

Package ‘agregR’

August 21, 2026

Encoding UTF-8

Type Package

URL <https://rnmag.github.io/agregR/>, <https://github.com/rnmag/agregR>

Title Bayesian State-Space Aggregation of Brazilian Presidential Polls

Description A set of dynamic measurement models to estimate latent vote shares from noisy polling sources. The models build on Jackman (2009, ISBN: 9780470011546) and feature specialized methods for bias adjustment based on past performance and correction for asymmetric errors based on candidate political alignment.

Version 1.0.4

Depends R (>= 4.1.0), ggdist (>= 3.1.0)

Imports instantiate, tibble, dplyr, tidyr, purrr, readr, stringr, tidyselect, ggplot2, ggrepel, scales, lubridate, cli, ragg, sysfonts, showtext, stringi, grid, systemfonts

Suggests bayesplot, cmdstanr (>= 0.5.0), testthat (>= 3.0.0)

Additional_repositories <https://stan-dev.r-universe.dev>

SystemRequirements CmdStan
(<https://mc-stan.org/users/interfaces/cmdstan>)

LazyData true

Config/testthat/edition 3

License MIT + file LICENSE

Config/roxygen2/version 8.1.0

NeedsCompilation yes

Author Rafael N. Magalhães [aut, cre]

Maintainer Rafael N. Magalhães <rnunesmagalhaes@gmail.com>

Repository CRAN

Date/Publication 2026-08-21 20:50:02 UTC

Contents

configurar_agregador	2
configurar_grafico	4
configurar_prioris	5
grafico_agregador	8
grafico_priori_posteriori	9
grafico_vies	10
historico_pesquisas_poder360	12
rodar_agregador	13

Index	17
--------------	-----------

configurar_agregador	<i>Configuration function for Poll Aggregator</i>
----------------------	---

Description

Defines configuration parameters for the poll aggregator, including Stan settings, and election details.

Usage

```
configurar_agregador(
  pesquisas = NULL,
  resultado_eleicao_passada = NULL,
  resultado_eleicao_atual = NULL,
  historico_pesquisas = NULL,
  candidaturas_1t = NULL,
  candidaturas_2t = NULL,
  direita_eleicao_atual = NULL,
  direita_eleicao_passada = "Bolsonaro",
  esquerda_eleicao_atual = NULL,
  esquerda_eleicao_passada = "Lula",
  eleicao_passada_primeiro_turno = "2/10/2022",
  eleicao_passada_segundo_turno = "30/10/2022",
  stan_cores = pmin(parallel::detectCores(), 4),
  stan_chains = 4,
  stan_warmup = 500,
  stan_sampling = 500,
  stan_init = 0.1,
  stan_adapt_delta = 0.95,
  saida_bases_tratadas = "resultados_agregador/bases_tratadas",
  saida_modelos_brutos = "resultados_agregador/modelos_brutos"
)
```

Arguments

<code>pesquisas</code>	Path to a CSV file or URL containing current poll data. Defaults to a GitHub Raw URL.
<code>resultado_eleicao_passada</code>	Path to a CSV file containing results from the previous election. Defaults to a GitHub Raw URL.
<code>resultado_eleicao_atual</code>	Path to a CSV file containing results for the current election (useful for retrospective model). Defaults to a GitHub Raw URL.
<code>historico_pesquisas</code>	Path to a CSV/RDS file containing historical poll data. If NULL (default), uses the package's internal dataset.
<code>candidaturas_1t</code>	Character vector of candidates in the 1st round. If NULL, uses default candidates.
<code>candidaturas_2t</code>	Character vector of candidates in the 2nd round. If NULL, uses default candidates.
<code>direita_eleicao_atual</code>	Character vector of right-wing candidates in the current race. If NULL, uses default candidates. The model can compensate institute errors against right-wing candidates in the last election.
<code>direita_eleicao_passada</code>	Name of the right-wing candidate in the previous election.
<code>esquerda_eleicao_atual</code>	Character vector of left-wing candidates in the current race. If NULL, uses default candidates. The model can compensate institute errors against left-wing candidates in the last election.
<code>esquerda_eleicao_passada</code>	Name of the left-wing candidate in the previous election.
<code>eleicao_passada_primeiro_turno</code>	Date of the previous 1st round (e.g., "2/10/2022").
<code>eleicao_passada_segundo_turno</code>	Date of the previous 2nd round (e.g., "30/10/2022").
<code>stan_cores</code>	Number of CPU cores for Stan to use.
<code>stan_chains</code>	Number of MCMC chains.
<code>stan_warmup</code>	Number of warmup iterations per chain.
<code>stan_sampling</code>	Number of sampling iterations per chain.
<code>stan_init</code>	Initial value for Stan parameters.
<code>stan_adapt_delta</code>	The target acceptance rate for Stan's NUTS algorithm.
<code>saida_bases_tratadas</code>	Directory where treated data will be saved.
<code>saida_modelos_brutos</code>	Directory where raw model objects will be saved.

Value

A list of configuration parameters.

Examples

```
# Create custom Stan settings
cfg_custom <- configurar_agregador(
  stan_warmup = 100,
  stan_sampling = 100
)
```

configurar_grafico	<i>Configuration for Graphics</i>
--------------------	-----------------------------------

Description

Defines configuration parameters for graphics, including colors, fonts, and dimensions.

Usage

```
configurar_grafico(
  fonte = "Fira Sans",
  cores_candidaturas = NULL,
  simbolos = NULL,
  graf_largura = 2918,
  graf_altura = 1913,
  graf_unidade = "px",
  graf_dpi = 320,
  dir_grafico = "resultados_agregador/graficos"
)
```

Arguments

fonte	Font family (default: "Fira Sans").
cores_candidaturas	Named vector or list of colors for candidates. Can be a partial override.
simbolos	Named vector or list of symbols for methodologies. Can be a partial override.
graf_largura	Width of saved plots.
graf_altura	Height of saved plots.
graf_unidade	Unit for dimensions ("px", "in", "cm", "mm").
graf_dpi	DPI for saved plots.
dir_grafico	Directory to save plots.

Value

A list of graphic configuration parameters.

Examples

```
# Alternative colors for use in the config_grafico argument in a plot
config_custom <- configurar_grafico(
  cores_candidaturas = c(Lula = "darkred")
)
```

configurar_prioris	<i>Configuration for Statistical Models</i>
--------------------	---

Description

Defines hyperparameters for the specific Bayesian models.

Usage

```
configurar_prioris(
  nome = NULL,
  mu_priori = NULL,
  sd_mu_priori = NULL,
  omega_eta_priori = NULL,
  nu_priori = NULL,
  sd_nu_priori = NULL,
  omega_zeta_priori = NULL,
  delta_priori = NULL,
  sd_delta_priori = NULL,
  tau_priori = NULL,
  sd_tau_priori = NULL,
  ...
)
```

Arguments

nome	Name of the model. If NULL, inherits the model used in <code>rodar_agregador()</code> .
mu_priori	Prior mean for the latent vote share at $t = 1$. See "Priors Details" section.
sd_mu_priori	Prior uncertainty for the initial latent vote.
omega_eta_priori	Prior mean for the level volatility (ω_η).
nu_priori	Prior mean for the initial trend (daily growth rate).
sd_nu_priori	Prior uncertainty for the initial trend.
omega_zeta_priori	Prior mean for the trend volatility (ω_ζ).
delta_priori	Mean expected bias for institutes.
sd_delta_priori	Scale of the bias prior.
tau_priori	Mean expected magnitude of errors not explained by sampling or house effects.
sd_tau_priori	Prior uncertainty for non-sampling error.
...	Additional arguments to override default hyperparameters.

Value

A list of model parameters.

Priors Details

These hyperparameters control the strength of assumptions regarding latent state evolution, institute bias, and non-sampling errors.

Variable names refer to the model notation described in <https://rnmag.github.io/agregR/index.html#conceptual-framework>

Recommended reading: <https://github.com/stan-dev/stan/wiki/prior-choice-recommendations>

State Model - Level (μ)

- `mu_priori`: Prior mean for the latent vote share at $t = 1$.
- `sd_mu_priori`: Prior uncertainty for the initial latent vote.
 - *Default values*: μ starts with a flat prior of $N(0.5, 0.5)$, allowing data to quickly dominate inference.
- `omega_eta_priori`: Prior mean for the level volatility (ω_η).
 - *Default values*: With `omega_eta_priori = 0.002`, the model assumes a **baseline drift** of approx. ± 2 percentage points over a month ($1.96 \times \sqrt{30} \times 0.002 \approx 0.02$).
 - *Higher values*: The latent vote (μ) can jump more from one day to the next. The model adapts more quickly to new polls but becomes more "jittery".
 - *Lower values*: The model assumes the public opinion level is more stable over time, resulting in smoother curves.

State Model - Trend (ν)

- `nu_priori`: Prior mean for the initial trend (daily growth rate).
- `sd_nu_priori`: Prior uncertainty for the initial trend.
 - *Default values*: With `nu_priori = 0` and `sd_nu_priori = 0.001`, the model expects an initial trend within ± 0.2 percentage points per day ($1.96 \times 0.001 \approx 0.002$).
- `omega_zeta_priori`: Prior mean for the trend volatility (ω_ζ).
 - *Default values*: With `omega_zeta_priori = 0.00001`, the model assumes a linear evolution, allowing the trend to shift rapidly (accelerations) only under strong evidence.
 - *Higher values*: The trend (ν) can change direction or magnitude rapidly.
 - *Lower values*: The trend is assumed to be more constant over time (more linear evolution).

Institute Bias (δ)

- `delta_priori`: Mean expected bias for institutes. Default is 0, except in "Viés Empírico" where it is anchored on past performance.
- `sd_delta_priori`: Scale of the bias prior.
 - *Default values*: With `delta_priori = 0` and `sd_delta_priori = 0.02`, the model assumes that 95% of institutes have a bias within ± 4 percentage points ($1.96 \times 0.02 \approx 0.04$).
 - *Higher values*: Allow for larger, more variable biases across institutes.

- *Lower values*: Constrain institutes to have similar biases (shrinkage toward the anchor).

Non-Sampling Error (τ)

- `tau_priori`: Mean expected magnitude of errors not explained by sampling or house effects. In weighted models, this is replaced by the empirical RMSE from past elections.
- `sd_tau_priori`: Prior uncertainty for non-sampling error.
 - *Default values*: With `tau_priori = 0.02` and `sd_tau_priori = 0.02`, the model assumes a **baseline** of ± 4 percentage points of "noise" in each poll, allowing it to spread closer to ± 7 percentage points.
 - *Higher values*: The model treats polls as less precise, widening the credible intervals of the latent state.
 - *Lower values*: The model trusts polling precision more, leading to tighter intervals and potentially more sensitivity to outliers.

Model Defaults

Each model has a specific set of default priors.

Viés Relativo com Pesos (Default)

- `mu_priori = 0.5`, `sd_mu_priori = 0.5`, `omega_eta_priori = 0.002`
- `nu_priori = 0`, `sd_nu_priori = 0.001`, `omega_zeta_priori = 0.00001`
- `delta_priori = 0`, `sd_delta_priori = 0.02`
- `sd_tau_priori = 0.02`
- *Not applicable*: `tau_priori` (model uses empirical values)

Viés Relativo sem Pesos

- `mu_priori = 0.5`, `sd_mu_priori = 0.5`, `omega_eta_priori = 0.002`
- `nu_priori = 0`, `sd_nu_priori = 0.001`, `omega_zeta_priori = 0.00001`
- `delta_priori = 0`, `sd_delta_priori = 0.02`
- `tau_priori = 0.02`, `sd_tau_priori = 0.02`

Viés Empírico

- `mu_priori = 0.5`, `sd_mu_priori = 0.5`, `omega_eta_priori = 0.002`
- `nu_priori = 0`, `sd_nu_priori = 0.001`, `omega_zeta_priori = 0.00001`
- `sd_delta_priori = 0.02`
- `sd_tau_priori = 0.02`
- *Not applicable*: `delta_priori` and `tau_priori` (model uses empirical values for both)

Retrospectivo

- `mu_priori = 0.5`, `sd_mu_priori = 0.5`, `omega_eta_priori = 0.002`
- `nu_priori = 0`, `sd_nu_priori = 0.001`, `omega_zeta_priori = 0.00001`
- `delta_priori = 0`, `sd_delta_priori = 0.02`

- `tau_priori = 0.02, sd_tau_priori = 0.02`

Naive

- `mu_priori = 0.5, sd_mu_priori = 0.5, omega_eta_priori = 0.002`
- *Not applicable:* `nu_priori, sd_nu_priori, omega_zeta_priori, delta_priori, sd_delta_priori, tau_priori, sd_tau_priori`

Examples

```
# Get default parameters for the "Naive" model
naive_params <- configurar_priors(nome = "Naive")

# Get parameters for "Naive" and override a default value
custom_params <- configurar_priors(nome = "Naive", sd_mu_priori = 0.2)
```

grafico_agregador	<i>Plot Aggregator Results</i>
-------------------	--------------------------------

Description

Generates a plot of the aggregated poll results over time.

Usage

```
grafico_agregador(
  bd,
  salvar = FALSE,
  config_grafico = configurar_grafico(),
  dir_saida = NULL,
  ...
)
```

Arguments

<code>bd</code>	The results object returned by <code>rodar_agregador()</code> .
<code>salvar</code>	Logical. If TRUE, saves the plot to disk.
<code>config_grafico</code>	A list of graphic parameters created by <code>configurar_grafico()</code> .
<code>dir_saida</code>	Output directory for the saved plot if <code>salvar = TRUE</code> .
<code>...</code>	Additional arguments.

Value

A ggplot2 object.

Examples

```
if (instantiate::stan_cmdstan_exists()) {
  result <- rodar_agregador(
    data_inicio = "01/01/2025",
    turno = 2,
    cenario = "Lula vs Bolsonaro"
  )

  # Standard plot
  std_plot <- grafico_agregador(result)

  # Altering candidate colors
  custom_plot <- grafico_agregador(
    result,
    config_grafico = configurar_grafico(
      cores_candidaturas = c(Lula = "yellow")
    )
  )
}
```

grafico_priori_posteriori

Plot Prior vs Posterior

Description

Generates a plot comparing prior and posterior distributions for candidates or bias.

Usage

```
grafico_priori_posteriori(
  bd,
  candidaturas,
  tipo = "Viés",
  salvar = FALSE,
  config_agregador = configurar_agregador(),
  config_grafico = configurar_grafico(),
  config_prioris = configurar_prioris(bd$nome_modelo),
  dir_saida = NULL
)
```

Arguments

bd	The results object returned by <code>rodar_agregador()</code> .
candidaturas	A character vector of candidate names to include in the plot.
tipo	The type of da to plot: "Viés" (for institute bias) or "Percentual" (for candidate voting share).

salvar Logical. If TRUE, saves the plot to disk.
config_agregador A list of configuration parameters created by `configurar_agregador()`.
config_grafico A list of graphic parameters created by `configurar_grafico()`.
config_prioris A list of model hyperparameters created by `configurar_prioris()`.
dir_saida Output directory for the saved plot if `salvar = TRUE`.

Value

A ggplot2 object.

Examples

```

if (instantiate::stan_cmdstan_exists()) {
  result <- rodar_agregador(
    data_inicio = "01/01/2025",
    turno = 2,
    cenario = "Lula vs Bolsonaro"
  )

  # Prior vs Posterior plot for institute bias
  std_plot <- grafico_priori_posteriori(
    result,
    tipo = "Viés",
    candidaturas = c("Lula", "Bolsonaro")
  )

  # Altering candidate colors
  custom_plot <- grafico_priori_posteriori(
    result,
    candidaturas = c("Lula", "Bolsonaro"),
    config_grafico = configurar_grafico(
      cores_candidaturas = c(Lula = "yellow")
    )
  )
}

```

grafico_vies

Plot Institute Bias

Description

Generates a plot visualizing the bias of polling institutes.

Usage

```
grafico_vies(  
  bd,  
  candidaturas,  
  salvar = FALSE,  
  config_grafico = configurar_grafico(),  
  dir_saida = NULL,  
  ...  
)
```

Arguments

bd	The results object returned by <code>rodar_agregador()</code> .
candidaturas	A character vector of candidate names to include in the plot.
salvar	Logical. If TRUE, saves the plot to disk.
config_grafico	A list of graphic parameters created by <code>configurar_grafico()</code> .
dir_saida	Output directory for the saved plot if <code>salvar = TRUE</code> .
...	Additional arguments.

Value

A ggplot2 object.

Examples

```
if (instantiate::stan_cmdstan_exists()) {  
  result <- rodar_agregador(  
    data_inicio = "01/01/2025",  
    turno = 2,  
    cenario = "Lula vs Bolsonaro"  
  )  
  
  # Standard bias plot  
  std_plot <- grafico_vies(  
    result,  
    candidaturas = c("Lula", "Bolsonaro")  
  )  
  
  # Altering candidate colors  
  custom_plot <- grafico_vies(  
    result,  
    candidaturas = c("Lula", "Bolsonaro"),  
    config_grafico = configurar_grafico(  
      cores_candidaturas = c(Lula = "yellow")  
    )  
  )  
}
```

historico_pesquisas_poder360

Historical Polls by Poder360

Description

A dataset containing historical electoral polls compiled by Poder360. This dataset is used to calculate empirical priors for the models.

Usage

historico_pesquisas_poder360

Format

A data frame with columns:

ano Election year
cargo Office being contested
condicao Condition (e.g., stimulated)
contratante Entity that paid for the poll
data Date of the poll
data_referencia Reference date for the poll
descricao_cenario Description of the electoral scenario
id_candidato_poder360 Unique ID for the candidate
id_cenario Unique ID for the scenario
id_pesquisa Unique ID for the poll
instituto Name of the polling institute
margem_mais Upper margin of error
margem_menos Lower margin of error
nome_candidato Candidate name
nome_municipio City name (if applicable)
numero_registro Official registration number
orgao_registro Entity where the poll was registered
percentual Voting intention percentage
quantidade_entrevistas Sample size
sigla_partido Political party abbreviation
sigla_uf State abbreviation
tipo Poll type
tipo_voto Vote type (Total, Valid, etc.)
turno Election round (1 or 2)

Source

Poder360 via [Base dos Dados](#)

rodar_agregador	<i>Run Poll Aggregator</i>
-----------------	----------------------------

Description

Main function to run the state-space model for poll aggregation.

Usage

```
rodar_agregador(
  bd = NULL,
  data_inicio = NULL,
  data_fim = Sys.Date(),
  cargo = "Presidente",
  ambito = "Brasil",
  cenario = NULL,
  turno,
  modelo = "Viés Relativo com Pesos",
  config_agregador = NULL,
  config_prioris = NULL,
  salvar = FALSE,
  dir_saida = NULL
)
```

Arguments

bd	Dataframe or path to a CSV file containing poll data.
data_inicio	Start date for the analysis (mandatory).
data_fim	End date for the analysis.
cargo	The office/position being contested (e.g., "Presidente"). Current data only contains presidential polls, but the package supports expansion for other offices.
ambito	The geographical scope (e.g., "Brasil"). Current data only contains national polls, but the package supports expansion for state races.
cenario	The specific electoral scenario. Mandatory for second round.
turno	The election round (1 or 2).
modelo	The name of the model to run. Options: "Viés Relativo com Pesos" (default), "Viés Relativo sem Pesos", "Viés Empírico", "Retrospectivo" and "Naive".
config_agregador	A list of configuration parameters created by configurar_agregador() . If NULL, uses defaults.

<code>config_priors</code>	A list of model hyperparameters created by <code>configurar_priors()</code> . If NULL, uses defaults based on <code>modelo</code> .
<code>salvar</code>	Logical. If TRUE, saves the results to disk.
<code>dir_saida</code>	Output directory for saved files if <code>salvar = TRUE</code> .

Value

A list containing the model name, estimated votes, institute bias, and the raw model object.

Model Details

The aggregator supports five types of Bayesian state-space models, each with specific assumptions about institute bias and non-sampling errors:

1. Viés Relativo com Pesos (Default)

- **Assumption:** Institute biases are relative to the average of all institutes (latent "truth" is anchored to the consensus).
- **Bias (δ):** Calculated relative to the mean bias of all institutes.
- **Weights (τ):** Uses past election performance to weight the non-sampling error. Institutes with larger historical errors have less influence on the current estimate.
- **Use case:** Best for general forecasting when historical data is available.

2. Viés Relativo sem Pesos

- **Assumption:** Same as above, but treats all institutes as having equal potential quality a priori.
- **Bias (δ):** Calculated relative to the mean bias.
- **Weights (τ):** None. All institutes share the same prior for non-sampling error.
- **Use case:** When historical data is unreliable or when a "fresh start" assumption is desired.

3. Viés Empírico

- **Assumption:** Institute biases are anchored to their specific historical performance.
- **Bias (δ):** Prior means are set to the bias observed in the previous election (directional error).
- **Weights (τ):** Uses past performance for non-sampling error, similar to the "Com Pesos" model.
- **Use case:** When institutes are expected to repeat their specific past directional errors (e.g., consistently underestimating a specific wing).

4. Retrospectivo

- **Assumption:** The true election result is known and used as the final anchor for the state-space model.
- **Method:** Runs the model "backwards" or constrained by the final result to estimate the true path of public opinion.
- **Use case:** Post-election analysis to diagnose institute performance and calculate accurate biases for future calibration.

5. Naive

- **Assumption:** Polls have no bias and no non-sampling error.
- **Method:** A random walk model where the only source of uncertainty is the sampling error (σ).
- **Use case:** Baseline comparison. Assumes "polls are perfect" within their margin of error.

Priors Details

The `config_priors` argument allows customization of the model's hyperparameters with the `configurar_priors()` function.

These hyperparameters control the strength of assumptions regarding latent state evolution, institute bias, and non-sampling errors.

Variable names refer to the model notation described in <https://rnmag.github.io/agregR/index.html#conceptual-framework>

Recommended reading: <https://github.com/stan-dev/stan/wiki/prior-choice-recommendations>

State Model - Level (μ)

- `mu_priori`: Prior mean for the latent vote share at $t = 1$.
- `sd_mu_priori`: Prior uncertainty for the initial latent vote.
 - *Default values:* μ starts with a flat prior of $N(0.5, 0.5)$, allowing data to quickly dominate inference.
- `omega_eta_priori`: Prior mean for the level volatility (ω_η).
 - *Default values:* With `omega_eta_priori = 0.002`, the model assumes a **baseline drift** of approx. ± 2 percentage points over a month ($1.96 \times \sqrt{30} \times 0.002 \approx 0.02$).
 - *Higher values:* The latent vote (μ) can jump more from one day to the next. The model adapts more quickly to new polls but becomes more "jittery".
 - *Lower values:* The model assumes the public opinion level is more stable over time, resulting in smoother curves.

State Model - Trend (ν)

- `nu_priori`: Prior mean for the initial trend (daily growth rate).
- `sd_nu_priori`: Prior uncertainty for the initial trend.
 - *Default values:* With `nu_priori = 0` and `sd_nu_priori = 0.001`, the model expects an initial trend within ± 0.2 percentage points per day ($1.96 \times 0.001 \approx 0.002$).
- `omega_zeta_priori`: Prior mean for the trend volatility (ω_ζ).
 - *Default values:* With `omega_zeta_priori = 0.00001`, the model assumes a linear evolution, allowing the trend to shift rapidly (accelerations) only under strong evidence.
 - *Higher values:* The trend (ν) can change direction or magnitude rapidly.
 - *Lower values:* The trend is assumed to be more constant over time (more linear evolution).

Institute Bias (δ)

- `delta_priori`: Mean expected bias for institutes. Default is 0, except in "Viés Empírico" where it is anchored on past performance.
- `sd_delta_priori`: Scale of the bias prior.
 - *Default values*: With `delta_priori = 0` and `sd_delta_priori = 0.02`, the model assumes that 95% of institutes have a bias within ± 4 percentage points ($1.96 \times 0.02 \approx 0.04$).
 - *Higher values*: Allow for larger, more variable biases across institutes.
 - *Lower values*: Constrain institutes to have similar biases (shrinkage toward the anchor).

Non-Sampling Error (τ)

- `tau_priori`: Mean expected magnitude of errors not explained by sampling or house effects. In weighted models, this is replaced by the empirical RMSE from past elections.
- `sd_tau_priori`: Prior uncertainty for non-sampling error.
 - *Default values*: With `tau_priori = 0.02` and `sd_tau_priori = 0.02`, the model assumes a **baseline** of ± 4 percentage points of "noise" in each poll, allowing it to spread closer to ± 7 percentage points.
 - *Higher values*: The model treats polls as less precise, widening the credible intervals of the latent state.
 - *Lower values*: The model trusts polling precision more, leading to tighter intervals and potentially more sensitivity to outliers.

Examples

```
# Running the default model for a second round scenario
if (instantiate::stan_cmdstan_exists()) {
  result <- rodar_agregador(
    data_inicio = "01/01/2025",
    turno = 2,
    cenario = "Lula vs Bolsonaro"
  )

# Tuning Stan, changing the model and altering specific priors
custom_result <- rodar_agregador(
  data_inicio = "01/01/2025",
  turno = 2,
  cenario = "Lula vs Bolsonaro",
  modelo = "Viés Relativo sem Pesos",
  config_agregador = list(stan_chains = 1, stan_warmup = 200),
  config_priors = list(tau_priori = 0.01)
)
}
```

Index

* datasets

historico_pesquisas_poder360, [12](#)

configurar_agregador, [2](#)

configurar_agregador(), [10](#), [13](#)

configurar_grafico, [4](#)

configurar_grafico(), [8](#), [10](#), [11](#)

configurar_prioris, [5](#)

configurar_prioris(), [10](#), [14](#), [15](#)

grafico_agregador, [8](#)

grafico_priori_posteriori, [9](#)

grafico_vies, [10](#)

historico_pesquisas_poder360, [12](#)

rodar_agregador, [13](#)

rodar_agregador(), [8](#), [9](#), [11](#)