This vignette explains how to use the R package `semisup`. Use the function `mixtura` for model fitting, and the function `scrutor` for hypothesis testing.

## 1 Initialisation

Start with installing `semisup` from Bioconductor¹:

```r
if (!requireNamespace("BiocManager", quietly=TRUE))
  install.packages("BiocManager")
BiocManager::install("semisup")
```

Then load and attach the package:

```r
library(semisup)
```

If you want to reproduce the examples, you should attach the toy database:

```r
attach(toydata)
```

The following commands access the reference manual:

```r
help(semisup)
?semisup
```

¹ `devtools` and GitHub: `devtools::install_github("rauschenberger/semisup")`
2 Scope

Data is available for \( n \) samples. Let \( y = (y_1, \ldots, y_n)^T \) represent the observations, \( x = (x_1, \ldots, x_n)^T \) the groups, and \( z = (z_1, \ldots, z_n)^T \) the classes. We assume all observations from the labelled group are in class A, and those from the unlabelled group are in class A or in class B.

\[
\begin{array}{c|c c c c c}
1 & \ldots & s & s+1 & \ldots & n = s + u \\
\hline
y & y_1 & \ldots & y_s & y_{s+1} & \ldots & y_{s+u} \\
x & 0 & \ldots & 0 & 1 & \ldots & 1 \\
z & A & \ldots & A & A/B & \ldots & A/B \\
\end{array}
\]

Table 1: Observations \( y \), groups \( x \), and classes \( z \). Here, the first \( s \) observations are labelled (class A), and the last \( u \) observations are unlabelled (class A or B).

We assume all observations come from the same probability distribution, but with different parameters for the two classes:

\[
Y_i | (Z_i = A) \sim F(\cdot, \theta_a), \\
Y_i | (Z_i = B) \sim F(\cdot, \theta_b).
\]

The mixing proportion \( \tau \) is the probability that a random unlabelled observations is in class B. It is of interest to test whether \( \tau \) is significantly larger than zero.

\[
\tau = P[Z_i = B | X_i = 1], \\
H_0 : \tau = 0, \\
H_1 : \tau > 0.
\]

The function \textit{mixtura} estimates the unknown parameters \((\theta_a, \theta_b, \tau)\) and predicts the missing class labels in \( z = (z_1, \ldots, z_n)^T \). The function \textit{scrutor} tests homogeneity \((\tau = 0)\) against heterogeneity \((\tau > 0)\).
3 Model fitting

Observing two groups of observations, we assume the *labelled* observations are in class A, and the *unlabelled* observations are in class A or in class B.

![Graph showing labelled and unlabelled observations]

The function `mixtura` estimates the unknown parameters and predicts the missing class labels:

```r
fit <- mixtura(y,z)
```

Here, 21% of the *unlabelled* observations are assigned to class B, and all other observations are assigned to class A:

```r
class <- round(fit$posterior)
```

![Graph showing predicted class assignments]

These are the parameter estimates:

```r
fit$estim1
```
4 Hypothesis testing

Under the null hypothesis, all observations are in class A. Under the alternative hypothesis, some unlabelled observations are in class B.

The function mixtura not only fits the model under the alternative hypothesis (see above), but also under the null hypothesis:

\[
\text{fit} \cdot \text{estim0}
\]

Because the null distribution of the likelihood-ratio test statistic is unknown, we compare the hypotheses by resampling. The function scrutor uses parametric bootstrapping or permutation:

\[
\text{scrutor}(y, z)
\]

If the \(p\)-value is less than or equal to the significance level, we reject the null hypothesis in favour of the alternative hypothesis.

Options

The functions mixtura and scrutor have similar arguments. Set dist equal to "norm" or "nbinom" to choose between the Gaussian and the negative binomial distributions. In the latter case, optionally provide a dispersion estimate phi or an offset gamma. All other arguments are technical.
5 Application

5.1 Data preparation

Let \( n \) be the sample size, \( q \) the number of quantitative traits, and \( p \) the number of single nucleotide polymorphisms (SNPs).

- Transform the quantitative trait to a vector of length \( n \), or transform the quantitative traits to a matrix with \( n \) rows (samples) and \( q \) columns (variables).
- Transform the SNP to a vector of length \( n \), or transform the SNPs to a matrix with \( n \) rows (samples) and \( p \) columns (variables).
- Binarise the SNP(s), indicating the labelled group by zero, and the unlabelled group by a missing value.

For example, assign observations with zero minor alleles to the labelled group, and those with one or two minor alleles to the unlabelled group:

\[
\begin{align*}
n & \leftarrow \text{length(snp)} \\
\text{z} & \leftarrow \text{rep(NA, times=n)} \\
z[\text{snp==0}] & \leftarrow 0 \\
n & \leftarrow \text{nrow(SNPs)} \\
p & \leftarrow \text{ncol(SNPs)} \\
\text{Z} & \leftarrow \text{matrix(NA, nrow=n, ncol=p)} \\
\text{Z[SNPs==0]} & \leftarrow 0
\end{align*}
\]

5.2 Test of association

Use \textit{scrutor} to test for association between a quantitative trait (vector) and a SNP (vector). The function returns a test statistic and a \( p \)-value.

\[
y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} \quad z = \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{pmatrix}
\]

\texttt{scrutor(y,z)}
5.3 **Genome-wide association study**

Use **scrutor** to test for association between a quantitative trait (vector) and several SNPs (matrix). For each SNP, the function returns a test statistic and a $p$-value.

\[
y = \begin{pmatrix}
y_1 \\
y_2 \\
\vdots \\
y_n
\end{pmatrix}
\]

\[
Z_{n \times p} = \begin{pmatrix}
z_{1,1} & z_{1,2} & \cdots & z_{1,p} \\
z_{2,1} & z_{2,2} & \cdots & z_{2,p} \\
\vdots & \vdots & \ddots & \vdots \\
z_{n,1} & z_{n,2} & \cdots & z_{n,p}
\end{pmatrix}
\]

**scrutor**(y, Z)

5.4 **Differential expression analysis**

Use **scrutor** to test for association between several quantitative traits (matrix) and a SNP (vector). For each quantitative trait, the function returns a test statistic and a $p$-value.

\[
Y_{n \times q} = \begin{pmatrix}
y_{1,1} & y_{1,2} & \cdots & y_{1,q} \\
y_{2,1} & y_{2,2} & \cdots & y_{2,q} \\
\vdots & \vdots & \ddots & \vdots \\
y_{n,1} & y_{n,2} & \cdots & y_{n,q}
\end{pmatrix}
\]

\[
Z = \begin{pmatrix}
z_1 \\
z_2 \\
\vdots \\
z_n
\end{pmatrix}
\]

**scrutor**(Y, z)

5.5 **Expression quantitative trait loci analysis**

Use **scrutor** to test for association between several quantitative traits (matrix) and several SNPs (matrix). If their numbers are different, all pairwise combinations are considered. If their numbers are equal, a one-to-one correspondence
is assumed. For each combination, the function returns a test statistic and a $p$-value.

\[
Y_{n \times q} = \begin{pmatrix}
y_{1,1} & y_{1,2} & \cdots & y_{1,q} \\
y_{2,1} & y_{2,2} & \cdots & y_{2,q} \\
\vdots & \vdots & \ddots & \vdots \\
y_{n,1} & y_{n,2} & \cdots & y_{n,q}
\end{pmatrix}
\]

\[
Z_{n \times p} = \begin{pmatrix}
z_{1,1} & z_{1,2} & \cdots & z_{1,p} \\
z_{2,1} & z_{2,2} & \cdots & z_{2,p} \\
\vdots & \vdots & \ddots & \vdots \\
z_{n,1} & z_{n,2} & \cdots & z_{n,p}
\end{pmatrix}
\]

\text{scrutor}(Y, Z)

References

The R package semisup is based on Rauschenberger et al. [1], where detailed references to previous work are given. If you use semisup for publications, please cite Rauschenberger et al. [1].

Consider shrinkage estimation (Robinson et al. [3]) and scale normalisation (Robinson et al. [2]) to improve the negative binomial mixture model (R package edgeR). Use the non-parametric mixture test (van Wieringen et al. [4]) to increase robustness against outliers (R package PDGEtest).


