Package 'blsplotGG'

February 19, 2025

Title Plots for Seasonal Adjustment Analysts using ggplot2

Version 2.1

Description
Generates several types of time series plots useful for seasonal adjustment analysis using ggplot2. These routines rely heavily on the seasonal package to extract series and components from the ' seasonal adjustments generated by the US Census Bureau's X-13ARIMA-SEATS software, and can be generated from a single seas object or a list of seas objects. Types of plots include line plots, ratio plots, forecast plots, forecast error diagnostic plots, spectral plots, seasonal factor plots, seasonal adjustment component plots, revisions history diagnostic plots, and SEATS diagnostic plots.

Users can add grid lines, remove background shading, and shade recession regions in selected plots.

License MIT + file LICENSE

Encoding UTF-8

LazyData TRUE

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Imports astsa,
BrailleR,
dplyr (>= 1.0.10),
ggfortify,
ggplot2 (>= 3.0.0),
ggpubr,
grDevices,
lubridate,
magrittr (>= 1.5),
RColorBrewer,
rlang (>= 1.1.1),
scales (>= 1.3.0),
seasonal (>= 1.10.0),
stringr (>= 1.5.0),
tidyr (>= 1.3.0),
tis,
tsbox

Suggests zoo

Depends R (>= 3.5)

Contents

`absmax` *Maximum absolute value of a vector*

Description

Generates the maximum of the absolute value of a numeric vector.

Usage

```
absmax(x)
```

Arguments

x	vector of numbers

Details

Version 1.1, 3/29/2021

Value

Maximum of the absolute value of a vector

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
r50 <- rnorm(50)
r50.absmax <- absmax(r50)
```

add_outlier_lines *add lines for outliers*

Description

add lines for outliers to a ggplot plot object

Usage

```
add_outlier_lines(
  this_p = NULL,
  seas_obj = NULL,
  line_color = c("red", "blue", "orangered", "green", "steelblue", "blue"),
  this_palette = "Dark2",
 this_line_type = c("dashed", "dotdash", "dashed", "twodash", "dotdash", "dotdash")
)
```

Arguments

this_p	A ggplot object of a time series plot. This is a required entry.
seas_obj	seas object generated from a call of seas on a single time series. This is a required entry.
line_color	Character array of length 6; color used for different outliers, with the order being 'ao', 'ls', 'tc', 'so', 'rp', 'tls'. Default is the RColorBrewer palette "Dark2".
this_palette	Character string; default RColorBrewer palette. Deault is "Dark2".
this_line_type	Character array of length 6; Line type used for different outliers, with the order being 'ao', 'ls', 'tc', 'so', 'rp', 'tls'. Default is c('dashed', 'dotdash', 'dashed', 'twodash', 'dotdash', 'dotdash').

Details

Version 3.2, 9/5/2024

Value

Revised ggplot object with lines for outliers added.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
air_seas <-
   seasonal::seas(AirPassengers, arima.model = "(0 1 1)(0 1 1)", x11 = "")
air_df   <-
   data.frame(date = tsbox::ts_df(AirPassengers)$time,
              ori = as.double(seasonal::original(air_seas)),
              sa = as.double(seasonal::final(air_seas)),
              trend = as.double(seasonal::trend(air_seas)))
```

```
this_x11_p <- ggplot2::ggplot() +
  ggplot2::geom_line(mapping = ggplot2::aes(x=date, y = ori),
                     color = "grey", data = air_df) +
  ggplot2::geom_line(mapping = ggplot2::aes(x=date, y = sa),
                     color="steelblue", linetype="twodash", data = air_df) +
  ggplot2::geom_line(mapping = ggplot2::aes(x=date, y = trend),
                     color="darkred", linetype="twodash", data = air_df) +
  ggplot2::labs(
    title = "Airline Passenger X-11 Seasonal Adjustment",
    subtitle = NULL,
    x = "Time",
    y = "Airline Passengers")
this_x11_p_with_outlier_lines <-
    add_outlier_lines(this_x11_p, air_seas, this_palette = "Paired",
                      this_line_type = rep("dotted", 6))
```

`add_recession_shade` *Add shading for NBER recession dates*

Description

Add shading for US NBER recession dates ro ggplot plot object.

Usage

```
add_recession_shade(
  this_p = NULL,
  shade_color = "lightblue1",
  shade_alpha = 0.2
)
```

Arguments

`this_p`	`ggplot` object of a time series plot. This is a required entry.
`shade_color`	Character scalar; shading for recession region. Default is `"pink"`.
`shade_alpha`	numeric scalar; controls the intensity of the shading. Default is `0.2`.

Details

Version 3.2, 5/6/2024

Value

`ggplot` object with shading for recession added.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
air_seas <-
   seasonal::seas(AirPassengers, arima.model = "(0 1 1)(0 1 1)", x11 = "")
air_df   <-
   data.frame(date = tsbox::ts_df(AirPassengers)$time,
              ori = as.double(seasonal::original(air_seas)),
              sa = as.double(seasonal::final(air_seas)),
              trend = as.double(seasonal::trend(air_seas)))
this_p <- ggplot2::ggplot(air_df) +
  ggplot2::geom_line(ggplot2::aes(x=date, y = ori), color = "grey") +
 ggplot2::geom_line(ggplot2::aes(x=date, y = sa), color="steelblue", linetype="twodash") +
 ggplot2::geom_line(ggplot2::aes(x=date, y = trend), color="darkred", linetype="twodash") +
  ggplot2::labs(
    title = "Airline Passenger X-11 Seasonal Adjustment",
    subtitle = NULL,
    x = "Time",
    y = "Airline Passengers")
this_p_with_recession_shading <-
    add_recession_shade(this_p, shade_color = "steelblue")
```

`color_blind_palette` *Color-blind friendly color palette*

Description

Color palettes that can be used that can be distinguished by color-blind people (either from `RColorBrewer` or Cookbook for R - Colors (ggplot2)).

Usage

```
color_blind_palette(with_grey = TRUE, brewer_palette = NULL)
```

Arguments

`with_grey`	Logical scalar; whether color blind pallate contains `'grey'`, otherwise the palette contains `black`. Default is TRUE.
`brewer_palette`	Character string; a `RColorBrewer` palette. There is no default - must be a color-blind friendly palette.

Details

Version 2.2, 8/2/2024

Value

Vector of hexadecimal color codes that form a color palette that can be distinguished by color-blind people.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

References

https://CRAN.R-project.org/package=RColorBrewer, http://www.cookbook-r.com/Graphs/Colors_(ggplot2)/http://www.cookbook-r.com/Graphs/Colors_(ggplot2)/

Examples

```
this_color_blind <- color_blind_palette(FALSE, brewer_palette = "Dark2")
```

`convert_identify_acf` *Convert matrix of ACFs or PACFs generated by X-13ARIMA-SEATS* `identify` *spec to a list object*

Description

Generates a list of the ACF or PACF generated by the `identify` spec

Usage

```
convert_identify_acf(seas_obj = NULL, this_plot = "iac")
```

Arguments

`seas_obj`	`seas` object generated from a call of `seas` on a single time series This is a required entry.
`this_plot`	Character string; three character code for the type of plot to be generated. Allowed entries are `"iac"` (sample autocorrelation function, default), `"ipf"` (sample partial autocorrelation function).

Details

Version 2.0, 10/23/2024

Value

A list of matrices of ACF or PACFs produced for different orders of differncing. The list entries are named based on the orders of differencing (`d0sd0` denotes no regular difference, no seasonal difference, `d1sd0` denotes one regular difference, no seasonal difference, etc.)

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
shoes_identify_seas <-
   seasonal::seas(shoes2008,
                  identify.diff = c(0, 1),
                  identify.sdiff = c(0, 1),
                  identify.save = c("iac", "ipc"),
                  arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log",
                  forecast.maxlead = 36,
```

```
     check.maxlag = 36,
     check.acflimit = 1.96,
     check.qlimit = 0.01,
                 check.print = c( 'pacf', 'pacfplot' ))
shoes_identify_acf_list <-
    convert_identify_acf(shoes_identify_seas, 'iac')
shoes_identify_pacf_list <-
    convert_identify_acf(shoes_identify_seas, 'ipc')
```

`convert_spectrum_code` *Convert spectrum code to visual peak code*

Description

Convert the three character code used by the `plot_spectrum` function so that it can be used with the `visual_sig_peaks` function from the `blsplot` package.

Usage

```
convert_spectrum_code(this_spectrum = "sp0")
```

Arguments

`this_spectrum`	Character string; three character code for the X-13 spectrum to be generated. Allowed entries are `"sp0"` (modified original series), `"sp1"` (modified X-11 seasonally adjusted series), `"sp2"` (modified X-11 irregular), `"s1s"` (modified SEATS seasonally adjusted series), `"s2s"` (modified SEATS irregular), `"is0"` (modified composite series), `"is1"` (modified indirect seasonally adjusted series), `"is2"` (modified indirect irregular), `spr` (model residuals), or `"ser"` (extended residuals). Default: `"sp0"`.

Details

Version 1.4, 5/6/2024

Value

Text for spectrum associated with code used in `plot_spectrum` function. If improper `this_spectrum` specified, function will return `NULL`.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
this_code <- convert_spectrum_code("sp2")
```

display_color_blind_palettes *Return color-blind friendly palettes*

Description

Returns the names of color palettes from the RColorBrewer package that can be distinguished by color-blind people.

Usage

```
display_color_blind_palettes(this_category = NULL)
```

Arguments

this_category	Character string; specify which catagory of color palette will be returned. Possible choices are "div" (diverging), "qual" (qualitative), "seq". If not specified, all color palettes are returned

Details

Version 1.3, 3/6/2024

Value

Vector of color palette names from the RColorBrewer package that can be distinguished by color-blind people.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
qual_color_blind_palettes <- display_color_blind_palettes("qual")
```

employment_data_mts *US Unemployment Series, four main components in an mts object*

Description

An mts object of the four main components of US Employment expressed as time series objects that end in December, 2022

Usage

```
employment_data_mts
```

Format

An mts object with 4 time series elements in four columns:

n2000013 Employed Males 16-19

n2000014 Employed Females 16-19

n2000025 Employed Males 20+

n2000026 Employed Females 20+

---

employment_list *US Employment Series, four main components in a list object*

Description

A list object of the four main components of US Employment expressed as time series objects that end in December, 2022

Usage

```
employment_list
```

Format

A list object with 4 time series elements:

n2000013 Employed Males 16-19

n2000014 Employed Females 16-19

n2000025 Employed Males 20+

n2000026 Employed Females 20+

---

extract_range_from_ggplot *Extract range of data from ggplot object*

Description

Computes the range of all data plotted in given ggplot object

Usage

```
extract_range_from_ggplot(this_p = NULL)
```

Arguments

this_p	ggplot object of a time series plot. This is a required entry.

Details

Version 1.2, 10/19/2023

Value

Vector of length 2 with the range of the data used to generate a given ggplot object of a time series plot.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
air_seas <-
   seasonal::seas(AirPassengers, arima.model = "(0 1 1)(0 1 1)", x11 = "")
air_df   <-
   data.frame(date = tsbox::ts_df(AirPassengers)$time,
              ori = as.double(seasonal::original(air_seas)),
              sa = as.double(seasonal::final(air_seas)),
              trend = as.double(seasonal::trend(air_seas)))
air_p <- ggplot2::ggplot(air_df, ggplot2::aes(x=date)) +
  ggplot2::geom_line(ggplot2::aes(y = ori), color = "grey") +
  ggplot2::geom_line(ggplot2::aes(y = trend), color="darkred",
                     linetype="twodash") +
  ggplot2::labs(
    title = "Airline Passenger X-11 Trend Component",
    subtitle = NULL,
    x = "Time",
    y = "Airline Passengers")
air_short_p <- plot_date_span(air_p, "1-1-1956", "1-1-1961")
air_short_range <- extract_range_from_ggplot(air_short_p)
```

---

flag_peak *Flag visual significant peaks in spectra*

Description

Determine positions of visual significant peaks in spectra

Usage

```
flag_peak(seas_obj = NULL, spec_type = NULL, spec_freq_code = NULL)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This is a required argument.
spec_type	Character string; type of spectrum. Possible values are 'ori','irr','rsd','sa'.
spec_freq_code	Character string; type of frequency being tested. Possible values are 's' or 't'.

Details

Version 3.0, 6/21/2024

Value

If visually significant peaks are found, a numeric vector of the position of the peak frequencies are returned. If no peaks found, the function returns 0.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
air_seas <- seasonal::seas(AirPassengers, arima.model = '(0 1 1)(0 1 1)', x11='')
this_flagged_peak_seas <- flag_peak(air_seas,'ori','s')
this_flagged_peak_td <- flag_peak(air_seas,'ori','t')
```

---

generate_alt_text *Generate alt text for ggplot graphs*

Description

Generates alt text for ggplot graph objects using the BrailleR package and adding text suggested by Amy Casale in her article "Writing Alt Text for Data Visualization".

Usage

```
generate_alt_text(
  gg_object = NULL,
  chart_type = NULL,
  data_type = NULL,
  reason_text = NULL,
  short_alt = FALSE,
  BrailleR_only = FALSE
)
```

Arguments

gg_object	ggplot object from which alt text will be generated. Required entry if short_alt = FALSE.
chart_type	character scalar telling what type of plot is used in gg_object. This is a required entry.
data_type	character scalar detailing what data is used in gg_object. This is a required entry.
reason_text	character scalar detailing the reason gg_object is plotted. This is a required entry.
short_alt	logical scalar if TRUE BrailleR text will not be appended to the alt text. Default is FALSE.
BrailleR_only	logical scalar if TRUE only BrailleR text will returned. Default is FALSE.

Details

Version 3.1, 9/9/2024

Value

generate alt text for plot produced by gg_object

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

References

https://CRAN.R-project.org/package=BrailleR and https://medium.com/nightingale/writing-alt-text-f

Examples

```
air_seas <- seasonal::seas(AirPassengers, arima.model = "(0 1 1)(0 1 1)", x11="")
air_df   <-
   data.frame(date = tsbox::ts_df(AirPassengers)$time,
              ori = as.double(seasonal::original(air_seas)),
              sa = as.double(seasonal::final(air_seas)),
              trend = as.double(seasonal::trend(air_seas)))
air_p <- ggplot2::ggplot(air_df, ggplot2::aes(x=date)) +
  ggplot2::geom_line(ggplot2::aes(y = ori), color = "grey") +
  ggplot2::geom_line(ggplot2::aes(y = sa),
                     color="steelblue", linetype="twodash") +
  ggplot2::geom_line(ggplot2::aes(y = trend),
                     color="darkred", linetype="dotdash") +
  ggplot2::ggtitle("Airline Passenger X-11 Seasonal Adjustment")
air_alt_text <-
   generate_alt_text(air_p,
                     "Time series plot",
                     "International Airline Passengers time series",
                     "compare seasonal adjustment and trend to original series")
```

---

gen_outlier_label *generate x-axis label for outliers*

Description

Generate an x-axis label when adding lines for outliers to a ggplot plot object

Usage

```
gen_outlier_label(
  seas_obj = NULL,
  this_color = c("red", "blue", "green", "brown", "grey", "yellow")
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series. This is a required entry.
this_color	Character array of length 6; color used for different outliers, with the order being "ao", "ls", "tc", "so", "rp", "tls". Default is c("red", "blue", "green", "brown", "grey", "yellow").

Details

Version 1.1, 5/15/2024

Value

Character string with description of outliers for x-axis label of a ggplot.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
air_seas <-
   seasonal::seas(AirPassengers, arima.model = "(0 1 1)(0 1 1)", x11 = "")
air_df   <-
   data.frame(date = tsbox::ts_df(AirPassengers)$time,
              ori = as.double(seasonal::original(air_seas)),
              sa = as.double(seasonal::final(air_seas)),
              trend = as.double(seasonal::trend(air_seas)))
this_p <- ggplot2::ggplot(air_df) +
  ggplot2::geom_line(ggplot2::aes(x=date, y = ori), color = "grey") +
 ggplot2::geom_line(ggplot2::aes(x=date, y = sa), color="steelblue", linetype="twodash") +
 ggplot2::geom_line(ggplot2::aes(x=date, y = trend), color="darkred", linetype="twodash") +
  ggplot2::labs(
    title = "Airline Passenger X-11 Seasonal Adjustment",
    subtitle = NULL,
    y = "Airline Passengers")
this_p_with_outlier_lines <- add_outlier_lines(this_p, air_seas)
outlier_lines_label <- gen_outlier_label(air_seas)
this_p_with_outlier_lines <-
    this_p_with_outlier_lines + ggplot2::xlab(outlier_lines_label)
```

---

get_auto_outlier_string

Get automatic outlier names

Description

Get the names of outliers identified in the seas object for a single series.

Usage

```
get_auto_outlier_string(seas_obj = NULL)
```

Arguments

seas_obj	A seas object for a single series generated from the seasonal package. This is a required entry.

Details

Version 3.0, 5/14/2024

Value

Character string containing a summary of the outliers identified in the regARIMA model. If no regressors or automatic outliers in the model, the routine will return a blank character.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
air_seas <- seasonal::seas(AirPassengers, arima.model = "(0 1 1)(0 1 1)", x11="")
this_auto_outlier <- get_auto_outlier_string(air_seas)
```

---

get_month_index *Generate index of month abbreviation*

Description

Process string of month abbrev to return a numeric index.

Usage

```
get_month_index(this_month_string)
```

Arguments

`this_month_string`
: Character string; 3 character abbreviation of month

Details

Version 2.4, 12/11/2024

Value

Index of month - 1 for 'Jan', 2 for 'Feb', etc.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
thisOtl <- 'AO2015.Jan'
thisCode <- 'AO'
thisPerChar <- substr(thisOtl,nchar(thisCode)+6,nchar(thisOtl))
thisPerIndex <- get_month_index(thisPerChar)
```

get_reg_string *Get names of regressors*

Description

Generate string of names for the regressors used in the model fit for a given series.

Usage

```
get_reg_string(seas_obj = NULL, xreg_names = NULL)
```

Arguments

seas_obj	seas object generated by the seasonal package for a single series. This is a required entry.
xreg_names	Character vector with names of user defined regressors used in model. Default is NULL, no user defined regressors. Number of names in this vector should match number of user-defined regressors; if not, a warning message will be produced.

Details

Version 2.9, 5/14/2024

Value

Character string containing a summary of the regressors in the regARIMA model. If no regressors in the model, the routine will return a blank character.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
air_seas <- seasonal::seas(AirPassengers, slidingspans = "",
                  transform.function = "log",
                  x11 = "", forecast.maxlead=36,
                  check.print = c( "pacf", "pacfplot" ))
air_reg <- get_reg_string(air_seas)
```

get_udg_index *Index for entry in* UDG *diagnostics list*

Description

Return index for entry in UDG diagnostics list from seas object.

Usage

```
get_udg_index(udg_list = NULL, this_key = NULL)
```

Arguments

udg_list	List object generated by udg() function of the seasonal package. This is a required entry.
this_key	Keyword found in udg files generated by X-13ARIMA-SEATS This is a required entry.

Details

Version 2.3, 5/25/2023

Value

An integer denoting which element in the udg output matches the key provided by the user. If there is no match, the function returns the number 0.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
ukgas_seas <- seasonal::seas(UKgas, series.period = 4, arima.model = "(0 1 1)(0 1 1)",
                             x11="", transform.function = "log", forecast.maxlead=20,
                             check.print = c( "pacf", "pacfplot" ))
ukgas_udg <- seasonal::udg(ukgas_seas)
ukgas_udg_index_a <- get_udg_index(ukgas_udg, this_key='a')
```

---

plot_acf *Generate ACF plot of the regARIMA model residuals.*

Description

Generate ACF, PACF, or squared ACF plot of the regARIMA model residuals.

Usage

```
plot_acf(
  seas_obj = NULL,
  this_plot = "acf",
  acf_range = NULL,
  add_ci = TRUE,
  main_title = "ACF Plot",
  sub_title = NULL,
  this_x_label = "Lag",
  this_y_label = "ACF",
  this_x_axis_breaks = NULL,
  do_grid = FALSE,
  do_background = FALSE,
  acf_color = "steelblue",
  ci_color = "grey"
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This is a required entry.
this_plot	Character string; three character code for the type of plot to be generated. Allowed entries are "acf" (sample autocorrelation function, default), "pcf" (sample partial autocorrelation function), "ac2" (squared autocorrelation function).
acf_range	Range of values you wish the acf plot to be plotted over. Default is range of the series.
add_ci	Logical scalar; indicates if confidence intervals are added to the plot. Default is confidence intervals are added.
main_title	Title for the plot. Default is generated based on the value of this_plot.
sub_title	Subtitle for the plot. Default is NULL, or no subtitle.
this_x_label	Label for X axis. Default is "Lags".
this_y_label	Label for Y axis. Default is generated based on the value of this_plot.
this_x_axis_breaks	Numeric vector; sets the values for the x-axis. Default uses the value of this_frequency to set x-axis.
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
acf_color	Color used for lines in ACF plot. Default is "steelblue".
ci_color	Color used for confidence interval in ACF plot. Default is "grey".

Details

Version 4.2, 2/18/2025

Value

A ggplot object that produces an ACF, PACF, or squared ACF plot of the regARIMA residuals.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
ukgas_x11_seas <-
   seasonal::seas(UKgas, series.period = 4,
                  arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log",
                  forecast.maxlead = 20,
                  x11.seasonalma = "s3x5",
                  check.print = c( 'pacf', 'pacfplot' ),
                  check.maxlag = 12,
                  check.save = c("acf", "pcf", "ac2"))
p_ukgas_acf   <-
   plot_acf(ukgas_x11_seas, this_plot = "acf",
            main_title = "UK Gas Model Residual ACF",
```

```
            acf_color = "darkblue")
p_ukgas_pacf   <-
   plot_acf(ukgas_x11_seas, this_plot = "pcf",
            main_title = "UK Gas Model Residual PACF",
            acf_color = "darkblue")
p_ukgas_acf2   <-
   plot_acf(ukgas_x11_seas, this_plot = "ac2",
            main_title = "UK Gas Model Squared ACF",
            acf_color = "darkblue")
```

`plot_acf_and_pacf` *Generate ACF and PACF plot of the regARIMA model residuals.*

Description

Generate a single plot with ACF and PACF of the regARIMA model residuals.

Usage

```
plot_acf_and_pacf(
  seas_obj = NULL,
  add_ci = TRUE,
  overall_title = NULL,
  acf_title = "ACF Plot",
  pacf_title = "PACF Plot",
  this_x_label = "Lag",
  this_y_label = c("ACF", "PACF"),
  this_x_axis_breaks = NULL,
  do_grid = FALSE,
  do_background = FALSE,
  acf_color = "steelblue",
  ci_color = "grey"
)
```

Arguments

`seas_obj`	`seas` object generated from a call of `seas` on a single time series This is a required entry.
`add_ci`	Character scalar; Overall title for the plot. Default is confidence intervals are added.
`overall_title`	Overall title for the combined plot. Default is `NULL`, which doesn't produce an overall title.
`acf_title`	Subtitle for the ACF Plot. Default is character string `'ACF Plot'`.
`pacf_title`	Subtitle for the PACF Plot. Default is character string `'PACF Plot'`.
`this_x_label`	Label for X axis. Default is `"Lags"`.
`this_y_label`	Character vector of length two. Labels for each Y axis. Default is `c("ACF", "PACF")`.
`this_x_axis_breaks`	Numeric vector; sets the values for the x-axis. Default uses the value of `this_frequency` to set x-axis.

do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
acf_color	Color used for lines in ACF plot. Default is "steelblue".
ci_color	Color used for confidence interval in ACF plot. Default is "grey".

Details

Version 4.0, 2/18/2025

Value

A ggplot object that produces an ACF and PACF plot for the regARIMA model residuals.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
ukgas_x11_seas <-
   seasonal::seas(UKgas, series.period = 4,
                  arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log",
                  forecast.maxlead = 20,
                  x11.seasonalma = "s3x5",
                  check.print = c( 'pacf', 'pacfplot' ),
                  check.maxlag = 12,
                  check.save = c("acf", "pcf", "ac2"))
p_ukgas_acf_and_pacf   <-
   plot_acf_and_pacf(ukgas_x11_seas,
            overall_title = "UK Gas Model",
            acf_color = "darkblue")
```

---

plot_acf_and_pacf_identify

Generate ACF and PACF plot of the regARIMA model residuals.

Description

Generate a single plot with ACF and PACF of the regARIMA model residuals.

Usage

```
plot_acf_and_pacf_identify(
  seas_obj = NULL,
  this_diff = 0,
  this_sdiff = 0,
  add_ci = TRUE,
  overall_title = NULL,
  acf_title = "ACF Plot",
```

```
  pacf_title = "PACF Plot",
  this_x_label = "Lag",
  this_y_label = c("ACF", "PACF"),
  this_x_axis_breaks = NULL,
  do_grid = FALSE,
  do_background = FALSE,
  acf_color = "steelblue",
  ci_color = "grey"
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This is a required entry.
this_diff	Integer scalar; one of the regular differences specified in the diff argument of the identify spec. Default is 0.
this_sdiff	Integer vector; one of the seasonal differences specified in the sdiff argument of the identify spec. Default is 0.
add_ci	Character scalar; Overall title for the plot. Default is confidence intervals are added.
overall_title	Overall title for the combined plot. Default is a text string showing the orders of differencing selected.
acf_title	Subtitle for the ACF Plot. Default is character string 'ACF Plot'.
pacf_title	Subtitle for the PACF Plot. Default is character string 'PACF Plot'.
this_x_label	Label for X axis. Default is "Lags".
this_y_label	Character vector of length two. Labels for each Y axis. Default is c("ACF", "PACF").
this_x_axis_breaks	Numeric vector; sets the values for the x-axis. Default uses the value of this_frequency to set x-axis.
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
acf_color	Color used for lines in ACF plot. Default is "steelblue".
ci_color	Color used for confidence interval in ACF plot. Default is "grey".

Details

Version 4.0, 2/18/2025

Value

A ggplot object that produces an ACF and PACF plot for the original series with user-specified orders of differencing.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
shoes_identify_seas <-
   seasonal::seas(shoes2008,
                  identify.diff = c(0, 1),
                  identify.sdiff = c(0, 1),
                  identify.save = c("iac", "ipc"),
                  arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log",
                  forecast.maxlead = 36,
     check.maxlag = 36,
     check.acflimit = 1.96,
     check.qlimit = 0.01,
                  check.print = c( 'pacf', 'pacfplot' ))
p_shoes_acf_and_pacf_identify_d0_sd0   <-
   plot_acf_and_pacf_identify(shoes_identify_seas,
            overall_title = "US Shoe Sales - No Differencing",
            acf_color = "darkblue")
p_shoes_acf_and_pacf_identify_d1_sd1   <-
   plot_acf_and_pacf_identify(shoes_identify_seas,
           this_diff = 1,
   this_sdiff = 1,
            overall_title = "US Shoe Sales - Regular and Seasonal Differencing",
            acf_color = "darkblue")
```

`plot_acf_matrix` *Generate ACF plot of the regARIMA model residuals from a matrix of the ACF.*

Description

Generate ACF, PACF, or squared ACF plot of the regARIMA model residuals from a matrix of the ACF.

Usage

```
plot_acf_matrix(
  acf_matrix = NULL,
  acf_range = NULL,
  add_ci = TRUE,
  acflimit = 1.6,
  main_title = "ACF Plot",
  sub_title = NULL,
  this_x_label = "Lag",
  this_y_label = "ACF",
  this_x_axis_breaks = NULL,
  this_frequency = 12,
  acf_color = "steelblue",
  ci_color = "grey"
)
```

Arguments

acf_matrix	Numeric matrix containing the ACF, PACF, or squared ACF with columns with SE, Ljung Box Q, lags, if associated with the file. This is a required entry.
acf_range	Range of values you wish the acf plot to be plotted over. Default is range of the series.
add_ci	Logical scalar; indicates if confidence intervals are added to the plot. Default is confidence intervals are added.
acflimit	Numeric scalar; the multiplier for the confidence interval usually read from the udg. Default: 1.6.
main_title	Title for the plot. Default is "ACF Plot".
sub_title	Subtitle for the plot. Default is NULL, or no subtitle.
this_x_label	Label for X axis. Default is "Lags".
this_y_label	Label for Y axis. Default is "ACF".
this_x_axis_breaks	Numeric vector; sets the values for the x-axis. Default uses the value of this_frequency to set x-axis.
this_frequency	Integer scalar; Frequency of the time series. Default is 12.
acf_color	Color used for lines in ACF plot. Default is "steelblue".
ci_color	Color used for confidence interval in ACF plot. Default is "grey".

Details

Version 3.1, 2/18/2025

Value

A ggplot object that produces an ACF, PACF, or squared ACF plot of the regARIMA residuals.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
ukgas_x11_seas <-
   seasonal::seas(UKgas, series.period = 4,
                  arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log",
                  forecast.maxlead = 20,
                  x11.seasonalma = "s3x5",
                  check.print = c( 'pacf', 'pacfplot' ),
                  check.maxlag = 12,
                  check.save = c("acf", "pcf", "ac2"))
ukgas_acf_matrix   <-
   seasonal::series(ukgas_x11_seas, "acf")
p_ukgas_acf_matrix   <-
   plot_acf_matrix(ukgas_acf_matrix,
            main_title = "UK Gas Model Squared ACF",
            this_frequency = 4,
            acf_color = "darkblue")
```

plot_all_trend_lags *History Plot of All Trend Revisions*

Description

Generates a ggplot2 object of estimates from a revisions history of a trend component for a given series, incorporating all trend lag revisions.

Usage

```
plot_all_trend_lags(
  seas_obj = NULL,
  this_trend_name = NULL,
  main_title = "All Trend Revision Plot",
  sub_title = NULL,
  this_y_label = NULL,
  this_x_label = "Time",
  do_grid = FALSE,
  do_background = FALSE,
  base_color = "darkblue",
  whisker_color = "darkgrey"
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This is a required entry.
this_trend_name	Character string; name of time series. No default.
main_title	Title for the plot. Default is character string 'Trend History Plot'.
sub_title	Subtitle for the plot. Default is NULL.
this_y_label	Label for Y axis. Default is "Ratio".
this_x_label	Label for X axis. Default is "Time".
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
base_color	Character scalar for plot of the initial trend. Default is "darkblue".
whisker_color	Character scalar for color used for lines representing lagged trend estimates in the trend history plot. Default is "darkgrey"

Details

Version 1.8, 9/5/20244

Value

A ggplot object that produces a history plot of the trend component. The seas object in the seas_obj argument must contain output for a revisions history analysis for trends with trendlags set to some values - these lags should be in sequence, such as history.trendlags = 1:4.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
shoes_seas <-
    seasonal::seas(shoes2008, x11.save = "d13",
       x11 = "", transform.function = "log",
       check.print = c("none", "+acf", "+acfplot", "+normalitytest"),
       regression.aictest = c("td", "easter"),
  regression.save = c("td", "hol"),
       outlier.types = "all",
       arima.model = "(0 1 1)(0 1 1)",
       forecast.maxlead = 60,
      history.estimates = c("sadj", "sadjchng", "trend"),
      history.trendlags = 1:4,
      history.print = "all",
      history.save = c("tre", "trr"))
p_shoes_sa_history <-
    plot_all_trend_lags(shoes_seas, this_trend_name = "shoes",
        main_title = "All Trend Revisions, Lags 1 to 4",
        sub_title = "US Retail Sales of Shoes")
```

---

plot_changes_history *Revisions History Plot for Seasonal Adjustment Changes*

Description

Generates a ggplot object of estimates from a revisions history of a seasonal adjustment changes for a given series.

Usage

```
plot_changes_history(
  seas_obj = NULL,
  plot_start = NULL,
  main_title = "SA Change History Graph",
  sub_title = NULL,
  this_x_label = "Time",
  this_y_label = NULL,
  do_grid = FALSE,
  do_background = FALSE,
  line_color = NULL,
  this_palette = "Paired"
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This is a required entry.
plot_start	Integer vector of length two. Start of the plot. Default is NULL, which defaults to the start of the history analysis.

main_title	Title for the plot. Default is character string 'SA Change History Graph'.
sub_title	Subtitle for the plot. Default is NULL.
this_x_label	Label for X axis. Default is "Time".
this_y_label	Label for Y axis. Default is NULL.
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
line_color	Vector with colors used for lines in history plot. Should be of length 2. Default is NULL, which indicates that the palette specified in this_palette is used to generate colors for this plot.
this_palette	Color used for lines in plot. Default is "Paired"

Details

Version 1.5, 11/6/2024

Value

A ggplot object that produces a history plot of the seasonal adjustment changes. The seas object in the seas_obj argument must contain output for a revisions history analysis for seasonal adjustment changes.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
shoes_seas <-
    seasonal::seas(shoes2008, x11.save = "d13",
       x11 = "", transform.function = "log",
       check.print = c("none", "+acf", "+acfplot", "+normalitytest"),
       regression.aictest = c("td", "easter"),
  regression.save = c("td", "hol"),
       outlier.types = "all",
       arima.model = "(0 1 1)(0 1 1)",
       forecast.maxlead = 60,
      history.estimates = c("sadj", "sadjchng", "trend"),
      history.sadjlags = c(1,12),
      history.print = "all",
      history.save = c("che", "chr"))
p_shoes_sa_history <-
    plot_changes_history(shoes_seas, plot_start = c(2003, 1),
        main_title = "Seasonal Adjustment Change History Graph",
        sub_title = "US Retail Sales of Shoes")
```

plot_component_facet *Component facet plot*

Description

Generates a plot for the components of a seasonal adjustment - trend, seasonal, and irreglar.

Usage

```
plot_component_facet(
  seas_obj = NULL,
  main_title = "Component Facet Plot",
  sub_title = NULL,
  this_x_label = "Time",
  this_y_label = " ",
  do_grid = FALSE,
  do_background = FALSE,
  this_scale = "free_y",
  line_color = "steelblue"
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This is a required entry.
main_title	Title for the plot. Default is character string 'Component Facet Plot'.
sub_title	Subtitle for the plot. Default is either "SEATS seasonal adjustment" or "X-11 seasonal adjustment".
this_x_label	Label for X axis. Default is "Time".
this_y_label	Label for Y axis. Default is " ".
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
this_scale	Character scalar; set the scale for the y-axis of the facet plots. Default is "free_y". To set axis to be the same for the different plots, set this_scale = "fixed".
line_color	Color used for lines in component plot. Default is "steelblue".

Details

Version 1.0, 1/6/2025

Value

A ggplot object that generates a ratio facet plot for the factors provided in the ratio_tables argument.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
shoes_x11_seas <-
   seasonal::seas(shoes2008, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 36,
                  regression.aictest = c("td", "easter"),
              regression.save = c("td", "hol"),
                  x11.seasonalma = "s3x5", check.print = c( 'pacf', 'pacfplot' ))
shoes_comp_plot <-
   plot_component_facet(shoes_x11_seas,
                        main_title = "US Retail Shoe Sales",
                        sub_title = "X-11 (3x5 Seasonal filter)",
                        )
```

---

plot_cpgram_resid *Generate cumulative periodogram of the regARIMA residuals*

Description

Generates a plot of the cumulative periodogram of the regARIMA residuals.

Usage

```
plot_cpgram_resid(
  seas_obj = NULL,
  main_title = "Cumulative periodogram",
  this_palette = NULL
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series. This is a required entry.
main_title	Title for the plot. Default is character string 'Cumulative periodogram'.
this_palette	Color used for lines in plot. Default is a color-blind friendly palette generated by the function color_blind_palette(FALSE)

Details

Version 1.6, 7/1/2024

Value

Generates a ggplot object of the cumulative periodogram of the regARIMA residuals. Diagnostic information is included in the plot subheader.

Examples

```
air_seas <-
   seasonal::seas(AirPassengers, transform.function= "log",
                  arima.model = "(0 1 1)(0 1 1)", check.save = "acf")
plot_cpgram_resid_air <-
     plot_cpgram_resid(air_seas,
                   main_title = "Cumulative periodogram for Airline Passenger Residuals")
```

plot_date_span *Plot a span of data*

Description

Shortens the time span of an existing time series ggplot object by limiting the X axis to user specified dates.

Usage

```
plot_date_span(
  this_p = NULL,
  this_start_span = NULL,
  this_end_span = NULL,
  this_date_breaks = "1 year",
  this_date_format = "%Y",
  reset_y_limit = FALSE
)
```

Arguments

this_p	A ggplot object of a time series plot. This is a required entry.
this_start_span	Character scalar with the date of the start of the span to be plotted. This is a required entry.
this_end_span	Character scalar with the date of the end of the span to be plotted. This is a required entry.
this_date_breaks	Character scalar with the interval for tic marks on the x-axis. Default is "1 year".
this_date_format	Character scalar with the format used for the date label on the x-axis.
reset_y_limit	Logical scalar that if TRUE, the range of the y-axis is reset. Default is FALSE.

Details

Version 2.2, 8/29/2024

Value

A ggplot object that produces a subplot of the submitted plot.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
air_seas <-
   seasonal::seas(AirPassengers, arima.model = "(0 1 1)(0 1 1)", x11 = "")
air_df   <-
   data.frame(date = tsbox::ts_df(AirPassengers)$time,
              ori = as.double(seasonal::original(air_seas)),
              sa = as.double(seasonal::final(air_seas)),
              trend = as.double(seasonal::trend(air_seas)))
air_p <- ggplot2::ggplot(air_df, ggplot2::aes(x=date)) +
  ggplot2::geom_line(ggplot2::aes(y = ori), color = "grey") +
  ggplot2::geom_line(ggplot2::aes(y = trend),
                     color="darkred",
                     linetype="twodash") +
  ggplot2::labs(
    title = "Airline Passenger X-11 Trend Component",
    subtitle = NULL,
    x = "Time",
    y = "Airline Passengers")
air_short_p <-
   plot_date_span(air_p, "1-1-1956", "1-1-1962", reset_y_limit = TRUE)
```

---

plot_double_spectrum *Generate double spectrum plot of the original and seasonally adjusted series.*

Description

Generate plot of spectrum of original series and seasonally adjusted series on same axis.

Usage

```
plot_double_spectrum(
  seas_obj = NULL,
  xaxis_bls = TRUE,
  main_title = "AR Spectrum",
  series_name = NULL,
  this_color = c("darkblue", "darkgreen"),
  this_median_color = c("blue", "green"),
  this_freq_color = c("steelblue", "forestgreen"),
  this_peak_color = c("violet", "brown")
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This is a required entry.
xaxis_bls	Logical scalar; indicates if x-axis of spectral plot will be frequency by month rather than the actual frequencies. Default sets x-axis to frequency by month.
main_title	Character string; main title of plot. Default is 'AR Spectrum'.
series_name	Character scalar; name of the time series used in seas_obj. Used as the label of the Y-axis if specified.

this_color Character vector of length 2. Colors used for original and seasonally adjusted spectrum in plot. Defaults are `c("darkblue", "darkgreen")`.

this_median_color Character vector of length 2. Colors used for medians of the original and seasonally adjusted spectrum, respectively. Defaults are `c("blue", "green")`.

this_freq_color Character vector of length 2. Colors used for seasonal and trading day frequencies, respectively. Defaults are `c("steelblue", "forestgreen")`.

this_peak_color Character vector of length 2. Colors used for peaks at seasonal and trading day frequencies, respectively. Defaults are `c("violet", "brown")`.

Details

Version 2.5, 7/1/2024

Value

`ggplot` object of spectrum of original series and seasonally adjusted series on same axis.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
air_seas <- seasonal::seas(AirPassengers, arima.model = "(0 1 1)(0 1 1)", x11="",
                           spectrum.save = c("sp0", "sp1"))
plot_double_spectrum_air <-
     plot_double_spectrum(air_seas, series_name = "AirPassengers",
                          this_color = c("steelblue", "forestgreen"),
                          this_median_color = c("blue", "green"),
                          this_freq_color = c("darkblue", "darkgreen"),
                          this_peak_color = c("red", "orange"))
```

---

plot_fcst *Forecast plot*

Description

Generates a `ggplot` plot of regARIMA forecasts with confidence bounds.

Usage

```
plot_fcst(
  seas_obj = NULL,
  main_title = "ARIMA forecasts",
  sub_title = NULL,
  this_x_label = "Time",
  this_y_label = " ",
  length_ori = 2,
  do_grid = FALSE,
```

```
  do_background = FALSE,
  this_palette = c("darkgrey", "blue", "darkgreen", "darkgreen"),
  this_guide_legend = "Forecast"
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This is a required entry.
main_title	Character string; main title of plot. Default is 'ARIMA forecasts'.
sub_title	Subtitle for the plot. Default is to generate the subtitle.
this_x_label	Label for X-axis. Default is "Time"
this_y_label	Label for Y-axis. Default is " "
length_ori	Integer scalar; number of years of the original series to show with forecasts. Default is 2 years.
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
this_palette	Array of character strings; color used for original series, forecast, and upper and lower forecast bounds. Default is c("darkgrey", "blue", "darkgreen", "darkgreen").
this_guide_legend	Title for legend. Default is "Forecast"

Details

Version 3.2, 11/6/2024

Value

A ggplot object of the regARIMA forecasts with confidence bounds.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
air_seas <- seasonal::seas(AirPassengers, arima.model = "(0 1 1)(0 1 1)",
                           forecast.maxlead = 60, forecast.save = "fct",
                           series.save = "a1")
air_fcst_p <-
    plot_fcst(air_seas,
              main_title = "Forecasts for Airline Passengers", do_grid = TRUE)
```

plot_fcst_history *Generate forecast history plot*

Description

Generate forecast history plot, which compares the sum of squared forecast errors for two models.

Usage

```
plot_fcst_history(
  seas_mdl1 = NULL,
  seas_mdl2 = NULL,
  main_title = "Differences in the Sum of Squared Forecast Errors",
  name_mdl1 = "Model 1",
  name_mdl2 = "Model 2",
  do_grid = FALSE,
  do_background = FALSE,
  this_x_label = "Time",
  this_y_label = " ",
  this_palette = c("steelblue", "darkgreen"),
  this_guide_legend = "Fcst History"
)
```

Arguments

seas_mdl1	seas object generated from a call of seas on a single time series for the first model This is a required entry.
seas_mdl2	seas object generated from a call of seas on a single time series for the second model This is a required entry.
main_title	Character string; main title of plot. Default is 'Differences in the Sum of Squared Forecast Errors'.
name_mdl1	Character string; Description of first model for use in the subtitle. Default is 'Model 1'.
name_mdl2	Character string; Description of second model for use in the subtitle. Default is 'Model 2'.
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background.
this_x_label	Label for X-axis. Default is "Time"
this_y_label	Label for Y-axis. Default is " "
this_palette	Character array of length 2; color used for each forecast lag. Default is c("steelblue", "darkgreen").
this_guide_legend	Title for legend. Default is "Fcst History"

Details

Version 2.5 11/6/2024

Value

A ggplot object that produces a forecast history plot. If series not specified, print out error message and return NULL.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
air_seas_mdl <-
    seasonal::seas(AirPassengers, x11="", slidingspans = "", transform.function = "log",
                arima.model = "(0 1 1)(0 1 1)", regression.aictest = NULL, outlier = NULL,
                   forecast.maxlead = 36, check.print = c( 'pacf', 'pacfplot' ),
                   history.fstep = c(1, 12), history.estimates = 'fcst',
                   history.save = 'fcsterrors')
air_seas_mdl2 <-
    seasonal::seas(AirPassengers, x11="", slidingspans = "", transform.function = "log",
                   arima.model = "(0 1 1)(0 1 1)", regression.variables = c("td"),
                   forecast.maxlead = 36, check.print = c( 'pacf', 'pacfplot' ),
                   history.fstep = c(1, 12), history.estimates = 'fcst',
                   history.save = 'fcsterrors')
plot_fcst_history_air <-
    plot_fcst_history(air_seas_mdl, air_seas_mdl2,
      main_title = 'Differences in the Sum of Squared Forecast Errors for Airline Passengers',
         name_mdl1 = 'Airline model', name_mdl2 = 'Airline model + regressors')
```

---

plot_fcst_two *Forecast plot for two models*

Description

Generates plot that compares regARIMA forecasts for two models of the same series

Usage

```
plot_fcst_two(
  seas_obj_one = NULL,
  seas_obj_two = NULL,
  main_title = "ARIMA forecasts",
  sub_title = NULL,
  name_mdl1 = "Model 1",
  name_mdl2 = "Model 2",
  this_x_label = "Time",
  this_y_label = " ",
  length_ori = 2,
  do_grid = FALSE,
  do_background = FALSE,
  this_palette = c("darkgrey", "steelblue", "darkgreen"),
  this_guide_legend = "Forecast"
)
```

Arguments

seas_obj_one	seas object generated from a call of seas on a single time series This is a required entry.
seas_obj_two	seas object generated from a call of seas on the same time series, but a different regARIMA model. This is a required entry.
main_title	Character string; main title of plot. Default is 'ARIMA Residuals'.
sub_title	Subtitle for the plot. Default is to generate the subtitle.
name_mdl1	Character string; Description of first model for use in the subtitle. Default is 'Model 1'.
name_mdl2	Character string; Description of second model for use in the subtitle. Default is 'Model 2'.
this_x_label	Label for X-axis. Default is "Time"
this_y_label	Label for Y-axis. Default is " "
length_ori	Integer scalar; number of years of the original series to show with forecasts. #' Default is 2 years.
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
this_palette	Array of character strings; color used for original series, forecast, and upper and lower forecast bounds. Default is c("darkgrey", "steelblue", "darkgreen").
this_guide_legend	Title for legend. Default is "Forecast"

Details

Version 1.7, 11/6/2024

Value

A ggplot object of the regARIMA forecasts for two models of the same series.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
shoes_seas_one <-
    seasonal::seas(shoes2008, x11="", slidingspans = "", transform.function = "log",
                   arima.model = "(0 1 1)(0 1 1)", regression.aictest = NULL,
                   outlier = NULL, forecast.maxlead = 60, forecast.save = "fct",
                   check.print = c( 'pacf', 'pacfplot' ), series.save = "a1")
shoes_seas_two <-
    seasonal::seas(shoes2008, x11="", slidingspans = "", transform.function = "log",
                   arima.model = "(0 1 1)(0 1 1)", regression.variables = c("td"),
                   forecast.maxlead = 60, forecast.save = "fct",
                   check.print = c( 'pacf', 'pacfplot' ), series.save = "a1")
shoes_fcst_two_p <-
    plot_fcst_two(shoes_seas_one, shoes_seas_two,
```

```
main_title = "Forecast Comparison Plot",
name_mdl1 = 'Airline', name_mdl2 = 'Airline + reg',
do_grid = TRUE)
```

`plot_first_difference` *First Difference Plot*

Description

Generates a ggplot2 object of the first difference of a time series grouped by months or quarters.

Usage

```
plot_first_difference(
  this_series = NULL,
  take_log = FALSE,
  main_title = "First Difference Plot",
  remove_y_axis = TRUE,
  x_title_size = 10,
  geom_text_size = 2.5,
  geom_text_color = "steelblue"
)
```

Arguments

`this_series`	`ts` object of a single time series This is a required entry.
`take_log`	Logical scalar, specifies that a log transformation will be taken before differencing. Default is `FALSE`
`main_title`	Title for the plot. Default is character string `'First Difference Plot'`.
`remove_y_axis`	Logical scalar. If `TRUE`, removes the y-axis labels and tick marks from all subplots. Default is `FALSE`, which keeps the y-axis labels and tick marks.
`x_title_size`	Integer scalar, size of the x-axis title. Default is `10`.
`geom_text_size`	Integer scalar, size of the plotting characters. Default is `2.5`.
`geom_text_color`	Character scalar, color of the plotting characters. Default is `"steelblue"`.

Details

Version 1.1, 8/22/2024

Value

A `ggplot` object that produces a plot of the first differences of a series specified by `this_series` grouped by month or quarter. The time series specified should be either a monthly or quarterly series.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
shoes_seas <-
    seasonal::seas(shoes2008, x11.save = "d13",
       x11 = "", transform.function = "log",
       check.print = c("none", "+acf", "+acfplot", "+normalitytest"),
       regression.aictest = c("td", "easter"),
  regression.save = c("td", "hol"),
       outlier.types = "all",
       arima.model = "(0 1 1)(0 1 1)",
       forecast.maxlead = 60)
shoesSA <- seasonal::final(shoes_seas)
p_shoes_sa_diff <-
    plot_first_difference(shoesSA,
        main_title = "Seasonal Adjustment First Differences",
        remove_y_axis = FALSE, x_title_size = 10)
```

`plot_fts` *Final t-statistics for the outlier identification procedure plot*

Description

Generates a plot of the final t-statistics for the outlier identification procedure.

Usage

```
plot_fts(
  seas_obj = NULL,
  start_plot = NULL,
  main_title = "Outlier T-Values",
  this_y_label = NULL,
  this_x_label = "Time",
  add_identified_otl = FALSE,
  color_otl = NULL,
  this_palette = "Dark2"
)
```

Arguments

`seas_obj`	`seas` object generated from a call of `seas` on a single time series This is a requited entry.
`start_plot`	Integer vector of length 2; Starting date for plot. Default is starting date for the time series.
`main_title`	Character string; main title of plot. Default is `'Outlier T-Values'`.
`this_y_label`	Character string; y-axis label for plot, if specified.
`this_x_label`	Label for X axis. Default is `"Time"`.
`add_identified_otl`	Logical scalar; indicates if outlier plots will include identified outliers. Default is not including identified outliers.
`color_otl`	Character array of length 3; color used for different outliers, with the order being `'ao','ls','tc'`. Default is NULL.
`this_palette`	Character string; default `RColorBrewer` palette. Deault is `"Dark2"`.

Details

Version 2.6, 9/5/2024++

Value

A ggplot object which produces a plot of the final t-statistics from the automatic outlier identification procedure.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
air_seas_outlier <-
    seasonal::seas(AirPassengers, arima.model = "(0 1 1)(0 1 1)",
                   outlier.save = "fts", outlier.types = "all")
plot_fts_air <-
    plot_fts(air_seas_outlier,
             main_title = "Outlier T-Values for Airline Passengers")
```

---

plot_fully_differenced_transformed

Plot fully differenced transformed series from an X-13ARIMA-SEATS SEATS seasonal adjustment run.

Description

Generate plot of the fully differenced transformed series from a SEATS adjustment from a seas object generated by the seasonal package.

Usage

```
plot_fully_differenced_transformed(
  seas_obj = NULL,
  this_series = "ori",
  main_title = NULL,
  sub_title = NULL,
  this_y_label = NULL,
  y_limit = NULL,
  this_x_label = "Time",
  start_plot = NULL,
  do_grid = FALSE,
  do_background = FALSE,
  draw_recess = FALSE,
  recess_color = "lightgrey",
  recess_sub = TRUE,
  line_color = "steelblue"
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This is a required entry.
this_series	Character string; series for which SEATS produces a fully differenced transformed series, limited to the original series (ori, the default), the seasonally adjusted series (sa), or the trend component (trend). For other entries, the function will print an error message and return a NULL.
main_title	Character string; main title of plot. A title will be generated if no title is specified.
sub_title	Character string; subtitle of plot. There is no default subtitle.
this_y_label	Character string; y-axis label for plot, if specified.
y_limit	Numeric vector of length 2; Range of values you wish the plot to be plotted over. Default is range of the series specified.
this_x_label	Label for X axis. Default is "Time".
start_plot	Integer vector of length 2; Starting date for plot. Default is starting date for the time series.
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
draw_recess	Logical scalar; indicates if certain plots will have shaded areas for NBER recession dates. Default is no recession shading.
recess_color	Character string; color used for shading of recession region. Default is 'lightgrey'.
recess_sub	Logical scalar; indicates if x-axis label for recession is produced for this plot. Default is x-axis label.
line_color	Character string; color used for series in the plot. Default is 'steelblue'.

Details

Version 1.6, 8/29/2024

Value

A ggplot object that generates a plot of the fully differenced transformed series from a SEATS seasonal adjustment. If SEATS seasonal adjustment not produced, print out error message and return NULL.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
shoes_seats_seas <-
   seasonal::seas(shoes2008, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log",
                  forecast.maxlead = 36,
                  check.print = c( 'pacf', 'pacfplot' ),
                  seats.finite = "yes",
```

```
                  seats.save = c( 'dor', 'dsa', 'dtr' ) )
p_ori_full_diff <- plot_fully_differenced_transformed(shoes_seats_seas, "ori",
sub_title = "US Shoe Sales",
     do_grid = TRUE,
line_color = "darkgreen")
p_sa_full_diff <- plot_fully_differenced_transformed(shoes_seats_seas, "sa",
sub_title = "US Shoe Sales",
     do_grid = TRUE,
line_color = "darkgreen")
p_trn_full_diff <- plot_fully_differenced_transformed(shoes_seats_seas, "trn",
sub_title = "US Shoe Sales",
     do_grid = TRUE,
line_color = "darkgreen")
```

`plot_matrix` *Plot time series matrix*

Description

Generate plot of a matrix of user-specified time series.

Usage

```
plot_matrix(
  this_matrix = NULL,
  main_title = deparse(substitute(this_matrix)),
  sub_title = NULL,
  this_y_label = NULL,
  y_limit = NULL,
  this_x_label = "Time",
  start_plot = NULL,
  do_grid = FALSE,
  do_background = FALSE,
  line_color = NULL,
  this_palette = "Dark2",
  this_line_type = "solid",
  do_facet = FALSE,
  reset_facet_y_axis = FALSE
)
```

Arguments

`this_matrix`	Numeric matrix; columns of time series object to be plotted.
`main_title`	Character string; main title of plot. The default title is the name of the matrix passed to this function.
`sub_title`	Character string; subtitle of plot. There is no default subtitle.
`this_y_label`	Character string; y-axis label for plot, if specified.
`y_limit`	Numeric vector of length 2; Range of values on plot y-axis Default is range of the series specified.
`this_x_label`	Label for X axis. Default is `"Time"`.

start_plot	Integer vector of length 2; Starting date for plot. Default is starting date for the time series.
do_grid	Logical scalar; indicates if plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
line_color	Character scalar; color used for plot. User should specify one color for each column of the matrix specified. Default is the RColorBrewer palette "Dark2".
this_palette	Character string; default RColorBrewer palette. Deault is "Dark2".
this_line_type	Character string; indicates line type of each plot produced. Default is "solid".
do_facet	Logical scalar; indicates if a facet plot is generated of the different colums. Default is FALSE.
reset_facet_y_axis	Logical scalar; indicates if y-axis for facet plots reset to y_limit Default is FALSE.

Details

Version 4.2, 11/6/2024

Value

A ggplot object that produces a plot of user-specified time series. If matrix not specified, print out error message and return NULL.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
BP_Region_Matrix <-
   cbind(blsplotGG::xt_data_list$mw1u, blsplotGG::xt_data_list$ne1u,
         blsplotGG::xt_data_list$so1u, blsplotGG::xt_data_list$we1u)
colnames(BP_Region_Matrix) <- names(blsplotGG::xt_data_list)
p_BP <- blsplotGG::plot_matrix(BP_Region_Matrix, this_y_label = 'Building Permits',
   main_title = "US Building Permits, 1 Family Units",
   do_grid = TRUE, this_line_type = rep("solid", 4),
   line_color = c("orange", "steelblue", "forestgreen", "brown"))
p_BP_facet <- blsplotGG::plot_matrix(BP_Region_Matrix, this_y_label = 'Building Permits',
   main_title = "US Building Permits, 1 Family Units",
   do_grid = FALSE, do_facet = TRUE,
   line_color = c("orange", "steelblue", "forestgreen", "brown"))
```

---

plot_maximum_percent_difference

Maximum percent difference plot

Description

Generates a ggplot object with a time series of the maximum percent difference from a sliding spans analysis of seasonal factors or changes.

Usage

```
plot_maximum_percent_difference(
  seas_obj = NULL,
  this_series = "sfs",
  main_title = "Maximum Percent Difference Plot",
  sub_title = NULL,
  this_x_label = "Time",
  this_y_label = " ",
  do_grid = FALSE,
  do_background = FALSE,
  line_color = "steelblue",
  cut_color = "red"
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This is a required entry.
this_series	Character string; three character code for the type of series to be generated. Allowed entries are "sfs" (seasonal factors, default), "chs" (period-to-period changes), "sis" (indirect seasonal factors), "cis" (indirect period-to-period changes).
main_title	Title for the plot. Default is character string 'Maximum Percent Difference Plot'.
sub_title	Subtitle for the plot. Optional entry.
this_x_label	Label for X-axis. Default is "Time"
this_y_label	Label for Y-axis. Default is " "
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
line_color	Color used for lines in the maximum percentage difference plot. Default is "steelblue".
cut_color	Color use to show the slidings spans cut off for this type of series. Default is "red".

Details

Version 1.4, 11/7/2024

Value

A ggplot object that generates a plot of the maximum percent difference from a sliding spans analysis of seasonal factors or changes.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
shoes_sspan_seas <-
   seasonal::seas(blsplotGG::shoes2008,
                  arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log",
                  forecast.maxlead = 36,
                  slidingspans.save = c('sfs', 'chs'))
p_shoes_maxpct_sf <-
    plot_maximum_percent_difference(shoes_sspan_seas, "sfs",
                      main_title = "Maximum Percent Difference Plot of Seasonal Factors",
                        sub_title = "US Shoe Sales",
                        line_color = "blue", cut_color = "purple")
```

---

plot_maximum_percent_difference_by_period

Maximum percent difference for sliding spans by period

Description

Generate a plot of the maximum percent difference from a sliding spans analysis by month or quarter.

Usage

```
plot_maximum_percent_difference_by_period(
  seas_obj = NULL,
  this_series = "sfs",
  main_title = NULL,
  sub_title = NULL,
  this_y_label = NULL,
  this_x_label = NULL,
  do_grid = FALSE,
  do_background = FALSE,
  line_color = NULL,
  this_palette = "Set3",
  cut_color = "red"
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This is a required entry.
this_series	Character string; three character code for the type of series to be generated. Allowed entries are "sfs" (seasonal factors, default), "chs" (period-to-period changes), "sis" (indirect seasonal factors), "cis" (indirect period-to-period changes).
main_title	Character string; main title of plot. Default is character string 'Maximum Percent Difference Plot by Period', where Period is replaced by Month or Quarter.
sub_title	Character string; subtitle of plot. There is no default subtitle.
this_y_label	Character string; y-axis label for plot, if specified.

this_x_label	Label for X axis. Default is "Month" or "Quarter".
do_grid	Logical scalar; indicates if plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background.
line_color	Character scalar; color used for plot. User should specify one color for each column of the matrix specified. Default is the RColorBrewer palette "Set3".
this_palette	Character string; default RColorBrewer palette. Default is "Set3".
cut_color	Color use to show the slidings spans cut off for this type of series. Default is "red".

Details

Version 1.4, 11/7/2024

Value

Generate ggplot object generating a plot of the maximum percent difference from a sliding spans analysis by month or quarter

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
shoes_sspan_seas <-
   seasonal::seas(blsplotGG::shoes2008,
                  arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log",
                  forecast.maxlead = 36,
                  slidingspans.save = c('sfs', 'chs'))
p_shoes_maxpct_sf <-
    plot_maximum_percent_difference_by_period(shoes_sspan_seas, "sfs",
                      main_title = "Maximum Percent Difference Plot of Seasonal Factors",
                        sub_title = "US Shoe Sales",
                        this_palette = "Paired")
```

---

plot_ratio *Ratio plot*

Description

Generates a high-definition plot around a reference line other than zero.

Usage

```
plot_ratio(
  ratio_series = NULL,
  ratio_range = NULL,
  main_title = "Ratio Plot",
  sub_title = NULL,
  this_x_label = "Time",
  this_y_label = "Ratio",
  do_grid = FALSE,
  do_background = FALSE,
  draw_recess = FALSE,
  recess_color = "lightgrey",
  recess_sub = TRUE,
  add_line = TRUE,
  ratio_mean = 1,
  ratio_color = "steelblue"
)
```

Arguments

ratio_series	Time series of ratios/factors for which you want to generate a high definition plot
ratio_range	Range of values you wish the plot to be plotted over. Default is range of the series.
main_title	Title for the plot. Default is character string 'Ratio Plot'.
sub_title	Subtitle for the plot. Default is NULL.
this_x_label	Label for X axis. Default is "Time".
this_y_label	Label for Y axis. Default is "Ratio".
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
draw_recess	Logical scalar; indicates if certain plots will have shaded areas for NBER recession dates. Default is no recession shading.
recess_color	Character string; color used for shading of recession region. Default is 'lightgrey'.
recess_sub	Logical scalar; indicates if x-axis label for recession is produced for this plot. Default is x-axis label.
add_line	Logical scalar; add solid line for assumed mean. Default is TRUE.
ratio_mean	Assumed mean value for the ratio. Default is 1.0
ratio_color	Color used for lines in ratio plot. Default is "steelblue".

Details

Version 3.2, 11/6/2024

Value

A ggplot object that produces a ratio plot of a time series.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
ukgas_x11_seas <-
   seasonal::seas(UKgas, series.period = 4, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 20,
                  x11.seasonalma = "s3x5", check.print = c( 'pacf', 'pacfplot' ))
ukgas_x11_sf     <- seasonal::series(ukgas_x11_seas, "d10")
ukgas_sf_ratio   <- plot_ratio(ukgas_x11_sf, main_title = "UK Gas X-11 Seasonal Factors",
                               sub_title = "Seasonal Filter: 3x5")
```

---

plot_ratio_facet *Ratio facet plot*

Description

Generates a high-definition plot for a number of X-13 factors around a reference line other than zero.

Usage

```
plot_ratio_facet(
  seas_obj = NULL,
  ratio_tables = NULL,
  main_title = "Ratio Facet Plot",
  sub_title = NULL,
  this_x_label = "Time",
  this_y_label = "Ratio",
  do_grid = FALSE,
  do_background = FALSE,
  add_line = TRUE,
  this_scale = "fixed",
  ratio_mean = 1,
  ratio_color = "steelblue"
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This is a required entry.
ratio_tables	Array of tables for which you want to generate a high definition plot. Possible entries are "sf" (seasonal factor), "td" (trading day factor), "hol" (holiday factors), and "irr" (irregular).
main_title	Title for the plot. Default is character string 'Ratio Facet Plot'.
sub_title	Subtitle for the plot. Default is NULL.
this_x_label	Label for X axis. Default is "Time".
this_y_label	Label for Y axis. Default is "Ratio".

do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
add_line	Logical scalar; add solid line for assumed mean. Default is TRUE.
this_scale	Character scalar; set the scale for the y-axis of the facet plots. Default is "fixed". To set axis to be different for the different plots, set this_scale = "free_y".
ratio_mean	Assumed mean value for the ratio. Default is 1.0
ratio_color	Color used for lines in ratio plot. Default is "steelblue".

Details

Version 3.0, 1/6/2025

Value

A ggplot object that generates a ratio facet plot for the factors provided in the ratio_tables argument.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
shoes_x11_seas <-
   seasonal::seas(shoes2008, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 36,
                  regression.aictest = c("td", "easter"),
              regression.save = c("td", "hol"),
                  x11.seasonalma = "s3x5", check.print = c( 'pacf', 'pacfplot' ))
shoes_factor_ratio_one   <-
   plot_ratio_facet(shoes_x11_seas, c("sf", "irr"),
                    main_title = "US Retail Shoe Sales",
                    sub_title = "X-11 Seasonal (3x5) and Irregular Factors")
shoes_factor_ratio_two   <-
   plot_ratio_facet(shoes_x11_seas, c("td", "hol", "irr"),
                    main_title = "US Retail Shoe Sales",
                    sub_title = "Trading Day, Easter, and Irregular Factors",
                    this_scale = "free_y")
```

---

plot_ratio_two *Ratio plots for two series*

Description

Generates ratio plots for two series, and a third plot of the ratio/difference of the two series if the user requests it.

Usage

```
plot_ratio_two(
  ratio_one = NULL,
  ratio_two = NULL,
  ratio_range = NULL,
  do_comparison_plot = TRUE,
  overall_title = NULL,
  ratio_one_title = "First Ratio Plot",
  ratio_two_title = "Second Ratio Plot",
  comparison_title = "Comparison Plot",
  this_x_label = NULL,
  this_y_label = NULL,
  do_grid = FALSE,
  do_background = FALSE,
  add_line = TRUE,
  ratio_mean = c(1, 1),
  ratio_color = "steelblue"
)
```

Arguments

ratio_one	First time series of ratios/factors for which you want to generate a high definition plot. This is a required entry.
ratio_two	Second time series of ratios/factors for which you want to generate a high definition plot. This is a required entry.
ratio_range	Range of values you wish the plot to be plotted over. Default is range of the series, if they are the same type of factor.
do_comparison_plot	Logical scalar. If TRUE, a ratio/difference plot of the two factors will be generated from ratio_one and ratio_two, if the series have the same periodicity and are the same type of factor. Default is TRUE.
overall_title	Title for the combined plot. Default is NULL.
ratio_one_title	Title for the first plot. Default is character string 'First Ratio Plot'.
ratio_two_title	Title for the plot. Default is character string 'Second Ratio Plot'.
comparison_title	Title for the comparison plot, if requested. Default is character string 'Comparison Plot'.
this_x_label	Label for X axis. Default is "Time".
this_y_label	Label for Y axis. Default is "Ratio".
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
add_line	Logical scalar; add solid line for assumed mean. Default is TRUE.
ratio_mean	Numeric vector of length two; the assumed mean value for the ratio of each series. Default is c(1.0, 1.0)
ratio_color	Color used for lines in ratio plot. Default is "steelblue".

Details

Version 1.5, 1/2/2025

Value

A ggplot object that generates a stacked plot with the

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
ukgas_x11_seas   <-
   seasonal::seas(UKgas, series.period = 4, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 20,
                  x11.seasonalma = "s3x5", x11.save = c("d10", "d11"),
                  check.print = c( 'pacf', 'pacfplot' ))
ukgas_seats_seas <-
   seasonal::seas(UKgas, series.period = 4, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 20,
                  seats.save = c("s10", "s11"),
                  check.print = c( 'pacf', 'pacfplot' ))
ukgas_x11_sf      <- seasonal::series(ukgas_x11_seas, "d10")
ukgas_seats_sf    <- seasonal::series(ukgas_seats_seas, "s10")
ukgas_sf_two_plot <-
     plot_ratio_two(ukgas_x11_sf, ukgas_seats_sf,
         overall_title = "UK Gas Production",
ratio_one_title = "X-11 Seasonal Factors",
ratio_two_title = "SEATS Seasonal Factors",
comparison_title = "Ratio of Seasonal Factors (X11 / SEATS)",
         ratio_color = "darkgreen")
```

---

plot_resid *Residual plot*

Description

Generates a plot of the regARIMA residuals with diagnostic information

Usage

```
plot_resid(
  seas_obj = NULL,
  main_title = "ARIMA Residuals",
  series_name = NULL,
  this_x_label = "Time",
  this_y_label = NULL,
  do_grid = TRUE,
  do_background = FALSE,
  draw_recess = FALSE,
  recess_color = NULL,
  recess_sub = TRUE,
```

```
  use_ratio = FALSE,
  add_line = TRUE,
  line_color = "steelblue"
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This is a requited entry.
main_title	Character string; main title of plot. Default is 'ARIMA Residuals'.
series_name	Character scalar; name of the time series used in seas_obj.
this_x_label	Label for X axis. Default is "Time".
this_y_label	Label for Y axis. Default is series_name. if specified.
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is grid lines plotted.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
draw_recess	Logical scalar; indicates if certain plots will have shaded areas for NBER recession dates. Default is recession shading not plotted.
recess_color	Character string; color used for shading of recession region. Default is 'lightgrey'.
recess_sub	Logical scalar; indicates if x-axis label for recession is produced for this plot. Default is x-axis label is produced
use_ratio	Logical scalar; indicates if plots of seasonal factors, irregular, and residuals are done as ratio plots. Default has these plots as time series line plots.
add_line	Logical scalar; add solid line for assumed mean. Default is TRUE.
line_color	Character string; color used for residuals. Default is "green".

Details

Version 2.6, 11/6/2024

Value

Generates a ggplot object of a plot of the regARIMA residuals with diagnostic information in the sub-headers.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
air_seas <- seasonal::seas(AirPassengers, arima.model = "(0 1 1)(0 1 1)")
plot_resid_air <-
     plot_resid(air_seas, main_title = "ARIMA Residuals for Airline Passengers",
                use_ratio = TRUE, line_color="darkblue")
```

plot_sadj_and_ori *Plot X-13 seasonal adjustment, original series on same axis*

Description

Generates a ggplot object with a time series plot that compares an X-13 seasonal adjustment and trend, optionally including the original series.

Usage

```
plot_sadj_and_ori(
  this_sa = NULL,
  this_ori = NULL,
  this_sa_type = "SEATS",
  main_title = NULL,
  sub_title = NULL,
  this_x_label = "Time",
  this_y_label = " ",
  do_grid = FALSE,
  do_background = FALSE,
  line_color = c("grey", "darkblue"),
  line_label = c("Ori", "SA"),
  this_guide_legend = "Series"
)
```

Arguments

this_sa	Time series of a seasonal adjustment. This is a required entry.
this_ori	Time series of the original series. Optional entry.
this_sa_type	Character string; type of seasonal adjustment. Default is "SEATS".
main_title	Title for the plot. By default, the routine will generate a title based on the type of adjustment (X-11 and SEATS) done.
sub_title	Subtitle for the plot. Optional entry.
this_x_label	Label for X-axis. Default is "Time"
this_y_label	Label for Y-axis. Default is " "
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
line_color	Character vector of length 2; color used for lines in the plot, in the order of original series, seasonally adjusted series. Default is c("grey", "darkblue").
line_label	Caracter vector of length 2; labels used for lines in the plot, in the order of original series, seasonally adjusted series. Default is c("Ori", "SA").
this_guide_legend	Title for legend. Default is "Series"

Details

Version 2.3, 1/2/2025

Value

A ggplot object that generates a plot comparing a seasonally adjusted series with the original series.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
ukgas_x11_seas   <-
   seasonal::seas(UKgas, series.period = 4, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 20,
                  x11.seasonalma = "s3x5",
                  check.print = c( 'pacf', 'pacfplot' ))
ukgas_x11_sa     <- seasonal::final(ukgas_x11_seas)
ukgas_sa_ori_p <-
    plot_sadj_and_ori(this_sa = ukgas_x11_sa,
                      this_ori = UKgas,
 this_sa_type = "X-11",
                      main_title = "UK Gas Seasonal Adjustment",
                      line_color = c("lightgrey", "forestgreen"))
```

---

plot_sadj_and_trend *Plot X-13 seasonal adjustment, trend on same axis*

Description

Generates a ggplot object with a time series plot that compares an X-13 seasonal adjustment and trend, optionally including the original series.

Usage

```
plot_sadj_and_trend(
  seas_obj = NULL,
  plot_ori = TRUE,
  main_title = NULL,
  sub_title = NULL,
  this_x_label = "Time",
  this_y_label = " ",
  do_grid = FALSE,
  do_background = FALSE,
  line_color = NULL,
  this_palette = "Dark2",
  line_label = NULL,
  this_guide_legend = "Series"
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This is a required entry.
plot_ori	Logical scalar, indicates if original series is included with plot. Default is TRUE.

main_title	Title for the plot. By default, the routine will generate a trend based on the type of adjustment (X-11 and SEATS) done.
sub_title	Subtitle for the plot. Optional entry.
this_x_label	Label for X-axis. Default is "Time"
this_y_label	Label for Y-axis. Default is " "
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
line_color	Character vector of length 2 (if plot_ori is FALSE) or 3 (if plot_ori is TRUE); color used for lines in the plot, in the order of original series (if specified), seasonally adjusted series, trend. Default is generated from the RColorBrewer palette "Dark2".
this_palette	Character string; default RColorBrewer palette. Deault is "Dark2".
line_label	Caracter vector of length 2; labels used for lines in the plot, in the order of original series, seasonally adjusted series. Default is c("Ori", "SA", "Trend") (if original series plotted) or c("SA", "Trend").
this_guide_legend	Title for legend. Default is "Series"

Details

Version 2.0, 1/2/2025

Value

A ggplot object that generates a plot comparing a seasonally adjusted series with the trend generated from the same X-13ARIMA-SEATS seasonal adjustment.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
ukgas_x11_seas   <-
   seasonal::seas(UKgas, series.period = 4, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 20,
                  x11.seasonalma = "s3x5", x11.save = "d12",
                  check.print = c( 'pacf', 'pacfplot' ))
ukgas_x11_sadj_and_trend_p <-
    plot_sadj_and_trend(ukgas_x11_seas, plot_ori = TRUE,
                        main_title = "UK Gas",
                        sub_title = "X-11 Seasonal Adjustment",
                        line_color = c("lightgrey", "steelblue", "forestgreen"))
ukgas_seats_seas <-
   seasonal::seas(UKgas, series.period = 4, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 20,
                  seats.save = "s12",
                  check.print = c( 'pacf', 'pacfplot' ))
ukgas_seats_sadj_and_trend_p <-
    plot_sadj_and_trend(ukgas_seats_seas, plot_ori = FALSE,
```

```
                        main_title = "UK Gas Series",
                        sub_title = "SEATS Seasonal Adjustment",
                        line_color = c("steelblue", "forestgreen"))
```

`plot_sadj_and_trend_facet` *Plot X-13 seasonal adjustment, trend in a facet plot*

Description

Generates a `ggplot` object with a time series facet plot that compares an X-13 seasonal adjustment and trend, optionally including the original series.

Usage

```
plot_sadj_and_trend_facet(
  seas_obj = NULL,
  plot_ori = TRUE,
  main_title = NULL,
  sub_title = NULL,
  this_x_label = "Time",
  this_y_label = " ",
  line_color = NULL,
  this_palette = "Dark2",
  line_label = NULL,
  this_scale = "fixed",
  remove_legend = FALSE
)
```

Arguments

`seas_obj`	`seas` object generated from a call of `seas` on a single time series This is a required entry.
`plot_ori`	Logical scalar, indicates if original series is included with plot. Default is `TRUE`.
`main_title`	Character string; title for the plot. By default, the routine will generate a title based on the type of adjustment (X-11 and SEATS) done.
`sub_title`	Subtitle for the plot. Optional entry.
`this_x_label`	Label for X-axis. Default is `"Time"`
`this_y_label`	Label for Y-axis. Default is `" "`
`line_color`	Character vector of length 2 (if `plot_ori` is `FALSE`) or 3 (if `plot_ori` is `TRUE`); color used for lines in the plot, in the order of original series (if specified), seasonally adjusted series, trend. Default is generated from the `RColorBrewer` palette `"Dark2"`.
`this_palette`	Character string; default `RColorBrewer` palette. Deault is `"Dark2"`.
`line_label`	Caracter vector of length 2; labels used for lines in the plot, in the order of original series, seasonally adjusted series. Default is `c("Ori", "SA", "Trend")` (if original series plotted) or `c("SA", "Trend")`.
`this_scale`	Character scalar; set the scale for the y-axis of the facet plots. Default is `"fixed"`. To set axis to be different for the different plots, set `this_scale = "free_y"`.
`remove_legend`	Logical scalar; if TRUE, plot legend will be removed. Default is FALSE.

Details

Version 3.0, 1/7/2025

Value

A ggplot object that generates a facet plot comparing a seasonally adjusted series with the trend generated from the same X-13ARIMA-SEATS seasonal adjustment.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
ukgas_x11_seas   <-
   seasonal::seas(UKgas, series.period = 4, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 20,
                  x11.seasonalma = "s3x5", x11.save = "d12",
                  check.print = c( 'pacf', 'pacfplot' ))
ukgas_x11_sadj_and_trend_facet_p <-
    plot_sadj_and_trend_facet(ukgas_x11_seas, plot_ori = TRUE,
                         main_title = "UK Gas",
                         sub_title = "X-11 Seasonal Adjustment",
                         line_color = c("steelblue", "forestgreen", "grey"))
ukgas_seats_seas <-
   seasonal::seas(UKgas, series.period = 4, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 20,
                  seats.save = "s12",
                  check.print = c( 'pacf', 'pacfplot' ))
ukgas_seats_sadj_and_trend_facet_p <-
    plot_sadj_and_trend_facet(ukgas_seats_seas, plot_ori = FALSE,
                         main_title = "UK Gas Series",
                         sub_title = "SEATS Seasonal Adjustment",
                         line_color = c("steelblue", "forestgreen"),
                         remove_legend = TRUE)
```

---

plot_sa_history *Revisions History Plot for Seasonal Adjustments*

Description

Generates a ggplot2 object of estimates from a revisions history of a seasonal adjustment for a given series.

Usage

```
plot_sa_history(
  seas_obj = NULL,
  add_ori = TRUE,
  main_title = "Seasonal Adjustment History Graph",
  sub_title = NULL,
  this_x_label = "Time",
  this_y_label = NULL,
```

```
  do_grid = FALSE,
  do_background = FALSE,
  line_color = NULL,
  this_palette = "Dark2"
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This is a required entry.
add_ori	Logical scalar; add the original series to the plot. Default is TRUE.
main_title	Character string. Title for the plot. Default is 'Seasonal Adjustment History Graph'.
sub_title	Subtitle for the plot. Default is NULL.
this_x_label	Label for X axis. Default is "Time".
this_y_label	Label for Y axis. Default is NULL.
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
line_color	Vector with colors used for lines in history plot. Should be of length 4 (if add_ori = FALSE) or 5 (if add_ori = TRUE) Default is NULL, which indicates that the palette specified in this_palette is used to generate colors for this plot.
this_palette	Color used for lines in plot. Default is "Dark2"

Details

Version 1.9, 11/6/2024

Value

A ggplot object that generates a history plot of the seasonal adjustment. The seas object in the seas_obj argument must contain output for a revisions history analysis for seasonal adjustments with sadjlags set to 1 and 12 (for monthly series) or 4 (for quarterly series).

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
shoes_seas <-
    seasonal::seas(shoes2008, x11.save = "d13",
       x11 = "", transform.function = "log",
       check.print = c("none", "+acf", "+acfplot", "+normalitytest"),
       regression.aictest = c("td", "easter"),
  regression.save = c("td", "hol"),
       outlier.types = "all",
       arima.model = "(0 1 1)(0 1 1)",
       forecast.maxlead = 60,
      history.estimates = c("sadj", "sadjchng", "trend"),
```

```
      history.sadjlags = c(1,12),
      history.print = "all",
      history.save = c("sae", "sar"))
p_shoes_sa_history <-
    plot_sa_history(shoes_seas, add_ori = FALSE,
        main_title = "SA History Graph, Lag 1 and 12",
        sub_title = "US Retail Sales of Shoes")
```

`plot_seasonal_sums` *Plot of the seasonal period length sums of the SEATS seasonal factors from an X-13ARIMA-SEATS SEATS seasonal adjustment run.*

Description

Generate plot of the seasonal period length sums of the SEATS seasonal factors from a SEATS adjustment from a `seas` object generated by the `seasonal` package.

Usage

```
plot_seasonal_sums(
  seas_obj = NULL,
  main_title = NULL,
  sub_title = NULL,
  this_y_label = NULL,
  this_x_label = "Time",
  do_grid = FALSE,
  do_background = FALSE,
  draw_recess = FALSE,
  recess_color = "lightgrey",
  recess_sub = TRUE,
  line_color = "steelblue"
)
```

Arguments

`seas_obj`	`seas` object generated from a call of `seas` on a single time series This is a required entry.
`main_title`	Character string; main title of plot. A title will be generated if no title is specified.
`sub_title`	Character string; subtitle of plot. There is no default subtitle.
`this_y_label`	Character string; y-axis label for plot, if specified.
`this_x_label`	Label for X axis. Default is `"Time"`.
`do_grid`	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
`do_background`	Logical scalar; indicates grey background included in plot. Default is no grey background;
`draw_recess`	Logical scalar; indicates if certain plots will have shaded areas for NBER recession dates. Default is no recession shading.
`recess_color`	Character string; color used for shading of recession region. Default is `'lightgrey'`.

recess_sub	Logical scalar; indicates if x-axis label for recession is produced for this plot. Default is x-axis label.
line_color	Character string; color used for series in the plot. Default is 'steelblue'.

Details

Version 1.5, 8/29/2024

Value

A ggplot object which generates a plot of the seasonal period length sums of the SEATS seasonal factors. If SEATS seaonal adjustent not producted, print out error message and return NULL.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
shoes_seats_seas <-
   seasonal::seas(shoes2008, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log",
                  forecast.maxlead = 36,
                  check.print = c( 'pacf', 'pacfplot' ),
                  seats.finite = "yes",
                  seats.save = c( 'ssm' ) )
p_seasonal_sum <- plot_seasonal_sums(shoes_seats_seas,
sub_title = "US Shoe Sales",
     do_grid = TRUE,
line_color = "darkgreen")
```

---

plot_seats_filter *Plot adjustment or trend filter from an X-13ARIMA-SEATS SEATS seasonal adjustment run.*

Description

Generate plot of the fully differenced transformed series from a SEATS adjustment from a seas object generated by the seasonal package.

Usage

```
plot_seats_filter(
  seas_obj = NULL,
  this_series = "sa",
  this_filter = "symmetric",
  main_title = NULL,
  sub_title = NULL,
  this_y_label = "Filter Coefficient",
  this_x_label = "Index",
  do_grid = FALSE,
  do_background = FALSE,
```

```
  point_color = "steelblue",
  point_size = 1.5,
  point_shape = 20
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This is a required entry.
this_series	Character string; series for which SEATS produces a set of filter weights, limited to the seasonally adjusted series (sa, the default), or the trend component (trend). For other entries, the function will print an error message and return a NULL.
this_filter	Character string; type of filter for which SEATS produces filter coefficients, limited to the symmetric (symmetric, the default), or the concurrent (trend) filter. For other entries, the function will print an error message and return a NULL.
main_title	Character string; main title of plot. A title will be generated if no title is specified.
sub_title	Character string; subtitle of plot. There is no default subtitle.
this_y_label	Character string; y-axis label for plot. Default is "Filter Coefficient".
this_x_label	Label for X axis. Default is "Index".
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
point_color	Character string; color used for points in the plot. Default is 'steelblue'.
point_size	Integer scalar; relative size of points in filter plot. Default is 1.5.
point_shape	Integer scalar; code for the shape of points in filter plot. Default is 20, a small filled circle.

Details

Version 2.4, 11/6/2024

Value

A ggplot object that generates a plot of the adjustment or trend filter from a SEATS seasonal adjustment from X-13ARIMA-SEATS. If SEATS seaonal adjustment (with finite = yes) not producted, print out error message and return NULL.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
shoes_seats_seas <-
   seasonal::seas(shoes2008, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log",
                  forecast.maxlead = 36,
                  check.print = c( 'pacf', 'pacfplot' ),
                  seats.finite = "yes",
                  seats.save = c( 'fac', 'faf', 'ftc', 'ftf' ) )
p_sa_sym_filter <- plot_seats_filter(shoes_seats_seas, "sa", "symmetric",
sub_title = "US Shoe Sales",
point_color = "darkgreen")
p_trn_sym_filter <- plot_seats_filter(shoes_seats_seas, "trn", "symmetric",
sub_title = "US Shoe Sales",
point_color = "darkgreen")
p_sa_conc_filter <- plot_seats_filter(shoes_seats_seas, "sa", "concurrent",
sub_title = "US Shoe Sales",
point_color = "darkgreen")
p_trn_conc_filter <- plot_seats_filter(shoes_seats_seas, "trn", "concurrent",
sub_title = "US Shoe Sales",
point_color = "darkgreen")
```

---

plot_series *Plot time series object.*

Description

Generate plot of user-specified time series (ts) object.

Usage

```
plot_series(
  this_series = NULL,
  this_series_name = NULL,
  main_title = NULL,
  sub_title = NULL,
  this_y_label = NULL,
  y_limit = NULL,
  this_x_label = "Time",
  start_plot = NULL,
  do_grid = FALSE,
  do_background = FALSE,
  draw_recess = FALSE,
  recess_color = "lightgrey",
  recess_sub = TRUE,
  this_line_type = "solid",
  line_color = "grey"
)
```

Arguments

this_series	Time series object; This is a required entry.

`this_series_name`
: Character string; name of time series. No default.

`main_title`
: Character string; main title of plot. A title will be generated if no title is specified.

`sub_title`
: Character string; subtitle of plot. There is no default subtitle.

`this_y_label`
: Character string; y-axis label for plot. If not specified, set to `this_series_name`, if specified.

`y_limit`
: Numeric vector of length 2; Range of values you wish the plot to be plotted over. Default is range of the series specified.

`this_x_label`
: Label for X axis. Default is `"Time"`.

`start_plot`
: Integer vector of length 2; Starting date for plot. Default is starting date for the time series.

`do_grid`
: Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.

`do_background`
: Logical scalar; indicates grey background included in plot. Default is no grey background;

`draw_recess`
: Logical scalar; indicates if certain plots will have shaded areas for NBER recession dates. Default is no recession shading.

`recess_color`
: Character string; color used for shading of recession region. Default is `'lightgrey'`.

`recess_sub`
: Logical scalar; indicates if x-axis label for recession is produced for this plot. Default is x-axis label.

`this_line_type`
: Character string; indicates line type of each plot produced. Default is "solid".

`line_color`
: Character string; color used for series in the plot. Default is `'grey'`.

Details

Version 2.5, 11/6/2024

Value

Generate `ggplot` plot of user-specified series. If series not specified, print out error message and return NULL.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
air_seas <-
  seasonal::seas(AirPassengers, arima.model = "(0 1 1)(0 1 1)", x11="",
                 transform.function = "log")
air_seas_d11 <- seasonal::final(air_seas)
air_d11_p <- plot_series(air_seas_d11, this_series_name = "AirPassengers",
         main_title = 'X-11 Seasonal Adjustment of Airline Passengers',
         sub_title = 'Box-Jenkins Airline series',
         do_grid = TRUE, draw_recess = TRUE, line_color = "darkblue")
```

plot_sf_mean *Seasonal factor mean plot using ggplot*

Description

Generates a plot of the means of the seasonal factors

Usage

```
plot_sf_mean(
  this_sf_matrix = NULL,
  main_title = deparse(substitute(this_sf_matrix)),
  sub_title = NULL,
  this_y_label = NULL,
  this_x_label = "Time",
  do_grid = FALSE,
  do_background = FALSE,
  line_color = NULL,
  this_palette = "Dark2",
  this_line_type = rep("solid", ncol(this_sf_matrix)),
  do_facet = FALSE,
  this_scale = "free_y",
  y_limit = NULL,
  forecast = 0,
  this_legend_title = "SF Means",
  this_legend_entry = colnames(this_sf_matrix)
)
```

Arguments

this_sf_matrix	time series object of the seasonal factors from a seasonal adjustment
main_title	Character string; main title of plot. Default is 'Mean of Seasonal Factors'.
sub_title	Character string; subtitle of plot. There is no default subtitle.
this_y_label	Character string; y-axis label for plot, if specified.
this_x_label	Label for X axis. Default is "Time".
do_grid	Logical scalar; indicates if plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
line_color	Character scalar; color used for plot. User should specify one color for each column of the matrix specified. Default is the RColorBrewer palette "Dark2".
this_palette	Character string; default RColorBrewer palette Deault is "Dark2".
this_line_type	Character string; indicates line type of each plot produced. Default is rep("solid", ncol(this_sf_matrix)).
do_facet	Logical scalar; indicates if a facet plot is generated of the different colums. Default is FALSE.
this_scale	Character scalar; set the scale for the y-axis of the facet plots. Default is "free_y". To set axis to be the same for the different plots, set this_scale = "fixed".

y_limit — Numeric vector of length 2; Range of values you wish the plot to be plotted over. Default is range of the seasonal factors.

forecast — Integer scalar; Number of forecasts appended to the seasonal factors. Default is 0.

this_legend_title — Character string; indicates title of legend. Default is 'Series'.

this_legend_entry — Character array; entries for the legend. When do_facet = TRUE, entries are used as the facet labels. Default is colnames(this_sf_matrix)

Details

Version 3.1, 1/7/2025

Value

Generate plot of the means of seasonal factors by period. If seasonal factors not specified, print out error message and return NULL.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
EM_individual_seas <-
 seasonal::seas(
    x11 = "", transform.function = "log",
    check.print = c("none", "+acf", "+acfplot", "+normalitytest"),
    regression.aictest = NULL,
    outlier.types = "all",
    arima.model = "(0 1 1)(0 1 1)",
    forecast.maxlead = 60,
    list = list(
        list(x = employment_list$n2000013),
        list(x = employment_list$n2000014),
        list(x = employment_list$n2000025),
        list(x = employment_list$n2000026)
    )
)
EM_names <- names(employment_list)
# Use Filter function to grab seas objects
EM_individual_seas_only <-
  Filter(function(x) inherits(x, "seas"), EM_individual_seas)

names(EM_individual_seas_only) <- EM_names

EM_Comp_Sf <-
 cbind(seasonal::series(EM_individual_seas_only$n2000013, "d10"),
      seasonal::series(EM_individual_seas_only$n2000014, "d10"),
       seasonal::series(EM_individual_seas_only$n2000025, "d10"),
       seasonal::series(EM_individual_seas_only$n2000026, "d10"))
colnames(EM_Comp_Sf) <- EM_names

em_plot <- blsplotGG::plot_sf_mean(EM_Comp_Sf,
```

```
  main_title = 'US Employment Seasonal Means',
  sub_title = 'X-11 Seasonals',
  forecast = 60,
  this_legend_title = "SF Means",
  this_legend_entry = c("M 16-19", "F 16-19", "M 20+", "F 20+")
  )

em_plot_facet <- blsplotGG::plot_sf_mean(EM_Comp_Sf,
  main_title = 'US Employment Seasonal Means',
  sub_title = 'X-11 Seasonals',
  forecast = 60,
  do_facet = TRUE,
  this_scale = "fixed",
  this_legend_entry = c("M 16-19", "F 16-19", "M 20+", "F 20+")
  )
```

`plot_sf_series` *Seasonal factor plot grouped by month/quarter*

Description

Generates a special plot of the seasonal factors grouped by month/quarter.

Usage

```
plot_sf_series(
  this_sf = NULL,
  y_limit = NULL,
  this_trans = TRUE,
  main_title = "Seasonal Sub-Plots",
  sub_title = NULL,
  this_xlab = NULL,
  do_grid = FALSE,
  do_background = FALSE,
  this_color_sf = "darkblue",
  this_color_mean = "darkgrey",
  first_year = NULL,
  add_mean_line = TRUE,
  this_legend_title = "SF Plot",
  this_legend_text = c("SF", "SF Mean"),
  legend_title_size = 12,
  legend_text_size = 10
)
```

Arguments

`this_sf`	array of seasonal factors stored as a time series
`y_limit`	Numeric vector of length 2; Range of values you wish the plot to be plotted over. Default is range of the seasonal factors.
`this_trans`	Logical scalar; indicates if the adjustment was done with a log transform. Default is TRUE.

main_title	Character string; main title of plot. Default is 'Seasonal Sub-Plots'.
sub_title	Character string; subtitle of plot. Subtitle not produced if not specified.
this_xlab	Character string; label for x-axis of plot. Default is a blank x-axis.
do_grid	Logical scalar; indicates if plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
this_color_sf	Character string; color used for seasonal factors. Default is "darkblue".
this_color_mean	Character string; color used for means of the seasonal factors. Default is "darkgrey".
first_year	Integer scalar; First year used in plot. Default is start of the series.
add_mean_line	Logical scalar; indicates if seasonal factor plots will include lines for seasonal means. Default includes lines for seasonal means.
this_legend_title	Character string; indicates title of legend. Default is 'Series'.
this_legend_text	Array of character strings; indicates text for each seasonal factor in plot. Default is c("SF", "SF Mean").
legend_title_size	integer scalar; Size of the legend title. Default is 12.
legend_text_size	integer scalar; Size of the legend title. Default is 10.

Details

Version 2.3, 12/11/2024

Value

A ggplot object which generates a plot of the seasonal factors (and the SI-ratios) grouped by month/quarter.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
air_seas    <-
   seasonal::seas(AirPassengers, arima.model = '(0 1 1)(0 1 1)',
                  x11 = '')
air_sf      <- seasonal::series(air_seas, 'd10')
plot_air_sf <- plot_sf_series(air_sf,
   add_mean_line = TRUE,
   main_title = 'Air Passengers Seasonal Sub-Plots',
   this_color_sf = 'darkgreen',
   this_color_mean = 'lightgreen',
   this_legend_title = 'X-11 Seasonal',
   this_legend_text = c('sf', 'mean'))
```

plot_sf_two *Seasonal factor plot grouped by month/quarter for two series*

Description

Generates a plot of the seasonal factors grouped by month/quarter for two adjustments.

Usage

```
plot_sf_two(
  this_sf_one = NULL,
  this_sf_two = NULL,
  y_limit = NULL,
  this_trans = TRUE,
  main_title = "Seasonal Sub-Plots",
  sub_title = NULL,
  this_xlab = NULL,
  do_grid = FALSE,
  do_background = FALSE,
  this_color_sf = NULL,
  this_color_mean = NULL,
  this_palette = "Paired",
  first_year = NULL,
  this_legend_title = "SF Plot",
  this_legend_text = c("SF One", "SF Mean One", "SF Two", "SF Mean Two"),
  legend_title_size = 12,
  legend_text_size = 10
)
```

Arguments

this_sf_one	array of seasonal factors stored as a time series
this_sf_two	array of seasonal factors stored as a time series
y_limit	Numeric vector of length 2; Range of values you wish the plot to be plotted over. Default is range of the seasonal factors.
this_trans	Logical scalar; indicates if the adjustment was done with a log transform. Default is TRUE.
main_title	Character string; main title of plot. Default is 'Seasonal Sub-Plots'.
sub_title	Character string; subtitle of plot. Subtitle not produced if not specified.
this_xlab	Character string; label for x-axis of plot. Default is a blank x-axis.
do_grid	Logical scalar; indicates if plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
this_color_sf	Vector of character strings; colors used for seasonal factors. Should be of length two. Default is NULL, which indicates that the palette specified in this_palette is used to generate colors for this plot.

this_color_mean
: Vector of character strings; color used for means of the seasonal factors. Should be of length two. Default is NULL, which indicates that the palette specified in this_palette is used to generate colors for this plot.

this_palette
: Color used for lines in plot. Default is "Paired"

first_year
: Integer scalar; First year used in plot. Default is start of the series.

this_legend_title
: Character string; indicates title of legend. Default is 'Series'.

this_legend_text
: Array of character strings; indicates text for each seasonal factor in plot. Default is c("SF One", "SF Mean One", "SF Two", "SF Mean Two")).

legend_title_size
: integer scalar; Size of the legend title. Default is 12.

legend_text_size
: integer scalar; Size of the legend title. Default is 10.

Details

Version 1.2, 12/11/2024

Value

A ggplot object which generates a plot of the seasonal factors (and the SI-ratios) grouped by month/quarter.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
air_seas_x11    <-
   seasonal::seas(AirPassengers, arima.model = '(0 1 1)(0 1 1)',
                  x11 = '')
air_sf_x11      <- seasonal::series(air_seas_x11, 'd10')
air_seas_seats  <-
   seasonal::seas(AirPassengers, arima.model = '(0 1 1)(0 1 1)',
                  seats.save = "s10")
air_sf_seats    <- seasonal::series(air_seas_seats, 's10')
plot_air_sf_two <- plot_sf_two(air_sf_x11, air_sf_seats,
   main_title = 'Air Passengers Seasonal Sub-Plots',
   this_legend_text = c('X-11 sf', 'X-11 mean', 'SEATS sf', 'SEATS mean'),
   legend_title_size = 10,
   legend_text_size = 8)
```

plot_sliding_spans *Compare spans from the sliding spans analysis*

Description

Generates a ggplot object with a time series plot that compares the series from each of the sliding spans generated from a sliding spans analysis

Usage

```
plot_sliding_spans(
  seas_obj = NULL,
  this_series = "sfs",
  main_title = "Plot of Sliding Spans",
  sub_title = NULL,
  do_grid = FALSE,
  do_background = FALSE,
  line_color = NULL,
  this_palette = "Dark2",
  this_guide_legend = "Sliding Spans"
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This is a required entry.
this_series	Character string; three character code for the type of series to be generated. Allowed entries are "sfs" (seasonal factors, default), "chs" (period-to-period changes), "sis" (indirect seasonal factors), "cis" (indirect period-to-period changes).
main_title	Title for the plot. Default is character string 'Plot of Sliding Spans'.
sub_title	Subtitle for the plot. Optional entry.
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
line_color	Character vector of length 2 to 4 based on the number of spans. Default is generated from the RColorBrewer palette "Dark2".
this_palette	Character string; default RColorBrewer palette. Deault is "Dark2".
this_guide_legend	Title for legend. Default is "Sliding Spans"

Details

Version 1.7, 11/7/2024

Value

A ggplot object that generates a plot compares the series from each of the sliding spans generated from a sliding spans analysis

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
shoes_sspan_seas <-
   seasonal::seas(blsplotGG::shoes2008,
                  arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log",
                  forecast.maxlead = 36,
                  slidingspans.save = c('sfs', 'chs'))
p_shoes_spans_sf <-
    plot_sliding_spans(shoes_sspan_seas, "sfs",
                       main_title = "Spans of Seasonal Factors",
                       sub_title = "US Shoe Sales",
                       this_palette = "Set1")
```

---

plot_sliding_spans_two

Maximum percent difference plot for two sliding spans runs

Description

Generates a ggplot object comparing the time series of the maximum percent difference from the sliding spans analysis of seasonal factors or changes for two sliding spans runs. The sliding spans analysis for both series should have the same length of spans.

Usage

```
plot_sliding_spans_two(
  seas_obj_one = NULL,
  seas_obj_two = NULL,
  this_series = "sfs",
  main_title = "Maximum Percent Difference Comparison Plot",
  sub_title = NULL,
  this_x_label = "Time",
  this_y_label = " ",
  do_grid = FALSE,
  do_background = FALSE,
  line_color = "grey",
  cut_color = "red",
  this_legend_title = "Sliding Spans",
  this_legend_entry = c("ss1", "ss2"),
  legend_title_size = 12,
  legend_text_size = 10
)
```

Arguments

seas_obj_one	seas object generated from a call of seas on a single time series This is a required entry.

seas_obj_two	seas object generated from a call of seas on a single time series This is a required entry.
this_series	Character string; three character code for the type of series to be generated. Allowed entries are "sfs" (seasonal factors, default), "chs" (period-to-period changes), "sis" (indirect seasonal factors), "cis" (indirect period-to-period changes).
main_title	Title for the plot. Default is character string 'Maximum Percent Difference Comparison Plot'.
sub_title	Subtitle for the plot. Optional entry.
this_x_label	Label for X-axis. Default is "Time"
this_y_label	Label for Y-axis. Default is " "
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
line_color	Color used for lines in the maximum percentage difference plot. Default is "steelblue".
cut_color	Color use to show the slidings spans cut off for this type of series. Default is "red".
this_legend_title	Character string; indicates title of legend. Default is 'Sliding Spans'.
this_legend_entry	Character array; entries for the legend. Default is c("ss1", "ss2").
legend_title_size	integer scalar; Size of the legend title. Default is 12.
legend_text_size	integer scalar; Size of the legend title. Default is 10.

Details

Version 1.5, 12/11/2024

Value

A ggplot object that generates a plot of the maximum percent difference from a sliding spans analysis of seasonal factors or changes.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
shoes_sspan_seas <-
   seasonal::seas(blsplotGG::shoes2008,
                  arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log",
                  forecast.maxlead = 36,
                  slidingspans.save = c('sfs', 'chs'))

shoes_sspan_x11_seas <-
```

```
    seasonal::seas(blsplotGG::shoes2008,
                   arima.model = "(0 1 1)(0 1 1)",
                   transform.function = "log",
                   forecast.maxlead = 36,
                   x11.seasonalma = "s3x3",
                   slidingspans.length = 76,
                   slidingspans.numspans = 4,
                   slidingspans.print = 'sfs',
                   slidingspans.save = c('sfs', 'chs'))
p_shoes_maxpct_sf <-
    plot_sliding_spans_two(shoes_sspan_seas, shoes_sspan_x11_seas, "sfs",
     main_title = "Maximum Difference of the Seasonal Factors",
         sub_title = "US Retail Shoe Sales",
         line_color = "black",
         cut_color = "purple",
         this_legend_entry = c("Seats", "X-11"))
```

`plot_spectrum` *Generate spectrum plot of either the original, seasonally adjusted, irregular, or model residuals.*

Description

Generate plot of spectrum from X-13ARIMA-SEATS specified by the user.

Usage

```
plot_spectrum(
  seas_obj = NULL,
  this_spectrum = "sp0",
  xaxis_bls = TRUE,
  main_title = "AR Spectrum",
  sub_title = NULL,
  series_name = NULL,
  do_grid = FALSE,
  do_background = FALSE,
  this_color = "darkblue",
  this_median_color = "blue",
  this_freq_color = c("steelblue", "forestgreen"),
  this_peak_color = c("violet", "brown")
)
```

Arguments

`seas_obj`	`seas` object generated from a call of `seas` on a single time series This is a required entry.
`this_spectrum`	Character string; three character code for the X-13 spectrum to be generated. Allowed entries are `"sp0"` (modified original series), `"sp1"` (modified X-11 seasonally adjusted series), `"sp2"` (modified X-11 irregular), `"s1s"` (modified SEATS seasonally adjusted series), `"s2s"` (modified SEATS irregular), `"is0"` (modified composite series), `"is1"` (modified indirect seasonally adjusted series), `"is2"` (modified indirect irregular), `spr` (model residuals), or `"ser"` (extended residuals). Default: `"sp0"`.

xaxis_bls Logical scalar; indicates if x-axis of spectral plot will be frequency by month rather than the actual frequencies. Default sets x-axis to frequency by month.

main_title Character string; main title of plot. Default is 'AR Spectrum'.

sub_title Character scalar; Description of time series used in seas_obj. Used as the subtitle of the plot if specified.

series_name Character scalar; name of the time series used in seas_obj. Used as the label of the Y-axis if specified.

do_grid Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.

do_background Logical scalar; indicates grey background included in plot. Default is no grey background;

this_color Character string. Colors used for spectrum in plot. Default is "darkblue".

this_median_color
Character string. Colors used for medians of the spectrum. Default is "blue".

this_freq_color
Character vector of length 2. Colors used for seasonal and trading day frequencies, respectively. Defaults are c("steelblue", "forestgreen").

this_peak_color
Character vector of length 2. Colors used for peaks at seasonal and trading day frequencies, respectively. Defaults are c("violet", "brown").

Details

Version 2.8, 11/6/2024

Value

A ggplot object which generates a spectrum plot generated by X-13ARIMA-SEATS.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
air_seas <- seasonal::seas(AirPassengers, arima.model = "(0 1 1)(0 1 1)", x11="",
                           spectrum.save = "sp2")
plot_air_spectrum <-
       plot_spectrum(air_seas, this_spectrum = "sp2",
                     series_name = "AirPassengers",
                     sub_title = "Airline Passengers",
                     this_color = "steelblue", this_median_color = "blue",
                     this_freq_color = c("darkblue", "darkgreen"),
                     this_peak_color = c("red", "orange"))
```

plot_squared_gain *Plot of the squared gains for filters generated by an X-13ARIMA-SEATS SEATS seasonal adjustment run.*

Description

Generate squared gains plot of the concurrent and symmetric SEATS seasonal adjustment and trend filters from a SEATS adjustment from a seas object generated by the seasonal package.

Usage

```
plot_squared_gain(
  seas_obj = NULL,
  this_series = "sa",
  main_title = NULL,
  sub_title = NULL,
  this_y_label = "Squared Gain",
  this_x_label = "Cycles per Year",
  do_grid = FALSE,
  do_background = FALSE,
  line_color = NULL,
  this_palette = "Paired",
  this_guide_legend = "Filter"
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This is a required entry.
this_series	Character string; series for which SEATS produces a squared gain plot, limited to the seasonally adjusted series (sa, the default), or the trend component (trend). For other entries, the function will print an error message and return a NULL.
main_title	Character string; main title of plot. A title will be generated if no title is specified.
sub_title	Character string; subtitle of plot. There is no default subtitle.
this_y_label	Character string; y-axis label for plot. Default is "Squared Gain".
this_x_label	Label for X axis. Default is "Cycles per Year".
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
line_color	Character vector of length two; colors used for the squared gain #' in the plot. Default is NULL, which indicates that the palette specified in this_palette is used to generate colors for this plot.
this_palette	Color used for lines in plot. Default is "Paired"
this_guide_legend	Title for legend. Default is "Filter"

Details

Version 1.7, 11/6/2024

Value

A ggplot object which generates a plot of the squared gains for filters generated by the SEATS seasonal factors. If SEATS seaonal adjustment (with finite = yes) not producted, print out error message and return NULL.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
shoes_seats_seas <-
   seasonal::seas(shoes2008, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log",
                  forecast.maxlead = 36,
                  check.print = c( 'pacf', 'pacfplot' ),
                  seats.finite = "yes",
                  seats.save = c( 'gac', 'gaf', 'gtc', 'gtf' ) )
p_sa_squared_gain <- plot_squared_gain(shoes_seats_seas,
sub_title = "US Shoe Sales",
this_palette = "Set2")
```

---

plot_table *Plot table from X-13ARIMA-SEATS seasonal adjustment run.*

Description

Generate plot of user-specified series from a seas object generated by the seasonal package.

Usage

```
plot_table(
  seas_obj = NULL,
  this_table = NULL,
  main_title = NULL,
  sub_title = NULL,
  this_y_label = NULL,
  y_limit = NULL,
  this_x_label = "Time",
  start_plot = NULL,
  do_grid = FALSE,
  do_background = FALSE,
  draw_recess = FALSE,
  recess_color = "lightgrey",
  recess_sub = TRUE,
  add_outlier = FALSE,
  use_ratio = FALSE,
  ratio_mean = 1,
```

```
  this_line_type = "solid",
  line_color = "grey",
  outlier_color = c("red", "blue", "orangered", "green", "steelblue", "blue"),
 outlier_line_type = c("dashed", "dotdash", "dashed", "twodash", "dotdash", "dotdash")
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This is a required entry.
this_table	Character string; X-13ARIMA-SEATS table name or abbreviation. If not a valid table name, the function will print an error message and return a NULL.
main_title	Character string; main title of plot. A title will be generated if no title is specified.
sub_title	Character string; subtitle of plot. There is no default subtitle.
this_y_label	Character string; y-axis label for plot, if specified.
y_limit	Numeric vector of length 2; Range of values for the y-axis. Default is range of the series specified.
this_x_label	Label for X axis. Default is "Time".
start_plot	Integer vector of length 2; Starting date for plot. Default is starting date for the time series.
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
draw_recess	Logical scalar; indicates if certain plots will have shaded areas for NBER recession dates. Default is no recession shading.
recess_color	Character string; color used for shading of recession region. Default is 'lightgrey'.
recess_sub	Logical scalar; indicates if x-axis label for recession is produced for this plot. Default is x-axis label.
add_outlier	Logical scalar; indicates if lines for identified outliers are included in series plots. Default is not including lines for identified outliers.
use_ratio	Logical scalar; indicates if plots of seasonal factors, irregular, and residuals are done as ratio plots. Default has these plots as time series line plots.
ratio_mean	Assumed mean value for the ratio. Default is 1.0
this_line_type	Character string; indicates line type of each plot produced. Default is "solid".
line_color	Character string; color used for series in the plot. Default is 'grey'.
outlier_color	Character array of length 6; color used for different outliers, with the order being 'ao', 'ls', 'tc', 'so', 'rp', 'tls'. Default is c("red", "blue", 'orangered', "green", "steelblue", "blue").
outlier_line_type	Character array of length 6; Line type used for different outliers, with the order being 'ao', 'ls', 'tc', 'so', 'rp', 'tls'. Default is c('dashed', 'dotdash', 'dashed', 'twodash', 'dotdash', 'dotdash').

Details

Version 2.8, 11/6/2024

Value

A ggplot object that generates a plot of user-specified series from an X-13ARIMA-SEATS table. If series not specified, print out error message and return NULL.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
air_seas <-
  seasonal::seas(AirPassengers, arima.model = "(0 1 1)(0 1 1)", x11="",
                 series.save = 'b1', transform.function = "log",
                 x11.save = "e3")
air_d11_p <- blsplotGG::plot_table(air_seas, "d11",
         this_y_label = "AirPassengers",
         main_title = "X-11 Seasonal Adjustment of Airline Passengers",
         sub_title = "Box-Jenkins Airline series",
         do_grid = TRUE, draw_recess = TRUE,
         use_ratio = FALSE, add_outlier = TRUE, line_color = "darkblue")
air_d16_p <- blsplotGG::plot_table(air_seas, "d16",
         this_y_label = "AirPassengers",
         main_title = "X-11 Seasonal Adjustment of Airline Passengers",
         do_grid = FALSE, draw_recess = TRUE,
         use_ratio = TRUE, add_outlier = TRUE, line_color = "steelblue")
air_e3_p <- blsplotGG::plot_table(air_seas, "e3",
         this_y_label = "AirPassengers",
        main_title = "X-11 Seasonal Adjustment (Extreme Adjusted) of Airline Passengers",
         do_grid = FALSE, draw_recess = TRUE,
         use_ratio = FALSE, add_outlier = TRUE, line_color = "steelblue")
```

---

plot_time_shift *Plot of the squared gains for filters generated by an X-13ARIMA-SEATS SEATS seasonal adjustment run.*

Description

Generate squared gains plot of the concurrent and symmetric SEATS seasonal adjustment and trend filters from a SEATS adjustment from a seas object generated by the seasonal package.

Usage

```
plot_time_shift(
  seas_obj = NULL,
  main_title = NULL,
  sub_title = NULL,
  this_y_label = "Time Shift",
  this_x_label = "Cycles per Year",
  do_grid = FALSE,
  do_background = FALSE,
  line_color = NULL,
  this_palette = "Paired",
  this_guide_legend = "Filter"
)
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This is a required entry.
main_title	Character string; main title of plot. A title will be generated if no title is specified.
sub_title	Character string; subtitle of plot. There is no default subtitle.
this_y_label	Character string; y-axis label for plot. Default is "Time Shift".
this_x_label	Label for X axis. Default is "Cycles per Year".
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
line_color	Character vector of length two; colors used for the squared gain #' in the plot. Default is NULL, which indicates that the palette specified in this_palette is used to generate colors for this plot.
this_palette	Color used for lines in plot. Default is "Paired"
this_guide_legend	Title for legend. Default is "Filter"

Details

Version 1.10, 11/6/2024

Value

A ggplot object which generates a plot of the squared gains for filters generated by the SEATS seasonal factors. If SEATS seaonal adjustment (with finite = yes) not producted, print out error message and return NULL.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
shoes_time_shift <-
   seasonal::seas(shoes2008, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log",
                  forecast.maxlead = 36,
                  check.print = c( 'pacf', 'pacfplot' ),
                  seats.finite = "yes",
                  seats.save = c( 'tac', 'ttc' ) )
p_time_shift <- plot_time_shift(shoes_time_shift,
sub_title = "US Shoe Sales",
this_palette = "Set2")
```

plot_two_sa *Compare two seasonal adjustments*

Description

Generates a ggplot object with a time series plot that compares two seasonal adjustments of the same series, optionally including the original series.

Usage

```
plot_two_sa(
  this_sa_one = NULL,
  this_sa_two = NULL,
  this_ori = NULL,
  main_title = "Compare X-11 and SEATS",
  sub_title = NULL,
  this_x_label = "Time",
  this_y_label = " ",
  this_sa_text = c("X-11", "SEATS"),
  do_grid = FALSE,
  do_background = FALSE,
  line_color = NULL,
  this_palette = "Dark2",
  this_guide_legend = "Series"
)
```

Arguments

this_sa_one	Time series of the X-11 seasonal adjustment. This is a required entry.
this_sa_two	Time series of the SEATS seasonal adjustment. This is a required entry.
this_ori	Time series of the original series. Optional entry.
main_title	Title for the plot. Default is character string 'Comparison of Seasonal Adjustments'.
sub_title	Subtitle for the plot. Optional entry.
this_x_label	Label for X-axis. Default is "Time"
this_y_label	Label for Y-axis. Default is " "
this_sa_text	Labels for different seasonal adjustments. Default is c('X-11', 'SEATS')
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
line_color	Character scalar; color used for plot. User should specify colors for each series in this_sa_text. Default is the RColorBrewer palette "Dark2".
this_palette	Character string; default RColorBrewer palette. Deault is "Dark2".
this_guide_legend	Title for legend. Default is "Series"

Details

Version 2.5, 12/19/2024

Value

A ggplot object that generates a plot comparing two seasonal adjustments, trend, or factors.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
ukgas_x11_seas   <-
   seasonal::seas(UKgas, series.period = 4, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 20,
                  x11.seasonalma = "s3x5",
                  check.print = c( 'pacf', 'pacfplot' ))
ukgas_seats_seas <-
   seasonal::seas(UKgas, series.period = 4, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 20,
                  check.print = c( 'pacf', 'pacfplot' ))
ukgas_x11_sa     <- seasonal::final(ukgas_x11_seas)
ukgas_seats_sa   <- seasonal::final(ukgas_seats_seas)
ukgas_seats_v_x11_p <-
    plot_two_sa(this_sa_one = ukgas_x11_sa, this_sa_two = ukgas_seats_sa,
                       main_title = "UK Gas Seasonal Adjustments",
                       sub_title = "X-11 - Blue, SEATS - Green",
                       line_color = c("steelblue", "forestgreen"))
```

---

plot_two_sa_facet *Compare two seasonal adjustments in a facet plot*

Description

Generates a ggplot object with a time series facet plot that compares two seasonal adjustments of the same series, optionally including the original series.

Usage

```
plot_two_sa_facet(
  this_sa_one = NULL,
  this_sa_two = NULL,
  this_ori = NULL,
  main_title = "Compare X-11 and SEATS",
  sub_title = NULL,
  this_x_label = "Time",
  this_y_label = " ",
  line_color = "steelblue",
  line_label = c("sadj1", "sadj2"),
  this_scale = "fixed",
  remove_legend = FALSE
)
```

Arguments

this_sa_one	Time series of the first seasonal adjustment. This is a required entry.
this_sa_two	Time series of the second seasonal adjustment. This is a required entry.
this_ori	Time series of the original series. Optional entry.
main_title	Title for the plot. Default is character string 'Comparison of Seasonal Adjustments'.
sub_title	Subtitle for the plot. Optional entry.
this_x_label	Label for X-axis. Default is "Time"
this_y_label	Label for Y-axis. Default is " "
line_color	Color used for lines in plot. Default is "steelblue".
line_label	Caracter vector of length 2; labels used for lines in the plot, Default is c("sadj1", "sadj2").
this_scale	Character scalar; set the scale for the y-axis of the facet plots. Default is "fixed". To set axis to be different for the different plots, set this_scale = "free_y".
remove_legend	Logical scalar; if TRUE, plot legend will be removed. Default is FALSE.

Details

Version 4.0, 1/7/2025

Value

A ggplot object that generates a facet plot comparing two seasonal adjustments, trends, or factors.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
ukgas_x11_seas   <-
   seasonal::seas(UKgas, series.period = 4, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 20,
                  x11.seasonalma = "s3x5",
                  check.print = c( 'pacf', 'pacfplot' ))
ukgas_seats_seas <-
   seasonal::seas(UKgas, series.period = 4, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 20,
                  check.print = c( 'pacf', 'pacfplot' ))
ukgas_x11_sa     <- seasonal::final(ukgas_x11_seas)
ukgas_seats_sa   <- seasonal::final(ukgas_seats_seas)
ukgas_two_sa_facet_p <-
    plot_two_sa_facet(this_sa_one = ukgas_x11_sa,
                      this_sa_two = ukgas_seats_sa,
                      main_title = "UK Gas Seasonal Adjustments",
                      line_color = "forestgreen",
                      line_label = c("X-11", "Seats"))
```

plot_x11_and_seats *Compare X-11 and SEATS seasonal adjustment*

Description

Generates a ggplot object with a time series plot that compares an X-11 and SEATS seasonal adjustment, optionally including the original series.

Usage

```
plot_x11_and_seats(
  this_x11 = NULL,
  this_seats = NULL,
  this_ori = NULL,
  main_title = "Compare X-11 and SEATS",
  sub_title = NULL,
  this_x_label = "Time",
  this_y_label = " ",
  do_grid = FALSE,
  do_background = FALSE,
  line_color = NULL,
  this_palette = "Dark2",
  this_guide_legend = "Series"
)
```

Arguments

this_x11	Time series of the X-11 seasonal adjustment. This is a required entry.
this_seats	Time series of the SEATS seasonal adjustment. This is a required entry.
this_ori	Time series of the original series. Optional entry.
main_title	Title for the plot. Default is character string 'Comparison of X-11 and SEATS Seasonal Adjustments'.
sub_title	Subtitle for the plot. Optional entry.
this_x_label	Label for X-axis. Default is "Time"
this_y_label	Label for Y-axis. Default is " "
do_grid	Logical scalar; indicates if certain plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background;
line_color	Character vector of length 2 (if this_ori is not specified) or 3 (if plot_ori is specified); color used for lines in the plot, in the order of seasonally adjusted series, trend, original series. Default is generated from the RColorBrewer palette "Dark2".
this_palette	Character string; default RColorBrewer palette. Deault is "Dark2".
this_guide_legend	Title for legend. Default is "Series"

Details

Version 4.3, 11/6/2024

Value

A ggplot object that generates a plot comparing an X-11 and SEATS seasonal adjustment, trend, or factors.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
ukgas_x11_seas   <-
   seasonal::seas(UKgas, series.period = 4, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 20,
                  x11.seasonalma = "s3x5",
                  check.print = c( 'pacf', 'pacfplot' ))
ukgas_seats_seas <-
   seasonal::seas(UKgas, series.period = 4, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 20,
                  check.print = c( 'pacf', 'pacfplot' ))
ukgas_x11_sa     <- seasonal::final(ukgas_x11_seas)
ukgas_seats_sa   <- seasonal::final(ukgas_seats_seas)
ukgas_seats_v_x11_p <-
    plot_x11_and_seats(this_x11 = ukgas_x11_sa, this_seats = ukgas_seats_sa,
                       main_title = "UK Gas Seasonal Adjustments",
                       sub_title = "X-11 - Blue, SEATS - Green",
                       line_color = c("steelblue", "forestgreen"))
```

---

plot_x11_and_seats_facet

Compare X-11 and SEATS seasonal adjustments in a facet plot

Description

Generates a ggplot object with a time series facet plot that compares an X-11 and SEATS seasonal adjustment, optionally including the original series.

Usage

```
plot_x11_and_seats_facet(
  this_x11 = NULL,
  this_seats = NULL,
  this_ori = NULL,
  main_title = "Compare X-11 and SEATS",
  sub_title = NULL,
  this_x_label = "Time",
  this_y_label = " ",
  line_color = "steelblue",
  this_scale = "fixed",
  remove_legend = FALSE
)
```

Arguments

this_x11	Time series of the X-11 seasonal adjustment. This is a required entry.
this_seats	Time series of the SEATS seasonal adjustment. This is a required entry.
this_ori	Time series of the original series. Optional entry.
main_title	Title for the plot. Default is character string 'Comparison of X-11 and SEATS Seasonal Adjustments'.
sub_title	Subtitle for the plot. Optional entry.
this_x_label	Label for X-axis. Default is "Time"
this_y_label	Label for Y-axis. Default is " "
line_color	Color used for lines in plot. Default is "steelblue".
this_scale	Character scalar; set the scale for the y-axis of the facet plots. Default is "fixed". To set axis to be different for the different plots, set this_scale = "free_y".
remove_legend	Logical scalar; if TRUE, plot legend will be removed. Default is FALSE.

Details

Version 7.1, 1/7/2025

Value

A ggplot object that generates a facet plot comparing an X-11 and SEATS seasonal adjustment, trend, or factor.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
ukgas_x11_seas   <-
   seasonal::seas(UKgas, series.period = 4, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 20,
                  x11.seasonalma = "s3x5",
                  check.print = c( 'pacf', 'pacfplot' ))
ukgas_seats_seas <-
   seasonal::seas(UKgas, series.period = 4, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 20,
                  check.print = c( 'pacf', 'pacfplot' ))
ukgas_x11_sa     <- seasonal::final(ukgas_x11_seas)
ukgas_seats_sa   <- seasonal::final(ukgas_seats_seas)
ukgas_seats_v_x11_facet_p <-
    plot_x11_and_seats_facet(this_x11 = ukgas_x11_sa, this_seats = ukgas_seats_sa,
                       main_title = "UK Gas Seasonal Adjustments",
                       line_color = "forestgreen")
```

plot_year_over_year *Plot year over year plot*

Description

Generate year over year plot of a user-specified ts object.

Usage

```
plot_year_over_year(
  this_series = NULL,
  main_title = NULL,
  sub_title = NULL,
  this_y_label = NULL,
  y_limit = NULL,
  this_x_label = NULL,
  start_plot = NULL,
  do_grid = FALSE,
  do_background = FALSE,
  line_color = NULL,
  this_palette = "Paired",
  detrend_series = FALSE,
  detrend_lowess = FALSE
)
```

Arguments

this_series	Numeric matrix; columns of time series object to be plotted.
main_title	Character string; main title of plot. The default title is the name of the series passed to this function.
sub_title	Character string; subtitle of plot. There is no default subtitle.
this_y_label	Character string; y-axis label for plot, if specified.
y_limit	Numeric vector of length 2; Range of values you wish the plot to be plotted over. Default is range of the series specified.
this_x_label	Label for X axis. Default is "Month" or "Quarter".
start_plot	Integer vector of length 2; Starting date for plot. Default is starting date for the time series.
do_grid	Logical scalar; indicates if plots will have grid lines. Default is no grid lines.
do_background	Logical scalar; indicates grey background included in plot. Default is no grey background.
line_color	Character scalar; color used for plot. User should specify one color for each column of the matrix specified. Default is the RColorBrewer palette "Paired".
this_palette	Character string; default RColorBrewer palette. Default is "Paired".
detrend_series	Logical scalar; indicates if the series plotted is to be detrended. Default is the original series is plotted.
detrend_lowess	Logical scalar; indicates lowess is used to generate the trend used to detrend the series. Default is loess is not used.

Details

Version 3.2, 11/6/2024

Value

Generate ggplot object generating a year to year plot of a time series object. If time series object not specified, print out error message and return NULL.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
this_yyplot <-
    plot_year_over_year(AirPassengers, this_y_label = "Air", this_palette = "Dark2")
this_yyplot_detrend <-
    plot_year_over_year(AirPassengers, this_y_label = "Air", this_palette = "Dark2",
                        detrend_series = TRUE, detrend_lowess = TRUE)
```

---

proc_outlier *Extract dates from outlier text*

Description

Process name of outlier regressor to extract the dates associated with the outlier

Usage

```
proc_outlier(this_outlier = NULL, this_freq = 12, add_type = TRUE)
```

Arguments

this_outlier	Character string; outlier regressor. This is a required entry.
this_freq	integer scalar; time series frequency. Default is 12.
add_type	logical scalar; determines if type of outlier is added to the output. Default is TRUE.

Details

Version 2.1, 5/2/2024

Value

List of either year and month/quarter of outlier, or year and month/quarter of start and end of outlier

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
air_seas <-
   seasonal::seas(AirPassengers, x11="", slidingspans = "",
                     transform.function = "log", arima.model = "(0 1 1)(0 1 1)",
                     regression.aictest = 'td', forecast.maxlead=60,
                     check.print = c( "pacf", "pacfplot" ))
this_auto_outlier <- get_auto_outlier_string(air_seas)
this_outlier      <- proc_outlier(this_auto_outlier)
```

---

seasonal_subplot *Seasonal sub-plot*

Description

Generates a seasonal sub-plot from a ts object of seasonal (or combined adjustment) factors

Usage

```
seasonal_subplot(
  this_sf = NULL,
  this_sf_range = NULL,
  main_title = "Seasonal Subplot",
  sub_title = NULL,
  this_x_label = "Month",
  this_y_label = "",
  subplot_color = "steelblue"
)
```

Arguments

this_sf	Time series of seasonal factors from X-11 or SEATS
this_sf_range	Range of values you wish the plot to be plotted over. Default is range of the series.
main_title	Title for the plot. Default is character string 'Ratio Plot'.
sub_title	Subtitle for the plot. Default is NULL.
this_x_label	Label for X axis. Default is "Time".
this_y_label	Label for Y axis. Default is "Ratio".
subplot_color	Color used for lines in ratio plot. Default is "steelblue".

Details

Version 2.0, 5/6/2024

Value

A ggplot object that generates a ratio plot.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
ukgas_x11_seas <-
   seasonal::seas(UKgas, series.period = 4, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 20,
                  x11.seasonalma = "s3x5",
                  check.print = c( 'pacf', 'pacfplot' ))
ukgas_x11_sf     <- seasonal::series(ukgas_x11_seas, "d10")
p_ukgas_sf_sub   <-
    seasonal_subplot(ukgas_x11_sf,
                     main_title = "UK Gas X-11 Seasonal Factors",
                     sub_title = "Seasonal Filter: 3x5",
                     this_x_label = "Quarter")
```

`seasonal_subplot_two` *Seasonal sub-plot for two sets of seasonal factors*

Description

Generates a seasonal sub-plot from two ts objects of seasonal (or combined adjustment) factors

Usage

```
seasonal_subplot_two(
  this_sf = NULL,
  this_sf_two = NULL,
  this_sf_range = NULL,
  main_title = "Seasonal Subplot",
  sub_title = NULL,
  this_x_label = NULL,
  this_y_label = NULL,
  this_sf_label = NULL,
  this_sf_label_two = NULL,
  subplot_color = c("blue", "lightblue", "red", "pink")
)
```

Arguments

`this_sf`	Time series of seasonal factors from X-11 or SEATS
`this_sf_two`	Time series of seasonal factors from X-11 or SEATS
`this_sf_range`	Range of values you wish the plot to be plotted over. Default is range of the series.
`main_title`	Title for the plot. Default is character string `'Ratio Plot'`.
`sub_title`	Subtitle for the plot. Default is `NULL`.
`this_x_label`	Label for X axis. Default is `NULL`.
`this_y_label`	Label for Y axis. Default is `NULL`.
`this_sf_label`	Character scalar, provides a brief description of the first seasonal factors. Default is `NULL`.
`this_sf_label_two`	Character scalar, provides a brief description of the second seasonal factors. Default is `NULL`.

subplot_color Character vector of length four, setting color used for lines in ratio plot in the order of first factor, first factor mean, second factor, second factor mean. Default is c("blue", "lightblue", "red", "pink").

Details

Version 1.8, 9/25/2024

Value

A ggplot object that generates a ratio plot.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
ukgas_x11_seas <-
   seasonal::seas(UKgas, series.period = 4, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 20,
                  x11.seasonalma = "s3x5",
                  check.print = c( 'pacf', 'pacfplot' ))
ukgas_seats_seas <-
   seasonal::seas(UKgas, series.period = 4, arima.model = "(0 1 1)(0 1 1)",
                  transform.function = "log", forecast.maxlead = 20,
                  seats.finite = "yes", seats.save = "s10",
                  check.print = c( 'pacf', 'pacfplot' ))
ukgas_x11_sf     <- seasonal::series(ukgas_x11_seas, "d10")
ukgas_seats_sf   <- seasonal::series(ukgas_seats_seas, "s10")
p_ukgas_sf_sub   <-
    seasonal_subplot_two(ukgas_x11_sf, ukgas_seats_sf,
                     main_title = "UK Gas Seasonal Subplots",
                     sub_title = "X-11 and SEATS Seasonal Factors",
                     this_x_label = "Quarter",
                     this_sf_label = "X-11",
                     this_sf_label_two = "SEATS")
```

---

shoes2008 *Retail sales of shoes, 2008*

Description

A time series object containing retail sales of shoes

Usage

```
shoes2008
```

Format

Retail sales of shoes ending in April of 2008

update_vector *Update vector.*

Description

Fill unspecified elements of a vector with the first element of the input series

Usage

```
update_vector(this_series = NULL, this_num = NULL)
```

Arguments

this_series	Original time series. This is a required entry.
this_num	Length of updated series. Must be more than the length of this_series. This is a required entry.

Details

Version 2.3, 5/25/2023

Value

An updated vector of length this_num augmented with the first value of the input series.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
this_vector <- c(1,2)
updated_vector <- update_vector(this_vector, 4)
```

visual_sig_peaks *Flag visual significant peaks in spectra*

Description

Determine positions of visual significant peaks in spectra

Usage

```
visual_sig_peaks(seas_obj = NULL, spec_type = "sa", spec_freq_code = "seas")
```

Arguments

seas_obj	seas object generated from a call of seas on a single time series This entry is required.
spec_type	Character string; type of spectrum. Possible values are 'ori', 'irr', 'rsd', 'sa', 'comp', 'indsa', 'indirr', 'extrsd'. Default is 'sa'.
spec_freq_code	Character string; type of frequency being tested. Possible values are 'seas' or 'td'. Default is 'seas'.

Details

Version 3.4, 5/14/2024

Value

If visually significant peaks found, a numveric vector of the position of the peak frequecies. If no peaks found, 0.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
air_seas <- seasonal::seas(AirPassengers, arima.model = '(0 1 1)(0 1 1)', x11='')
vp_ori_seas <- visual_sig_peaks(air_seas, spec_type = 'ori')
vp_ori_td   <- visual_sig_peaks(air_seas, spec_type = 'sa',
                                spec_freq_code = 'td')
```

---

what_spectrum *What spectrum is plotted*

Description

Return which spectrum plot is generated based on the three character code used by the plot_spectrum function.

Usage

```
what_spectrum(this_spectrum = "sp0", use_title_case = FALSE)
```

Arguments

this_spectrum	Character string; three character code for the X-13 spectrum to be generated. Allowed entries are "sp0" (modified original series), "sp1" (modified X-11 seasonally adjusted series), "sp2" (modified X-11 irregular), "s1s" (modified SEATS seasonally adjusted series), "s2s" (modified SEATS irregular), "is0" (modified composite series), "is1" (modified indirect seasonally adjusted series), "is2" (modified indirect irregular), spr (model residuals), or "ser" (extended residuals). Default: "sp0".
use_title_case	Logical scalar; convert string to title case. Default is FALSE.

Details

Version 1.5, 5/6/2024

Value

Text for spectrum associated with code used in `plot_spectrum` function. If improper value set for `this_spectrum`, function will return `NULL`.

Author(s)

Brian C. Monsell, <monsell.brian@bls.gov> or <bcmonsell@gmail.com>

Examples

```
sub_title <- what_spectrum("sp2", use_title_case = TRUE)
```

---

xt_data_list *US Building Permits, One Family Units*

Description

A list object of US One family Building Permits for four regions expressed as time series objects that end in October, 2006

Usage

```
xt_data_list
```

Format

A list object with 4 time series elements:

mw1u Midwest one family building permits

ne1u Northeast one family building permits

so1u South one family building permits

we1u West one family building permits

Index