

R e RStudio para Iniciantes

**Material de Apoio para Cursos Quantitativos do Instituto de Economia da
Universidade Federal do Rio de Janeiro (IE/UFRJ)**

GPEQ/UFRJ

2024-04-11

Índice

Prefácio

O que você vai aprender

Pretendemos que você domine o *mínimo* necessário de programação em R para executar as tarefas que podem ser requisitadas pelo seu professor, independentemente do curso da área quantitativa em que estiver. Em outras palavras, se te pedirem algo que deva ser elaborado com auxílio de programação em R, você será capaz de fazê-lo após ler este material¹.

Na prática, o quê significa *dominar o mínimo necessário de programação em R*? Inclui entender alguns *conceitos* básicos – para quê serve a programação em nosso contexto, o que é a linguagem de programação R, o que é o RStudio, entre outros – assim como a *sintaxe* da linguagem – ou seja, o ato de escrever um código interpretável propriamente dito.

O que você não vai aprender

Não estamos em um curso de Ciência da Computação: você não irá aprender terminologias difíceis e/ou como a programação, de modo geral, funciona nos *detalhes*. Em outras palavras, vamos nos concentrar apenas em entender o necessário para construir e executar *códigos* em R (não se preocupe, ainda explicaremos o que é um *código em R*) a partir das tarefas que seu professor poderá pedir.

Além disso, o material não te dará proficiência em R. O que queremos dizer com isso? Bom, queremos dizer que você não será uma pessoa que dominará o R de forma *avançada*. Novamente: aqui, te ensinaremos apenas o necessário para que consiga concluir os cursos da área quantitativa. Mas, se você realmente quiser alcançar níveis mais altos, alguns livros podem te ajudar:

- [R for Data Science \(2ª edição\)](#)
- [Ciência de Dados em R](#)
- [Data Science for Psychologists](#)
- [An Introduction to R for Research](#)

¹Esperamos que os empecilhos que apareçam não sejam por conta de alguma dificuldade no ato de programar em si, mas por dúvidas com relação à matéria propriamente dita. De qualquer forma, fique tranquilo: se você não entendeu alguma parte do material, estaremos **sempre** abertos a te ajudar!

Preciso saber alguma coisa de forma antecipada?

Não. Você não precisa saber absolutamente *nada* de programação em R – não precisa nem mesmo saber o que o termo *programação* significa. O intuito do material é justamente te introduzir aos conceitos mais básicos!

A única coisa que você precisará será de acesso à um computador com internet. Utilizar um computador é necessário pois é nele onde ocorre o ato de programar; ter internet é importante porque, ao longo dos capítulos, precisaremos que você realize o *download* de certos arquivos – seja para instalar o R e o RStudio ou para *importar* algum arquivo diretamente para este último (não se preocupe, ainda explicaremos o que *importação* de um arquivo significa).

Como o material está organizado

O material está organizado em sete capítulos: o primeiro, que te mostra a motivação para programar, além de outros seis que buscam, em primeiro lugar, te guiar na instalação do R e RStudio e, na sequência, ensinar comandos e conceitos básicos que serão necessários ao longo dos cursos. Com intuito de facilitar o aprendizado, cada capítulo foi repartido em um certo número de seções (e subseções, quando necessário).

A lista de capítulos pode ser observada no menu à *esquerda*. Por sua vez, a lista de seções do capítulo em que você estiver pode ser observada no menu à *direita*. Perceba que, para ser direcionado a um determinado capítulo/seção, basta clicar em seu nome.

Prefácio

1 Por que programar?

Instalação



2 Instalando o R

3 Instalando o RStudio

Programando em R



4 Primeiros passos

5 Objetos

6 Funções e pacotes

7 Importando dados

Índice

Prefácio

O que você vai aprender

O que você **não** vai aprender

Preciso saber alguma coisa de forma antecipada?

Como o material está organizado

Dúvidas e sugestões: com quem falar?

“Caramba, queria tanto acessar uma parte específica do material que não lembro muito bem onde está... E agora?” Sem problemas: você pode pesquisar partes do texto ou palavras-chave no campo em branco logo acima do Prefácio!

Dúvidas e sugestões: com quem falar?

“Ué, no meu computador não aparece isso!”

“Caramba, achei aquele trequinho ali meio confuso... podia melhorar...”

“Nossa, que material show!”

Surgiu alguma dúvida ou então quer dar alguma sugestão de melhoria? Estamos totalmente abertos à qualquer tipo de crítica! Envie uma mensagem para pedro.hemsley@ie.ufrj.br.

1 Por que programar?

De forma simplificada, é possível definir o ato de programar como a passagem de determinados comandos para o computador, com a finalidade de que ele execute determinada tarefa. Se você deseja algo que pode ser feito de forma mais eficiente por uma máquina, provavelmente escreverá um código que seja interpretável por esta, de modo que seu desejo se concretize.

A capacidade de programar tornou-se uma habilidade essencial, especialmente para aqueles que desejam explorar o mundo da estatística e da matemática aplicados à determinada ciência social. Por exemplo, no contexto de interseção entre economia e matemática – principalmente na elaboração e solução de modelos teóricos – e entre economia e estatística – testando hipóteses e realizando previsões – a programação se coloca como uma ferramenta muito útil para economizar tempo de cálculo e garantir que, caso necessário, o mesmo processo seja concluído múltiplas vezes sem erros. Em outras palavras, a programação aplicada à determinada ciência social, como a economia, traz duas principais vantagens, exploradas melhoras a seguir.

1.1 Redução no tempo de cálculo

A primeira vantagem é a redução no tempo de cálculo de certos procedimentos que, se feitos de forma manual, levariam vários minutos, horas ou até mesmo dias. Vamos deixar mais claro com um exemplo.

No ensino fundamental, você aprendeu a resolver um sistema de equações simultâneas com 2 variáveis e 2 equações, muito provavelmente pelo método de substituição. Não levava muito tempo, certo? Acontece que, na cadeira de Álgebra Linear, você aprenderá como solucionar sistemas de n equações e n variáveis. Normalmente, quanto maior n , maior será a dificuldade de encontrar a solução do sistema. Ainda que existam *algoritmos* que permitam encontrar a solução de forma mais rápida, certo tempo será perdido se você os replicar de forma *manual*.

Com auxílio da programação, no entanto, é possível implementar estes mesmos algoritmos para obter o resultado de forma quase que *instantânea*. *O tempo que você levaria fazendo o procedimento manual praticamente se reduz a zero – ou fica mínimo, em relação ao inicial.* Observe que você ainda deve focar em saber como o algoritmo funciona, do contrário não será capaz de julgar se o que a máquina fez é realmente aquilo que você desejava.

1.2 Automação de processos

Na seção anterior, repare que estávamos discorrendo implicitamente sobre cálculos de ocorrência única – ou seja, realizamos o cálculo uma vez e não teríamos mais interesse de fazê-lo novamente em um futuro próximo. No entanto, outro benefício prático do ato de programar é a automação de tarefas repetitivas. Com a programação, é possível escrever e salvar *scripts* que automatizam tarefas tediosas de manipulação e análise de dados, permitindo que os pesquisadores se concentrem em questões analíticas de maior relevância.

Por exemplo, imagine que alguém te peça para calcular a média de certos valores que mudam de dia para dia. Você pode facilmente elaborar um *script* que, a partir de determinados números (sem especificar quais são), calcule sua média. Uma vez escrito e salvo, você pode passar a executá-lo sempre que quiser – no exemplo, todos os dias.

1.3 Vamos programar!

Em suma, aprender a programar oferece uma série de vantagens tangíveis para quem trabalha com estatística e matemática. Ela torna o trabalho mais eficiente e produtivo, permitindo que os profissionais explorem dados de maneiras antes inimagináveis e desenvolvam soluções personalizadas para os desafios enfrentados em suas áreas de atuação.

No restante do material, aprenderemos a programar utilizando a *linguagem de programação R*. Em outras palavras, aprenderemos sua *sintaxe*, isto é, a forma de escrever comandos corretamente para que a máquina seja capaz de interpretar e executar o que queremos como resultado.

Parte I

Instalação

Nessa parte, você irá aprender como baixar e instalar o R e o RStudio, além da composição de *layout* de ambos. Construimos cada seção de instalação como um guia do tipo *passo a passo*, de maneira que você precisa apenas segui-los de forma direta. Neste capítulo, é importante que você já comece a explorar um pouco a interface dos ambientes de programação que te mostraremos.

2 Instalando o R

Nesse capítulo, iremos aprender como baixar e instalar o R para Windows¹! Optamos por dividir o passo a passo em 7 etapas – mas fique tranquilo, não são passos grandes, apenas fizemos dessa forma para que o conteúdo fique bem *mastigado*, fácil de entender.

💡 Alguns conceitos iniciais (Opcional)

Antes de começar, vamos entender alguns conceitos. A ideia aqui é te ensinar o que significam algumas nomenclaturas e siglas que aparecem ao longo do processo de instalação, em especial *R Foundation* e *CRAN*. Essa parte é totalmente *opcional* e você pode pular direto para o passo a passo caso esteja sem tempo – ou até mesmo interesse.

- **R Foundation:** é uma empresa sem fins lucrativos, criada pelos principais desenvolvedores da linguagem. Quais são seus objetivos? Basicamente três: (i) administrar os direitos autorais da linguagem – e, por consequência, manter seu uso como livre; (ii) apoiar o desenvolvimento do R como um todo, isto é, fornecer informações e criar novos usos básicos, elaborar conferências, guias, entre outros; (iii) servir como ponto focal para todos os usuários da linguagem que desejem interagir com a comunidade de desenvolvedores. De forma resumida, a R Foundation é como se fosse a instituição provedora do básico da linguagem, que busca sempre atualizar e mantê-lo de pé. Se você instala o R e, logo em seguida, percebe que alguma de suas atribuições não está em perfeito funcionamento, provavelmente terá que comunicar à essas pessoas. Grosso modo, exerce um papel próximo ao da Microsoft com o Excel, por exemplo. Uma observação (importante): como o R é um software livre, qualquer pessoa pode desenvolver novas funções ou recursos a partir da linguagem. Por esse motivo, para recursos que estejam além da *base do R*, você deve recorrer à quem os criou! Por exemplo, com relação ao RStudio (que conheceremos mais à frente), devemos nos reportar à empresa Posit, sua desenvolvedora. Na prática, raramente (para não dizer nunca) iremos reportar alguma coisa à R Foundation, mas sim aos desenvolvedores daquele pacote/extensão específico (fique tranquilo, explicaremos mais à frente o conceito de *pacote* para a linguagem).

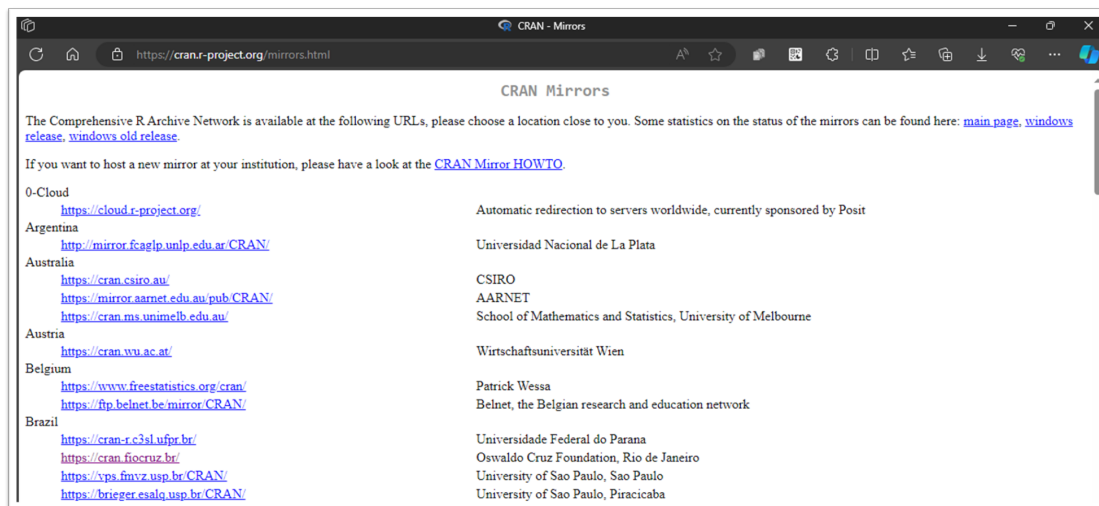
¹Você pode realizar procedimento equivalente para sistemas operacionais Linux, apenas alterando a opção de *download* quando necessário – isto é, selecionando as opções em que esteja escrito ‘Linux’, ao invés de ‘Windows’.

- **CRAN** (Comprehensive R Archive Network): segundo o próprio, é “*uma coleção de sites que carrega material idêntico, consistindo nas distribuições do R, extensões contriubídas, documentação e arquivos binários de R*”. ‘*Meu Deus, o que isso significa?*’ Simples: apenas uma coleção de endereços da internet em que podemos baixar a versão mais recente do R, assim como pacotes. Quem mantém o CRAN? Instituições voluntárias; em seus sites, a parte onde é possível baixar arquivos relacionados ao R é chamada de *espelho*. E com quais recursos o CRAN se mantém? Com os da própria instituição participante (principalmente em termos de colaboradores) e, também, da R Foundation!

Essa história toda para dizer: **o arquivo básico que iremos baixar para instalar o R será obtido através de algum *espelho* do CRAN, isto é, a parte do site de alguma instituição voluntária em colaboração com a R Foundation.**

2.1 Sete passos

1. O primeiro passo consiste em escolher um repositório (*espelho*) para baixar o R. No endereço <https://cran.r-project.org/mirrors.html> encontramos todas as opções disponíveis, por país e em ordem alfabética. No seu computador, deverá aparecer a seguinte tela:



2. Por questões de rapidez/latência, o ideal é escolher o repositório mais próximo de você. Considerando que todos estejam no Rio de Janeiro, vamos então utilizar o *espelho* da Fiocruz.