# Package ‘phenopath’

March 7, 2024

**Type** Package  

**Title** Genomic trajectories with heterogeneous genetic and environmental backgrounds  

**Version** 1.26.0  

**Date** 2017-10-24  

**Author** Kieran Campbell  

**Maintainer** Kieran Campbell &lt;kieranrcampbell@gmail.com&gt;  

**Description** PhenoPath infers genomic trajectories (pseudotimes) in the presence of heterogeneous genetic and environmental backgrounds and tests for interactions between them.  

**License** Apache License (== 2.0)  

**Imports** Rcpp (&gt;= 0.12.8), SummarizedExperiment, methods, stats, dplyr, tibble, ggplot2, tidyr  

**Suggests** knitr, rmarkdown, forcats, testthat, BiocStyle, SingleCellExperiment  

**biocViews** ImmunoOncology, RNASeq, GeneExpression, Bayesian, SingleCell, PrincipalComponent  

**LinkingTo** Rcpp  

**RoxygenNote** 6.0.1  

**VignetteBuilder** knitr  

**git_url** https://git.bioconductor.org/packages/phenopath  

**git_branch** RELEASE_3_18  

**git_last_commit** 2f83790  

**git_last_commit_date** 2023-10-24  

**Repository** Bioconductor 3.18  

**Date/Publication** 2024-03-06
clvm Fit a CLVM Model

Description
Fit a covariate latent variable model using coordinate ascent variational inference.

Usage
clvm(y, x, maxiter = 10000, elbo_tol = 1e-05, thin = 1, verbose = TRUE,
z_init = 1, tau_q = 1, tau_mu = 1, tau_c = 1, a = 2, b = 2,
tau_alpha = 1, a_beta = 10, b_beta = 1, q = rep(0, nrow(y)),
model_mu = FALSE, scale_y = TRUE)

Arguments
y A N-by-G (dynamic) input matrix
x A N-by-P (static) input matrix
maxiter Maximum number of CAVI iterations
elbo_tol The (percent) change in the ELBO below which it is considered converged
thin The number of iterations to wait each time before re-calculating the elbo
verbose Print convergence messages
z_init The initialisation of the latent trajectory. Should be one of
1. A positive integer describing which principal component of the data should be used for initialisation (default 1), or
2. A numeric vector of length number of samples to be used directly for initialisation, or
3. The text character "random", for random initialisation from a standard normal distribution.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tau_q</td>
<td>Hyperparameter tau_q</td>
</tr>
<tr>
<td>tau_mu</td>
<td>Hyperparameter tau_mu</td>
</tr>
<tr>
<td>tau_c</td>
<td>Hyperparameter tau_c</td>
</tr>
<tr>
<td>a</td>
<td>Hyperparameter a</td>
</tr>
<tr>
<td>b</td>
<td>Hyperparameter b</td>
</tr>
<tr>
<td>tau_alpha</td>
<td>Hyperparameter tau_alpha</td>
</tr>
<tr>
<td>a_beta</td>
<td>Hyperparameter a_beta</td>
</tr>
<tr>
<td>b_beta</td>
<td>Hyperparameter b_beta</td>
</tr>
<tr>
<td>q</td>
<td>Priors on the latent variables</td>
</tr>
<tr>
<td>model_mu</td>
<td>Logical - should a gene-specific intercept term be modelled?</td>
</tr>
<tr>
<td>scale_y</td>
<td>Logical - should the expression matrix be centre scaled?</td>
</tr>
</tbody>
</table>

Value

A list whose entries correspond to the converged values of the variational parameters along with the ELBO.

Examples

```r
sim <- simulate_phenopath()
fit <- clvm(sim$y, matrix(sim$x))
```

---

**interactions** | **Tidy summary of interactions**

**Description**

Computes a tidy data frame of the interaction effects, pathway scores, and significance for each gene and covariate interaction.

**Usage**

```r
interactions(phenopath_fit, n = 3)
```

**Arguments**

- **phenopath_fit** An object returned by a call to phenopath
- **n** The number of standard deviations away from 0 the posterior of the interaction effect sizes should be to be designated as significant
interaction_effects

Value

A data frame with the following columns:

- **feature** The feature (usually gene)
- **covariate** The covariate, specified from the order originally supplied to the call to phenopath
- **interaction_effect_size** The effect size of the interaction ($\beta$ from the statistical model)
- **significant** Boolean for whether the interaction effect is significantly different from 0
- **chi** The precision of the ARD prior on $\beta$
- **pathway_loading** The pathway loading $\lambda$, showing the overall effect for each gene marginalised over the covariate

Examples

```r
sim <- simulate_phenopath()
fit <- phenopath(sim$y, sim$x, elbo_tol = 1e-2)
interactions(fit)
```

Description

Get the interaction effect sizes

Usage

```r
interaction_effects(phenopath_fit)
```

Arguments

- **phenopath_fit** An object of class phenopath_fit

Value

TODO

Examples

```r
sim <- simulate_phenopath()
fit <- phenopath(sim$y, sim$x, elbo_tol = 1e-2)
beta <- interaction_effects(fit)
```
interaction_sds

Get the interaction standard deviations

Description

Get the interaction standard deviations

Usage

interaction_sds(phenopath_fit)

Arguments

phenopath_fit An object of class phenopath_fit

Value

TODO

Examples

sim <- simulate_phenopath()
fit <- phenopath(sim$y, sim$x, elbo_tol = 1e-2)
beta_sd <- interaction_sds(fit)

phenopath

PhenoPath - genomic trajectories with heterogeneous backgrounds

Description

PhenoPath learns genomic trajectories in the presence of heterogeneous environmental and genetic backgrounds. It takes input gene expression measurements that are modelled by a single unobserved factor (the "trajectory"). The regulation of genes along the trajectory is perturbed by an additional set of covariates (such as genetic or environmental status) allowing for the identification of covariate-trajectory interactions. The model is fitted using mean-field co-ordinate ascent variational inference.

Usage

phenopath(exprs_obj, x, sce_assay = "exprs", elbo_tol = 1e-05, z_init = 1, ...)
Arguments

exprs_obj  
Input gene expression, either
1. An `SummarizedExperiment` object, or

x  
The covariate vector, either
1. The name of a column of `colData(exprs_obj)` if `exprs_obj` is an `SummarizedExperiment`, or
2. A numeric of factor vector of length equal to the number of cells, or
3. A formula from which to build a model matrix from `colData(exprs_obj)`, if `exprs_obj` is a `SummarizedExperiment`

sce_assay  
The assay from `exprs_obj` to use as expression if `exprs_object` is a `SummarizedExperiment`

elbo_tol  
The relative pct change in the ELBO below which is considered converged. See convergence section in details below.

z_init  
The initialisation of the latent trajectory. Should be one of
1. A positive integer describing which principal component of the data should be used for initialisation (default 1), or
2. A numeric vector of length number of samples to be used directly for initialisation, or
3. The text character "random", for random initialisation from a standard normal distribution.

...  
Additional arguments to be passed to `clvm`. See description below for more details or call `?clvm`.

Details

Input expression
If an `SummarizedExperiment` is provided, `assay(exprs_obj, sce_assay)` is used. This is assumed to be in a form that is suitably normalised and approximately normal, such as the log of TPM values (plus a suitable offset) or similar.

Encoding covariates
See the vignette.

Convergence of variational inference
It is strongly recommended to call `plot_elbo(phenopath_fit)` after the fitting procedure to ensure the ELBO has approximately converged (though convergence metrics are printed during the fitting process). For a good introduction to variational inference see Blei, D.M., Kucukelbir, A. & McAuliffe, J.D., 2017. Variational Inference: A Review for Statisticians. Journal of the American Statistical Association.

Additional arguments
Addition arguments to `clvm` are passed via .... For full documentation, call `?clvm`. Some notable options:

- `thin` - The ELBO is expensive to compute for larger datasets. The model will compute the ELBO and compare convergence every `thin` iterations.
plot_elbo

- `q` and `tau_q` - Priors (such as capture times) for the latent space. Note that `model_mu` should be true if `q` is non-zero.
- `scale_y` By default the input expression is centre-scaled for each gene. If `scale_y` is `FALSE` this does not happen - but note that `model_mu` should be `TRUE` in such a case.

Value

An S3 structure with the following entries:

- `m_z` The converged mean estimates of the trajectory
- `s_z` The converged standard deviation estimates of z
- `m_beta` A P-by-G matrix of interaction coefficients
- `s_beta` A P-by-G matrix of interaction standard deviations

See Also

`clvm` for the underlying CAVI function, `trajectory` to extract the latent trajectory, `interaction_effects` for the interaction effect sizes, `significant_interactions` for the results of Bayesian significance testing.

Examples

```r
sim <- simulate_phenopath()  # returns a list with gene expression in y and covariates in x
fit <- phenopath(sim$y, sim$x, elbo_tol = 1e-2)

# Extract the trajectory
z <- trajectory(fit)
```

---

plot_elbo

Plots the ELBO

Description

Plots the evidence lower bound (ELBO) as a function of iterations

Usage

```r
plot_elbo(fit)
```

Arguments

- `fit` An object returned by a call to `phenopath`

Value

A `ggplot2` object of the ELBO against the number of iterations
Examples

```
sim <- simulate_phenopath()
fit <- phenopath(sim$y, sim$x)
plot_elbo(fit)
```

print.phenopath_fit  Print a PhenoPath fit

Description

Print a PhenoPath fit

Usage

```
## S3 method for class 'phenopath_fit'
print(x, ...)
```

Arguments

- `x` A phenopath_fit returned by a call to phenopath
- `...` Additional arguments

Value

A string representation of a phenopath_fit object.

Examples

```
sim <- simulate_phenopath()
fit <- phenopath(sim$y, sim$x, elbo_tol = 1e-2)
print(fit)
```

sample_fns  Sample parameters for simulation

Description

Sample parameters from de regime
Sample parameters from pseudotime regime
Sample parameters from pseudotime-interaction regime
Sample parameters from de-pseudotime-interaction regime
scale_vec

Usage

sample_de()
sample_pst()
sample_pst_beta()
sample_de_pst_beta()

Value

A length-3 vector of parameters corresponding to the particular simulation regime

scale_vec Scale a vector to have mean 0 and variance 1

Description

Scales vector to mean 0 variance 1 unless input standard deviation is 0 in which case original vector is returned

Usage

scale_vec(x)

Arguments

x Input vector to scale

Value

Scaled vector

significant_interactions

Significance testing for interaction features

Description

Given the results of clvm, decide which features show significant interactions between the latent trajectory and covariates. Significant features are designated as those where the variational mean of the interaction coefficient falls outside the \( n\sigma \) interval of 0.

Usage

significant_interactions(phenopath_fit, n = 3)
simulate_phenopath

Arguments

- phenopath_fit: The results of a call to clvm
- n: The number of standard deviations away from 0 the posterior estimate of beta should be to be designated significant.

Value

A logical vector describing whether each feature passes the significance test.

Examples

```r
sim <- simulate_phenopath()
fit <- phenopath(sim$y, sim$x, elbo_tol = 1e-2)
signints <- significant_interactions(fit)
```

simulate_one_gene

Simulate one gene

Description

Simulate one gene from the model given parameters, z and covariates

Usage

```r
simulate_one_gene(N, pst, x, alpha = 0, c = 0, beta = 0, tau = 1e+06)
```

Value

A length-N gene expression vector simulated with the PhenoPath mean function for the given parameters

simulate_phenopath

Simulate from phenopath model

Description

Simulate synthetic data from the phenopath model, where each gene is sampled from one of four types (see details).

Usage

```r
simulate_phenopath(N = 100, G = 40, G_de = NULL, G_pst = NULL, G_pst_beta = NULL, G_de_pst_beta = NULL)
```
**trajectory**

Arguments

- **N**
  Number of samples to simulate

- **G**
  Number of genes to simulate. Should be divisible by 4

- **G_de**
  Number of genes to simulate from the *differential expression* regime

- **G_pst**
  Number of genes to simulate from the *pseudotime* regime

- **G_pst_beta**
  Number of genes to simulate from the *pseudotime + beta interactions* regime

- **G_de_pst_beta**
  Number of genes to simulate from the *differential expression + pseudotime + interactions* regime

Details

Will simulate data for a number of genes corresponding to one of four regimes:

1. **de** (‘differential expression’), where the gene has no association to the latent trajectory and exhibits differential expression only
2. **pst** (‘pseudotime’), the gene shows pseudotemporal regulation but no differential regulation
3. **pst_beta** (‘pseudotime + beta interactions’), the gene shows pseudotemporal regulation that is modulated by covariate interactions
4. **de_pst_beta** (‘differential expression + pseudotime + interactions’), all of the above

Value

A list with four entries:

- **parameters** A tibble with an entry for each gene and a column for each parameter value and simulation regime (see details).
- **y** The N-by-G simulated gene expression matrix.
- **x** The N-length vector of covariates.
- **z** The N-length vector of pseudotimes.

Examples

```r
sim <- simulate_phenopath()
```

---

**trajectory**

Get the latent PhenoPath trajectory

**Description**

Get the latent PhenoPath trajectory

**Usage**

```r
trajectory(phenopath_fit)
```
**Arguments**

- `phenopath_fit`  An object of class `phenopath_fit`

**Value**

A vector of latent trajectory (pseudotime) values

**Examples**

```r
sim <- simulate_phenopath()
fit <- phenopath(sim$y, sim$x, elbo_tol = 1e-2)
z <- trajectory(fit)
```
Index

* internal
  sample_fns, 8
  scale_vec, 9
  simulate_one_gene, 10

clvm, 2, 6, 7

interaction_effects, 4, 7
interaction_sds, 5
interactions, 3

phenopath, 5
plot_elbo, 7
print.phenopath_fit, 8

sample_de(sample_fns), 8
sample_de_pst_beta(sample_fns), 8
sample_fns, 8
sample_pst(sample_fns), 8
sample_pst_beta(sample_fns), 8
scale_vec, 9
significant_interactions, 7, 9
simulate_one_gene, 10
simulate_phenopath, 10
SummarizedExperiment, 6

trajectory, 7, 11