Tutorial on using genphen

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

May 3, 2016

Contents

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

1 genphen computes genotype-phenotype associations

Quantifying the association between genotypes and phenotypes is an important task in genetic association studies. In this type of analyses, we 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 poymorphic genotypes were found.

In general, the goal of genetic association studies is to identify the specific genotypes which are causally linked to the given phenotype. genphen implements procedures which allow us to quantify the genotype-phenotype association for both SNPs and SAAPs. Its methodology combines statistical learning techniques such as random forests and support vector machines, with wffcts sizes such as the Cohen's d effect size, to identify the genotypes which have the higher explanatory power of the phenotype. The statistical methods outperform the simple (linear) methods used in genetic association,

and together with the effect size statistics allow for intuitive inter-study comparison of genotype-phenotype association scores. Finally, `genphen` implements a set of visualization procedures, which allow the user to inspect the results of the main association analyses and pinpoint the relevant genotypes.

2 Conducting association studies 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). We can use `genphen` to quantify the association between each of the SNPs and the phenotype.

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 contain amino acid polymorphisms). The phenotype can 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 sites of genetic polymorphism, and the N rows represent different individuals or sequences for which we have the some measured phenotypes. On the other hand, we can think of the phenotype as a numerical vector of length N.

2.2 Method

Once we have the genotype and the phenotype data, we can compute the association between each specific genotype vector (either SNP or SAAP) and the phenotype. `genphen` computes the following two metrics for each specific genotype with respect to the phenotype:

- **Cohen's effect size** (d): Each specific genotype is composed of different genetic states (nucleotide alleles in SNPs or amino acids in SAAPs). We want to measure the phenotypic effect when the different genetic states are replaced with each other. If the genotype data is composed

of SNPs, for each SNP we only need to compute one effect size, i.e. between the two alleles x and y of that SNP, as follows:

$$\Delta p = \hat{p}_x - \hat{p}_y$$

$$pooled\ \sigma = \sqrt{\frac{(n_x - 1) * \sigma_x^2 + (n_y - 1) * \sigma_y}{n_x + n_y - 2}}$$

$$d = \frac{\Delta p}{pooled\ \sigma}$$

First we compute the delta phenotype (Δp) by subtracting the mean phenotypes observed at each allele ($\hat{p}_x$ and $\hat{p}_y$). In the next equation, the pooled standard deviation *pooled* σ is computed, whereby *pooled* σ is used to normalize the Δp in the final estimation of the effect size. The terms n_x and n_y represent the counts of one the first and second allele in the genotype vector, respectivelly. σ_x and σ_y are the standard deviations of the phenotype observed at the two alleles.

The effect size computation for SAAPs is very similar to the one we presented for SNPs, with a slight technical difference. Compared to SNPs, SAAPs may contain more than two genetic states, i.e. more than two amino acid types at a given SAAP. If this is the case, we first need to decompose the SAAP into pairs of amino acid substitutions, and then compute the effect size for each substitution pair using the same equation as before. This means that if a SAAP is made up of 3 amino acids A, B and C, we first create subsets of both the specific SAAP data and the corresponding phenotype based on the following amino acid substitution pairs (A, B), (A, C) and (B, C), and then compute the effect sizes for each pair just as it was done before.

High d estimations indicate that there is a large difference in the measured phenotype between the two genetic states of the specific genotype. For instance the first genotype might be associated with low phenotypes, while the second genotype with high phenotype. 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 as implemented by the package `effsize`.

- **Classification accuracy**(ACC): this metric is used to quantify the strength of the association between each specific genotype and the phe-

notype. `genphen` uses statistical learning techniques (random forests or linear support vector machines) to build a classification model between the phenotype (a numerical predictor) and a specific genotype (a categorical response). The more accurate is the classification model between the two vectors, the stronger is their mutual association. To obtain a robust classification accuracy we applied bootstrapping, whereby the algorithm first selects a subset of the genotype-phenotype data at random with which it trains a classifier, and subsequently uses the remaining data to test the model. After many bootstrapping iterations (typically higher than 100 as defined by the user), the individually computed classification accuracies of the built models are aggregate into a final mean classification accuracy.

The following procedure describes one iteration step of the computation of the ACC between a phenotype vector and a SNP vector. First a classification model is built between the phenotype as an independent variable and the SNP as a dependent variable composed of two allele types. The model is tested resulting in a confusion matrix as shown in Table 1.

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

Table 1: Confusion matrix resulting from a classification analysis

From the confusion matrix we can compute the observed classification accuracy of the iteration ACC_i:

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

The final classification accuracy between the specific genotype and the phenotype is the mean observed classification accuracy computed from many iterations as:

$$ACC = \frac{1}{boots} \sum_{i=1}^{boots} ACC_i$$

In addition to the classification accuracy estimate one can also compute the 95% confidence interval. This can be achieved via the bootstrap percentile method, using the observed classification estimates.

Therefore it is important that the number of bootstrap iterations selected by the user is high.

A specific genotype which is attributed with a high ACC and a narrow 95% CI implies that the genetic states of that specific genotype can accurately be predicted from the phenotypes, and their mutual association strength is high.

- **Cohen's κ statistics**: often we are interested in comparing the observed ACC (the accuracy computed by the statistical learning tool) and the expected accuracy ($ACC_{expected}$), i.e. the accuracy expected by chance. Cohen's κ statistics is used to make such as comparison. The formula used to compute the κ statistics is completely derived from the confusion matrix shown before in Table 1 as follows:

$$\kappa = \frac{ACC - ACC_{expected}}{1 - ACC_{expected}}$$

$$ACC_{expected} = \frac{(a+b)*(a+c)}{a+b+c+d} + \frac{(c+d)*(b+d)}{a+b+c+d}$$

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

The κ statistics is only a supporting metric that can be used to validate ACC. Cohen defines the following meaningful κ intervals: [κ<0]: "no agreement", [0.0-0.2]: "slignt 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". These intervals can be used in a post-processing scenario to help the user rank the association scores. To estimate a robust statistics we compute a mean κ statistics aggregated from the individual κ estimates computed in each iteration:

$$\kappa = \frac{1}{boots} \sum_{i=1}^{boots} \kappa_i$$

Ideally, we would like to identify the specific genotypes which have high d, ACC and κ metrics.

2.3 Example 1: SNP-phenotype association with `runGenphenSnp`

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 describe the two metrics and their estimation using the procedure `runGenphenSnp`.

- Input:
  - genotype: a SNP column vector of 14 elements, where the two alleles A and T are contained with 5 and 9 elements, respectivelly. 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`?
- Method: we can use the `genphen` procedure `runGenphenSnp` to compute the association.
  - $d = -3.338$. Change from allele A with a second allele T is linked with a substantial effect size given that $|d| > 0.8$.

$$\hat{p}_A = 5.42,\ \sigma_A = 0.914$$

$$\hat{p}_T = 7.66,\ \sigma_T = 0.511$$

$$\Delta p = 5.42 - 7.66 = -2.247$$

$$pooled\ \sigma = \sqrt{\frac{(5-1) * 0.914^2 + (9-1) * 0.511^2}{5+9-2}} = 0.673$$

$$d = \frac{-2.247}{0.673} = -3.338$$

  - $ACC = 0.97$ (97%). This is an estimate obtained after performing 100 classification analyses, whereby in each analysis 66% of the data was used for training a classifier and the remaining 34%

for testing. The mean classification accuracy obtained after the 100 analyses was taken as the final classification accuracy. 0.959 or 97% classification accuracy means that on average 97% of the alleles of the given SNP are correctly classified using the phenotype as a predictor.

- $\kappa = 0.961$. This is the mean κ estimate aggregated from the 100 classification analyses. Based on the κ intervals defined previously this κ allows us to conclude that there is a concordance between the observed classification accuracy and the expected classification accuracy.
- Output: the actual result obtained when using the `runGenphenSnp` in this example is given in Table 2. In addition to the three metrics presented before, several supplementary metrics have also been estimated such as he corresponding confidence intervals to the effect size and the 95% classification accuracy and other general statistics concerning the specific genotype-phenotype pair.

$allele_1$	$allele_2$	n_A	n_T	d	95% CI	ACC	95% CI	κ	anova
A	T	5	9	-3.338	(-5.41, -1.27)	0.97	(0.60, 1)	0.961	0.0001

Table 2: Results of the `runGenphenSnp` procedure using the data provided in Example 1.

2.4 Example 2: SAAP-phenotype association with `runGenphenSaap`

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. We describe the two metrics and their computation wth a help of a simple example. The procedure used to compute the association in this example is `runGenphenSaap`.

- 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.

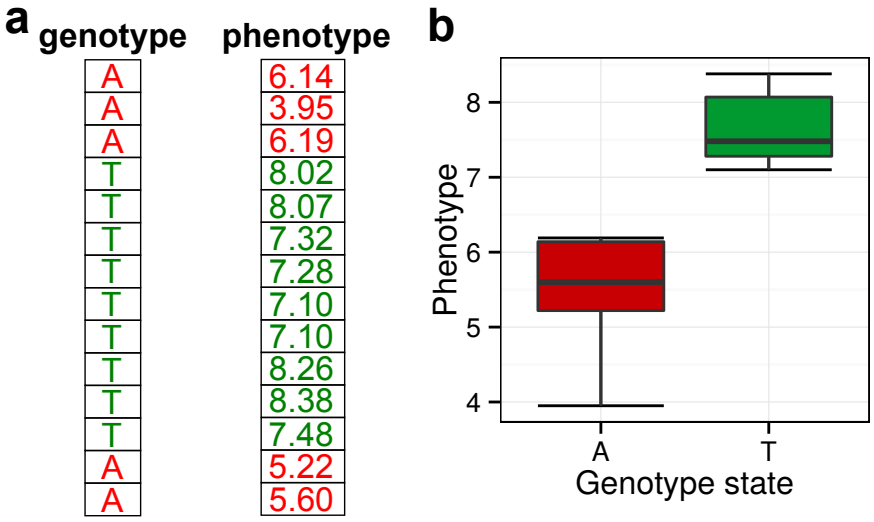

Figure 1: **a**, A genotype-phenotype vector pair, whereby the genotype is a SNP with two alleles A and T, found in 14 strains. **b**, a boxplot showing the phenotypic distribution of the specific genotype as a function of its two alleles. This boxplot can be generated using the procedure `plotSpecificGenotype`.

  - 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 `runGenphenSaap` 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.
- Output: the results of the analysis are summarized in Table 3, where each row gives us the association results obtained for a specific amino acid substitution pair. One can notice that certain amino acid states are only represented by few amino acid instances, e.g. K is represented by a single instance and N is represented by 2 instances. On the other hand amino acid states such as H and Q are abundantly represented by 62 and 55 amino acid states, respectively. All these general statistics

of the specific genotype are provided in the columns aa_1, aa_2, n_{aa_1} and n_{aa_2}. It is most usual to see that whenever an amino acid state is represented by very few instances, the corresponding substitution pairs might be characterized with NAs and very wide confidence intervals. The rows in which a NA is found are usually to be discarded or treated with extreme care as they are influenced by a lack of data.

In this particular example only the substitution pair (H, Q) seems to contain enough data. It is noticeable that the substitution (H, Q) has the largest d and a tight confidence interval compared. The ACC has been estimated to 0.880 (88%). This means that the amino acid states H and Q have been classified with a mean accuracy of 88% from the phenotype. The 95% confidence interval for this estimate confirm this observation as it is quite narrow. The κ statistics allows us to conclude that there is a concordance between the observed and expected accuracy as well. The high association scores presented before are also confirmed by the remarkable low p-value score obtained from an ANOVA test.

aa_1	aa_2	n_{aa_1}	n_{aa_2}	d	95% CI	ACC	95% CI	κ	anova
h	q	62	55	2.381	(1.895, 2.866)	0.881	(0.80, 0.95)	0.847	10^{-24}
h	k	62	1	NA	(NA, NA)	1	(1, 1)	NA	0.243
h	n	62	2	2.339	(0.819, 3.860)	0.975	(0.955, 1)	0.955	0.002
q	k	55	1	NA	(NA, NA)	1	(1, 1)	NA	0.237
q	n	55	2	-0.103	(-1.571, 1.366)	0.974	(0.947, 1)	NA	0.887
k	n	1	2	NA	(NA, NA)	0.478	(0, 1)	NA	0.584

Table 3: Results of the `runGenphenSaap` procedure using the data provided in Example 2. The table presents some of the main metrics as well as some general statistics such as the number of amino acid instances for each amino acid state.

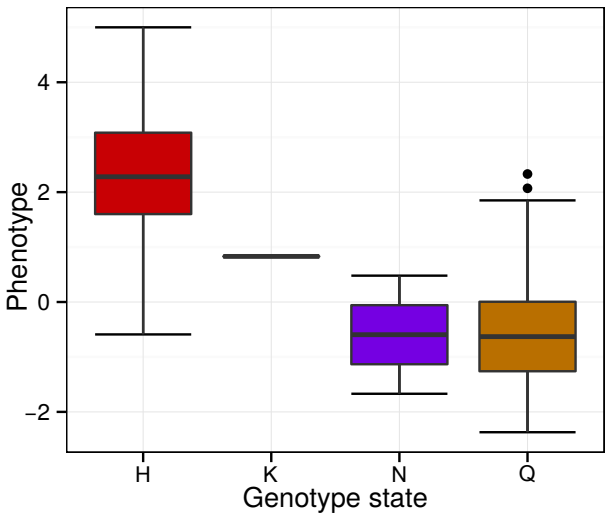

Figure 2: Boxplot showing the phenotypic distribution as a function of four amino acids (H, Q, N, K) at a given SAAP of 120 amino acids in total. This boxplot can be generated using the procedure `plotSpecificGenotype`.

2.5 Visualization

We have previously introduced `plotSpecificGenotype`, which is one of the plotting procedures of `genphen` used to inspect a specific genotype-phenotype pair. This tool also provides plotting procedures to visualize the results of a genetic association study between many SNPs or SAAPs and a phenotype. The results of the procedure `plotGenphenResults` are shown in Fig 3, where we show the association between a genotype data composed of 100 SNPs and a phenotype. Each SNP is represented by a point plotted with respect to its two metrics effect size and classification accuracy. The color of the points is proportional to the classification accuracy as well. If one is interested in finding the SNPs which are related to the phenotype, the SNPs found in the upper right corner of the plots should be considered first as they are characterized with both a large phenotypic effect and a high strength of association with the phenotype.

`genphen` also provides an auxiliary plotting procedure with which the user can generate the so-called Manhattan plots. They are generated using the p-values obtained from the ANOVA tests which are done in parallel to the main algorithm. This plot is not the main contribution of this package as it is not based on any of its primary metrics (classification accuracy and effect size). Nevertheless, we hope that with this procedure we can assist those users which are inclined on using Manhattan plots and p-values. The procedure `plotManhattan` plots the Manhattan plots, whereby the raw p-values obtained by `genphen` are corrected with FDR prior to the plotting

(see Fig 4). This plot corresponds to the one presented in Fig 3, whereby the SNPs with the lowest p-values are exactly the ones which are located in the upper-right corner of the previous plot. The main advantage of using classification accuracies and effect sizes instead of p-values is the intuitivness of the results, whereas p-values are often difficult to interpret as well as to figure out what is the appropriate p-value thershold beyond which the results are to be discarded.

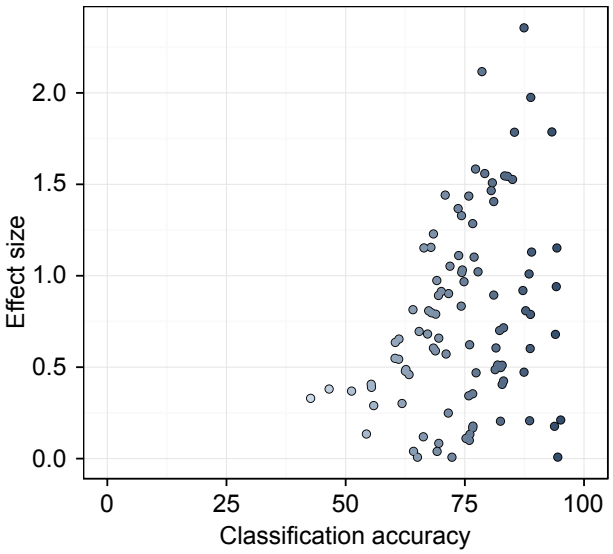

Figure 3: Visualization of the results of a genotype-phenotype association study with the help of `plotGenphenResults`. 100 points are shown, each representing a SNP plotted with respect to its classification accuracy and effect size.

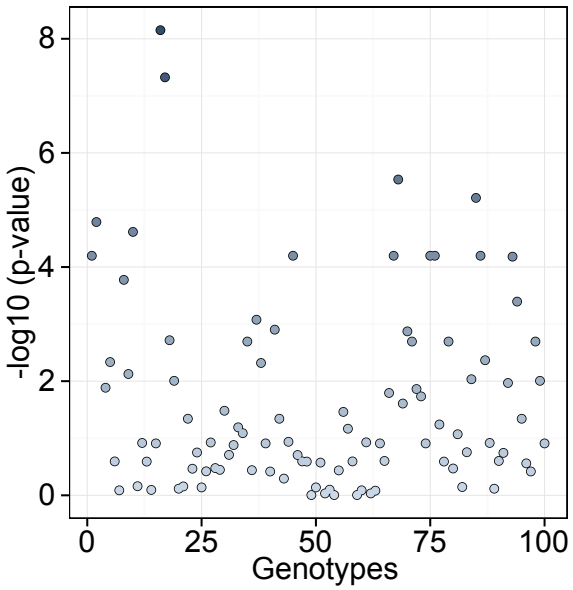

Figure 4: Visualization of the results of a genotype-phenotype association study with the help of `plotManhattan`. 100 points are shown, each representing a SNP plotted with respect its p-value obtained via an ANOVA test.

3 Two worked examples

3.1 Workflow 1

1. Loading data

```
> library(genphen)
> data(genotype.snp)
> #alternatively you can take as genotype an object of class
> # DNAMultipleAlignment with data(genotype.snp.msa)
> data(phenotype.snp)
```

2. Inspect phenotype as function of genotype

3. Running the genphen algorithm

```
> # if DNAMultipleAlignment is loaded you cannot subset
> # with genotype.snp[, 1:5]
> genphen.results <- runGenphenSnp(genotype = genotype.snp[, 1:5],
+                                  phenotype = phenotype.snp,
+                                  technique = "svm",
+                                  fold.cv = 0.66,
+                                  boots = 100)
```

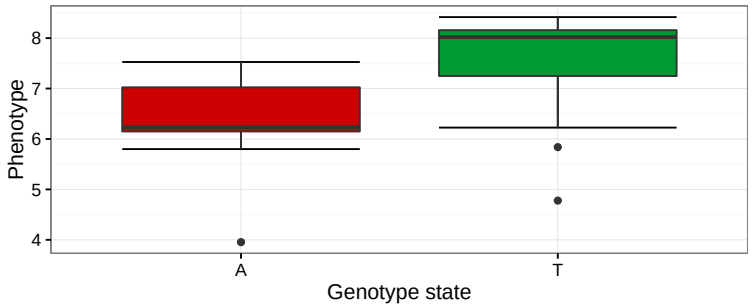

Figure 5: Manhatten plot

4. Filtering good results based on Cohen's κ

```
> genphen.results <- genphen.results[complete.cases(genphen.results), ]
> genphen.results <- genphen.results[genphen.results$kappa > 0.4, ]
> genphen.results[, c(1:3, 6, 9, 14, 18)]
```

```
  site allele1 allele2 effect.size        ca     kappa  anova.score
1    1       A       T  -1.5082608 0.8147059 0.7931766 4.657960e-06
2    2       A       C  -1.5835282 0.7858824 0.7067173 8.219192e-07
5    5       A       G  -0.9674114 0.7523529 0.6873645 1.168137e-03
```

5. Plotting results

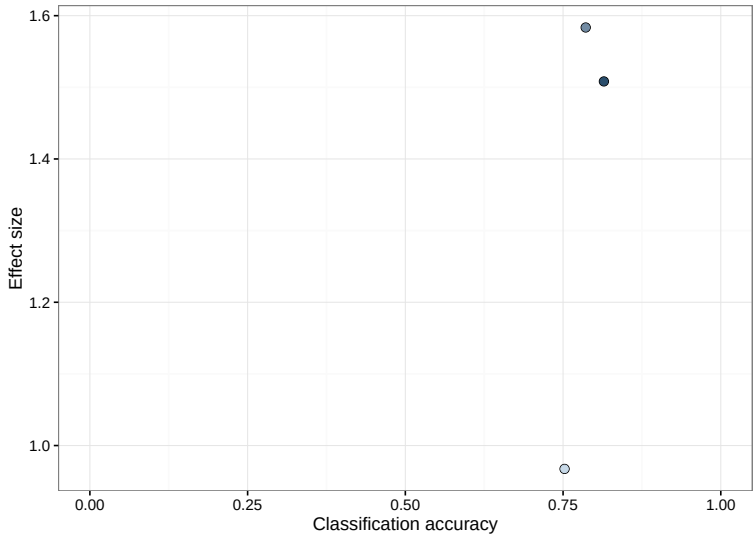

Figure 6: Effect site - classification accuracy plot

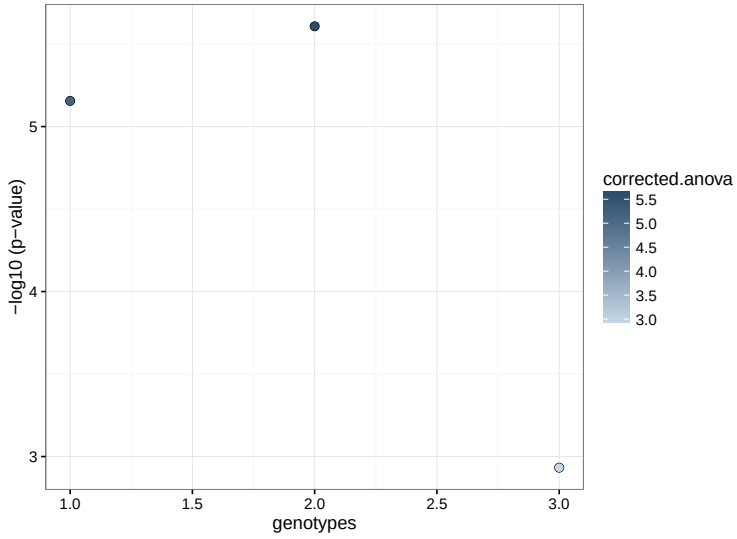

Figure 7: Manhatten plot

3.2 Workflow 2

1. Loading data

```
> library(genphen)
> data(genotype.saap)
> # alternatively you can take as genotype an object of class
> # AAMultipleAlignment with data(genotype.saap.msa)
> data(phenotype.saap)
```

2. Inspect phenotype as function of genotype

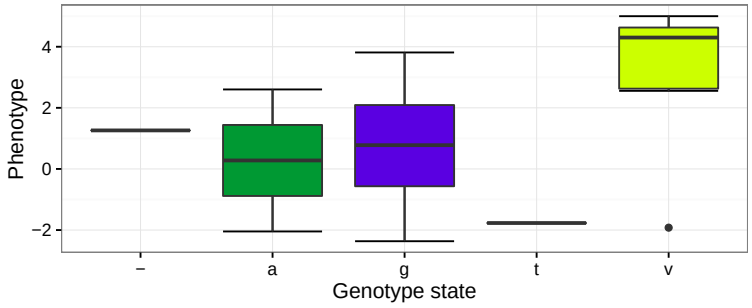

Figure 8: Manhatten plot

3. Running the genphen algorithm

```
> # if AAMultipleAlignment is loaded you cannot subset
> # with genotype.saap[, 1:5]
> genphen.results <- runGenphenSaap(genotype = genotype.saap[, 1:5],
+                                   phenotype = phenotype.saap,
+                                   technique = "svm",
+                                   fold.cv = 0.66,
+                                   boots = 100)
```

4. Filtering NAs

```
> genphen.results <- genphen.results[complete.cases(genphen.results), ]
> genphen.results[, c(1:3, 6, 9, 14, 18)]
```

```
   site aa1 aa2 effect.size        ca       kappa  anova.score
2     2   g   a  -0.3134329 0.9867936  0.00000000 6.614199e-01
3     2   g   v  -1.5666034 0.9294872  0.13120863 1.558701e-05
6     2   a   v  -1.3310687 0.7861111 -0.03864734 1.228263e-01
16    5   d   e  -1.5329228 0.9525000  0.00000000 3.811304e-04
```

5. Plotting results

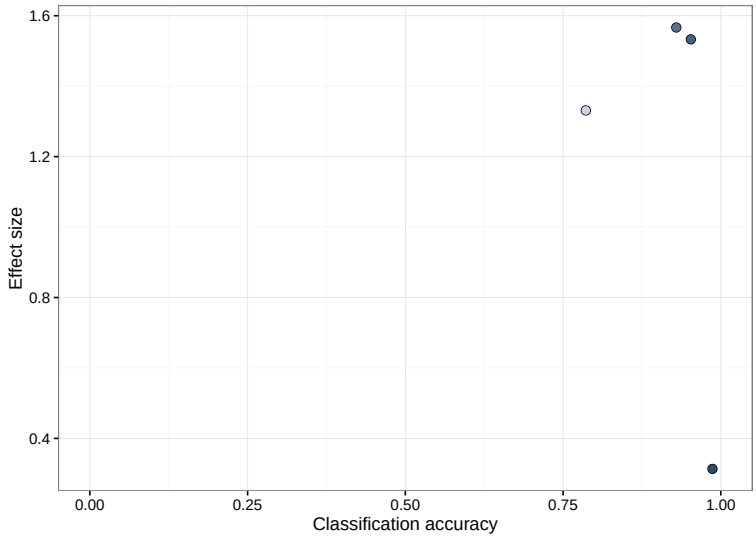

Figure 9: Effect site - classification accuracy plot

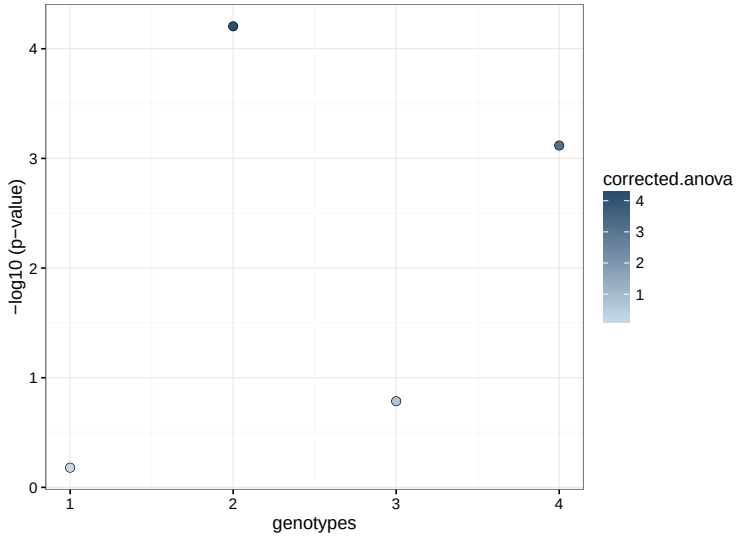

Figure 10: Manhatten plot