

- PROGRAMAÇÃO _
- DATA SCIENCE _
- DEVOPS _
- MOBILE _
- FRONT-END _
- INTELIGÊNCIA ARTIFICIAL _
- UX & DESIGN _
- INOVAÇÃO & GESTÃO _

Artigos > Data Science

Como remover linhas e colunas no Pandas

	total_bill	tip	smoker	day	time	size
0	16.99	1.01	No	Sun	Dinner	2
1	10.34	1.66	No	Sun	Dinner	3
2	21.01	3.50	No	Sun	Dinner	3
3	23.68	3.31	No	Sun	Dinner	2
4	24.59	3.61	No	Sun	Dinner	4



Yuri Matheus
24/12/2019
Atualizado em 14 de novembro

COMPARTILHE





mineração, realizamos muitas mudanças no **dataset**.

Existem colunas que não queremos, ou precisamos usar, ou linhas que não contém dados válidos que precisam ser removidas.

Confira neste artigo:

- [Removendo colunas no Pandas](#)
- [Removendo linhas](#)
- [Para saber mais](#)

//

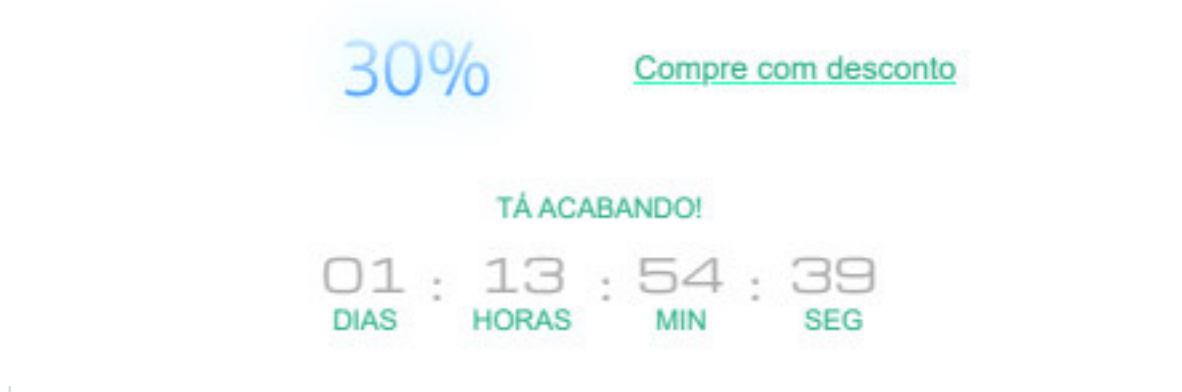
Mas quantos mais dados não é melhor?

Quando estamos analisando dados, principalmente para treinar **modelos de inteligência artificial**, alguns dados podem prejudicar ao invés de ajudar. Por isso, temos que utilizar apenas os dados que fazem sentido para o modelo.

Legal! Já sabemos que precisamos tratar os dados, mas como podemos fazer isso, como remover os dados?

Removendo colunas no Pandas

Estou fazendo uma análise sobre gorjetas em transportes como táxis. Para isso, estou utilizando o [Pandas](#) que é uma biblioteca que facilita a manipulação dos dados.



	total_bill	tip	smoker	day	time	size
0	16.99	1.01	No	Sun	Dinner	2
1	10.34	1.66	No	Sun	Dinner	3
2	21.01	3.50	No	Sun	Dinner	3
3	23.68	3.31	No	Sun	Dinner	2
4	24.59	3.61	No	Sun	Dinner	4

Podemos ver que o **DataFrame** lembra muito uma planilha. Cada coluna também conhecida como variável tem um nome, e cada linha conhecida como observação tem um índice e os dados que representam cada variável.

Para a análise que eu quero fazer, não preciso saber se as pessoas são fumantes ou não, por isso, não faz necessário analisar a coluna **smoker**, como podemos removê-la?

A primeira coisa que precisamos fazer é excluir a coluna que nos diz se as pessoas são fumantes ou não. Essa coluna não é necessária para o tipo de análise que queremos fazer. Então vamos removê-la. Mas como?

No Pandas, existe o **método drop** para os data frames. Nesse método, podemos passar o índice da linha que desejamos remover.:

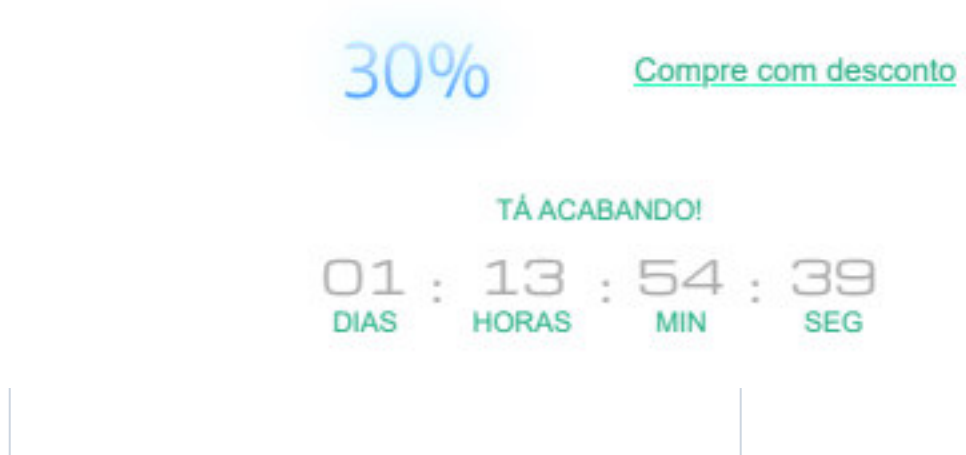
```
>>> # no interpretador
>>> tips.drop(2)
```



Esse comando, removemos a linha de índice 2. Mas não queremos remover uma linha e sim uma coluna. Para isso, basta passar o parâmetro `columns`, esse parâmetro recebe uma lista com o nome das colunas que desejamos remover, `smoker` no nosso caso.

Após isso, vamos falar para o Pandas nos mostrar o cabeçalho desse data frame:

```
tips.drop(columns=['smoker'])
tips.head()
```



Ué? A coluna ainda está ali. O que aconteceu?

Por padrão, **as operações que efetuamos em um DataFrame não afetam ele**. Isto é, sempre que removemos uma coluna, linha, etc, o que o Pandas faz é devolver um novo DataFrame sem aquele dado. Ou seja, o DataFrame original se mantém intacto.

O que podemos fazer para resolver isso é atribuir esse data frame que é devolvido pelo método na mesma variável:

```
tips = tips.drop(columns=['smoker'])
tips.head()
```

	total_bill	tip	day	time	size
0	16.99	1.01	Sun	Dinner	2
1	10.34	1.66	Sun	Dinner	3
2	21.01	3.50	Sun	Dinner	3
3	23.68	3.31	Sun	Dinner	2
4	24.59	3.61	Sun	Dinner	4

Legal! Conseguimos remover a coluna que precisávamos. Agora, vamos para outra parte do tratamento dos dados. Na análise que estou fazendo, quero



Removendo linhas

Sabemos que podemos utilizar o método **drop()** para retirar uma linha. Mas como remover todas as linhas que contém apenas um passageiro?

Podemos fazer um loop e iterar por cada linha no data frame e checar se o número de passageiros é maior que um. Se sim, mantemos esse dado, do contrário, removemos do data frame. Mas qual o problema dessa solução?

Mas o Pandas já tem diversas formas para trabalhar com seus data frames. O que podemos fazer é criar um filtro no qual é retornado um novo DataFrame com os dados já filtrados.

O Pandas permite filtrar um DataFrame pelos dados de sua coluna. Por exemplo, queremos pegar do data frame de gorjetas (**tips**) a coluna que define o número de passageiros (**size**) e com base nisso, queremos as linhas nas quais a coluna size é maior que um:

```
filtro = tips['size'] > 1
```

Agora, basta passarmos esse filtro como seletor no DataFrame:

30%

[Compre com desconto](#)

TÁ ACABANDO!

01 : 13 : 54 : 39
DIAS HORAS MIN SEG

```
gorgetas_em_grupo.head()
```

	total_bill	tip	day	time	size
0	16.99	1.01	Sun	Dinner	2
1	10.34	1.66	Sun	Dinner	3
2	21.01	3.50	Sun	Dinner	3
3	23.68	3.31	Sun	Dinner	2
4	24.59	3.61	Sun	Dinner	4

Para saber mais

Os filtros utilizados podem ser mais complexos. Podemos utilizar estruturas condicionais, como o E e OU e checar o valor de mais de uma coluna. Além disso, podemos fazer outros tipos de filtros.

Como checar se alguma linha tem um valor nulo, ou um valor de um tipo diferente do esperado. Para isso, o [Pandas](#) utiliza o conceito de [comparação rica, ou rich comparison](#).

Além de remoção, conseguimos fazer outras operações com os **data frames**, **operações que envolvem string**, **renomear colunas** e **adicionar colunas novas**, **juntar um data frame a outro**, entre diversas outras operações.


Aqui na Alura, temos uma [formação de data science com Pandas](#). Nele você verá como manipular data frames de diversas formas. Além de ver como **criar gráficos**, **visualizar estatísticas descritivas**, **tratar dados faltantes** e **muito mais**.

30% [Compre com desconto](#)

TÁ ACABANDO!

01 : 13 : 54 : 39
DIAS HORAS MIN SEG

- cursos em uma única assinatura
- Novos lançamentos a cada semana
- Desafios práticos



SAIBA MAIS



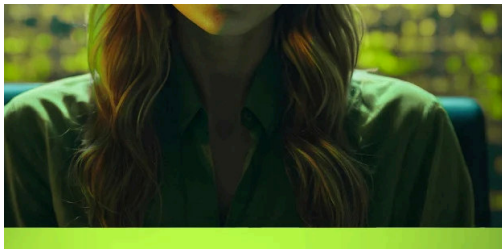
Yuri Matheus

Yuri é desenvolvedor e instrutor. É estudante de Sistemas de Informação na FIAP e formado como Técnico em Informática no Senac SP. O seu foco é nas plataformas Java e Python e em outras áreas como Arquitetura de Software e Machine Learning. Yuri também atua como editor de conteúdo no blog da Alura, onde escreve, principalmente, sobre Redes, Docker, Linux, Java e Python.

30%

[Compre com desconto](#)

TÁ ACABANDO!

01 : 13 : 54 : 39
DIAS HORAS MIN SEG`em python :)")``mbém em python :)`

MySQL Administrator:
replicação de banco de dados

Polynote: O Jupyter notebook
da Netflix

Cor

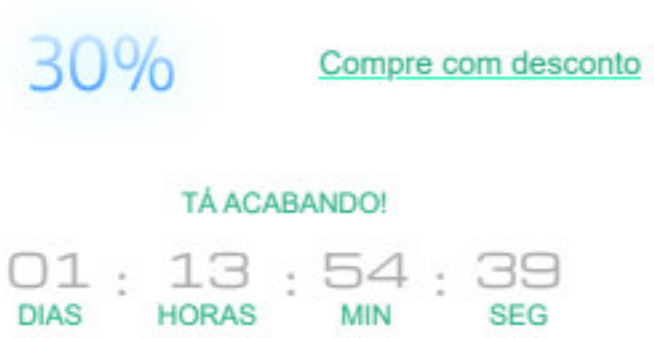


Veja outros artigos sobre
[Data Science](#)

Quer mergulhar em tecnologia e aprendizagem?

Receba conteúdos, dicas, notícias, inovações e tendências sobre o mercado tech diretamente na sua caixa de entrada.

bins.br@gmail.com



Nossas redes e apps



Institucional

- Sobre nós
- Trabalhe na Alura
- Para Empresas
- Para Sua Escola
- Política de Privacidade
- Compromisso de Integridade
- Termos de Uso
- Documentos Institucionais
- Status
- Uma empresa do grupo Alun

A Alura

- Como Funciona
- Formações
- Plataforma
- Depoimentos
- Instrutores(as)
- Dev em <T>
- Luri, a inteligência artificial da Alura
- IA Conference 2025
- Cursos imersivos
- Certificações

Conteúdos

Fale Conosco

30%

[Compre com desconto](#)

TÁ ACABANDO!

01 : 13 : 54 : 39
DIAS HORAS MIN SEG

Imersão Dados com Python II

Novidades e Lançamentos

ENVIAR

TRILHAS POR CARREIRA

Carreiras de IA

Engenharia de IA

Especialista em IA

Carreiras de Dados

Ciência de Dados

Análise de Dados

Governança de Dados

Carreiras de Cyber

Cloud Security

AppSec: Desenvolvimento Seguro de Aplicações

Carreiras de DevOps & Cloud

Platform Engineering

SRE (Site Reliability Engineering)

Carreiras de UX & UI

UI Design



Desenvolvimento Back-End Java

Desenvolvimento Back-End Node.js

Carreiras de Negócios

Liderança

Recursos Humanos (RH)

Social Media Marketing

Growth Marketing

CURSOS UNIVERSITÁRIOS FIAP

Graduação | Pós-graduação | MBA