library(ChIPseeker)

library(mosaics)

library(QuasR)

library(dada2)

library(Rsamtools)

library(systemPipeR)

library(ChIPQC)

library(GenomicAlignments)

library(GenomicRanges)

#Specify the file you want, will try multiple later

chipdatabam <- "ENCFF000BWX.bam"

#function to read the bam file

readchipdata <- function(chipdatabam){

# Makes a single list using samtools function

.unlist <- function (x){

x1 <- x[[1L]]

if (is.factor(x1)){

structure(unlist(x), class = "factor", levels = levels(x1))

} else {

do.call(c, x)

}

}

chipbam\_field <- names(chipdatabam[[1]])

list <- lapply(chipbam\_field, function(y) .unlist(lapply(chipdatabam, "[[", y)))

chip\_df <- do.call("DataFrame", list)

names(chip\_df) <- chipbam\_field

return(chip\_df)

}

#Before scanning and reading the bam file make sure to use this function in order to call it later

readBAM <- function(chipdatabam){

bam <- scanBam(chipdatabam)}

#Loads the bam file

bam1 <- readBAM(chipdatabam)

#bam1 will list all the data in that bam file

bam1

#This step takes forever

moduleload(module="python")

#peak data can’t be in .bam format, it accepts text format so far

mypeaks <- read.delim ("chipdatatest.txt", header=T)

colnames(mypeaks) <- c("chrom", "chromStart", "chromEnd", "name","score", "strand", "fold.enrichment","log10.pvalue", "log10.qvalue", "peak") head(mypeaks)

##### using FastQ files

targets <- read.delim("chipdatatest.txt", comment.char = "#")

list(targets)

read.fasta(file = system.file("sequences/ct.fasta.gz", package = "seqinr"),

seqtype = c("DNA", "AA"), as.string = FALSE, forceDNAtolower = TRUE,

set.attributes = TRUE, legacy.mode = TRUE, seqonly = FALSE, strip.desc = FALSE,

bfa = FALSE, sizeof.longlong = .Machine$sizeof.longlong,

endian = .Platform$endian, apply.mask = TRUE)

# Function in R that reads fasta files https://www.rdocumentation.org/packages/seqinr/versions/3.4-5/topics/read.fasta