awk
short intro

Robert Kofler

awk

- scripting language for reformatting text files or extracting information from text files
- good performance with large text files

⇒thus “awk” is like PCR for NGS data analysis

Why awk:

very powerful (you can answer a lot of NGS related questions with a simple awk command)

easy to learn

good performance with large text files

may be used with other Unix-command (pipe)

Our input file

```
robertkofler@i122mc121 %cat sample.txt
# Following a description of the file format
# chromosome position snps coverage
2L      50      3       17
2L      100     4       35
2L      150     13      100
2L      200     24      87
2L      250     12      60
2R      50      6       11
2R      100     11      91
2R      150     7       90
2R      200     3       67
2R      250     1       18
```

basics

- Name: Alfred Aho, Peter Weinberger, Brian Kernighan
- How to use awk

```
awk ‘command’ input_file.txt
or
cat input_file.txt | awk ‘command’
```

Let’s start:

```
awk ‘{print $1}’ sample.txt

cat sample.txt| awk ‘{print $1}’
```

Structure of awk command

full structure of a awk command:

```
awk 'BEGIN{command}condition{command}END{command}'
# BEGIN{command} => execute this command at the
  beginning
# condition{command} => MOST IMPORTANT PART, will be
  executed for every line in the file!
# END{command} => execute this command at the end
```

example:

```
awk 'BEGIN{print "File start"}$1=="2L"{print
  $0}END{print "End of file"}' sample.txt
```

```
File start
2L      50      3       17
2L      100     4       35
2L      150     13      100
2L      200     24      87
2L      250     12      60
End of file
```

Most things are optional

shortest example possible

```
awk ‘’ sample.txt => not doing anything
awk ‘$1’ sample.txt => prints line if $1 is not empty
# above, we are only using ‘condition{command}’
# we are not using ‘BEGIN{command}’ and ‘END{command}’
```

default for ‘condition{command}’ is the following ‘condition’ => ‘condition{print $0}’

```
awk ‘$1’ sample.txt
# is per default using the following command:
awk ‘$1{print $0}’
# {print $0} is the default for the loop!!
# {print $0} just means print the line
```

Now some more meaningful conditions

```
awk ‘$1==“2L”’ sample.txt => only print lines of
chromosome 2L

Exercise 1: what is the default of this command
Exercise 2: print only lines of chromosome 2R


Now let’s only print lines of chromosome 2L having a
position smaller than 150 (AND condition)
awk '$1=="2L" && $2<150' sample.txt


Exercise 3: print only lines of chromosome 2R having
a coverage higher than 50
```

More conditions

```
# print lines from chromosome 2R or 2L (OR condition)
awk ‘$1==“2L” || $1==“2R’

# only print SNPs of chromosome 2L
awk ‘$1==“2L”{print $3}’
# NOTE: above, we are not using the default!

Exercise 4: print the SNPs of chromosome 2L and
chromosome 2R
```

Introducing variables

```
# count all SNPs on chromosome 2L
awk '$1==“2L”{l+=$3; print l}' sample.txt

# ‘l’ is a variable; in every line we are adding the
number of SNPs; default starting value of ‘l=0’
# we are executing two commands at once they are
separated by ‘;’

More elegant solution:
awk '$1==“2L”{l+=$3}END{print l}' sample.txt

# Exercise 5: Count the number of windows of
chromosome 2R
# Exercise 6: Calculate the average coverage of
chromosome 2R
```

Regex conditions

```
# First a regular expression condition
# we want to skip the header
awk '$1 !~ /^#/' sample.txt
# column 1 does not start with a ‘#’

# lets print lines where column 1 just contains a ‘L’
awk ‘$1 ~ /L/’

Exercise 7: print lines where $1 does not start with
a ‘#’ and column1 contains a ‘R’
```

Print several columns

```
#In the following example we are not providing a
condition. Per default the command will be executed
for every line
# try the following two commands
awk '{print $1 $2}' sample.txt
awk '{print $1,$2}' sample.txt

# space is concatenating and
# ‘,’ is per default replaced with OFS (output field
separator); default OFS = space (not tab!!)

# change the OFS
awk 'BEGIN{OFS="\t"}{print $1,$2}' sample.txt

Exercise 8: modify the above command: lines starting
with ‘#’ need to be ignored
```

FINAL EXAM

Example 9: How many SNPs can be found on chromosome 2R between 0 and 150

Is there a potential bias...

Example 10: What is the average coverage for windows having more than 10 SNPs

Example 11: and what is the average coverage for windows having less than 10 SNPs

Solutions

```
E1: awk ‘$1==“2L”{print $0}’ sample.txt
E2: awk ‘$1==“2R”’ sample.txt
E3: awk '$1=="2R" && $4>50' sample.txt
E4: awk '$1=="2R" || $1=="2L"{print $3}' sample.txt
E5: awk '$1=="2R"{l+=1}END{print l}' sample.txt
E6: awk '$1=="2R"{count+=1; cov+=$4}END{print cov/
count}' sample.txt
E7: awk '$1 ~ /R/ && $1 !~ /^#/' sample.txt
E8: awk 'BEGIN{OFS="\t"}$1!~/^#/{print $1,$2}'
sample.txt
E9: cat sample.txt | awk '$1=="2R" && $2>=0 &&
$2<=150{l+=$3}END{print l}’
E10:cat sample.txt | awk '$1 !~/^#/ && $3>10{count
+=1; cov+=$4}END{print cov/count}’
E11: cat sample.txt | awk '$1 !~/^#/ && $3<10{count
+=1; cov+=$4}END{print cov/count}’
Possible Bias: YES
```