

Glossário

(palavras em negrito indicam termos que seriam definidos adiante)

Python: a **linguagem de programação** que usaremos.

Anaconda: uma distribuição do Python para análise de dados, que diz, o Python mais um conjunto de pacotes escritos em Python importantes para as análises.

Jupyter: um programa que permite executar comandos de uma linguagem em um navegador web. Ele foi desenvolvido primeiramente com o nome de **IPython**, mas mudou de nome pois começou a ser possível utilizar outras linguagens. O nome vem das iniciais das linguagens Julia, Python e R.

IPython: nome antigo do Jupyter e nome atual da versão do Jupyter que é executada em um terminal e executa apenas Python.

Jupyter Notebook: o formato de documento usado pelo Jupyter, aberto dentro do navegador. É a principal interface para analisar dados usando Python.

Linguagem de programação: vocabulário e conjunto de regras sobre como dar instruções para computadores executarem certas tarefas. Isto é, a linguagem de programação transforma **algoritmos** em códigos que o computador consiga entender. Esses códigos implementam operações básicas (p. ex., adicionar, subtrair e ordenar caracteres), estruturas de dados (listas, dicionários, arrays, conjuntos, dataframes etc.) e servem para criar estruturas de repetição, estruturas condicionais e **funções**. As linguagens ficam ainda mais interessantes por causa dos **pacotes** que são escritos a partir delas. No final deste glossário há uma lista das principais linguagens de programação.

Algoritmo: instruções para executar certas tarefas. P. ex., para cada um dos alunos, se ele acertou 70% ou mais das questões, entregue o certificado; caso contrário, não entregue o certificado. São criados usando estruturas de repetição, estruturas condicionais, estruturas de dados e funções.

Pacote (ou biblioteca): um conjunto de **funções** pré-definidas. Mais abaixo há uma lista com os principais pacotes usados para análise de dados em Python.

Função: **algoritmos** para transformar determinadas coisas (*inputs*, entradas) em outras coisas (*outputs*, saídas). Para criar uma função em Python basta digitar:

```
def soma(a, b):  
    return a + b
```

Feito isso, quando você executar a função `soma(3, 35)`, terá o resultado 38. As funções mais úteis são aquelas que simplificam tarefas muito complexas ou repetitivas, por exemplo, pesquisar “Justin Bieber sem camisa” em todas as páginas da internet, sugerir vídeos relacionados ao que você está assistindo, registrar a compra que você fez em seu cartão de crédito, criar gráficos, descompactar vários arquivos, identificar todas as palavras que terminam em “ão” em um determinado livro etc.

Métodos: **funções** que são atributos de objetos.

Dataframe: uma estrutura de dados semelhante a uma tabela ou planilha, com linhas representando observações e colunas representando variáveis.

Machine learning (aprendizagem de máquina): técnicas estatísticas para atualizar algoritmos.

Principais pacotes para análise de dados em Python

Pandas: para transformar dados em dataframes.

Numpy: para fazer operações numéricas complexas, especialmente álgebra matricial.

Matplotlib: para fazer gráficos.

Seaborn: para deixar os gráficos mais bonitos.

Statsmodels: para fazer análise de regressão.

Scipy.stats: ferramentas estatísticas.

Scikit-learn: aprendizagem de máquina (*machine learning*), incluindo análises estatísticas como agrupamento, componentes principais, árvores de decisão e redes neurais.

Principais linguagens de programação

Python: famosa por ser simples, elegante, poderosa e flexível, ela é utilizada para desenvolvimento web (p. ex., sites de compras), operações em redes e sistemas operacionais (p. ex., movimentar informações entre diferentes programas), pesquisas científicas (astronomia, física, neurociência etc), análise de dados e aprendizagem de máquina (*machine learning*).

C e C++: usadas em situações em que a velocidade de execução é muito importante (sistemas operacionais, jogos de videogame realistas, máquinas industriais e instituições financeiras). Em situações em que o Python precisa executar operações complexas rapidamente, ele executa códigos em C nos bastidores. C e C++ são famosas por serem linguagens difíceis, que exigem muitos detalhes para que os comandos funcionem corretamente.

Java: é a linguagem dos aplicativos de Android e de vários programas de empresas, você provavelmente já teve que atualizar a Java em seu computador antes de instalar algum programa.

Javascript: é a linguagem dos navegadores da web (e não tem relação com a Java, a escolha do nome foi apenas um estratégia de marketing). Uma página na web normalmente é criada usando três linguagens: HTML, CSS e Javascript. As duas primeiras são muito simples: uma serve para indicar qual parte do texto é um título, um parágrafo ou uma imagem, enquanto a outra serve para definir as cores, o tamanho e a fonte das letras, das linhas e dos fundos das diferentes partes da página. A Javascript é mais poderosa, é ela que lida com seus cliques, registra suas senhas e permite que o conteúdo de uma página mude sem precisar recarregar novamente a página (p. ex., a caixa de e-mails, o Facebook).

R: assim como a Javascript, o R é uma linguagem de domínio específico, ao contrário de linguagens que podem ser aplicadas a diversas situações, tais como C/C++, Java e Python. Enquanto a Javascript é direcionada para os navegadores, o R foi criado para análises estatísticas e reinava sozinho nessa área até cerca de 5 anos, quando o Python começou a competir nesse espaço. Porém, o R continua tendo uma grande vantagem: ele é a linguagem dos professores de estatística. Por isso, dentre os cerca de 10 mil pacotes escritos em R, há várias implementações de técnicas estatísticas resultantes de pesquisas de doutorado. Embora o Python possua mais de 100 mil pacotes em diversas áreas, o R ainda possui mais pacotes relacionados a técnicas estatísticas sofisticadas.

Além dessas, há dezenas de outras linguagens, algumas em decadência (Fortran, Matlab, PHP e Ruby), outras em ascensão (Rust, Go e Julia). E existem ainda diversas pequenas linguagens simples que são muito importantes para certas tarefas, já vimos o caso do HTML e do CSS na construção de páginas da internet. Outros exemplos são o Markdown, que serve para escrever HTML de uma maneira mais simples, Latex, que serve para escrever fórmulas matemáticas (e documentos inteiros) e Regex, que serve para procurar padrões em textos.