

From Novice to Expert in Bioinformatics

Linux, Perl and R for Bioinformatics

Shaojun Xie

2017-11-20

Contents

Preface

Who should read this book?

How to read this book?

The reason why I didn't write in Chinese?

About me

Acknowledgement

Disclaimer

Chapter 1

Why Linux?

1.1 What is Linux

1.2 Linux for bioinformatics

Chapter 2

Connecting to Linux

2.1 User interfaces

2.2 How to connect

2.2.1 Set up Linux Lab Environment

2.3 Transferring files between local computer and Linux server

2.3.1 Use command line tools

2.3.1.1 Copy files using `rsync`

2.3.1.2 Copying Files with `scp`

2.3.2 Download files

Chapter 3

File system and commands of Linux

3.1 Path

3.1.1 Relative and absolute path

3.1.2 Comand ls

3.1.3

3.2

3.3 Manipulations of files and directories

3.3.1 Command cp

3.3.1.1 Move one file

3.3.1.2 Move multiple files into a directory

3.3.1.3 Move a directory

3.3.1.4 Rename a file or a directory

3.3.1.5 How to create a

3.4 Viewing text files in Linux

3.5 Understand standard input and stardard output

3.5.1 STDIN

3.5.2 STDOUT

3.5.3 STDERR

3.6 Auto-completion

3.7 Advanced topic

Chapter 4

To find files larger than 100MB:

4.1 Check you job status

4.1.1 List files bigger than filesize specified

Chapter 5

Text editor in Linux

In Linux, we sometimes need to create or edit a text file like writing a new R program. So we need to use text editor.

As a newbie, someone would prefer a basic, GUI-based text editor with menus and traditional CUA key bindings. Here we recommend Sublime and Notepad++.

But GUI-based text editor is not always available in Linux. A powerful screen text editor vi (pronounced “vee-eye”) is available on nearly all Linux system. We highly recommend vi as a text editor, because something we’ll have to edit a text file on a system without a friendlier text editor. Once we get familiar with vi, we’ll find that it’s very fast and powerful.

But remember, it’s OK if you think this part is too difficult at the beginning. You can use either Sublime or Notepad++. If you are connecting to a Linux system without Sublime and Notepad++, you can write the file in a local computer and then upload the file onto Linux system.

5.1 Basic vi skills

As vi uses a lot of combination of keystrokes, it may be not easy for newbies to remember all the combinations in one fell swoop. Considering this, we’ll first introduce the basic skills someone needs to know to use vi. We need to first understand how three modes of vi work and then try to remember a few basic vi commands. Then we can use these skills to write Perl or R scripts in the following chapters for Perl and R (Figure 5.1).

5.2 Create new text file with vi

```
mkdir test_vi  ## generate a new folder
cd test_vi    ## go into the new folder
echo "Using `ls` we don't expect files in this folder."
ls
echo "No file displayed!"
```

```
## Using `ls` we don't expect files in this folder.
## No file displayed!
```

Using the code above, we made a new directory named `test_vi`. We didn’t see any file.

If we type `vi test.pl`, an empty file and screen are created into which you may enter text because the file does not exist((Figure 5.2)).

Working modes

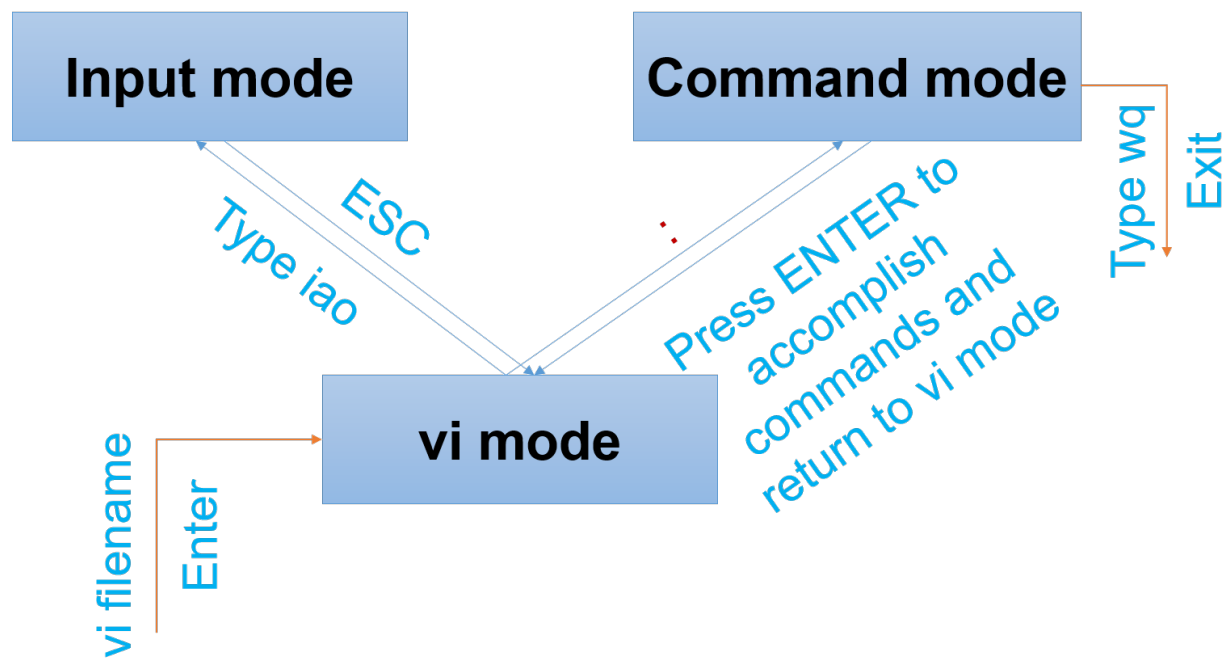


Figure 5.1: Three modes of vi.

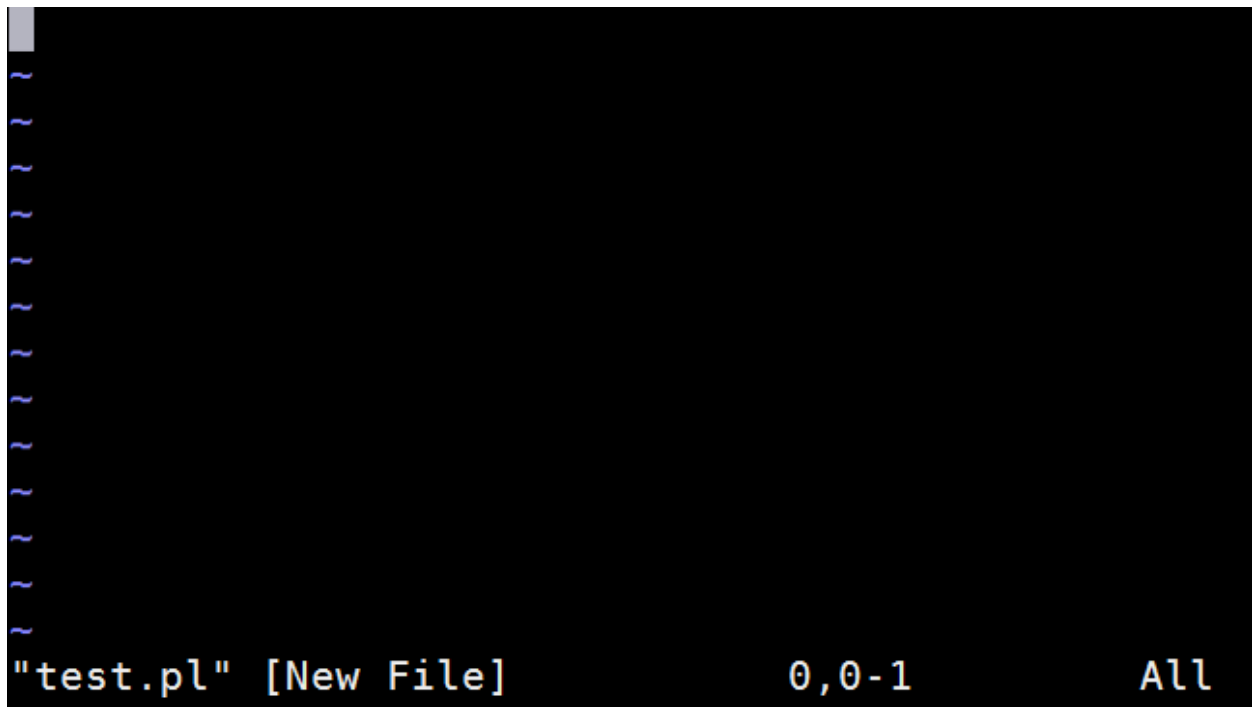


Figure 5.2: A screenshot of the vi test.pl.

```
vi test.pl
```

Now if you are in vi mode. To go to Input mode, you can type i, ‘a’ or ‘o’ (Figure 5.3).

Now you can type the content (codes or other information) (5.4).

Once you are done typing. You need to go to Command mode (Figure 5.1) if you want to save and exit the file. To do this, you need to press ESC button on the keyboard.

Now we just wrote a Perl script. We can run this script.

```
perl test.pl
```

```
## Hello Bioinformatics World!
```

5.3 An example for using editor R

```
qnorm(.975)
```

```
## [1] 1.959964
```

```
xval<-seq(-3.2,3.2, length=1000)
yval<-dnorm(xval)
plot(xval, yval, type="l",axes=T,lwd=3,xlab="",ylab="")
x<-seq(qnorm(.975), 3.2, length = 100)
polygon(c(x,rev(x)), c(dnorm(x), rep(0,length(x))), col="salmon")
```

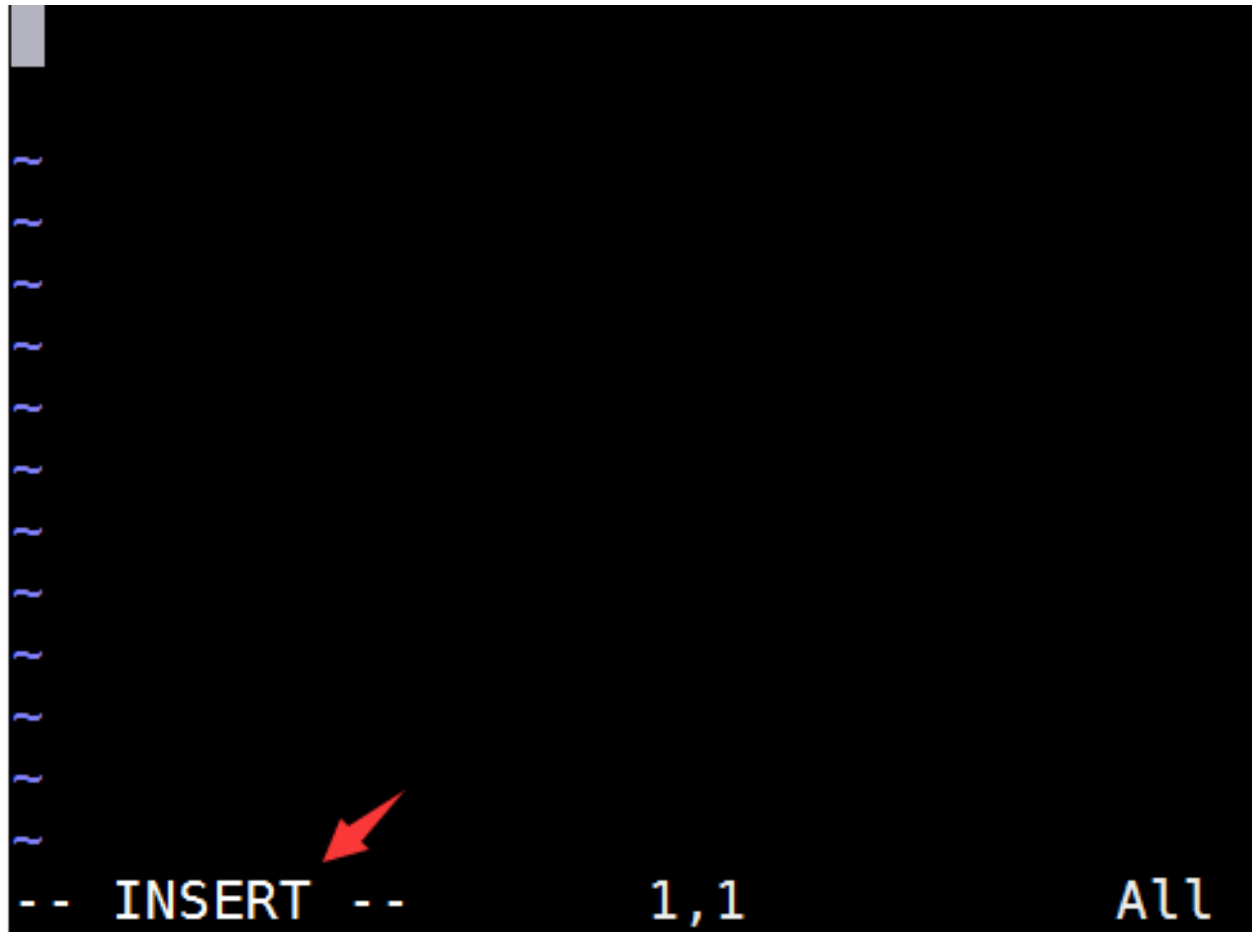
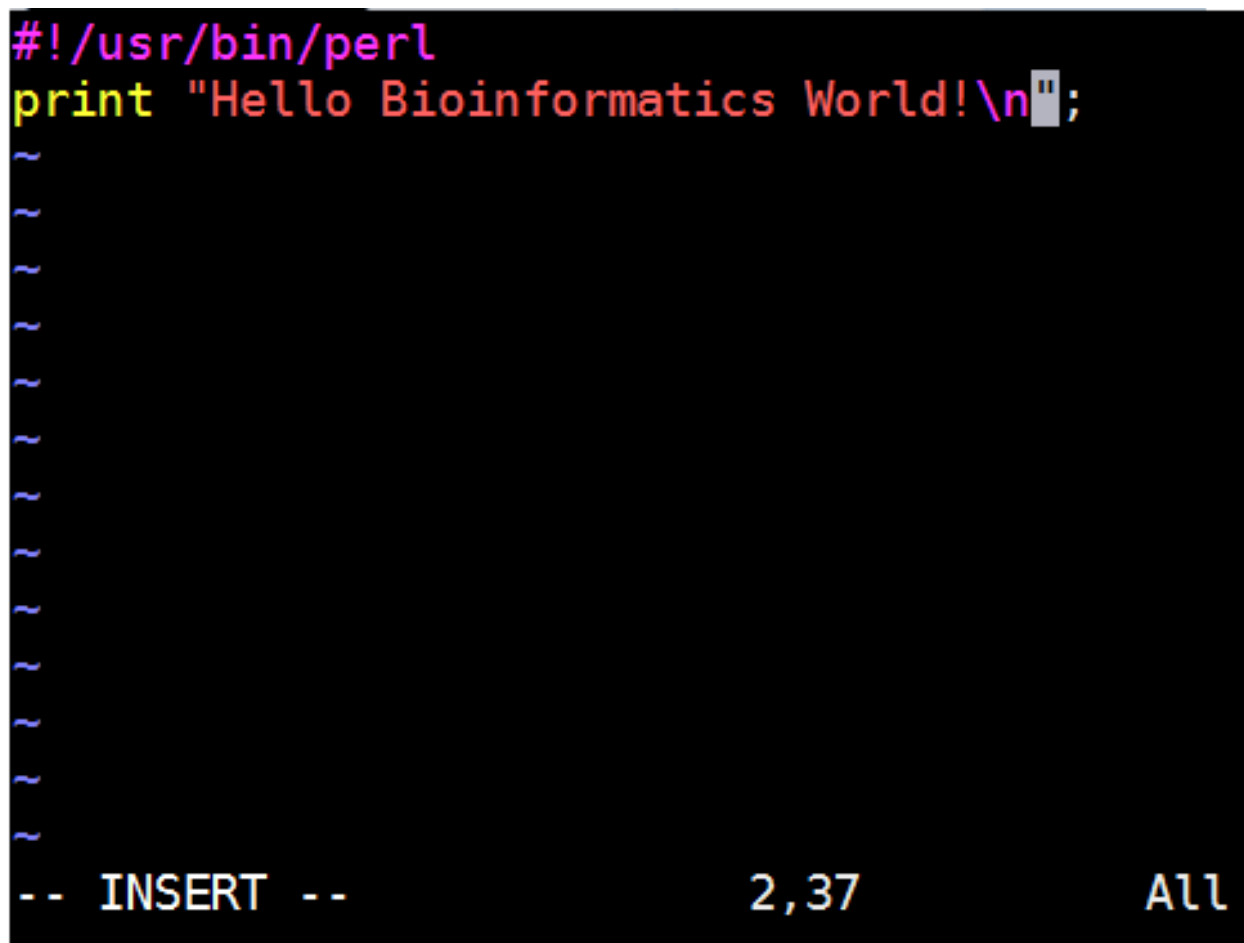


Figure 5.3: A screenshot of the vi `test.pl`.



```
#!/usr/bin/perl
print "Hello Bioinformatics World!\n";
~
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --                2,37                All
```

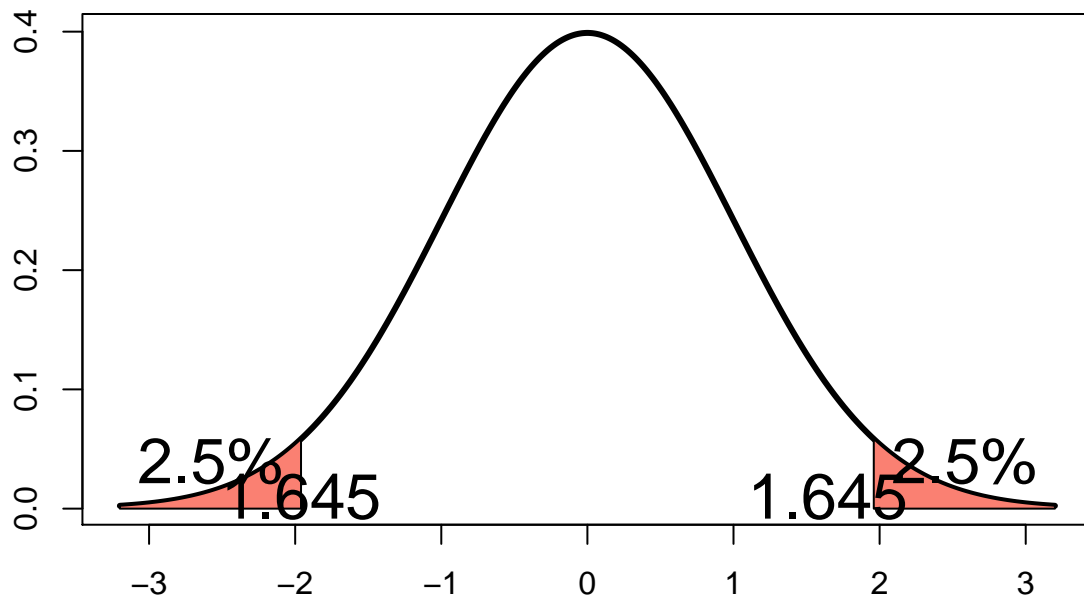
Figure 5.4: A screenshot of the vi `test.pl`.

```

text(mean(x),mean(dnorm(x))+0.02, "2.5%", cex=2)
text(qnorm(.95), 0.01, "1.645",cex=2)

x<-seq(-3.2, qnorm(.025), length =100)
polygon(c(x,rev(x)), c(dnorm(x), rep(0,length(x))), col="salmon")
text(mean(x),mean(dnorm(x))+0.02, "2.5%", cex=2)
text(qnorm(.025), 0.01, "1.645",cex=2)

```



5.3.1 Shell script

<https://wikis.utexas.edu/display/bioiteam/Example+BWA+alignment+script>

5.4 How to exit VIM

<https://stackoverflow.com/questions/11828270/how-to-exit-the-vim-editor>

split Vim window to view multiple files at once?

###

““

““

###

<https://stackoverflow.com/questions/1269603/to-switch-from-vertical-split-to-horizontal-split-fast-in-vim>

<https://vi.stackexchange.com/questions/64/is-it-possible-to-split-vim-window-to-view-multiple-files-at-once>

<https://askubuntu.com/questions/418396/what-is-the-difference-between-vi-and-vim>

<https://unix.stackexchange.com/questions/93144/exit-vim-more-quickly>

<https://stackoverflow.com/questions/53664/how-to-effectively-work-with-multiple-files-in-vim?rq=1>

Chapter 6

Install Bioinformatics software in Linux

6.1 Installation from source code

6.1.1 Install `bwa`

6.1.2 Install `samtools`

6.1.3 Align reads to genome using `bwa` and store the alignment results in SAM/BAM files

6.2 Installing a precompiled binary (executable)

6.2.1 Install `bwa`

Chapter 7

Placeholder

7.0.2 Achiving and compressing files

7.0.3

Chapter 8

First Perl Program

8.1 First Program

8.1.1 How to read extract a set of sequences from the reference genome

