

# GWAS\_vis\_vignette

*Arcadio*

*12/13/2018*

Loading up packages

Set directory, loading up start 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/gwasPolyObjectCreator.R")
```

Load up genotypes implementing Tassel code through rJava

```
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) # not working right now, but

## Extracting chromosome names for each position...
## ...is there a quicker way to get this? (~ Brandon)
tas_se

## class: RangedSummarizedExperiment
## dim: 3093 281
## metadata(0):
## assays(1): ''
## rownames: NULL
## rowData names(0):
## colnames: NULL
## colData names(1): Sample

Load phenotype data

####straight load as dataframe, skipping 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
##	38-11	: 2	B:280	1st Qu.: 48.50	1st Qu.:63.50
##	4226	: 2		Median : 60.20	Median :67.50
##	4722	: 2		Mean : 61.58	Mean :67.78
##	A188	: 2		3rd Qu.: 72.50	3rd Qu.:71.50
##	A214N	: 2		Max. :138.80	Max. :85.80
##	(Other):551		NA's :4	NA's :7	NA's :37
##	Q1		Q2	Q3	
##	Min.	:0.0010	Min.	:0.0010	Min.
##	1st Qu.	:0.0020	1st Qu.	:0.0050	1st Qu.
##	Median	:0.0100	Median	:0.5700	Median
##	Mean	:0.1744	Mean	:0.5011	Mean
##	3rd Qu.	:0.1205	3rd Qu.	:0.9680	3rd Qu.
##	Max.	:0.9990	Max.	:0.9980	Max.
##					

```
#### select single 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
##	38-11	: 1	1st Qu.: 48.12	1st Qu.:64.00
##	4226	: 1	Median : 60.50	Median :67.50
##	4722	: 1	Mean : 61.75	Mean :67.75
##	A188	: 1	3rd Qu.: 73.00	3rd Qu.:71.50
##	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
##
```

Run GWASpoly

```
## create GWASpoly object with coopted read.GWASpoly function
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
```

```
## 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))

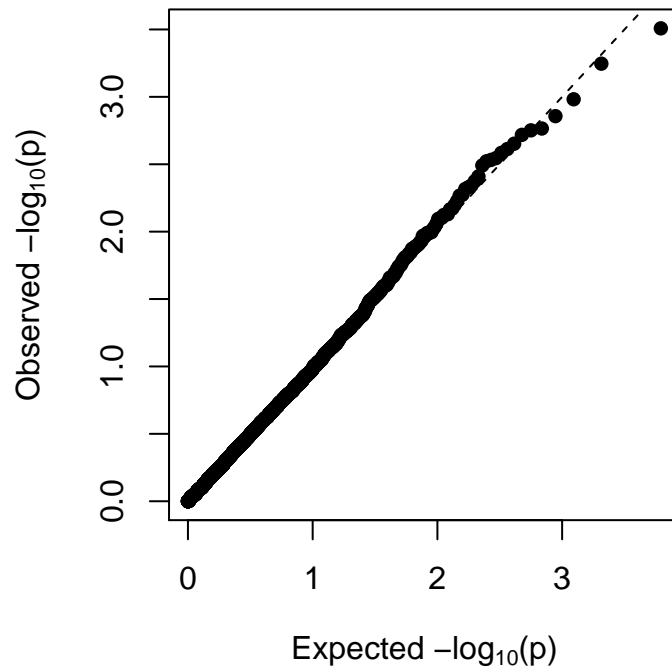
## run gwas
data_gwasPoly_res <- GWASpoly(data = data_gwasPoly, models = "additive",
                             params = params)
```

```
## 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
```

Create GWASpoly plots and set thresholds to get QTLs

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

## EarDia (additive)



```
## NULL
```

```
#can set Bonferroni and own pvalue
```

```
data_gwasPoly_res <- set.threshold(data_gwasPoly_res, method = "FDR", level=0.05)
```

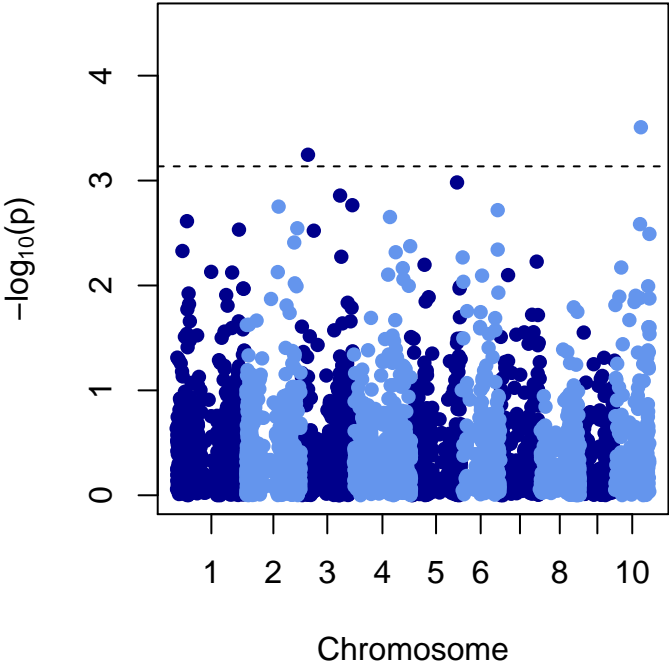
```
get.QTL(data = data_gwasPoly_res)
```

```
##      Trait    Model Threshold    Marker Chrom  Position Ref Alt Score
## 987 EarDia additive      3.14 dummy-987     3 31695534  0  1  3.25
## 2987 EarDia additive      3.14 dummy-2987    10 111608788  0  1  3.51
##      Effect
## 987    -1.08
## 2987    -1.15
```

```
#can set any of the 3 traits
```

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

EarDia (additive)



## NULL