# Package 'epoc'

February 19, 2015

**Version** 0.2.5-1

**Date** 2013-08-22

**Encoding** UTF-8

**Title** EPoC (Endogenous Perturbation analysis of Cancer)

**Author** Rebecka Jornsten, Tobias Abenius, Sven Nelander

**Maintainer** Tobias Abenius <Tobias.Abenius@Chalmers.se>

**Depends** R (>= 2.12.0), lassoshooting (>= 0.1.4), Matrix, methods, graph

**Imports** irr, elasticnet, survival, Rgraphviz

**Suggests** RCytoscape

**Description** Estimates sparse matrices A or G using fast lasso regression from mRNA transcript levels Y and CNA profiles U. Two models are provided, EPoC A where
AY + U + R = 0
and EPoC G where
Y = GU + E,
the matrices R and E are so far treated as noise. For details see the reference and the manual page of `lassoshooting'.

**License** LGPL-3

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2013-08-26 17:51:58

## R topics documented:

EPoC *EPoC*

## Description

EPoC (Endogenous Perturbation analysis of Cancer)

## Usage

```
epocA(Y, U=NULL, lambdas=NULL, thr=1.0e-10, trace=0, ...)
epocG(Y, U, lambdas=NULL, thr=1.0e-10, trace=0, ...)
epoc.lambdamax(X, Y, getall=F, predictorix=NULL)
as.graph.EPOCA(model,k=1)
as.graph.EPOCG(model,k=1)
write.sif(model, k=1, file="", append=F)
## S3 method for class 'EPOCA'
print(x,...)
## S3 method for class 'EPOCG'
print(x,...)
## S3 method for class 'EPOCA'
summary(object, k=NULL, ...)
## S3 method for class 'EPOCG'
summary(object, k=NULL,...)
## S3 method for class 'EPOCA'
coef(object, k=1, ...)
## S3 method for class 'EPOCG'
coef(object, k=1, ...)
## S3 method for class 'EPOCA'
predict(object, newdata,k=1,trace=0, ...)
## S3 method for class 'EPOCG'
predict(object, newdata,k=1,trace=0, ...)
```

## Arguments

| | |
|---|---|
| Y | N x p matrix of mRNA transcript levels for p genes and N samples for epocA and epocG. For epoc.lambdamax Y is a multi-response matrix |
| U | N x p matrix of DNA copy number |
| lambdas | Non-negative vector of relative regularization parameters for lasso. $\lambda = 0$ means no regularization which give dense solutions (and takes longer to compute). Default=NULL means let EPoC create a vector |
| thr | Threshold for convergence. Default value is 1e-10. Iterations stop when max absolute parameter change is less than thr |
| trace | Level of detail for printing out information as iterations proceed. Default 0 – no information |
| X | In epoc.lambdamax X is the design matrix, i.e. predictors |

| | |
|---|---|
| predictorix | For epoc.lambdamax when using a multi-response matrix Y predictors are set to zero for each corresponding response. predictorix tells which of the responses that have a corresponding predictor in the network case |
| getall | Logical. For epoc.lambdamax get a vector of all inf-norms instead of a single maximum |
| file | either a character string naming a file or a connection open for writing. "" indicates output to the console |
| append | logical. Only relevant if file is a character string. If TRUE, the output is appended to the file. If FALSE, any existing file of the name is destroyed |
| model | Model set from epocA or epocG |
| k | Select a model of sparsity level k in [1,K]. In summary default (NULL) means all. In plot default is first model. |
| newdata | List of Y and U matrices required for prediction. epocG requires just U. |
| x | Model parameter to print and plot |
| object | Model parameter to summary, coef and predict |
| ... | Parameters passed down to underlying function, e.g. print.default. For epocA and epocG ... are reserved for experimental options. |

## Details

epocA and epocG estimates sparse matrices $A$ or $G$ using fast lasso regression from mRNA transcript levels $Y$ and CNA profiles $U$. Two models are provided, EPoC A where

$$AY + U + R = 0$$

and EPoC G where

$$Y = GU + E.$$

The matrices $R$ and $E$ are so far treated as noise. For details see the reference section and the manual page of lassoshooting.

If you have different sizes of U and Y you need to sort your Y such that the U-columns correspond to the first Y-columns. Example: `Y.new <- cbind(Y[,haveCNA], Y[, -haveCNA])` CHANGES: predictorix used to be a parameter with a vector of a subset of the variables 1:p of U corresponding to transcripts in Y, Default was to use all which mean that Y and U must have same size.

epoc.lambdamax returns the maximal $\lambda$ value in a series of lasso regression models such that all coefficients are zero.

plot if type='graph' (default) plot graph of model using the Rgraphviz package arrows only tell direction, not inhibit or stimulate. If type='modelsel' see modelselPlot.

## Value

`epocA` and `epocG` returns an object of `class` '"epocA"' and '"epocG"' respectively.

The methods `summary`, `print`, `coef`, `predict` can be used as with other models. `coef` and `predict` take an extra optional integer parameter `k` (default 1) which gives the model at the given density level.

An object of these classes is a list containing at least the following components:

| | |
|---|---|
| `coefficients` | list of t(A) or t(G) matrices for the different $\lambda$s |
| `links` | the number of links for the different $\lambda$s |
| `lambdas` | the $\lambda$s used for this model |
| `R2` | $R^2$, coefficient of determination |
| `Cp` | Mallows Cp |
| `s2` | Estimate of the error variance |
| `RSS` | Residual Sum of Squares (SSreg) |
| `SS.tot` | Total sum of squares of the response |
| `inorms` | the infinity norm of predictors transposed times response for the different responses |
| `d` | Direct effects of CNA to corresponding gene |

## Note

The `coef` function returns transposed versions of the matrices $A$ and $G$.

## Author(s)

Tobias Abenius

## References

Rebecka Jörnsten, Tobias Abenius, Teresia Kling, Linnéa Schmidt, Erik Johansson, Torbjörn Nordling, Bodil Nordlander, Chris Sander, Peter Gennemark, Keiko Funa, Björn Nilsson, Linda Lindahl, Sven Nelander. (2011) Network modeling of the transcriptional effects of copy number aberrations in glioblastoma. Molecular Systems Biology 7 (to appear)

## See Also

`print`, `modelselPlot`, `epoc.validation`, `epoc.bootstrap`, `plot.EPoC.validation.pred`, `plot.EPoC.validation.W`, `coef`, `predict`

## Examples

```
## Not run:
modelA <- epocA(X,U)
modelG <- epocG(X,U)

# plot sparsest A and G models using the igraph package
# arrows only tell direction, not inhibit or stimulate
par(mfrow=c(1,2))
plot(modelA)
plot(modelG)

# OpenGL 3D plot on sphere using the igraph and rgl packages
plot(modelA,threed=T)

# Write the graph to a file in SIF format for import in e.g. Cytoscape
write.sif(modelA,file="modelA.sif")

# plot graph in Cytoscape using Cytoscape XMLRPC plugin and
# R packages RCytoscape, bioconductor graph, XMLRPC
require('graph')
require('RCytoscape')
g <- as.graph.EPOCA(modelA,k=5)
cw <- CytoscapeWindow("EPoC", graph = g)
displayGraph(cw)

# prediction
N <- dim(X)[1]
ii <- sample(1:N, N/3)

modelG <- epocG(X[ii,], U[ii,])
K <- length(modelA$lambda) # densest model index index
newdata <- list(U=U[-ii,])
e <- X[-ii,] - predict(modelA, newdata, k=K)
RSS <- sum(e^2)
cat("RMSD:", sqrt(RSS/N), "\n")


## End(Not run)
```

---

epoc.bootstrap *epoc.bootstrap*

---

## Description

Bootstrap for the EPoC methods

## Usage

```
epoc.bootstrap(Y, U, nboots=100, bthr=NULL, method='epocG',...)
```

```
  ## S3 method for class 'bootsize'
plot(x, lambda.boot=NULL, B, range=c(0,1), ...)
  epoc.final(epocboot, bthr=0.2, k)
```

## Arguments

| | |
|---|---|
| Y | mRNA, samples x genes. |
| U | CNA, samples x genes. |
| nboots | Number of bootstrap iterations to run. |
| method | For epoc.bootstrap method is "epocG" or "epocA". |
| x | A sparse network matrix or a list of the same, non-zeros are links. These come from e.g. epoc.final or epoc.bootstrap. |
| lambda.boot | The $\lambda$s used to run the bootstrap. |
| B | Number of bootstrap iterations ran for 'plot.bootsize'. |
| range | Range of bootstrap thresholds to display. |
| epocboot | For epoc.final give a list of bootstraped models from epoc.bootstrap. |
| k | For epoc.final and epoc.bootplot select the k sparsest model. |
| bthr | Require presence of links in 100*bthr% of the bootstrap iterations. |
| ... | Parameters passed down to an underlying function. For epoc.bootstrap these are passed down to "epocG" or "epocA" respectively. For epoc.bootplot and plot.bootsize parameters are passed down to the underlying plot command. |

## Details

epoc.bootstrap run epocA or epocG using bootstrap.

## Value

epoc.bootstrap returns a list of $p \times p$ arrays of values in $[0, 1]$ where 1 is presence of link in 100% of bootstrap iterations for the $k$ different $\lambda$ values for $p$ different genes. epoc.final returns a sparse matrix of $p \times p$ values in $[0, 1]$ where 1 is presence of link in 100% of bootstrap iterations, but thresholded such that all values have be greater than or equal to bthr.

## References

Rebecka Jörnsten, Tobias Abenius, Teresia Kling, Linnéa Schmidt, Erik Johansson, Torbjörn Nordling, Bodil Nordlander, Chris Sander, Peter Gennemark, Keiko Funa, Björn Nilsson, Linda Lindahl, Sven Nelander. (2011) Network modeling of the transcriptional effects of copy number aberrations in glioblastoma. Molecular Systems Biology 7 (to appear)

## See Also

epoc, plot.EPoC.validation, plot.EPOCA, plot.EPOCG

epoc.survival *epoc.survival*

## Description

Survival analysis

## Usage

```
  epoc.svd(model, k=1, C=1, numload=NULL)
  epoc.survival(G.svd, Y, U, surv, C=1, type=NULL)
epoc.svdplot(G.svd, C=1)
  ## S3 method for class 'EPoC.survival'
plot(x,...)
  ## S3 method for class 'EPoC.survival'
summary(object,...)
  ## S3 method for class 'summary.EPoC.survival'
print(x,...)
```

## Arguments

| | |
|---|---|
| model | An object from epocG or epocA or a Matrix from epoc.bootstrap and friends. |
| k | In case model come from epocG or epocA select a model of sparsity level k in [1,K]. The default k=1 means first/most sparse. |
| C | Default 1. For epoc.svd the number of components. For epoc.survival and epoc.svdplot, which component to use. |
| numload | Number of loadings in the sparse components, a vector for each component. Default 10 for all components. |
| G.svd | The list obtained from epoc.svd. |
| Y | mRNA, samples x genes. |
| U | CNA, samples x genes. |
| surv | Survival data for the samples. |
| type | 'G' means EPoC G and 'A' means EPoC A. |
| x | An object from epoc.survival |
| object | An object from epoc.survival |
| ... | Parameters passed down to underlying functions, plot.default for plot and print.default for print. |

## Details

Applies survival analysis using the first SVD component, but other components can also be used by changing the input value of C. Survival scores are generated as described in Subsect. 2.4 in the second paper referenced. A simple non-parametric survival analysis is performed, comparing survival between patientswith positive or negative scores (tumor fitness).

## Value

The epoc.survival object contains the summary information from a log-rank test comparing survival (survdiff) and survival fit objects.

## References

Rebecka Jörnsten, Tobias Abenius, Teresia Kling, Linnéa Schmidt, Erik Johansson, Torbjörn Nordling, Bodil Nordlander, Chris Sander, Peter Gennemark, Keiko Funa, Björn Nilsson, Linda Lindahl, Sven Nelander. (2011) Network modeling of the transcriptional effects of copy number aberrations in glioblastoma. Molecular Systems Biology 7

Tobias Abenius, Rebecka Jörnsten, Teresia Kling, Linnéa Schmidt, José Sánchez, Sven Nelander. (2012) System-scale network modeling of cancer using EPoC. Advances in experimental medicine and biology

## See Also

epoc, epoc.validation and spca

---

`epoc.validation` *epoc.validation*

---

## Description

Model validation using random split or cross-validation

## Usage

```
epoc.validation(type=c('pred','concordance'),repl,Y,U,lambdas=NULL,
   method='G',thr=1e-10,trace=0,...)
```

## Arguments

| | |
|---|---|
| `type` | `'pred'` for 10-fold CV of prediction error. `'concordance'` for random split network concordance using Kendall $W$. |
| `repl` | The number of replicates |
| `Y` | mRNA, samples x genes |
| `U` | CNA, samples x genes |
| `lambdas` | series of relative $\lambda$s or default=`NULL` which means let EPoC choose |
| `method` | `'G'` means EPoC G and `'A'` means EPoC A. |
| `thr` | Threshold for convergence to the LASSO solver |
| `trace` | Debug information |
| `...` | Extra parameters passed through to the EPoC solver |

### Details

In the case of `'pred'` assess CV prediction error using 10-fold cross-validation with `repl` replicates. In the case of `'concordance'` assess network concordance using random split and Kendall W with `repl` replicates.

### Value

A list of class `plot.EPoC.validation.pred` or `plot.EPoC.validation.W` respectively.

### References

Rebecka Jörnsten, Tobias Abenius, Teresia Kling, Linnéa Schmidt, Erik Johansson, Torbjörn Nordling, Bodil Nordlander, Chris Sander, Peter Gennemark, Keiko Funa, Björn Nilsson, Linda Lindahl, Sven Nelander. (2011) Network modeling of the transcriptional effects of copy number aberrations in glioblastoma. Molecular Systems Biology 7 (to appear)

### See Also

`epoc`, `plot.EPoC.validation`

---

## `modelselPlot` *Plot BIC, Mallow's Cp and $\lambda$*

---

### Description

Plot BIC, Mallow's Cp and $\lambda$

### Usage

```
modelselPlot(x, layout=NULL, k=1, showtitle=F, bthr=0,
             showself=F, type=c('graph','modelsel'), ...)
## S3 method for class 'EPOCA'
plot(x, layout=NULL, k=1, showtitle=F, bthr=0,
             showself=F, type=c('graph','modelsel'), ...)
## S3 method for class 'EPOCG'
plot(x, layout=NULL, k=1, showtitle=F, bthr=0,
             showself=F, type=c('graph','modelsel'), ...)
```

### Arguments

| | |
|---|---|
| `x` | An EPoC G or EPoC A object |
| `layout` | Not used only for `type='modelsel'` |
| `k` | Not used for `type='modelsel'` |
| `showtitle` | Not used for `type='modelsel'` |
| `bthr` | Not used for `type='modelsel'` |
| `showself` | Not used for `type='modelsel'` |

| | |
|---|---|
| type | This page documents type='modelsel' only, for type='graph' see plot.EPOCG and epoc.bootplot |
| ... | Parameters passed down to underlying functions, plot, lines, points, abline, axis, text and legend. |

## Details

Creates a plot that aids in model selection. Scale Bayesian information criterion (BIC) and Mallow's $C_p$ between zero on one and put that on the y-axis and put relative $\lambda$ values on the x-axis.

## References

Rebecka Jörnsten, Tobias Abenius, Teresia Kling, Linnéa Schmidt, Erik Johansson, Torbjörn Nordling, Bodil Nordlander, Chris Sander, Peter Gennemark, Keiko Funa, Björn Nilsson, Linda Lindahl, Sven Nelander. (2011) Network modeling of the transcriptional effects of copy number aberrations in glioblastoma. Molecular Systems Biology 7 (to appear)

## See Also

epoc

---

# plapply *Parallell list apply*

---

## Description

Parallell list apply

## Usage

```
plapply(X1, X2, FUN, ...)
```

## Arguments

| | |
|---|---|
| X1 | a vector (atomic or list). Other objects (including classed objects) will be coerced by as.list. |
| X2 | See X1. |
| FUN | the function to be applied to each pair of X1 and X2 at the corresponding positions. |
| ... | optional arguments to FUN. |

### Details

FUN is found by a call to match.fun and typically is specified as a function or a symbol (e.g. a back-quoted name) or a character string specifyign a function to be searched for from the environment of the call to plapply.

Function FUN must be able to accept as input any of the element pairs of X1 and X2. If any of these are atomic vectors, FUN will always be passed a length-one vector of the same type as X1, X2 respectively.

Users of S4 classes should pass a list to plapply: the internal coercion is done by the system as.list in the base namespace and not one defined by a user (e.g. by setting S4 methods on the system function).

### Value

A list.

### See Also

lapply

### Examples

```
X1 <- array(1:4,dim=c(2,2))
X2 <- array(5:8,dim=c(2,2))
X3 <- array(9:12,dim=c(2,2))
X4 <- array(13:16,dim=c(2,2))
l <- plapply(list(X1,X2),list(X3,X4), function(E1,E2) E2 - E1)
stopifnot(all(sapply(l,sum)/4 == 4*2))
```

---

## plot.EPoC.validation *Plot model validation criteria*

---

### Description

Plot model validation criteria

### Usage

```
  ## S3 method for class 'EPoC.validation.W'
plot(x, ...)
  ## S3 method for class 'EPoC.validation.pred'
plot(x, ...)
```

### Arguments

| | |
|---|---|
| x | An object of type EPoC.validation.W or EPoC.validation.W respectively. |
| ... | Parameters passed down to underlying functions, plot, lines, points, abline. |

## Details

Plot Kendall W or prediction error, respectively on the y-axis, network size on the upper x-axis and $\lambda$ on the lower x-axis. The plot fit a loess model of degree 1 to the points from the input object and finds the largest network size and corresponding $\lambda$ such that W is maximized or prediction error is minimized, respectively.

## References

Rebecka Jörnsten, Tobias Abenius, Teresia Kling, Linnéa Schmidt, Erik Johansson, Torbjörn Nordling, Bodil Nordlander, Chris Sander, Peter Gennemark, Keiko Funa, Björn Nilsson, Linda Lindahl, Sven Nelander. (2011) Network modeling of the transcriptional effects of copy number aberrations in glioblastoma. Molecular Systems Biology 7 (to appear)

## See Also

epoc, epoc.validation, plot.default

---

# synth *Blinded cancer mRNA, CNA and survival data*

---

## Description

This dataset contains blinded mRNA, CNA and survival data of 186 cancer tumors modified for demonstration usage. Some genes are randomly selected from 10672 probes, others are chosen for their characteristics.

mRNA is standardized to sd=1 and mean=0. CNA is centered to mean=0. survival is in days.

## Usage

```
synth
```

## Format

The synth data set is a list containing mRNA y, CNA u and surv survival data.

## Examples

```
## Not run:
  data(synth)
  y <- synth$y
  # standardize u
  u <- apply(synth$u, 2, function(x) x/sd(x))
  G <- epocG(Y=y, U=u)
  summary(G)
  plot(G)

## End(Not run)
```

# Index