

## Howto: Run STRUCTURE on the server (command line version)

Somehow running STRUCTURE on the server is slightly more difficult than other software. But you can make it.

### 1. Modify *.bashrc* on *evo-lutra*

In order to run the command line version of STRUCTURE on *evo-lutra*, you'll first need to modify one certain file on *evo-lutra* (*.bashrc*). The reason for this is that whenever you type a shell command (e.g. *structure inputfilename*), the system needs to know which software to start. It searches for an executable file called "structure" in a list folders (this list is called *\$PATH*). However, the folder which contains the STRUCTURE executable is not yet in this *\$PATH*, so we'll have to include it first.

Connect to *evo-lutra* using Secure Shell. I explained this in detail in the Howto: Run software on the server (general). Here's the short version: Use the Terminal and type:

```
ssh yourusername@evo-lutra.zoo.unibas.ch
```

and enter your password. Then type:

```
ls -a
```

You know the *ls* command already. Using the *-a* option, hidden files (those starting with a period) are also shown. The file *.bashrc* is included in the list:

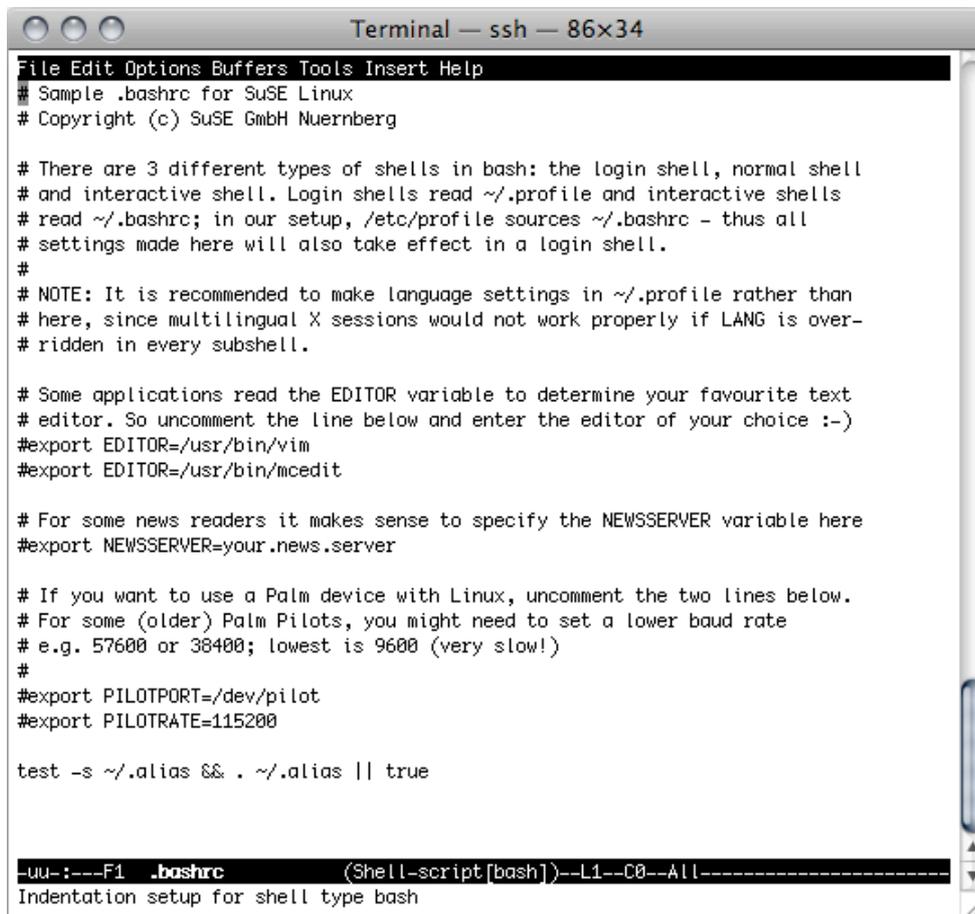


```
Terminal — ssh — 77x13
micham@evo-lutra:~> ls -a
.          Documents  .gnu-emacs  .profile   .xcoralrc
..         .dvipsrc    .inputrc   public_html .xemacs
AppleVolumes .emacs      .kernrc    .ssh       .xim.template
.bash_history .emacs.d    .lesshst   .structure .xinitrc.template
.bashrc     .exrc      .mozilla   .urview    .xtalkrc
.bashrc~    .fontconfig .muttrc    .viminfo
bin         .fonts     .mysql_history .vimrc
.bup_exclude .forward   PRIV_SHARE .xauthority
micham@evo-lutra:~>
```

Use the command line text editor *emacs* to modify *.bashrc*. To do so, type:

```
emacs .bashrc
```

You'll see the following:



```
Terminal — ssh — 86x34
File Edit Options Buffers Tools Insert Help
# Sample .bashrc for SuSE Linux
# Copyright (c) SuSE GmbH Nuernberg

# There are 3 different types of shells in bash: the login shell, normal shell
# and interactive shell. Login shells read ~/.profile and interactive shells
# read ~/.bashrc; in our setup, /etc/profile sources ~/.bashrc - thus all
# settings made here will also take effect in a login shell.
#
# NOTE: It is recommended to make language settings in ~/.profile rather than
# here, since multilingual X sessions would not work properly if LANG is over-
# ridden in every subshell.

# Some applications read the EDITOR variable to determine your favourite text
# editor. So uncomment the line below and enter the editor of your choice :-)
#export EDITOR=/usr/bin/vim
#export EDITOR=/usr/bin/mcedit

# For some news readers it makes sense to specify the NEWSERVER variable here
#export NEWSERVER=your.news.server

# If you want to use a Palm device with Linux, uncomment the two lines below.
# For some (older) Palm Pilots, you might need to set a lower baud rate
# e.g. 57600 or 38400; lowest is 9600 (very slow!)
#
#export PILOTPORT=/dev/pilot
#export PILOTRATE=115200

test -s ~/.alias && . ~/.alias || true

--uu-:---F1 .bashrc (Shell-script[bash])--L1--C0--All-----
Indentation setup for shell type bash
```

Move the cursor to the end of the file, below “test -s ~/.alias...”, and type:

```
export PATH=/usr/local/Structure/bin:$PATH
```

Close emacs by typing the key combination

Ctrl-X, Ctrl-C

before closing, emacs will ask you whether you want to save the file (in the window’s bottom line).

Confirm with “y”.

Close the Terminal window and open a new Terminal session. Log back in to evo-lutra. See whether the system knows now where structure is by typing

```
which structure_cl
```

(where “\_cl” specifies the command line version). You should receive the answer

```
/usr/local/Structure/bin/structure_cl
```

## 2. Copy your input file and the parameter files to evo-lutra

I explained this in detail in the Howto: Run software on the server (general). In addition to your input file, STRUCTURE 's command line version needs two parameter files, called mainparams and extraparams, where you specify how many generations to run, whether to use the admixture model, and so on. Find out about this in the STRUCTURE documentation, on pages 17 and 18 ([http://pritch.bsd.uchicago.edu/software/structure2\\_2.html](http://pritch.bsd.uchicago.edu/software/structure2_2.html)). Note that even though you define the number of assumed clusters (K) in the file mainparams, you can override this setting from the command line by adding the option -K , followed by whatever your K should be (e.g. -K 5). This will be useful because you sure don't want to modify mainparams after every run. See below.

## 3. Start the run

Use the Terminal and log back in to evo-lutra, if you logged out before. Navigate to the folder which contains your input file. Input file, output file and all the parameters (including K) are specified in the two parameter files mainparams and extraparams anyway, so you could simply start STRUCTURE with

```
structure_cl
```

However, it's much more efficient to start all runs with a single command the following way (with K, in this example between 1 and 3):

```
structure_cl -K 1 -o output_1.txt; structure_cl -K 2 -o output_2.txt; structure_cl -K 3 -o output_3.txt
```

It's important to specify different output files, otherwise the output file of the first run will be overwritten after every run.

## 4. Graphics

Use bar\_plotter.rb to create vector graphic barplots from STRUCTURE's output files. You'll find the script and a manual on the Salzburger lab homepage, in the Software section (<http://www.evolution.unibas.ch/salzburger/software.htm>). Don't forget to cite the author!