

Pacote R: cenários estocásticos da dívida pública

Alessandro Casalecchi

Nota Técnica - Nº 59

Setembro de 2025

SENADO FEDERAL

Presidente do Senado Federal
Senador Davi Alcolumbre (União-AP)

INSTITUIÇÃO FISCAL INDEPENDENTE**Diretor-Executivo**

Marcus Vinícius Caetano Pestana da Silva

Diretor

Alexandre Augusto Seijas de Andrade

Analistas

Alessandro Ribeiro de Carvalho Casalecchi
Lucas Gabriel Martins de Oliveira
Lucas Vinícius Penha Martins Bomfim Leal
Pedro Henrique Oliveira de Souza
Rafael da Rocha Mendonça Bacciotti

Assessora de Comunicação

Carmensita Corso

Estagiários

Lukas Cortez de Medeiros Balogh Fagundes
Márcio Eduardo Fernandes Domingos

Secretaria executiva

Thuane Vieira Rocha

Layout do relatório

CODIV/SECOM e SEFPRO/SEGRAF

Pacote R: cenários estocásticos da dívida pública

Alessandro Casalecchi¹

Esta nota técnica, de caráter estritamente metodológico, divulga o pacote computacional utilizado pela IFI para produzir o componente estocástico da ferramenta de análise de sustentabilidade da dívida pública (conhecido como Debt Sustainability Analysis – DSA na nomenclatura internacional), juntamente com um guia de uso. O pacote está hospedado em <https://github.com/ifibrasil/stocdebt>.

O texto tem quatro objetivos. O primeiro objetivo é ampliar a transparência dos métodos de análise da instituição. O segundo é facilitar a troca de experiências com outros órgãos governamentais brasileiros e IFIs de outros países. O terceiro é institucionalizar o método, perpetuando sua disponibilidade nos processos internos da IFI. O quarto objetivo é promover o aprimoramento da ferramenta de forma continuada, a partir de eventuais contribuições externas para o código, que é aberto.

O pacote, que foi elaborado na linguagem R, está totalmente documentado – para além desta nota técnica – por meio dos manuais que o acompanham, e está disponível em repositório online público.

O pacote vem sendo utilizado pela IFI desde 2023, nas edições do Relatório de Acompanhamento Fiscal em que há revisão de cenários macrofiscais de médio prazo. Nesses RAFs, costuma-se apresentar trajetórias prováveis para a dívida pública, geradas com essa ferramenta.

Esta nota não avalia o caso concreto da situação fiscal do Brasil na conjuntura atual, cumprindo estritamente o papel de divulgar uma ferramenta de análise da IFI.

¹ Analista da IFI.

Sumário

1 Introdução	5
2 Guia de uso	6
2.1 Informações preliminares	6
2.2 Instalação	7
2.3 Carregar dados econômicos no R	8
2.3.1 Exemplo 1: dados ilustrativos contidos no pacote	8
2.3.2 Exemplo 2: importando dados externos fornecidos pelo(a) usuário(a)	9
2.4 Gerar choques aleatórios	11
2.4.1 Gerar gráficos dos choques	16
2.5 Simular cenários estocásticos	19
2.5.1 Gerar gráficos dos cenários estocásticos	22
2.6 Estimar probabilidades	26
2.7 Exportar tabelas e <i>fan charts</i> para um arquivo Excel externo	27
3 Detalhes adicionais	28
3.1 Equação de dinâmica da dívida	28
3.2 Checagem inicial sobre o Python	29
3.3 Uso de <i>namespaces</i> para evitar erros	30
3.4 Leitura do código-fonte	30
3.5 Possibilidades de extensão e aprimoramento do código	30
3.5.1 Classes e métodos	31
3.5.2 Eficiência	31
Referências	32
Apêndice A	33
A.1 Windows (arquivo <code>.zip</code>)	33
A.2 Linux (arquivo <code>.tar.gz</code>)	33

1 Introdução

Nos anos de 2023 e 2024, a Instituição Fiscal Independente (IFI) publicou dois trabalhos sobre os cenários estocásticos para a dívida pública. O Estudo Especial nº 18 (Casalecchi, 2023) apresentou a metodologia de forma **geral**, e a Nota Técnica nº 54 (Casalecchi, 2024) realizou testes de **sensibilidade** sobre tal metodologia. A presente nota técnica é a terceira contribuição dentro deste tema, disponibilizando para a sociedade o **pacote** computacional, na linguagem R, utilizado pela instituição para gerar os cenários estocásticos. Ele é denominado `stocdebt` e pode ser obtido em: <https://github.com/ifibrasil/stocdebt>.

Esta nota não avalia o caso concreto da situação fiscal do Brasil na conjuntura atual, cumprindo estritamente o papel de divulgar uma ferramenta de análise da IFI.

A transparência das IFIs é tratada em quatro dos 22 “Princípios para Instituições Fiscais Independentes” (OCDE, 2014, pg. 4), documento que reúne recomendações de boas práticas para as IFIs de todos os países. Por exemplo, o princípio 7.1 afirma que a transparência dos trabalhos de uma IFI contribui (i) para a independência da instituição e (ii) para a credibilidade de suas análises frente à sociedade.

Entre outras IFIs que publicam seus códigos, temos o *Office for Budget Responsibility* (OBR), do Reino Unido,² e o *Congressional Budget Office* (CBO), dos Estados Unidos.³ Interessante notar que o código do OBR, em particular, é mantido em conjunto com o Tesouro britânico, porém preservando a autonomia do OBR, que pode utilizar premissas próprias no modelo.

Os cenários estocásticos produzidos pelo pacote `stocdebt` são publicados regularmente pela IFI brasileira, com frequência semestral, em edições especiais do Relatório de Acompanhamento Fiscal (RAF). Tais edições são dedicadas à revisão dos cenários macrofiscais de médio prazo (10 anos à frente) e, para tanto, atualizam as projeções da IFI de forma ampla, para muitas variáveis macroeconômicas e fiscais relevantes: PIB, inflação, resultado primário, receitas, despesas, dívida pública, etc. É nesse contexto que se costuma (i) estimar a probabilidade de a Dívida Bruta do Governo Geral (DBGG) cruzar limiares de referência, como 90% do PIB, e (ii) apresentar visualmente a incerteza que cerca as trajetórias determinísticas da dívida, por meio de gráficos do tipo “fan chart”.⁴ Esses cálculos são elaborados pelo pacote `stocdebt`, ora publicado.

Conforme explicado em Casalecchi (2023), a combinação dos cenários determinísticos e estocásticos, produzidos pela IFI, formam a ferramenta usualmente chamada de “análise de sustentabilidade da dívida” nos documentos internacionais – ou *Debt Sustainability Analysis* (DSA), em inglês. Entre os principais relatórios em que esse tipo de ferramenta é usada para análises de médio prazo está o *Debt Sustainability Monitor* (CE, 2025), elaborado

² Na linguagem do programa Eviews: <https://obr.uk/forecasts-in-depth/obr-macroeconomic-model/>.

³ Na linguagem R: <https://www.cbo.gov/publication/59629>.

⁴ Confira, por exemplo, o RAF nº 101 (IFI, 2025, pg. 45).

anualmente pela Comissão Europeia para acompanhar a situação fiscal dos membros da União Europeia. Portanto, pode-se afirmar que o pacote `stocdebt` gera a parte probabilística do DSA da IFI brasileira.

A divulgação do pacote também contribui para a cooperação internacional. IFIs de outros países, inclusive aquelas em processo de formação, passam a contar com essa ferramenta imediatamente. Ademais, tanto essas IFIs quanto aquelas estabelecidas há mais tempo poderão contribuir para o contínuo desenvolvimento do código, que é aberto, em benefício da IFI brasileira.

Em diversas oportunidades, a IFI manteve contato bilateral com as IFIs de Portugal,⁵ Áustria,⁶ Estados Unidos,⁷ Itália,⁸ Espanha,⁹ África do Sul,¹⁰ e Eslováquia¹¹. Além disso, a IFI mantém contatos multilaterais, por meio da rede de IFIs mantida pela OCDE.¹²

A próxima seção consiste em um guia de uso, em português. A funcionalidade demonstrada é exemplificativa, e não exaustiva. O pacote, uma vez instalado, contém documentação adicional bastante detalhada¹³, em inglês, como é padrão nesse tipo de pacote.

2 Guia de uso

2.1 Informações preliminares

O código-fonte do pacote pode ser consultado em seu repositório online: <https://github.com/ifibrasil/stocdebt>.

Sugestões de melhoria podem ser submetidas para o e-mail da IFI (ifi@senado.leg.br) ou no próprio repositório online.

O pacote é denominado `stocdebt`, sendo útil para

- simular cenários estocásticos para a dívida pública e seus determinantes: crescimento do PIB, inflação, resultado primário e taxa de juros implícita da dívida;
- estimar a probabilidade de eventos relacionados à dívida pública.

É possível exportar gráficos para Excel (como os *fan charts*). Neste caso, é preciso que outro programa, o Python, esteja instalado no computador¹⁴. Na elaboração do `stocdebt`,

⁵ Conselho das Finanças Públicas: <https://www.cfp.pt/pt/>.

⁶ Fiskalrat: <https://www.fiskalrat.at/en/>.

⁷ Congressional Budget Office: <https://www.cbo.gov/>.

⁸ Ufficio Parlamentare di Bilancio: <https://www.upbilancio.it/>.

⁹ Autoridad Independiente de Responsabilidad Fiscal: <https://www.airef.es/es/>.

¹⁰ Parliamentary Budget Office: <https://www.parliament.gov.za/parliamentary-budget-office>.

¹¹ Council for Budget Responsibility: <https://www.rzz.sk/en/>.

¹² Working Party of Parliamentary Budget Officials and Independent Fiscal Institutions

<https://www.oecd.org/en/topics/parliamentary-budget-offices-and-independent-fiscal-institutions.html>.

¹³ Na “vignette” e nos *help files* que o acompanham.

¹⁴ O Python está disponível através do link: <https://www.python.org/>.

recorreu-se ao Python para a exportação de gráficos porque ele tem maior capacidade de editar arquivos de Excel do que o R.

2.2 Instalação

Para instalar o pacote `stocdebt` no computador, diretamente do repositório online¹⁵, execute

```
# Instalar o pacote auxiliar "remotes"
install.packages("remotes")

# Instalar o pacote "stocdebt"
remotes::install_github("ifibrasil/stocdebt/stocdebt@<VERSÃO>",
build_vignettes = TRUE, upgrade = "never")
```

O trecho `<VERSÃO>` deve ser substituído por um dos números de versão **estável** do pacote, que constam na lista de “Versões” (ou “Releases”) do repositório (painele direito do site). Por exemplo `ifibrasil/stocdebt/stocdebt@0.2.0`. Caso `@<VERSÃO>` seja omitido do comando, possivelmente será instalada uma versão em **desenvolvimento** do pacote, que está sujeita a falhas.

No momento da publicação desta NT, a versão estável mais recente é a `0.2.0`, mas a numeração mudará conforme o tempo passar.

Se o(a) usuário(a) já instalou o pacote e deseja verificar a versão, basta executar

```
utils::packageDescription("stocdebt")$RemoteRef
```

Se o resultado for `<VERSÃO>` (atualmente `“0.2.0”`), a instalação está correta. É a versão estável mais recente. Se o resultado for `“HEAD”`, trata-se da versão de desenvolvimento.

O argumento `build_vignettes = TRUE` é opcional. Se o(a) usuário(a) não utilizá-lo, o pacote será instalado sem um guia de usuário (“User guide”). Apenas a documentação padrão (*help files*) estará disponível.

O argumento `upgrade = “never”` também é opcional, mas simplifica o processo de instalação.

Finalizada a instalação, carregue o pacote:

```
library(stocdebt)
```

¹⁵ Se não for possível instalar diretamente do repositório online, veja alternativas no [apêndice](#) desta nota.

Para acessar o índice de funções disponíveis, abra a página inicial de ajuda do pacote e clique em *Index*, ao final da página.

```
# Página inicial de ajuda do pacote  
?stocdebt
```

Para acessar explicações detalhadas sobre cada função, execute, por exemplo

```
# Ajuda da função "sim()  
?sim
```

2.3 Carregar dados econômicos no R

O pacote vem com dados ilustrativos (objeto `sample_series`) que podem ser usados para demonstrar o uso dos comandos, como no Exemplo 1, abaixo. Alternativamente, o(a) usuário(a) pode carregar dados próprios, salvos em um arquivo `xlsx` (Excel) externo ao R (Exemplo 2).

Os dados em `sample_series` são uma combinação de dados públicos do Banco Central e do IBGE, e de dados hipotéticos.

2.3.1 Exemplo 1: dados ilustrativos contidos no pacote

O objeto `sample_series` é da classe `list`. Essa lista contém, por sua vez, quatro objetos da classe `xts`:

```
# Dados realizados, em frequência anual, para a dívida e seus determinantes  
sample_series$realized_yearly_data  
  
# Dados realizados, em frequência trimestral, para a dívida e seus determinantes  
sample_series$realized_quarterly_data  
  
# Projeções determinísticas que formam um cenário base, em frequência anual,  
# para a dívida e seus determinantes  
sample_series$baseline_yearly_scenario  
  
# Dados de ajustes de estoque-fluxo ("stock-flow adjustments", ou "sfa") para  
# a dívida  
sample_series$realized_yearly_sfa_data
```

Por exemplo, as primeiras linhas dos dados realizados em frequência anual são as seguintes.

```
# Dados realizados, em frequência anual, para a dívida e seus determinantes
sample_series$realized_yearly_data

  real interest rate (w.r.t. gdp deflator)
2007-12-01                      6.1189900
2008-12-01                      3.9973603
2009-12-01                      3.2394612
2010-12-01                      2.6790455
  real interest rate (w.r.t. consumer price index)  real gdp growth
2007-12-01                      8.092563          6.1
2008-12-01                      6.842924          5.1
2009-12-01                      6.196710         -0.1
2010-12-01                      5.094009          7.5
  consumer price index gdp deflator  debt primary balance
2007-12-01                      4.457330          6.4 56.72          3.29
2008-12-01                      5.902313          8.8 55.98          3.27
2009-12-01                      4.312028          7.3 59.21          1.90
2010-12-01                      5.909068          8.4 51.77          2.56
  nominal interest rate
2007-12-01                      12.910605
2008-12-01                      13.149128
2009-12-01                      10.775942
2010-12-01                      11.304085
```

2.3.2 Exemplo 2: importando dados externos fornecidos pelo(a) usuário(a)

Suponha que o(a) usuário(a) tenha um arquivo Excel chamado `my_data.xlsx`, com dois tipos de dados:

- séries históricas **realizadas**, em frequência anual, de 2013 a 2023, em uma planilha (“aba”) chamada `realized_data`;
- um **cenário base** determinístico, de 2024 a 2033, em uma outra planilha no mesmo arquivo, chamada `baseline_scenario`.

Dentro do arquivo, os dados devem necessariamente estar dispostos como nas Tabelas 1 e 2, abaixo, com exceção da primeira coluna, que contém as datas. Os nomes das colunas são de escolha do(a) usuário(a). Neste exemplo, estão em inglês, para manter coerência com os nomes usados no pacote, que está nesse idioma.

Não é preciso que a primeira coluna contenha os anos, porque as datas de início e fim serão informadas adiante, explicitamente, como argumentos da função `import_custom_data()`.

TABELA 1. EXEMPLO DE DADOS REALIZADOS (2013-2023) CONTIDOS EM UM ARQUIVO EXCEL EXTERNO

Year	primary balance	nominal gdp growth	real gdp growth	gdp deflator	consumer price index	nominal interest rate	real interest rate (w.r.t. GDP deflator)	debt	real interest rate (w.r.t. consumer price index)
2013	1,7	10,7	3,0	7,5	5,9	10,6	4,4	51,5	4,4
2014	-0,6	8,4	0,5	7,8	6,4	11,1	4,4	56,3	4,4
2015	-1,9	3,8	-3,5	7,6	10,7	13,2	2,3	65,5	2,3
2016	-2,5	4,6	-3,3	8,1	6,3	13,1	6,4	69,8	6,4
2017	-1,7	5,0	1,3	3,7	2,9	9,9	6,8	73,7	6,8
2018	-1,5	6,4	1,8	4,5	3,7	8,3	4,4	75,3	4,4
2019	-0,8	5,5	1,2	4,2	4,3	7,8	3,3	74,4	3,3
2020	-9,2	3,0	-3,3	6,5	4,5	5,9	1,3	86,9	1,3
2021	0,7	18,4	4,8	13,0	10,1	7,6	-2,2	77,3	-2,2
2022	1,2	11,8	3,0	8,6	5,8	10,8	4,8	71,7	4,8
2023	-2,3	7,8	3,0	4,7	4,6	11,3	6,4	74,3	6,4

Elaboração: IFI, a partir de dados públicos do Banco Central e do IBGE, e de dados hipotéticos.

Nota 1: "w.r.t." significa "with respect to", isto é, "em relação a".

Nota 2: os valores são ilustrativos. Eles podem diferir daqueles usados em outras partes deste texto.

TABELA 2. EXEMPLO DE CENÁRIO BASE DETERMINÍSTICO (2024-2033) CONTIDO EM UM ARQUIVO EXCEL EXTERNO

Year	primary balance	nominal gdp growth	real gdp growth	gdp deflator	consumer price index	nominal interest rate	real interest rate (w.r.t. GDP deflator)	debt	real interest rate (w.r.t. consumer price index)
2024	-0,8	6,2	1,6	4,4	3,9	9,7	5,6	77,7	5,6
2025	-0,8	6,2	2,0	4,2	3,5	8,7	5,0	80,2	5,0
2026	-0,4	6,1	2,0	4,0	3,2	7,9	4,5	81,9	4,5
2027	0,0	5,9	2,0	3,8	3,0	7,2	4,1	82,9	4,1
2028	0,3	5,8	2,0	3,7	3,0	7,0	3,9	83,6	3,9
2029	0,6	5,7	2,0	3,7	3,0	7,0	3,9	83,9	3,9
2030	0,8	5,8	2,0	3,7	3,0	7,0	3,9	84,1	3,9
2031	0,9	5,7	2,0	3,7	3,0	7,0	3,9	84,3	3,9
2032	1,0	5,7	2,0	3,7	3,0	7,0	3,9	84,3	3,9
2033	1,1	5,8	2,0	3,7	3,0	7,0	3,9	84,2	3,9

Elaboração: IFI.

Nota 1: "w.r.t." significa "with respect to", isto é, "em relação a".

Nota 2: os valores são ilustrativos. Eles podem diferir daqueles usados em outras partes deste texto.

Para importar os dados do Excel para o R, salve o arquivo `my_data.xlsx` no `working directory` do R e execute o comando abaixo. A função retornará um objeto da classe `xts`.

```

i <- import_custom_data(
  realized_data = list(
    file_type = "xlsx",
    file_path = "my_data.xlsx",
    sheet = "realized_data",
    start = "2013-12-01",
    end = "2023-12-01",
    frequency = "year"
  ),
  baseline_scenario = list(
    file_type = "xlsx",
    file_path = "my_data.xlsx",
    sheet = "baseline_scenario",
    start = "2024-12-01",
    end = "2033-12-01",
    frequency = "year"
  )
)

```

2.4 Gerar choques aleatórios

Desta seção em diante, serão adotados os dados ilustrativos que acompanham o pacote (objeto `sample_series` do exemplo 1).

Para gerar choques, pode-se escolher entre os casos `normal` (isto é, distribuição normal de probabilidade)¹⁶ ou `ec` (sigla para “European Commission”, pois implementa um método inspirado na estratégia da Comissão Europeia^{17 18}). O comando a seguir usa o caso `ec` para gerar 1.000 choques por ano, para cada determinante da dívida (taxa de juros nominal implícita, deflator do PIB, crescimento real do PIB e resultado primário).

¹⁶ Trata-se de uma distribuição normal para cada determinante da dívida, com média zero e desvio padrão informado pelo(a) usuário(a). Essas distribuições são independentes entre si.

¹⁷ Para detalhes, confira o EE nº 18 (Casalecchi, 2023, pg. 41, seção III): “A IFI adotou metodologia baseada em (...) *Debt Sustainability Monitor* – por duas razões. Primeiro, porque o procedimento pertence a um padrão amplamente aceito, no qual choques são sorteados e aplicados sobre um cenário base determinístico, que a IFI já produz. Segundo, porque é de mais fácil implementação e explicação para o público, relativamente às outras metodologias de cenários estocásticos.”

¹⁸ A versão mais atual do método da Comissão Europeia pode ser consultado em CE (2025, pg. 259, Anexo A4).

```

shocks <- shocks_generator(
  case = "ec",
  n_stochastic_scenarios = 1000,
  realized_data = sample_series$realized_yearly_data,
  subintervals = "2009-12-01/2019-12-01",
  shocks_first_date = "2024-12-01",
  shocks_last_date = "2028-12-01",
  shocks_frequency = "year",
  i = "nominal interest rate",
  ti = "gdp deflator",
  g = "real gdp growth",
  p = "primary balance"
)

```

Ao atribuir `subintervals = "2009-12-01/2019-12-01"`, apenas observações de antes da pandemia de Covid-19 (2020) serão usadas, evitando-se observações sujeitas a alterações extremas. Outros subintervalos podem ser escolhidos livremente pelo(a) usuário(a).¹⁹

Vale dizer que o(a) usuário(a) pode determinar quais variáveis, entre aquelas presentes no argumento `realized_data`, são a taxa de juros nominais, o crescimento do PIB, etc. Essa determinação é feita por meio dos argumentos `i`, `ti`, `g` e `p`, onde são informados os nomes das colunas de `realized_data` correspondentes, respectivamente, (a) à taxa de juros nominal, (b) ao deflator do PIB, (c) ao crescimento real do PIB e (d) ao resultado primário.

A função `shocks_generator()` retorna uma `list` que contém quatro objetos:

- `shocks`: os **choques propriamente ditos**. Neste exemplo, é uma `list` com 1.000 objetos `xts`, sendo que cada um desses `xts` possui quatro colunas (uma para cada determinante da dívida);
- `historical_shocks`: as **séries temporais históricas** de onde os choques, aplicados sobre os determinantes da dívida, foram sorteados. Neste exemplo, onde se usou `case = ec`, trata-se simplesmente das primeiras diferenças das séries temporais realizadas desses determinantes;
- `var_cov_historical_shocks`: a matriz de **variância-covariância** das séries históricas dos choques aplicados sobre os determinantes da dívida;

¹⁹ Também é possível especificar um conjunto de subintervalos não contíguos. Por exemplo, `subintervals = c("2011-03-01/2014-12-01", "2017-03-01/2019-03-01")`. Além disso, note que, se `case = "ec"`, as datas em `subintervals` serão aquelas das **primeiras diferenças** dos determinantes da dívida, não de seus **níveis**. Neste último exemplo, os choques incluiriam a diferença entre valores nas datas 2017-03-01 e 2016-12-01, ainda que a segunda data não esteja explícita em `subintervals`.

- `correlation_tests`: os valores “p” de **testes de significância das correlações** entre os choques. Para os testes, são usados os procedimentos de Pearson, Spearman e Kendall.

Explicitando-se os quatro objetos:

```
# O primeiro grupo de choques, entre os 1.000 grupos (objetos "xts")
shocks$shocks[[1]]
```

	nominal	interest	rate	gdp	deflator	real	gdp	growth	primary	balance
2024-12-01	-0.4769958			-2.1680918		2.2288076		0.6342605		
2025-12-01	0.9962942			0.3937421		-0.5893138		-0.1489300		
2026-12-01	0.3610963			1.3059682		3.2194927		1.0578514		
2027-12-01	0.0439360			2.5631020		-6.0649486		-0.7784888		
2028-12-01	0.3594061			-1.0269495		-1.3775189		0.3472275		

```
# Choques históricos (primeiras diferenças das séries históricas)
shocks$historical_shocks
```

	nominal	interest	rate	gdp	deflator	real	gdp	growth	primary	balance
2009-12-01	-2.37318614			-1.5		-5.2		-1.37		
2010-12-01	0.52814342			1.1		7.6		0.66		
2011-12-01	1.14743512			-0.1		-3.5		0.33		
2012-12-01	-1.73732750			-0.4		-2.1		-0.66		
2013-12-01	-0.09841795			-0.4		1.1		-0.52		
2014-12-01	0.47212318			0.3		-2.5		-2.20		
2015-12-01	2.10812035			-0.2		-4.0		-1.29		
2016-12-01	-0.11272698			0.5		0.2		-0.69		
2017-12-01	-3.14127559			-4.3		4.6		0.78		
2018-12-01	-1.67353300			0.6		0.5		0.08		
2019-12-01	-0.51384751			-0.1		-0.6		0.61		

Por padrão, sempre que `shocks_generator()` for utilizada com os mesmos argumentos, serão produzidos exatamente os mesmo valores numéricos. Caso se deseje produzir resultados diferentes, deve-se alterar o valor do argumento `shocks_seed` (não demonstrado no código acima), cujo valor *default* é 1.

```
# Matriz de variância-covariância dos choques
shocks$var_cov_historical_shocks

nominal interest rate gdp deflator real gdp growth
nominal interest rate 2.5222187 1.4771853 -1.0647690
gdp deflator 1.4771853 2.1189091 -0.3955455
real gdp growth -1.0647690 -0.3955455 14.5347273
primary balance -0.3495626 -0.2640818 2.4069091
primary balance
nominal interest rate -0.3495626
gdp deflator -0.2640818
real gdp growth 2.4069091
primary balance 0.9436964
```

```
# Valores "p" de testes de correlação (Pearson, Spearman e Kendall)
shocks$correlation_tests
```

```
$spearson
```

	nominal interest rate	gdp deflator	real gdp growth	gdp growth
nominal interest rate	NA	0.03430654	0.60500071	
gdp deflator	0.03430654	NA	0.83503681	
real gdp growth	0.60500071	0.83503681	NA	
primary balance	0.50288014	0.58243588	0.03042334	
	primary balance			
nominal interest rate	0.50288014			
gdp deflator	0.58243588			
real gdp growth	0.03042334			
primary balance	NA			

```
$sspearman
```

	nominal interest rate	gdp deflator	real gdp growth	gdp growth
nominal interest rate	NA	0.1087939	0.61154238	
gdp deflator	0.1087939	NA	0.44896813	
real gdp growth	0.6115424	0.4489681	NA	
primary balance	0.6695786	0.8626730	0.01647098	
	primary balance			
nominal interest rate	0.66957865			
gdp deflator	0.86267305			
real gdp growth	0.01647098			
primary balance	NA			

```
$kendall
```

	nominal interest rate	gdp deflator	real gdp growth	gdp growth
nominal interest rate	NA	0.1183498	0.58578841	
gdp deflator	0.1183498	NA	0.27429883	
real gdp growth	0.5857884	0.2742988	NA	
primary balance	0.6970916	0.7547764	0.02396768	
	primary balance			
nominal interest rate	0.69709161			
gdp deflator	0.75477643			
real gdp growth	0.02396768			
primary balance	NA			

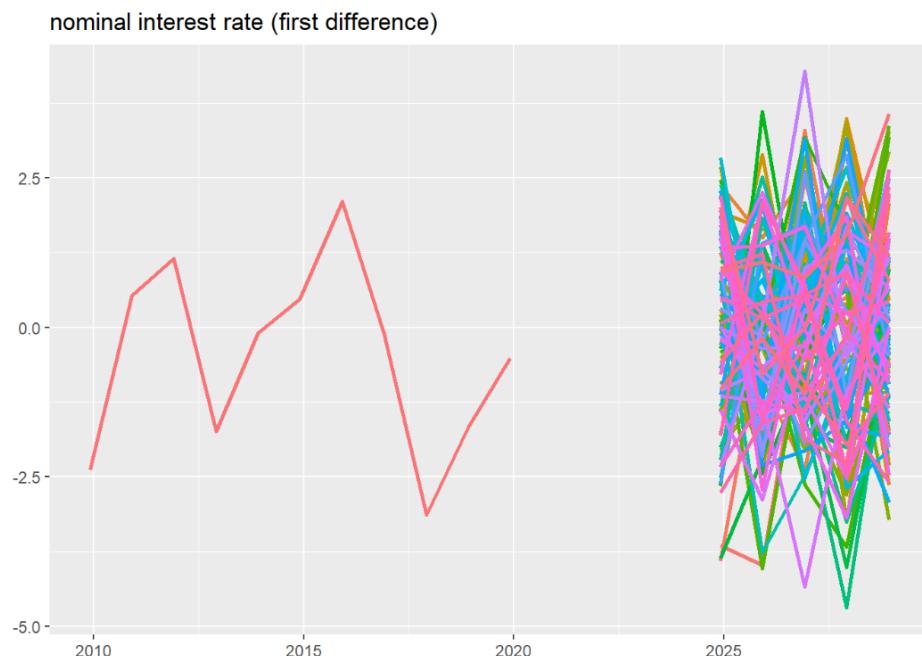
2.4.1 Gerar gráficos dos choques

Pode-se criar gráficos dos choques, para fins de diagnóstico visual. A função `charts()` cria um gráfico por variável.

```
x <- charts(object = shocks, type = "shocks")
```

Para visualizar os gráficos **não interativos**,²⁰ basta acessá-los como subitens do objeto (da classe `list`) retornado pela função, conforme abaixo.

```
# Gráfico não interativo dos choques na taxa de juros nominal
x$list_of_charts$shocks$ggplot[["nominal interest rate"]]
```



Como os choques foram gerados usando-se a estratégia da Comissão Europeia, isto é, `shocks_generator(case = "ec", ...)`, em cada gráfico há:

- à esquerda, a série histórica de primeiras diferenças (de onde os 1.000 choques foram sorteados);
- à direita, trajetórias de choques sorteados (no caso, 100 trajetórias);
- ao meio, uma área vazia, correspondente aos valores históricos descartados da amostra usada no sorteio dos choques, conforme escolha acima.²¹

Se o(a) usuário(a) especificar `subintervals` (na função `shocks_generator`) de maneira diferente, outras áreas do gráfico aparecerão vazias. Note que esse argumento é

²⁰ Construídos com o pacote `ggplot2`, que é uma das dependências do `stocdebt`. As dependências são instaladas automaticamente, quando se instala o `stocdebt` conforme indicado na seção 2.2.

²¹ Afinal, no comando `shocks_generator(...)` mais acima, optou-se por `subintervals = "2009-12-01/2019-12-01"`, levando ao descarte das observações a partir de 2020.

um dos mais importantes a serem especificados em toda a simulação, conforme explicado na NT nº 54,²² pois tem grande impacto sobre os resultados numéricos finais dos cenários estocásticos.

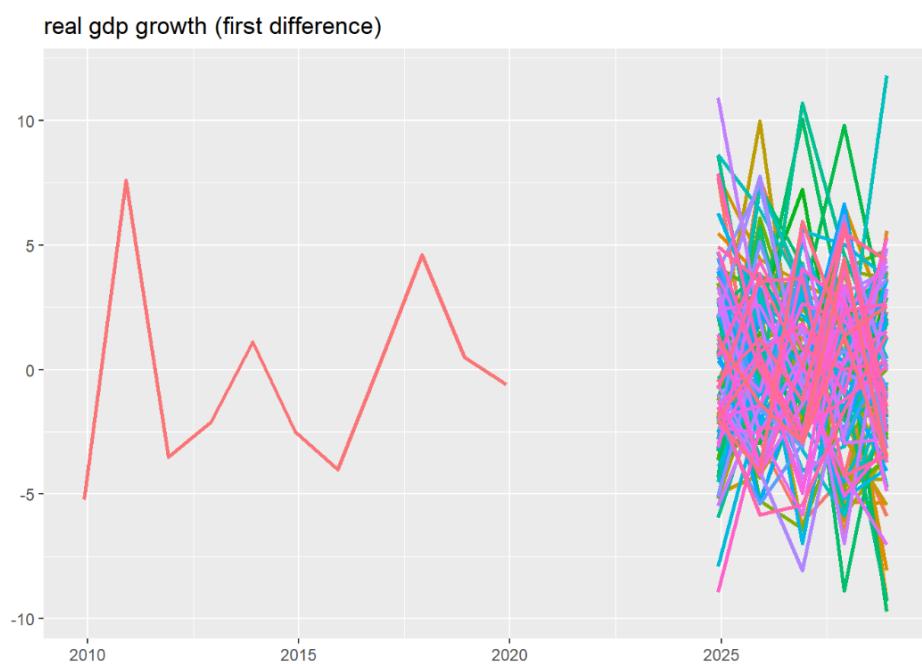
À primeira vista, a trajetória à esquerda parece ter assumido um valor em 2020. Entretanto, trata-se apenas de um efeito visual decorrente da escala do gráfico, que está em meses. A última observação histórica, na verdade, se refere a dezembro de 2019.

Note que a função `charts()` elabora um gráfico contendo apenas 100 trajetórias de choques (e não todas as 1.000 trajetórias), para evitar excesso de demanda computacional. A quantidade de trajetórias no gráfico pode ser definida no argumento `max_to_plot`, cujo valor padrão (`default`) é 100.

A escolha de quais 100 trajetórias (ou `max_to_plot` trajetórias), entre as 1.000, serão exportadas, é feita de forma aleatória. Sempre que o(a) usuário(a) executar a função com os mesmos argumentos, as mesmas trajetórias serão exportadas. Caso se deseje uma nova seleção aleatória das 100 trajetórias, deve-se modificar o argumento `seed_if_limited_export` (não demonstrada no código acima), cujo valor `default` é 1.

A seguir, o gráfico dos choques a serem aplicados sobre outra variável, o crescimento real do PIB.

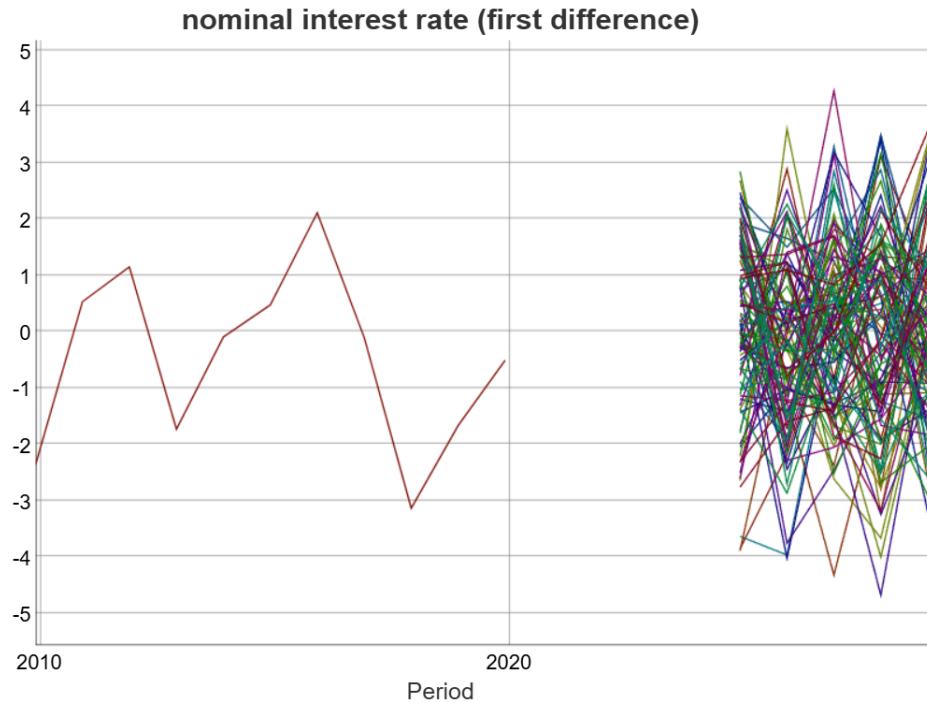
```
# Gráfico não interativo dos choques no crescimento real do PIB
x$list_of_charts$shocks$ggplot[["real gdp growth"]]
```



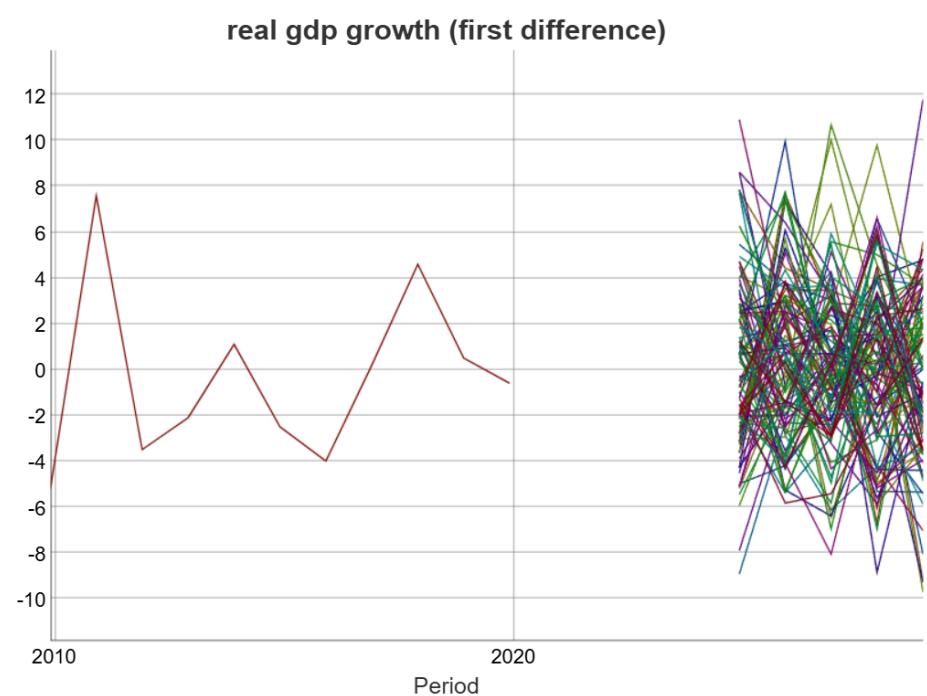
²² “Dentre os testes de sensibilidade implementados, a nota mostrou que a decisão metodológica mais relevante que a(o) usuário(a) deve tomar, ao fazer uso da ferramenta de simulação, é a escolha do período histórico que servirá de subamostra para sorteio dos choques.” (Casalecchi, 2024, pg. 18).

Para gráficos **interativos**,²³ execute:

```
# Gráfico interativo dos choques na taxa de juros nominal
x$list_of_charts$shocks$dygraphs[["nominal interest rate"]]
```



```
# Gráfico interativo dos choques no crescimento real do PIB
x$list_of_charts$shocks$dygraphs[["real gdp growth"]]
```



²³ Construídos com o pacote dygraphs.

2.5 Simular cenários estocásticos

Usando-se os choques gerados acima (shocks), pode-se finalmente simular 1.000 cenários estocásticos para cada determinante da dívida e, também, para a dívida em si.

```
s <- sim(
  first_date_to_simulate = "2024-12-01",
  last_date_to_simulate = "2028-12-01",
  realized_data = sample_series$realized_yearly_data,
  baseline_scenario = sample_series$baseline_yearly_scenario,
  baseline_as_median = TRUE,
  n_stochastic_scenarios = 1000,
  shocks = shocks$shocks,
  d = "debt",
  i = "nominal interest rate",
  ti = "gdp deflator",
  r = "real interest rate (w.r.t. gdp deflator)",
  g = "real gdp growth",
  p = "primary balance"
)
```

Após a simulação, pode-se verificar o **primeiro cenário estocástico simulado** (por exemplo). Ele já está empilhado com os valores realizados, de modo a formar séries temporais completas, da primeira data realizada (passado) à última data simulada (futuro):

```
s$scenarios$stochastic_scenarios[[1]]
```

	debt	real interest rate (w.r.t. gdp deflator)
2007-12-01	56.72000	6.1189900
2008-12-01	55.98000	3.9973603
2009-12-01	59.21000	3.2394612
2010-12-01	51.77000	2.6790455
2011-12-01	51.27000	3.8333522
2012-12-01	53.67000	2.6081492
2013-12-01	51.54000	2.8983953
2014-12-01	56.28000	3.0499983
2015-12-01	65.50000	5.2007607
2016-12-01	69.84000	4.6098904
2017-12-01	73.72000	5.9171637
2018-12-01	75.27000	3.7054434
2019-12-01	74.44000	3.3122104
2020-12-01	86.94000	-0.6021673
2021-12-01	77.31000	-4.7836096
2022-12-01	71.68000	2.0635500
2023-12-01	73.83000	6.3232608
2024-12-01	77.04570	7.4475849
2025-12-01	80.38962	5.5510892
2026-12-01	77.69439	3.6222648
2027-12-01	82.67857	1.5678071
2028-12-01	85.32071	5.2863036

	nominal	interest	rate	gdp	deflator	real	gdp	growth	primary	balance
2007-12-01		12.910605		6.400000		6.1000000		3.2900000		
2008-12-01		13.149128		8.800000		5.1000000		3.2700000		
2009-12-01		10.775942		7.300000		-0.1000000		1.9000000		
2010-12-01		11.304085		8.400000		7.5000000		2.5600000		
2011-12-01		12.451520		8.300000		4.0000000		2.8900000		
2012-12-01		10.714193		7.900000		1.9000000		2.2300000		
2013-12-01		10.615775		7.500000		3.0000000		1.7100000		
2014-12-01		11.087898		7.800000		0.5000000		-0.4900000		
2015-12-01		13.196019		7.600000		-3.5000000		-1.7800000		
2016-12-01		13.083292		8.100000		-3.3000000		-2.4700000		
2017-12-01		9.942016		3.800000		1.3000000		-1.6900000		
2018-12-01		8.268483		4.400000		1.8000000		-1.6100000		
2019-12-01		7.754635		4.300000		1.2000000		-1.0000000		
2020-12-01		5.858692		6.500000		-3.3000000		-9.2900000		
2021-12-01		7.594521		13.000000		4.8000000		0.6900000		
2022-12-01		10.841015		8.600000		3.0000000		1.2000000		
2023-12-01		11.320454		4.700000		3.2000000		-2.2500000		
2024-12-01		9.292990		2.153408		3.6201813		-0.2488523		
2025-12-01		9.688023		4.655882		1.3865784		-0.8684845		
2026-12-01		8.256946		5.336826		5.3290487		0.6217973		
2027-12-01		7.156503		6.337253		-4.3282603		-0.8530582		
2028-12-01		7.426774		2.737108		0.5158876		0.6669574		

Para ver o **cenário base determinístico** empilhado com os valores realizados, execute:

	debt	nominal	interest	rate	gdp	deflator
2007-12-01	56.72000		12.910605		6.400000	
2008-12-01	55.98000		13.149128		8.800000	
2009-12-01	59.21000		10.775942		7.300000	
2010-12-01	51.77000		11.304085		8.400000	
2011-12-01	51.27000		12.451520		8.300000	
2012-12-01	53.67000		10.714193		7.900000	
2013-12-01	51.54000		10.615775		7.500000	
2014-12-01	56.28000		11.087898		7.800000	
2015-12-01	65.50000		13.196019		7.600000	
2016-12-01	69.84000		13.083292		8.100000	
2017-12-01	73.72000		9.942016		3.800000	
2018-12-01	75.27000		8.268483		4.400000	
2019-12-01	74.44000		7.754635		4.300000	
2020-12-01	86.94000		5.858692		6.500000	
2021-12-01	77.31000		7.594521		13.000000	
2022-12-01	71.68000		10.841015		8.600000	
2023-12-01	73.83000		11.320454		4.700000	
2024-12-01	77.65987		9.733254		4.435770	
2025-12-01	80.18540		8.673754		4.181852	
2026-12-01	81.89241		7.912894		4.006824	
2027-12-01	82.94203		7.206000		3.764094	
2028-12-01	83.55540		7.000000		3.666734	

	real interest rate (w.r.t. gdp deflator)	real gdp growth
2007-12-01	6.1189900	6.100000
2008-12-01	3.9973603	5.100000
2009-12-01	3.2394612	-0.100000
2010-12-01	2.6790455	7.500000
2011-12-01	3.8333522	4.000000
2012-12-01	2.6081492	1.900000
2013-12-01	2.8983953	3.000000
2014-12-01	3.0499983	0.500000
2015-12-01	5.2007607	-3.500000
2016-12-01	4.6098904	-3.300000
2017-12-01	5.9171637	1.300000
2018-12-01	3.7054434	1.800000
2019-12-01	3.3122104	1.200000
2020-12-01	-0.6021673	-3.300000
2021-12-01	-4.7836096	4.800000
2022-12-01	2.0635500	3.000000
2023-12-01	6.3232608	3.200000
2024-12-01	5.6334610	1.645013
2025-12-01	5.0073750	1.962431
2026-12-01	4.5430270	2.047656
2027-12-01	4.0834950	2.033197
2028-12-01	3.8834950	2.012767
	primary balance	
2007-12-01	3.29000000	
2008-12-01	3.27000000	
2009-12-01	1.90000000	
2010-12-01	2.56000000	
2011-12-01	2.89000000	
2012-12-01	2.23000000	
2013-12-01	1.71000000	
2014-12-01	-0.49000000	
2015-12-01	-1.78000000	
2016-12-01	-2.47000000	
2017-12-01	-1.69000000	
2018-12-01	-1.61000000	
2019-12-01	-1.00000000	
2020-12-01	-9.29000000	
2021-12-01	0.69000000	
2022-12-01	1.20000000	
2023-12-01	-2.25000000	
2024-12-01	-0.84635246	
2025-12-01	-0.76147409	
2026-12-01	-0.38942001	
2027-12-01	-0.04297054	
2028-12-01	0.34125810	

Note que a função `sim` funcionou somente porque as variáveis (argumentos `d`, `i`, etc.) possuem o mesmo nome nas três fontes de dados (argumentos `realized_data`, `baseline_scenario` e `shocks`). Se os nomes não fossem idênticos, a função deveria ter sido utilizada da seguinte forma, bem mais extensa e detalhada:

```

s <- sim(
  first_date_to_simulate = "2024-12-01",
  last_date_to_simulate = "2028-12-01",
  realized_data = sample_series$realized_yearly_data,
  baseline_scenario = sample_series$baseline_yearly_scenario,
  baseline_as_median = TRUE,
  n_stochastic_scenarios = 1000,
  shocks = shocks$shocks,
  d = list(
    realized = "debt",
    baseline = "debt"
  ),
  i = list(
    realized = "nominal interest rate",
    baseline = "nominal interest rate",
    shocks = "nominal interest rate"
  ),
  ti = list(
    realized = "gdp deflator",
    baseline = "gdp deflator",
    shocks = "gdp deflator"
  ),
  r = list(
    realized = "real interest rate (w.r.t. gdp deflator)",
    baseline = "real interest rate (w.r.t. gdp deflator)"
  ),
  g = list(
    realized = "real gdp growth",
    baseline = "real gdp growth",
    shocks = "real gdp growth"
  ),
  p = list(
    realized = "primary balance",
    baseline = "primary balance",
    shocks = "primary balance"
  )
)

```

Não seria preciso informar o nome dos choques da taxa de juros real (*r*) acima, pois ela é calculada a partir da taxa nominal (*i*) e do deflator do PIB (*ti*) internamente pela função.

2.5.1 Gerar gráficos dos cenários estocásticos

Para cada variável, dois gráficos de cenários estocásticos podem ser gerados:

- um gráfico com várias trajetórias estocásticas;
- um *fan chart*.

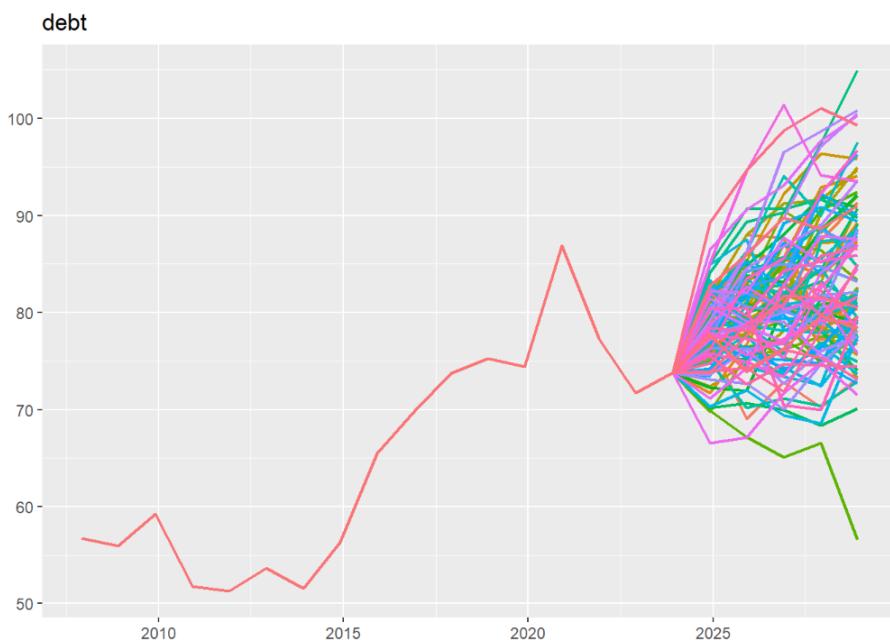
O primeiro contém apenas algumas trajetórias (não todas as 1.000), para evitar sobrecarga computacional. A quantidade de trajetórias é determinada pelo argumento `max_to_plot`, cujo valor padrão (*default*) é 100.

Os dois gráficos são gerados executando-se o comando:

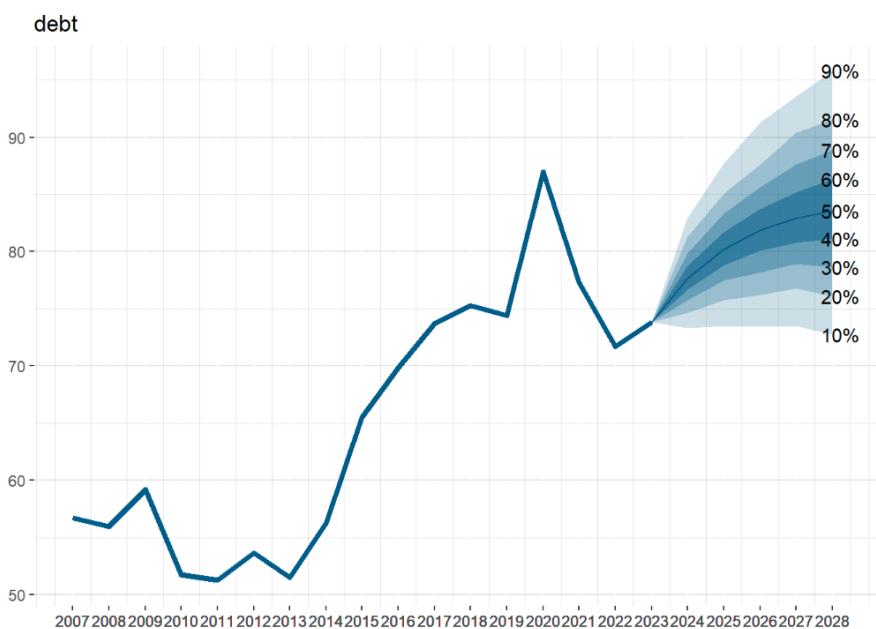
```
x <- charts(
  object = s,
  type = "simulation",
  fan_chart_percentiles = c(0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9)
)
```

Para acessar os gráficos **não interativos**:

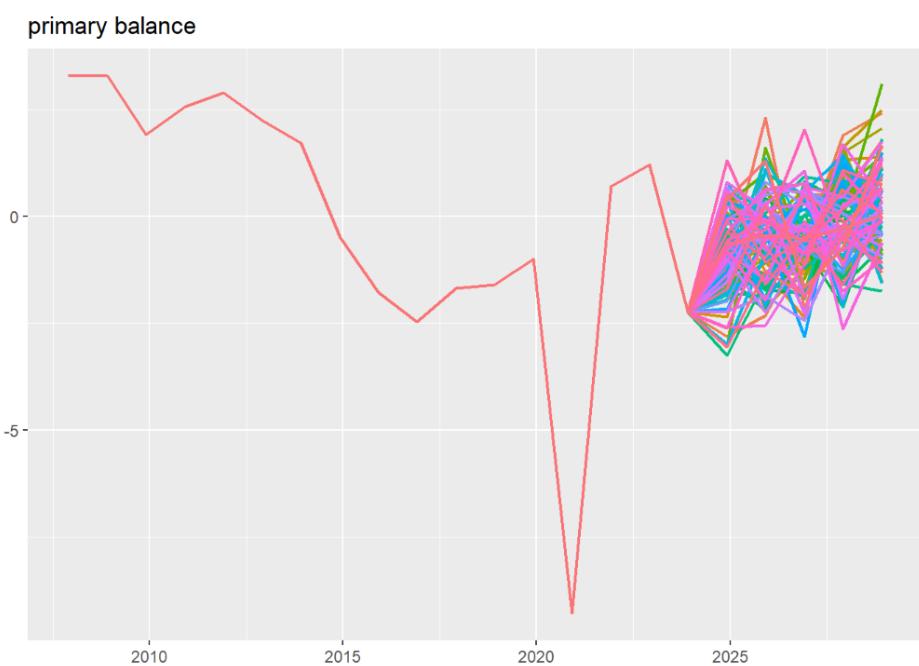
```
# Gráfico não interativo de trajetórias da dívida
x$list_of_charts$paths$ggplot[["debt"]]
```



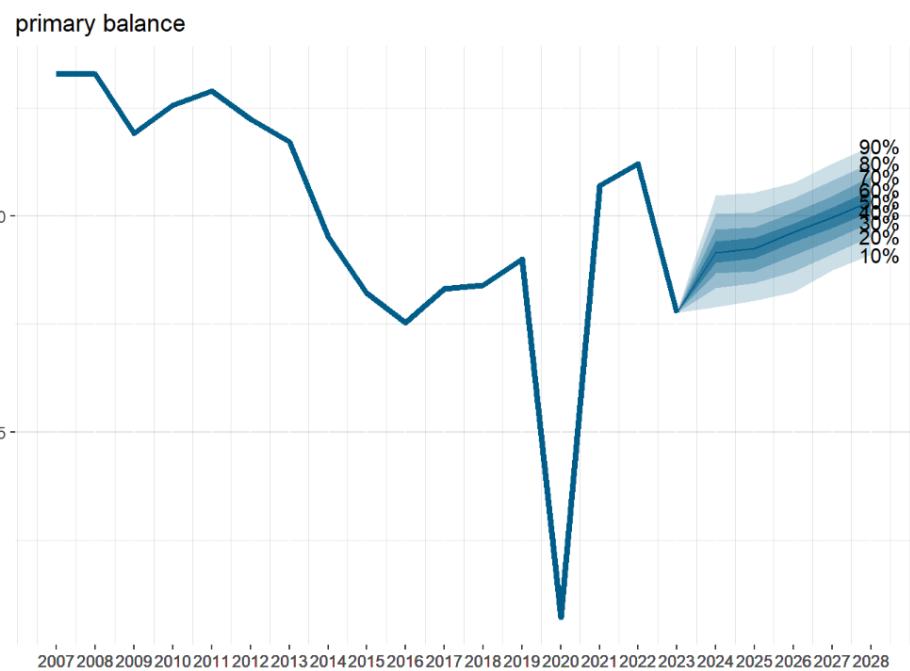
```
# Fan chart não interativo da dívida  
x$list_of_charts$fancharts$ggplot[["debt"]]
```



```
# Gráfico não interativo de trajetórias do resultado primário  
x$list_of_charts$paths$ggplot[["primary balance"]]
```

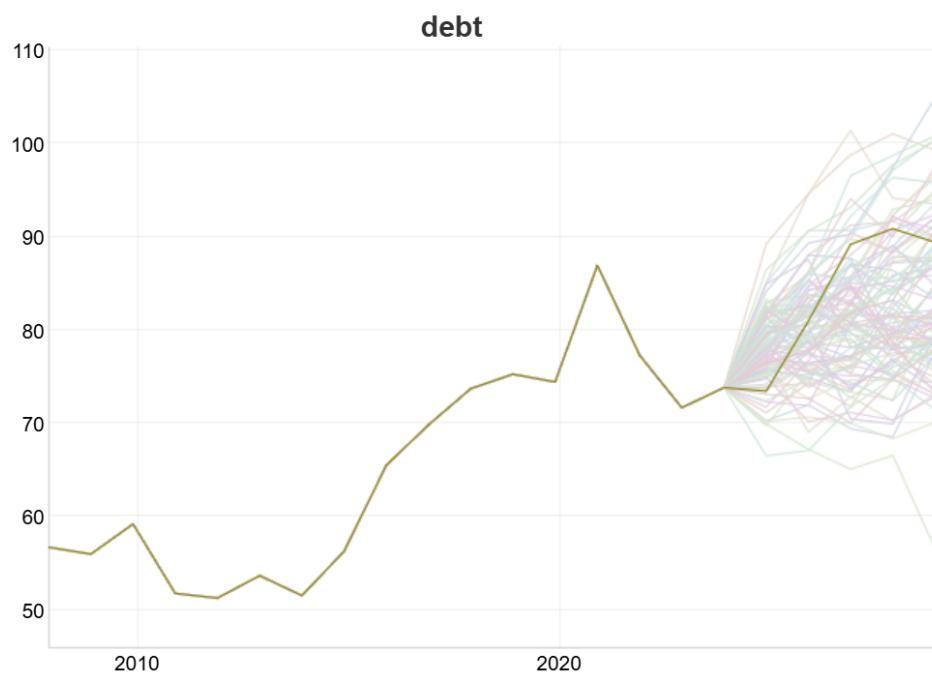


```
# Fan chart não interativo do resultado primário
x$list_of_charts$fancharts$ggplot[["primary balance"]]
```

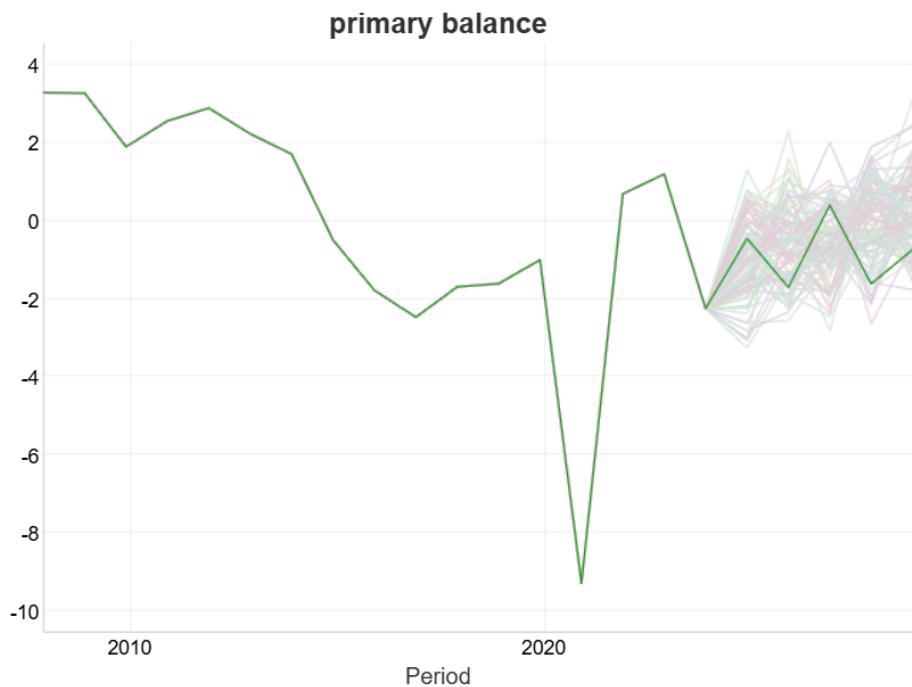


Para acessar os gráficos **interativos**, execute os comandos abaixo. As trajetórias realçadas nas imagens foram selecionadas passando-se o cursor (mouse) sobre elas.

```
# Gráfico interativo de trajetórias da dívida
x$list_of_charts$paths$dygraphs[["debt"]]
```



```
# Gráfico interativo de trajetórias do resultado primário
x$list_of_charts$pathss$dygraphs[["primary balance"]]
```



Fan charts interativos ainda não foram implementados no código.

2.6 Estimar probabilidades

A função `event_prob` estima a probabilidade de eventos futuros referentes à dívida pública. No exemplo abaixo, utiliza-se um limiar para a dívida, igual a 90% do PIB, a fim de se calcular probabilidades de interesse. Para maiores detalhes sobre a interpretação dessas probabilidades, inclusive com fórmulas, confira a documentação específica da função, que acompanha o pacote.

```
e <- event_prob(
  object = s,
  d = "debt",
  threshold = 90,
  interval = list(start="2024-12-01", end="2028-12-01")
)
```

Abaixo, os resultados retornados pela função.

```

e

$P
$p$`Probability that future debt will be above the threshold, at EACH future
period in the interval`  

  Estimated probability
2024-12-01          0.002
2025-12-01          0.048
2026-12-01          0.125
2027-12-01          0.213
2028-12-01          0.258

$p$`Probability that future debt will be above the threshold, at EVERY period
in the interval`  

[1] 0.002

$p$`Probability that future debt will be above the threshold, AT LEAST AT ONE
period in the interval`  

[1] 0.304

$p$`Probability that future debt will cross the threshold from below for the
FIRST TIME, at EACH future period in the interval`  

  Estimated probability
2024-12-01          0.002
2025-12-01          0.046
2026-12-01          0.085
2027-12-01          0.098
2028-12-01          0.073

```

2.7 Exportar tabelas e *fan charts* para um arquivo Excel externo

É possível exportar, entre outros objetos,

- tabelas com os cenários estocásticos simulados;
- *fan charts*;

para um arquivo Excel externo. No exemplo abaixo, o arquivo `simulation_results.xlsx` será criado.

```

export_to_excel(
  object = s,
  type = "simulation",
  percentiles = c(0.1, 0.25, 0.5, 0.75, 0.9),
  filename = "simulation_results.xlsx",
  draw_charts = c("debt", "primary balance"),
  date_format = "YYYY",
  y_axis = list(debt = c(50, 100))
)

```

A função `export_to_excel` também pode exportar outros resultados úteis da simulação (veja os exemplos na documentação específica da função, que acompanha o pacote).

Para gerar os *fan charts* no Excel, por meio do argumento `draw_charts`, é preciso que haja uma instalação de Python no computador. Para mais detalhes, veja a seção 3.2.

3 Detalhes adicionais

3.1 Equação de dinâmica da dívida

A equação de dinâmica da dívida implementada pela função `sim` é²⁴

$$d_t = \frac{(1 + r_t)}{(1 + g_t)} d_{t-1} - rp_t$$

em que

- d_t é a dívida, em percentual do PIB, no período t ;
- r_t é a taxa real de juros (em relação ao deflator do PIB) no período t ;
- rp_t é o resultado primário do governo, em percentual do PIB, no período t ;
- g_t é a taxa de crescimento real do PIB no período t .

Se o(a) usuário(a) utilizar os argumentos `tic` e `r_tic` da função `sim`, a equação de dinâmica da dívida implementada pela função será diferente (porém algebricamente equivalente à anterior²⁵):

$$d_t = \frac{(1 + \pi_t^{IPCA})}{(1 + \pi_t)} \frac{(1 + r_t^{IPCA})}{(1 + g_t)} d_{t-1} - rp_t$$

em que

- r_t^{IPCA} é a taxa real de juros (em relação ao IPCA) no período t ;
- π_t^{IPCA} é a inflação, calculada a partir do IPCA, no período t ;
- π_t é a inflação, calculada a partir do deflator do PIB, no período t .

Essa segunda equação de dinâmica da dívida tem duas utilidades práticas. Em primeiro lugar, ela permite verificações de consistência da simulação. Isto é, permite validar os cálculos. Afinal, é comum que outras organizações apresentem a equação em termos da inflação ao consumidor (Índice Nacional de Preços ao Consumidor Amplo – IPCA) e do deflator do PIB.

Em segundo lugar, ao se informar os argumentos `tic` e `r_tic` da função `sim`, *fan charts* adicionais serão gerados, referentes ao IPCA (π_t^{IPCA}) e à taxa real de juros em relação ao

²⁴ Confira a explicação completa sobre a equação no EE nº 18 (Casalecchi, 2023, pg. 51)

²⁵ Confira a explicação sobre essa equivalência no EE nº 18 (pg. 53).

IPCA (r_t^{IPCA}). Esses *fan charts* são uma forma de extrapolar a incerteza histórica dessas variáveis para o futuro, sendo útil para informar decisões razoáveis de projeção.

Para o uso cotidiano do pacote, com vistas a simular trajetórias para a dívida pública, a escolha do(a) usuário(a) por uma ou outra equação é inócuia, pois levará aos mesmos resultados numéricos.

Note que, se outro índice de inflação (por exemplo, o Índice Nacional de Preços ao Consumidor – INPC) for utilizado no lugar do IPCA, a explicação acima continua válida, bastando trocar as menções ao IPCA pelo INPC.

3.2 Checagem inicial sobre o Python

Para gerar os *fan charts* no Excel, por meio da função `export_to_excel()` e seu argumento `draw_charts`, é preciso que

- haja uma instalação de Python no computador;
- os seguintes módulos do Python estejam instalados: `pandas`, `numpy`, `xlsxwriter` e `openpyxl`.

Caso o(a) usuário(a) não tenha instalado o Python ou os módulos mencionados, o pacote `stocdebt` emitirá avisos. Mesmo assim, vale acrescentar as seguintes orientações.

Ao ser carregado pelo(a) usuário(a) no R, por meio do comando `library(stocdebt)`, o pacote busca por uma instalação do Python no computador. Essa busca é feita pela função `.onLoad()`, interna ao pacote, não visível. O Python é necessário apenas para exportar gráficos para arquivos Excel. Todas as demais operações funcionarão adequadamente, mesmo sem o Python.

As funções necessárias do Python serão acionadas a partir de dentro do pacote R, não havendo necessidade de o(a) usuário(a) interagir com o Python diretamente. Isso vale contanto que o Python esteja adequadamente presente na variável PATH do sistema.

Geralmente, ao instalar o Python no computador, o programa será adicionado automaticamente à PATH, não sendo necessário que o(a) usuário(a) se preocupe em adicioná-lo manualmente. Contudo, se após a instalação, ainda assim o R não detectar o Python, o pacote `stocdebt` emitirá um aviso, sendo talvez necessário que a adição à PATH seja manual.

Outra situação que pode exigir intervenção manual na PATH é o conflito entre versões do Python. Se houver mais de uma versão do Python instalada (por exemplo, 3.10 e 3.13), deve-se estar atento para garantir que a versão detectada pelo R seja aquela desejada pelo(a) usuário(a).

Pode-se usar o pacote `reticulate` para descobrir a versão do Python em uso pelo R:

```
reticulate::py_discover_config()
```

Pode-se também orientar o R a usar uma versão específica do Python, por meio do comando

```
reticulate::use_python("caminho/para/o/python")
```

em que “caminho\para\o\python” deve ser o caminho para a versão desejada (por exemplo, “C:\Usuários\...”).

3.3 Uso de *namespaces* para evitar erros

Ao escrever o código de cada função do pacote `stocdebt`, diversas outras funções foram utilizadas, como é usual. Essas outras funções pertencem a variados pacotes, incluindo os pacotes essenciais que compõem a instalação básica do R (`base`, `stats`, etc.).

Para garantir que as funções corretas fossem utilizadas, todas foram precedidas de seus pacotes (ou *namespaces*) quando eram chamadas dentro do código. Por exemplo, usou-se

```
base::colnames(x)  
zoo::index(y)
```

em lugar de, simplesmente,

```
colnames(x)  
index(y)
```

Essa escolha torna o código mais prolixo para o desenvolvedor. Em compensação, garante que as simulações estejam menos sujeitas a trocas desavisadas entre funções que, apesar de possuírem o mesmo nome, pertencem a pacotes diferentes.

3.4 Leitura do código-fonte

O código instalado pelo(a) usuário(a) no próprio computador não conterá **comentários** dentro do próprio código. Assim, o(a) leitor(a) que desejar conhecer detalhes do código terá mais facilidade de compreendê-lo no repositório online, onde há diversos comentários nos arquivos `.R`.

3.5 Possibilidades de extensão e aprimoramento do código

O código é aberto e admite muitas extensões possíveis. Entre elas, há espaço para os seguintes incrementos:

- probabilidade de mais eventos: estender a lista de eventos de interesse, cuja probabilidade é estimada por `event_prob`.
- melhor exibição (`print`) dos resultados: por exemplo, criar uma tabela, a ser exibida no terminal, contendo as estimativas retornadas por `event_prob`, em que cada linha apresenta a descrição do evento e a estimativa numérica.
- testes de validação: elaborar testes unitários (*unit tests*) e mais tratamentos de erro (*error handlers*);
- mais estratégias de geração de choques: a função `shocks_generator` foi elaborada de modo que pudesse ser facilmente estendida, aceitando novos blocos condicionais (`if`), acionados pelo argumento `case`.

3.5.1 Classes e métodos

Ao elaborar o código, optou-se por não criar novas **classes e métodos do R**. Essa escolha exigiu maior uso de condicionais (`if`) em algumas funções. Entendeu-se que, desta forma, o código permaneceria mais comprehensível para usuários(as) menos acostumados com a linguagem, ampliando a transparência, a disseminação e a possibilidade de receber contribuições da sociedade para o código.

3.5.2 Eficiência

Sempre há a possibilidade de modificar o código para aumentar sua **eficiência**, isto é, para que produza resultados mais rapidamente. Entretanto, sugere-se que esse tipo de modificação não seja feita de modo a prejudicar a facilidade de compreensão do código pelo público de potenciais contribuidores. Por exemplo, há *loops* que poderiam ser substituídos por operações vetoriais, mas tal alteração poderia tornar menos evidente o papel desempenhado por aquela parte do código para a obtenção do resultado final.

Referências

CASALECCHI, A. (2023). Metodologia de cenários estocásticos para a dívida pública.

Estudo Especial nº 18. Instituição Fiscal Independente. Disponível em:

<https://www2.senado.leg.br/bdsf/bitstream/handle/id/645203/EE18.pdf>.

CASALECCHI, A. (2024). Sensibilidade dos cenários estocásticos de dívida pública. Nota Técnica nº 54. Instituição Fiscal Independente. Disponível em:

https://www2.senado.leg.br/bdsf/bitstream/handle/id/651183/NT54_Sensibilidade_cenarios_estocasticos_dvida.pdf

[CE] COMISSÃO EUROPEIA (2025). Debt Sustainability Monitor 2024. Directorate-General for Economic and Financial Affairs. Institutional Paper 306. Disponível em:

https://economy-finance.ec.europa.eu/publications/debt-sustainability-monitor-2024_en.

[IFI] INSTITUIÇÃO FISCAL INDEPENDENTE (2025). Relatório de Acompanhamento Fiscal. Junho de 2025. Disponível em:

https://www2.senado.leg.br/bdsf/bitstream/handle/id/703934/RAF101_JUN2025.pdf.

[OCDE] ORGANIZAÇÃO PARA A COOPERAÇÃO E O DESENVOLVIMENTO ECONÔMICO (2014). Recommendation of the Council on Principles for Independent Fiscal Institutions. Public Governance and Territorial Development Directorate. Disponível em:

<https://www.oecd.org/content/dam/oecd/en/topics/policy-sub-issues/parliamentary-budget/OECD-Recommendation-on-Principles-for-Independent-Fiscal-Institutions.pdf>.

Apêndice A

Instalação a partir dos arquivos .tar.gz e .zip baixados

Se não for possível instalar diretamente – pela linha de comando do R – do repositório online GitHub, acesse esse site manualmente por meio de seu navegador web e baixe os arquivos anexos à [versão \(release\)](#) de interesse:

- `stocdebt_<VERSÃO>.zip` (para Windows);
- `stocdebt_<VERSÃO>.tar.gz` (para Linux).

Por exemplo, para a versão 1.0.0, os arquivos seriam nomeados como `stocdebt_1.0.0.zip` e `stocdebt_1.0.0.tar.gz`.

A.1 Windows (arquivo .zip)

Depois de baixar `stocdebt_<VERSÃO>.zip`, execute os seguintes comandos, substituindo o argumento `pkgs` pelo caminho até o arquivo baixado.

```
# Instalar dependências (outros pacotes, dos quais "stocdebt" depende para
# funcionar adequadamente)
install.packages(c("MASS", "readr", "withr", "zoo", "dplyr", "dygraphs", "ggplot2",
  "openxlsx", "rlang", "tidyverse", "scales", "reticulate", "xts"))

# Instalar o pacote "stocdebt"
install.packages(pkgs = "C:\...\...\stocdebt_<VERSÃO>.zip", repos = NULL)
```

A.2 Linux (arquivo .tar.gz)

Depois de baixar `stocdebt_<VERSÃO>.tar.gz`, execute os seguintes comandos, substituindo o argumento `pkgs` pelo caminho até o arquivo baixado.

```
# Instalar dependências (outros pacotes, dos quais "stocdebt" depende para
# funcionar adequadamente)
install.packages(c("MASS", "readr", "withr", "zoo", "dplyr", "dygraphs", "ggplot2",
  "openxlsx", "rlang", "tidyverse", "scales", "reticulate", "xts"))

# Instalar o pacote "stocdebt"
install.packages(pkgs = "/.../.../stocdebt_<VERSÃO>.tar.gz", repos = NULL)
```

ifi



-  /ifibrasil
-  @ifiBrasil
-  @ifibrasil
-  /company/ifibrasil
-  /ifibrasil
-  github.com/ifibrasil

ifi@senado.leg.br

61 3303 2875

