

Package ‘eNetXplorer’

June 11, 2018

Type Package

Title Quantitative Exploration of Elastic Net Families for Generalized Linear Models

Version 1.0.1

Date 2018-06-10

Author Julian Candia and John S. Tsang

Maintainer Julian Candia <julian.candia@nih.gov>

Depends R (>= 2.10)

Imports glmnet, stats, graphics, methods, grDevices, Matrix, progress, calibrate, RColorBrewer, gplots

Description Provides a quantitative toolkit to explore elastic net families and to uncover correlates contributing to prediction under a cross-validation framework. Fits linear, binomial (logistic) and multinomial models. Candia J and Tsang JS (2018), bioRxiv 305870 (under review).

License GPL-3

LazyData true

LazyLoad yes

NeedsCompilation no

Repository CRAN

Date/Publication 2018-06-10 22:04:22 UTC

R topics documented:

eNetXplorer-package	2
eNetXplorer	3
export	7
H1N1_Flow	8
Leukemia_miR	9
plot	10
plotContingency	11
plotFeatureCaterpillar	12
plotFeatureHeatmap	14

plotLambdaVsQF	16
plotMeasuredVsOOB	17
plotSummary	19
QuickStartEx	20
summary	21
summaryPDF	22

Index	23
--------------	-----------

eNetXplorer-package *explores elastic net families for generalized linear models*

Description

Provides a quantitative toolkit to explore elastic net families and to uncover correlates contributing to prediction under a cross-validation framework. Fits linear, binomial (logistic) and multinomial models.

Details

Package: eNetXplorer
 Type: Package
 Version: 1.0.1
 Date: 2018-06-10
 License: GPL-3

This package provides a full pipeline of analysis: [eNetXplorer](#) takes in x, y data to generate a family of elastic net models over a range of alpha values; [summary](#) generates a summary of results in tabular format; [plot](#) provides a variety of plots to visualize results; [summaryPDF](#) generates a report in PDF format; and [export](#) creates plain text output files for downstream processing.

Author(s)

Julian Candia and John S. Tsang
 Maintainer: Julian Candia <julian.candia@nih.gov>

References

Candia J and Tsang JS (2018) *eNetXplorer: an R package for the quantitative exploration of elastic net families for generalized linear models*, bioRxiv 305870 (under review).

See Also

[eNetXplorer](#), [summary](#), [plot](#), [summaryPDF](#), [export](#)

Examples

```
data(QuickStartEx)
fit = eNetXplorer(x=QuickStartEx$predictor,y=QuickStartEx$response,
family="gaussian",n_run=20,n_perm_null=10,seed=111)
summary(fit)
plot(x=fit,plot.type="measuredVsO0B",alpha.index=4)
suppressWarnings(summaryPDF(x=fit,path=tempdir()))
export(x=fit,path=tempdir())
```

eNetXplorer

generates family of elastic net models for different alphas

Description

Elastic net uses a mixing parameter α to tune the penalty term continuously from ridge ($\alpha=0$) to lasso ($\alpha=1$). eNetXplorer generates a family of elastic net models over different values of α for the quantitative exploration of the effects of shrinkage. For each α , the regularization parameter λ is chosen by optimizing a quality function based on out-of-bag cross-validation predictions. Statistical significance of each model, as well as that of individual features within a model, is assigned by comparison to a set of null models generated by random permutations of the response. eNetXplorer fits linear (gaussian), logistic (binomial) and multinomial models.

Usage

```
eNetXplorer(x, y, family=c("gaussian","binomial","multinomial"),
alpha=seq(0,1,by=0.2), nlambda=100, nlambda.ext=NULL, seed=NULL, scaled=T,
n_fold=5, n_run=100, n_perm_null=25, QF.FUN=NULL, QF_label=NULL,
cor_method=c("pearson","kendall","spearman"), fold_distrib_fail.max=100, ...)
```

Arguments

x	Input numerical matrix with instances as rows and features as columns. Instance and feature labels should be provided as row and column names, respectively. Can be in sparse matrix format (inherit from class "sparseMatrix" as in package Matrix). Cannot handle missing values.
y	Response variable. For family="gaussian", numerical vector. For family="binomial", factor with two levels. For family="multinomial", factor with two or more levels. For categorical families, if a vector is supplied, it will be coerced into a factor.
family	Response type: "gaussian" (numerical), "binomial" (2-level factor), or "multinomial" (factor with ≥ 2 levels).
alpha	Sequence of values for the mixing parameter penalty term in the elastic net family. Default is $\text{seq}(0, 1, \text{by}=0.2)$.

<code>nlambda</code>	Number of values for the regularization parameter <code>lambda</code> . Default is 100. Irrespective of <code>nlambda</code> , the range of <code>lambda</code> values is assigned by <code>glmnet</code> .
<code>nlambda.ext</code>	If set to a value larger than <code>nlambda</code> , this will be the number of values for <code>lambda</code> obtained by extending the range assigned by <code>glmnet</code> symmetrically while keeping the <code>lambda</code> density uniform in log scale. Default is NULL, which will not extend the range of <code>lambda</code> assigned by <code>glmnet</code> .
<code>seed</code>	Sets the pseudo-random number seed to enforce reproducibility. Default is NULL.
<code>scaled</code>	Z-score transformation of individual features across all instances. Default is TRUE.
<code>n_fold</code>	Number of cross-validation folds per run. <code>lambda</code> is chosen based on the maximization of a quality function on out-of-bag-instances averaged over all runs. Default is 5.
<code>n_run</code>	Number of runs; for each run, instances are randomly assigned to cross-validation folds. Default is 100.
<code>n_perm_null</code>	Number of random null-model permutations of the response per run. Default is 25.
<code>QF.FUN</code>	User-defined quality function as maximization criterion to select <code>lambda</code> based on response vs out-of-bag predicted instances. For <code>family="gaussian"</code> , default is correlation; for <code>family="binomial"</code> , it is accuracy; for <code>family="multinomial"</code> , it is average accuracy.
<code>QF_label</code>	Label for user-defined quality function, if <code>QF.FUN</code> is provided.
<code>cor_method</code>	For <code>family="gaussian"</code> , correlation method to be used in the default quality function <code>cor.test</code> . Default is "pearson".
<code>fold_distrib_fail.max</code>	For categorical models, maximum number of failed attempts per run to have all classes represented in each in-bag fold. If this number is exceeded, the execution is halted; try again with larger <code>n_fold</code> , by removing/reassigning classes of small size, and/or with larger <code>fold_distrib_fail.max</code> . Default is 100.
<code>...</code>	Accepts parameters from <code>glmnet.control(...)</code> to allow changes of factory default parameters in <code>glmnet</code> . If not explicitly set, it will use factory defaults.

Details

For each `alpha`, a set of `nlambda` values is obtained using the full data; if provided, `nlambda.ext` allows to extend the range of `lambda` values symmetrically while keeping its density uniform in log scale. Using these values of `lambda`, elastic net cross-validation models are generated for `n_run` random assignments of instances among `n_fold` folds; the best `lambda` is determined by the maximization of a quality function that compares out-of-bag predictions against the response. User-defined quality functions can be provided via `QF.FUN`, otherwise sensible defaults are used (e.g. correlation for gaussian models). For each run, using the same assignment of instances into folds, `n_perm_null` null models are generated by shuffling the response. By using the quality function to compare the out-of-bag performance of the model to that of the null models, an empirical significance p-value is assigned to the model. Similar procedures allow to obtain p-values for individual features based on absolute coefficient magnitude and on the frequency of non-zero coefficients. A family of elastic net models is thus generated for multiple values of `alpha` spanning the range from

ridge ($\alpha=0$) to lasso ($\alpha=1$). This function returns an eNetXplorer object on which summary, plotting and export functions in this package can be applied for further analysis. For details about the underlying elastic net models, please refer to the glmnet package and references therein.

Value

An object with S3 class "eNetXplorer".

predictor	Predictor matrix used for regression (in sparse matrix format).
response	Response variable used for regression.
family	Input parameter.
alpha	Input parameter.
nlambda	Input parameter.
nlambda.ext	Input parameter.
seed	Input parameter.
scaled	Input parameter.
n_fold	Input parameter.
n_run	Input parameter.
n_perm_null	Input parameter.
QF_label	Input parameter.
cor_method	Input parameter.
fold_distrib_fail.max	Input parameter.
instance	Instance labels.
feature	Feature labels.
glmnet_params	glmnet parameters used for regression.
best_lambda	lambda values chosen by cross-validation.
model_QF_est	Quality function values obtained by cross-validation.
QF_model_vs_null_pval	P-value from model vs null comparison to assess statistical significance.
lambda_values	List of lambda values used for each alpha.
lambda_QF_est	List of quality function values obtained for each alpha.
predicted_values	List of out-of-bag predicted values for each alpha; rows are instances and columns are median/mad predictions (for linear regression) or class predictions (for binomial and multinomial regression).
feature_coef_wmean	Mean of feature coefficients (over runs) weighted by non-zero frequency (over folds) in sparse matrix format, with features as rows and alpha values as columns. For multinomial regression, it is a list of matrices (one matrix for each class).

- `feature_coef_wsd`
Standard deviation of feature coefficients (over runs) weighted by non-zero frequency (over folds) in sparse matrix format, with features as rows and alpha values as columns. For multinomial regression, it is a list of matrices (one matrix for each class).
- `feature_freq_mean`
Mean of non-zero frequency in sparse matrix format, with features as rows and alpha values as columns. For multinomial regression, it is a list of matrices (one matrix for each class).
- `feature_freq_sd`
Standard deviation of non-zero frequency in sparse matrix format, with features as rows and alpha values as columns. For multinomial regression, it is a list of matrices (one matrix for each class).
- `null_feature_coef_wmean`
Analogous to `feature_coef_wmean` for null model permutations.
- `null_feature_coef_wsd`
Analogous to `feature_coef_wsd` for null model permutations.
- `null_feature_freq_mean`
Analogous to `feature_freq_mean` for null model permutations.
- `null_feature_freq_sd`
Analogous to `feature_freq_sd` for null model permutations.
- `feature_coef_model_vs_null_pval`
P-value from model vs null comparison to assess statistical significance of mean non-zero feature coefficients in sparse matrix format, with features as rows and alpha values as columns. For multinomial regression, it is a list of matrices (one matrix for each class).
- `feature_freq_model_vs_null_pval`
P-value from model vs null comparison to assess statistical significance of mean non-zero feature frequencies in sparse matrix format, with features as rows and alpha values as columns. For multinomial regression, it is a list of matrices (one matrix for each class).

Author(s)

Julian Candia and John S. Tsang
Maintainer: Julian Candia <julian.candia@nih.gov>

References

Candia J and Tsang JS (2018) *eNetXplorer: an R package for the quantitative exploration of elastic net families for generalized linear models*, bioRxiv 305870 (under review).

See Also

[summary](#), [plot](#), [summaryPDF](#), [export](#)

Examples

```
# Linear models (synthetic dataset comprised of 20 features and 75 instances):
data(QuickStartEx)
fit = eNetXplorer(x=QuickStartEx$predictor, y=QuickStartEx$response,
family="gaussian", n_run=20, n_perm_null=10, seed=111)

# Linear models to predict numerical day-70 H1N1 serum titers based on
# day-7 cell population frequencies:
data(H1N1_Flow)
fit = eNetXplorer(x=H1N1_Flow$predictor_day7, y=H1N1_Flow$response_numer[rownames(
H1N1_Flow$predictor_day7)], family="gaussian", n_run=25, n_perm_null=15, seed=111)

# Binomial models to predict acute myeloid (AML) vs acute lymphoblastic (ALL)
# leukemias:
data(Leukemia_miR)
fit = eNetXplorer(x=Leukemia_miR$predictor, y=Leukemia_miR$response_binomial,
family="binomial", n_run=25, n_perm_null=15, seed=111)

# Multinomial models to predict acute myeloid (AML), acute B-cell lymphoblastic
# (B-ALL) and acute T-cell lymphoblastic (T-ALL) leukemias:
data(Leukemia_miR)
fit = eNetXplorer(x=Leukemia_miR$predictor, y=Leukemia_miR$response_multinomial,
family="multinomial", n_run=25, n_perm_null=15, seed=111)

# Binomial models to predict B-ALL vs T-ALL:
data(Leukemia_miR)
fit = eNetXplorer(x=Leukemia_miR$predictor[Leukemia_miR$response_multinomial!="AML",],
y=Leukemia_miR$response_multinomial[Leukemia_miR$response_multinomial!="AML"],
family="binomial", n_run=25, n_perm_null=15, seed=111)
```

export

generates plain text files from eNetXplorer object

Description

This function enables the extraction of three different levels of data (input, summary, and detailed output results) from an eNetXplorer object. Plain text data files are produced with tab- or comma-separated-value formats.

Usage

```
export(x, path, delim=c("tab", "csv"), input.data=T,
summary.data=T, output.data=T)
```

Arguments

x	eNetXplorer object.
path	Destination file path.
delim	Delimiter for the generated files, either tab- or comma-separated values. Default is tab.
input.data	Logical variable indicating whether to generate files containing input data (i.e. data fed into the models and model arguments). Default is TRUE.
summary.data	Logical variable indicating whether to generate a file with summary results from the models. Default is TRUE.
output.data	Logical variable indicating whether to generate files with detailed results from the models. Default is TRUE.

Author(s)

Julian Candia and John S. Tsang
Maintainer: Julian Candia <julian.candia@nih.gov>

See Also

[eNetXplorer](#)

Examples

```
data(QuickStartEx)
fit = eNetXplorer(x=QuickStartEx$predictor,y=QuickStartEx$response,
family="gaussian",n_run=20,n_perm_null=10,seed=111)
export(x=fit,path=tempdir())
```

H1N1_Flow	<i>Longitudinal cell population frequencies and titer response upon H1N1 vaccination</i>
-----------	--

Description

Data from a cohort of healthy subjects vaccinated against influenza virus H1N1. Cell population frequencies from deep-phenotyping flow cytometry were determined longitudinally pre- (days -7, 0) and post-vaccination (days 1, 7, 70). The response is the adjusted maximum fold change (adjMFC) of serum titers at day 70 relative to baseline, as defined in Tsang et al (2014).

Usage

```
data(H1N1_Flow)
```


Format

For each timepoint (days -7, 0, 1, 7, 70), a numerical matrix of predictors is provided with subjects as rows and cell populations as columns. Two versions of the serum titer response are given: response_numer as a numerical vector and response_class as a categorical vector discretized into low (0), intermediate (1) and high (2) response classes. A metadata file with cell population annotations is also provided.

Details

Cell populations were manually gated and expressed as percent of parent. Samples and cell populations were filtered independently for each timepoint. Samples filter: excluded if median of viable cells fraction across all 5 tubes was <0.7. Cell population filter: excluded if >80% of samples had <20 cells. Data adjustment: data were log10-transformed and pooled across all timepoints, then adjusted for age, gender and ethnicity effects. For more details, see Tsang et al (2014).

References

Tsang JS et al (2014) *Global Analyses of Human Immune Variation Reveal Baseline Predictors of Postvaccination Responses*, Cell 157: 499-513.

Leukemia_miR

MicroRNA expression of acute leukemia phenotypes

Description

Data of human microRNA (miR) expression from acute myeloid (AML) and acute lymphoblastic (ALL) leukemia cell lines and primary (patient) samples. Acute lymphoblastic leukemia samples are further classified by B-cell (B-ALL) and T-cell (T-ALL) subphenotypes.

Usage

```
data(Leukemia_miR)
```

Format

A numerical matrix of predictors is provided with samples as rows and miRs as columns. Two categorical response vectors are provided for binomial (AML, ALL) and multinomial (AML, B-ALL, T-ALL) classification.

Details

An initial set of 847 human miRs was measured in multiple acute leukemia cell lines and primary samples. Exclusion criteria based on miR expression with mean intensity <1 or >14 (after RMA normalization and log2-transformation) led to a final set of 370 miRs. For more details, see Candia et al (2015).

References

Candia J et al (2015) *Uncovering low-dimensional, miR-based signatures of acute myeloid and lymphoblastic leukemias with a machine-learning-driven network approach*, *Converg Sci Phys Oncol* 1: 025002.

plot *generates plots from eNetXplorer object*

Description

This function is a wrapper for a variety of plots, namely:

summary: model performance across alpha (to assess the relative performance among different member models in the elastic net family, as well as in relation to permutation null models);

lambdaVsQF: given alpha, quality function across lambda (to examine the selection of the optimal penalty parameter);

measuredVsOOB: given alpha, response vs out-of-bag predictions across instances (to assess individual instances, examine outliers, etc);

contingency: (for categorical models) given alpha, response vs out-of-bag predictions across classes;

featureCaterpillar: given alpha, caterpillar plot of feature statistics compared to permutation null models (with statistical significance annotations for individual features); and

featureHeatmap: heatmap of feature statistics across alpha (including statistical significance annotations for individual features).

Usage

```
## S3 method for class 'eNetXplorer'
plot(x, plot.type=c("summary", "lambdaVsQF", "measuredVsOOB", "contingency",
"featureCaterpillar", "featureHeatmap"), alpha.index, stat=c("freq", "coef"), ...)
```

Arguments

x	eNetXplorer object.
plot.type	Type of plot to be produced. Available plots are "summary", "lambdaVsQF", "measuredVsOOB", "contingency" (categorical models only), "featureCaterpillar" and "featureHeatmap".
alpha.index	Integer indices to select alpha values. Default is 1:length(alpha)
stat	Feature statistic: "freq" for nonzero frequency, "coef" for mean nonzero coefficient. Used for plot types "featureHeatmap" and "featureCaterpillar", ignored otherwise.
...	Additional plotting parameters.

Author(s)

Julian Candia and John S. Tsang
 Maintainer: Julian Candia <julian.candia@nih.gov>

See Also

[eNetXplorer](#), [plotSummary](#), [plotLambdaVsQF](#), [plotMeasuredVsOOB](#), [plotContingency](#), [plotFeatureCaterpillar](#), [plotFeatureHeatmap](#)

Examples

```
data(QuickStartEx)
fit = eNetXplorer(x=QuickStartEx$predictor, y=QuickStartEx$response,
family="gaussian", n_run=20, n_perm_null=10, seed=111)
suppressWarnings(plot(x=fit, plot.type="summary"))
plot(x=fit, plot.type="lambdaVsQF", alpha.index=2)
plot(x=fit, plot.type="measuredVsOOB", alpha.index=c(1,3,5))
plot(x=fit, plot.type="featureCaterpillar", stat="coef")
plot(x=fit, plot.type="featureHeatmap", stat="freq")
```

plotContingency *generates plot of response vs out-of-bag predictions across classes*

Description

For categorical models, this function generates a graphical representation of the true vs predicted contingency matrix across classes for a given alpha.

Usage

```
plotContingency(x, alpha.index=NULL, xlab="class (true)", ylab="class (predicted)",
cex.lab=0.95, main=NULL, col.main="black", cex.main=0.85, cex.axis=1,
symbol.size.inches=0.5, bg.color="steelblue2", fg.color=NULL, margin=0.2,
frequency.label=T, frequency.label.cex=1, frequency.label.offset=0, ...)
```

Arguments

x	eNetXplorer object.
alpha.index	Integer indices to select alpha values. Default is 1:length(alpha)
xlab	Custom x-axis label.
ylab	Custom y-axis label.
cex.lab	Axis label size.
main	Custom title.
col.main	Title color.

cex.main	Title size.
cex.axis	Axis size.
symbol.size.inches	Symbol size.
bg.color	Symbol color.
fg.color	Color of symbol background.
margin	Margin size to accomodate symbols.
frequency.label	Logical to display class frequency labels. Default is TRUE.
frequency.label.cex	Size of class frequency labels.
frequency.label.offset	Offset of class frequency labels.
...	Additional plotting parameters.

Author(s)

Julian Candia and John S. Tsang
 Maintainer: Julian Candia <julian.candia@nih.gov>

See Also

[eNetXplorer](#), [plot](#)

Examples

```
data(QuickStartEx)
binarized=rep("low",length(QuickStartEx$response))
binarized[QuickStartEx$response>median(QuickStartEx$response)]= "high"
fit = eNetXplorer(x=QuickStartEx$predictor,y=binarized,family="binomial",n_run=20,
n_perm_null=10,seed=111)
plot(x=fit,plot.type="contingency")
plotContingency(x=fit,alpha.index=6)
```

plotFeatureCaterpillar

generates caterpillar plot of feature statistics

Description

Given alpha, this function generates a caterpillar plot of feature statistics compared to permutation null models, which includes statistical significance annotations for individual features. By default, features are selected (and ordered top-down) by statistical significance; options are provided to customize feature selection and display.

Usage

```
plotFeatureCaterpillar(x, alpha.index=NULL, stat=c("freq","coef"), feature.all=F,
  feature.pval.thres=NULL, feature.set=NULL, feature.top.n=25, signif.code=T,
  xlab=NULL, ylab=NULL, main=NULL, col.main="black", cex.main=0.85, line=1.5,
  subtitle=NULL, col.subtitle="darkgray", line.subtitle=0.5, cex.subtitle=0.55,
  cexRow=NULL, cex.lab=0.95, legend=T, ...)
```

Arguments

<code>x</code>	eNetXplorer object.
<code>alpha.index</code>	Integer indices to select alpha values. Default is <code>1:length(alpha)</code>
<code>stat</code>	Feature statistic: "freq" for nonzero frequency, "coef" for mean nonzero coefficient.
<code>feature.all</code>	(Feature selection argument 1) Logical to show all features. Default is FALSE.
<code>feature.pval.thres</code>	(Feature selection argument 2) P-value threshold to select features. Default is NULL.
<code>feature.set</code>	(Feature selection argument 3) Character vector of feature names to display. Default is NULL.
<code>feature.top.n</code>	(Feature selection argument 4) Number of top features (ordered by p-value) to display. Default is 25.
<code>signif.code</code>	Logical to display significance annotations. Default is TRUE.
<code>xlab</code>	Label for x axis.
<code>ylab</code>	Label for y axis.
<code>main</code>	Custom title.
<code>col.main</code>	Title color.
<code>cex.main</code>	Title size.
<code>line</code>	Title location.
<code>subtitle</code>	Custom subtitle.
<code>col.subtitle</code>	Subtitle color.
<code>line.subtitle</code>	Subtitle location.
<code>cex.subtitle</code>	Subtitle size.
<code>cexRow</code>	Size of row labels.
<code>cex.lab</code>	Axis label size.
<code>legend</code>	Logical to display legend. Default is TRUE.
<code>...</code>	Additional plotting parameters.

Details

Feature selection criteria are hierarchical based on arguments 1 through 4 (see argument description above), with argument 1 at the top of the hierarchy. E.g. if `feature.all` is explicitly set to TRUE, it will take precedence over any other feature selection argument. By default, the top 25 features are displayed, ordered top-down by significance based on the given value of alpha.

Author(s)

Julian Candia and John S. Tsang
 Maintainer: Julian Candia <julian.candia@nih.gov>

See Also

[eNetXplorer](#), [plot](#)

Examples

```
data(QuickStartEx)
fit = eNetXplorer(x=QuickStartEx$predictor, y=QuickStartEx$response,
family="gaussian", n_run=20, n_perm_null=10, seed=111)
plot(x=fit,plot.type="featureCaterpillar",stat="coef")
plotFeatureCaterpillar(x=fit,alpha.index=3,stat="coef",main="custom title")
```

plotFeatureHeatmap *generates heatmap plot of feature statistics*

Description

This function generates a heatmap plot of feature statistics across alpha, which includes statistical significance annotations for individual features. By default, features are selected (and ordered top-down) by statistical significance based on a given value of alpha; options are provided to customize feature selection and display.

Usage

```
plotFeatureHeatmap(x, alpha.index=NULL, stat=c("freq","coef"), feature.all,
feature.pval.thres, feature.set, feature.top.n, signif.code, xlab=NULL,
ylab=NULL, main=NULL, col.main="black", cex.main=0.95, line=1, col=NULL,
breaks=NULL, scale="none", Rowv=F, Colv=F, na.color=NULL, cexRow=NULL, srtRow=0,
cexCol=0.75, srtCol=45, margins=c(5, 5), key=T, key.title=NA, dendogram="none",
trace="none", notecol.freq="black", notecol.coef="white", notecex=1,
subtitle1=NULL, col.subtitle1="black", line.subtitle1=-1, cex.subtitle1=0.65,
subtitle2=NULL, col.subtitle2="darkgray", line.subtitle2=-2, cex.subtitle2=0.55,
...)
```

Arguments

x	eNetXplorer object.
alpha.index	Integer indices to select alpha values. Default is 1:length(alpha)
stat	Feature statistic: "freq" for nonzero frequency, "coef" for mean nonzero coefficient.

feature.all	(Feature selection argument 1) Logical to show all features. Default is FALSE.
feature.pval.thres	(Feature selection argument 2) P-value threshold to select features. Default is NULL.
feature.set	(Feature selection argument 3) Character vector of feature names to display. Default is NULL.
feature.top.n	(Feature selection argument 4) Number of top features (ordered by p-value) to display. Default is 25.
signif.code	Logical to display statistical significance annotations. Default is TRUE.
xlab	Label for x axis.
ylab	Label for y axis.
main	Custom title.
col.main	Title color.
cex.main	Title size.
line	Title location.
col	Heatmap color vector. Length must be one less than number of breaks.
breaks	Color breaks vector. Default number of breaks is 10.
scale	Logical to scale the data for heatmap in either the row or column direction. Default is "none".
Rowv	Logical to reorder rows by hierarchical clustering. Default is FALSE.
Colv	Logical to reorder columns by hierarchical clustering. Default is FALSE.
na.color	Color to use for missing values.
cexRow	Size of row labels.
srtRow	Angle of row labels, in degrees from horizontal.
cexCol	Size of column labels.
srtCol	Angle of column labels, in degrees from horizontal.
margins	Numeric vector of length 2 containing the margins for column and row names, respectively.
key	Logical to display key. Default is TRUE.
key.title	Main title of the color key.
dendogram	To draw dendograms. Default is "none".
trace	To display trace lines. Default is "none".
notecol.freq	Color of statistical significance annotations for feature frequency heatmaps.
notecol.coef	Color of statistical significance annotations for feature coefficient heatmaps.
notecex	Size of significance annotations.
subtitle1	Custom subtitle 1.
col.subtitle1	Color of subtitle 1.
line.subtitle1	Position of subtitle 1.
cex.subtitle1	Size of subtitle 1.

subtitle2 Custom subtitle 2.
 col.subtitle2 Color of subtitle 2.
 line.subtitle2 Position of subtitle 2.
 cex.subtitle2 Size of subtitle 2.
 ... Additional plotting parameters.

Details

Feature selection criteria are hierarchical based on arguments 1 through 4 (see argument description above), with argument 1 at the top of the hierarchy. E.g. if `feature.all` is explicitly set to `TRUE`, it will take precedence over any other feature selection argument. By default, the top 25 features are displayed, ordered top-down by significance based on the given value of `alpha`.

Author(s)

Julian Candia and John S. Tsang
 Maintainer: Julian Candia <julian.candia@nih.gov>

See Also

[eNetXplorer](#), [plot](#)

Examples

```

data(QuickStartEx)
fit = eNetXplorer(x=QuickStartEx$predictor,y=QuickStartEx$response,
family="gaussian",n_run=20,n_perm_null=10,seed=111)
plot(x=fit,plot.type="featureHeatmap",stat="freq")
plotFeatureHeatmap(x=fit,alpha.index=3,stat="freq",main="custom title")

```

`plotLambdaVsQF` *generates plot of quality function across lambda*

Description

Given `alpha`, this function generates a plot of the quality function across `lambda`, which is useful to examine the selection of the best penalty parameter.

Usage

```

plotLambdaVsQF(x, alpha.index=NULL, xlab="lambda",
ylab="QF (response vs out-of-bag predicted)", cex.lab=0.95, main=NULL,
col.main="black", cex.main=0.95, log="x", type="b", ...)

```


Arguments

x	eNetXplorer object.
alpha.index	Integer indices to select alpha values. Default is 1:length(alpha)
xlab	Custom x-axis label.
ylab	Custom y-axis label.
cex.lab	Axis label size.
main	Custom title.
col.main	Title color.
cex.main	Title size.
log	Log scale axis.
type	Plot type.
...	Additional plotting parameters.

Author(s)

Julian Candia and John S. Tsang
 Maintainer: Julian Candia <julian.candia@nih.gov>

See Also

[eNetXplorer](#), [plot](#)

Examples

```
data(QuickStartEx)
fit = eNetXplorer(x=QuickStartEx$predictor,y=QuickStartEx$response,
family="gaussian",n_run=20,n_perm_null=10,seed=111)
plot(x=fit,plot.type="lambdaVsQF")
plotLambdaVsQF(x=fit,alpha.index=c(1,3),main="custom title",col.main="red")
```

plotMeasuredVsOOB *generates plot of response vs out-of-bag predictions across instances*

Description

Given alpha, this function generates plots of response vs out-of-bag predictions across instances, which can be used to assess individual instances, examine outliers, etc. For linear regression models, it generates a response vs out-of-bag prediction scatterplot; it also displays the best linear fit and its 95% confidence level region. For categorical models, it generates an out-of-bag prediction accuracy boxplot per class.

Usage

```
plotMeasuredVsOOB(x, alpha.index=NULL, xlab=NULL, ylab=NULL,
  cex.lab=0.95, main=NULL, col.main="black", cex.main=0.85, instance.label=T,
  instance.label.cex=NULL, instance.label.offset=NULL, instance.label.added.margin=NULL,
  col=NULL, transparency=NULL, jitter=NULL, cex.pt=NULL, class.color=NULL, ...)
```

Arguments

<code>x</code>	eNetXplorer object.
<code>alpha.index</code>	Integer indices to select alpha values. Default is <code>1:length(alpha)</code>
<code>xlab</code>	Custom x-axis label.
<code>ylab</code>	Custom y-axis label.
<code>cex.lab</code>	Axis label size.
<code>main</code>	Custom title.
<code>col.main</code>	Title color.
<code>cex.main</code>	Title size.
<code>instance.label</code>	Logical to display instance labels. Default is TRUE.
<code>instance.label.cex</code>	Size of instance labels.
<code>instance.label.offset</code>	Offset of instance labels.
<code>instance.label.added.margin</code>	(linear regression only) Margin size to accomodate instance label display.
<code>col</code>	(linear regression only) Symbol color.
<code>transparency</code>	(categorical models only) Symbol transparency.
<code>jitter</code>	(categorical models only) Symbol jitter.
<code>cex.pt</code>	(categorical models only) Symbol size.
<code>class.color</code>	(categorical models only) Vector of class colors.
<code>...</code>	Additional plotting parameters.

Author(s)

Julian Candia and John S. Tsang
 Maintainer: Julian Candia <julian.candia@nih.gov>

See Also

[eNetXplorer](#), [plot](#)

Examples

```
data(QuickStartEx)
fit = eNetXplorer(x=QuickStartEx$predictor,y=QuickStartEx$response,
family="gaussian",n_run=20,n_perm_null=10,seed=111)
plot(x=fit,plot.type="measuredVs00B")
plotMeasuredVs00B(x=fit,alpha.index=2)
```

```
data(QuickStartEx)
binarized=rep("low",length(QuickStartEx$response))
binarized[QuickStartEx$response>median(QuickStartEx$response)]= "high"
fit = eNetXplorer(x=QuickStartEx$predictor,y=binarized,family="binomial",n_run=20,
n_perm_null=10,seed=111)
plot(x=fit,plot.type="measuredVs00B")
plotMeasuredVs00B(x=fit,alpha.index=2)
```

plotSummary

generates summary plots of model performance across alpha

Description

This function generates summary plots to display the performance of all models in the elastic net family. Two measures are used: 1) mean quality function of response vs out-of-bag predictions, and 2) model vs null p-values. Taken together, these plots enable visual assessments of the relative performance among different member models in the elastic net family, as well as in relation to permutation null models.

Usage

```
plotSummary(x, show.pval.ref = T, main = NULL, col.main = "black",
cex.main = 0.95, line = 1, ...)
```

Arguments

x	eNetXplorer object.
show.pval.ref	Logical to display reference lines of significance (if within range of model vs null p-values). Default is TRUE.
main	Custom title.
col.main	Title color.
cex.main	Title size.
line	Title position.
...	Additional plotting parameters.

Author(s)

Julian Candia and John S. Tsang
Maintainer: Julian Candia <julian.candia@nih.gov>

See Also

[eNetXplorer](#), [plot](#)

Examples

```
data(QuickStartEx)
fit = eNetXplorer(x=QuickStartEx$predictor,y=QuickStartEx$response,
family="gaussian",n_run=20,n_perm_null=10,seed=111)
suppressWarnings(plot(x=fit, plot.type="summary"))
suppressWarnings(plotSummary(x=fit,show.pval.ref=FALSE))
```

QuickStartEx

Synthetic dataset

Description

75 instances with 20 predictors and a numerical response to be used as a quick start example.

Usage

```
data(QuickStartEx)
```

Format

A numerical matrix of predictors is provided with instances as rows and predictors as columns. A numerical response is provided as a quick start example for linear regression models; it can be easily discretized to serve as example for binary and multinomial models as well.

summary	<i>generates list of model statistics</i>
---------	---

Description

This function generates a standard list of model statistics. For each alpha, it contains the best value of lambda (obtained by maximizing a quality function over out-of-bag instances), the corresponding maximum value of the quality function, and the model significance (p-value based on comparison to permutation null models).

Usage

```
## S3 method for class 'eNetXplorer'  
summary(object, ...)
```

Arguments

object	eNetXplorer object.
...	Additional parameters.

Value

alpha	Vector of alpha values.
best_lambda	Best lambda obtained by maximization of the quality function.
model_QF_est	Maximum of the quality function.
QF_model_vs_null_pval	P-value from model vs null comparison to assess statistical significance.

Author(s)

Julian Candia and John S. Tsang
Maintainer: Julian Candia <julian.candia@nih.gov>

See Also

[eNetXplorer](#)

Examples

```
data(QuickStartEx)  
fit = eNetXplorer(x=QuickStartEx$predictor,y=QuickStartEx$response,  
family="gaussian",n_run=20,n_perm_null=10,seed=111)  
summary(fit)
```

`summaryPDF`*generates PDF report with summary of main results*

Description

This function generates a PDF report that contains a plot of model performance across the alpha range, followed by plots showing detailed results for each value of alpha.

Usage

```
summaryPDF(x, path, filename="eNetXplorerSummary.pdf")
```

Arguments

<code>x</code>	eNetXplorer object.
<code>path</code>	Directory path for output PDF file.
<code>filename</code>	Name for output PDF file.

Author(s)

Julian Candia and John S. Tsang
Maintainer: Julian Candia <julian.candia@nih.gov>

See Also

[eNetXplorer](#), [plot](#)

Examples

```
data(QuickStartEx)
fit = eNetXplorer(x=QuickStartEx$predictor,y=QuickStartEx$response,
family="gaussian",n_run=20,n_perm_null=10,seed=111)
suppressWarnings(summaryPDF(x=fit,path=tempdir()))
```

Index

*Topic **datasets**

H1N1_Flow, [8](#)

Leukemia_miR, [9](#)

QuickStartEx, [20](#)

*Topic **package**

eNetXplorer-package, [2](#)

eNetXplorer, [2](#), [3](#), [8](#), [11](#), [12](#), [14](#), [16–18](#), [20–22](#)

eNetXplorer-package, [2](#)

export, [2](#), [6](#), [7](#)

H1N1_Flow, [8](#)

Leukemia_miR, [9](#)

plot, [2](#), [6](#), [10](#), [12](#), [14](#), [16–18](#), [20](#), [22](#)

plotContingency, [11](#), [11](#)

plotFeatureCaterpillar, [11](#), [12](#)

plotFeatureHeatmap, [11](#), [14](#)

plotLambdaVsQF, [11](#), [16](#)

plotMeasuredVsO0B, [11](#), [17](#)

plotSummary, [11](#), [19](#)

QuickStartEx, [20](#)

summary, [2](#), [6](#), [21](#)

summaryPDF, [2](#), [6](#), [22](#)