

01.5_CompareCredDfs.Rmd

Jason Torres

March 15, 2017

Setup libraries and directories

```
"%&%<- function(a,b) paste0(a,b)
library("dplyr")
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##   filter, lag
## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
```

```
library("data.table")
```

```
## -----
## data.table + dplyr code now lives in dtplyr.
## Please library(dtplyr)!
## -----
```

```
##
## Attaching package: 'data.table'
## The following objects are masked from 'package:dplyr':
##
##   between, first, last
```

```
library("ggplot2")
```

```
## Warning: package 'ggplot2' was built under R version 3.3.2
```

```
library("ggbio")
```

```
## 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:dplyr':
##
##   combine, intersect, setdiff, union
```

```

## The following objects are masked from 'package:stats':
##
##   IQR, mad, xtabs

## The following objects are masked from 'package:base':
##
##   anyDuplicated, append, as.data.frame, cbind, colnames,
##   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, rownames, sapply, setdiff,
##   sort, table, tapply, union, unique, unsplit

## Warning: replacing previous import 'ggplot2::Position' by
## 'BiocGenerics::Position' when loading 'ggbio'

## Need specific help about ggbio? try mailing
## the maintainer or visit http://tengfei.github.com/ggbio/

##
## Attaching package: 'ggbio'

## The following objects are masked from 'package:ggplot2':
##
##   geom_bar, geom_rect, geom_segment, ggsave, stat_bin,
##   stat_identity, xlim

library("gridExtra")

##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:BiocGenerics':
##
##   combine

## The following object is masked from 'package:dplyr':
##
##   combine

serv.dir <- "/Users/jtorres/FUSE/"
rds.dir <- serv.dir %>% "projects/wtsa/enhancer_driven/probe_design/rds/"
txt.dir <- serv.dir %>% "projects/wtsa/enhancer_driven/probe_design/txt/"

te.dir <- serv.dir %>% "reference/gwas/trans-ethnic_single/"

```

Function to determine number of SNPs in 99% credible set

```

num99 <- function(dframe){
  index <- match("PPA",names(dframe))
  vec <- sort(dframe[,index],decreasing=TRUE)
  count=0
  sum=0
  for (v in vec){
    count <- count + 1
    sum <- sum + v
    if (sum >= 0.99){
      break
    }
  }
}

```

```

}
out.df <- arrange(dframe,desc(PPA))[1:count,]
#print(out.df)
return(out.df)
}

```

Build Credible Set dataframe

```

build_cred_df <- function(){
  f.vec <- list.files(te.dir)
  out.df <- c()
  for (f in f.vec){
    fname <- te.dir %>% f #f.vec[1]
    loc.name <- strsplit(f,split="_")[[1]][1]
    temp.df <- fread("cat " %>% fname %>% " | zmore") %>%
      dplyr::select(one_of("ID","P","LOGBF","POST"))
    names(temp.df) <- c("SNPID","P","LNBF","PPA")
    CHR <- as.character(sapply(1:dim(temp.df)[1],function(i){
      snpid <- temp.df$SNPID[i]
      chrom <- strsplit(snpid,split=":")[[1]][1]
      return(chrom)
    })))
    POS <- as.integer(sapply(1:dim(temp.df)[1],function(i){
      snpid <- temp.df$SNPID[i]
      pos <- strsplit(snpid,split=":")[[1]][2]
      return(pos)
    })))
    temp.df$CHR <- CHR
    temp.df$POS <- POS
    LOCUS <- rep(loc.name,dim(temp.df)[1])
    temp.df$LOCUS <- LOCUS
    temp.df$STUDY <- rep("DIAMANTE.Morris",dim(temp.df)[1])
    temp.df <- num99(as.data.frame(temp.df))
    out.df <- rbind(out.df,temp.df)
  }
  chrom <- as.integer(sapply(1:dim(out.df)[1],function(i){
    c <- strsplit(out.df$CHR[i],split="chr")[[1]][2]
    return(c)
  })))
  out.df$chrom <- chrom
  out.df <- arrange(out.df,chrom,POS)
  out.df$SEGNUMBER <- 1:dim(out.df)[1]
  out.df <- dplyr::select(out.df,one_of("SNPID","CHR","POS",
    "SEGNUMBER","PPA","LOCUS","STUDY",
    "P","LNBF"))

  return(out.df)
}

te.df <- build_cred_df()
write.table(x=te.df,txt.dir%>% "fgwas-cred-diamante17.txt",sep="\t",quote=FALSE,row.names=F)

```