GRADA Example

Lucas F. Voges

2021

GRADA

This R-Package will analyze Sequencing data and find all sequences and theire position inside. It will utilize a "agrep" and "wc" unix command (future releases may contain functions without unix dependency). In general one could use this tool to find any "String" in a given text like file.

> The option PE (paired end) is only for analyzing two read files at once. It is not neccessary to use this. You could analyze R1 and R2 seperatly, or combined in one file. Though for adpter detection, this is commonly a good practice.

Analyze first!

First the data is to be analyzed with GRADA. in the following the analyze function is explained. Here an example file is used with 22 reads.

> Input: at the moment fastq.gz is not possible to use. You have to extract the files.
>
> seq adapter files is shown below
>
> in the reads, the index and NNN sequence is part of the @SEQ_ID and will therefore be 22 time present beacause of this!
>
> The Sequence NNN will be found once! in NNNNNN. So you cannot count the same adapter multiple times in one read! Though it will be shown in the plot!

---

seq_adapter.txt file:

```
>My_adapter              # Name has to start with ">"
ACTTCTGGACT
>AGA Adapter
AGAAGAAGAAGAAGAAGA
> NNN?
NNN
```

```r
library(GRADA)
library(parallel)

grada_analyze(
  PE = FALSE,                 # analyze paired data (or 2 read files)
  read1 = "example test.fastq",    # read file (fastq, txt ... anything agrep works with.)
  read2 = NULL,               # read2 file if PE
  seq = "seq_adapter.txt",    # adapter files
  M_min = 0,                  # minimal mismatches (0 is required for plotting.)
  M_max = 2,                  # maximal mismatches
  output = "temp/example/",   # the folder for the created data (will be large)
  numCores = detectCores()/2  # parallel computing (will be slow at "1")
)

grada_analyze_positions(
  PE = FALSE,
  readlength = 150,           # this option is needed, it is the maximal read length.
  input = "temp/example/",    # grada_table.txt should be here!
  numCores = detectCores()/2
)
```

> readlength = 150: the nice thing is, that this will be used for plotting. So you can use a different number here! In this example the "TGCTGC" tail is after the 150 nt. So you can put readlength = 200 to see them in the plot! *will be removed or set automatically in the future*
>
> will give warning: "number of columns of result is not a multiple of vector length (arg 2)" if set lower then max read length.

What is needed afterwards?

GRADA will produce some tempoary files, which you can use for your own analysis. But if not needed anymore, you may want to delete it (takes some space!). Please note, that if analyzed again, all files **will be overwritten**.

> The file "grada_table.txt" is needed for the table creation and can be loaded in other programs. it is just the text based output of the analysis.
>
> The file "adapter_positions.Rdata" is needet for plotting the data.
>
> Every other file can be deleted! But you are maybe interested in looking at the reads found by GRADA. Also this data will be used by grada_analyze_positions() and maybe some in development functions.

Results table

Grada can show you the results in a table Format.

> There are different table functions. they will do exactly the same, but have different dependencies. The datatable (DT) is more flexible and interactive!
>
> But you can use the “grada_table.txt” file to make your own tables!

```
library(GRADA)
library(rmarkdown)
library(DT)

grada_table_simple(input = "temp/example/")    # needs (kable, knitr)
```

Table 1: Sequence content in the read files. (M: mismatch)

Adapter	Sequence	Length	R1M2	R1M1	R1M0
My_adapter	ACTTCTGGACT	11	12	9	7
AGA Adapter	AGAAGAAGAAGAAGAAGA	18	20	17	12
NNN?	NNN	3	48	23	23
Index	AGAT	4	44	44	31
TGC-Tail	TGCTGCTGC	9	8	1	1

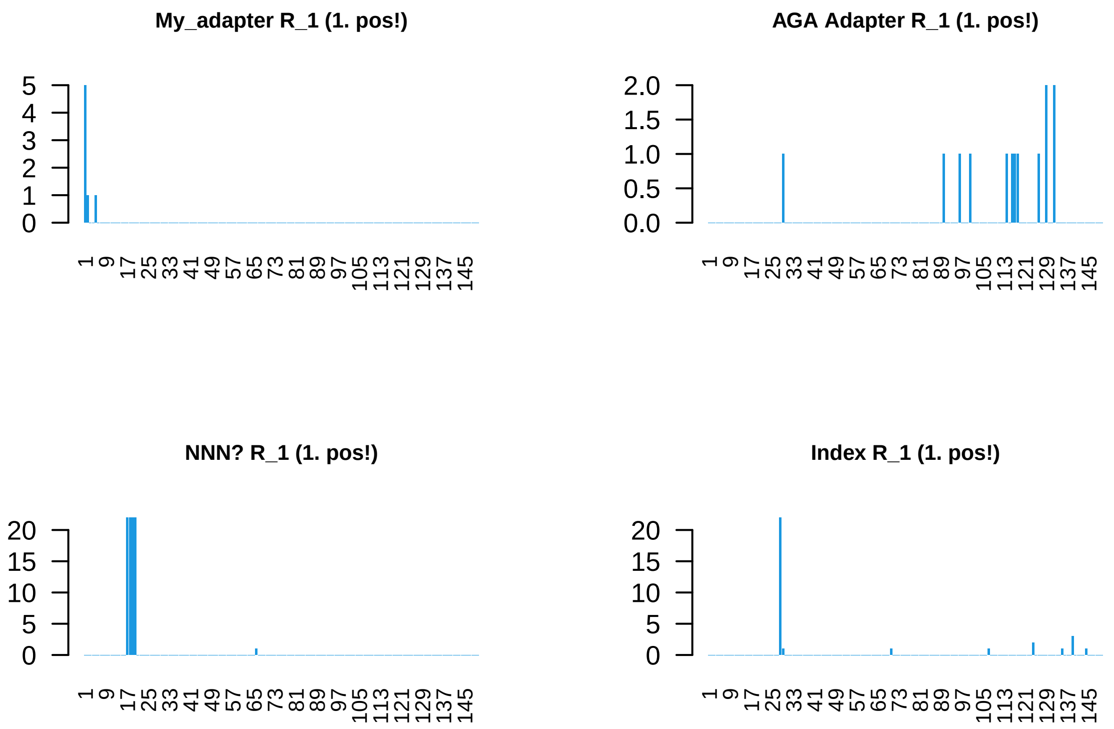
```
#grada_table_md(input = "temp/example/")        # needs (rmarkdown, html)

#grada_table_DT(input = "temp/example/")        # needs (DT)
```

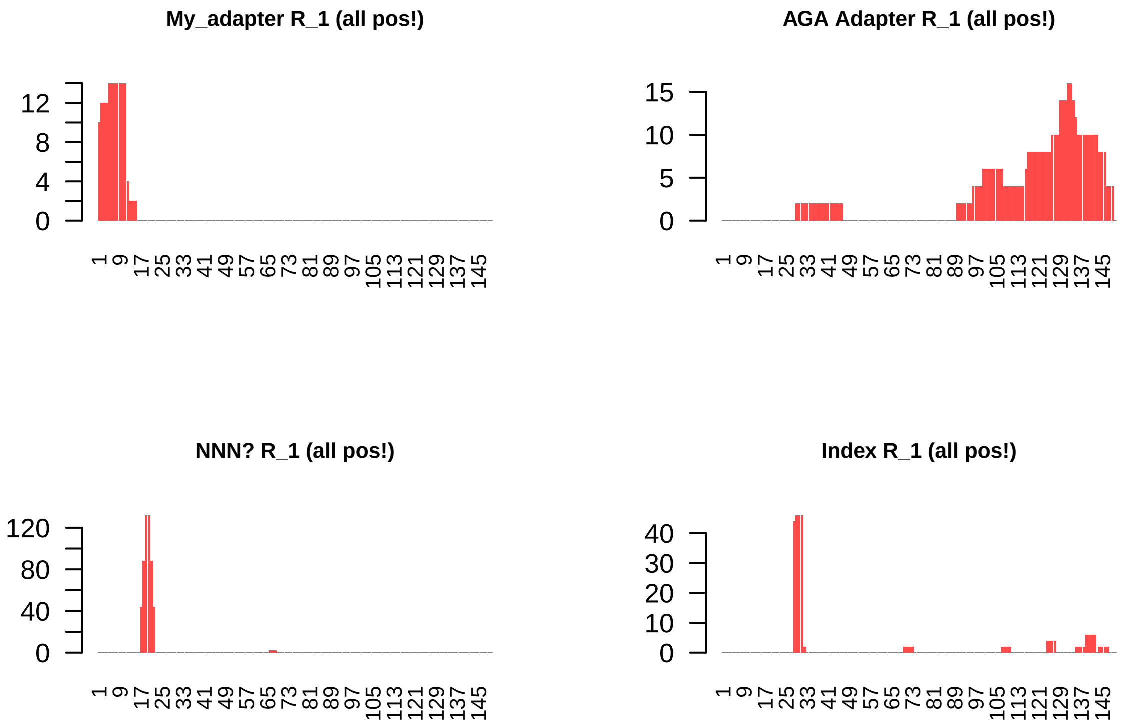
Result plot

```r
library(GRADA)

grada_plot_bar(PE = FALSE, skip = TRUE, input = "temp/example/")
```



```r
grada_plot_bar_full(PE = FALSE, skip = TRUE, input = "temp/example/")
```


My_adapter R_1 (all pos!)
AGA Adapter R_1 (all pos!)
NNN? R_1 (all pos!)
Index R_1 (all pos!)