GWAS_vis_vignette

Arcadio

12/13/2018

Loading up packages

```
library(rJava)
library(GenomicRanges)
library(SummarizedExperiment)
library(GWASpoly)
library(ggplot2)
library(ggrepel)
library(stringr)
library(rrBLUP)
library(plyr)
```

Set directory, loading up starting functions and rTassel

```
is_experimental <- TRUE

#set workdir for rtassel
setwd("~/myBins/bucklerlabBitbucket/rtassel/")

path_exp_tassel <- paste0(getwd(),"/inst/java/sTASSEL.jar")
path_exp_tassel_libs <- paste0(getwd(),"/inst/java/lib")

## jinit
rJava::.jinit(parameters="-Xmx6g")
.jcall(.jnew("java/lang/Runtime"), "J", "totalMemory")
```

```
## [1] 257425408
```

```
.jcall(.jnew("java/lang/Runtime"), "J", "maxMemory")
```

```
## [1] 5726797824
```

```
## Add class paths
if(is_experimental == TRUE) {
  tasselPath <- path_exp_tassel
  tasselLibs <- path_exp_tassel_libs
}

rJava::.jaddClassPath(tasselPath)
rJava::.jaddClassPath(tasselLibs)
print(.jclassPath())
```

```
## [1] "/Users/jav246/myBins/R-packages/rJava/java"
## [2] "/Users/jav246/myBins/bucklerlabBitbucket/rtassel/inst/java/sTASSEL.jar"
## [3] "/Users/jav246/myBins/bucklerlabBitbucket/rtassel/inst/java/lib"
```

```
## Source files
source("R/AllGenerics.R")
source("R/AllClasses.R")
source("R/TasselPluginWrappers.R")
source("R/PullFunctions.R")
source("R/GWASVisAnnotFuncs.R")
```

Load up genotypes implementing rTassel

```
geno_fileName <- "/Users/jav246/myBins/bucklerlabBitbucket/rtassel/data/mdp_genotype.hmp.txt"

## Make genotype table from tasses sample data
tasGenoTable <- readGenotypeTable(geno_fileName)

## Make summarized experiment from genotypetable
tas_se <- summarizeExperimentFromGenotypeTable(tasGenoTable)

tas_se
```

```
## class: RangedSummarizedExperiment
## dim: 3093 281
## metadata(0):
## assays(1): ''
## rownames: NULL
## rowData names(3): tasselIndex refAllele altAllele
## colnames(281): 33-16 38-11 ... WF9 YU796NS
## colData names(3): Sample TasselIndex
##   matrix.unlist.fourNewCols...nrow...length.fourNewCols...byrow...T.
```

```
genoDF <- GWASpolyGenoFromSummarizedExperiment(tas_se)

dim(genoDF)
```

```
## [1] 3093  284
```

```
genoDF[1:4, 1:8]
```

```
##   markerName chr     pos 33-16 38-11 4226 4722 A188
## 1    dummy-1   1  157104     0     0    0    0    0
## 2    dummy-2   1 1947984     0     2    0    2    0
## 3    dummy-3   1 2914066     0     0    0    0    0
## 4    dummy-4   1 2914171     0     0    0    0    0
```

```
#writting data for gwas poly
write.table(genoDF, "~/Downloads/GWASpoly_download/maizeGenotypes_GWASpoly.txt", sep = "\t", col.names =
```

Load phenotype data

```
###straight load as dataframe, skpping first two rows on tassel specific phenotype table format
pheno_fileName <- "/Users/jav246/myBins/bucklerlabBitbucket/rtassel/data/mdp_phenotype.txt"
phenos <- read.table(file = pheno_fileName, skip = 2, header = T, sep = "\t", na.strings = "NaN")
summary(phenos)
```

```
##       Taxa     location     EarHT            dpoll          EarDia
##  33-16   :  2   A:283    Min.   :  6.40   Min.   :52.60   Min.   :23.72
##  38-11   :  2   B:280    1st Qu.: 48.50   1st Qu.:63.50   1st Qu.:34.35
##  4226    :  2            Median : 60.20   Median :67.50   Median :37.00
##  4722    :  2            Mean   : 61.58   Mean   :67.78   Mean   :37.06
##  A188    :  2            3rd Qu.: 72.50   3rd Qu.:71.50   3rd Qu.:40.09
##  A214N   :  2            Max.   :138.80   Max.   :85.80   Max.   :49.30
##  (Other):551             NA's   :4        NA's   :7       NA's   :37
##        Q1                Q2               Q3
##  Min.   :0.0010   Min.   :0.0010   Min.   :0.0000  
##  1st Qu.:0.0020   1st Qu.:0.0050   1st Qu.:0.0020  
##  Median :0.0100   Median :0.5700   Median :0.0190  
##  Mean   :0.1744   Mean   :0.5011   Mean   :0.3245  
##  3rd Qu.:0.1205   3rd Qu.:0.9680   3rd Qu.:0.7940  
##  Max.   :0.9990   Max.   :0.9980   Max.   :0.9980  
## 
```

```
### select sinlge location, as GWASpoly requires single entries for taxa.
phenosOneLoc <- phenos[phenos$location == "A",]
rownames(phenosOneLoc) <- phenosOneLoc$Taxa
###remove location as it is now redundant.
###Also, GWASpoly expects all traits as initial columns, and fixed effect covariates last
phenosOneLoc <- phenosOneLoc[,-c(2)]

summary(phenosOneLoc)
```

```
##       Taxa          EarHT            dpoll           EarDia     
##  33-16  :  1   Min.   :  8.00   Min.   :54.50   Min.   :23.72  
##  38-11  :  1   1st Qu.: 48.12   1st Qu.:64.00   1st Qu.:34.86  
##  4226   :  1   Median : 60.50   Median :67.50   Median :37.32  
##  4722   :  1   Mean   : 61.75   Mean   :67.75   Mean   :37.20  
##  A188   :  1   3rd Qu.: 73.00   3rd Qu.:71.50   3rd Qu.:40.02  
##  A214N  :  1   Max.   :136.00   Max.   :85.00   Max.   :46.35  
##  (Other):277   NA's   :1        NA's   :3       NA's   :33     
##        Q1               Q2              Q3        
##  Min.   :0.0010   Min.   :0.001   Min.   :0.0000  
##  1st Qu.:0.0020   1st Qu.:0.005   1st Qu.:0.0020  
##  Median :0.0090   Median :0.579   Median :0.0230  
##  Mean   :0.1728   Mean   :0.502   Mean   :0.3253  
##  3rd Qu.:0.1160   3rd Qu.:0.968   3rd Qu.:0.7940  
##  Max.   :0.9990   Max.   :0.998   Max.   :0.9980  
## 
```

```
write.table(phenosOneLoc, "~/Downloads/GWASpoly_download/maizePhenotypes_GWASpoly.txt", sep = "\t", col
```

Run GWAS

```
## create GWASpoly object with coopted read.GWASpoly function
#uses tassel created summarizedExperiment for genotypes
data_gwasPoly <- se_createGWASpolyObject(ploidy = 2, phenoDF = phenosOneLoc,
                                         SummarizedExperimentObject =  tas_se,
                                         format = "numeric", n.traits = 3)
```

```
## Number of polymorphic markers: 3093 
## Missing marker data imputed with population mode 
## N = 264 individuals with phenotypic and genotypic information 
## Detected following fixed effects:
## Q1
## Q2
## Q3
## Detected following traits:
## EarHT
## dpoll
## EarDia
```

```
#same as above, but reading written files
data_gwasPoly2 <- read.GWASpoly(ploidy = 2, pheno.file = "~/Downloads/GWASpoly_download/maizePhenotypes_
```

```
## Number of polymorphic markers: 3093
## Missing marker data imputed with population mode
## N = 264 individuals with phenotypic and genotypic information
## Detected following fixed effects:
## Q1
## Q2
## Q3
## Detected following traits:
## EarHT
## dpoll
## EarDia
```

```
all.equal(current = data_gwasPoly, target = data_gwasPoly2)
```

```
## [1] "Attributes: < Component \"fixed\": Attributes: < Component \"row.names\": Modes: numeric, chara
## [2] "Attributes: < Component \"fixed\": Attributes: < Component \"row.names\": target is numeric, cu
## [3] "Attributes: < Component \"pheno\": Attributes: < Component \"row.names\": Modes: numeric, chara
## [4] "Attributes: < Component \"pheno\": Attributes: < Component \"row.names\": target is numeric, cu
## [5] "Attributes: < Component \"pheno\": Component \"Taxa\": Modes: character, numeric >"
## [6] "Attributes: < Component \"pheno\": Component \"Taxa\": Attributes: < target is NULL, current is
## [7] "Attributes: < Component \"pheno\": Component \"Taxa\": target is character, current is factor >
```

```
## add kinship information to object
data_gwasPoly <- set.K(data_gwasPoly)

## set parameters for mixed model
params <- set.params(fixed=unlist(strsplit("Q1,Q2,Q3", ",")),
                     fixed.type=rep("numeric",3))

## run gwas with GWASpoly
data_gwasPoly_res <- GWASpoly(data = data_gwasPoly, models = "additive",
                              params = params)
```

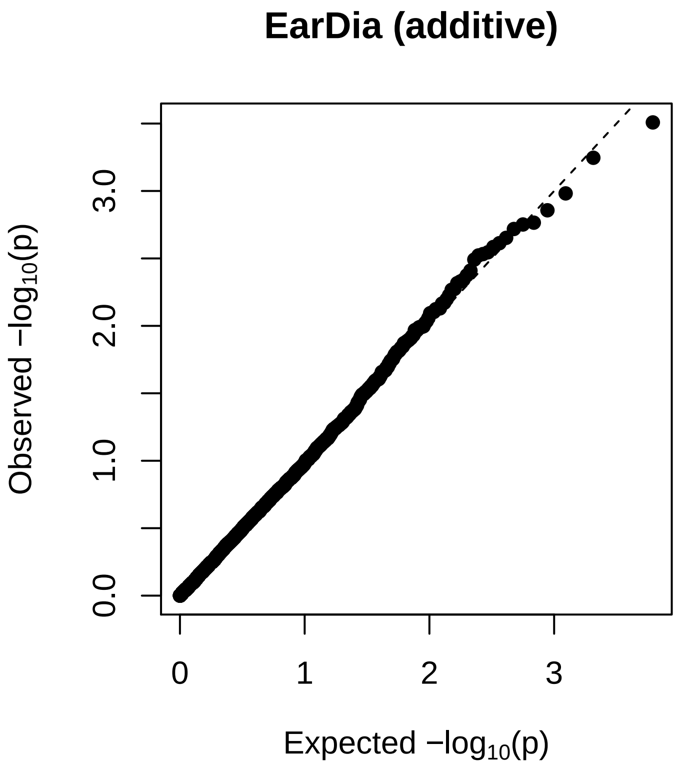
```
## Analyzing trait: EarHT
## P3D approach: Estimating variance components...Completed
## Testing markers for model: additive
## Analyzing trait: dpoll
## P3D approach: Estimating variance components...Completed
## Testing markers for model: additive
## Analyzing trait: EarDia
## P3D approach: Estimating variance components...Completed
## Testing markers for model: additive
```

```
#sanity check to ensure markers in object with scores matches markers order in genotype map
if(all(rownames(data_gwasPoly_res@scores$EarDia) == data_gwasPoly_res@map$Marker)){
  message("markers in scores and genotype map match, moving on")
}else{stop("marker names don't match ordering between geno map and scores object")}
```

```
## markers in scores and genotype map match, moving on
```

Create GWASpoly plots and set thresholds to get QTLs

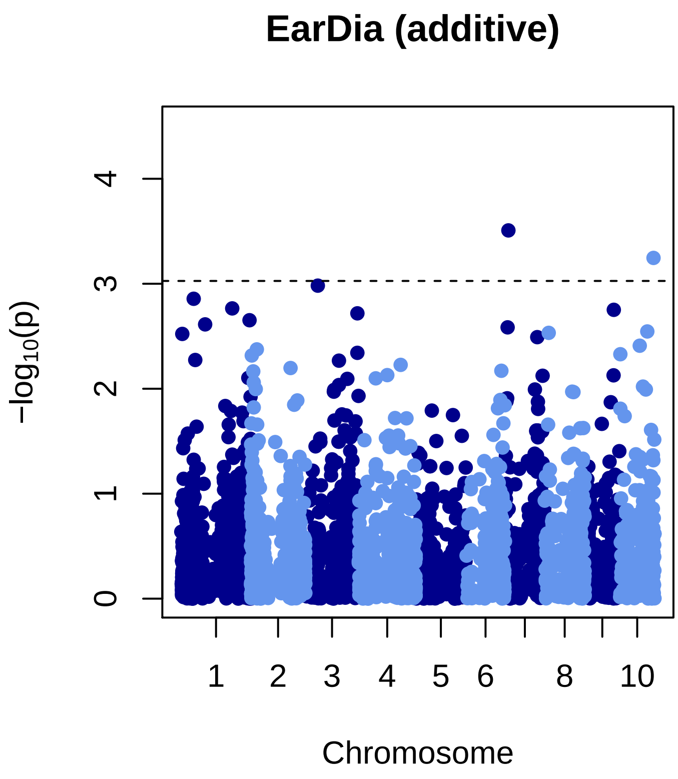
```
qq.plot(data_gwasPoly_res, trait = "EarDia", model = "additive")
```



```
## NULL
```

```
#can set Bonferroni/FDR and own pvalue
data_gwasPoly_res <- set.threshold(data_gwasPoly_res, method = "FDR", level=0.05)

#can set any of the 3 traits
manhattan.plot(data_gwasPoly_res, trait = "EarDia", model = "additive")
```



```
## NULL
```

```
get.QTL(data = data_gwasPoly_res)
```

```
##       Trait    Model Threshold     Marker Chrom  Position Ref Alt Score
## 2209 EarDia additive      3.03 dummy-2209     7  14349767   0   1  3.51
## 3080 EarDia additive      3.03 dummy-3080    10 144548839   0   1  3.25
##      Effect
## 2209  -1.15
## 3080  -1.08
```

Unwrap gwaspoly results class object

```
traitGWASresults <- gwasPolyToDF(data_gwasPoly_res)
```

```
## getting results for: EarHT
```

```
## getting results for: dpoll
```

```
## getting results for: EarDia
```

```
traitGWASresults[traitGWASresults$markerLogPVal> traitGWASresults$sigTreshold,]
```

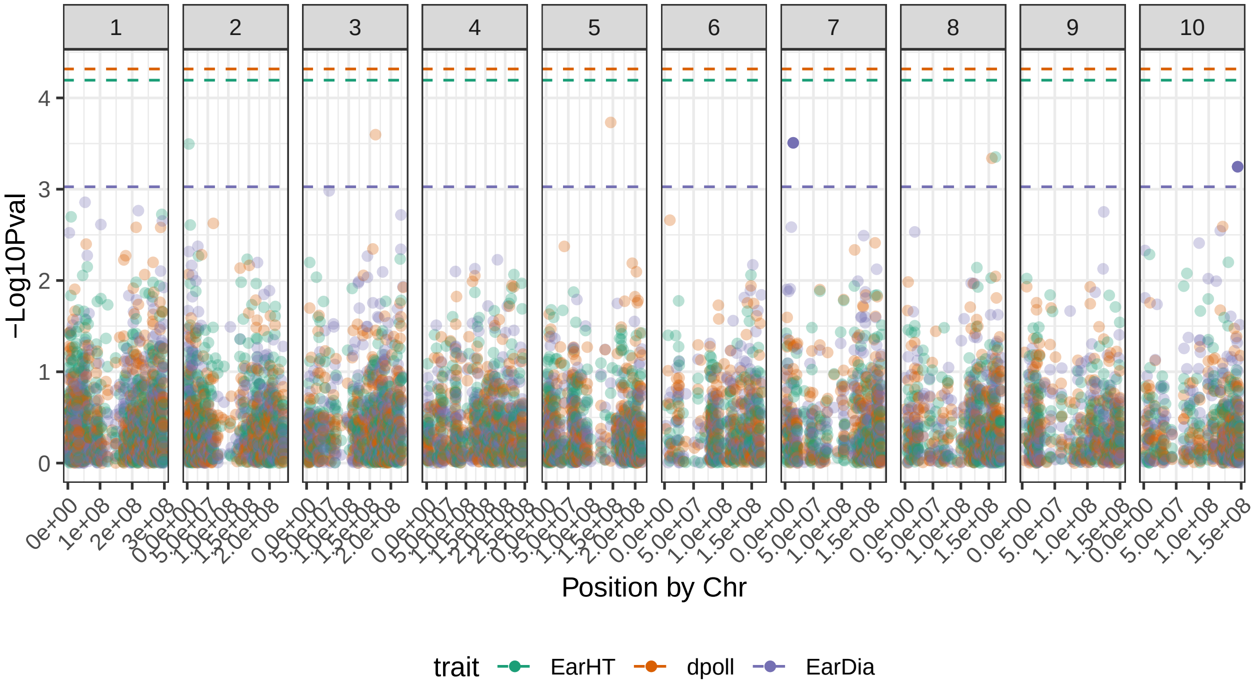
```
##          Marker markerpVal markerLogPVal markerEffect  trait sigTreshold
## 4036 dummy-2209 0.02995077      3.508200    -1.145989 EarDia    3.026561
## 6942 dummy-3080 0.03892190      3.246198    -1.083329 EarDia    3.026561
##      Chrom  Position Ref Alt
## 4036     7  14349767   0   1
## 6942    10 144548839   0   1
```

```
summary(traitGWASresults)
```

```
##         Marker       markerpVal      markerLogPVal       markerEffect
##  dummy-1    :    3   Min.   :0.02396   Min.   :0.000007   Min.   :-4.37836
##  dummy-10  :   3   1st Qu.:0.54946   1st Qu.:0.127346   1st Qu.:-0.29295
##  dummy-100 :   3   Median :0.73557   Median :0.307111   Median : 0.02879
##  dummy-1000:   3   Mean   :0.69642   Mean   :0.434249   Mean   : 0.07314
##  dummy-1001:   3   3rd Qu.:0.88043   3rd Qu.:0.598824   3rd Qu.: 0.36503
##  dummy-1002:   3   Max.   :0.99999   Max.   :3.731254   Max.   : 5.34599
##  (Other)   :9261
##     trait       sigTreshold        Chrom         Position
##  EarHT :3093   Min.   :3.027   1      :1620   Min.   :   139753
##  dpoll :3093   1st Qu.:3.027   2      :1179   1st Qu.: 43868122
##  EarDia:3093   Median :4.194   5      :1071   Median :128402775
##                Mean   :3.846   3      :1065   Mean   :119893324
##                3rd Qu.:4.317   4      : 957   3rd Qu.:175159119
##                Max.   :4.317   8      : 768   Max.   :299170077
##                                (Other):2619
##       Ref          Alt
##  Min.   :0   Min.   :1
##  1st Qu.:0   1st Qu.:1
##  Median :0   Median :1
##  Mean   :0   Mean   :1
##  3rd Qu.:0   3rd Qu.:1
##  Max.   :0   Max.   :1
##
```

Create simple manhattan like plot for all traits

```
manhattan_trait_plot(traitGWASresults = traitGWASresults, traitIDcol = "trait",
                     positionIDcol = "Position", chromIDcol = "Chrom", pValIDcol = "markerpVal")
```



Parse GFF file to get genes and create GenomicRanges object

```
maizeGFFgenesGR <- gffToGeneGR(gffFile = "~/Box/projectMaize/PHG/cimmyt_assemblies_analy/b73/Zea_mays.A
maizeGFFgenesGR
```

```
## GRanges object with 39179 ranges and 6 metadata columns:
```

```
##           seqnames        ranges strand |   Source annotType    other
##              <Rle>     <IRanges>  <Rle> | <factor>  <factor> <factor>
##         2        1   44289-49837      + |  gramene      gene        .
##        24        1   50877-55716      - |  gramene      gene        .
##       170        1   92299-95134      - |  gramene      gene        .
##       184        1 111655-118312      - |  gramene      gene        .
##       217        1 118683-119739      - |  gramene      gene        .
##       ...      ...           ...    ... .      ...       ...      ...
##   2804827       Pt 134341-134862      - |  gramene      gene        .
##   2804831       Pt 134923-135222      - |  gramene      gene        .
##   2804835       Pt 138323-139807      + |  gramene      gene        .
##   2804849       Pt 139824-140048      + |  gramene      gene        .
##   2804853       Pt 140068-140361      + |  gramene      gene        .
##             other2
##           <factor>
##         2        .
##        24        .
##       170        .
##       184        .
##       217        .
##       ...      ...
##   2804827        .
##   2804831        .
##   2804835        .
##   2804849        .
##   2804853        .
##
##
##         2                                 ID=gene:Zm00001d027230;biotype=protein_coding;description=
##        24                                                           ID=gene:Zm00001d027231;biotype=pr
##       170
##       184
##       217
##       ...
##   2804827          ID=gene:GRMZM5G885905;Name=ycf73-A;biotype=protein_coding;description=Uncharacteri
##   2804831 ID=gene:GRMZM5G866761;Name=ycf15-A;biotype=protein_coding;description=Putative uncharacteri
##   2804835
##   2804849
##   2804853
##                     Gene
##              <character>
##         2 Zm00001d027230
##        24 Zm00001d027231
##       170 Zm00001d027232
##       184 Zm00001d027233
##       217 Zm00001d027234
##       ...            ...
##   2804827  GRMZM5G885905
##   2804831  GRMZM5G866761
##   2804835  GRMZM5G818111
##   2804849  GRMZM5G866064
##   2804853  GRMZM5G855343
##   -------
##   seqinfo: 12 sequences from an unspecified genome; no seqlengths
```

Annotating SNPs with their closest gene. Best for annotating purposes.

```r
traitGWASresults_annotated <- annotate_gwasRes_byNearest(gwasResDF = traitGWASresults,
                  annotationGR = maizeGFFgenesGR, positionIDcol_gwas = "Position",
                  chromIDcol_gwas = "Chrom", outFmt = "data.frame")
```

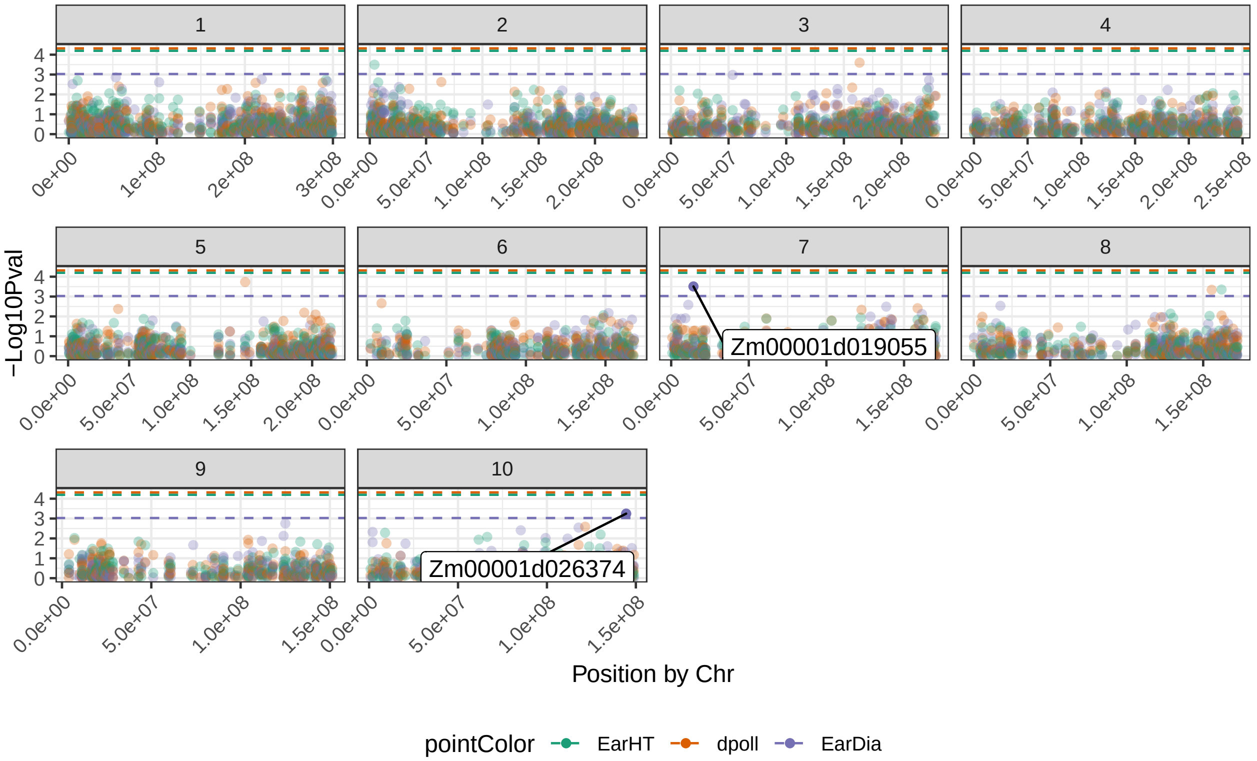
```
## Returning data.frame
```

```r
summary(traitGWASresults_annotated)
```

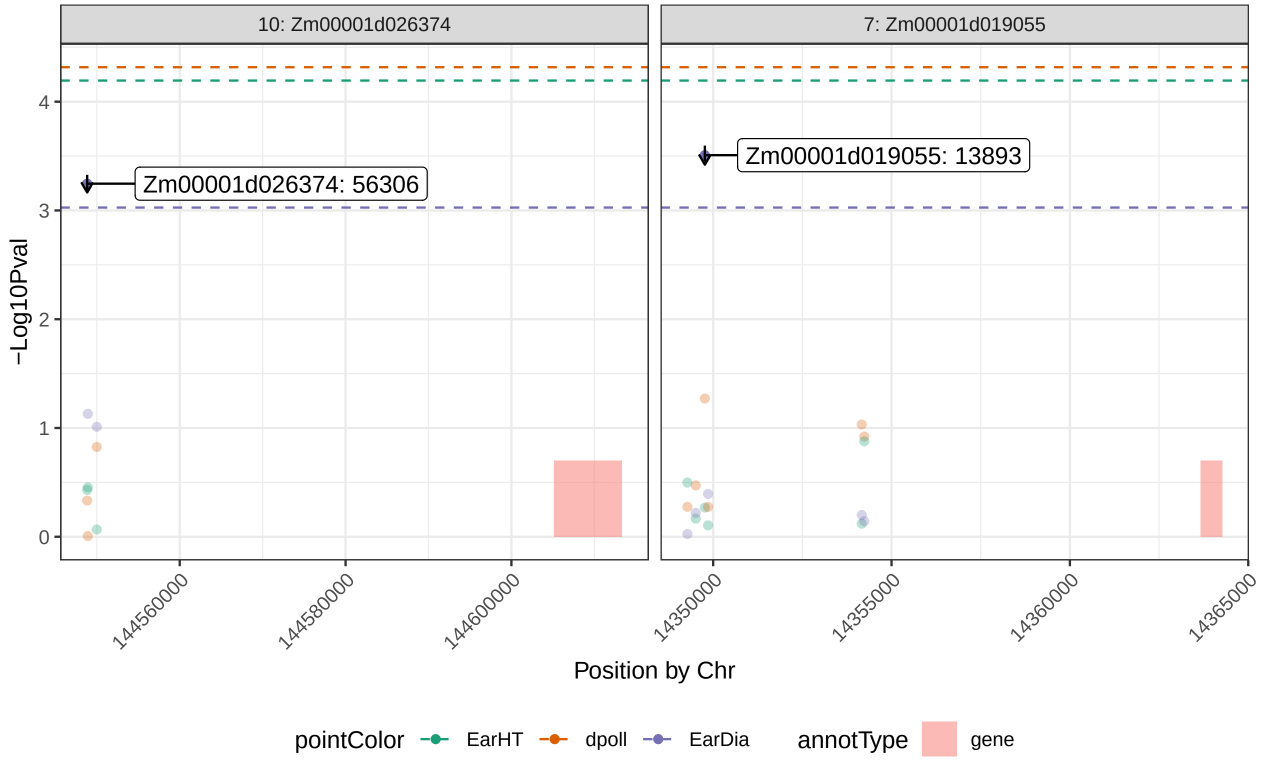
```
##      Marker_gwas   markerpVal_gwas   markerLogPVal_gwas markerEffect_gwas
##  dummy-1   :   3   Min.   :0.02396   Min.   :0.000007   Min.   :-4.37836
##  dummy-10  :   3   1st Qu.:0.54946   1st Qu.:0.127346   1st Qu.:-0.29295
##  dummy-100 :   3   Median :0.73557   Median :0.307111   Median : 0.02879
##  dummy-1000:   3   Mean   :0.69642   Mean   :0.434249   Mean   : 0.07314
##  dummy-1001:   3   3rd Qu.:0.88043   3rd Qu.:0.598824   3rd Qu.: 0.36503
##  dummy-1002:   3   Max.   :0.99999   Max.   :3.731254   Max.   : 5.34599
##  (Other)   :9261
##   trait_gwas   sigTreshold_gwas   Chrom_gwas   Position_gwas
##  EarHT :3093   Min.   :3.027    1      :1620   Min.   :   139753
##  dpoll :3093   1st Qu.:3.027    2      :1179   1st Qu.: 43868122
##  EarDia:3093   Median :4.194    5      :1071   Median :128402775
##                Mean   :3.846    3      :1065   Mean   :119893324
##                3rd Qu.:4.317    4      : 957   3rd Qu.:175159119
##                Max.   :4.317    8      : 768   Max.   :299170077
##                                 (Other):2619
##     Ref_gwas     Alt_gwas     seqnames        start
##  Min.   :0   Min.   :1   1      :1620   Min.   :   138378
##  1st Qu.:0   1st Qu.:1   2      :1179   1st Qu.: 43879919
##  Median :0   Median :1   5      :1071   Median :128422717
##  Mean   :0   Mean   :1   3      :1065   Mean   :119890122
##  3rd Qu.:0   3rd Qu.:1   4      : 957   3rd Qu.:175156310
##  Max.   :0   Max.   :1   8      : 768   Max.   :299188693
##                          (Other):2619
##       end                width       strand             Source
##  Min.   :   139043   Min.   :  198   +:4572   Ensembl_Plants:   0
##  1st Qu.: 43882845   1st Qu.: 1389   -:4707   gramene       :9279
##  Median :128427744   Median : 3090   *:   0   wareLab       :   0
##  Mean   :119895000   Mean   : 4879
##  3rd Qu.:175162258   3rd Qu.: 5729
##  Max.   :299193386   Max.   :89500
##
##          annotType    other    other2
##  gene          :9279   .:9279   .:9279
##  CDS           :   0            0:   0
##  chromosome    :   0            1:   0
##  exon          :   0            2:   0
##  five_prime_UTR:   0
##  lnc_RNA       :   0
##  (Other)       :   0
##
##  ID=gene:Zm00001d033507;biotype=protein_coding;gene_id=Zm00001d033507;logic_name=maker_gene
##  ID=gene:Zm00001d026248;biotype=protein_coding;description=Putative RING zinc finger domain superfami
##  ID=gene:Zm00001d033603;biotype=protein_coding;gene_id=Zm00001d033603;logic_name=maker_gene
##  ID=gene:Zm00001d002184;biotype=protein_coding;description=Peroxisome biogenesis protein 22;gene_id=Z
##  ID=gene:Zm00001d002937;biotype=protein_coding;description=cytochrome P450 family 72 subfamily A pol
##  ID=gene:Zm00001d025528;biotype=protein_coding;description=NAD(P)-linked oxidoreductase superfamily p
##  (Other)
##      Gene            distanceToNearestAnnot
##  Length:9279        Min.   :      0
##  Class :character   1st Qu.:   4388
##  Mode  :character   Median :  16117
##                     Mean   :  28585
##                     3rd Qu.:  36861
##                     Max.   :1275769
##
```

Plot manhattan with nearest annotation on significant SNPs or by genomicRange set Can pass other ggplot functions to modify output visualization

```r
#plotting by chromosome
manhattan_annot_plot(annotatedGWASresults = traitGWASresults_annotated,
                     labelType = "annotationName", zoomGR = "none")
```



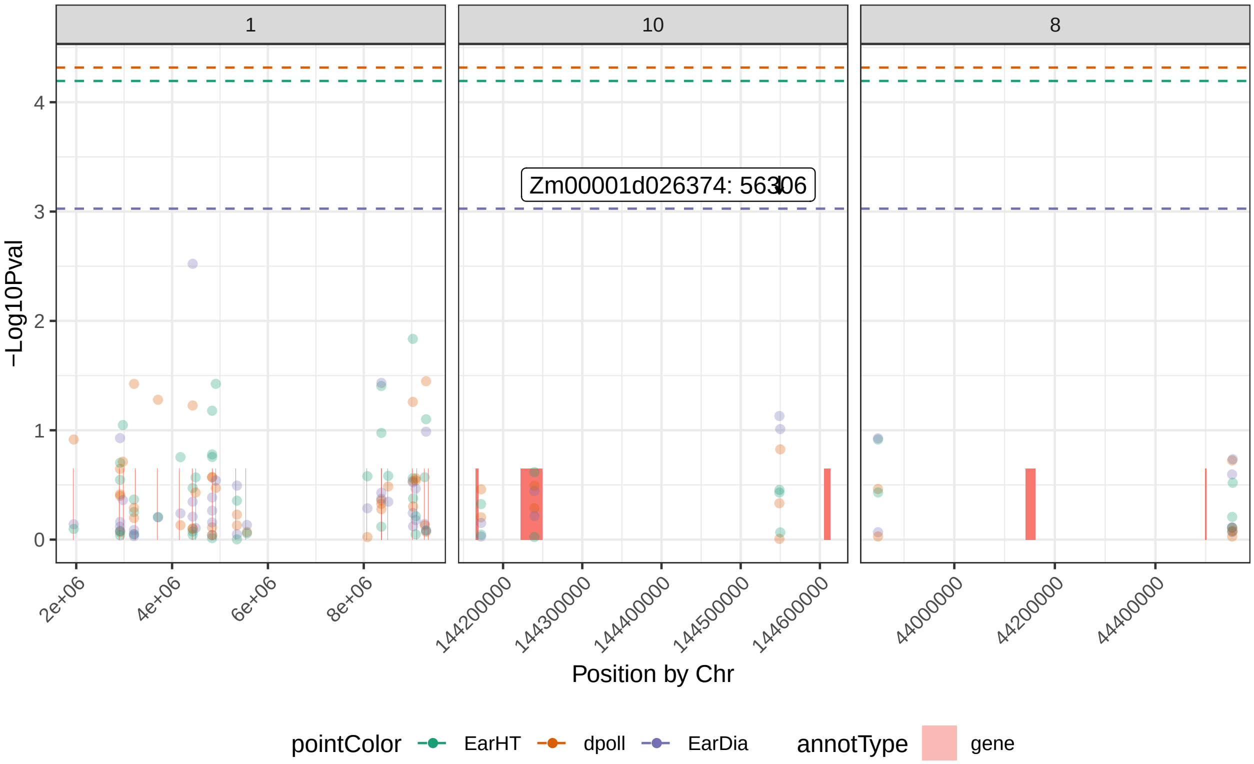
```r
#plotting with zoom over each closest significant annotations
manhattan_annot_plot(annotatedGWASresults = traitGWASresults_annotated,
                     labelType = "composite", zoomGR = "auto", annotateEffect = T,
                     effectSizeIDcol = "markerEffect_gwas")
```



```
#plotting with zoom over genomicRanges
#define genomic ranges of interest
myGRegions <- GRanges(seqnames = c(10, 8, 1),
                      ranges = IRanges(start=c(144605146-5e5, 44605146-5e5, 1e6),
                                       end=c(144605146+5e5, 44605146+5e5, 1e7)))
myGRegions
```

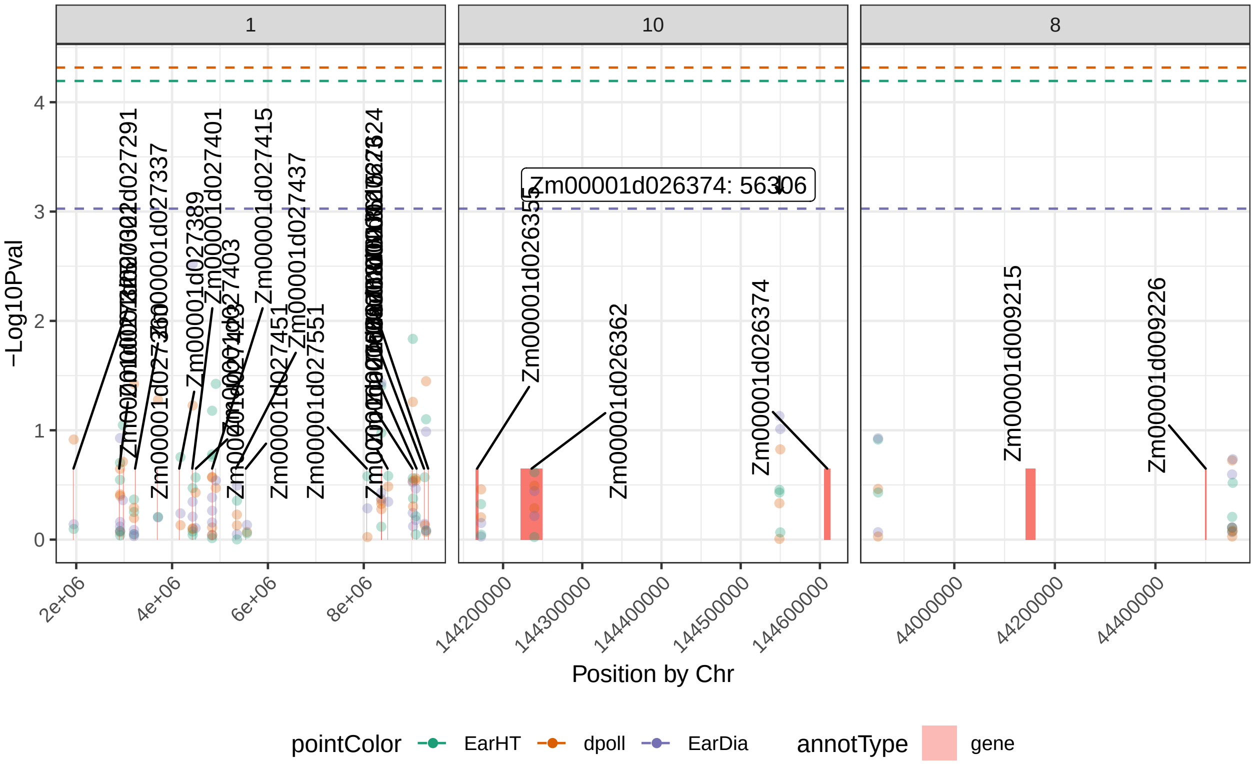
```
## GRanges object with 3 ranges and 0 metadata columns:
##       seqnames              ranges strand
##          <Rle>           <IRanges>  <Rle>
##   [1]       10 144105146-145105146      *
##   [2]        8   44105146-45105146      *
##   [3]        1     1000000-10000000      *
##   -------
##   seqinfo: 3 sequences from an unspecified genome; no seqlengths
```

```
#actual plotting on region still dependent on having any SNPs within the defined range
manhattan_annot_plot(annotatedGWASresults = traitGWASresults_annotated,
                     labelType = "composite", zoomGR = myGRegions,
                     annotateEffect = T, effectSizeIDcol = "markerEffect_gwas")
```



```
#inclde names of genes in the region
manhattan_annot_plot(annotatedGWASresults = traitGWASresults_annotated,
                     labelType = "composite", zoomGR = myGRegions, annotateEffect = T,
                     effectSizeIDcol = "markerEffect_gwas", labelAnnots = T)
```

```
## 25 annotations will be labeled, it might take a bit...
```



Get most significant SNP for each annotation/gene

Run gwas with rrBLUP

```
markers_rrblup_mat <- apply(genoDF[,-(1:3)],1,convert.snp)

dim(markers_rrblup_mat)
```

```
## [1]  281 3093
```

```
markers_rrblup <- data.frame(genoDF[,c(1:3)], t(markers_rrblup_mat))

colnames(markers_rrblup) <- colnames(genoDF)

dim(markers_rrblup)
```

```
## [1] 3093  284
```

```
markers_rrblup[1:4, 1:8]
```

```
##   markerName chr     pos 33-16 38-11 4226 4722 A188
## 1    dummy-1   1  157104    -1    -1   -1   -1   -1
## 2    dummy-2   1 1947984    -1     1   -1    1   -1
## 3    dummy-3   1 2914066    -1    -1   -1   -1   -1
## 4    dummy-4   1 2914171    -1    -1   -1   -1   -1
```

```
k_rrblup <- A.mat(markers_rrblup_mat)

phenosOneLoc_rrblup <- phenosOneLoc[phenosOneLoc$Taxa %in% colnames(markers_rrblup), ]

gwas_rrblup <- GWAS(pheno = phenosOneLoc_rrblup, geno = markers_rrblup, K = k_rrblup,
                    fixed = unlist(strsplit("Q1,Q2,Q3", ",")), P3D = T, n.core=6, plot = F)
```

```
## [1] "GWAS for trait: EarHT"
## [1] "Variance components estimated. Testing markers."
## [1] "GWAS for trait: dpoll"
## [1] "Variance components estimated. Testing markers."
## [1] "GWAS for trait: EarDia"
## [1] "Variance components estimated. Testing markers."
```