Package ‘APCtools’

May 6, 2026

Title Routines for Descriptive and Model-Based APC Analysis

Version 1.0.8

Maintainer Alexander Bauer <baueralexander@posteo.de>

Description Age-Period-Cohort (APC) analyses are used to differentiate relevant drivers for long-term developments.
The 'APCtools' package offers visualization techniques and general routines to simplify the workflow of an APC analysis.
Sophisticated functions are available both for descriptive and regression model-based analyses.
For the former, we use density (or ridgeline) matrices and (hexagonally binned) heatmaps as innovative visualization
techniques building on the concept of Lexis diagrams.
Model-based analyses build on the separation of the temporal dimensions based on generalized additive models,
where a tensor product interaction surface (usually between age and period) is utilized
to represent the third dimension (usually cohort) on its diagonal.
Such tensor product surfaces can also be estimated while accounting for
further covariates in the regression model.
See Weigert et al. (2021) <doi:10.1177/1354816620987198> for methodological details.

License MIT + file LICENSE

URL https://bauer-alex.github.io/APCtools/

BugReports https://github.com/bauer-alex/APCtools/issues

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Imports ggpubr, checkmate, knitr, ggplot2, colorspace, dplyr, mgcv, scales, tidyr, stringr

Suggests testthat (>= 3.0.0), rmarkdown, covr

VignetteBuilder knitr

Depends R (>= 3.5)

Config/testthat/edition 3

NeedsCompilation no

Author Alexander Bauer [aut, cre] (ORCID: <https://orcid.org/0000-0003-3495-5131>), Maximilian Weigert [aut] (ORCID: <https://orcid.org/0000-0003-4400-134X>), Hawre Jalal [aut] (ORCID: <https://orcid.org/0000-0002-8224-6834>)

Repository CRAN

Date/Publication 2025-06-18 20:20:06 UTC

Contents

calc_density . . . 3
capitalize_firstLetter . . . 3
compute_marginalAPCeffects . . . 4
compute_xCoordinate . . . 4
compute_yCoordinate . . . 5
create_APCsummary . . . 5
create_groupVariable . . . 6
create_highlightDiagonalData . . . 7
create_modelSummary . . . 8
create_oneAPCsummaryTable . . . 9
drug_deaths . . . 10
ensure_segmentsInPlotRange . . . 11
extract_summary_linearEffects . . . 11
get_plotGAMobject . . . 12
gg_addReferenceLines . . . 12
gg_highlightDiagonals . . . 13
plot_1Dsmooth . . . 13
plot_APCheatmap . . . 15
plot_APChexamap . . . 18
plot_density . . . 20
plot_densityMatrix . . . 22
plot_density_categorical . . . 25
plot_density_metric . . . 25
plot_jointMarginalAPCeffects . . . 27
plot_linearEffects . . . 28
plot_marginalAPCeffects . . . 29
plot_partialAPCeffects . . . 31
plot_variable . . . 33
travel . . . 34

Index **36**

calc_density — *Internal helper to calculate the (group-specific) density of a variable*

Description

Internal helper function that is called in `plot_density` to calculate the density of a metric variable. If `plot_density` is called from within `plot_densityMatrix` (i.e., when some of the columns `c("age_group","period_group","cohort_group")` are part of the dataset, the density is computed individually for all respective APC groups.

Usage

```
calc_density(dat, y_var, weights_var = NULL, ...)
```

Arguments

`dat`	Dataset with columns `period` and `age` and the main variable specified through argument `y_var`.
`y_var`	Character name of the main variable to be plotted.
`weights_var`	Optional character name of a weights variable used to project the results in the sample to some population.
`...`	Additional arguments passed to `density`.

Value

Dataset with the calculated densities.

capitalize_firstLetter — *Internal function to capitalize the first letter of a character*

Description

Internal helper function to capitalize the first letter of a character value. The use case is to create a plot label like 'Age' from a variable name like 'age'.

Usage

```
capitalize_firstLetter(char)
```

Arguments

`char`	Character value whose first letter should be capitalized

compute_marginalAPCeffects

Internal helper to compute marginal APC effects and their confidence intervals

Description

Internal helper function to add lower and upper confidence boundaries pointwise

Usage

```
compute_marginalAPCeffects(dat, model, variable, plot_CI = FALSE)
```

Arguments

dat	Dataset containing predicted effects for a grid of all APC dimensions and covariates used in the model.
model	Model fitted with gam or bam.
variable	One of c("age","period","cohort"), specifying the temporal dimension for which the partial effect plots should be created.
plot_CI	Indicator if 95% confidence intervals for marginal APC effects should be computed. Defaults to FALSE.

Details

If the model was estimated with a log or logit link, the function automatically performs an exponential transformation of the effect.

compute_xCoordinate

Internal helper to tilt the x-axis for the hexamap plot

Description

Internal helper function to be called in plot_APChexamap, to tilt the x-axis for the hexamap plot.

Usage

```
compute_xCoordinate(period_vec)
```

Arguments

period_vec	Numeric vector of period values.

compute_yCoordinate *Internal helper to tilt the x-axis for the hexamap plot*

Description

Internal helper function to be called in plot_APChexamap, to tilt the x-axis for the hexamap plot.

Usage

```
compute_yCoordinate(period_vec, age_vec)
```

Arguments

period_vec	Numeric vector of period values.
age_vec	Numeric vector of age values.

create_APCsummary *Create a summary table for multiple estimated GAM models*

Description

Create a table to summarize the overall effect strengths of the age, period and cohort effects for models fitted with gam or bam. The output format can be adjusted by passing arguments to kable via the ... argument.

Usage

```
create_APCsummary(
  model_list,
  dat,
  digits = 2,
  apc_range = NULL,
  kable = TRUE,
  ...
)
```

Arguments

model_list	A list of regression models estimated with gam or bam. If the list is named, the names are used as labels. Can also be a single model object instead of a list.
dat	Dataset with columns period and age. If y_var is specified, the dataset must contain the respective column. If model is specified, the dataset must have been used for model estimation with gam or bam.
digits	Number of digits for numeric columns. Defaults to 2.

apc_range	Optional list with one or multiple elements with names
kable	Should the output be a table in kable style? Defaults to TRUE. "age","period","cohort" to filter the data. Each element should contain a numeric vector of values for the respective variable that should be kept in the data. All other values are deleted before producing the table.
...	Optional additional arguments passed to kable.

Details

If the model was estimated with a log or logit link, the function automatically performs an exponential transformation of the effect.

Value

Table created with kable.

Author(s)

Alexander Bauer <alexander.bauer@stat.uni-muenchen.de>

Examples

```
library(APCtools)
library(mgcv)

data(travel)

# create the summary table for one model
model_pure <- gam(mainTrip_distance ~ te(age, period), data = travel)
create_APCsummary(model_pure, dat = travel)

# create the summary table for multiple models
model_cov  <- gam(mainTrip_distance ~ te(age, period) + s(household_income),
                  data = travel)
model_list <- list("pure model"      = model_pure,
                   "covariate model" = model_cov)
create_APCsummary(model_list, dat = travel)
```

---

create_groupVariable *Internal helper to create a group variable as base for a density matrix*

Description

Internal helper function to create a group variable based on the categorization of either age, period or cohort. To be called from within plot_densityMatrix.

Usage

```
create_groupVariable(dat, APC_var, groups_list)
```

Arguments

dat	Dataset with a column "age", "period" or "cohort", dependent on the specified APC_var.
APC_var	One of c("age","period","cohort").
groups_list	A list with each element specifying the borders of one row or column in the density matrix. E.g., if the period should be visualized in decade columns from 1980 to 2009, specify groups_list = list(c(1980,1989), c(1990,1999), c(2000,2009)). The list can be named to specify labels for the categories.

Value

Vector for the grouping that can be added as additional column to the data.

---

create_highlightDiagonalData

Internal helper to create a dataset for ggplot2 to highlight diagonals

---

Description

Internal helper function to create a dataset for ggplot2 that can be used to highlight specific diagonals in a density matrix.

Usage

```
create_highlightDiagonalData(dat, highlight_diagonals)
```

Arguments

dat	Dataset with columns period and age and the main variable specified through argument y_var.
highlight_diagonals	Optional internal parameter which is only specified when plot_density is called from within plot_densityMatrix. See plot_densityMatrix for details.

`create_modelSummary` *Create model summary tables for multiple estimated GAM models*

Description

Create publication-ready summary tables of all linear and nonlinear effects for models fitted with `gam` or `bam`. The output format of the tables can be adjusted by passing arguments to `kable` via the `...` argument.

Usage

```
create_modelSummary(
  model_list,
  digits = 2,
  method_expTransform = "simple",
  ...
)
```

Arguments

`model_list`	list of APC models
`digits`	number of displayed digits
`method_expTransform`	One of `c("simple","delta")`, stating if standard errors and confidence interval limits should be transformed by a simple exp transformation or using the delta method. The delta method can be unstable in situations and lead to negative confidence interval limits. Only used when the model was estimated with a log or logit link.
`...`	additional arguments to `kable`

Details

If the model was estimated with a log or logit link, the function automatically performs an exponential transformation of the effects.

The table for linear coefficients includes the estimated coefficient (`coef`), the corresponding standard error (`se`), lower and upper limits of 95% confidence intervals (`CI_lower`, `CI_upper`) and the p-values for all coefficients apart from the intercept.

The table for nonlinear coefficients include the estimated degrees of freedom (`edf`) and the p-value for each estimate.

Value

List of tables created with `kable`.

Author(s)

Alexander Bauer <alexander.bauer@stat.uni-muenchen.de>

Examples

```
library(APCtools)
library(mgcv)

data(travel)
model <- gam(mainTrip_distance ~ te(age, period) + residence_region +
             household_size + s(household_income), data = travel)

create_modelSummary(list(model), dat = travel)
```

`create_oneAPCsummaryTable`

Internal helper to create a summary table for one estimated GAM model

---

Description

Internal helper function to be called in `create_APCsummary`. This function creates the summary table for one model estimated with `gam` or `bam`.

Usage

```
create_oneAPCsummaryTable(model, dat, apc_range = NULL)
```

Arguments

`model`	Optional regression model estimated with `gam` or `bam` to estimate a smoothed APC surface. Only used if `y_var` is not specified.
`dat`	Dataset with columns `period` and `age`. If `y_var` is specified, the dataset must contain the respective column. If `model` is specified, the dataset must have been used for model estimation with `gam` or `bam`.
`apc_range`	Optional list with one or multiple elements with names `"age"`,`"period"`,`"cohort"` to filter the data. Each element should contain a numeric vector of values for the respective variable that should be kept in the data. All other values are deleted.

Value

`data.frame` containing aggregated information on the individual effects.

`drug_deaths` *Drug deaths of white men in the United States*

Description

Dataset on the number of unintentional drug overdose deaths in the United States for each age group between 1999 and 2019, retrieved from the CDC WONDER Online Database. The data only cover white men.

Usage

```
data(drug_deaths)
```

Format

A dataframe containing

period Calendar year

age Age group.

deaths Number of observed unintentional drug overdose deaths in the respective age group and calendar year.

population Number of white men in the respective age group and calendar year in the U.S. population.

mortality_rate Drug overdose mortality rate for the respective age group and calendar year, reported as the number of deaths per 100,000 people. Calculated as `100000 * deaths / population`.

Details

The data were exported from the CDC WONDER Online Database (see link in references down below), based on the following settings:

- Group by *Year* and by *Single-Year Ages*
- Demographics: Gender *Male*; Ethnicity *White*
- Cause of death: *Drug / Alcohol Induced Causes*. Then select the more specific category *Drug poisonings (overdose) Unintentional (X40-X44)*.

References

Jalal, H., & Burke, D. S. (2020). Hexamaps for Age-Period-Cohort Data Visualization and Implementation in R. *Epidemiology (Cambridge, Mass.)*, 31(6), e47. doi:10.1097/EDE.0000000000001236.

Centers for Disease Control and Prevention, National Center for Health Statistics. Underlying Cause of Death 1999-2019 on CDC WONDER Online Database, released in 2020. Data are from the Multiple Cause of Death Files, 1999-2019, as compiled from data provided by the 57 vital statistics jurisdictions through the Vital Statistics Cooperative Program. Accessed at wonder.cdc.gov/ucd-icd10.html on 18 June 2025.

ensure_segmentsInPlotRange

Internal helper for gg_addReferenceLines to keep diagonal lines in the plot range

Description

Internal helper function to be called from within gg_addReferenceLines. This function takes the dataset prepared for adding diagonal reference lines in the plot, checks if some diagonals exceed the plot limits, cuts them accordingly, if necessary, and again returns the corrected dataset.

Usage

```
ensure_segmentsInPlotRange(dat_segments, plot_dat)
```

Arguments

dat_segments	Dataset containing information on the diagonal reference lines.
plot_dat	Dataset used for creating the heatmap.

extract_summary_linearEffects

Internal helper to extract summary of linear effects in a gam model

Description

Internal helper function to create a data.frame containing the linear effects summary of a model fitted with gam or bam.

Usage

```
extract_summary_linearEffects(model, method_expTransform = "simple")
```

Arguments

model	Model fitted with gam or bam.
method_expTransform	One of c("simple","delta"), stating if standard errors and confidence interval limits should be transformed by a simple exp transformation or using the delta method. The delta method can be unstable in situations and lead to negative confidence interval limits. Only used when the model was estimated with a log or logit link.

Details

If the model was estimated with a log or logit link, the function automatically performs an exponential transformation of the effect, see argument method_expTransform.

get_plotGAMobject — *Extract returned values of plot.gam() while suppressing creation of the plot*

Description

Internal helper function to extract the values returned of `plot.gam` while suppressing creation of the plot.

Usage

```
get_plotGAMobject(model)
```

Arguments

`model`	GAM model fitted with `gam` or `bam`.

gg_addReferenceLines — *Internal helper to add reference lines in an APC heatmap*

Description

Internal helper function to add reference lines in an APC heatmap (vertically, horizontally or diagonally). The function takes an existing list of ggplot objects, adds the specified reference lines in each plot and returns the edited ggplot list again. To be called from within `plot_APCheatmap`.

Usage

```
gg_addReferenceLines(
  gg_list,
  dimensions,
  plot_dat,
  markLines_list,
  markLines_displayLabels
)
```

Arguments

`gg_list`	Existing list of ggplot objects where the reference lines should be marked in each individual ggplot.
`dimensions`	Character vector specifying the two APC dimensions that should be visualized along the x-axis and y-axis. Defaults to `c("period","age")`.
`plot_dat`	Dataset used for creating the heatmap.

`markLines_list` Optional list that can be used to highlight the borders of specific age groups, time intervals or cohorts. Each element must be a numeric vector of values where horizontal, vertical or diagonal lines should be drawn (depends on which APC dimension is displayed on which axis). The list can maximally have three elements and must have names out of `c("age","period","cohort")`.

`markLines_displayLabels`
Optional character vector defining for which dimensions the lines defined through `markLines_list` should be marked by a respective label. The vector should be a subset of `c("age","period","cohort")`, or NULL to suppress all labels. Defaults to `c("age","period","cohort")`.

---

gg_highlightDiagonals *Internal helper to add the diagonal highlighting to a ggplot*

Description

Internal helper function to highlight diagonals in a density matrix. The function takes an existing ggplot object, adds the diagonal highlighting and returns the edited ggplot object again.

Usage

```
gg_highlightDiagonals(gg, dat, dat_highlightDiagonals)
```

Arguments

`gg` Existing ggplot object to which the diagonal highlighting should be added.

`dat` Dataset with columns `period` and `age` and the main variable specified through argument `y_var`.

`dat_highlightDiagonals`
Dataset created by `create_highlightDiagonalData` to highlight specific diagonals in a density matrix.

---

plot_1Dsmooth *Plot 1D smooth effects for gam models*

Description

Plots 1D smooth effects for a GAM model fitted with `gam` or `bam`.

Usage

```
plot_1Dsmooth(
  model,
  plot_ci = TRUE,
  select,
  alpha = 0.05,
  ylim = NULL,
  method_expTransform = "simple",
  return_plotData = FALSE
)
```

Arguments

`model`	GAM model fitted with `gam` or `bam`.
`plot_ci`	If `TRUE` CIs are plotted. Only used if `plot_type = 1`.
`select`	Index of smooth term to be plotted.
`alpha`	`(1-alpha)` CIs are calculated. The default 0.05 leads to 95% CIs.
`ylim`	Optional limits of the y-axis.
`method_expTransform`	One of `c("simple","delta")`, stating if standard errors and confidence interval limits should be transformed by a simple exp transformation or using the delta method. The delta method can be unstable in situations and lead to negative confidence interval limits. Only used when the model was estimated with a log or logit link.
`return_plotData`	If TRUE, the dataset prepared for plotting is returned. Defaults to FALSE.

Details

If the model was estimated with a log or logit link, the function automatically performs an exponential transformation of the effect, see argument `method_expTransform`.

Value

ggplot object

Author(s)

Alexander Bauer <alexander.bauer@stat.uni-muenchen.de>

Examples

```
library(APCtools)
library(mgcv)

data(travel)
model <- gam(mainTrip_distance ~ te(age, period) + residence_region +
             household_size + s(household_income), data = travel)
```

```
plot_1Dsmooth(model, select = 2)
```

---

plot_APCheatmap *Heatmap of an APC surface*

Description

Plot the heatmap of an APC structure. The function can be used in two ways: Either to plot the observed mean structure of a metric variable, by specifying `dat` and the variable `y_var`, or by specifying `dat` and the `model` object, to plot some mean structure represented by an estimated two-dimensional tensor product surface. The model must be estimated with `gam` or `bam`.

Usage

```
plot_APCheatmap(
  dat,
  y_var = NULL,
  model = NULL,
  dimensions = c("period", "age"),
  apc_range = NULL,
  bin_heatmap = TRUE,
  bin_heatmapGrid_list = NULL,
  markLines_list = NULL,
  markLines_displayLabels = c("age", "period", "cohort"),
  y_var_logScale = FALSE,
  plot_CI = TRUE,
  method_expTransform = "simple",
  legend_limits = NULL,
  legend_title = NULL
)
```

Arguments

`dat`	Dataset with columns `period` and `age`. If `y_var` is specified, the dataset must contain the respective column. If `model` is specified, the dataset must have been used for model estimation with `gam` or `bam`.
`y_var`	Optional character name of a metric variable to be plotted.
`model`	Optional regression model estimated with `gam` or `bam` to estimate a smoothed APC surface. Only used if `y_var` is not specified.
`dimensions`	Character vector specifying the two APC dimensions that should be visualized along the x-axis and y-axis. Defaults to `c("period","age")`.
`apc_range`	Optional list with one or multiple elements with names `"age","period","cohort"` to filter the data. Each element should contain a numeric vector of values for the respective variable that should be kept in the data. All other values are deleted.

`bin_heatmap`, `bin_heatmapGrid_list`
: `bin_heatmap` indicates if the heatmap surface should be binned. Defaults to TRUE. If TRUE, the binning grid borders are defined by `bin_heatmapGrid_list`. This is a list with each element a numeric vector and a name out of `c("age","period","cohort")`. Can maximally have three elements. Defaults to NULL, where the heatmap is binned in 5 year steps along the x-axis and the y-axis.

`markLines_list`
: Optional list that can be used to highlight the borders of specific age groups, time intervals or cohorts. Each element must be a numeric vector of values where horizontal, vertical or diagonal lines should be drawn (depends on which APC dimension is displayed on which axis). The list can maximally have three elements and must have names out of `c("age","period","cohort")`.

`markLines_displayLabels`
: Optional character vector defining for which dimensions the lines defined through `markLines_list` should be marked by a respective label. The vector should be a subset of `c("age","period","cohort")`, or NULL to suppress all labels. Defaults to `c("age","period","cohort")`.

`y_var_logScale`
: Indicator if `y_var` should be log10 transformed. Only used if `y_var` is specified. Defaults to FALSE.

`plot_CI`
: Indicator if the confidence intervals should be plotted. Only used if `y_var` is not specified. Defaults to TRUE.

`method_expTransform`
: One of `c("simple","delta")`, stating if confidence interval limits should be transformed by a simple exp transformation or using the delta method. The delta method can be unstable in situations and lead to negative confidence interval limits. Only used when the model was estimated with a log or logit link and confidence intervals are supposed to be plotted. Defaults to `simple`.

`legend_limits`
: Optional numeric vector passed as argument `limits` to `scale_fill_gradient2`.

`legend_title`
: Optional character legend title.

Details

See also `plot_APChexamap` to plot a hexagonal heatmap with adapted axes.

If the plot is created based on the `model` object and the model was estimated with a log or logit link, the function automatically performs an exponential transformation of the effect.

Value

Plot grid created with `ggarrange` (if `plot_CI` is TRUE) or a `ggplot2` object (if `plot_CI` is FALSE).

Author(s)

Alexander Bauer <alexander.bauer@stat.uni-muenchen.de>, Maximilian Weigert <maximilian.weigert@stat.uni-m

References

Weigert, M., Bauer, A., Gernert, J., Karl, M., Nalmpatian, A., Küchenhoff, H., and Schmude, J. (2021). Semiparametric APC analysis of destination choice patterns: Using generalized additive models to quantify the impact of age, period, and cohort on travel distances. *Tourism Economics*. doi:10.1177/1354816620987198.

See Also

plot_APChexamap

Examples

```
library(APCtools)
library(mgcv)

data(travel)

# variant A: plot observed mean structures
# observed heatmap
plot_APCheatmap(dat = travel, y_var = "mainTrip_distance",
                bin_heatmap = FALSE, y_var_logScale = TRUE)

# with binning
plot_APCheatmap(dat = travel, y_var = "mainTrip_distance",
                bin_heatmap = TRUE, y_var_logScale = TRUE)

# variant B: plot some smoothed, estimated mean structure
model <- gam(mainTrip_distance ~ te(age, period) + residence_region +
             household_size + s(household_income), data = travel)

# plot the smooth tensor product surface
plot_APCheatmap(dat = travel, model = model, bin_heatmap = FALSE, plot_CI = FALSE)

# ... same plot including the confidence intervals
plot_APCheatmap(dat = travel, model = model, bin_heatmap = FALSE)

# the APC dimensions can be flexibly assigned to the x-axis and y-axis
plot_APCheatmap(dat = travel, model = model, dimensions = c("age","cohort"),
                bin_heatmap = FALSE, plot_CI = FALSE)

# add some reference lines
plot_APCheatmap(dat = travel, model = model, bin_heatmap = FALSE, plot_CI = FALSE,
                markLines_list = list(cohort = c(1910,1939,1955,1980)))

# default binning of the tensor product surface in 5-year-blocks
plot_APCheatmap(dat = travel, model = model, plot_CI = FALSE)

# manual binning
manual_binning <- list(period = seq(min(travel$period, na.rm = TRUE) - 1,
                                    max(travel$period, na.rm = TRUE), by = 5),
                       cohort = seq(min(travel$period - travel$age, na.rm = TRUE) - 1,
                                    max(travel$period - travel$age, na.rm = TRUE), by = 10))
plot_APCheatmap(dat = travel, model = model, plot_CI = FALSE,
                bin_heatmapGrid_list = manual_binning)
```

plot_APChexamap *Hexamap of an APC surface*

Description

Plot the heatmap of an APC structure using a hexagon-based plot with adapted axes. In this way, the one temporal dimension that is represented by the diagonal structure is visually not underrepresented compared to the other two dimensions on the x-axis and y-axis.
The function can be used in two ways: Either to plot the observed mean structure of a metric variable, by specifying `dat` and the variable `y_var`, or by specifying `dat` and the `model` object, to plot some mean structure represented by an estimated two-dimensional tensor product surface. The model must be estimated with `gam` or `bam`.

Usage

```
plot_APChexamap(
  dat,
  y_var = NULL,
  model = NULL,
  apc_range = NULL,
  y_var_logScale = FALSE,
  obs_interval = 1,
  iso_interval = 5,
  color_vec = NULL,
  color_range = NULL,
  line_width = 0.5,
  line_color = gray(0.5),
  label_size = 0.5,
  label_color = "black",
  legend_title = NULL
)
```

Arguments

`dat`	Dataset with columns `period` and `age`. If `y_var` is specified, the dataset must contain the respective column. If `model` is specified, the dataset must have been used for model estimation with `gam` or `bam`.
`y_var`	Optional character name of a metric variable to be plotted.
`model`	Optional regression model estimated with `gam` or `bam` to estimate a smoothed APC surface. Only used if `y_var` is not specified.
`apc_range`	Optional list with one or multiple elements with names `"age"`,`"period"`,`"cohort"` to filter the data. Each element should contain a numeric vector of values for the respective variable that should be kept in the data. All other values are deleted.
`y_var_logScale`	Indicator if `y_var` should be log10 transformed. Only used if `y_var` is specified. Defaults to FALSE.

obs_interval	Numeric specifying the interval width based on which the data is spaced. Only used if y_var is specified. Defaults to 1, i.e. observations each year.
iso_interval	Numeric specifying the interval width between the isolines along each axis. Defaults to 5.
color_vec	Optional character vector of color names, specifying the color continuum.
color_range	Optional numeric vector with two elements, specifying the ends of the color scale in the legend.
line_width	Line width of the isolines. Defaults to 0.5.
line_color	Character color name for the isolines. Defaults to gray.
label_size	Size of the labels along the axes. Defaults to 0.5.
label_color	Character color name for the labels along the axes.
legend_title	Optional character title for the legend.

Details

See also plot_APCheatmap to plot a regular heatmap.

If the plot is created based on the model object and the model was estimated with a log or logit link, the function automatically performs an exponential transformation of the effect.

Value

Creates a plot with base R functions (not ggplot2).

Author(s)

Hawre Jalal <hjalal@pitt.edu>, Alexander Bauer <alexander.bauer@stat.uni-muenchen.de>

References

Jalal, H., Burke, D. (2020). Hexamaps for Age–Period–Cohort Data Visualization and Implementation in R. *Epidemiology*, 31 (6), e47-e49. doi: 10.1097/EDE.0000000000001236.

See Also

plot_APCheatmap

Examples

```
library(APCtools)
library(mgcv)
library(dplyr)

data(drug_deaths)

# restrict to data where the mortality rate is available
drug_deaths <- drug_deaths %>% filter(!is.na(mortality_rate))

# hexamap of an observed structure
```

```
plot_APChexamap(dat         = drug_deaths,
                y_var       = "mortality_rate",
                color_range = c(0,40))

# hexamap of a smoothed structure
model <- gam(mortality_rate ~ te(age, period, bs = "ps", k = c(8,8)),
             data = drug_deaths)

plot_APChexamap(dat = drug_deaths, model = model)
```

`plot_density` *Plot the density of one metric or categorical variable*

Description

Create a density plot or a boxplot of one metric variable or a barplot of one categorical variable, based on a specific subset of the data.

Usage

```
plot_density(
  dat,
  y_var,
  plot_type = "density",
  apc_range = NULL,
  highlight_diagonals = NULL,
  y_var_cat_breaks = NULL,
  y_var_cat_labels = NULL,
  weights_var = NULL,
  log_scale = FALSE,
  xlab = NULL,
  ylab = NULL,
  legend_title = NULL,
  ...
)
```

Arguments

`dat`	Dataset with columns `period` and `age` and the main variable specified through argument `y_var`.
`y_var`	Character name of the main variable to be plotted.
`plot_type`	One of `c("density","boxplot")`. Only used if the `y_var` column is metric.
`apc_range`	Optional list with one or multiple elements with names `"age","period","cohort"` to filter the data. Each element should contain a numeric vector of values for the respective variable that should be kept in the data. All other values are deleted.

highlight_diagonals
: Optional internal parameter which is only specified when plot_density is called from within plot_densityMatrix. See plot_densityMatrix for details.

y_var_cat_breaks
: Optional numeric vector of breaks to categorize y_var based on calling function cut. Only used to highlight the categories based on different colors. And only used if the y_var column is numeric.

y_var_cat_labels
: Optional character vector for the names of the categories that were defined based on y_var_cat_breaks. The length of this vector must be one shorter than length(y_var_cat_breaks). Only used if the y_var column is numeric.

weights_var
: Optional character name of a weights variable used to project the results in the sample to some population.

log_scale
: Indicator if the main variable should be log10 transformed. Only used if the y_var column is numeric. Defaults to FALSE.

xlab, ylab, legend_title
: Optional plot annotations.

...
: Additional arguments passed to density.

Details

If plot_density is called internally from within plot_densityMatrix (i.e., if the dataset contains some of the columns c("age_group","period_group","cohort_group")), this function will calculate the metric densities individually for these groups.

Value

ggplot object

Author(s)

Alexander Bauer <alexander.bauer@stat.uni-muenchen.de>, Maximilian Weigert <maximilian.weigert@stat.uni-m

Examples

```
library(APCtools)
data(travel)

plot_density(dat = travel, y_var = "mainTrip_distance")

plot_density(dat = travel, y_var = "mainTrip_distance")
```

plot_densityMatrix *Create a matrix of density plots*

Description

This function creates a matrix of individual density plots (i.e., a *ridgeline matrix*) or boxplots (for metric variables) or of individual barplots (for categorical variables). The age, period or cohort information can each either be plotted on the x-axis or the y-axis.

Usage

```
plot_densityMatrix(
  dat,
  y_var,
  dimensions = c("period", "age"),
  age_groups = NULL,
  period_groups = NULL,
  cohort_groups = NULL,
  plot_type = "density",
  highlight_diagonals = NULL,
  y_var_cat_breaks = NULL,
  y_var_cat_labels = NULL,
  weights_var = NULL,
  log_scale = FALSE,
  legend_title = NULL,
  ...
)
```

Arguments

`dat` Dataset with columns `period` and `age` and the main variable specified through argument `y_var`.

`y_var` Character name of the main variable to be plotted.

`dimensions` Character vector specifying the two APC dimensions that should be visualized along the x-axis and y-axis. Defaults to `c("period","age")`.

`age_groups`, `period_groups`, `cohort_groups`
Each a list. Either containing purely scalar values or with each element specifying the two borders of one row or column in the density matrix. E.g., if the period should be visualized in decade columns from 1980 to 2009, specify `period_groups = list(c(1980,1989), c(1990,1999), c(2000,2009))`. The list can be named to specify labels for the categories. Only the two arguments must be passed that were specified by the `dimensions` argument.

`plot_type` One of `c("density","boxplot")`. Only used if the `y_var` column is metric.

`highlight_diagonals`
Optional list to define diagonals in the density that should be highlighted with different colors. Each list element should be a numeric vector stating the index

of the diagonals (counted from the top left) that should be highlighted in the same color. If the list is named, the names are used as legend labels.

`y_var_cat_breaks`
: Optional numeric vector of breaks to categorize `y_var` based on calling function `cut`. Only used to highlight the categories based on different colors. And only used if the `y_var` column is numeric.

`y_var_cat_labels`
: Optional character vector for the names of the categories that were defined based on `y_var_cat_breaks`. The length of this vector must be one shorter than `length(y_var_cat_breaks)`. Only used if the `y_var` column is numeric.

`weights_var`
: Optional character name of a weights variable used to project the results in the sample to some population.

`log_scale`
: Indicator if the main variable should be log10 transformed. Only used if the `y_var` column is numeric. Defaults to FALSE.

`legend_title`
: Optional plot annotation.

`...`
: Additional arguments passed to `plot_density`.

Value

ggplot object

Author(s)

Alexander Bauer <alexander.bauer@stat.uni-muenchen.de>, Maximilian Weigert <maximilian.weigert@stat.uni-m

References

Weigert, M., Bauer, A., Gernert, J., Karl, M., Nalmpatian, A., Küchenhoff, H., and Schmude, J. (2021). Semiparametric APC analysis of destination choice patterns: Using generalized additive models to quantify the impact of age, period, and cohort on travel distances. *Tourism Economics*. doi:10.1177/1354816620987198.

Examples

```
library(APCtools)

# define categorizations for the main trip distance
dist_cat_breaks <- c(1,500,1000,2000,6000,100000)
dist_cat_labels <- c("< 500 km","500 - 1,000 km", "1,000 - 2,000 km",
                     "2,000 - 6,000 km", "> 6,000 km")

age_groups    <- list(c(80,89),c(70,79),c(60,69),c(50,59),c(40,49),c(30,39),c(20,29))
period_groups <- list(c(1970,1979),c(1980,1989),c(1990,1999),c(2000,2009),c(2010,2019))
cohort_groups <- list(c(1980,1989),c(1970,1979),c(1960,1969),c(1950,1959),c(1940,1949),
                      c(1930,1939),c(1920,1929))

plot_densityMatrix(dat              = travel,
                   y_var            = "mainTrip_distance",
                   age_groups       = age_groups,
```

```
                    period_groups    = period_groups,
                    log_scale        = TRUE)


# highlight two cohorts
plot_densityMatrix(dat                 = travel,
                    y_var               = "mainTrip_distance",
                    age_groups          = age_groups,
                    period_groups       = period_groups,
                    highlight_diagonals = list(8, 10),
                    log_scale           = TRUE)

# also mark different distance categories
plot_densityMatrix(dat              = travel,
                    y_var            = "mainTrip_distance",
                    age_groups       = age_groups,
                    period_groups    = period_groups,
                    log_scale        = TRUE,
                    y_var_cat_breaks = dist_cat_breaks,
                    y_var_cat_labels = dist_cat_labels,
                    highlight_diagonals = list(8, 10),
                    legend_title     = "Distance category")

# flexibly assign the APC dimensions to the x-axis and y-axis
plot_densityMatrix(dat              = travel,
                    y_var            = "mainTrip_distance",
                    dimensions       = c("period","cohort"),
                    period_groups    = period_groups,
                    cohort_groups    = cohort_groups,
                    log_scale        = TRUE,
                    y_var_cat_breaks = dist_cat_breaks,
                    y_var_cat_labels = dist_cat_labels,
                    legend_title     = "Distance category")

# use boxplots instead of densities
plot_densityMatrix(dat           = travel,
                    y_var         = "mainTrip_distance",
                    plot_type     = "boxplot",
                    age_groups    = age_groups,
                    period_groups = period_groups,
                    log_scale     = TRUE,
                    highlight_diagonals = list(8, 10))

# plot categorical variables instead of metric ones
plot_densityMatrix(dat                 = travel,
                    y_var               = "household_size",
                    age_groups          = age_groups,
                    period_groups       = period_groups,
                    highlight_diagonals = list(8, 10))
```

plot_density_categorical

Internal helper to plot a categorical density

Description

Internal helper function to plot one categorical density, to be called from within `plot_density`.

Usage

```
plot_density_categorical(
  dat,
  y_var,
  dat_highlightDiagonals = NULL,
  weights_var = NULL,
  xlab = NULL,
  ylab = NULL
)
```

Arguments

`dat`	Dataset with columns `period` and `age` and the main variable specified through argument `y_var`.
`y_var`	Character name of the main variable to be plotted.
`dat_highlightDiagonals`	Optional dataset created by `create_highlightDiagonalData` to highlight specific diagonals in a density matrix.
`weights_var`	Optional character name of a weights variable used to project the results in the sample to some population.
`xlab, ylab`	Optional plot annotations.

plot_density_metric

Internal helper to plot a metric density

Description

Internal helper function to plot one metric density, to be called from within `plot_density`.

Usage

```
plot_density_metric(
  dat,
  y_var,
  plot_type = "density",
  dat_highlightDiagonals = NULL,
  y_var_cat_breaks = NULL,
  y_var_cat_labels = NULL,
  weights_var = NULL,
  log_scale = FALSE,
  xlab = NULL,
  ylab = NULL,
  legend_title = NULL,
  ...
)
```

Arguments

`dat` Dataset with columns `period` and `age` and the main variable specified through argument `y_var`.

`y_var` Character name of the main variable to be plotted.

`plot_type` One of `c("density","boxplot")`. Only used if the `y_var` column is metric.

`dat_highlightDiagonals` Optional dataset created by `create_highlightDiagonalData` to highlight specific diagonals in a density matrix.

`y_var_cat_breaks` Optional numeric vector of breaks to categorize `y_var` based on calling function `cut`. Only used to highlight the categories based on different colors. And only used if the `y_var` column is numeric.

`y_var_cat_labels` Optional character vector for the names of the categories that were defined based on `y_var_cat_breaks`. The length of this vector must be one shorter than `length(y_var_cat_breaks)`. Only used if the `y_var` column is numeric.

`weights_var` Optional character name of a weights variable used to project the results in the sample to some population.

`log_scale` Indicator if the main variable should be log10 transformed. Only used if the `y_var` column is numeric. Defaults to FALSE.

`xlab`, `ylab`, `legend_title` Optional plot annotations.

`...` Additional arguments passed to `density`.

plot_jointMarginalAPCeffects

Joint plot to compare the marginal APC effects of multiple models

Description

This function creates a joint plot of the marginal APC effects of multiple estimated models. It creates a plot with one pane per age, period and cohort effect, each containing one lines for each estimated model.

Usage

```
plot_jointMarginalAPCeffects(
  model_list,
  dat,
  vlines_list = NULL,
  ylab = NULL,
  ylim = NULL,
  plot_CI = FALSE
)
```

Arguments

`model_list`	A list of regression models estimated with `gam` or `bam`. If the list is named, the names are used as labels. Can also be a single model object instead of a list.
`dat`	Dataset with columns `period` and `age`. If `y_var` is specified, the dataset must contain the respective column. If `model` is specified, the dataset must have been used for model estimation with `gam` or `bam`.
`vlines_list`	Optional list that can be used to highlight the borders of specific age groups, time intervals or cohorts. Each element must be a numeric vector of values on the x-axis where vertical lines should be drawn. The list can maximally have three elements and must have names out of `c("age","period","cohort"`.
`ylab`, `ylim`	Optional ggplot2 styling arguments.
`plot_CI`	Indicator if 95% confidence intervals should be plotted. Defaults to FALSE.

Details

If the model was estimated with a log or logit link, the function automatically performs an exponential transformation of the effect.

Since the plot output created by the function is no `ggplot2` object, but an object created with `ggpubr::ggarrange`, the overall theme of the plot cannot be changed by adding the theme in the form of '`plot_jointMarginalAPCeffects(...) + theme_minimal(...)`'. Instead, you can call `theme_set(theme_minimal(...))` as an individual call before calling `plot_jointMarginalAPCeffects(...)`. The latter function will then use this global plotting theme.

Value

Plot grid created with ggarrange.

Author(s)

Alexander Bauer <alexander.bauer@stat.uni-muenchen.de>, Maximilian Weigert <maximilian.weigert@stat.uni-m

Examples

```
library(APCtools)
library(mgcv)

data(travel)

# plot marginal effects of one model
model_pure <- gam(mainTrip_distance ~ te(age, period), data = travel)
plot_jointMarginalAPCeffects(model_pure, dat = travel)

# plot marginal effects of multiple models
model_cov  <- gam(mainTrip_distance ~ te(age, period) + s(household_income),
                  data = travel)
model_list <- list("pure model"      = model_pure,
                   "covariate model" = model_cov)
plot_jointMarginalAPCeffects(model_list, dat = travel)

# mark specific cohorts
plot_jointMarginalAPCeffects(model_list, dat = travel,
                             vlines_list = list("cohort" = c(1966.5,1982.5,1994.5)))
```

---

plot_linearEffects *Plot linear effects of a gam in an effect plot*

Description

Create an effect plot of linear effects of a model fitted with gam or bam.

Usage

```
plot_linearEffects(
  model,
  variables = NULL,
  return_plotData = FALSE,
  refCat = FALSE,
  ...
)
```

Arguments

`model`	Model fitted with `gam` or `bam`.
`variables`	Optional character vector of variable names specifying which effects should be plotted. The order of the vector corresponds to the order in the effect plot. If the argument is not specified, all linear effects are plotted according to the order of their appearance in the model output.
`return_plotData`	If TRUE, the dataset prepared for plotting is returned. Defaults to FALSE.
`refCat`	If TRUE, reference categories are added to the output for categorical covariates. Defaults to FALSE.
`...`	Additional arguments passed to `extract_summary_linearEffects`.

Details

If the model was estimated with a log or logit link, the function automatically performs an exponential transformation of the effect.

Value

ggplot object

Author(s)

Alexander Bauer <alexander.bauer@stat.uni-muenchen.de>

Examples

```
library(APCtools)
library(mgcv)

data(travel)
model <- gam(mainTrip_distance ~ te(age, period) + residence_region +
             household_size + s(household_income), data = travel)

plot_linearEffects(model)
```

---

plot_marginalAPCeffects

Plot of marginal APC effects based on an estimated GAM model

Description

Plot the marginal effect of age, period or cohort, based on an APC model estimated as a semiparametric additive regression model with `gam` or `bam`. This function is a simple wrapper to `plot_partialAPCeffects`, called with argument `hide_partialEffects = TRUE`.

Usage

```
plot_marginalAPCeffects(
  model,
  dat,
  variable = "age",
  vlines_vec = NULL,
  plot_CI = FALSE,
  return_plotData = FALSE
)
```

Arguments

`model`	Optional regression model estimated with `gam` or `bam` to estimate a smoothed APC surface. Only used if `y_var` is not specified.
`dat`	Dataset with columns `period` and `age`. If `y_var` is specified, the dataset must contain the respective column. If `model` is specified, the dataset must have been used for model estimation with `gam` or `bam`.
`variable`	One of `c("age","period","cohort")`, specifying the temporal dimension for which the partial effect plots should be created.
`vlines_vec`	Optional numeric vector of values on the x-axis where vertical lines should be drawn. Can be used to highlight the borders of specific age groups, time intervals or cohorts.
`plot_CI`	Indicator if 95% confidence intervals should be plotted. Defaults to FALSE.
`return_plotData`	If TRUE, a list of the datasets prepared for plotting is returned instead of the ggplot object. The list contains one dataset each for the overall effect (= evaluations of the APC surface to plot the partial effects) and for each marginal APC effect (no matter the specified value of the argument `variable`). Defaults to FALSE.

Value

ggplot object

Author(s)

Alexander Bauer <alexander.bauer@stat.uni-muenchen.de>, Maximilian Weigert <maximilian.weigert@stat.uni-m

References

Weigert, M., Bauer, A., Gernert, J., Karl, M., Nalmpatian, A., Küchenhoff, H., and Schmude, J. (2021). Semiparametric APC analysis of destination choice patterns: Using generalized additive models to quantify the impact of age, period, and cohort on travel distances. *Tourism Economics*. doi:10.1177/1354816620987198.

Examples

```
library(APCtools)
library(mgcv)

data(travel)
model <- gam(mainTrip_distance ~ te(age, period), data = travel)

plot_marginalAPCeffects(model, dat = travel, variable = "age")

# mark specific cohorts
plot_marginalAPCeffects(model, dat = travel, variable = "cohort",
                        vlines_vec = c(1966.5,1982.5,1994.5))
```

---

plot_partialAPCeffects

Partial APC plots based on an estimated GAM model

Description

Create the partial APC plots based on an APC model estimated as a semiparametric additive regression model with gam or bam.

Usage

```
plot_partialAPCeffects(
  model,
  dat,
  variable = "age",
  hide_partialEffects = FALSE,
  vlines_vec = NULL,
  plot_CI = FALSE,
  return_plotData = FALSE
)
```

Arguments

model	Optional regression model estimated with gam or bam to estimate a smoothed APC surface. Only used if y_var is not specified.
dat	Dataset with columns period and age. If y_var is specified, the dataset must contain the respective column. If model is specified, the dataset must have been used for model estimation with gam or bam.
variable	One of c("age","period","cohort"), specifying the temporal dimension for which the partial effect plots should be created.
hide_partialEffects	If TRUE, only the marginal effect will be plotted. Defaults to FALSE.

`vlines_vec` Optional numeric vector of values on the x-axis where vertical lines should be drawn. Can be used to highlight the borders of specific age groups, time intervals or cohorts.

`plot_CI` Indicator if 95% confidence intervals for marginal APC effects should be plotted. Only used if `hide_partialEffects` is set to TRUE. Defaults to FALSE.

`return_plotData` If TRUE, a list of the datasets prepared for plotting is returned instead of the ggplot object. The list contains one dataset each for the overall effect (= evaluations of the APC surface to plot the partial effects) and for each marginal APC effect (no matter the specified value of the argument `variable`). Defaults to FALSE.

Details

If the model was estimated with a log or logit link, the function automatically performs an exponential transformation of the effect.

Value

ggplot object (if `hide_partialEffects` is TRUE) or a plot grid created with `ggarrange` (if FALSE).

Author(s)

Alexander Bauer <alexander.bauer@stat.uni-muenchen.de>, Maximilian Weigert <maximilian.weigert@stat.uni-m

References

Weigert, M., Bauer, A., Gernert, J., Karl, M., Nalmpatian, A., Küchenhoff, H., and Schmude, J. (2021). Semiparametric APC analysis of destination choice patterns: Using generalized additive models to quantify the impact of age, period, and cohort on travel distances. *Tourism Economics*. doi:10.1177/1354816620987198.

Examples

```
library(APCtools)
library(mgcv)

data(travel)
model <- gam(mainTrip_distance ~ te(age, period), data = travel)

plot_partialAPCeffects(model, dat = travel, variable = "age")

# mark specific cohorts
plot_partialAPCeffects(model, dat = travel, variable = "cohort",
                       vlines_vec = c(1966.5,1982.5,1994.5))
```

plot_variable *Distribution plot of one variable against one APC dimension*

Description

Plot the distribution of one variable in the data against age, period or cohort. Creates a bar plot for categorical variables (see argument geomBar_position) and boxplots or a line plot of median values for metric variables (see plot_type).

Usage

```
plot_variable(
  dat,
  y_var,
  apc_dimension = "period",
  log_scale = FALSE,
  plot_type = "boxplot",
  geomBar_position = "fill",
  legend_title = NULL,
  ylab = NULL,
  ylim = NULL
)
```

Arguments

dat	Dataset containing columns age and period.
y_var	Character name of the variable to plot.
apc_dimension	One of c("age","period","cohort"). Defaults to "period".
log_scale	Indicator if the visualized variable should be log10 transformed. Only used if the variable is numeric. Defaults to FALSE.
plot_type	One of c("boxplot","line","line-points"), specifying if boxplots or a line plot of median values should be drawn for metric variables. "line-points" adds points to the line plot where observations are available.
geomBar_position	Value passed to geom_bar as position argument. Only used if the visualized variable is categorical. Defaults to "fill".
legend_title	Optional character title for the legend which is drawn for categorical variables.
ylab, ylim	Optional arguments for styling the ggplot.

Value

ggplot object

Author(s)

Alexander Bauer <alexander.bauer@stat.uni-muenchen.de>

Examples

```
library(APCtools)
data(travel)

# plot a metric variable
plot_variable(dat = travel, y_var = "mainTrip_distance",
              apc_dimension = "period", log_scale = TRUE)
plot_variable(dat = travel, y_var = "mainTrip_distance",
              apc_dimension = "period", log_scale = TRUE, plot_type = "line")

# plot a categorical variable
plot_variable(dat = travel, y_var = "household_size", apc_dimension = "period")
plot_variable(dat = travel, y_var = "household_size", apc_dimension = "period",
              geomBar_position = "stack")
```

`travel` *Data from the German Reiseanalyse survey*

Description

This dataset from the Reiseanalyse survey comprises travel information on German travelers between 1971 and 2018. Data were collected in a yearly repeated cross-sectional survey of German pleasure travels, based on a sample representative for the (West) German citizens (until 2009) or for all German-speaking residents (starting 2010). Travelers from former East Germany are only included since 1990. Note that the sample only contains trips with at least five days of trip length. For details see Weigert et al. (2021).

Usage

```
data(travel)
```

Format

A dataframe containing

period Year in which the respondent traveled.

age Age of the respondent.

sampling_weight Individual weight of each respondent to account for a not perfectly representative sample and project the sample results to the population of German citizens (until 2009) or of German-speaking residents (starting 2010). Only available since 1974.

german_citizenship Indicator if the respondent is German citizen or not. Only available since 2010. Until 2009, all respondents were German citizens.

residence_region Indicator if the respondent's main residence is in a federal state in the former area of West Germany or in the former area of East Germany.

household size Categorized size of the respondent's household.

household_income Joint income (in €) of the respondent's household.

mainTrip_duration Categorized trip length of the respondent's *main trip*. The main trip is the trip which the respondent stated was his/her most important trip in the respective year.

mainTrip_distance Distance (in km) between the center of the respondent's federal state and the center of the country of destination, for the *main trip*. The main trip is the trip which the respondent stated was his/her most important trip in the respective year.

Details

The data are a 10% random sample of all respondents who undertook at least one trip in the respective year, between 1971 and 2018. We thank the Forschungsgemeinschaft Urlaub und Reisen e.V. for allowing us to publish this sample.

References

Weigert, M., Bauer, A., Gernert, J., Karl, M., Nalmpatian, A., Küchenhoff, H., and Schmude, J. (2021). Semiparametric APC analysis of destination choice patterns: Using generalized additive models to quantify the impact of age, period, and cohort on travel distances. *Tourism Economics*. doi:10.1177/1354816620987198.

Forschungsgemeinschaft Urlaub und Reisen e.V. (FUR) (2020b) *Survey of tourist demand in Germany for holiday travel and short breaks*. Available at: https://reiseanalyse.de/wp-content/uploads/2022/11/RA2020_First-results_EN.pdf (accessed 13 January 2023).

Index

∗ **datasets**
drug_deaths, 10
travel, 34

bam, *4, 5*, *8, 9*, *11–15*, *18*, *27–31*

calc_density, 3
capitalize_firstLetter, 3
compute_marginalAPCeffects, 4
compute_xCoordinate, 4
compute_yCoordinate, 5
create_APCsummary, 5, *9*
create_groupVariable, 6
create_highlightDiagonalData, 7, *13*, *25, 26*
create_modelSummary, 8
create_oneAPCsummaryTable, 9
cut, *21*, *23*, *26*

density, *3*, *21*, *26*
drug_deaths, 10

ensure_segmentsInPlotRange, 11
extract_summary_linearEffects, 11, *29*

gam, *4, 5*, *8, 9*, *11–15*, *18*, *27–31*
geom_bar, *33*
get_plotGAMobject, 12
gg_addReferenceLines, *11*, 12
gg_highlightDiagonals, 13
ggarrange, *16*, *28*, *32*

kable, *5, 6*, *8*

plot.gam, *12*
plot_1Dsmooth, 13
plot_APCheatmap, *12*, 15, *19*
plot_APChexamap, *4, 5*, *16*, 18
plot_density, *3*, 20, *23*, *25*
plot_density_categorical, 25
plot_density_metric, 25
plot_densityMatrix, *3*, *6, 7*, *21*, 22
plot_jointMarginalAPCeffects, 27
plot_linearEffects, 28
plot_marginalAPCeffects, 29
plot_partialAPCeffects, *29*, 31
plot_variable, 33

scale_fill_gradient2, *16*

travel, 34