

# Package: LearnVizLMM (via r-universe)

March 8, 2025

**Title** Learning and Communicating Linear Mixed Models Without Data

**Version** 1.0.0

**Description** Summarizes characteristics of linear mixed effects models without data or a fitted model by converting code for fitting `lmer()` from 'lme4' and `lme()` from 'nlme' into tables, equations, and visuals. Outputs can be used to learn how to fit linear mixed effects models in 'R' and to communicate about these models in presentations, manuscripts, and analysis plans.

**URL** <https://github.com/kzavez/LearnVizLMM>

**BugReports** <https://github.com/kzavez/LearnVizLMM/issues>

**License** GPL (>= 3)

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.2

**Imports** cli, DiagrammeR, DiagrammeRsvg, dplyr, rsvg, stringr

**Suggests** kableExtra, knitr, rmarkdown, testthat (>= 3.0.0)

**Config/testthat/edition** 3

**VignetteBuilder** knitr

**Config/pak/sysreqs** libglpk-dev make libicu-dev librsvg2-dev  
libxml2-dev libssl-dev libnode-dev libx11-dev

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

**RemoteUrl** <https://github.com/kzavez/learnvizlmm>

**RemoteRef** HEAD

**RemoteSha** 632d549679ef1b46cb25aecd93762a4d0525f73e

## Contents

<code>extract_equation</code> . . . . .	2
<code>extract_structure</code> . . . . .	3
<code>extract_variables</code> . . . . .	4

---

extract_equation	<i>Model equation in 'LaTeX' format</i>
------------------	---

---

### Description

extract\_equation() takes the `nlme::lme()` or `lme4::lmer()` code for fitting a linear mixed effect model and returns the corresponding model equation written in 'LaTeX' notation.

### Usage

```
extract_equation(
  model,
  cat_vars = NULL,
  cat_vars_nlevels = NULL,
  output_type = "latex"
)
```

### Arguments

model	Code for fitting a <code>nlme::lme()</code> or <code>lme4::lmer()</code> model given as a string.
cat_vars	Optional character vector of the names of categorical predictor variables included in the model. Default is NULL, which assumes that all predictor variables are numeric.
cat_vars_nlevels	Optional numeric vector of the number of levels (i.e. categories) for each variable in <code>cat_vars</code> . Must be a vector of same length as <code>cat_vars</code> . Values must be whole numbers greater than 1 and less than 10. Only applies if <code>cat_vars</code> is not NULL.
output_type	Output type can be "latex" (default), "string", or "none".

### Value

None (invisible NULL) (`output_type = "latex"`), a string (`output_type = "string"`), or no output (`output_type = "none"`).

### Examples

```
# Different ways to write the same lme model
extract_equation(model = "lme(score ~ age, random=~age|subject)")
extract_equation(model = "lme(score ~ age, random=list(subject=~age))")

# Correlated vs. Uncorrelated
extract_equation(model = "lmer(score ~ age + (age|subject))")
extract_equation(model = "lmer(score ~ age + (age||subject))")

# Add a categorical predictor and interaction
extract_equation(model = "lmer(score ~ age*treat + (age|subject))",
```

```
cat_vars = "treat",
cat_vars_nlevels = 3)
```

---

extract\_structure      *Image of the data structure*

---

## Description

extract\_structure generates an image of the multilevel data structure. It does this in two steps. First, characteristics of the group(s) or grouping factor(s) are identified via the model input or the n\_gf, gf\_description, and gf\_names inputs. Second, this information is used to run `DiagrammeR::grViz()`, which returns an image.

## Usage

```
extract_structure(
  model = NULL,
  n_gf = NULL,
  gf_description = NULL,
  gf_names = NULL,
  gf_nlevels = NULL,
  gf3_index = "i",
  label_levels = "yes",
  export_type = "print"
)
```

## Arguments

model	Code for fitting a <code>nlme::lme()</code> or <code>lme4::lmer()</code> model given as a string.
n_gf	Number of groups or grouping factors: 1, 2, or 3. Only applies if model is NULL.
gf_description	Description of the structure of the groups or grouping factors: "nested", "crossed", "crossed with nested", or "crossed within nested". Only applies if n_gf is greater than 1 and model is NULL.
gf_names	Character vector of the names of group(s) or grouping factor(s). For nested, order names by level from highest to lowest. Must be a vector of length equal to n_gf. Only applies if model is NULL.
gf_nlevels	Optional numeric or character vector of the number of levels for each group or grouping factor in the model or gf_names.
gf3_index	String for the index of the highest-level group or grouping factor. Only applies if n_gf is 3. Default is "i".
label_levels	Indicates whether levels of the data structure should be labeled on the left-hand side of the figure (default) or not (label_levels = "no").
export_type	Export type can be "print" (default), "png" to save as a PNG file, or "text" to get the input used to run <code>DiagrammeR::grViz()</code> .

**Value**

A PNG (export\_type = "png"), character (export\_type = "text"), or object of class htmlwidget that will print in the R console, within R Markdown documents, and within Shiny output bindings (export\_type = "print").

**Examples**

```
# Using the model input
extract_structure(model = "lme(Score ~ type, random=list(School=pdDiag(~1+type),Class=~1))")
extract_structure(model = "lme(Weight ~ Time, random=~Time|Subject, data)",
  gf_nlevels = 47)
extract_structure(model = "lmer(Strength ~ 1 + (1|Machine) + (1|Worker))",
  gf_nlevels = c("23", "J"))

# Using the n_gf, gf_description, and gf_names inputs
extract_structure(n_gf = 1,
  gf_names = "Subject")
extract_structure(n_gf = 3,
  gf_description = "nested",
  gf_names = c("District", "School", "Class"),
  gf_nlevels = c(8, 15, 5),
  label_levels = "no")
```

---

extract\_variables      *Roles of variables*

---

**Description**

extract\_variables() returns a data frame of information of the variables in a `nlme::lme()` or `lme4::lmer()` model. The columns of the data frame include: Effect (whether the effect is random or fixed), Group (group or grouping factor associated with random effects), Term (notation used to include the variable in the model), Description (description of the Term), and Parameter (parameter estimated when the model is fit).

**Usage**

```
extract_variables(model, cat_vars = NULL, cat_vars_nlevels = NULL)
```

**Arguments**

model	Code for fitting a <code>nlme::lme()</code> or <code>lme4::lmer()</code> model given as a string.
cat_vars	Optional character vector of the names of categorical predictor variables included in the model. Default is NULL, which assumes that all predictor variables are numeric.
cat_vars_nlevels	Optional numeric vector of the number of levels (i.e. categories) for each variable in <code>cat_vars</code> . Must be a vector of same length as <code>cat_vars</code> . Values must be whole numbers greater than 1 and less than 10. Only applies if <code>cat_vars</code> is not NULL.

**Value**

A data frame.

**Examples**

```
# lme()
extract_variables(model = "lme(Score~type,random=list(School=pdDiag(~1+type),Class=~1))",
  cat_vars = "type",
  cat_vars_nlevels = 2)
extract_variables(model = "lme(weight~1+Time+I(Time^2),random=~Time+I(Time^2)|ID)")

# lmer()
extract_variables(model = "lmer(Strength ~ 1 + (1|Machine) + (1|Worker))")
extract_variables(model = "lmer(score ~ age*treat + (age|subject))",
  cat_vars = "treat",
  cat_vars_nlevels = 3)
```

# Index

DiagrammeR::grViz(), 3

extract\_equation, 2

extract\_structure, 3

extract\_variables, 4

lme4::lmer(), 2–4

nlme::lme(), 2–4