



VER PLANOS

PROGRAMAÇÃO _

DATA SCIENCE _

DEVOPS _

MOBILE _

FRONT-END _

INTELIGÊNCIA ARTIFICIAL _

UX & DESIGN _

INOVAÇÃO & GESTÃO _

Artigos > **Data Science**

Oi! Posso indicar os melhores artigos para tirar suas dúvidas!

Melhorando a análise com o Boxplot



Daniel Siqueira

02/05/2021

COMPARTILHE



Esse artigo faz parte da
Formação Data Science

O que é um BoxPlot?

Um boxplot (ou diagrama de caixa, numa tradução livre) mostra a distribuição quantitativa dos dados de um jeito que facilita a comparação entre as variáveis, ou através dos níveis categóricos das variáveis.

Confira neste artigo:

- [Quer saber mais sobre o assunto?](#)

Essa caixa ("box") mostra os quartis do dataset enquanto os "whiskers" mostram o resto da distribuição, exceto os pontos que são chamados de

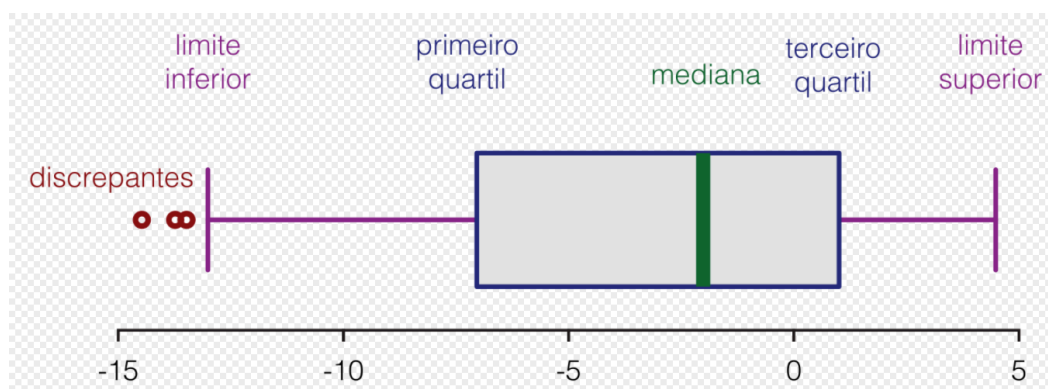
outliers.

Matricule-se na escola de DATA SCIENCE

Junte-se a uma comunidade de **+500 mil** estudantes

- Acesso a **TODOS** os cursos em uma única assinatura
- Novos lançamentos a cada semana
- Desafios práticos

SAIBA MAIS



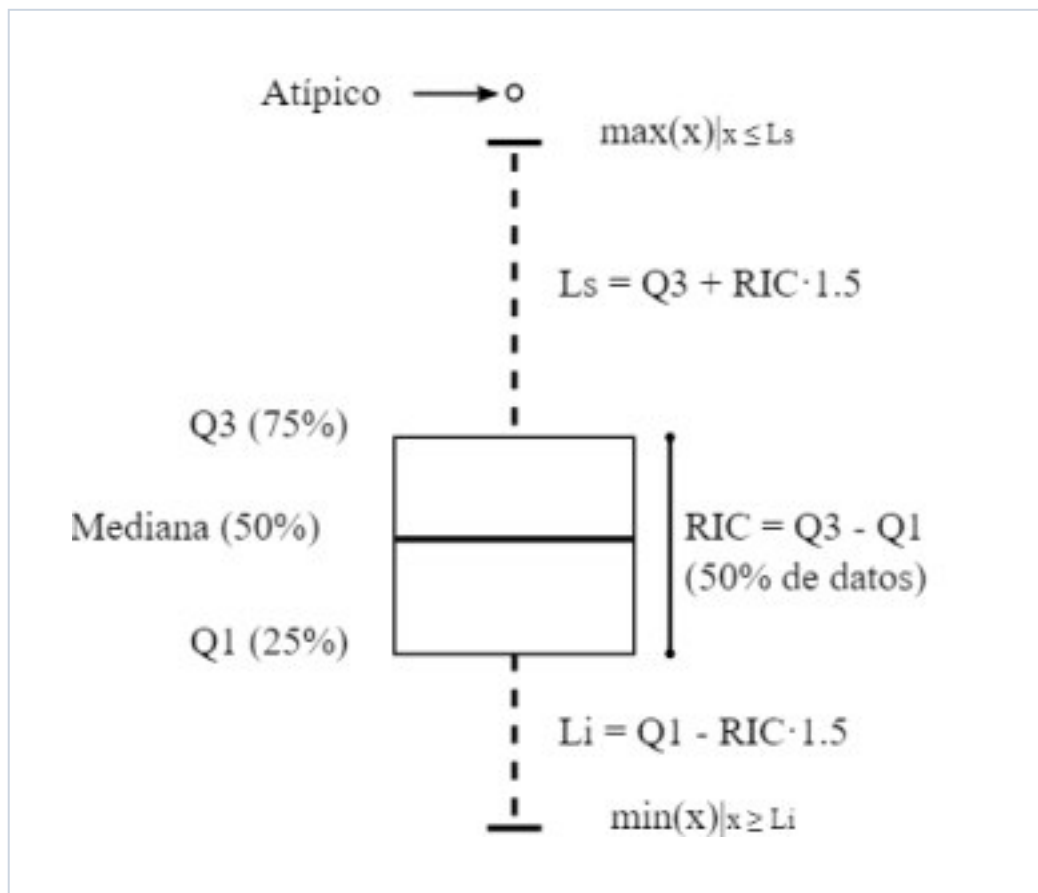
A linha no centro da figura em verde, no nosso caso, representa a mediana. A linha azul que está na aresta esquerda da figura representa 25% da minha distribuição, e a linha que está na aresta direita representa 75% da distribuição dos dados.

Nas linhas roxas que estão nas extremidades de ambos lados, saem da figura central e mostram uma delimitação, temos o início e o fim da área dos quartis. Agora, repare que há alguns pontos após essa delimitação no lado esquerdo. Esses pontos são chamados de outliers, ou seja, são pontos “isolados” na nossa distribuição.

Para sabermos até onde vão os whiskers, que são esses “bigodinhos de gato” que delimitam os quartis e marcam onde se inicia os outliers, vamos fazer um cálculo. Uma vez que encontramos a mediana, o ponto central da distribuição, sabemos que temos 50% da distribuição à esquerda e 50% à direita. Após isso, vamos dividir novamente os lados na metade, para obtermos duas partes de 25% à direita e 25% à esquerda.

Agora que temos os 4 quartis, cada um com 25%, podemos desenhar a caixa, o “box”, e delimitamos com o 2º e o 3º quartis. Para encontrarmos então a

delimitação dos whiskers vamos multiplicar a distância do 2º quartil até a mediana por 1,5. Fazemos o mesmo procedimento com a distância do 3º quartil até a mediana: vamos multiplicá-la por 1,5.



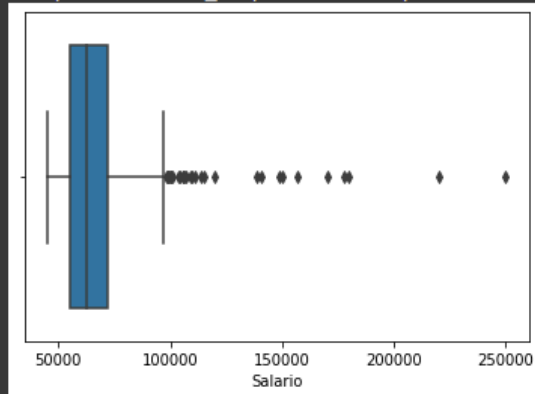
Em suma, o boxplot nos auxilia a visualizar a distribuição de dados divididos nos quartis. Ademais, mostra onde os dados estão mais concentrados e se há outliers fora dos nossos quartis.

Como gerar um BoxPlot usando Python?

Para gerar o Boxplot no [Python](#) vamos usar a biblioteca do Seaborn. Nós poderíamos também gerar o Boxplot através de outros métodos, porém uma das vantagens de se utilizar o Seaborn é que ele ficará mais bonito, claro e apresentável.

```
[12] sns.boxplot(x = dados['Salario'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f70d22902d0>
```

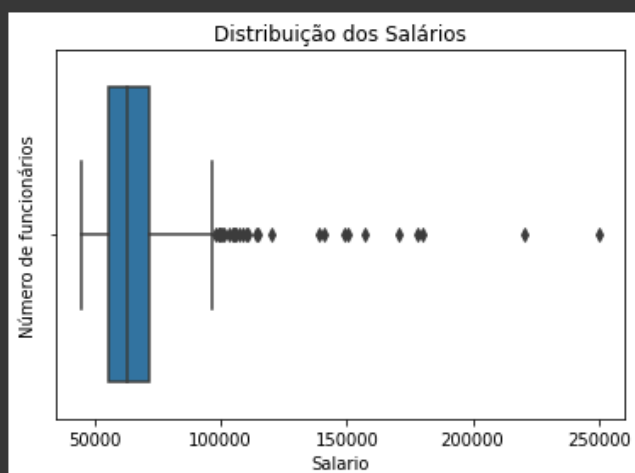


Para alterar os eixos x e y do nosso diagrama podemos utilizar a biblioteca matplotlib, através do código: `import matplotlib.pyplot as plt` `sns.boxplot(x = dados['Salario'])` `plt.ylabel('Número de funcionários')` `plt.show()`



```
import matplotlib.pyplot as plt
sns.boxplot(x = dados['Salario'])

plt.title('Distribuição dos Salários')
plt.ylabel('Número de funcionários')
plt.show()
```

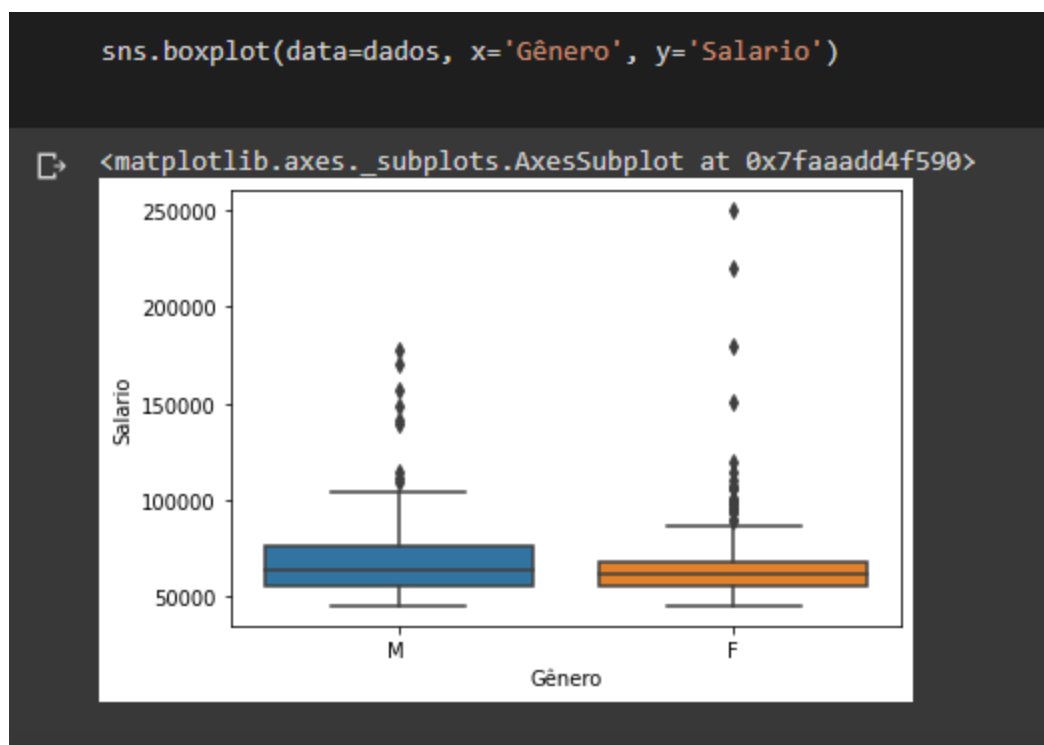


Fazendo análises e hipóteses com o BoxPlot

Analisando esse boxplot, percebemos que a maior parte dos dados estão um pouco acima dos 50000 e antes dos 100000. Se analisarmos o histograma gerado com esses mesmos dados (temos um artigo lançado, o link está logo abaixo), percebemos que as informações se complementam. Temos poucos dados (outliers) que estão acima de 100000 por ano.

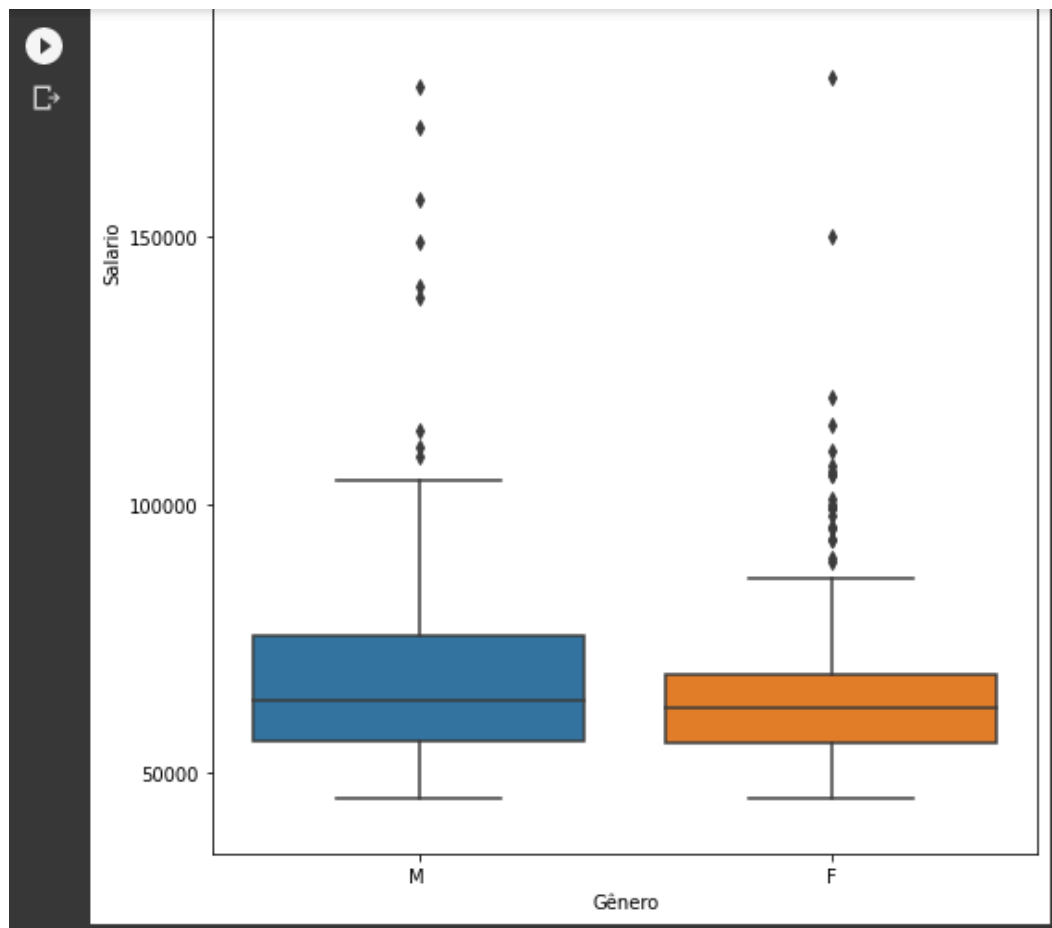
Nós podemos ir além no uso dos boxplots. Podemos utilizar mais de um boxplot com diferentes **categorias** (colunas) para fazer análises de **comparação** entre elas.

Vamos fazer um experimento: analisar a que tipos de conclusões podemos chegar comparando o salário anual com o gênero do(a) funcionário(a). Para isso, vamos gerar o boxplot com o seguinte código:



Como na categoria do gênero feminino temos mais outliers, o gráfico ficou um pouco achatado. Vamos utilizar o matplotlib para melhorar um pouco a visualização:

```
plt.figure(figsize=(8, 12))  
sns.boxplot(data=dados, x='Gênero', y='Salario')
```

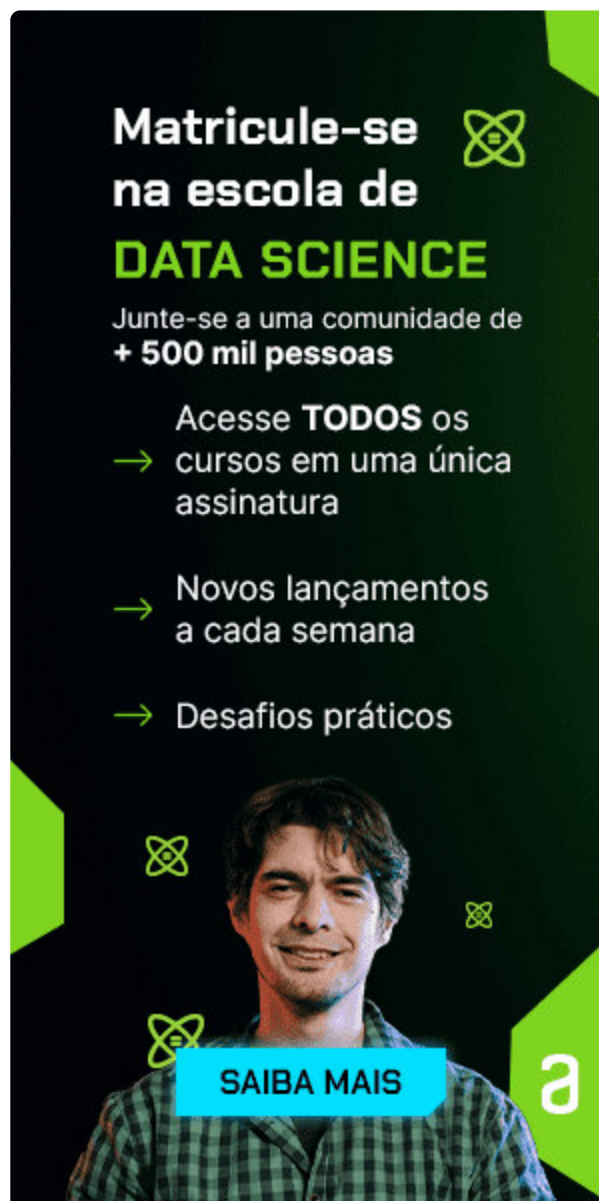


Vamos analisar um pouco esses boxplots. Não conseguimos responder diretamente se os homens ganham mais que as mulheres, ao menos somente com este passo, já que percebemos que a mediana dos dois boxplots está bem próxima uma da outra. Mas conseguimos analisar que o 3º quartil do gênero feminino acaba bem antes do gênero masculino, o que pode nos dizer que conforme o salário anual aumenta temos uma maior concentração de pessoas do gênero masculino. Percebemos também que há muito mais outliers, dados atípicos na distribuição, de pessoas do gênero feminino que têm altos salários.

Agora é sua vez! Seguindo esses mesmos passos, analise um conjunto de dados. Você pode analisar, por exemplo, os salários de alguma empresa brasileira. Use o boxplot e nos diga nos comentários quais conclusões você pôde obter.

Quer saber mais sobre o assunto?

Para saber mais sobre o tema, você pode conferir o artigo de [histograma] (<https://www.alura.com.br/artigos/o-que-e-um-histograma>). E também checar a [documentação](#).

A promotional banner for Alura's Data Science school. It features a dark green background with a portrait of a smiling man in a plaid shirt at the bottom. The text is in white and light green. It includes the Alura logo (a stylized atom) and a large white 'a' in the bottom right corner.

Matricule-se na escola de DATA SCIENCE

Junte-se a uma comunidade de **+ 500 mil pessoas**

- Acesse **TODOS** os cursos em uma única assinatura
- Novos lançamentos a cada semana
- Desafios práticos

SAIBA MAIS



Daniel Siqueira

Daniel é instrutor na escola de Dados e professor de Matemática, Física, Química e Inglês. Tem verdadeira paixão em aprender coisas e assuntos novos, e transmitir seus conhecimentos.

[Artigo Anterior](#)

[ML Engineer, o Dev em <T>](#)

[Próximo Artigo](#)

[Otimização de hiperparâmetros](#)



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

ENVIAR

Nossas redes e apps



Institucional

Sobre nós

Carreiras Alura

Para Empresas

Para Sua Escola

Política de Privacidade

Compromisso de Integridade

Termos de Uso

A Alura

Como Funciona

Formações

Plataforma

Depoimentos

Instrutores(as)

Dev em <T>

Luri, a inteligência artificial da Alura

Documentos Institucionais

IA Conference 2025

Status

Cursos imersivos

Certificações

Uma empresa do grupo Alun

Conteúdos

Fale Conosco

Alura Cases

Email e telefone

Imersões

Perguntas frequentes

Artigos

Podcasts

Artigos de educação
corporativa

Imersão Dados com Python

Novidades e Lançamentos

ENVIAR

CURSOS

Cursos de Programação

Lógica | Python | PHP | Java | .NET | Node JS | C | Computação | Jogos | IoT

Cursos de Front-end

HTML, CSS | React | Angular | JavaScript | jQuery

Cursos de Data Science

Ciência de dados | BI | SQL e Banco de Dados | Excel | Machine Learning | NoSQL | Estatística

Cursos de Inteligência Artificial

IA para Programação | IA para Dados

Cursos de DevOps

AWS | Azure | Docker | Segurança | IaC | Linux

Cursos de UX & Design

Usabilidade e UX | Vídeo e Motion | 3D

Cursos de Mobile

Flutter | iOS e Swift | Android, Kotlin | Jogos

Cursos de Inovação & Gestão

Métodos Ágeis | Softskills | Liderança e Gestão | Startups | Vendas

CURSOS UNIVERSITÁRIOS FIAP

Graduação | Pós-graduação | MBA