Tutorial on using genphen

Simo Kitanovski,
Bioinformatics, University of Duisburg-Essen,
Essen, Germany

April 24, 2017

Contents

This tutorial gives you some of the technical background underlying genphen that should enable you to understand and use this tool.

1 genphen quantifies genotype-phenotype associations

Genome wide association studies (GWAS) have become an important tool to understand the association between genotypes and phenotypes. With GWAS we try to answer questions such as "what are the genotypes in the human genome which predispose to a disease?" or "what are the genotypes in certain strains of mice which allow them to be more resistant against a specific virus?". The are countless applications of genotype-phenotype

association studies, whereby the genotype can either be a set of single nucleotide polymorphisms (SNPs) found in nucleotide sequences, or a set of single amino acid polymorphisms (SAAPs) found at sites in specific protein sequence. The phenotype can be any measured quantity related to the individuals or sequences in which the polymorphic genotypes were found.

To conduct GWAS frequentist statistical methods are typically used, relying on P values to report the strength of the association between the genotypes and the phenotypes. This often leads to massive multiple hypothesis problems, typically answered with stringent P value correction methods, which are used to curb the false positives but have the effect of introducing many false negatives. More sophisticated statistical learning approaches such as random forest (RF) and support vector machines (SVM) alleviate the P value problem and are able to capture the most complex relationships between the genotypes and phenotypes but have the drawback of poor interpretability. Bayesian inference on the other hand, provides a solution to both the P value issue and the interpretabilitiy of the statistical learning approaches, but comes at a substantially higher computational cost.

Here we introduce `genphen`, a tool for GWAS which implements both statistical learning techniques such as RF and SVM as well as Bayesian inference using hierarchical models, to quantify the association between genotypes and continuous phenotypes. Additionally, `genphen` implements a set of visualization procedures which allow the user to inspect the vast results of the main association analyses and pinpoint the significant genotype-phenotype associations.

2 Conducting GWAS with `genphen`

2.1 Input

Two data types are necessary to perform a genetic association study, namely the genotype data and the phenotype data. As an example of genotype data imagine a set 1000 SNPs obtained for 10 different strains of laboratory mice taken from the Mouse Hapmap Project. The phenotype, on the other hand can be an experimental continuous measurement made for each of the 10 different mouse strains (e.g. height, body weight, temperature, immune response, etc.) (see SNP-Phenotype example in Fig 1a).

Another example of a genotype data can be a multiple sequence alignment (MSA) of 100 protein homologs (e.g. 120 aligned protein sequences of different organisms with 154 protein sites, some of which may contain amino acid polymorphisms). The phenotype can once again be continuous measurement made for each of the 120 organisms. Similar to the previous example,

in this case too we can use `genphen` to estimate the association between the polymorphic protein sites and a given phenotype of the organisms.

More specifically, we can think of the genotype data as a character matrix with dimensions N×M, whereby the M columns represent different SNPs or SAAPs, and the N rows represent different individuals or sequences for which we have measures some phenotype. On the other hand, we can think of the phenotype as a numerical vector of length N, where each phenotype corresponds to a particular individual.

2.2 Methods

The T-test is a popular method which is often used to quantify the association between the genotypes and phenotypes. Given a SNP like the one shown in Fig. 1a, the T-test checks whether there exists a significant difference between the two states (alleles) of the SNP w.r.t. the phenotype, whereby the strength of the association is summarized with a P value. `genphen` implements three statistical methods which perform similar tasks, but are superior to the T-test. These methods can be used through the following three interfaces:

- runGenphenRf - RF-based analysis
- runGenphenSvm - SVM-based analysis
- runGenphenBayes - Bayesian inference based analysis

2.2.1 runGenphenRf and runGenphenSvm

The procedures runGenphenRf and runGenphenSvm perform similar tasks using different statistical learning techniques, namely RF and SVM. The following metrics are estimated using each of them:

Classification accuracy(CA) This metric is used to quantify the strength of the association between each specific genotype and the phenotype. Both of the procedures generate classification models between the phenotype (a numerical predictor) and a specific genotype (a categorical response). If there exists a strong association between the genotype and the phenotype, one should be able to build an accurate classification model be ($CA = 1$, for a perfect classifier). To obtain a robust CA we applied cross-validation, where a subset of the genotype-phenotype data is selected at random for training the classifier, followed by testing based on the remaining data. The

final CA is the mean accuracy of resulting from this procedure. The following confusion matrix represents the result of one cross-validation step and is the data based on which CA is computed:

		Real	
		$allele_1$	$allele_2$
Predicted	$allele_1$	a	b
	$allele_2$	c	d

Table 1: Confusion matrix resulting from a classification analysis

The CA of the cross-validation step i is then estimated as:

$$CA_i = \frac{a+d}{a+b+c+d}$$

The final CA for N cross-validation steps is then estimated as:

$$CA = \frac{1}{N}\sum_{i=1}^{N} CA_i$$

In addition to estimating CA, one can also compute the 95% highest density interval (95% HDI) from the distribution of the individual CAs obtained throughout the cross-validation. In order to obtain reliable HDIs, the user should run these procedures with at least a hundred cross-validation steps (parameter nboots > 100). Those genotypes for which a CA is close to 1, coupled with a narrow CI, have the most significant association with the phenotype.

The metric CA has two advantages over the P values. First, it is more intuitive that a P value as a $CA = 0.9$ means that on average, one is able to correctly predict 90% of the states of the genotype from the phenotypes. Second, we do not run into multiple hypothesis problems and therefore there is no need for a P value correction. One drawback of CA is that it is difficult to find a "perfect" cuttoff to separate the strong associations from the weak ones.

Cohen's κ statistics Often we are interested in comparing the observed CA with classification accuracy which is expected simply by chance (CA_{exp}). This is in particular useful when the genetic states of the genotype are not evenly represented, i.e. allele A of a given SNP may be represented in 80% of the individuals, while the other allele T may be represented in only 20% of the individuals. Such uneven composition of the genotype can affect the classification analysis, resulting in high CAs simply because the

classifier only predicts the dominant label. Cohen's κ statistics can be used to estimate how much stronger the observed CA is, compared to CA_{exp}. To compute the κ statistics, the confusion matrix shown before in Table 1 is used:

$$\kappa = \frac{CA - CA_{exp}}{1 - CA_{exp}}$$

$$CA_{exp} = \frac{a+b}{a+b+c+d} \cdot \frac{a+c}{a+b+c+d} + \frac{c+d}{a+b+c+d} \cdot \frac{b+d}{a+b+c+d}$$

The κ statistics is a quality metric, which is to be used together with CA. Cohen defines the following meaningful κ intervals: [κ<0]: "no agreement", [0.0-0.2]: "slight agreement", [0.2-0.4]: "fair agreement" , [0.4-0.6]: "moderate agreement", [0.6-0.8]: "substantial agreement" and [0.8-1.0]: "almost perfect agreement". Similarly to the estimation of CA, the final Cohen's κ is also estimated by averaging the individual κs computed for each step of the cross-validation. Here too, highest density intervals are estimated.

Cohen's effect size (d) Using the package `effsize`, we compute the Cohen's d for each genotype-phenotype pair. High Cohen's d estimations indicate that there is a large difference in the measured phenotype between the two genetic states of the specific genotype. Cohen (1992) defines thresholds which define the magnitude of the effects as: $|d|$<0.2 "negligible", $|d|$<0.5 "small", $|d|$<0.8 "medium", otherwise "large". `genphen` computes both the Cohen's d statistics and the corresponding 95% confidence intervals. The Cohen's d is computed as follows:

$$d = \frac{\mu_1 - \mu_2}{\sqrt{\frac{(n_1-1)*\sigma_1+(n_2-1)*\sigma_2}{n_1+n_2-2}}}$$

where μ_1, μ_2 and σ_1, σ_2 represent the mean and the standard deviations of the phenotypes in the two genetic states of the genotype, while n_1 and n_2 represent the sample sizes of the two genetic states of the genotype.

Both runGenphenRf and runGenphenSvm estimate the three metrics CA, κ and d with their corresponding highest density intervals, and with that provide an alternative approach for GWAS.

2.2.2 runGenphenBayes

The procedure runGenphenBayes employes Bayesian inference to quantify the association between genotypes and phenotypes. It performs Bayesian

T-test (Kruschke, 2014), using hierarchical models. The results of this procedure are the most probable estimates (posterior estimates) for the parameters μ_1 and μ_2 which represent the mean phenotypes in each group of a given genotype, as well as the standard deviations σ_1 and σ_2. While the classical T-test has to adhere to two assumptions, namely: 1) the phenotypes should be normally distributed in each group and 2) the variances in the two groups should be equal; the Bayesian hierarchical models have been designed such that both of these assumptions are relaxed. Using the posterior estimates for μ_1 and μ_2 for a given genotype-phenotype combination, one can estimate the contrast $\mu_1 - \mu_2$ and the so-called highest density interval (HDI) to evaluate whether there exists a significant phenotypic effect between the two groups which excludes the null-effect.

Using runGenphenBayes and analysing genotype-phenotype association based on the Bayesian effect size $\mu_1 - \mu_2$ and the corresponding HDI has multiple advantages:

- the effect size is easy to interpret
- it helps avoid the multiple hypothesis problem
- the HDI of the effect size provides a convenient filtering technique with which the associations can be classified as either reliable (the HDI excludes the null-effect) or unreliable (the HDI spans the null-effect)
- the Bayesian hierarchical models relax the assumptions of the classical T-test, and employ prior which model more accurately the observed data

The only disadvantage of applying this method genome-wide is its computational complexity.

2.3 Case studies

2.3.1 SNP-phenotype association with `genphen`

This example is intended to guide the user through the previously described `genphen` methodology. In particular, we present a situation in which the association is to be computed between SNPs and phenotypes. We use a simple example shown in Fig 1a to explain the `genphen` procedures runGenphenRf/runGenphenSvm and runGenphenBayes.

- Input:
  - genotype: a SNP column vector of 14 elements, where the two alleles A and T are contained with 5 and 9 elements, respectively. The 14 elements of this vector represent 14 mouse species.

  - phenotype: a vector of 14 elements, where each element represents the measured immune response of a specific mouse.
  - Hint: we can inspect the given genotype-phenotype pair using the procedure `plotSpecificGenotype` whose results are shown in Fig. 1b.

- Goal: can we quantify the association between the genotype and the phenotype vectors using `genphen`?
- Methods: we use the procedures runGenphenRf/runGenphenSvm and runGenphenBayes to compute the association.

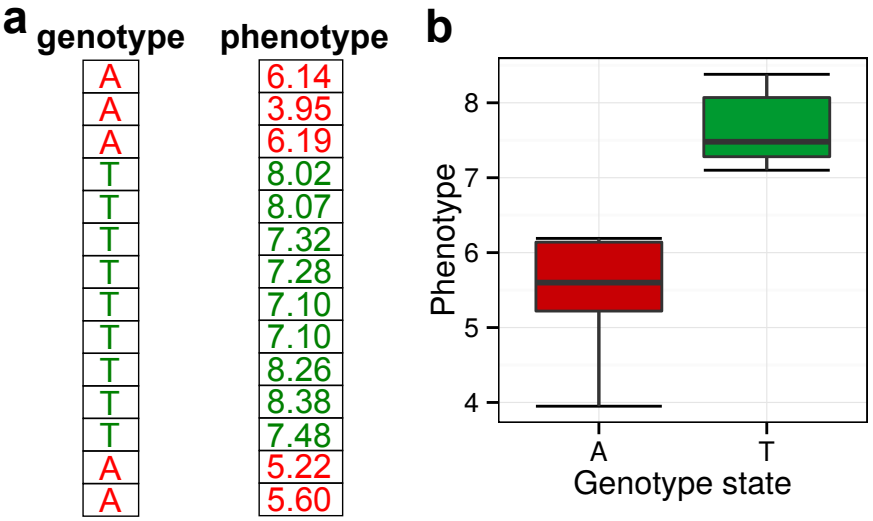

Figure 1: **a**, genotype-phenotype data pair. The genotype is a SNP with two genetic states (alleles A and T), found in 14 individuals. **b**, boxplot showing the phenotypic distribution of the genotype as a function of the two alleles.

runGenphenSvm and runGenphenRf

```
> library(genphen)
> genotype <- c("A", "A", "A", "T", "T", "T", "T", "T", "T",
+               "T", "T", "T", "A", "A")
> phenotype <- c(6.14, 3.95, 6.19, 8.02, 8.07, 7.32, 7.28,
+                7.10, 7.10, 8.26, 8.38, 7.48, 5.22, 5.60)

> result <- runGenphenSvm(genotype = genotype,
+                         phenotype = phenotype,
+                         cv.fold = 0.66,
```

```
+                                   cv.steps = 1000,
+                                   hdi.level = 0.95)
```

site	g.1	g.2	count.1	count.2	d	CA	κ	t.test.pvalue
1	A	T	5	9	-3.339	0.960	0.918	0.003

Table 2: Results of the runGenphenSvm procedure (HDIs have been left out)

In Fig. 1b we notice that the two states of the genotype are associated with divergent distributions of the phenotype. Hence, we can expect a significant association between the SNP and the phenotype. This is confirmed by the results of the procedure runGenphenSvm, which yield CA close to 1 ($CA = 0.960$), high Cohen's κ ($\kappa = 0.918$) and a substantial Cohen's effect size ($d = -3.339$). Similar results can be obtained with the alternative statistical method (RF).

runGenphenBayes

```
> result <- runGenphenBayes(genotype = genotype,
+                           phenotype = phenotype,
+                           chain.nr = 2,
+                           mcmc.iter = 1000,
+                           model = "tdist",
+                           hdi.levels = 0.95)
```

site	g.1	g.2	count.1	count.2	mu.1	mu.2	effect	effect.0.95.HDI
1	A	T	5	9	5.460	7.662	-2.202	(-3.771,-0.757)

Table 3: Results of the runGenphenBayes procedure

The results of runGenphenBayes confirm the previous claims about the association between the SNP and the phenotype. The state A is associatied with a mean phenotype of 5.460, while T is associated with 7.662. The Bayesian effect size therefore equals to -2.202 with 95% HDI of (-3.771, -0.757), which excludes the null-effect. The HDI criterion is an intuitive quality measure which can be used to filter out poor association. In contrast to this, the filtering is not as intuitive based on CA and κ in the case of the statistical learning procedures.

2.3.2 SAAP-phenotype association with `genphen`

This example is intended to guide the user through the previously described `genphen` methodology, this time when the association is to be computed between a SAAP and a phenotype.

- Input:
  - genotype: a SAAP column vector of 120 elements, where the four amino acid states are present H, Q, N and K with the following counts 62, 55, 2 and 1, respectively. This genotype vector is a single site taken from the protein sequence of 120 organisms, therefore each amino acid state corresponds to a specific organism.
  - phenotype: a vector of 120 numerical elements (artificially generated).
  - Hint: we can first inspect the genotype-phenotype pair using the procedure `plotSpecificGenotype` whose results are shown in Fig 2.
- Goal: can we quantify the association between the genotype and the phenotype vectors using `genphen`?
- Method: we can use the `genphen` method `runGenphen` to compute the association. This procedure will first need to decompose the genotype-phenotype pair into the 6 possible amino acid substitution pairs, namely (H, Q), (H, K), (H, N), (Q, K), (Q, N) and (K, N), and then compute the association between substitution pair and the phenotype just as it was presented in the previous example.

runGenphenSvm and runGenphenRf

```
> library(genphen)
> data(genotype.saap)
> data(phenotype.saap)
> genotype <- genotype.saap[, 82]
> phenotype <- phenotype.saap

> result <- runGenphenSvm(genotype = genotype,
+                         phenotype = phenotype,
+                         cv.fold = 0.66,
+                         cv.steps = 1000,
+                         hdi.level = 0.95)
```

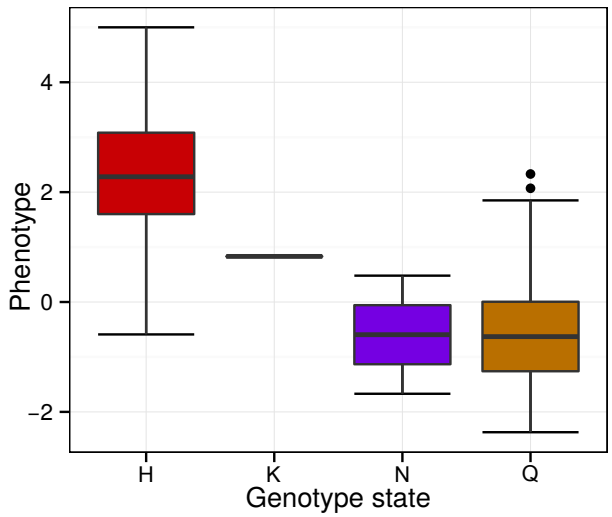

Figure 2: Boxplot showing the phenotypic distribution as a function of four amino acids (H, Q, N, K) at a given SAAP of 120 amino acids in total.

site	g.1	g.2	count.1	count.2	d	CA	κ	t.test.pvalue
1	H	Q	62	55	2.381	0.879	0.755	0
1	H	K	62	1	NA	NA	NA	NA
1	H	N	62	2	2.339	0.974	-0.0001	0.221
1	Q	K	55	1	NA	NA	NA	NA
1	Q	N	55	2	-0.102	0.973	0	0.934
1	K	N	1	2	NA	NA	NA	NA

Table 4: Results of the runGenphenSvm procedure (HDIs have been left out)

For each SAAP (amino acid substitution), runGenphenSvm and runGenphenRf estimates the previously defined metrics w.r.t. the phenotype. One can then use the CA, Cohen's κ and Cohen's effect size d, to find the most significant associations. The SAAP involving the amino acid (H, Q) stands out in this example not only due to its high CA, κ and d, but also because these estimates are based on substantial data of 62 + 55 data points. If only a single data point is available for a given genetic state of a SAAP (e.g. amino acid state K), then NA (not available) estimates are returned.

runGenphenBayes

```
> result <- runGenphenBayes(genotype = genotype,
+                           phenotype = phenotype,
```

```
+                                chain.nr = 2,
+                                mcmc.iter = 1000,
+                                model = "tdist",
+                                hdi.levels = 0.95)
```

site	g.1	g.2	count.1	count.2	mu.1	mu.2	effect	effect.0.95.HDI
1	H	Q	62	55	2.285	-0.514	2.798	(2.355, 3.219)
1	H	K	62	1	NA	NA	NA	(NA, NA)
1	H	N	62	2	2.286	-0.575	2.861	(-6.145, 12.663)
1	Q	K	55	1	NA	NA	NA	(NA, NA)
1	Q	N	55	2	-0.513	-0.604	0.091	(-8.121, 8.140)
1	K	N	1	2	NA	NA	NA	(NA, NA)

Table 5: Results of the runGenphenBayes procedure

The results of runGenphenBayes once again confirm the previous claims about the association between the different SAAPs and the phenotype. The amino acid state H is associatied with a mean phenotype of 2.285, while Q is associated with -0.514. The Bayesian effect size therefore equals to 2.798 with 95% HDI of (2.355, 3.219), which excludes the null-effect. None of the other associations satisfies this criterion.

2.4 Visualization

The tool `genphen` implements visualization procedures which help the user evaluate its results. Here we introduce the following four procedures:

- plotSpecificGenotype
- plotGenphenRfSvm
- plotGenphenBayes
- plotManhattan

plotSpecificGenotype The results of the procedure `plotGenphenResults` are shown in Fig 3. With it, one can plot the phenotype as a function of a specific genotype, in which the user might be interested (e.g. specific SNP).

plotGenphenRfSvm The results generated either by runGenphenRf and runGenphenSvm can be inspected via this procedure. With it one visualizes the quantified genotype-phenotype associations with respect to their CA and d measures.

plotGenphenBayes The results generated by plotGenphenBayes can be inspected via this procedure. With it one visualizes the quantified genotype-phenotype associations with respect to their Bayesian effect sizes $\mu_1 - \mu_2$ and the corresponding HDIs.

plotManhattan The -log10 transformed P values estimated via the two-sample T-test for each genotype-phenotype association can be inspected via this procedure using via a Manhattan plot.

All four plotting procedures are applied in the following section 3.

3 Complete worked example

3.1 Workflow 1

1. Loading data

```
> library(genphen)
> data(genotype.saap)
> # One can also use a multiple sequence alignment as a genotype input (in the
> # form of either a DNAMultipleAlignment or AAMultipleAlignment objects of the
> # Biostring package).
> data(phenotype.saap)
```

2. Running genphen algorithm using random forests (RF) and linear support vector machines (LSVM):

```
> # if DNAMultipleAlignment is loaded you cannot subset
> # with genotype.snp[, 1:10]
> genphen.rf <- runGenphenRf(genotype = genotype.saap[, 1:10],
+                            phenotype = phenotype.saap,
+                            cv.fold = 0.66,
+                            cv.steps = 100,
+                            hdi.level = 0.99,
+                            ntree = 1000)
> genphen.svm <- runGenphenSvm(genotype = genotype.saap[, 1:10],
+                              phenotype = phenotype.saap,
+                              cv.fold = 0.66,
+                              cv.steps = 100,
+                              hdi.level = 0.99)
```

3. Filtering out association which could not be quantified by `genphen`, due to lack of data:

```
> genphen.rf <- genphen.rf[complete.cases(genphen.rf), ]
> genphen.svm <- genphen.svm[complete.cases(genphen.svm), ]
```

4. Plotting results

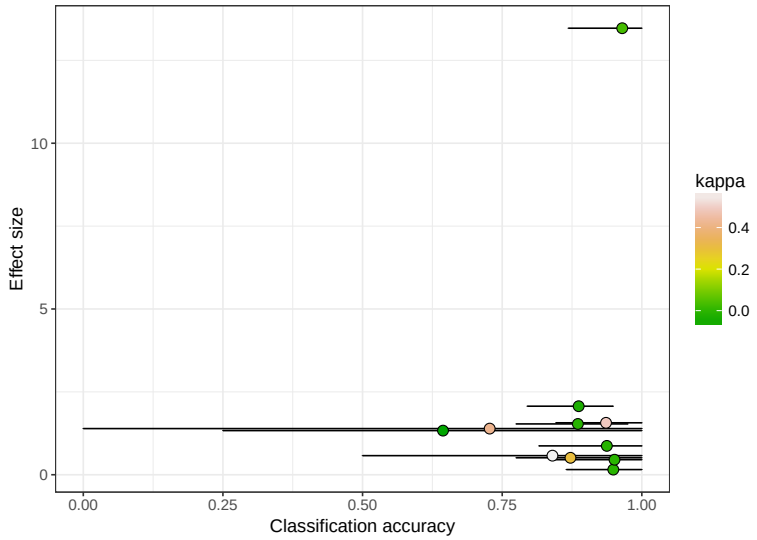

Figure 3: Effect site - classification accuracy plot

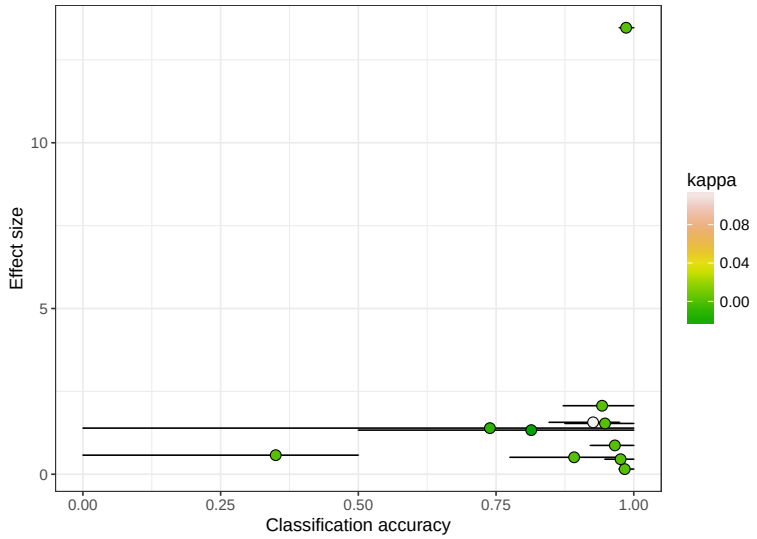

Figure 4: Effect site - classification accuracy plot

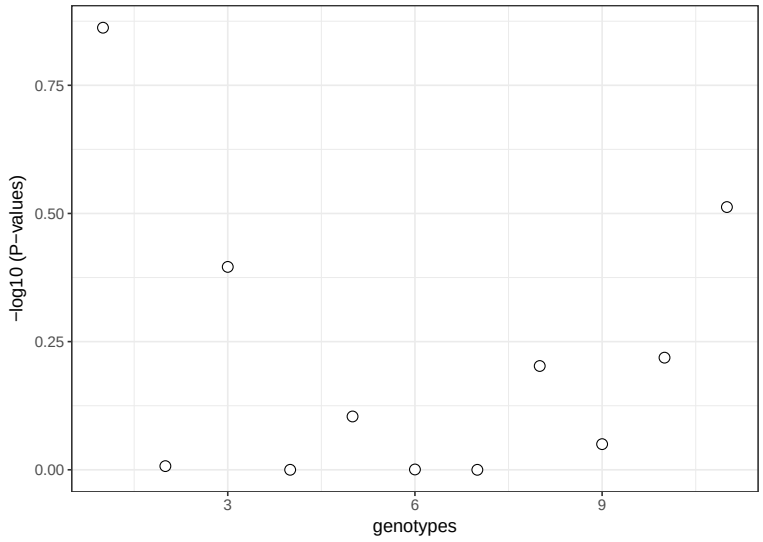

Figure 5: Manhattan plot

5. Running the genphen algorithm using Bayesian inference:

```
> # if DNAMultipleAlignment is loaded you cannot subset
> # with genotype.snp[, 1:10]
> genphen.bayes <- runGenphenBayes(genotype = genotype.saap[, 1:10],
+                                  phenotype = phenotype.saap,
+                                  chain.nr = 2,
+                                  mcmc.iter = 1000,
+                                  model = "tdist",
+                                  hdi.levels = c(0.90, 0.95, 0.99))
```

6. Filtering out association which could not be quantified by `genphen`, due to lack of data:

```
> genphen.bayes <- genphen.bayes[complete.cases(genphen.bayes), ]
```

7. Plotting results

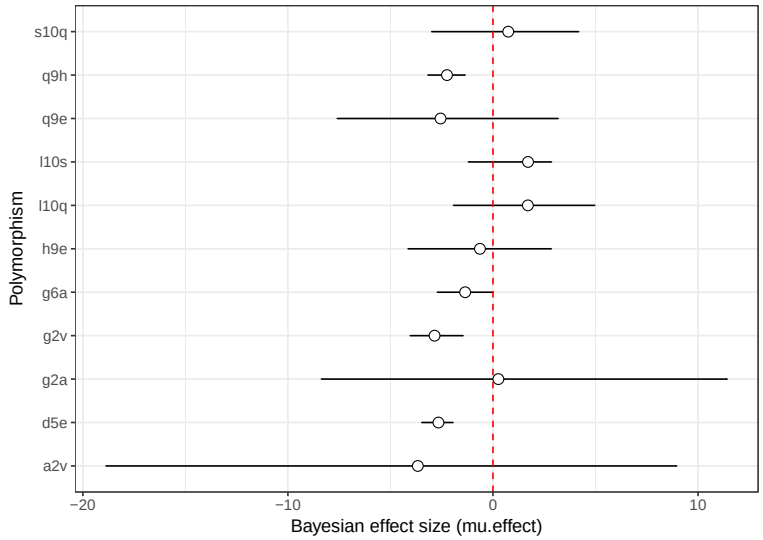

Figure 6: Effect site - Bayesian effect size plot

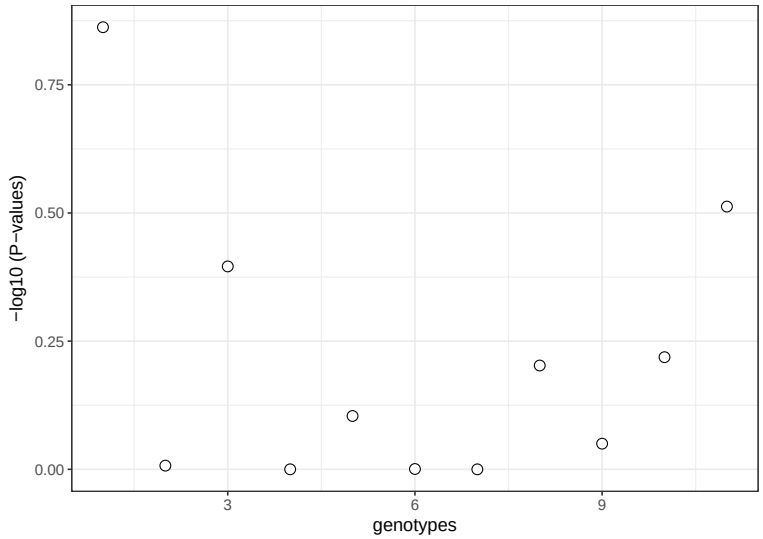

Figure 7: Manhattan plot

8. Visual inspection of the association between a specific genotypes and the phenotypes:

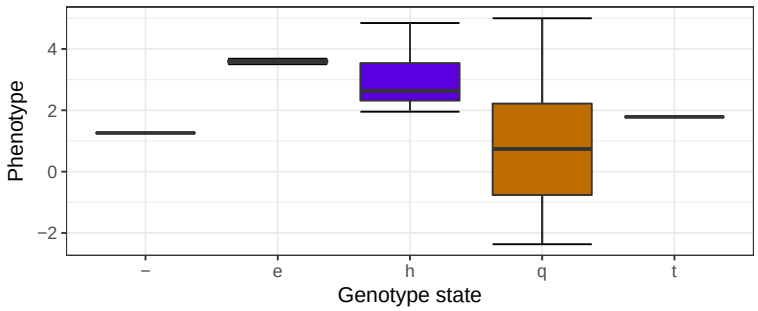

Figure 8: Specific genotype-phenotype plot