```r
#source("https://bioconductor.org/biocLite.R")
#biocLite("GenomicRanges")
#biocLite("rtracklayer")
#biocLite("GenomicAlignments")
library(GenomicRanges)

## Loading required package:  stats4
## Loading required package:  BiocGenerics
## Loading required package:  parallel
##
## Attaching package:  'BiocGenerics'
## The following objects are masked from 'package:parallel':
##
##     clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
##     clusterExport, clusterMap, parApply, parCapply, parLapply,
##     parLapplyLB, parRapply, parSapply, parSapplyLB
## The following objects are masked from 'package:stats':
##
##     IQR, mad, sd, var, xtabs
## The following objects are masked from 'package:base':
##
##     anyDuplicated, append, as.data.frame, cbind, colMeans,
##     colnames, colSums, do.call, duplicated, eval, evalq, Filter,
##     Find, get, grep, grepl, intersect, is.unsorted, lapply,
##     lengths, Map, mapply, match, mget, order, paste, pmax,
##     pmax.int, pmin, pmin.int, Position, rank, rbind, Reduce,
##     rowMeans, rownames, rowSums, sapply, setdiff, sort, table,
##     tapply, union, unique, unsplit, which, which.max, which.min
## Loading required package:  S4Vectors
##
## Attaching package:  'S4Vectors'
## The following object is masked from 'package:base':
##
##     expand.grid
## Loading required package:  IRanges
## Loading required package:  GenomeInfoDb

library(GenomicAlignments)

## Loading required package:  SummarizedExperiment
## Loading required package:  Biobase
## Welcome to Bioconductor
##
##     Vignettes contain introductory material; view with
##     'browseVignettes()'.  To cite Bioconductor, see
##     'citation("Biobase")', and for packages 'citation("pkgname")'.
```

```
## Loading required package:  DelayedArray
## Loading required package:  matrixStats
##
## Attaching package:  'matrixStats'
## The following objects are masked from 'package:Biobase':
##
##     anyMissing, rowMedians
##
## Attaching package:  'DelayedArray'
## The following objects are masked from 'package:matrixStats':
##
##     colMaxs, colMins, colRanges, rowMaxs, rowMins, rowRanges
## The following object is masked from 'package:base':
##
##     apply
## Loading required package:  Biostrings
## Loading required package:  XVector
##
## Attaching package:  'Biostrings'
## The following object is masked from 'package:DelayedArray':
##
##     type
## The following object is masked from 'package:base':
##
##     strsplit
## Loading required package:  Rsamtools

library(rtracklayer)
library(parallel)
source(file = "~/ISGlobal/Scripts/Funcions_1.R")
```

```
in_path <- "/home/lucas/ISGlobal/Chip_Seq/DATA/Aligns"
files <- list.files(path = "/home/lucas/ISGlobal/Chip_Seq/DATA/Aligns")
bams <- files[substrR(files, 8) == "sort.bam"]
bams <- lapply(bams, function(x) paste0(in_path,"/",x))
me_bams <- bams[substrR(bams, 11) == "me_sort.bam"]
aligns <- lapply(me_bams, readGAlignmentPairs)
```

0.1   Calculating Coverage

```
cov <- lapply(aligns, coverage)
```

1 Normalization

1.1 By Number of Mapped Reads

To get the number of mapped reads in the sample we use `samtools` as follows:

```
samtools view -F 0x4 foo.sorted.bam | cut -f 1 | sort | uniq | wc -l
-F : tells samtools to exclude given flag. Flag 0x4 means unaligned.
```

```r
nreads <- c()
for (file in me_bams) {
  cmd <- paste0("/home/lucas/Programs/samtools-1.3.1/samtools view -F 0x4 ", file, " | cut -
  nreads <- c(nreads,system(cmd, intern = TRUE))
}
names(nreads) <- me_bams
```

1.2 Normalize Coverage

Normalize the coverage divding by the number of mapped reads (abd multiplying by a huge factor).

```r
norm_cov <- c()
for (i in range(1,length(nreads))){
  norm_cov <- c(norm_cov, cov[[i]]/as.numeric(nreads[i])*1000000000)
}
names(norm_cov) <- me_bams

## Error in names(norm_cov) <- me_bams:  'names' attribute [5] must
be the same length as the vector [2]
```

```r
for (i in range(1, length(norm_cov))){
  con <- gsub("_sort.bam", "_cov.bed", bams[i])
  export(norm_cov[[i]], con = con, format = "BED")
}
```

2 Substractions

```r
subs_list <- combn(me_bams, 2, simplify = FALSE)
for (element in subs_list){
  substract <-
}
```

```
substraction_A7K9_E5K9 <- norm_cov[[1]] - norm_cov[[2]]
export(substraction_A7K9_E5K9, con = "/home/lucas/ISGlobal/Chip_Seq/DATA/Substractions/A7K9_

smooth_sub <- runmean(substraction_A7K9_E5K9, 10, endrule = "constant")
export(smooth_sub, con = "/home/lucas/ISGlobal/Chip_Seq/DATA/Substractions/A7K9_E5K9_substr_

smooth_sub <- runmean(substraction_A7K9_E5K9, 1000, endrule = "constant")
export(smooth_sub, con = "/home/lucas/ISGlobal/Chip_Seq/DATA/Substractions/A7K9_E5K9_substr_

smoothviews <- lapply(substraction_A7K9_E5K9, function(x) {successiveViews(x, width = 10)})
lapply(smoothviews, function(x) {mean(x)})
#class(substraction_A7K9_E5K9)
```

3   Session Info

```
sessionInfo()

## R version 3.4.1 (2017-06-30)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 16.04.3 LTS
##
## Matrix products: default
## BLAS: /usr/lib/openblas-base/libblas.so.3
## LAPACK: /usr/lib/libopenblasp-r0.2.18.so
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=es_ES.UTF-8        LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=es_ES.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=es_ES.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=es_ES.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] parallel  stats4    stats     graphics  grDevices utils     datasets
## [8] methods   base
##
## other attached packages:
##  [1] rtracklayer_1.36.4        GenomicAlignments_1.12.2
##  [3] Rsamtools_1.28.0          Biostrings_2.44.2
##  [5] XVector_0.16.0            SummarizedExperiment_1.6.3
##  [7] DelayedArray_0.2.7        matrixStats_0.52.2
```

```
##  [9] Biobase_2.36.2            GenomicRanges_1.28.5
## [11] GenomeInfoDb_1.12.2       IRanges_2.10.3
## [13] S4Vectors_0.14.4          BiocGenerics_0.22.0
## [15] knitr_1.17
##
## loaded via a namespace (and not attached):
##  [1] magrittr_1.5             zlibbioc_1.22.0
##  [3] BiocParallel_1.10.1      lattice_0.20-35
##  [5] highr_0.6                stringr_1.2.0
##  [7] tools_3.4.1              grid_3.4.1
##  [9] Matrix_1.2-11            GenomeInfoDbData_0.99.0
## [11] bitops_1.0-6             RCurl_1.95-4.8
## [13] evaluate_0.10.1          stringi_1.1.5
## [15] compiler_3.4.1           XML_3.98-1.3
```