Package 'MLSP'

October 29, 2026

Type Package

Title Machine Learning Models for Soil Properties from Spectroscopy

Version 0.2.0

Description Soil spectroscopy studies use many different combinations of spectral preprocessing and calibration methods, which makes results difficult to compare or generalize across studies. For visible and near-infrared (VNIR) spectra (350-2500 nm), runs every combination of preprocessing and calibration methods on identical repeated calibration and validation splits. The 26 preprocessing methods are raw reflectance, standard normal variate, pseudo-absorbance, and 23 Savitzky-Golay smoothers, first derivatives, and second derivatives with linear, quadratic, or cubic polynomials and 3- to 11-point windows, excluding settings that give mathematically identical filters. The calibration methods are principal component regression (PCR), partial least squares regression (PLSR), least absolute shrinkage and selection operator (LASSO), 'Cubist', and random forest. Model complexity is tuned by cross-validation within each calibration set, and combinations are ranked by the coefficient of determination (R2), root mean squared error (RMSE), ratio of performance to interquartile distance (RPIQ), or bias, reporting the top 10 by default. The methods are those most commonly used in published soil spectroscopy studies, following Savitzky and Golay (1964) <doi:10.1021/ac60214a047>, Barnes et al. (1989) <doi:10.1366/0003702894202201>, Rinnan et al. (2009) <doi:10.1016/j.trac.2009.07.007>, Bellon-Maurel et al. (2010) <doi:10.1016/j.trac.2010.05.006>, and Clingensmith et al. (2019) <doi:10.1111/ejss.12753>.

License GPL-2

Encoding UTF-8

RoxygenNote 7.3.1

Imports stats, utils, gsignal, pls, glmnet, Cubist, randomForest

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author Pengyuan Chen [aut],
Christopher Clingensmith [aut],
Chenglong Ye [aut],
Sabine Grunwald [aut],
Katsutoshi Mizuta [aut, cre]

Maintainer Katsutoshi Mizuta <toshi.m@uky.edu>

Repository CRAN

Date/Publication 2026-10-29 15:20:02 UTC

Contents

colsMean 2
merge_of_lab_and_spectrum 3
ml_f 4
msd.comp 7
preprocess_methods 8
raw 9
results 10
soil_preprocess 11
spectral_preprocess 13

Index **15**

`colsMean` *Compute Row-wise Means for Groups of Columns*

Description

This function computes the row-wise mean for consecutive groups of columns in a matrix or data frame.

Usage

```
colsMean(x, ncols)
```

Arguments

`x`	A numeric matrix or data frame.
`ncols`	An integer specifying the number of consecutive columns to group together.

Details

The function splits the columns of `x` into consecutive groups of `ncols` columns and calculates the mean of each row for each group. The number of columns in `x` must be divisible by `ncols`.

Value

A numeric matrix with the same number of rows as `x` and `ncol(x) / ncols` columns, where each column is the row-wise mean of a group of `ncols` columns.

Examples

```
mat <- matrix(1:12, nrow = 3)
colsMean(mat, 2)
```

`merge_of_lab_and_spectrum` *Merge Soil Laboratory Data with Spectral Data*

Description

Averages replicate scans of each sample, matches the spectra to the soil laboratory data, and creates repeated random calibration/validation splits used by every model in `ml_f`.

Usage

```
merge_of_lab_and_spectrum(
  soil_data,
  data_NaturaSpec_cleaned,
  soil_id = "LAB_NUM",
  spec_id = "Wavelength",
  n_splits = 4,
  cal_prop = 0.7
)
```

Arguments

`soil_data`	A data frame of soil laboratory data with a sample identifier column (default `LAB_NUM`) and one column per soil property.
`data_NaturaSpec_cleaned`	A data frame of spectra with a sample identifier column (default `Wavelength`, matching `LAB_NUM`) and 2151 reflectance columns (350-2500 nm at 1 nm). Several rows (scans) per sample are averaged.
`soil_id`	Name of the identifier column in `soil_data`.
`spec_id`	Name of the identifier column in `data_NaturaSpec_cleaned`.
`n_splits`	Number of random calibration/validation splits.
`cal_prop`	Proportion of samples in each calibration set.

Details

Splits are drawn with `sample`; call `set.seed` beforehand for reproducible results. All preprocessing and calibration methods are evaluated on the same splits, so their accuracy metrics are directly comparable.

Value

A named list:

soil Soil laboratory data for the matched samples (identifier first).

vnir.matrix Matrix of averaged reflectance (2151 columns).

j List of `n_splits` vectors of calibration-row indices.

rm1, rm2, rm3, rm4 Indices of the 1-nm bands excluded as edges or detector artifacts (350-395, 986-1015, 1786-1815, and 2476-2500 nm).

ind Indices of the 1-nm bands averaged into 10-nm bands.

rmove Indices of the 10-nm bands removed after averaging.

vars Names of the 202 retained 10-nm bands.

Examples

```
# Simulated data: 50 soil samples scanned at 350-2500 nm (2151 bands).
set.seed(1)
n <- 50
wl <- 350:2500
soc <- runif(n, 0.5, 4)
spec <- t(sapply(soc, function(s)
  0.5 - 0.05 * s * exp(-((wl - 1400) / 600)^2) + rnorm(length(wl), 0, 0.01)))
colnames(spec) <- wl
spectra <- data.frame(Wavelength = 1:n, spec, check.names = FALSE)
soil_data <- data.frame(LAB_NUM = 1:n, SOC = soc, clay = runif(n, 5, 40),
  pH = runif(n, 5, 8))
merged <- merge_of_lab_and_spectrum(soil_data, spectra)
str(merged[1:3], max.level = 1)
```

`ml_f` *Compare Spectral Preprocessing and Calibration Methods*

Description

Runs every combination of the selected spectral preprocessing methods and calibration methods on the same calibration/validation splits and ranks the combinations by accuracy.

Usage

```
ml_f(
  x,
  y,
  smoother_selection = "all",
  type_of_soil = 2,
  model_selection = TRUE,
  models = c("pcr", "plsr", "lasso", "cubist", "rf"),
  top_n = 10,
  rank_by = "val-rpiq",
  verbose = FALSE,
  ...
)
```

Arguments

x	A data frame of soil laboratory data with a LAB_NUM identifier column and one column per soil property (passed to merge_of_lab_and_spectrum).
y	A data frame of spectra with a sample identifier column (Wavelength, matching LAB_NUM) and 2151 reflectance columns (350-2500 nm at 1 nm).
smoother_selection	Preprocessing methods to compare: "all" (default; all 26 methods in preprocess_methods: raw reflectance, SNV, $\log_{10}(1/R)$, and 23 Savitzky-Golay filters), a character vector of method names, or a single preprocessing function such as raw.
type_of_soil	The soil property to model: a column name or index of x (index 1 is the identifier, so the first property is 2).
model_selection	Logical; if TRUE (default), return the top_n best combinations; if FALSE, return all combinations.
models	Calibration methods to compare: any of "pcr", "plsr", "lasso", "cubist", and "rf".
top_n	Number of combinations returned when model_selection = TRUE.
rank_by	Metric used for ranking, one of the column names of the output (e.g., "val-rpiq", "val-rmse", "val-r2"). Higher is better for R2 and RPIQ, lower for RMSE and MSE, and closer to zero for bias.
verbose	Logical; print progress messages.
...	Further arguments passed to the model functions (ncomp_max, rf_ntree, cv_folds); see soil_preprocess.

Details

Spectra are matched to the soil data and split once (see merge_of_lab_and_spectrum), so every combination is evaluated on identical calibration and validation sets. Each preprocessing method is computed once and reused by all calibration methods. Metrics are averaged over the splits (see results). A combination that fails to fit is reported with a warning and NA metrics instead of stopping the run. Call set.seed first for reproducible splits.

Running all 26 distinct preprocessing methods with all five calibration methods fits 130 combinations on each split and may take several minutes or longer for large spectral libraries. Restricting `models`, e.g. to `c("plsr", "lasso")`, is much faster.

The preprocessing methods and the Savitzky-Golay naming (`sg.<polynomial>.<window>.<derivative>`) follow Clingensmith et al. (2019); see `preprocess_methods` for how window size, polynomial order, and derivative order affect the spectra.

Value

A data frame ranked by `rank_by`, with columns `rank`, `preprocessing`, `model`, `LV`, the cross-validation metrics (`cv-r2`, `cv-bias`, `cv-rmse`, `cv-mse`, `cv-rpiq`), and the validation metrics (`val-r2`, `val-bias`, `val-rmse`, `val-mse`, `val-rpiq`).

References

Clingensmith CM, Grunwald S, Wani SP (2019). Evaluation of calibration subsetting and new chemometric methods on the spectral prediction of key soil properties in a data-limited environment. European Journal of Soil Science 70(1). doi:10.1111/ejss.12753

See Also

`preprocess_methods`, `soil_preprocess`

Examples

```
# Simulated data: 50 soil samples scanned at 350-2500 nm (2151 bands).
set.seed(1)
n <- 50
wl <- 350:2500
soc <- runif(n, 0.5, 4)
spec <- t(sapply(soc, function(s)
  0.5 - 0.05 * s * exp(-((wl - 1400) / 600)^2) + rnorm(length(wl), 0, 0.01)))
colnames(spec) <- wl
spectra <- data.frame(Wavelength = 1:n, spec, check.names = FALSE)
soil_data <- data.frame(LAB_NUM = 1:n, SOC = soc, clay = runif(n, 5, 40))

# Three preprocessing methods x two calibration methods
ml_f(soil_data, spectra,
     smoother_selection = c("raw", "snv", "sg.q.11.1"),
     type_of_soil = "SOC", models = c("pcr", "plsr"), top_n = 5)

# All 26 distinct preprocessing methods with PLSR
top10 <- ml_f(soil_data, spectra, type_of_soil = "SOC", models = "plsr")
top10
```

`msd.comp` *Compute Model Evaluation Metrics*

Description

This function computes various statistics for comparing observed values ‘y‘ with predicted values ‘yhat‘. It includes correlation, regression coefficients, bias, RMSE, MSE, and predictive performance metrics like RPD and RPIQ.

Usage

```
msd.comp(y, yhat)
```

Arguments

`y`	Numeric vector of observed values.
`yhat`	Numeric vector of predicted values (same length as ‘y‘).

Value

A named numeric vector with the following components:

r Pearson correlation between ‘y‘ and ‘yhat‘

int Intercept of regression of ‘y‘ on ‘yhat‘

slope Slope of regression of ‘y‘ on ‘yhat‘

r2 Coefficient of determination (R-squared)

bias Mean bias: mean(yhat) - mean(y)

rmse Root mean squared error

mse Mean squared error

sb Systematic bias component of MSE

nu Non-unity slope component of MSE

lc Lack-of-correlation component of MSE

rmse.c Corrected RMSE after removing bias

mse.c Corrected MSE after removing bias

rpd Ratio of standard deviation to RMSE (RPD)

rpiq Ratio of interquartile range to RMSE (RPIQ)

Examples

```
y_obs <- c(1.2, 3.4, 2.5, 4.1)
y_pred <- c(1.1, 3.5, 2.4, 4.0)
msd.comp(y_obs, y_pred)
```

`preprocess_methods` *Catalogue of Spectral Preprocessing Methods*

Description

Lists the 26 spectral preprocessing methods in the package: raw reflectance, standard normal variate (SNV), pseudo-absorbance ($\log_{10}(1/R)$), and 23 Savitzky-Golay (SG) filters.

Usage

```
preprocess_methods()
```

Value

A data frame with one row per method and columns `method` (the name used in `spectral_preprocess` and `ml_f`), `family`, `poly_order`, `window`, `derivative`, `description`, and `equivalent_to` (other SG settings that give the same filter).

Savitzky-Golay filters

An SG filter fits a polynomial by least squares to the points inside a moving window centred on each wavelength and replaces the centre value with the fitted value (smoothing) or with a derivative of the fitted polynomial (Savitzky and Golay, 1964). Three settings define each filter, following the `SG.P.N.D` naming used by Clingensmith et al. (2019), who applied SG smoothers and first and second derivatives with 11-point windows to soil VNIR spectra:

Polynomial order (P) 1 = linear (1), `q` = quadratic (2), `c` = cubic (3). Higher orders follow narrow absorption features more closely; lower orders smooth more strongly. A linear SG smoother is a simple moving average.

Window size (N) 3, 5, 7, 9, or 11 points (nm). Wider windows remove more noise but can flatten narrow absorption features. The window must exceed the polynomial order.

Derivative (D) 0 = smoothing, 1 = first derivative, which removes additive baseline offsets, and 2 = second derivative, which also removes linear baseline slopes and separates overlapping absorption bands.

The 23 filters are linear smoothers (3- to 11-point windows), quadratic smoothers (5 to 11), quadratic first derivatives (3 to 11), cubic first derivatives (5 to 11), and quadratic second derivatives (3 to 11). For example, `sg.q.11.1` is the first derivative from a quadratic polynomial fitted in an 11-point window.

Equivalent settings that are not included

Because the window is symmetric, raising the polynomial order from even to odd does not change the fitted value, slope, or curvature at the window centre (Savitzky and Golay, 1964). Fourteen further settings therefore reproduce a method above exactly and are not included:

- cubic smoothers equal quadratic smoothers (`sg.c.N.0` = `sg.q.N.0`, N = 5 to 11);
- a 3-point quadratic smoother returns the spectrum unchanged (`sg.q.3.0` = `raw`);

- linear first derivatives equal quadratic first derivatives (`sg.l.N.1` = `sg.q.N.1`, N = 3 to 11);
- cubic second derivatives equal quadratic second derivatives (`sg.c.N.2` = `sg.q.N.2`, N = 5 to 11).

Results reported under these names, such as `SG.C.11.0` in Clingensmith et al. (2019), are reproduced by the equivalent method.

References

Savitzky A, Golay MJE (1964). Smoothing and differentiation of data by simplified least squares procedures. Analytical Chemistry 36, 1627-1639. doi:10.1021/ac60214a047

Barnes RJ, Dhanoa MS, Lister SJ (1989). Standard normal variate transformation and de-trending of near-infrared diffuse reflectance spectra. Applied Spectroscopy 43, 772-777. doi:10.1366/0003702894202201

Rinnan A, van den Berg F, Engelsen SB (2009). Review of the most common pre-processing techniques for near-infrared spectra. TrAC Trends in Analytical Chemistry 28, 1201-1222. doi:10.1016/j.trac.2009.07.007

Clingensmith CM, Grunwald S, Wani SP (2019). Evaluation of calibration subsetting and new chemometric methods on the spectral prediction of key soil properties in a data-limited environment. European Journal of Soil Science 70(1). doi:10.1111/ejss.12753

See Also

`spectral_preprocess`, `ml_f`

Examples

```
methods <- preprocess_methods()
table(methods$family)
# Savitzky-Golay filters by polynomial order and derivative
sg <- methods[methods$family == "Savitzky-Golay", ]
table(sg$poly_order, sg$derivative)
methods[, c("method", "description", "equivalent_to")]
```

---

raw *Raw Reflectance Predictors*

Description

Averages raw VNIR reflectance (350-2500 nm, 1-nm intervals) into 10-nm bands and removes the noisy spectral edges and the detector-artifact regions near 1000 nm and 1800 nm. Equivalent to `spectral_preprocess(vnir.matrix, "raw")`.

Usage

```
raw(vnir.matrix)
```

Arguments

`vnir.matrix`	A numeric data frame or matrix with one reflectance spectrum per row and 2151 columns (350-2500 nm).

Value

A data frame with one row per sample and 202 columns.

See Also

`spectral_preprocess`

Examples

```
vnir <- data.frame(matrix(runif(3 * 2151, 0.2, 0.6), nrow = 3))
names(vnir) <- 350:2500
dim(raw(vnir))
```

`results` *Summarize Model Performance Metrics*

Description

Averages accuracy metrics over the calibration/validation splits returned by `pcr_preprocess`, `plsr_preprocess`, `lasso_preprocess`, `rf_preprocess`, or `cubist_preprocess`.

Usage

```
results(metric.list, soil_type)
```

Arguments

`metric.list`	A list of `msd.comp` matrices (one per split) with columns `cal` and `val`.
`soil_type`	Not used; kept for backward compatibility.

Value

A named numeric vector of metrics averaged over splits:

LV Number of latent variables (PCR/PLSR; `NA` otherwise).

cv-r2, cv-bias, cv-rmse, cv-mse, cv-rpiq Coefficient of determination, bias, root mean squared error, mean squared error, and ratio of performance to interquartile distance from cross-validation within the calibration set.

val-r2, val-bias, val-rmse, val-mse, val-rpiq The same metrics on the independent validation set.

References

Bellon-Maurel V, Fernandez-Ahumada E, Palagos B, Roger J-M, McBratney A (2010). Critical review of chemometric indicators commonly used for assessing the quality of the prediction of soil attributes by NIR spectroscopy. TrAC Trends in Analytical Chemistry 29, 1073-1081. doi:10.1016/j.trac.2010.05.006

See Also

soil_preprocess, ml_f

Examples

```
# Two splits of observed and predicted values
split_metrics <- function(y, yc, yv) cbind(cal = msd.comp(y, yc), val = msd.comp(y, yv))
y <- c(1.2, 3.4, 2.5, 4.1, 2.2)
metric.list <- list(
  split1 = split_metrics(y, y + 0.1, y - 0.2),
  split2 = split_metrics(y, y - 0.1, y + 0.3))
results(metric.list)
```

---

soil_preprocess *Fit and Evaluate One Calibration Method*

Description

Each function fits one calibration method to preprocessed spectra on every calibration/validation split and computes accuracy metrics with msd.comp:

- pcr_preprocess(): principal component regression (PCR)
- plsr_preprocess(): partial least squares regression (PLSR)
- lasso_preprocess(): least absolute shrinkage and selection operator (LASSO)
- rf_preprocess(): random forest
- cubist_preprocess(): 'Cubist' rule-based regression

Usage

```
pcr_preprocess(
  soil,
  vnir.matrix,
  j,
  preprocess,
  type_of_soil,
  ncomp_max = 20,
  cv_folds = 10
)
```

```
plsr_preprocess(
  soil,
  vnir.matrix,
  j,
  preprocess,
  type_of_soil,
  ncomp_max = 20,
  cv_folds = 10
)

lasso_preprocess(soil, vnir.matrix, j, preprocess, type_of_soil, cv_folds = 10)

rf_preprocess(soil, vnir.matrix, j, preprocess, type_of_soil, rf_ntree = 500)

cubist_preprocess(
  soil,
  vnir.matrix,
  j,
  preprocess,
  type_of_soil,
  cv_folds = 10
)
```

Arguments

soil	A data frame of soil properties, e.g. the soil element returned by merge_of_lab_and_spectrum.
vnir.matrix	A numeric matrix or data frame of spectra (2151 columns, 350-2500 nm).
j	A list of calibration-row index vectors (e.g., from merge_of_lab_and_spectrum).
preprocess	A preprocessing method: a method name from preprocess_methods (e.g., "sg.q.11.1") or a function that takes the spectra and returns a data frame of predictors (e.g., raw).
type_of_soil	The soil property to model: a column name or index of soil.
ncomp_max	Maximum number of PCR/PLSR components considered.
cv_folds	Number of CV folds used within each calibration set.
rf_ntree	Number of trees for random forest.

Details

Within each calibration set, model complexity is tuned and accuracy is estimated by cross-validation (CV): the number of PCR/PLSR components is chosen with the one-standard-error rule of selectNcomp (at most ncomp_max components), the LASSO penalty is the CV minimum from cv.glmnet, random forest uses out-of-bag predictions, and 'Cubist' uses k-fold CV predictions. The CV predictions give the cal metrics; the fitted model is then applied to the held-out validation set (val metrics). Predictors are scaled for PCR and PLSR.

Value

A list with one element per split. Each element is a matrix of the msd.comp statistics with columns cal (CV) and val (validation), and attribute LV (the number of latent variables for PCR/PLSR, NA otherwise).

References

Mevik B-H, Wehrens R (2007). The pls package: principal component and partial least squares regression in R. Journal of Statistical Software 18(2), 1-23. doi:10.18637/jss.v018.i02

See Also

ml_f, results

Examples

```
# Simulated data: 50 soil samples scanned at 350-2500 nm (2151 bands).
set.seed(1)
n <- 50
wl <- 350:2500
soc <- runif(n, 0.5, 4)
spec <- t(sapply(soc, function(s)
  0.5 - 0.05 * s * exp(-((wl - 1400) / 600)^2) + rnorm(length(wl), 0, 0.01)))
colnames(spec) <- wl
spectra <- data.frame(Wavelength = 1:n, spec, check.names = FALSE)
soil_data <- data.frame(LAB_NUM = 1:n, SOC = soc, clay = runif(n, 5, 40))
xy <- merge_of_lab_and_spectrum(soil_data, spectra)

# PLSR on first-derivative spectra (quadratic, 11-point window)
fit_plsr <- plsr_preprocess(xy$soil, xy$vnir.matrix, xy$j,
                            preprocess = "sg.q.11.1", type_of_soil = "SOC")
results(fit_plsr)

# Random forest on raw reflectance
fit_rf <- rf_preprocess(xy$soil, xy$vnir.matrix, xy$j,
                        preprocess = "raw", type_of_soil = "SOC")
results(fit_rf)
```

---

spectral_preprocess *Preprocess VNIR Spectra*

Description

Applies one of the 26 spectral preprocessing methods listed by preprocess_methods to VNIR spectra measured from 350 to 2500 nm at 1-nm intervals (2151 bands), averages the result into 10-nm bands, and removes the noisy spectral edges and the detector-artifact regions near 1000 nm and 1800 nm (350-395, 986-1015, 1786-1815, and 2476-2500 nm), leaving 202 predictor bands.

Usage

```
spectral_preprocess(vnir.matrix, method = "raw")
```

Arguments

vnir.matrix	A numeric data frame or matrix with one reflectance spectrum per row and 2151 columns (350-2500 nm).
method	Name of the preprocessing method: "raw", "snv", "log1_R", or a Savitzky-Golay filter named sg.<polynomial>.<window>.<derivative> (see preprocess_methods).

Details

Filters and transformations are applied to the full 1-nm spectrum before band averaging, so Savitzky-Golay windows are in nanometres and derivatives are per nanometre. SNV centres and scales each spectrum with the mean and standard deviation of the retained (non-artifact) bands (Barnes et al., 1989). The removed regions follow the detector edge effects described by Clingensmith et al. (2019).

Value

A data frame with one row per sample and 202 columns named by the wavelength at the centre of each 10-nm band.

References

Barnes RJ, Dhanoa MS, Lister SJ (1989). Standard normal variate transformation and de-trending of near-infrared diffuse reflectance spectra. Applied Spectroscopy 43, 772-777. doi:10.1366/0003702894202201

Clingensmith CM, Grunwald S, Wani SP (2019). Evaluation of calibration subsetting and new chemometric methods on the spectral prediction of key soil properties in a data-limited environment. European Journal of Soil Science 70(1). doi:10.1111/ejss.12753

See Also

preprocess_methods, raw

Examples

```
wl <- 350:2500
vnir <- data.frame(t(sapply(c(0.30, 0.35, 0.40), function(b)
  b + 0.1 * (wl - 350) / 2150 - 0.03 * exp(-((wl - 1900) / 40)^2))))
names(vnir) <- wl
smooth <- spectral_preprocess(vnir, "sg.l.11.0")  # moving average, 11 nm
d1 <- spectral_preprocess(vnir, "sg.q.11.1")      # 1st derivative, quadratic
d2 <- spectral_preprocess(vnir, "sg.q.11.2")      # 2nd derivative, quadratic
dim(d1)
# The first derivative removes the offsets between the three spectra
round(apply(d1, 2, sd)[1:5], 6)
```

Index

colsMean, 2
cubist_preprocess, *10*
cubist_preprocess *(*soil_preprocess*)*, 11
cv.glmnet, *12*

lasso_preprocess, *10*
lasso_preprocess *(*soil_preprocess*)*, 11

merge_of_lab_and_spectrum, 3, *5*, *12*
ml_f, *3*, 4, *8*, *9*, *11*, *13*
msd.comp, 7, *10*, *11*, *13*

pcr_preprocess, *10*
pcr_preprocess *(*soil_preprocess*)*, 11
plsr_preprocess, *10*
plsr_preprocess *(*soil_preprocess*)*, 11
preprocess_methods, *5*, *6*, 8, *12–14*

raw, *5*, 9, *12*, *14*
results, *5*, 10, *13*
rf_preprocess, *10*
rf_preprocess *(*soil_preprocess*)*, 11

sample, *4*
selectNcomp, *12*
set.seed, *4*, *5*
soil_preprocess, *5*, *6*, *11*, 11
spectral_preprocess, *8–10*, 13