

Package ‘IntegMultiReg’

August 9, 2026

Title Integrative Bayesian Multiple Regression for Multi-Platform Biomarkers

Version 0.1.0

Description A Bayesian framework that integrates several regression models to identify a parsimonious set of biomarkers shared across disparate data platforms (for example genomic, transcriptomic and proteomic assays). Subjects are partitioned into subgroups defined by their pattern of platform availability, so that no subject with partially missing platform data is excluded, and information is borrowed across subgroups through a Markov random field prior on the variable-selection indicators together with non-local (product moment) priors on the regression effects. The methodology was introduced for time-to-event outcomes by Chekouo, Stingo, Doecke and Do (2017) <[doi:10.1111/biom.12587](https://doi.org/10.1111/biom.12587)>; this package additionally supports continuous (Gaussian) and binary (probit) outcomes. Posterior inference is carried out by a Markov chain Monte Carlo sampler implemented in C for computational efficiency.

License GPL (>= 3)

Encoding UTF-8

LazyData true

Depends R (>= 4.0.0)

Imports graphics, grDevices, stats, utils

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

SystemRequirements GSL (>= 2.0)

VignetteBuilder knitr

Config/testthat/edition 3

NeedsCompilation yes

Config/roxygen2/version 8.0.0

RoxygenNote 7.3.2

Author Sinian Zhang [aut],
Jianfeng Wang [aut],
Thierry Chekouo [aut, cre] (ORCID:
<<https://orcid.org/0000-0002-3594-9182>>)

Maintainer Thierry Chekouo <tchekouo@umn.edu>
Repository CRAN
Date/Publication 2026-08-09 08:20:07 UTC

Contents

coef.imr	2
cv_imr	3
imr	4
kircIMR	7
plot.imr	8
plot_subgroup_sizes	10
plot_top_features	12
predict.imr	13
print.imr	15
simIMR	16
summary.imr	17
Index	18

coef.imr	<i>Marginal Posterior Inclusion Probabilities of an IMR Fit</i>
----------	---

Description

Extracts the posterior mean variable-selection probabilities (the marginal posterior inclusion probabilities, mPIP) of a fitted model.

Usage

```
## S3 method for class 'imr'  
coef(object, ...)  
  
coef_imr(object, ...)
```

Arguments

object	A fitted object of class <code>"imr"</code> .
...	Unused; present for S3 compatibility.

Value

A named list with one matrix per platform. Rows are the subgroups containing that platform (labelled by their availability bitstrings) and columns are the platform features.

See Also

[imr()]

Description

‘cv_imr()’ evaluates predictive accuracy of a model fitted with [imr()] using repeated K -fold splits and Bayesian model averaging over the fitted MCMC samples. The sampler is not re-run inside each training fold; fit a separate object with ‘method = "BMS"’ if a no-borrowing comparison is needed. The accuracy measure depends on the outcome type: the concordance index (C-index) for right-censored outcomes, the area under the ROC curve (AUC) for binary outcomes, and the mean squared error (MSE) for continuous outcomes.

Usage

```
cv_imr(
  object,
  k = 5,
  rounds = 2,
  method = NULL,
  max_models = 100,
  verbose = FALSE
)
```

Arguments

object	A fitted object of class “imr” returned by [imr()].
k	Integer number of cross-validation folds per round (default ‘5’). Must be at least ‘2’, and each availability subgroup must contain at least ‘k’ subjects.
rounds	Integer number of independent cross-validation rounds to average over (default ‘2’). Must be positive.
method	Optional compatibility argument. If supplied, it must match the method stored in ‘object’; ‘cv_imr()’ cannot turn an IMR fit into a BMS fit or vice versa.
max_models	Integer maximum number of selection models used for Bayesian model averaging (default ‘100’). Must be positive.
verbose	Logical; if ‘TRUE’, print the C routine’s cross-validation diagnostics. Defaults to ‘FALSE’.

Value

A list with two numeric matrices, each of dimension ‘rounds’ x ‘(n_subgroups + 1)’, whose last column corresponds to all subjects pooled and whose remaining columns are named by the availability subgroup bitstrings:

total_cindex	Accuracy computed on the pooled cross-validated predictions within each availability subgroup (and overall).
subset_cindex	Accuracy computed fold-by-fold within each availability subgroup.

The accuracy metric ("C-index", "AUC" or "MSE") is recorded in the "metric" attribute of the returned list.

See Also

[imr()], [predict.imr()]

Examples

```
data("simIMR", package = "IntegMultiReg")
fit <- imr(
  platform_data_list = simIMR$platforms, outcome = simIMR$outcome,
  cov = simIMR$covariates, type_outcome = "binary",
  nu = c(-4, -3, -4), sample_mcmc = c(200, 100), ssize = 5, seed = 1
)
cv <- cv_imr(fit, k = 5, rounds = 2)
cv$total_cindex
```

imr

Fit the Integrative Bayesian Multi-Platform Regression Model

Description

'imr()' fits the integrative multi-regression (IMR) model of Chekouo et al. (2017) by Markov chain Monte Carlo (MCMC). It identifies biomarkers associated with a time-to-event, binary or continuous outcome while borrowing information across all subjects, regardless of which platforms each subject has measured. Subjects are partitioned into availability subgroups defined by their pattern of platform availability; one regression is built per subgroup and information is shared across subgroups through a Markov random field (MRF) prior on the variable-selection indicators together with non-local priors on the regression coefficients. The sampler is implemented in C for efficiency.

The arguments are grouped by the orthogonal aspect of the analysis that each one controls: the **data** ('platform_data_list', 'outcome', 'cov'), the **likelihood** ('type_outcome'), the **model** ('method'), subgroup **filtering** ('ssize'), the **priors** ('nu', 'hh', 'h0', 'sig_alpha_psi', 'thet_alph_bet'), the **computation** ('sample_mcmc', 'seed') and the **output** ('verbose').

Usage

```
imr(
  platform_data_list,
  outcome,
  cov = NULL,
  type_outcome = c("right.censored", "binary", "continuous"),
  method = c("IMR", "BMS"),
  ssize = 30,
  nu = rep(-3, length(platform_data_list)),
  hh = 0.087,
  h0 = 10000,
```

```

sig_alpha_psi = c(0.001, 0.001),
thet_alph_bet = c(40, 10),
sample_mcmc = c(2000, 1000),
seed = NULL,
verbose = FALSE
)

```

Arguments

platform_data_list	A list of data frames, one per platform. Each data frame must contain an ‘id’ column (the subject identifier, taken to be the first column); the remaining columns are finite numeric features measured on that platform. Subject identifiers must be unique within each data frame. The ‘id’ column links subjects across platforms and to the outcome and covariate data.
outcome	A data frame containing an ‘id’ column and the response. For ‘type_outcome = "right.censored"’ it must also contain the event/censoring time and a censoring indicator (three columns in total). For “binary” the response must be coded 0/1 and for “continuous” it is a numeric response (two columns in total). The ‘id’ column must be first and unique.
cov	An optional data frame of clinical covariates including an ‘id’ column followed by finite numeric covariates. Covariates are always included in every regression (they are not subject to selection). Defaults to ‘NULL’ (no covariates).
type_outcome	Character string specifying the outcome type, one of “right.censored” (default), “binary” or “continuous”.
method	Character string specifying the method, “IMR” (default) for the integrative model that shares information across subgroups via the MRF prior, or “BMS” for the non-integrative Bayesian multi-step variant that fits each subgroup independently (MRF interaction parameters set to zero).
ssize	Minimum availability subgroup size for a subgroup to be modelled. Subgroups with at most ‘ssize’ subjects are dropped. Default is ‘30’.
nu	A numeric vector of prior log-odds of inclusion, one value per platform, controlling the prior sparsity of the selected features. Defaults to ‘rep(-3, length(platform_data_list))’.
hh	Scale of the non-local (product moment) prior on the slab regression effects (default ‘0.087’).
h0	Prior variance of the intercept (default ‘10000’).
sig_alpha_psi	Length-2 numeric vector with the shape and rate of the inverse-gamma prior on the response error variance (default ‘c(0.001, 0.001)’). Ignored when ‘type_outcome = "binary"’: a probit model has residual variance fixed at 1 for identifiability, so the error variance is pinned to 1 rather than estimated.
thet_alph_bet	Length-2 numeric vector with the shape and rate of the gamma prior on the MRF interaction parameters theta, which borrow information across subgroups (default ‘c(40, 10)’).
sample_mcmc	An integer vector ‘c(n_retained, n_burnin)’ giving the number of post-burn-in draws to retain and the number of burn-in iterations to discard. The sampler runs ‘n_retained + n_burnin’ iterations in total, so the returned ‘log_posterior’ has ‘sum(sample_mcmc)’ entries. Default is ‘c(2000, 1000)’.

seed	Optional integer seed for the sampler. If 'NULL' (the default) a seed is drawn from the current R RNG state, so a run is reproducible whenever [set.seed()] is called beforehand or an explicit 'seed' is passed.
verbose	Logical; if 'TRUE', print the sampler's progress and diagnostics to the console. Defaults to 'FALSE' (quiet).

Details

All feature and covariate data are standardized internally (mean 0, standard deviation 1); the centring and scaling factors are stored in the returned object so that [predict.imr()] can apply the same transformation to new subjects. For right-censored outcomes the latent log-survival times of censored subjects are imputed within the sampler; for binary outcomes a probit data-augmentation latent variable is sampled.

Value

An object of class "imr": a list with components

gam_mean	List (one matrix per platform) of posterior mean variable-selection probabilities; rows index the subgroups containing the platform and columns index the platform features.
theta_mean	List (one matrix per platform) of posterior mean MRF interaction parameters between subgroups.
estimate_latent_y	List (one vector per subgroup) of posterior mean latent responses, useful for censored or binary data.
log_posterior	Numeric vector of log-posterior values across all MCMC iterations (burn-in included).
gam_sample	Post-burn-in MCMC samples of the selection indicators.
theta_sample	Post-burn-in MCMC samples of the MRF interaction parameters (absent when 'method = "BMS"').
list_hyperpara, data1, data2	Hyper-parameters and pre-processed data retained for prediction and cross-validation.
call, type_outcome, n_platform, platform_names, feature_names, covariate_names, model_bitstrings, sample_size, model_platforms, platform_models, sample_mcmc, nu, method	Run metadata used by the print, summary, plot and predict methods.

References

Chekouo T, Stingo FC, Doecke JD, Do K-A (2017). "A Bayesian Integrative Approach for Multi-Platform Genomic Data: A Kidney Cancer Case Study." *Biometrics*, **73**(2), 615–624. doi:10.1111/biom.12587

See Also

[predict.imr()], [cv_imr()], [summary.imr()], [plot.imr()]

Examples

```
data("simIMR", package = "IntegMultiReg")
fit <- imr(
  platform_data_list = simIMR$platforms,
  outcome = simIMR$outcome,
  cov = simIMR$covariates,
  type_outcome = "binary",
  nu = c(-4, -3, -4),
  sample_mcmc = c(200, 100),
  ssize = 5,
  seed = 1
)
fit
```

kircIMR

Reduced TCGA-KIRC Multi-Platform Survival Example

Description

A reduced real-data example derived from public UCSC Xena TCGA kidney renal clear cell carcinoma (KIRC) data, not from controlled-access TCGA/GDC files. The data set mirrors the multi-platform structure of Chekouni, Stingo, Doecke and Do (2017): mRNA expression, miRNA expression and DNA methylation are observed on overlapping but not identical sets of patients, with clinical covariates and right-censored survival outcome.

Usage

```
kircIMR
```

Format

A named list with the components:

platforms A list of three data frames, ‘mrna’, ‘mirna’ and ‘methylation’. Each has an ‘id’ column containing package-internal patient labels followed by screened molecular features.

covariates Clinical covariates with an ‘id’ column: age at initial diagnosis, female sex indicator, pathologic stage and histologic grade.

outcome Alias of ‘outcome.survival’, for convenience.

outcome.survival Data frame with ‘id’, observed survival/follow-up time in days and event indicator ‘status’ (1 = deceased, 0 = censored).

feature_screening Per-platform feature-ranking information and a short description of the screening procedure used to create the reduced panel.

platform_availability Patient-level logical matrix indicating which of the three platforms are available.

subgroup_sizes Counts of the non-empty availability subgroups induced by the reduced data.

model_subgroup_sizes The availability subgroups that would be modelled under the default real-data illustration threshold 'ssize = 30'.

paper_alignment Metadata describing how the reduced example maps to the Biometrics KIRC case study.

source Source URLs and reference metadata.

Details

The full TCGA source files are not distributed with the package. Instead, this object stores a small screened feature panel suitable for examples and vignette analyses. Patient identifiers are package-internal labels ('KIRC001', 'KIRC002', ...) rather than TCGA barcodes; no barcode mapping is distributed with the package. Users should not attempt participant re-identification or linkage to external resources. The generating script is provided in the source repository under 'data-raw/make_kircIMR.R'.

The source data are public UCSC Xena TCGA-KIRC sampleMap files: mRNA expression ('HiSeqV2'), miRNA expression ('miRNA_HiSeq_gene'), DNA methylation ('HumanMethylation450') and the KIRC clinical matrix. Molecular features are screened with univariable Cox models after basic filtering, and missing molecular values in the retained panel are median-imputed within feature. The full Biometrics analysis used 776 mRNA, 91 miRNA and 729 DNA methylation features; this package example retains 50, 30 and 50 features, respectively, to keep examples lightweight. The package does not distribute controlled-access TCGA/GDC data or the full Biometrics/Wiley supplementary data files.

References

Chekouo T, Stingo FC, Doecke JD, Do K-A (2017). "A Bayesian Integrative Approach for Multi-Platform Genomic Data: A Kidney Cancer Case Study." *Biometrics*, **73**(2), 615–624. doi:10.1111/biom.12587

The Cancer Genome Atlas Research Network; National Cancer Institute Genomic Data Commons; UCSC Xena.

Examples

```
data("kircIMR", package = "IntegMultiReg")
sapply(kircIMR$platforms, dim)
kircIMR$subgroup_sizes
kircIMR$model_subgroup_sizes
```

plot.imr

Plot Method for IMR Fits

Description

Visualizes a fitted '"imr"' object. Three plot types are available:

"selection" Heatmap of the marginal posterior inclusion probabilities (mPIP), one panel per platform, with features on the horizontal axis, availability subgroups on the vertical axis and a small intensity legend showing that darker values are closer to 1.

"theta" Heatmap of the posterior mean MRF interaction parameters between availability subgroups, one panel per platform.

"trace" Trace plot of the log-posterior across MCMC iterations.

See [plot_top_features()] and [plot_subgroup_sizes()] for two further ready made displays.

Usage

```
## S3 method for class 'imr'
plot(
  x,
  type = c("selection", "theta", "trace"),
  platform = NULL,
  base_cex = 1,
  cex_axis = NULL,
  cex_lab = NULL,
  cex_main = NULL,
  col = NULL,
  palette = NULL,
  legend = TRUE,
  legend_width = 0.28,
  mar = NULL,
  mgp = NULL,
  ...
)

plot_imr(x, ...)
```

Arguments

x	A fitted object of class <code>"imr"</code> .
type	Character; one of <code>"selection"</code> (default), <code>"theta"</code> or <code>"trace"</code> .
platform	Optional integer vector selecting which platforms to display for the <code>"selection"</code> and <code>"theta"</code> plots; defaults to all platforms.
base_cex	Overall text-size multiplier. The <code>'cex_*'</code> arguments default to values derived from this multiplier (default <code>'1'</code>).
cex_axis	Axis-label size multiplier. If <code>'NULL'</code> , a plot-specific default derived from <code>'base_cex'</code> is used.
cex_lab	Axis-title size multiplier. If <code>'NULL'</code> , a plot-specific default derived from <code>'base_cex'</code> is used.
cex_main	Main-title size multiplier. If <code>'NULL'</code> , a plot-specific default derived from <code>'base_cex'</code> is used.
col	Optional colours. For heatmaps this is the colour scale; for trace plots this is the line colour.
palette	Optional heatmap palette name used when <code>'col = NULL'</code> . Selection plots default to <code>"grey"</code> to preserve the mPIP intensity scale; theta plots default to the muted grey-blue <code>"heatmap"</code> palette.

legend	Logical; for "selection" plots, should the mPIP intensity legend be drawn (default 'TRUE')?
legend_width	Relative width of the intensity-legend panel for "selection" plots (default '0.28').
mar, mgp	Optional graphical margin and axis-title placement vectors passed to [graphics::par()] for finer layout control.
...	Further graphical parameters passed to the underlying plotting functions.

Value

'NULL', invisibly; called for the side effect of producing a plot.

See Also

[imr()], [plot_top_features()], [plot_subgroup_sizes()]

plot_subgroup_sizes *Plot the Availability Subgroup Sizes of an IMR Fit*

Description

Draws a bar chart of the number of subjects in each modelled availability subgroup (the non-empty regions of the Venn diagram), giving a quick picture of how the sample is distributed across subgroups.

Usage

```
plot_subgroup_sizes(
  object,
  base_cex = 1,
  cex_axis = NULL,
  cex_lab = NULL,
  cex_main = NULL,
  cex_values = NULL,
  col = NULL,
  palette = "subgroup",
  show_values = TRUE,
  ylim = NULL,
  mar = NULL,
  mgp = NULL,
  ...
)
```

Arguments

object	A fitted object of class "imr" returned by [imr()].
base_cex	Overall text-size multiplier. The 'cex_*' arguments default to values derived from this multiplier (default '1').
cex_axis	Axis-label size multiplier. If 'NULL', a plot-specific default derived from 'base_cex' is used.
cex_lab	Axis-title size multiplier. If 'NULL', a plot-specific default derived from 'base_cex' is used.
cex_main	Main-title size multiplier. If 'NULL', a plot-specific default derived from 'base_cex' is used.
cex_values	Bar-value label size multiplier. If 'NULL', a plot-specific default derived from 'base_cex' is used.
col	Optional bar colours. If 'NULL', colours are generated from 'palette'.
palette	Palette name used when 'col = NULL'; "subgroup" gives the package's standard neutral grey-blue subgroup colours (default).
show_values	Logical; should sample sizes be printed above the bars (default 'TRUE')?
ylim	Optional numeric vector of length two giving the vertical axis limits. If 'NULL', limits are chosen from the subgroup sizes.
mar, mgp	Optional graphical margin and axis-title placement vectors passed to [graphics::par()] for finer layout control.
...	Further graphical parameters passed to [graphics::barplot()].

Value

Invisibly, the named integer vector of subgroup sizes.

See Also

[imr()], [plot.imr()], [plot_top_features()]

Examples

```
data("simIMR", package = "IntegMultiReg")
fit <- imr(
  platform_data_list = simIMR$platforms, outcome = simIMR$outcome,
  cov = simIMR$covariates, type_outcome = "binary",
  nu = c(-4, -3, -4), sample_mcmc = c(200, 100), ssize = 5, seed = 1
)
plot_subgroup_sizes(fit)
```

plot_top_features	<i>Plot the Top Selected Features of an IMR Fit</i>
-------------------	---

Description

Draws a horizontal bar chart of the features with the highest marginal posterior inclusion probability (mPIP). For each platform-feature pair, the plotted score is the highest mPIP attained among the availability subgroups containing that platform; the displayed bars are the largest such scores across all platforms. This complements the per-platform heatmap of `plot(fit, type = "selection")`.

Usage

```
plot_top_features(
  object,
  top = 10,
  base_cex = 1,
  cex_names = NULL,
  cex_axis = NULL,
  cex_lab = NULL,
  cex_main = NULL,
  cex_legend = NULL,
  col = NULL,
  palette = "platform",
  reference = 0.5,
  show_source = TRUE,
  legend = TRUE,
  xlim = c(0, 1),
  mar = NULL,
  mgp = NULL,
  ...
)
```

Arguments

<code>object</code>	A fitted object of class <code>"imr"</code> returned by <code>[imr()]</code> .
<code>top</code>	Integer; the number of highest-mPIP features to display (default <code>'10'</code>).
<code>base_cex</code>	Overall text-size multiplier. The <code>'cex_*'</code> arguments default to values derived from this multiplier (default <code>'1'</code>).
<code>cex_names</code>	Feature-label size multiplier. If <code>'NULL'</code> , a plot-specific default derived from <code>'base_cex'</code> is used.
<code>cex_axis</code>	Axis-label size multiplier. If <code>'NULL'</code> , a plot-specific default derived from <code>'base_cex'</code> is used.
<code>cex_lab</code>	Axis-title size multiplier. If <code>'NULL'</code> , a plot-specific default derived from <code>'base_cex'</code> is used.

cex_main	Main-title size multiplier. If 'NULL', a plot-specific default derived from 'base_cex' is used.
cex_legend	Legend text size multiplier. If 'NULL', a plot-specific default derived from 'base_cex' is used.
col	Optional bar colours. If 'NULL', colours are generated from 'palette'.
palette	Palette name used when 'col = NULL'; '"platform"' gives the package's standard muted platform colours (default).
reference	Optional vertical reference line; use 'NULL' to suppress it (default '0.5').
show_source	Logical; should each bar be annotated with the availability subgroup (bitstring) in which the feature attained its maximum mPIP (default 'TRUE')?
legend	Logical; should the platform legend be drawn (default 'TRUE')?
xlim	Numeric vector of length two giving the horizontal axis limits (default 'c(0, 1)').
mar, mgp	Optional graphical margin and axis-title placement vectors passed to [graphics::par()] for finer layout control.
...	Further graphical parameters passed to [graphics::barplot()].

Value

Invisibly, a data frame of the displayed features with columns 'platform', 'feature', 'mpip' and 'subgroup', where 'mpip' is the maximum subgroup mPIP for that platform-feature pair and 'subgroup' is the subgroup where the maximum is attained. Rows are ordered by decreasing mPIP.

See Also

[imr()], [plot.imr()], [plot_subgroup_sizes()]

Examples

```
data("simIMR", package = "IntegMultiReg")
fit <- imr(
  platform_data_list = simIMR$platforms, outcome = simIMR$outcome,
  cov = simIMR$covariates, type_outcome = "binary",
  nu = c(-4, -3, -4), sample_mcmc = c(200, 100), ssize = 5, seed = 1
)
plot_top_features(fit, top = 8)
```

predict.imr

Predict Outcomes for New Subjects

Description

'predict()' method for objects of class '"imr"' produced by [imr()]. New subjects are matched to the availability subgroup models learned during training, missing platforms are handled automatically, and the test features are standardized with the training set's centring and scaling factors before Bayesian model averaging.

Usage

```
## S3 method for class 'imr'
predict(
  object,
  newdata,
  platform_names = NULL,
  covariates = NULL,
  max_models = 100,
  verbose = FALSE,
  ...
)

predict_imr(object, newdata, ...)
```

Arguments

<code>object</code>	A fitted object of class <code>"imr"</code> returned by <code>[imr()]</code> .
<code>newdata</code>	A list of data frames with the new platform measurements. Each data frame must include <code>'id'</code> as the first column, followed by finite numeric feature columns matching the corresponding training platform.
<code>platform_names</code>	A character vector giving, for each element of <code>'newdata'</code> , the index (<code>"1"</code> , <code>"2"</code> , ...) of the corresponding training platform. Defaults to <code>'NULL'</code> , meaning the elements of <code>'newdata'</code> are taken to be in the same order as the platforms supplied to <code>[imr()]</code> .
<code>covariates</code>	An optional data frame of clinical covariates for the test subjects, including <code>'id'</code> as the first column. Required when the model was fitted with covariates and ignored with a warning when it was not.
<code>max_models</code>	Integer; the maximum number of distinct selection models (gamma configurations) used for Bayesian model averaging. Default <code>'100'</code> .
<code>verbose</code>	Logical; if <code>'TRUE'</code> , print the C routine's diagnostics. Defaults to <code>'FALSE'</code> .
<code>...</code>	Unused; present for S3 compatibility.

Details

Predictions are only produced for subjects observed on at least one platform (and, when `'covariates'` is supplied, with covariate data). For `"binary"` outcomes the returned `'predict'` column is a probability obtained through the probit link (`'pnorm'`); for `"continuous"` and `"right.censored"` outcomes it is the predicted (latent) response.

Value

A named list with one data frame per active availability subgroup model. Each data frame has columns `'id'` (subject identifier) and `'predict'` (predicted value or probability, rounded to three decimals).

See Also

`[imr()]`, `[cv_imr()]`

Examples

```
data("simIMR", package = "IntegMultiReg")
fit <- imr(
  platform_data_list = simIMR$platforms, outcome = simIMR$outcome,
  cov = simIMR$covariates, type_outcome = "binary",
  nu = c(-4, -3, -4), sample_mcmc = c(200, 100), ssize = 5, seed = 1
)
new_x <- simIMR$platforms[[1]][1:10, ]
new_z <- simIMR$platforms[[2]][1:7, ]
predict(fit, newdata = list(new_x, new_z), covariates = simIMR$covariates)
```

print.imr

Print Method for IMR Fits

Description

Prints a compact overview of the fit, including a platform key ('P1', 'P2', ...) that decodes the availability-subgroup bitstrings. Optionally, it can also print the top-ranked features per platform, ranked by each feature's maximum mPIP over availability subgroups containing that platform.

Usage

```
## S3 method for class 'imr'
print(x, threshold = 0.5, rank = FALSE, top = 5, ...)
```

Arguments

x	A fitted object of class '"imr"'.
threshold	Inclusion-probability threshold used to count selected features (default '0.5').
rank	Logical; if 'TRUE', print a short ranked feature table for each platform (default 'FALSE').
top	Integer; when 'rank = TRUE', the number of top-ranked features to show per platform (default '5').
...	Unused; present for S3 compatibility.

Value

'x', invisibly.

simIMR

*Simulated Multi-Platform Example Data***Description**

A small simulated data set illustrating the integrative multi-regression (IMR) setting: three molecular platforms measured on overlapping but partially missing sets of subjects, clinical covariates available for everyone, and three outcome types (continuous, binary and right-censored) driven by the same latent signal.

Usage

```
simIMR
```

Format

A named list with the components:

platforms A list of three data frames, ‘genomic’ (20 features, subjects 1–240), ‘proteomic’ (10 features, subjects 1–180) and ‘metabolomic’ (8 features, subjects 121–300). Each has an ‘id’ column followed by the feature columns.

covariates A data frame of three clinical covariates (‘age’, ‘sex’, ‘stage’) with an ‘id’ column, available for all 300 subjects.

outcome The binary outcome data frame (alias of ‘outcome.binary’), used as the canonical demonstration outcome.

outcome.binary Data frame with ‘id’ and a 0/1 response ‘y’.

outcome.continuous Data frame with ‘id’ and a numeric response ‘y’.

outcome.survival Data frame with ‘id’, observed ‘time’ and event indicator ‘status’ (1 = event, 0 = censored).

truth A list giving, per platform, the column indices of the features with non-zero effects used to generate the outcomes.

Details

The platforms are observed on the following subjects, which induces four non-empty availability subgroups (bitstrings over genomic/proteomic/ metabolomic): ‘011’ (120 subjects), ‘111’ (60), ‘101’ (60) and ‘100’ (60).

The generating script is provided in ‘data-raw/make_simIMR.R’.

Examples

```
data("simIMR", package = "IntegMultiReg")
sapply(simIMR$platforms, dim)
str(simIMR$truth)
```

summary.imr	<i>Summarize an IMR Fit</i>
-------------	-----------------------------

Description

Produces a per-platform summary of the selected features (those whose marginal posterior inclusion probability exceeds ‘threshold’ in at least one subgroup), ranked by their maximum inclusion probability.

Usage

```
## S3 method for class 'imr'
summary(object, threshold = 0.5, ...)

## S3 method for class 'summary.imr'
print(x, ...)

summary_imr(object, ...)
```

Arguments

object	A fitted object of class “imr”.
threshold	Inclusion-probability threshold for selection (default ‘0.5’).
...	Unused; present for S3 compatibility.
x	A “summary.imr” object.

Value

An object of class “summary.imr”: a list with the run metadata and, for each platform, a data frame of selected features with their maximum mPIP and the subgroup achieving it.

Index

* datasets

kircIMR, [7](#)

simIMR, [16](#)

coef.imr, [2](#)

coef_imr(coef.imr), [2](#)

cv_imr, [3](#)

imr, [4](#)

kircIMR, [7](#)

plot.imr, [8](#)

plot_imr(plot.imr), [8](#)

plot_subgroup_sizes, [10](#)

plot_top_features, [12](#)

predict.imr, [13](#)

predict_imr(predict.imr), [13](#)

print.imr, [15](#)

print.summary.imr(summary.imr), [17](#)

simIMR, [16](#)

summary.imr, [17](#)

summary_imr(summary.imr), [17](#)