A Brief Tutorial of the R Package vGWAS

Setup

To use the **vGWAS** package, of course, an R environment is required. Visit:

http://www.r-project.org

and install R for the operating system.

Start R and in the R console, type the following command to install the package:

```
install.packages('vGWAS', repos = 'http://r-forge.r-project.org')
```

If everything works fine, something like the following should show:

```
trying URL ...
Content type ... length ... bytes (... Kb)
opened URL
==================================================
downloaded ... Kb
```

Now the package is installed in the R library.

Example

In the R console, the command:

```
vignette('vGWAS')
```

opens this PDF document together with the package documentation. An example can be found in the documentation. Type:

```
require(vGWAS)
```

to load the package, then four main functions in the package are ready to use - `brown.forsythe.test`, `vGWAS`, `plot.vGWAS` (S3 method) and `vGWAS.heritability`. Run the following commands to load the example data:

```
data(pheno)
data(geno)
data(chr)
data(map)
```

`pheno` is a numeric vector of the simulated phenotypic values. By running:

```
hist(pheno, breaks = 30, density = 15, col = 'darkred')
```

a histogram of the phenotype distribution is produced as Figure 1.

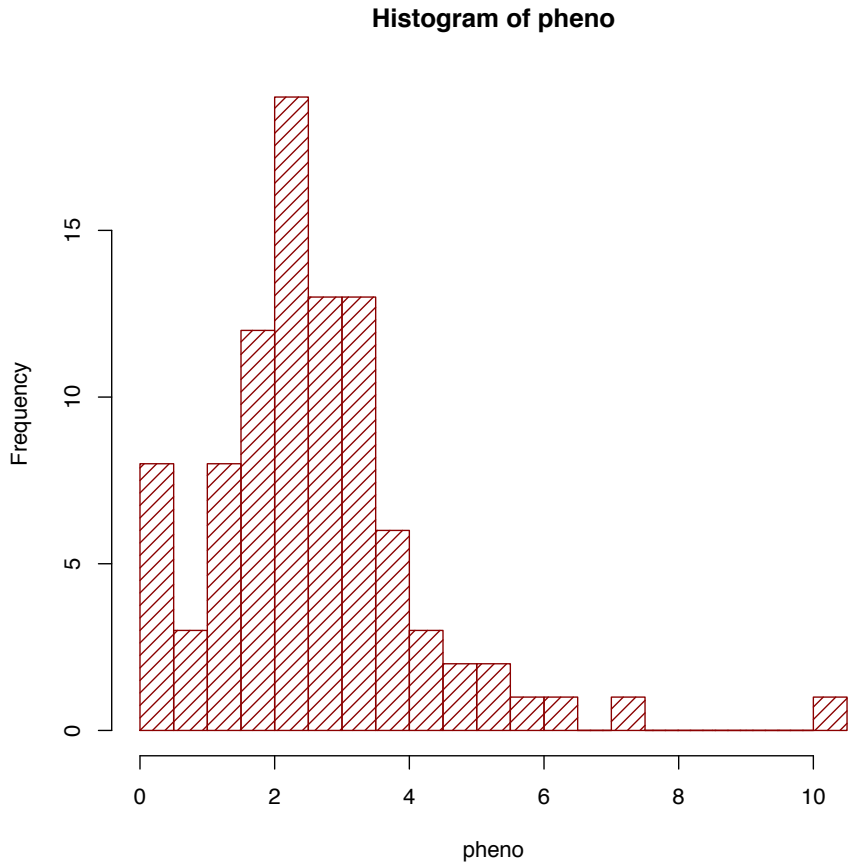

Figure 1: Histogram of the simulated phenotype.

The command:

```
table(chr)
```

produces:

```
chr
   1    2    3    4    5
5000 3000 4000 2000 6000
```

This shows exactly the number of markers on each of the five simulated chromosomes. Now, the *objects* loaded in R are ready for a vGWA scan, which can be done using the single command:

```
vgwa <- vGWAS(phenotype = pheno, geno.matrix = geno, marker.map = map,
              chr.index = chr)
```

A progress bar will show to indicate the progress of the scan, such like:

```
|===                                                              |   4%
```

When the scan is finished, all the output statistics will be returned as a *list* into the object vgwa, which belongs to the *class* 'vGWAS'. Any object that has a *structure* belonging to

class 'vGWAS' can be directly passed into S3 method function `plot.GWAS`. For instance, simply run the following command, we can plot the results in `vgwa`:

```
plot(vgwa)
```

which produces Figure 2. There is a clear peak above the Bonferroni corrected threshold (dashed orange line).

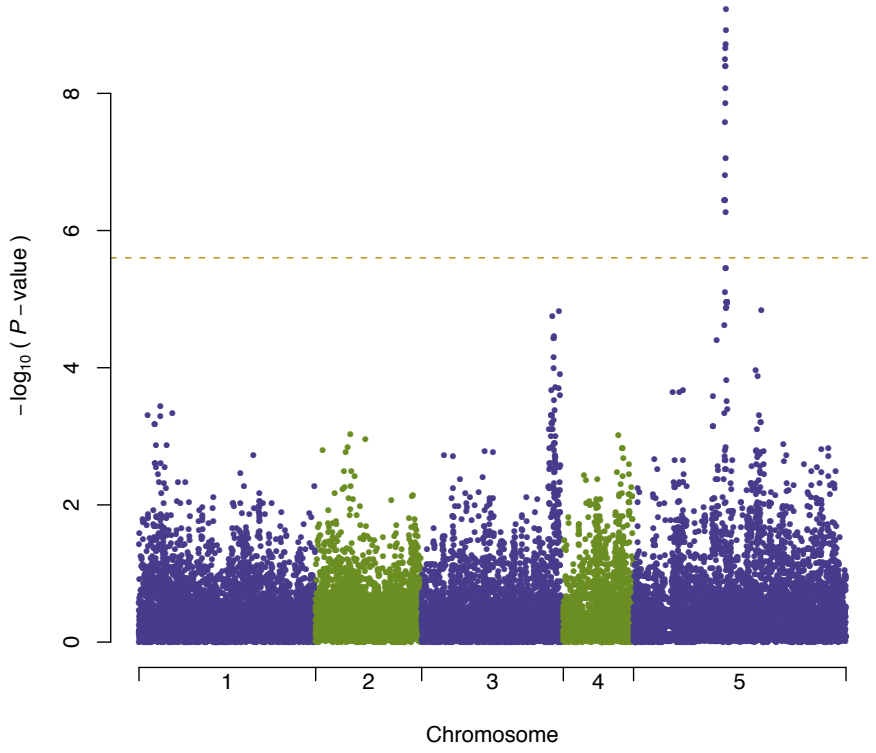

Figure 2: vGWA results of the simulated data.

Regarding the marker that gave the highest score, the heritability explained by the mean and variance can be split and calculated via:

```
vGWAS.heritability(phenotype = pheno,
                   marker.genotype = geno[,vgwa$p.value == min(vgwa$p.value)])
```

which prints out:

```
heritability explained by the mean part of model:
1.85 %
heritability explained by the variance part of model:
27.66 %
heritability in total:
29.51 %
```

The output can also be stored if assigning the function call to an object.

Remarks

The package source and further development information are on the R-Forge project page: https://r-forge.r-project.org/projects/vgwas/

Package 'vGWAS'

October 13, 2010

Type Package

Title Variance Genome-wide Association

Version 2010.10.12

Date 2010-10-12

Author Xia Shen

Maintainer Xia Shen <xia.shen@lcb.uu.se>

Description The package provides functions for genome-wide association using nonparametric variance test and some other further visualization and calculation.

License GPL

LazyLoad yes

Depends dglm

R topics documented:

vGWAS-package *Variance Genome-wide Association*

Description

The package provides functions for genome-wide association using nonparametric variance test and some other further visualization and calculation.

Details

Package:	vGWAS
Type:	Package
Version:	2010.10.12
Date:	2010-10-12
License:	GPL
LazyLoad:	yes
Depends:	dglm

Author(s)

Xia Shen

Maintainer: Xia Shen <xia.shen@lcb.uu.se>

References

Shen et al. (2010)

See Also

package `lawstat` for other types of nonparametric variance tests.

`brown.forsythe.test`

Brown-Forsythe's Test of Equality of Variances

Description

The function performs the robust Brown-Forsythe test using the group medians. Instead of the ANOVA statistic, the Kruskal-Wallis ANOVA may also be applied using this function.

Usage

```
brown.forsythe.test(y, group, kruskal.test=FALSE)
```

Arguments

`y`	a numeric vector of data values.
`group`	factor of the data.
`kruskal.test`	a `logical` value specifying whether to use Kruskal-Wallis statistic. The default option is `FALSE`, i.e., the usual ANOVA statistic is used in place of Kruskal-Wallis statistic.

Details

Levene (1960) proposed a test for homogeneity of variances in k groups which is based on the ANOVA statistic applied to absolute deviations of observations from the corresponding group mean. The robust Brown-Forsythe version of the Levene-type test substites the group mean by the group median in the classical Levene statistic.

Value

A list with the following numeric components.

`statistic`	the value of the test statistic.
`p.value`	the p-value of the test.
`method`	type of test performed.
`data.name`	a character string giving the name of the data.

Acknowledgement

The authors of package `lawstat` is ackownledged for their source code under free GPL license.

Note

Modified from the `lawstat` package.

Author(s)

Xia Shen

References

Brown, M. B. and Forsythe, A.B. (1974). *Robust tests for equality of variances.* Journal of the American Statistical Association, **69**, 364-367.

Levene, H. (1960). *Robust Tests for Equality of Variances*, in Contributions to Probability and Statistics, ed. I. Olkin, Palo Alto, CA: Stanford Univ. Press.

Examples

```
## Not run:

data(pheno)
data(geno)
brown.forsythe.test(pheno, geno[,911])


## End(Not run)
```

---

chr *Chromosome Indices for The Markers of The Simulated Data*

Description

Chromosome indices for the markers of the simulated data

Usage

```
data(chr)
```

Format

A numeric vector of chromosome indices for the 20K simulated markers.

Examples

```
data(chr)
table(chr)
```

---

geno *The Marker Genotypes of The Simulated Data*

Description

The marker genotypes of the simulated data

Usage

```
data(geno)
```

Format

A character matrix of size (number of individuals) times (number of markers in the genome).

Details

Note that there is only one column for each marker.

Examples

```
data(geno)
```

---

map *Map Positions for The Markers of The Simulated Data*

Description

Map positions for the markers of the simulated data

Usage

```
data(chr)
```

Format

A numeric vector of chromosomal map positions of the 20K simulated markers.

Examples

```
data(map)
```

`pheno` *Phenotypic Values for The Markers of The Simulated Data*

Description

Phenotypic values for the markers of the simulated data

Usage

```
data(pheno)
```

Format

A numeric vector of the phenotypic values of 93 simulated individuals.

Examples

```
data(pheno)
hist(pheno, breaks = 30)
```

`plot.vGWAS` *Variance GWA Manhattan Plot*

Description

The function plots the variance GWA result for the giving scan object.

Usage

```
## S3 method for class 'vGWAS':
plot(x, sig.threshold = NULL, low.log.p = 0, pch = 16,
           cex = 0.6, col.manhattan = c("slateblue4", "olivedrab"),
           col.sig.threshold = "darkgoldenrod", ...)
```

Arguments

x	a result object from vGWAS scan. It can be any list or data.frame that contains chromosome, marker.map, and p.value, with class = 'vGWAS'. See vGWAS.
sig.threshold	a numeric value giving the significance threshold for -log(pvalues, 10). If NULL, Bonferroni correction will be used.
low.log.p	a numeric value giving the lower limit of the -log(pvalues, 10) to plot.
pch	point character. See par.
cex	size of points. See par.
col.manhattan	two colors as a vector for the Manhattan plot.
col.sig.threshold	one color for the significance threshold.
...	not used.

Value

a plot for viewing vGWAS result.

Author(s)

Xia Shen

References

Shen et al. (2010)

See Also

vGWAS-package, vGWAS

Examples

```
## Not run:

# ----- load data ----- #

data(pheno)
data(geno)
data(chr)
data(map)

# ----- variance GWA scan ----- #

vgwa <- vGWAS(phenotype = pheno, geno.matrix = geno,
              marker.map = map, chr.index = chr)

# ----- visualize the scan ----- #

plot(vgwa)

# ----- calculate the heritability of strongest the marker ----- #
```

```
vGWAS.heritability(phenotype = pheno,
                   marker.genotype = geno[,vgwa$p.value == min(vgwa$p.value)])


## End(Not run)
```

---

vGWAS *Variance Genome-wide Association*

Description

Genome-wide association for using nonparametric variance test

Usage

```
vGWAS(phenotype, geno.matrix, kruskal.test = FALSE,
      marker.map = NULL, chr.index = NULL)
```

Arguments

`phenotype`	a `numeric` or `logical` vector of the phenotyic values. See **Examples**.
`geno.matrix`	a `matrix` or `data.frame` with individuals as rows and markers as columns. The marker genotypes for each marker are coded as one column. See **Examples**.
`kruskal.test`	a `logical` value specifying whether to use Kruskal-Wallis statistic. The default option is `FALSE`, i.e., the usual ANOVA statistic is used in place of Kruskal-Wallis statistic.
`marker.map`	a `numeric` vector giving the marker map positions for each chromosome. See **Examples**.
`chr.index`	a `numeric` vector giving the chromosome index for each marker. See **Examples**.

Value

a `data.frame` containing columns of `marker` names, `chromosome` indices, `marker.map` positions, test `statistic` values, and `p.value` for each position.

Author(s)

Xia Shen

References

Shen et al. (2010)

See Also

`vGWAS-package`

Examples

```
## Not run:

# ----- load data ----- #

data(pheno)
data(geno)
data(chr)
data(map)

# ----- variance GWA scan ----- #

vgwa <- vGWAS(phenotype = pheno, geno.matrix = geno,
              marker.map = map, chr.index = chr)

# ----- visualize the scan ----- #

plot(vgwa)

# ----- calculate the heritability of strongest the marker ----- #

vGWAS.heritability(phenotype = pheno,
                   marker.genotype = geno[,vgwa$p.value == min(vgwa$p.value)])


## End(Not run)
```

---

vGWAS.heritability *Calculating Heritability for A Single Marker*

Description

The function calculates and reports the heritability for a single marker by fitting a double generalized linear model. It gives both the heritability from the mean and variance parts of model.

Usage

```
vGWAS.heritability(phenotype, marker.genotype, only.print = TRUE)
```

Arguments

`phenotype`	a `numeric` vector of the phenotyic values. See **Examples**.
`marker.genotype`	a `numeric` or `character` or `factor` vector of the genotypes of a single marker. See **Examples**.
`only.print`	a `logical` value. If `FALSE`, the heritability values will be returned for storage.

Details

The **Value** will only be available if `only.print = FALSE`.

Value

`heritability.mean`
: the heritability from the mean part of model.

`heritability.disp`
: the heritability from the variance part of model.

Author(s)

Xia Shen

References

Shen et al. (2010)

See Also

`vGWAS-package`, `vGWAS`, `plot.vGWAS`

Examples

```
## Not run:

# ----- load data ----- #

data(pheno)
data(geno)
data(chr)
data(map)

# ----- variance GWA scan ----- #

vgwa <- vGWAS(phenotype = pheno, geno.matrix = geno,
              marker.map = map, chr.index = chr)

# ----- visualize the scan ----- #

plot(vgwa)

# ----- calculate the heritability of strongest the marker ----- #

vGWAS.heritability(phenotype = pheno,
                   marker.genotype = geno[,vgwa$p.value == min(vgwa$p.value)])


## End(Not run)
```

Index