

Useful commands in Linux and other tools for quality control

Ignacio Aguilar

INIA Uruguay

05-2018

Unix Basic Commands

`pwd`

show working directory

`ls`

list files in working directory

`ll`

as before but with more information

`mkdir d`

make a directory `d`

`cd d`

change to directory `d`

Copy and moving commands

To copy file

```
cp /home/user/is .
```

To copy file directory

```
cp -r /home/folder .
```

to move file aa into bb in folder test

```
mv aa ./test/bb
```

To delete

```
rm yy
```

delete the file yy

```
rm -r xx
```

delete the folder xx

Redirections & pipe

Redirection useful to read/write from file !!

aa < bb

program **aa** reads from file **bb**

blupf90 < in

aa > bb

program **aa** write in file **bb**

blupf90 < in > log



Redirections & pipe

“|” similar to redirection but instead to write to a file,
passes content as input to other command

`tee` copy standard input to standard output and save in a file
`echo` copy stream to standard output

Example: program `blupf90` reads name of parameter file and
writes output in terminal and in file `log`

```
echo par.b90 | blupf90 | tee blup.log
```

Other popular commands

head file
tail file
less file
wc -l file
grep text file
cat file1 file2

print first 10 lines list *file* page-by-page
print last 10 lines
list *file* line-by-line or page-by-page
count lines
find lines that contains text
concatenate files

sort
cut
join
paste
expand
uniq

sort file
cuts specific columns
join lines of two files on specific columns
paste lines of two file
replace TAB with spaces
retain unique lines on a sorted file

head / tail

```
$ head pedigree.txt
```

```
1 0 0
```

```
2 0 0
```

```
3 0 0
```

```
4 0 0
```

```
5 0 0
```

```
6 0 0
```

```
7 0 0
```

```
8 0 0
```

```
9 0 0
```

```
10 0 0
```

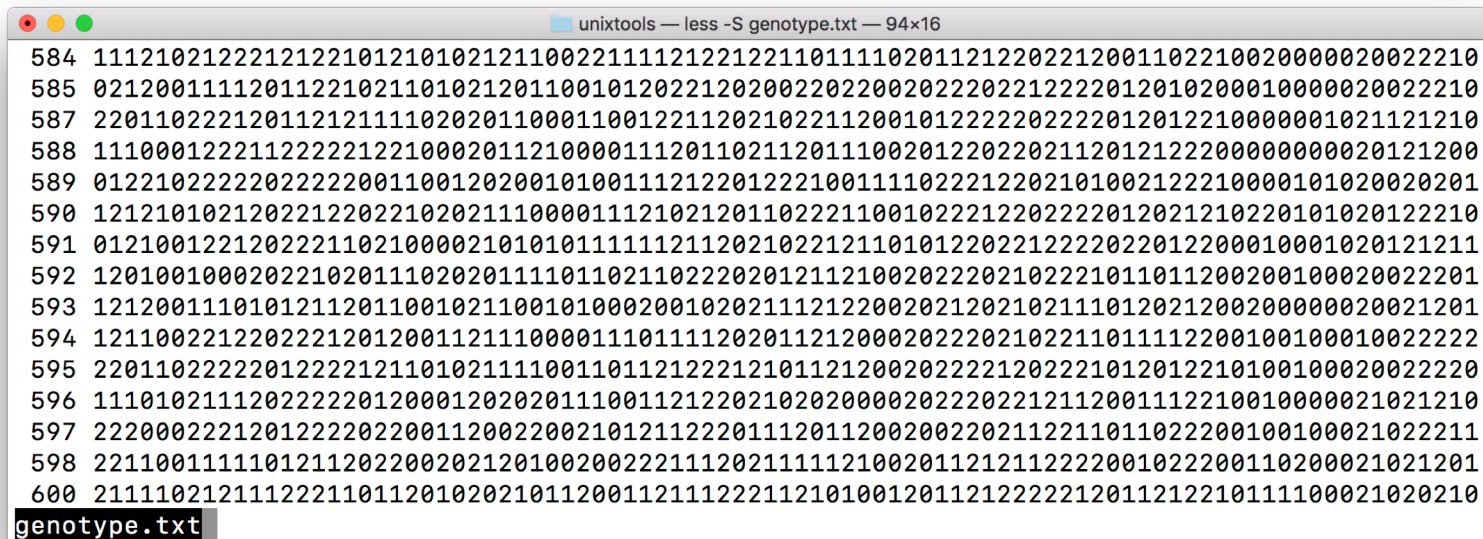
Huge volume of information with genomic

- Example 50kv2 (54609 SNP)
 - For 104 individuals
 - Illumina final report file:
 - 5,679,346 records
 - 302 MB
- Not efficient to read/edit with regular editors (vim, gedit...)

less command

- Allow to view content of file and move forward and backward
- For files with long lines use option `-S`

```
less -S genot.txt
```



```
unixtools — less -S genotype.txt — 94x16
584 11121021222121221012101021211002211112122122110111102011212202212001102210020000020022210
585 02120011112011221021101021201100101202212020022022002022202212222012010200010000020022210
587 22011022212011212111102020110001100122112021022112001012222202222012012210000001021121210
588 11100012221122221221000201121000011201102112011100201220220211201212220000000020121200
589 01221022222022222001100120200101001112122012221001111022212202101002122210000101020020201
590 12121010212022122022102021110000111210212011022211001022212202222012021210220101020122210
591 0121001221202221102100002101010111112112021022121101012202212222022012200010001020121211
592 12010010002022102011102020111101102110222020121121002022202102221011011200200100020022201
593 1212001110101211201100102110010100020010202112122002021202102111012021200200000020021201
594 1211002212202221201200112111000011101112020112120002022202102211011112200100100010022222
595 22011022222012222121101021111001101121222121011212002022221202221012012210100100020022220
596 11101021112022222012000120202011100112122021020200002022202212112001112210010000021021210
597 22200022212012222022001120022002101211222011120112002002202112211011022200100100021022211
598 2211001111101211202200202120100200222112021111121002011212112222001022200110200021021201
600 21111021211122211011201020210112001121112221121010012011212222212011212210111100021020210
genotype.txt
```

Count inside files

- Command `wc` count number of words/lines/characters/bytes
- Number of lines of a file(s)

```
$ wc -l genotype.txt pedigree.txt
3534 genotype.txt
6473 pedigree.txt
```

Concatenate files

Put content of file1 and file2 in outputfile

```
cat file1 file2 > output_file
```

Add content of file3 to output_file using >> redirection

Append content at the end of the file

```
Cat file3 >> output_file
```

paste / expand

paste Merge files line by line with TAB delimiter

expand replace TAB with spaces

```
head sol_a sol_b
```

```
==> sol_a <==
```

trait/effect	level	solution
1 1	1	10.23612694
1 1	2	12.09158350
1 1	3	13.08139319
1 2	1	-0.00804515
1 2	2	0.00804485
1 2	3	-0.01608983
1 2	4	0.01609003
1 2	5	0.03218022
1 2	6	-0.03218023

```
==> sol_b <==
```

trait/effect	level	solution
1 1	1	14.09761235
1 1	2	15.95388962
1 1	3	16.94297738
1 2	1	-0.00806027
1 2	2	0.00805982
1 2	3	-0.01608645
1 2	4	0.01608642
1 2	5	0.03217834
1 2	6	-0.03217873

```
paste sol_a sol_b | expand | head
```

trait/effect	level	solution	trait/effect	level	solution
1 1	1	10.23612694	1 1	1	14.09761235
1 1	2	12.09158350	1 1	2	15.95388962
1 1	3	13.08139319	1 1	3	16.94297738
1 2	1	-0.00804515	1 2	1	-0.00806027
1 2	2	0.00804485	1 2	2	0.00805982
1 2	3	-0.01608983	1 2	3	-0.01608645
1 2	4	0.01609003	1 2	4	0.01608642
1 2	5	0.03218022	1 2	5	0.03217834
1 2	6	-0.03218023	1 2	6	-0.03217873

sort / join

- Sort file, usually we want to specify on a certain column

```
sort -k 2,2 file1 > a
```

```
sort -k 1,1 file2 > b
```

Merge both file by column 2 and column 1

```
join -1 2 -2 1 a b > new_file
```

grep

grep is useful to find pattern with a file and list all lines that match the pattern

```
grep sire1 pedigree.txt
```

Option -v show all lines does not match pattern

Pattern with spaces use -e

```
grep -e "pattern with spaces" file1
```

sed

Sed is a stream editor.

Reads input file and apply command that match pattern

Sustitution of a pattern

```
sed 's/pattern 1/new pattern/g' file > newfile
```

Delete lines that contains “pattern to match”

```
sed '/pattern to match/d' file
```

awk

- Interpreted program language, that process data stream of file line by line
- Very useful and fast command to work with text files
- Can be used as a database query program
 - Select specific columns or create new ones
 - Select specific rows matching some criteria
- Use of if/else and for structure
- Easy implementation of hash-table arrays

awk

Selection of columns: $\$1$, $\$2$, ..., $\$n$. are first, second and the last column

Also $\$0$ select the full line

Extract equations solutions for a particular effect (2) and print EBV and reliabilities (r^2)

```
awk '{ if ($2==2) print $3,$4,1- $\$5*\$5/20$ }' solutions
```

Implicit variables

- NF - number of fields

- NR - record number

- FS - input field separator

- OFS - output field separator

Process CSV files

```
awk 'BEGIN {FS=","} {print $1,$2,$3}' pedigree.txt
```

awk hash tables

Arrays can be index by alphanumeric variables in a efficient way

Awk version to count progeny by sire

```
awk '{ sire[$2]+=1} END { for (s in sire)
    {print "Sire " s, sire[s]}}' pedigree.txt
```

uniq

- Command `uniq` list all unique lines of a file
- Option `-c` count number of times occurs in a file

Example count progeny by sire in a pedigree file

```
awk '$2>0{ print $2}' ped | sort | uniq -c
```

qcf90

- Quality control program from the BLUPF90 family programs
- Check pedigree and genotype files either in raw formats (ie alphanumeric fields) or with renumf90 output file
- Run in command line with different arguments
- Generate new "clean" files
- `qcf90` with no arguments prints current options

qcf90

```
unixtools — -bash — 85x14
This is qcf90, Version 0.9.5.
Yutaka Masuda, Ignacio Aguilar, and Ignacy Misztal
University of Georgia

usage: qcf90 [options]

Options:
Source file specification
  --snpfile mfile      marker file
  --mapfile cfile      read map file
  --maffile ffile      read MAF file
  --pedfile pfile      pedigree file
  --xrefid xfile       XrefID file; assumed renumbered pedigree
  --statfile qfile     read QC status from a file instead of computing
```

qcf90 Quality Control options

Marker quality-control (QC) options

```
--qc {items}          * Quality control; see --long-help for details
--crm n               call rate for markers; default=0.90
--cra n               call rate for animals; default=0.90
--maf n               minimum allele frequency; default=0.05
--hwe n               statistic for Hardy-Weinberg equilibrium
--exclude-marker-list file marker positions to be excluded in QC
--exclude-animal-list file animal ID (same as pedigree if supplied) to be excluded in QC
--remove-markers      * remove unqualified markers in subsequent QC steps
--remove-animals      * remove unqualified animals in subsequent QC steps
--check-parentage     check Mendelian inconsistency; equivalent to --qc par
--check-format        * check file format precisely
```

Quality control options: --qc {crm,maf,mono,hwe,cra,parm,para}

```
crm      call rate for markers
maf      minor allele frequency
mono     monomorphic markers
hwe      Hardy-Weinberg equilibrium
cra      call rate for animals
par      parentage (Mendelian inconsistency) for markers and animals
```

qcf90 output files

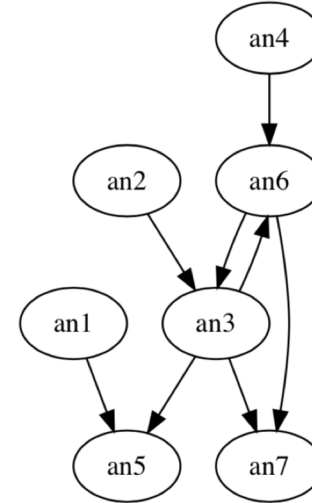
File output options

<code>--save-log [file]</code>	* save log file; default=qcf90.log
<code>--save-status [file]</code>	* save QC status; default=qcf90.status
<code>--save-dot [file]</code>	* save dot file to visualize pedigree loops; default=pfile+".dot"
<code>--save-clean [suffix]</code>	save qualified files with the suffix; default="_clean"
<code>--cleanup-marker-flags n..</code>	flags for markers being removed from clean files; default=all
<code>--cleanup-animal-flags n..</code>	flags for animals being removed from clean files; default=all

qcf90 pedigree check

- Basic check on sire and dam ids
- Detection of 'loops' in pedigree
 - i.e. animal is an ancestor of itself
 - an3 and an6

an1	0	0
an2	0	0
an3	an6	an2
an4	an1	an2
an5	an1	an3
an6	an4	an3
an7	an6	an3



seekparentf90

- Program to check and assign parents using genomic information
- Detect parent-offspring incompatibilities based on counts of conflicts (opposite homozygous)
 - Hayes 2010 JAS
 - Wiggans et al 2010 JDS

seekparentf90

Usage

```
seekparentf90 --pedfile <pedigree_file_name> --snpsfile <snps_file> [ ... ]
```

Optional arguments

`--yob`

Indicate that year of birth should be read in the 4th column.
If yob information is present, it will be used to validate a putative parent.

`--seeksire <sire_file>`

Indicate a list of sires that will be used to search for a parent

`--seekdam <sire_file>`

Indicate a list of dams that will be used to search for a parent

`--seeksire_in_ped`

Create a list of genotyped sires from the pedigree and use it as a list to search for a parent.

`--seekdam_in_ped`

Create a list of genotyped dams from the pedigree and use it as a list to search for a parent

`--seektype n`

Set the which animals will be used to search for a parent

Codes:

- 1: search only non-match parent (default)
- 2: search all genotyped individuals

Check parentage using different chips

Chip and SNP information

Chips with different number of SNP can be used in the analyses.

In such case the genotype file must have the second column indicating the chip number and a map file must be provided to map SNP to chips.

Each sample in the genotype file should contain only the SNP present for that chip (see example below)

```
--chips <file>
```

Genotype file

```
1353 1 2110101100201201101101011011111121111121
8014 1 2111010151110112022111011151111210111221
516 2 2110510120
181 3 11101111122011205502
```

Map file

SNP_ID	Chr	pos	chip1	chip2	chip3
SNP_1	1	135098	1	1	1
SNP_2	1	267940	2	0	2
SNP_3	1	305793	3	2	3
SNP_4	1	353745	4	0	0
SNP_5	1	393248	5	0	4
SNP_6	1	434180	6	0	5
SNP_7	1	471078	7	0	0
SNP_8	1	516404	8	0	6

illumina2pregs

A program to convert genomic information from Illumina files for the BLUPF90 family of programs

- Tool for converting FinalReport and SNP_Map.txt files from Genome Studio into files suitable for reading with blupf90
- Reads only AB codes and generates integer variable 0,1,2 or 5 for missing

Illumina Raw files

mgaDBSNP — less — 105×30

```
[Header]
GSGT Version 1.9.4
Processing Date 8/29/2011 10:25 AM
Content BovineSNP50_v2_C.bpm
Num SNPs 54609
Total SNPs 54609
Num Samples 104
Total Samples 104
[Data]
```

SNP Name	Sample ID	Allele1 - Forward			Allele2 - Forward			Allele1 - Top		
Top	Allele2 - AB	Allele1 - AB	GC	Score	X	Y				
ARS-BFGL-BAC-10172	8448	G	G	G	G	B	B	0.9506	0.026	1.013
ARS-BFGL-BAC-1020	8448	A	G	A	G	B	A	0.9673	0.318	0.374
ARS-BFGL-BAC-10245	8448	C	C	G	G	B	B	0.7579	0.091	1.363
ARS-BFGL-BAC-10345	8448	A	C	A	C	B	A	0.9276	0.727	0.724
ARS-BFGL-BAC-10365	8448	G	G	C	C	B	B	0.5335	0.000	0.958
ARS-BFGL-BAC-10375	8448	G	G	G	G	B	B	0.9567	0.010	0.765
ARS-BFGL-BAC-10591	8448	A	A	A	A	A	A	0.9003	0.479	0.029

FinalReport

SNP_Map.txt

mgaDBSNP — less — 112×19

Index	Name	Chromosome	Position	GenTrain	Score	SNP	ILMN	Strand
1	ARS-BFGL-BAC-10172	14	6371334	0.9176	[A/G]	TOP	TOP	2
2	ARS-BFGL-BAC-1020	14	7928189	0.9413	[T/C]	BOT	TOP	2
3	ARS-BFGL-BAC-10245	14	31819743	0.7646	[T/C]	BOT	BOT	
4	ARS-BFGL-BAC-10345	14	6133529	0.8906	[A/C]	TOP	TOP	2
5	ARS-BFGL-BAC-10365	14	27005721	0.9206	[A/C]	TOP	BOT	
6	ARS-BFGL-BAC-10375	14	6616434	0.9258	[A/G]	TOP	TOP	2
7	ARS-BFGL-BAC-10591	14	17544926	0.8639	[A/G]	TOP	TOP	
8	ARS-BFGL-BAC-10867	14	34639444	0.9085	[G/C]	BOT	BOT	
9	ARS-BFGL-BAC-10919	14	31267746	0.8255	[A/G]	TOP	TOP	
10	ARS-BFGL-BAC-10951	10	17911906	0.9056	[T/G]	BOT	BOT	
11	ARS-BFGL-BAC-10952	10	18882288	0.9184	[A/G]	TOP	TOP	
12	ARS-BFGL-BAC-10960	10	20609250	0.9205	[A/G]	TOP	TOP	
13	ARS-BFGL-BAC-10972	10	20792754	0.8432	[G/C]	BOT	BOT	
14	ARS-BFGL-BAC-10975	10	21225382	0.7991	[A/G]	TOP	TOP	
15	ARS-BFGL-BAC-10986	10	26527257	0.8941	[A/C]	TOP	BOT	
16	ARS-BFGL-BAC-10993	10	78512500	0.8649	[A/G]	TOP	BOT	
17	ARS-BFGL-BAC-11000	10	79252023	0.9433	[T/G]	BOT	BOT	

Usage

Illumina2preGS program

```
illumina2pregs --snpfile <yourSnpFinalReportFilename>
```

Optional Arguments

```
--mapfile
```

File name for the map file. Default "SNP_Map.txt"

```
--genome
```

Indicate type of genome (1 - bovine, 2 - ovine)

```
--nosortmap
```

Genotype file will be in the same order as given

```
--codeab <n>
```

Indicate column number for the first allele with code AB (default 7)

```
--gc <n thr>
```

Set column number (n) of GC and the cutoff (thr) for exclude call (default from GenomeStudio 0.15)

```
--alphasize
```

Set the maximum length to store alphanumeric IDs (default 10)

```
--snpfile_out
```

Set a file name for output SNP file (default snps2pregs)

Illumina2preGS program

Output files

snps2pregs

```
1 ID (alphanumeric identification of lenght "alphasize", default 10)
2 one space
3 genotypes: 0, 1, 2, for AA, AB and BB respectively and 5 for missing.
```

snp_map

```
1 index
2 chromosome
3 position
4 name
5 index in original SNP_Map.txt
```

Illumina2preGS program

'snps2pregs' marker file

```
80      21101011002012011011010110111111211111210100
8014    21110101511101120221110111511112101112210100
516     21100101202252021120210121102111202212111101
181     21110111112201120550200020101022212211111100
```

'snp_map' file

	Chromosome #	Position	SNP Name	Original position in raw files
SNP index →	1	120183	BovineHD0100000035	10372
	2	135098	Hapmap43437-BTA-101873	75864
	3	158820	BovineHD0100000048	10373
	4	183040	BovineHD0100000057	10374
	5	208728	BovineHD0100000064	10375
	6	218271	BovineHD0100000067	10376
	7	267940	ARS-BFGL-NGS-16466	71981
	8	278952	BovineHD0100000079	10377
	9	288500	BovineHD0100000082	10378

SNP Databases

 OPEN ACCESS  PEER-REVIEWED

RESEARCH ARTICLE

TheSNPpit—A High Performance Database System for Managing Large Scale SNP Data

Eildert Groeneveld , Helmut Lichtenberg

Published: October 25, 2016 • <https://doi.org/10.1371/journal.pone.0164043>

SNP Database

- Example DB @ INIA Las Brujas
 - Actually SNP (bovine, ovine, trees) from 32 different chips with different sizes
 - Storing different samples by individual
 - Efficient pack storage
 - 50kV2: 13472 54609 bytes in 13472 bytes ~ 25%
 - Support for original or user-created query from different chips
- Loading and extract information by Python programs
- Extraction
 - SNP in common by chips
 - Imputation type of files
 - *main* chip and SNP in common from other chips

SNP Databases

 OPEN ACCESS  PEER-REVIEWED

RESEARCH ARTICLE

TheSNPpit—A High Performance Database System for Managing Large Scale SNP Data

Eildert Groeneveld , Helmut Lichtenberg

Published: October 25, 2016 • <https://doi.org/10.1371/journal.pone.0164043>

Useful commands for Linux

- Several tutorials on the WEB !!
- [unixcombined.pdf](#) from Misztal web page
 - <http://nce.ads.uga.edu/~ignacy/ads8200/unixcombined.pdf>
- genomeek blog (F. Guillaume)
 - <http://genomeek.wordpress.com>