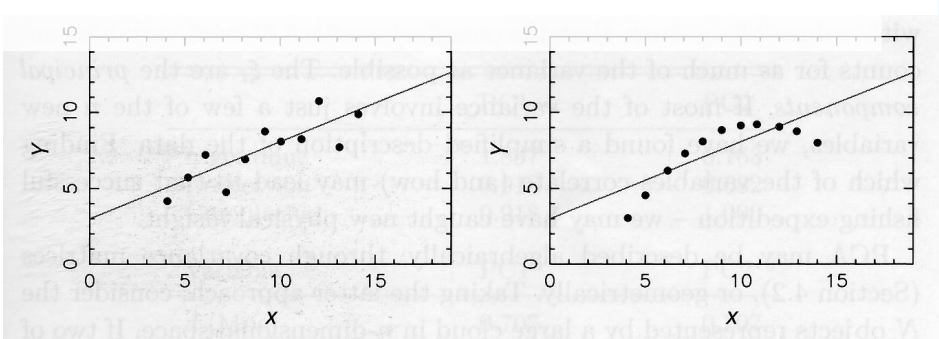
Linux and Emacs

Dr. Hans-Günter Ludwig

Summer term 2018

(original lecture by N. Christlieb)





Computer access

- Log in with your URZ account.

Linux

- Unix-like operating system invented by Linus Torvalds in 1991.
- Freely available; open source (GNU General Public License).
- Many things can most effectively be done or can be done *only* via the command line, hence you need to learn some of the basics.
- A GUI let‘s you point, a command line let‘s you talk!

Exercise 01: Linux tutorial

- Start a web browser via the menue
  -> Anwendungen -> Internet -> Firefox Web Browser
- Read the following chapters of the Linux tutorial in the order listed below:

  www.linux-tutorial.info/modules.php?name=MContent&pageid=49
  www.linux-tutorial.info/modules.php?name=MContent&pageid=51
  www.linux-tutorial.info/modules.php?name=MContent&pageid=15
  www.linux-tutorial.info/modules.php?name=MContent&pageid=16
  www.linux-tutorial.info/modules.php?name=MContent&pageid=17
  www.linux-tutorial.info/modules.php?name=MContent&pageid=21
  www.linux-tutorial.info/modules.php?name=MContent&pageid=30
  www.linux-tutorial.info/modules.php?name=MContent&pageid=55
  www.linux-tutorial.info/modules.php?name=MContent&pageid=31
  www.linux-tutorial.info/modules.php?name=MContent&pageid=32

- To try out some of the commands described in the tutorial, start a terminal (search among the applications!)

Create a workspace for the exercises

- After reading the tutorial you should now be able to create directories and navigate between them (i.e. set your current **working directory** to that path) on the command line. Create one top-level directory for this course, and within this a subdirectory for the first day:

```
$ cd                  # change to home directory
$ mkdir UKSta18       # create directory
$ cd UKSta18          # change to new directory
$ mkdir Monday        # create subdirectory
$ cd UKSta18          # change to subdirectory
```

Reading and printing PDF files

- Note: since Linux is an open operating system, there are often software contributions from many different sources, and multiple ways to perform a given task.
- You can open a PDF from the desktop menu, or start one of the installed viewing programs from the command line, e.g.:

```
$ acroread filename.pdf &
```

- In the viewer look for the menu ‘Datei -> Drucken‘; choose the printer named “**xxx**”. This is the black & white laser printer which is located in the back of the room.

DO NOT SEND ANY FILES TO “physi”!!!

File editing

- Since you frequently have to edit files during the R exercises to write your own code, you need to learn how to use one of the standard editors provided in Linux.
- There is a “religious” fight about whether emacs, vi, or whatever else is the “one true editor” under Linux. We will use Emacs here.

Exercise 02: Emacs tutorial

- Download the example file "square.R" from the course webpage and save it to the "Monday" directory you have created before.
- In the terminal window make sure you are still in that subdirectory; otherwise you can directly navigate to it at the command line prompt:

```
$ pwd                          # print working directory
$ cd ~/UKSta18/Monday          # change directory
```

- If the file should still be in the default download directory, you can move it to your working directory with

```
$ mv ~/Downloads/square.R .  # "." represents current dir.
```

- Load the downloaded file into Emacs:

```
$ emacs square.R &
```

- Emacs should now open in its own window.

Emacs tutorial (continued)

- Start the Emacs tutorial by either clicking on “Emacs Tutorial”, or by selecting this item from the “Help” menu in the text menu bar at the top (the latter also offers to start the tutorial in different languages).
  Pay especially attention on how to
  - read-in and write-out files
  - create and switch between different file buffers
  - navigate efficiently in a text (there is a multitude of key combinations that initially might appear confusing, but many commands can also be effected with the standard arrow/PgUp/PgDown keys and the scroll wheel
- For a listing of the most important Emacs commands, see the “GNU Emacs Reference Card” (PDF in Moodle).

First steps in Emacs

- When you started the tutorial, our originally opened file “square.R” will have disappeared. It is however still loaded in the editor. Find out how to get the text back into the editor window using keyboard commands (hint:while you can access it through the “Buffers” text menu as well, this will also show you reminders of the corresponding keyboard shortcuts).
- This file contains your first lines of **R** source code defining a function that calculates the square of a number. Even without knowing any **R** syntax at this point, it should be obvious how to modify the function such as to compute the cube (3rd power) instead.

First steps in Emacs

- After you have modified the source code, save it into a new file “cube.R”:
  - `C-x C-w path-and-name-of-file`
- Now we will learn how to make some simple customisations to Emacs to adapt it to our own taste; these are by default written in a hidden file “.emacs” in your home directory:
  - `C-x C-f ~/.emacs`
- If “~/.emacs” did not previously exist, this will have opened a new empty file, otherwise navigate to the end of the file buffer to make any new additions. “~/.emacs” is read every time Emacs is newly opened.
- The buffer of the previously edited “cube.R” is now again replaced by the new buffer.

First steps in Emacs

- To switch back to the last buffer in a single step you can use the command “bury-buffer”; like any Emacs command not assigned to a specific key this can be accessed with the `M-x` shortcut (the “Meta” modifier is the Option, Alt or Escape key).
  - `M-x bury-buffer`
  - `M-x unbury-buffer`
- Repeatedly using the “bury/unbury” combinations lets you move though the whole sequence of currently opened buffers.
- You can also view buffers simultaneously by editing them in independent windows (“frames”), using the “Frame” commands in the “File” menu.

First steps in Emacs

- To access these functions more easily, you can “bind” them to a shortcut of your own choice; here we will use the combinations “Ctrl[+Shift]+tabulator key”:

```
(define-key global-map [C-tab]   'bury-buffer)
(define-key global-map [S-C-tab]   'unbury-buffer)
```

- After writing these lines or any other customisations into your “~/.emacs” file it is wise to first test the code to make sure you do not accidentally break your Emacs installation; for this you can use the menu item “Emacs-Lisp→Evaluate Buffer” or just type:
  - `M-x eval-buffer`
- Note that you can always hit the tabulator key after starting to write anything to let Emacs automatically complete the text.