Tutorial on (un)compressing files

Timothée Flutre

Tutorial on (un)compressing files

Timothée Flutre

February 10, 2015

Contents

1 Introduction

When working with data, everyone usually starts by playing with text files, but these files can soon become very large! For instance, it is now frequent for a DNA sequencing project to generate Tb of data... To save disk space, normal text files (i.e. not binary files) can often be compressed to 10% of their original size.

If one uses the gzip compression algorithm (or bzip2), under many circumstances one can interact with the file as if it were not compressed, always (un)compressing on the fly. These days processors are fast enough that it hardly takes more time to read and uncompress than it simply takes to read. The only time you don't want to do this, is if you need random access to the file.

Below are tips for interacting with gzipped files in different languages; don't hesitate to contact me if you want to improve this document. The document itself is versioned with git and hosted on GitHub.

2 Bash

`https://www.gnu.org/software/gzip/`

Make a test file and compress it with gzip:

```
echo -e "a\t1\nb\t2" | gzip > example.txt.gz
```

Look at the content of the file:

```
zcat example.txt.gz | less
```

Count the number of lines:

```
zcat example.txt.gz | wc -l
```

Extract the second column:

```
zcat example.txt.gz | cut -f 2
```

Have also a look at this.

3 R

http://stat.ethz.ch/R-manual/R-devel/library/base/html/connections.html

Make a test data set:

```
df <- data.frame(x=rbinom(n=10, size=2, prob=0.3),
                 y=rnorm(n=10, mean=0, sd=1))
```

Write the data frame as a gzipped file:

```
write.table(df, file=gzfile("example.txt.gz"))
```

Read the gzipped file as a data.frame:

```
df <- read.table(file=gzfile("example.txt.gz"), header=TRUE)
```

4 Python

https://docs.python.org/2/library/gzip.html

Make a test file and compress it with gzip:

```
import sys, os, gzip
f = gzip.open("example.txt.gz", "w")
f.writelines("a\t1\nb\t2\n")
f.close()
```

Read the content of the file:

```
import sys, os, gzip
f = gzip.open("example.txt.gz")
line = f.readline()
while line:
    print line
    line = f.readline()
f.close()
```

5 Perl

`http://perldoc.perl.org/IO/Compress/Gzip.html`

`http://perldoc.perl.org/IO/Uncompress/Gunzip.html`

Make a test file and compress it with gzip:

```
use IO::Compress::Gzip qw(gzip $GzipError);
my $f = new IO::Compress::Gzip "example.txt.gz"
    or die "IO::Compress::Gzip failed: $GzipError\n";
$f->print("a\t1\nb\t2\n");
$f->close() ;
```

Read the content of the file:

```
use IO::Uncompress::Gunzip qw(gunzip $GunzipError);
my $f = new IO::Uncompress::Gunzip "example.txt.gz"
    or die "IO::Uncompress::Gunzip failed: $GunzipError\n";
while( my $line = $f->getline() ) {
    print $line;
}
$f->close() ;
```

6 C/C++

`http://zlib.net/`

Include the header of the `zlib` library:

```
#include "zlib.h"
```

and then use `gzopen`, `gzclose`, `gzgetc` and `gzputs`.

A self-contained example is available here.