Package 'LRMiss'

February 20, 2026

Type Package

Title Linear Regression with Missing Data

Version 0.0.1

Description Provides methods for linear regression in the presence of missing data, including missingness in covariates and responses. The package implements two estimators: oss_estimator(), a low-dimensional semi-supervised method, and dantzig_missing(), a high-dimensional approach. The tuning parameter can be selected automatically via cv_dantzig_missing(). See Risebrow and Berrett (2026) <doi:10.48550/arXiv.2602.13729>. Optional support for the 'gurobi' optimizer via the 'gurobi' R package (available from Gurobi, see <https://docs.gurobi.com/projects/optimizer/en/current/reference/r.html>).

Imports MASS, stats, Rglpk, fastDummies, Rdpack

Suggests gurobi

RdMacros Rdpack

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.2.9000

URL https://github.com/benrisebrow/LRMiss

BugReports https://github.com/benrisebrow/LRMiss/issues

NeedsCompilation no

Author Benedict Risebrow [aut, cre], Thomas Berrett [aut]

Maintainer Benedict Risebrow <Benedict.risebrow@warwick.ac.uk>

Repository CRAN

Date/Publication 2026-02-20 08:10:10 UTC

Contents

`cv_dantzig_missing` *Cross-validated Dantzig estimator with missing covariates*

Description

Performs K-fold cross-validation for the Dantzig selector in linear regression models with missing covariates. The method optionally incorporates unlabelled covariate data to improve estimation of second-moment matrices. This function is based on Section 3 of Risebrow and Berrett (2026).

Usage

```
cv_dantzig_missing(
  X, y, X_unlabeled = NULL,
  lambdas = NULL, nlambda = 30, lambda_min_ratio = 1e-3,
  K = 5, standardise = TRUE, gurobi = FALSE,
  seed = 123, fold_ids = NULL, verbose = TRUE,
  plot_path = TRUE
)
```

Arguments

`X`	Labelled covariates.
`y`	Response variables for the labelled data.
`X_unlabeled`	Optional unlabeled covariates.
`lambdas`	Optional sequence of regularisation parameters.
`nlambda`	Number of lambdas if `lambdas` is not supplied.
`lambda_min_ratio`	Smallest lambda as a fraction of the largest.
`K`	Number of cross-validation folds.
`standardise`	Logical; if TRUE covariates are standardised.
`gurobi`	Logical; if TRUE uses Gurobi to solve the linear programs.
`seed`	Random seed for fold assignment.
`fold_ids`	Optional fold assignments for labelled or combined data.
`verbose`	Logical; print progress messages.
`plot_path`	Logical; if TRUE computes and plots the solution path.

Details

For each candidate value of the regularisation parameter, the Dantzig selector is fitted using moment estimates computed from the training folds. Prediction performance is assessed on held-out folds via the maximum absolute moment mismatch. The tuning parameter is selected using both the minimum mean cross-validation score and the one-standard-error (1-SE) rule.

Value

A named list with the following components:

lambdas Numeric vector of tuning parameters used.

cv_scores_matrix Numeric matrix of cross-validation scores (folds × lambdas).

mean_scores Mean CV score for each lambda.

se_scores Standard error of CV scores for each lambda.

lambda_min_mean Lambda minimising mean CV score.

lambda_1se Lambda chosen by the 1-SE rule.

beta_path Optional coefficient path matrix (present if `plot_path=TRUE`).

design_colnames Optional design column names (matching `beta_path` rows).

beta_est Optional saved coefficient vector from full-data path.

intercept_est Optional saved intercept corresponding to `beta_est`.

References

Risebrow BM, Berrett TB (2026). "Semi-supervised linear regression with missing covariates." *arXiv:2602.13729*.

Examples

```
set.seed(1)
n <- 50; p <- 5
X <- matrix(rnorm(n * p), n, p)
y <- X[, 1] + 0.5 * X[, 2] + rnorm(n)
X_unlabeled <- matrix(rnorm(100 * p), 100, p)

cv_fit <- cv_dantzig_missing(
  X = X,
  y = y,
  X_unlabeled = X_unlabeled,
  K = 5,
  nlambda = 20
)

cv_fit$lambda_1se
```

`dantzig_missing` *Dantzig estimator with missing covariates*

Description

High-dimensional linear regression estimator based on the Dantzig selector that accommodates missing covariates and optionally leverages unlabelled covariate data. This function is a user-facing wrapper that dispatches to either a standardised or unstandardised implementation depending on the value of `standardise`. This function is based on Section 3 of Risebrow and Berrett (2026).

Usage

```
dantzig_missing(
  X_labeled, y, X_unlabeled = NULL, lambda,
  gurobi = FALSE, standardise = TRUE
)
```

Arguments

`X_labeled`	Numeric matrix or data.frame of labelled covariates, with rows corresponding to observations and columns to covariates. Missing values are allowed.
`y`	Numeric response vector of length `nrow(X_labeled)`.
`X_unlabeled`	Optional numeric matrix or data.frame of unlabelled covariates. If supplied, these observations are used only for estimating second moments of the covariates and do not contribute to the response.
`lambda`	Positive numeric scalar giving the Dantzig regularisation parameter.
`gurobi`	Logical; if TRUE, the linear programs are solved using the **gurobi** optimizer (a valid Gurobi installation and license are required). If FALSE, the open-source solver from **Rglpk** is used instead.
`standardise`	Logical; if TRUE, covariates are standardised prior to estimation and the resulting coefficients are mapped back to the original scale with an intercept term returned.

Details

Categorical covariates are internally dummy-encoded, with missing values preserved. When `standardise = TRUE`, covariates are centred and scaled using empirical means and standard deviations computed from the combined labelled and unlabelled samples.

Value

A list with at least the following component:

beta_hat Numeric vector of estimated regression coefficients, with names corresponding to the encoded design matrix columns.

If `standardise = TRUE`, the list also contains:

intercept Numeric scalar giving the estimated intercept term.

References

Risebrow BM, Berrett TB (2026). "Semi-supervised linear regression with missing covariates." *arXiv:2602.13729*.

Examples

```
set.seed(1)
n <- 50; p <- 5
X_full <- matrix(rnorm(n * p), n, p)
beta_true <- c(1, 0.5, rep(0, p - 2))
y <- X_full[, 1] * beta_true[1] + X_full[, 2] * beta_true[2] + rnorm(n)

# introduce missingness into covariates
X_miss <- X_full
X_miss[sample(length(X_miss), size = 0.1 * length(X_miss))] <- NA

# fit Dantzig estimator (example lambda; tune in practice)
fit <- dantzig_missing(
  X_labeled = X_miss,
  y = y,
  lambda = 0.1,
  standardise = TRUE
)
fit$beta_hat
```

`estimate_cov_raw` *Covariance estimator with missing data*

Description

Estimates the covariance matrix of a design matrix in the presence of missing values. Each covariance entry is computed using all observations for which the corresponding pair of covariates is jointly observed.

Usage

```
estimate_cov_raw(X)
```

Arguments

`X`	Numeric matrix (or object coercible to a matrix) containing covariates. Rows correspond to observations and columns to variables. Missing values (`NA`) are allowed.

Details

Let X_{ij} denote the j-th covariate for observation i. For each pair of variables (j, k), the covariance estimate is

$$\hat{\Sigma}_{jk} = \frac{1}{n_{jk}} \sum_{i: X_{ij}, X_{ik} \text{ observed}} X_{ij} X_{ik},$$

where n_{jk} is the number of observations for which both entries are observed. If no such observations exist, the corresponding covariance entry is set to `NA`.

This estimator is symmetric by construction and reduces to the usual sample second-moment matrix when the data contain no missing values.

Value

A numeric p x p matrix containing the estimated covariance matrix, where p = ncol(X). Entries corresponding to variable pairs that are never jointly observed are NA.

Examples

```
set.seed(1)
X <- matrix(rnorm(25), 25, 5)
X[sample(length(X), 10)] <- NA
Sigma_hat <- estimate_cov_raw(X)
Sigma_hat
```

---

oss_estimator *Linear regression with missing data*

Description

Fits a linear regression model in the presence of missing covariates and/or missing responses using the OSS (Ordinary Semi-Supervised) estimator. This corresponds to Section 2 of Risebrow and Berrett (2026). The method exploits partially observed covariates and optionally unlabelled observations to improve estimation efficiency. If sufficient complete cases are present, the weights are estimated from them, otherwise the weights are estimated using an initial consistent estimate. If sufficient unlabelled data is present the covariance matrix is estimated exclusively from them, otherwise the covariance is estimated elementwise.

Usage

```
oss_estimator(formula, data,
              all_weights_one = FALSE,
              crossfitting = FALSE)
```

Arguments

formula	A model formula specifying the linear regression, e.g. y ~ x1 + x2.
data	A data.frame containing the variables in the model. Rows with missing responses are treated as unlabelled observations.
all_weights_one	Logical; if TRUE, all missingness-pattern weights are set to one, yielding an unweighted OSS estimator.
crossfitting	Logical; if TRUE, a two-fold cross-fitted version of the OSS estimator is used.

Value

An invisible list with components:

coef Numeric vector of estimated regression coefficients.

sigma2_hat Estimated noise variance, or NA if not computed.

weights Named vector of weights associated with each missingness pattern.

groups Data.frame mapping labelled observations to missingness patterns.

beta_cc Complete-case coefficient estimates if used, otherwise NULL.

References

Risebrow BM, Berrett TB (2026). "Semi-supervised linear regression with missing covariates." *arXiv:2602.13729*.

Examples

```
dat <- data.frame(
  y = c(1.0, NA, 2.3, 0.5),
  x1 = rnorm(4),
  x2 = rnorm(4)
)

## Without cross-fitting
res <- oss_estimator(y ~ x1 + x2, dat)

## With cross-fitting
res_cf <- oss_estimator(y ~ x1 + x2, dat, crossfitting = TRUE)
```

Index