

Package ‘brfinance’

December 21, 2025

Title Simplified Access to Brazilian Financial and Macroeconomic Data

Version 0.6.0

Description It offers simplified access to Brazilian macroeconomic and financial indicators selected from official sources, such as the 'IBGE' (Brazilian Institute of Geography and Statistics) via the 'SIDRA' API and the 'Central Bank of Brazil' via the 'SGS' API. It allows users to quickly retrieve and visualize data series such as the unemployment rate and the Selic interest rate. This package was developed for data access and visualization purposes, without generating forecasts or statistical results. For more information, see the official APIs: <<https://sidra.ibge.gov.br/>> and <<https://dadosabertos.bcb.gov.br/dataset/>>.

License MIT + file LICENSE

URL <https://github.com/efram2/brfinance>

BugReports <https://github.com/efram2/brfinance/issues>

Encoding UTF-8

RoxygenNote 7.3.2

Depends R (>= 4.1.0)

Imports dplyr, ggplot2, scales, httr2, lubridate, sidrar, stringr, janitor, labelled

Suggests testthat (>= 3.0.0), rmarkdown, knitr, tidyr

VignetteBuilder knitr

NeedsCompilation no

Author João Paulo dos Santos Pereira Barbosa [aut, cre]

Maintainer João Paulo dos Santos Pereira Barbosa <joao.31582129@gmail.com>

Repository CRAN

Date/Publication 2025-12-21 22:50:07 UTC

Contents

get_cdi_rate	2
get_exchange_rate	3

get_gdp_growth	5
get_inflation_rate	6
get_selic_rate	8
get_unemployment	10
plot_cdi_rate	12
plot_exchange_rate	13
plot_inflation_rate	13
plot_selic_rate	14
plot_series_comparison	15
plot_unemployment	17

Index	18
--------------	-----------

get_cdi_rate	<i>Get CDI Rate (Interbank Deposit Certificate)</i>
--------------	---

Description

Downloads daily CDI (Certificado de Depósito Interbancário) rate from BCB/SGS. This function retrieves the daily CDI rate (SGS series 12), which is the benchmark interest rate for interbank transactions in Brazil.

Usage

```
get_cdi_rate(  
  start_date = NULL,  
  end_date = NULL,  
  language = "eng",  
  labels = TRUE  
)
```

Arguments

start_date	Start date for the data period. Accepts multiple formats: • "YYYY" for year only (e.g., "2020" becomes "2020-01-01") • "YYYY-MM" for year and month (e.g., "2020-06" becomes "2020-06-01") • "YYYY-MM-DD" for a specific date (e.g., "2020-06-15")
end_date	End date for the data period. Accepts the same formats as start_date: • "YYYY" (e.g., "2023" becomes "2023-12-31") • "YYYY-MM" (e.g., "2023-12" becomes the last day of December 2023) • "YYYY-MM-DD" for a specific date • NULL defaults to the current date (today)
language	Language for column names in the returned data.frame: • "eng" (default): Returns columns date and cdi_rate • "pt": Returns columns data_referencia and taxa_cdi
labels	Logical indicating whether to add variable labels using the labelled package. Labels provide descriptive text for each column when available.

Value

A data.frame with columns:

date Reference date

value Daily CDI rate (% per day)

value_annualized Annualized CDI rate (% per year, 252 business days)

Note

Series information: This function uses SGS series 12, which represents the daily CDI rate (taxa de juros - CDI). The CDI is a key benchmark for fixed income investments and interbank lending in Brazil. Data is available from 1986 onward with daily frequency (business days only).

Examples

```
## Not run:
# Default: last 30 days of CDI rate
df <- get_cdi_rate()

# Specific period
df2 <- get_cdi_rate("2023-01-01", "2023-03-31")

# Using year-month format for a specific month
df3 <- get_cdi_rate("2023-06", "2023-06")

# Portuguese column names and labels
df4 <- get_cdi_rate(language = "pt")

# Complete example with all parameters
df5 <- get_cdi_rate("2023-01-01", "2023-12-31", language = "pt", labels = TRUE)

# Historical analysis
df6 <- get_cdi_rate("2020-03-01", "2020-04-30") # COVID period

## End(Not run)
```

get_exchange_rate	<i>Get US Dollar Exchange Rate (Commercial)</i>
-------------------	---

Description

Downloads daily US dollar exchange rate (commercial, selling) from BCB/SGS. This function retrieves the daily exchange rate (SGS series 1) in Brazilian Real (R\$).

Usage

```
get_exchange_rate(
  start_date = NULL,
  end_date = NULL,
  language = "eng",
  labels = TRUE
)
```

Arguments

start_date	Start date for the data period. Accepts multiple formats: • "YYYY" for year only (e.g., "2020" becomes "2020-01-01") • "YYYY-MM" for year and month (e.g., "2020-06" becomes "2020-06-01") • "YYYY-MM-DD" for a specific date (e.g., "2020-06-15")
end_date	End date for the data period. Accepts the same formats as start_date: • "YYYY" (e.g., "2023" becomes "2023-12-31") • "YYYY-MM" (e.g., "2023-12" becomes the last day of December 2023) • "YYYY-MM-DD" for a specific date • NULL defaults to the current date (today)
language	Language for column names in the returned data.frame: • "eng" (default): Returns columns date and exchange_rate • "pt": Returns columns data_referencia and taxa_cambio
labels	Logical indicating whether to add variable labels using the labelled package. Labels provide descriptive text for each column when available.

Value

A data.frame with US dollar exchange rate. Columns depend on the language parameter:

- English (language = "eng"): date (Date), exchange_rate (numeric, R\$/US\$)
- Portuguese (language = "pt"): data_referencia (Date), taxa_cambio (numeric, R\$/US\$)

Note

Series information: This function uses SGS series 1, which represents the commercial US dollar selling rate (taxa de câmbio livre - dólar americano - venda). Data is available from 1984 onward with daily frequency.

Examples

```
## Not run:
# Default: last 30 days of exchange rate
df <- get_exchange_rate()

# Specific period
df2 <- get_exchange_rate("2023-01-01", "2023-03-31")
```

```

# Using year-month format for a specific month
df3 <- get_exchange_rate("2023-06", "2023-06")

# Portuguese column names and labels
df4 <- get_exchange_rate(language = "pt")

# Complete example with all parameters
df5 <- get_exchange_rate("2023-01-01", "2023-12-31", language = "pt", labels = TRUE)

## End(Not run)

```

get_gdp_growth

Get GDP Growth Rate

Description

Downloads quarterly GDP growth data (% change) from BCB/SGS (Brazilian Central Bank). This function retrieves nominal GDP values (SGS series 2010) and calculates the quarter-over-quarter growth rate.

Usage

```

get_gdp_growth(
  start_date = "2000-01-01",
  end_date = NULL,
  language = "eng",
  labels = TRUE
)

```

Arguments

start_date	Start date for the data period. Accepts multiple formats: "YYYY" for year only (e.g., "2020" becomes "2020-01-01") "YYYY-MM" for year and month (e.g., "2020-06" becomes "2020-06-01") "YYYY-MM-DD" for a specific date (e.g., "2020-06-15")
end_date	End date for the data period. Accepts the same formats as start_date: "YYYY" (e.g., "2023" becomes "2023-12-31") "YYYY-MM" (e.g., "2023-12" becomes the last day of December 2023) "YYYY-MM-DD" for a specific date - NULL defaults to the current date (today)
language	Language for column names in the returned data.frame: "eng" (default): Returns columns date and gdp_growth "pt": Returns columns data_referencia and crescimento_pib
labels	Logical indicating whether to add variable labels using the labelled package. Labels provide descriptive text for each column when available.

Value

A data.frame with GDP growth rate. Columns depend on the language parameter:

- English (language = "eng"): date (Date), gdp_growth (numeric, %)
- Portuguese (language = "pt"): data_referencia (Date), crescimento_pib (numeric, %)

Note

Important limitation: The nominal GDP series (SGS 2010) is currently available only until 2014. Requests for periods after 2014 will return empty results or a warning.

Examples

```
## Not run:
# Default: data from 2000 to current date (but limited to 2014)
df <- get_gdp_growth()

# Specific period (within available range)
df2 <- get_gdp_growth("2010", "2014")

# Using year-month format
df3 <- get_gdp_growth("2012-06", "2013-12")

# End date only (from earliest available to 2020-12-31)
df4 <- get_gdp_growth(end_date = "2020-12-01")

# Portuguese column names and labels
df5 <- get_gdp_growth(language = "pt")

# Complete example with all parameters
df6 <- get_gdp_growth("2011-01-01", "2014-12-31", language = "pt", labels = TRUE)

## End(Not run)
```

get_inflation_rate	<i>Get IPCA Inflation Data</i>
--------------------	--------------------------------

Description

Downloads monthly IPCA (Broad National Consumer Price Index) inflation data from the Brazilian Central Bank's SGS API and calculates accumulated inflation rates. IPCA is Brazil's official inflation index, calculated monthly by IBGE.

Usage

```
get_inflation_rate(
  start_date = "2012-01-01",
  end_date = NULL,
  language = "eng",
  labels = TRUE
)
```

Arguments

start_date	Start date for the data period. Accepts multiple formats: • "YYYY" for year only (e.g., "2020" becomes "2020-01-01") • "YYYY-MM" for year and month (e.g., "2020-06" becomes "2020-06-01") • "YYYY-MM-DD" for a specific date (e.g., "2020-06-15") • NULL defaults to "2020-01-01" (aligned with other functions in the package)
end_date	End date for the data period. Accepts the same formats as start_date: • "YYYY" (e.g., "2023" becomes "2023-12-31") • "YYYY-MM" (e.g., "2023-12" becomes the last day of December 2023) • "YYYY-MM-DD" for a specific date • NULL defaults to the current date (today)
language	Language for column names in the returned data.frame: • "eng" (default): Returns columns date, monthly_inflation, ytd_inflation, twelve_month_inflation • "pt": Returns columns data_referencia, inflacao_mensal, inflacao_acumulada_ano, inflacao_12_meses
labels	Logical indicating whether to add variable labels using the labelled package. Labels provide descriptive text for each column when available.

Value

A data.frame with inflation metrics. Columns depend on the language parameter:

- English (language = "eng"):
 - date (Date): Reference month
 - monthly_inflation (numeric): Monthly IPCA variation (%)
 - ytd_inflation (numeric): Year-to-date accumulated inflation (%)
 - twelve_month_inflation (numeric): 12-month accumulated inflation (%)
- Portuguese (language = "pt"):
 - data_referencia (Date): Mes de referencia
 - inflacao_mensal (numeric): Variacao mensal do IPCA (%)
 - inflacao_acumulada_ano (numeric): Inflacao acumulada no ano (%)
 - inflacao_12_meses (numeric): Inflacao acumulada nos ultimos 12 meses (%)

Note

Default Period: When `start_date = NULL`, defaults to "2020-01-01", providing data from the start of 2020. This period covers significant economic events including the COVID-19 pandemic and recent inflationary pressures in Brazil.

Data Processing: This function automatically downloads an extra 12 months of historical data to calculate 12-month accumulated inflation correctly. The final output is filtered to show only the requested period.

Calculation Details:

- Year-to-date inflation: Cumulative product of monthly rates within each calendar year
- 12-month inflation: Rolling 12-month cumulative product of monthly rates

Examples

```
## Not run:
# Default: from 2020 to current date (aligned with SELIC function)
df <- get_inflation_rate()

# Specific period with year-only format
df2 <- get_inflation_rate("2021", "2023")

# Using year-month format for precise month selection
df3 <- get_inflation_rate("2022-03", "2023-06")

# Portuguese column names and labels
df4 <- get_inflation_rate(language = "pt")

# Without variable labels
df5 <- get_inflation_rate("2020-01-01", "2022-12-31", labels = FALSE)

# Current year analysis
current_year <- format(Sys.Date(), "%Y")
df6 <- get_inflation_rate(start_date = current_year)

# Compare with SELIC rate (same default period)
selic_data <- get_selic_rate() # Also starts at 2020-01-01
inflation_data <- get_inflation_rate() # Same start date

## End(Not run)
```

get_selic_rate

Get Annual Brazilian SELIC Rate (Annualized, Base 252)

Description

Downloads the annual SELIC rate series from the Central Bank of Brazil's SGS API. The SELIC rate (Special System for Settlement and Custody) is Brazil's benchmark overnight interest rate, used as the primary monetary policy instrument.

Usage

```
get_selic_rate(
  start_date = "2020-01-01",
  end_date = NULL,
  language = "eng",
  labels = TRUE
)
```

Arguments

start_date	Start date for the data period. Accepts multiple formats: • "YYYY" for year only (e.g., "2020" becomes "2020-01-01") • "YYYY-MM" for year and month (e.g., "2020-06" becomes "2020-06-01") • "YYYY-MM-DD" for a specific date (e.g., "2020-06-15") • NULL defaults to "2020-01-01" (sensible default for analysis)
end_date	End date for the data period. Accepts the same formats as start_date: • "YYYY" (e.g., "2023" becomes "2023-12-31") • "YYYY-MM" (e.g., "2023-12" becomes the last day of December 2023) • "YYYY-MM-DD" for a specific date • NULL defaults to the current date (today)
language	Language for column names in the returned data.frame: • "eng" (default): Returns columns date and selic_rate • "pt": Returns columns data_referencia and taxa_selic
labels	Logical indicating whether to add variable labels using the labelled package. Labels provide descriptive text for each column when available.

Value

A data.frame with SELIC rate. Columns depend on the language parameter:

- English (language = "eng"): date (Date), selic_rate (numeric, % per year)
- Portuguese (language = "pt"): data_referencia (Date), taxa_selic (numeric, % ao ano)

Note

IMPORTANT API LIMITATION: The BCB API imposes a **10-year maximum window** for daily frequency series like SELIC. Requests spanning more than 10 years will fail. For longer historical analyses, split your request into multiple 10-year periods.

DEFAULT PERIOD: When start_date = NULL, defaults to "2020-01-01" (start of 2020), providing recent data while avoiding the 10-year API limit with current dates.

Examples

```
## Not run:
# Default: from 2020 to current date
df <- get_selic_rate()

# Specific period within 10-year limit
df2 <- get_selic_rate("2020-01-01", "2023-12-31")

# Last 5 years (respecting 10-year limit)
df3 <- get_selic_rate(start_date = "2019")

# Portuguese column names and labels
df4 <- get_selic_rate(language = "pt")

# Complete year analysis
df5 <- get_selic_rate("2018", "2023")

## End(Not run)
```

get_unemployment

Get Brazilian Quarterly Unemployment Rate

Description

Downloads quarterly unemployment rate data from IBGE's Continuous PNAD survey via the SIDRA API (table 6381). The unemployment rate represents the percentage of the labor force that is unemployed and actively seeking employment.

Usage

```
get_unemployment(
  start_date = "2020-01-01",
  end_date = NULL,
  language = "eng",
  labels = TRUE
)
```

Arguments

start_date	Start date for the data period. Accepts multiple formats: • "YYYY" for year only (e.g., "2020" becomes "2020-01-01") • "YYYY-MM" for year and month (e.g., "2020-06" becomes "2020-06-01") • "YYYY-MM-DD" for a specific date (e.g., "2020-06-15") • NULL defaults to "2020-01-01" (aligned with other functions in the package)
end_date	End date for the data period. Accepts the same formats as start_date:

	• "YYYY" (e.g., "2023" becomes "2023-12-31") • "YYYY-MM" (e.g., "2023-12" becomes the last day of December 2023) • "YYYY-MM-DD" for a specific date • NULL defaults to the current date (today)
language	Language for column names in the returned data.frame: • "eng" (default): Returns columns quarter, rate, month, year, date • "pt": Returns columns trimestre, taxa, mes, ano, data
labels	Logical indicating whether to add variable labels using the labelled package. Labels provide descriptive text for each column when available.

Value

A data.frame with unemployment rate. Columns depend on the language parameter:

- English (language = "eng"):
 - quarter (character): Moving quarter reference (e.g., "jan-mar 2023")
 - rate (numeric): Unemployment rate (%)
 - month (character): Quarter ending month abbreviation
 - year (numeric): Year
 - date (Date): Reference date (first day of quarter ending month)
- Portuguese (language = "pt"):
 - trimestre (character): Trimestre móvel de referência
 - taxa (numeric): Taxa de desemprego (%)
 - mes (character): Mês de término do trimestre
 - ano (numeric): Ano
 - data (Date): Data de referência (primeiro dia do mês de término)

Note

Data Source: IBGE's Continuous National Household Sample Survey (PNAD Contínua), table 6381 (Unemployment rate), series 4099. Data is available from 2012 onward with quarterly frequency (moving quarters).

Date Calculation: The date column represents the first day of the month that ends the moving quarter (e.g., "jan-mar 2023" becomes "2023-03-01").

Examples

```
## Not run:
# Default: from 2020 to current date (aligned with other functions)
df <- get_unemployment()

# Specific period with year-only format
df2 <- get_unemployment("2018", "2023")

# Portuguese column names and labels
df3 <- get_unemployment(language = "pt")
```

```
# Without variable labels
df4 <- get_unemployment("2020-01-01", "2022-12-31", labels = FALSE)

# Compare unemployment with inflation (same period)
unemployment_data <- get_unemployment("2020", "2023")
inflation_data <- get_inflation_rate("2020", "2023")

## End(Not run)
```

plot_cdi_rate

Plot Brazilian CDI rate

Description

Generates a time series plot of the CDI (Certificado de Depósito Interbancário) rate. The CDI is the benchmark interest rate for interbank deposits in Brazil and serves as a reference for many fixed income investments.

Usage

```
plot_cdi_rate(data, language = "eng")
```

Arguments

data	Tibble returned by get_cdi_rate(), with columns date and value.
language	Language for titles and labels: "pt" (Portuguese) or "eng" (English).

Value

A ggplot2 object showing the CDI rate over time.

Examples

```
## Not run:
# Example 1: English version
cdi_data <- get_cdi_rate(2020, 2024)
cdi_plot <- plot_cdi_rate(cdi_data)
print(cdi_plot)

# Example 2: Portuguese version
dados_cdi <- get_cdi_rate(2020, 2024, language = "pt")
grafico_cdi <- plot_cdi_rate(dados_cdi, language = "pt")
print(grafico_cdi)

## End(Not run)
```

plot_exchange_rate	<i>Plot Brazilian exchange rate (USD/BRL)</i>
--------------------	---

Description

Generates a time series plot of the USD/BRL exchange rate using data from `get_exchange_rate()`. Shows the commercial exchange rate for US Dollar to Brazilian Real.

Usage

```
plot_exchange_rate(data, language = "eng")
```

Arguments

data	Tibble returned by <code>get_exchange_rate()</code>
language	Language for titles and labels: "pt" (Portuguese) or "eng" (English).

Value

A ggplot2 object showing the exchange rate over time.

Examples

```
## Not run:
# Example 1: English version
exchange_data <- get_exchange_rate("2023-01-01", "2023-12-31")
exchange_plot <- plot_exchange_rate(exchange_data)
print(exchange_plot)

# Example 2: Portuguese version
dados_cambio <- get_exchange_rate("2023-01-01", "2023-12-31", language = "pt")
grafico_cambio <- plot_exchange_rate(dados_cambio, language = "pt")
print(grafico_cambio)

## End(Not run)
```

plot_inflation_rate	<i>Plot Brazilian Inflation Rate (IPCA)</i>
---------------------	---

Description

Generates a time series plot of Brazil's monthly inflation rate measured by the IPCA (Índice Nacional de Preços ao Consumidor Amplo). The IPCA is Brazil's official inflation index and is widely used for monetary policy decisions, contract indexation, and economic analysis.

Usage

```
plot_inflation_rate(data, language = "eng")
```

Arguments

- data Tibble returned by get_inflation_rate(), with columns date and value. The value column represents the monthly IPCA inflation rate (%).
- language Language for titles and labels: "pt" (Portuguese) or "eng" (English).

Value

A ggplot2 object showing the monthly inflation rate over time.

Examples

```
## Not run:
# Example 1: English version
inflation_data <- get_inflation_rate(2020, 2024)
inflation_plot <- plot_inflation_rate(inflation_data)
print(inflation_plot)

# Example 2: Portuguese version
dados_inflacao <- get_inflation_rate(2020, 2024, language = "pt")
grafico_inflacao <- plot_inflation_rate(dados_inflacao, language = "pt")
print(grafico_inflacao)

## End(Not run)
```

plot_selic_rate	<i>Plot Brazilian SELIC rate (annualized, base 252)</i>
-----------------	---

Description

Generates a time series plot of the SELIC interest rate using data from get_selic(). The SELIC rate ("Sistema Especial de Liquidação e de Custódia") represents the effective annualized rate (252-business-day basis) for overnight interbank loans and is the main instrument of Brazil's monetary policy.

Usage

```
plot_selic_rate(data, language = "eng")
```

Arguments

- data Tibble returned by get_selic_rate()
- language Language for titles and labels: "pt" (Portuguese) or "eng" (English).

Value

A ggplot2 object showing the SELIC rate over time.

Examples

```
## Not run:
# Example 1: English version
selic_data <- get_selic_rate(2020, 2024)
selic_plot <- plot_selic_rate(selic_data)
print(selic_plot)

# Example 2: Portuguese version
dados_selic <- get_selic_rate(2020, 2024, language = "pt")
grafico_selic <- plot_selic_rate(dados_selic, language = "pt")
print(grafico_selic)

## End(Not run)
```

plot_series_comparison

Compare multiple financial/time series indices

Description

Plots multiple time series on the same chart for comparison.

Usage

```
plot_series_comparison(
  data_list,
  y_vars,
  date_vars,
  language = "eng",
  scale_type = c("none", "index", "percent_change"),
  title = NULL,
  subtitle = NULL,
  y_label = NULL,
  caption = NULL,
  colors = NULL,
  line_types = NULL,
  show_legend = TRUE,
  legend_position = "bottom"
)
```

Arguments

<code>data_list</code>	Named list of data frames, each returned by a <code>get_*</code> function
<code>y_vars</code>	Vector of column names containing the values to plot from each data frame
<code>date_vars</code>	Vector of column names containing dates from each data frame
<code>language</code>	Language for labels: "pt" (Portuguese) or "eng" (English)
<code>scale_type</code>	Type of scaling: "none" (raw values), "index" (index to 100), "percent_change" (percentage change from first observation)
<code>title</code>	Plot title
<code>subtitle</code>	Plot subtitle
<code>y_label</code>	Y-axis label
<code>caption</code>	Plot caption
<code>colors</code>	Vector of colors for each series
<code>line_types</code>	Vector of line types for each series
<code>show_legend</code>	Whether to show the legend (default: TRUE)
<code>legend_position</code>	Position of legend ("bottom", "top", "left", "right", or "none")

Value

A ggplot2 object

Examples

```
## Not run:
# Example comparing multiple series
selic <- get_selic_rate(2020, 2024)
ipca <- get_ipca(2020, 2024)
igpm <- get_igpm(2020, 2024)

comparison_plot <- plot_series_comparison(
  data_list = list(SELIC = selic, IPCA = ipca, IGP-M = igpm),
  y_vars = c("rate", "value", "value"),
  date_vars = c("date", "date", "date"),
  scale_type = "index",
  title = "Comparison of Brazilian Economic Indicators",
  y_label = "Index (2020-01 = 100)",
  language = "eng"
)
print(comparison_plot)

## End(Not run)
```

plot_unemployment	<i>Plot Brazil's quarterly unemployment rate</i>
-------------------	--

Description

Generates a ggplot2 line chart of unemployment rate in Brazil.

Usage

```
plot_unemployment(data, language = "eng")
```

Arguments

data	Tibble returned by get_unemployment()
language	Language for column names: "pt" for Portuguese or "eng" (default) for English

Value

A ggplot2 object

Examples

```
## Not run:
# Example 1: English version
unemployment_data <- get_unemployment(2020, 2024)
unemployment_plot <- plot_unemployment(unemployment_data)
print(unemployment_plot)

# Example 2: Portuguese version
dados_desemprego <- get_unemployment(2020, 2024, language = "pt")
grafico_desemprego <- plot_unemployment(dados_desemprego, language = "pt")
print(grafico_desemprego)

## End(Not run)
```

Index

`get_cdi_rate`, [2](#)
`get_exchange_rate`, [3](#)
`get_gdp_growth`, [5](#)
`get_inflation_rate`, [6](#)
`get_selic_rate`, [8](#)
`get_unemployment`, [10](#)

`plot_cdi_rate`, [12](#)
`plot_exchange_rate`, [13](#)
`plot_inflation_rate`, [13](#)
`plot_selic_rate`, [14](#)
`plot_series_comparison`, [15](#)
`plot_unemployment`, [17](#)