

# Package: swaprinc (via r-universe)

September 10, 2024

**Title** Swap Principal Components into Regression Models

**Version** 1.0.1.9000

**Description** Obtaining accurate and stable estimates of regression coefficients can be challenging when the suggested statistical model has issues related to multicollinearity, convergence, or overfitting. One solution is to use principal component analysis (PCA) results in the regression, as discussed in Chan and Park (2005) <[doi:10.1080/01446190500039812](https://doi.org/10.1080/01446190500039812)>. The swaprinc() package streamlines comparisons between a raw regression model with the full set of raw independent variables and a principal component regression model where principal components are estimated on a subset of the independent variables, then swapped into the regression model in place of those variables. The swaprinc() function compares one raw regression model to one principal component regression model, while the compswap() function compares one raw regression model to many principal component regression models. Package functions include parameters to center, scale, and undo centering and scaling, as described by Harvey and Hansen (2022) <[https://cran.r-project.org/package=LearnPCA/vignettes/Vig\\_03\\_Step\\_By\\_Step\\_PCA.pdf](https://cran.r-project.org/package=LearnPCA/vignettes/Vig_03_Step_By_Step_PCA.pdf)>. Additionally, the package supports using Gifi methods to extract principal components from categorical variables, as outlined by Rossiter (2021) <[https://www.css.cornell.edu/faculty/dgr2/\\_static/files/R\\_html/NonlinearPCA.html#2\\_Package](https://www.css.cornell.edu/faculty/dgr2/_static/files/R_html/NonlinearPCA.html#2_Package)>.

**License** MIT + file LICENSE

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.2.3

**URL** <https://github.com/mncube/swaprinc>

**BugReports** <https://github.com/mncube/swaprinc/issues>

**Imports** broom, broom.mixed, dplyr, Gifi, lme4, magrittr, rlang, tidyselect

**Suggests** testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Repository** <https://mncube.r-universe.dev>

**RemoteUrl** <https://github.com/mncube/swaprinc>

**RemoteRef** HEAD

**RemoteSha** 9d9ef19b7c0da66b591fb5c8c321da9784eac441

## Contents

compswap . . . . .	2
swaprinc . . . . .	4
<b>Index</b>	<b>7</b>

---

compswap	<i>Compare swaprinc Models</i>
----------	--------------------------------

---

## Description

The `swaprinc` function compares a regression model using raw variables to a model with principal components swapped in. The `compswap` function compares a regression model with raw variables to multiple models with principal components swapped in. Parameter lists are recycled to ensure they are the same length as the longest parameter list.

## Usage

```
compswap(
  data,
  formula,
  engine = "stats",
  .prc_eng_list = list("stats"),
  .pca_varlist = list(c(NULL)),
  .n_pca_list = list(NULL),
  .lpca_center_list = list("none"),
  .lpca_scale_list = list("none"),
  .lpca_undo_list = list(FALSE),
  .gifi_transform_list = list("none"),
  .gifi_trans_vars_list = list(c(NULL)),
  .gifi_trans_dims_list = list(NULL),
  .no_tresp_list = list(FALSE),
  .miss_handler_list = list("none"),
  .model_options_list = list("noaddpars"),
  .prcomp_options_list = list("noaddpars"),
```

```

    .gifi_princals_options_list = list("noaddpars"),
    .gifi_trans_options_list = list("noaddpars")
  )

```

### Arguments

<code>data</code>	A dataframe
<code>formula</code>	A quoted model formula
<code>engine</code>	The engine for fitting the model. Options are "stats" or "lme4".
<code>.prc_eng_list</code>	A list of <code>prc_eng</code> values (see <code>swaprinc</code> documentation)
<code>.pca_varlist</code>	A list of <code>pca_vars</code> (see <code>swaprinc</code> documentation)
<code>.n_pca_list</code>	A list of <code>n_pca_components</code> (see <code>swaprinc</code> documentation)
<code>.lpca_center_list</code>	A list of <code>lpca_center</code> values (see <code>swaprinc</code> documentation)
<code>.lpca_scale_list</code>	A list of <code>lpca_scale</code> values (see <code>swaprinc</code> documentation)
<code>.lpca_undo_list</code>	A list of <code>lpca_undo</code> values (see <code>swaprinc</code> documentation)
<code>.gifi_transform_list</code>	A list of <code>gifi_transform</code> values (see <code>swaprinc</code> documentation)
<code>.gifi_trans_vars_list</code>	A list of <code>gifi_trans_vars</code> values (see <code>swaprinc</code> documentation)
<code>.gifi_trans_dims_list</code>	A list of <code>gifi_trans_dims</code> values (see <code>swaprinc</code> documentation)
<code>.no_tresp_list</code>	A list of <code>no_tresp</code> values (see <code>swaprinc</code> documentation)
<code>.miss_handler_list</code>	A list of <code>miss_handler</code> values (see <code>swaprinc</code> documentation)
<code>.model_options_list</code>	A list of <code>model_options</code> (see <code>swaprinc</code> documentation)
<code>.prcomp_options_list</code>	A list of <code>prcomp_options</code> (see <code>swaprinc</code> documentation)
<code>.gifi_princals_options_list</code>	A list of <code>gifi_princals_options</code> (see <code>swaprinc</code> documentation)
<code>.gifi_trans_options_list</code>	A list of <code>gifi_trans_options</code> (see <code>swaprinc</code> documentation)

### Value

A list containing a list of fitted models and a comparison metrics data frame.

**Examples**

```

# Load the iris dataset
data(iris)

# Define the formula
formula <- "Sepal.Length ~ Sepal.Width + Petal.Length + Petal.Width"

# Define the pca_varlist
pca_varlist <- list(c("Sepal.Width", "Petal.Length"),
                  c("Sepal.Width", "Petal.Width"))

# Define the n_pca_list
n_pca_list <- list(2, 2)

# Set scaling values
lpca_center_list <- list("none", "none")
lpca_scale_list <- list("none", "none")
lpca_undo_list <- list(FALSE, FALSE)

# Run compswap
compswap_results <- compswap(data = iris,
                             formula = formula,
                             engine = "stats",
                             .pca_varlist = pca_varlist,
                             .n_pca_list = n_pca_list,
                             .lpca_center_list = lpca_center_list,
                             .lpca_scale_list = lpca_scale_list,
                             .lpca_undo_list = lpca_undo_list)

```

---

swaprinc

*Swap in Principal Components*


---

**Description**

Compare a regression model using raw variables with another model where principal components are extracted from a subset of the raw independent variables, and a user-defined number of these principal components are then used to replace the original subset of variables in the regression model.

**Usage**

```

swaprinc(
  data,
  formula,
  engine = "stats",
  prc_eng = "stats",
  pca_vars,
  n_pca_components,
  norun_raw = FALSE,

```

```

  lpca_center = "none",
  lpca_scale = "none",
  lpca_undo = FALSE,
  gifi_transform = "none",
  gifi_trans_vars,
  gifi_trans_dims,
  no_tresp = FALSE,
  miss_handler = "none",
  model_options = "noaddpars",
  prcomp_options = "noaddpars",
  gifi_princals_options = "noaddpars",
  gifi_trans_options = "noaddpars"
)

```

### Arguments

<code>data</code>	A dataframe
<code>formula</code>	A quoted model formula
<code>engine</code>	The engine for fitting the model. Options are 'stats' or 'lme4'.
<code>prc_eng</code>	Then engine or extracting principal components. Options are 'stats', 'Gifi', and 'stats_Gifi'. The stats_Gifi engine uses <code>tidyselect::where(is.numeric)</code> to select the <code>pca_vars</code> for <code>stats::prcomp</code> and <code>-tidyselect::where(is.numeric)</code> to select the <code>pca_vars</code> for <code>Gifi::princals</code> . Read Rossiter (2021) for more on princals.
<code>pca_vars</code>	Variables to include in the principal component analysis. These variables will be swapped out for principal components
<code>n_pca_components</code>	The number of principal components to include in the model. If using a complex <code>prc_eng</code> (i.e., <code>stats_Gifi</code> ) then provide a named vector (i.e., <code>n_pca_components = c("stats" = 2, "Gifi" = 3)</code> ).
<code>norun_raw</code>	Include regression on raw variables if TRUE, exclude if FALSE.
<code>lpca_center</code>	Center data as in the Step-by-Step PCA vignette (Harvey & Hanson, 2022). Only numeric variables will be included in the centering. Parameter takes values 'all' to center raw and pca variables, 'raw' to only center variables for the raw variable model fitting, 'pca' to only center <code>pca_vars</code> before pca regression model fitting, and 'none' to skip lpca centering.
<code>lpca_scale</code>	Scale data as in the Step-by-Step PCA vignette. Only numeric variables will be included in the scaling. Parameter takes values 'all' to scale raw and pca variables, 'raw' to only scale variables for the raw variable model fitting, 'pca' to only scale <code>pca_vars</code> before pca regression model fitting, and 'none' to skip lpca scaling.
<code>lpca_undo</code>	Undo centering and scaling of <code>pca_vars</code> as in the Step-by-Step PCA vignette.
<code>gifi_transform</code>	Use Gifi optimal scaling to transform a set of variables. Parameter takes values 'none', 'all', 'raw', and 'pca'
<code>gifi_trans_vars</code>	A vector of variables to include in the Gifi optimal scaling transformation

<code>gifi_trans_dims</code>	Number of dimensions to extract in the Gifi optimal scaling transformation algorithm
<code>no_tresp</code>	When set to TRUE, <code>no_tresp</code> (No transform response) will exclude the response variable from pre-modeling and pre-pca transformations. Specifically, setting <code>no_tresp</code> to TRUE will exclude the response variable from the transformation specified in <code>lpca_center</code> and <code>lpca_scale</code> .
<code>miss_handler</code>	Choose how <code>swaprinc</code> handles missing data on the input data. Default is 'none'. Use 'omit' for complete case analysis.
<code>model_options</code>	Pass additional arguments to statistical modeling functions (i.e., <code>stats::lm</code> , <code>stats::glm</code> , <code>lme4::lmer</code> , <code>lme4::glmer</code> ) Default is 'noaddpars' (no additional parameters)
<code>prcomp_options</code>	Pass additional arguments to <code>stats::prcomp</code> for <code>prc_eng = 'stats'</code> and <code>prc_eng = 'stats_Gifi'</code> call. Default is 'noaddpars' (no additional parameters)
<code>gifi_princals_options</code>	Pass additional arguments to <code>Gifi::princals</code> for <code>prc_eng = 'Gifi'</code> and <code>prc_eng = 'stats_Gifi'</code> call. Default is 'noaddpars' (no additional parameters)
<code>gifi_trans_options</code>	Pass additional arguments to <code>Gifi::princals</code> for <code>gifi_transform</code> . Default is 'noaddpars' (no additional parameters)

## Value

A list with fitted models

## References

1. Rossiter, D. G. Nonlinear Principal Components Analysis: Multivariate Analysis with Optimal Scaling (MVAOS). (2021) [https://www.css.cornell.edu/faculty/dgr2/\\_static/files/R\\_html/NonlinearPCA.html](https://www.css.cornell.edu/faculty/dgr2/_static/files/R_html/NonlinearPCA.html)
2. Harvey, D. T., & Hanson, B. A. Step-by-Step PCA. (2022) [https://cran.r-project.org/package=LearnPCA/vignettes/Vig\\_03\\_Step\\_By\\_Step\\_PCA.pdf](https://cran.r-project.org/package=LearnPCA/vignettes/Vig_03_Step_By_Step_PCA.pdf)

## Examples

```
data(iris)
res <- swaprinc(iris,
  "Sepal.Length ~ Sepal.Width + Petal.Length + Petal.Width",
  pca_vars = c("Sepal.Width", "Petal.Length", "Petal.Width"),
  n_pca_components = 2)
```

# Index

compswap, [2](#)

swaprinc, [4](#)