Package 'manhattanVAAST'

February 8, 2012

Type Package

Title test

Version 1.0

Date 2012-02-08

Author Zev

Maintainer <zev.kronenberg@gmail.com>

Description no

License none

R topics documented:

`manhattanVAAST-package`
manhattanVAAST

Description

manhattanVAAST plot VAAST pvalues over range. Similar to GWAS style manhattan plots.

Details

Package:	manhattanVAAST
Type:	Package
Version:	1.001
Date:	2012-02-08
License:	none

Author(s)

Zev N. Kronenberg

Maintainer: <zev.kronenberg@gmail.com> Lab page: <http://www.yandell-lab.org/software/vaast.html>

References

A probabilistic disease-gene finder for personal genomes Yandell M Huff CD Hu H Singleton M Moore B Xing J Jorde L Reese MG Genome Res. 2011 Jul

Using VAAST to Identify an X-Linked Disorder Resulting in Lethality in Male Infants Due to N-Terminal Acetyltransferase Deficiency Rope AF Wang K Evjenth R Xing J Johnston JJ Swensen JJ Johnson WJ Moore B Huff CD Bird LM Carey JC Opitz JM Stevens CA Jiang T Schank C Fain HD Robison R Dalley B Chin S South ST Pysher TJ Jorde LB Hakonarson H Lillehaug JR Biesecker LG Yandell M Arnesen T Lyon GJ Am J Hum Genet. 2011 Jul 15;89(1):28-43

See Also

~~ Optional links to other man pages, e.g. ~~

`manhattanVAAST` *a function to plot VAAST simple reports.*

Description

given some data and few lines parameters the package will generate a plot.

Usage

```
manhattanVAAST(n.features=23000, sig.level=0.05, vaast.simple, title, sig.line = TRUE, axis.text
```

Arguments

`n.features`	the number of genes or features VAAST scored
`sig.level`	The signficane level sig.level = 0.05
`vaast.simple`	The File to do work on
`title`	main title
`sig.line`	Do you want the significance line in your plot?
`axis.text`	Do you want the seqids on the plot?
`custom.xlab`	Place your x-label here
`sig.hjust`	move the sig.line text horizontally + right - left (on same scale). use large numbers.
`sig.vjust`	move the sig.line text vertically + up - down (on same scale)

Details

no additional details

Author(s)

Zev N. Kronenberg

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (n.features, sig.level, vaast.simple, title, sig.line = TRUE,
    axis.text = TRUE, custom.xlab = "index", sig.hjust = 0, sig.vjust = 0)
{
    dat <- read.table(vaast.simple, header = FALSE, skip = 1,
        sep = "\t", fill = TRUE)
    dat <- dat[dat$V4 > 0, ]
    dat <- dat[, 1:6]
    dat <- dat[grep(dat$V5, pattern = "chr"), ]
    pos.dat <- strsplit(as.character(dat$V5), split = ":|;",
        perl = TRUE)
    dat$seqid <- sapply(pos.dat, FUN = function(xx) {
        return(xx[[1]][1])
    })
    dat$pos <- as.numeric(as.character(sapply(pos.dat, FUN = function(xx) {
        return(xx[[2]][1])
    })))
    dat <- dat[order(dat$seqid, dat$pos), ]
    dat$index.vec <- cumsum(1:length(dat$pos))
    dat$len.correction1 <- unlist(tapply(dat$pos, INDEX = dat$seqid,
        FUN = function(x) {
            c(max(x), rep(0, length(x) - 1))
        }, simplify = TRUE))
    dat$len.correction3 <- unlist(tapply(dat$pos, INDEX = dat$seqid,
        FUN = function(x) {
            rep(max(x), length(x))
        }, simplify = TRUE))
    dat <- dat[order(-dat$len.correction3, dat$seqid, dat$pos),
        ]
    dat$len.correction2 <- cumsum(dat$len.correction1)
    dat$relative.pos <- as.numeric(as.character(dat$len.correction2)) +
        as.numeric(as.character(dat$pos))
    axis.name <- unique(dat$seqid)
    axis.name.pos.rel <- as.vector(tapply(dat$index, INDEX = dat$seqid,
        FUN = function(x) {
            middle <- (max(x) + min(x))/2
        }))
    axis.name.pos.real <- sort(as.vector(tapply(dat$relative.pos,
        INDEX = dat$seqid, FUN = function(x) {
            middle <- (max(x) + min(x))/2
        })))
    plot(y = -log10(as.numeric(as.character(dat$V3))), x = dat$relative.pos,
        col = as.factor(dat$seqid), xaxt = "n", xlab = custom.xlab,
        pch = 20, ylab = "-log10(p-value)", main = title)
    if (sig.line == TRUE) {
        abline(h = -log10(sig.level/n.features), lty = 2, lwd = 3,
            col = "grey")
```

```
        text(x = (0.5 * max(dat$relative.pos)) + sig.hjust, y = 0.5 +
            -log10(sig.level/n.features) + sig.vjust, "genome-wide sig. level",
            cex = 1)
    }
    if (axis.text == TRUE) {
        axis(1, at = axis.name.pos.real, labels = axis.name,
            las = 2)
    }
    if (axis.text == FALSE) {
        axis(1, at = axis.name.pos.real, labels = FALSE, las = 2)
    }
    return(dat)
}
```

Index