**Tópico 2 – Gráficos de dispersão para mais de duas variáveis**

**OBJETIVOS**

* Aprender a criar e modificar gráficos de dispersão para representar variáveis categóricas, utilizando seaborn;
* Aprender a criar e modificar gráficos de dispersão para representar mais de duas variáveis quantitativas.

Quando os conjuntos de dados possuem múltiplas variáveis, pode ser útil visualizar várias delas na mesma visualização. Portanto, considere o conjunto de dados de gorjetas. Até aqui, você já descobriu que existe uma relação geral entre o valor da conta e o da gorjeta. Entretanto, digamos que o dono do restaurante deseja saber se essa relação se mantém para as gorjetas dadas somente por mulheres ou somente por homens, e se existe algum dos sexos que seja mais generoso nas gorjetas. Uma forma de abordar isso seria fazer uma seleção dos dados e gerar duas visualizações separadas, uma para cada gênero. Porém, isso dificultaria a comparação.

Uma melhor abordagem é visualizar os dois no mesmo gráfico de dispersão. Isso pode ser feito utilizando outros atributos visuais, como a cor ou a forma dos pontos, para distinguir variáveis categóricas. Além disso, existe uma terceira variável categórica referente ao tamanho da mesa. Assim, três variáveis quantitativas podem ser representadas no mesmo gráfico de dispersão, mapeando uma delas para uma escala de cor ou para o tamanho dos pontos. Esse tipo de gráfico também é chamado de gráfico de bolhas.

Neste tópico, você aprenderá a criar visualizações, que representam múltiplas variáveis em um gráfico de dispersão, e como criar gráficos de bolhas, utilizando o seaborn.

Vamos tentar, primeiro, responder à pergunta quanto aos comportamentos de cada gênero em relação à gorjeta. Ao visualizar variáveis categóricas em um gráfico de dispersão, é muito útil utilizar a cor para distinguir as categorias. A diferença entre cores bem escolhidas ressalta na nossa percepção visual, tornando padrões evidentes. Portanto, para mapear uma variável para a cor dos pontos na função scatterplot(), basta passar o nome da coluna correspondente ao parâmetro hue (matiz). Confira, no exemplo a seguir, como colorir os pontos de acordo com o sexo:

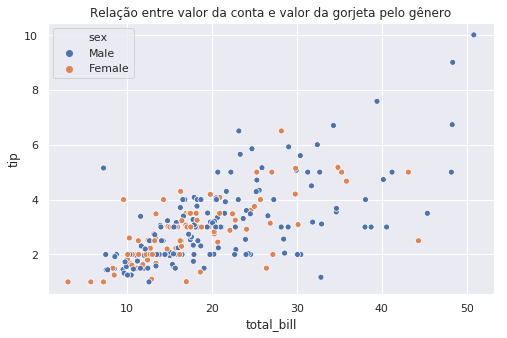
1 plt.figure(figsize=(8,5))

2 plt.title('Relação entre valor da conta e valor da gorjeta pelo gênero')

3 sns.scatterplot(x="total\_bill", y="tip", hue = 'sex', data=tips)

4 plt.show()

Download do código sem numeração no link a seguir: [Download código](https://leadfortaleza.com.br/ead2pcd/conteudo/tmp/myopenolat_1_102915159613561/aula/code/codet2p2s11.txt)

Figura 4 – Gráfico de dispersão com categorias

Perceba que a única mudança, em relação ao que já foi feito, é a passagem da coluna ‘sex’ para o parâmetro hue.

De forma bem geral, tanto homens quanto mulheres possuem uma tendência similar a dar maiores gorjetas, quando a conta for maior. Além disso, essa visualização permite descobrir alguns aspectos interessantes sobre o comportamento de cada gênero nesse conjunto de dados. Quais observações você pode fazer?

O uso de cores deve sempre ser feito com cuidado, pois, quando mal utilizadas, as cores podem atrapalhar a comunicação da informação desejada. Por exemplo, as cores podem ser difíceis de distinguir entre si. No seaborn, por padrão, são utilizadas paletas de cores pré-estabelecidas, facilitando a visualização. Assim, note que azul e laranja possuem um contraste bastante atrativo, facilitando a percepção das categorias.

Um cuidado que deve-se ter ao utilizar cores é quando existem muitas categorias. Nesse caso, haverá uma grande quantidade de cores, deixando a visualização poluída e confusa. Confira, no exemplo a seguir, onde o dia da semana é mapeado para a cor:

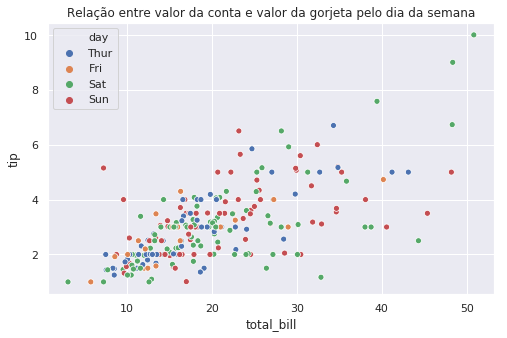
1 plt.figure(figsize=(8,5))

2 plt.title('Relação entre valor da conta e valor da gorjeta pelo dia da semana')

3 sns.scatterplot(x="total\_bill", y="tip", hue = 'day', data=tips)

4 plt.show()

Download do código sem numeração no link a seguir: [Download código](https://leadfortaleza.com.br/ead2pcd/conteudo/tmp/myopenolat_1_102915159613561/aula/code/codet2p3s12.txt)

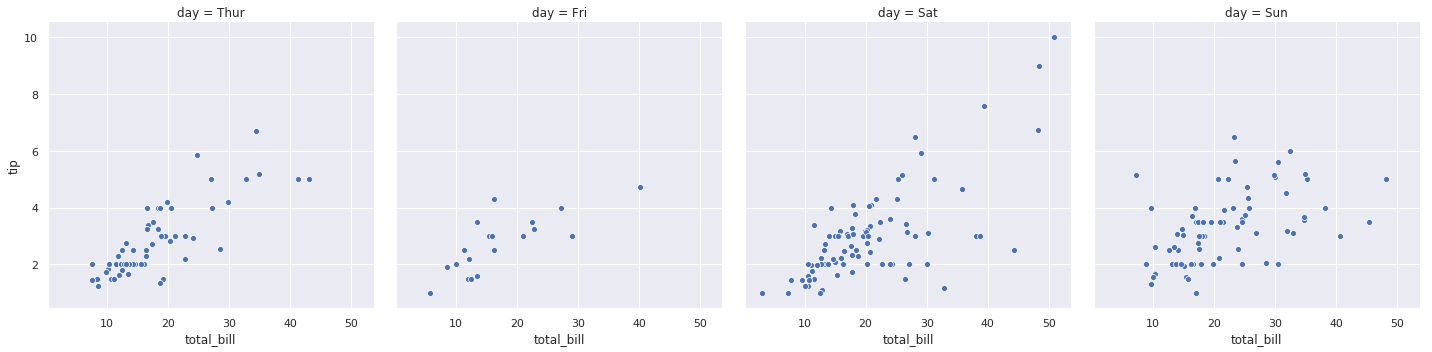
Figura 5 – Gráfico de dispersão com 4 categorias

Apesar de haver apenas 4 dias (quinta-feira a domingo), a sobreposição de cores traz muita informação para a visualização, e a percepção dos padrões existentes exige certo esforço. Idealmente, uma visualização deve ser bastante clara e deve deixar os padrões evidentes. Porém, isso nem sempre será possível. Então, você deverá avaliar, em cada caso particular, se colocar muita informação em uma visualização vale a pena, de acordo com o objetivo dessa visualização.

Nesse caso em particular, uma abordagem mais adequada é criar uma visualização separada para cada dia da semana, o que pode ser feito facilmente com a função relplot(), sobre a qual você deve saber mais a seguir.

A função relplot() é uma versão mais geral da scatterplot(), mas que é baseada em uma classe do seaborn chamada FacetGrid, que permite criar grades de visualizações. Para isso, basta passar o nome da coluna que se deseja separar em grade para o parâmetro col. De resto, ela é muito similar à função scatterplot(), com a diferença de que não é possível alterar as configurações visuais da visualização através das funções do matplotlib, devido ao FacetGrid possuir suas próprias configurações visuais. Devido a isso, observe, no exemplo a seguir, como criar uma grade de visualizações, onde cada visualização é um dia da semana:

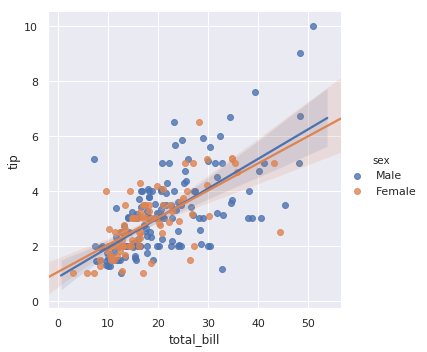
1 sns.relplot(x="total\_bill", y="tip", col = 'day', data=tips)

Figura 6 – Gráficos de dispersão em grade

Verifique o resultado que permite uma observação mais clara do padrão de consumo e da gorjeta em cada dia. Note, por exemplo, que a [correlação](https://leadfortaleza.com.br/ead/glossary/Correlacao) entre a conta e a gorjeta é mais forte no sábado e mais fraca no domingo, o que é um comportamento interessante. Por que você acha que isso ocorre? Talvez seja porque o sábado é um dia mais descontraído ou de festa. Mas, se quisermos deixar a [correlação](https://leadfortaleza.com.br/ead/glossary/Correlacao) entre os grupos mais explícita com uma reta de dispersão? É simples. Confira a seguir!

No gráfico de dispersão, onde visualizamos as diferenças entre homens e mulheres, talvez não tenha ficado evidente a correlação que existe em cada grupo. Uma forma de dar ênfase a essa comparação é através de retas de regressão. Todavia, a já estudada função regplot() não possui o parâmetro hue. Porém, existe uma função, baseada no FacetGrid, que permite fazer isso. Ela é chamada de lmplot(), que é uma forma mais geral do regplot(). Ou seja, você pode utilizá-la com os mesmos parâmetros passados para a função scatterplot(), lembrando que, como ela é baseada em FacetGrid, não pode ser utilizada em conjunto com as funções de título ou tamanho do matplotlib, por exemplo. Confira, no exemplo a seguir, como criar uma reta de regressão para o sexos masculino e feminino:

1 sns.lmplot(x="total\_bill", y="tip", hue = 'sex', data=tips)

Figura 7 – Retas de regressão para cada categoria

Agora, pode-se observar objetivamente que as retas de regressão são muito similares, indicando que a tendência que existe entre valor da conta e gorjeta é praticamente a mesma para ambos os sexos. A seguir, confira uma terceira variável quantitativa chamada size.

A variável quantitativa size representa o tamanho das mesas (mesa para 1, para 2 etc.). É possível visualizar ela em conjunto com o valor da conta e o da gorjeta? Vamos lá! Existem, basicamente, duas formas de fazer isso: através de uma escala de cor, e através do tamanho dos pontos. Para mapear uma variável quantitativa para a cor, é idêntico ao que foi feito ao mapear uma variável categórica para cor: basta, na passagem de parâmetros da função, passar o nome da coluna desejada para o parâmetro hue. A diferença é que, se você passar uma variável quantitativa, o seaborn irá utilizar uma escala de cor para representá-la. Confira, no exemplo a seguir, como visualizar o tamanho da mesa pela cor:

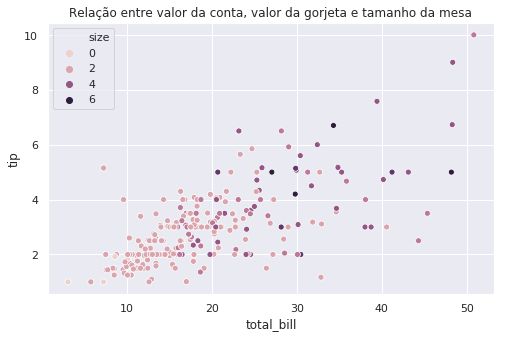
1 plt.figure(figsize=(8,5))

2 plt.title('Relação entre valor da conta, valor da gorjeta e tamanho da mesa')

3 sns.scatterplot(x="total\_bill", y="tip", hue = 'size', data=tips)

4 plt.show()

Download do código sem numeração no link a seguir: [Download código](https://leadfortaleza.com.br/ead2pcd/conteudo/tmp/myopenolat_1_102915159613561/aula/code/codet2p6s15.txt)

Figura 8 – Gráfico de dispersão com 3 variáveis quantitativas, mapeando uma variável quantitativa para a cor

Note que a única mudança no código é a passagem da coluna ‘size’ para o parâmetro hue da função scatterplot() na linha 3.

Na paleta padrão de cores do seaborn, os valores mais baixos são mapeados em cores mais claras, e os mais altos em cores mais escuras.

Essa visualização já permite notar uma certa [correlação](https://leadfortaleza.com.br/ead/glossary/Correlacao) positiva entre o tamanho da mesa, o valor da conta e o valor da gorjeta, pois os pontos mais claros tendem a ter menores contas, e os pontos mais escuros, maiores contas e gorjetas. A desvantagem desse tipo de visualização é que é difícil fazer comparações diretas, pois ela representa variáveis quantitativas em escalas diferentes: posições e cor.

Uma outra maneira de representar essa variável é através do tamanho dos círculos, de modo que quanto maior o círculo, maior o tamanho da mesa. Para isso, a função scatterplot() possui o parâmetro size. No exemplo a seguir, a coluna ‘size’ será passada ao parâmetro size, para mapear o tamanho das mesas no tamanho dos círculos. Confira:

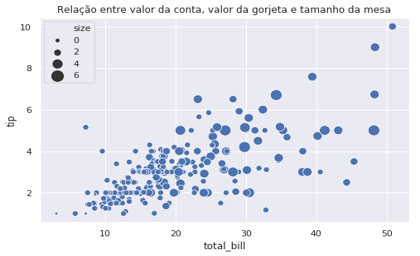
1 plt.figure(figsize=(8,5))

2 plt.title('Relação entre valor da conta, valor da gorjeta e tamanho da mesa')

3 sns.scatterplot(x="total\_bill", y="tip", size = 'size', sizes=(15, 200), data=tips)

4 plt.show()

Download do código sem numeração no link a seguir: [Download código](https://leadfortaleza.com.br/ead2pcd/conteudo/tmp/myopenolat_1_102915159613561/aula/code/codet2p7s16.txt)

Figura 9 – Gráfico de dispersão com 3 variáveis quantitativas, mapeando uma variável quantitativa para o tamanho

As linhas 1 e 2 configuram o tamanho e título da imagem, como já feito anteriormente. Na linha 3, são utilizados dois novos parâmetros da função scatterplot(): size e sizes. O parâmetro size controla quais valores serão mapeados para o tamanho dos círculos e recebe a coluna ‘size’ (o nome do parâmetro é o mesmo nome da coluna, mas isso é apenas uma coincidência). Já o parâmetro sizes estabelece a escala de valores, em unidades de área, que os círculos irão assumir. Nesse exemplo, é passada a faixa de 15 a 200, indicando que o menor tamanho de mesa será representado por um círculo de área 15, enquanto o maior valor de mesa será representado por um círculo de área 200. Se você não especificar o valor de sizes, o seaborn estabelece uma escala automaticamente. Experimente não passar qualquer valor para sizes e compare o resultado. Experimente também outras faixas de valores diferentes entre 15 e 200.

Ah, esse tipo de gráfico é conhecido como **gráfico de bolhas**, pois ele sofre da mesma desvantagem que o gráfico de dispersão anterior, no qual o size foi mapeado para uma escala de cor. Por isso, é difícil fazer comparações entre os tamanhos das mesas com o valor da conta e da gorjeta. Apesar disso, ele é uma visualização interessante que permite uma percepção geral do padrão, sem depender de cores, o que pode ser desejado em certos casos.

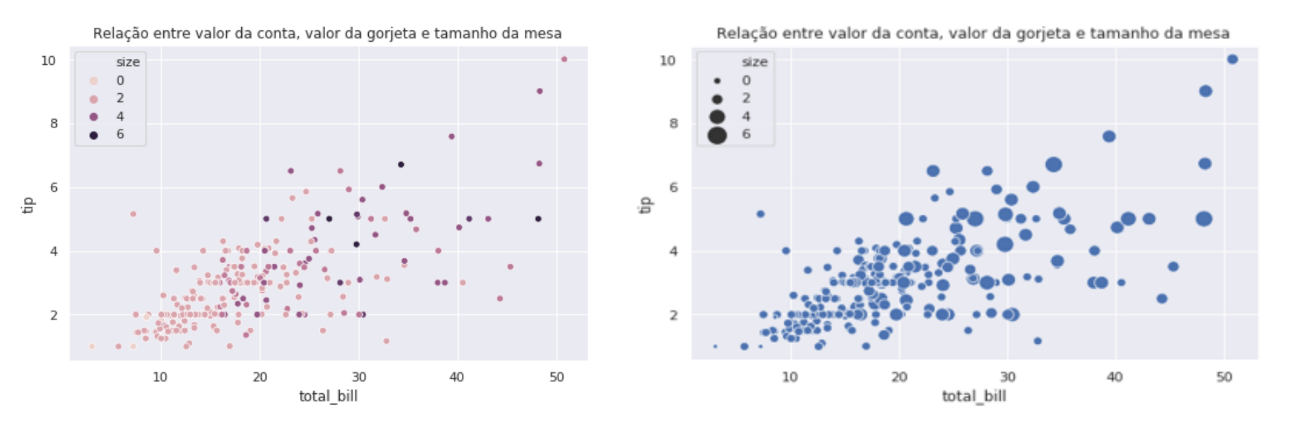
Você achou mais fácil perceber o padrão do tamanho das mesas pela cor ou pelo tamanho dos círculos? Esse resultado pode ser bastante subjetivo. A seguir, confira algumas dicas no box, depois Compare os gráficos na figura.

#### Ícone Fique Atento!



Quando os dados são muito amontoados, o gráfico de bolhas pode ser confuso e poluído, enquanto as cores são mais fáceis de ler. De forma inversa, quando os dados são mais dispersos, o tamanho pode destacar mais o padrão procurado do que a cor. Dessa forma, perceba que, nesses dados das gorjetas, as duas situações ocorrem, pois os pontos se aglomeram do lado esquerdo para o centro e se dispersam no lado direito, e cada visualização enfatiza uma situação.

Assim, como em muitas outras escolhas, em visualização de dados, não existe uma regra rígida de quando se deve utilizar a cor ou o gráfico de bolhas, dependendo do foco da visualização, do público a quem é destinada e dos próprios dados. Quando você estiver em dúvida sobre qual visualização utilizar, sempre teste e experimente diferentes técnicas.

Figura 10 – Comparação entre mapear uma terceira variável para a cor e para o tamanho em um gráfico de dispersão

Por fim, talvez você tenha se perguntado, é possível utilizar todas as técnicas de visualizações vistas nesta aula em conjunto, para criar uma visualização que representa múltiplas variáveis do conjunto de dados? A resposta é sim! Basta passar todas as colunas desejadas para os parâmetros da função scatterplot.

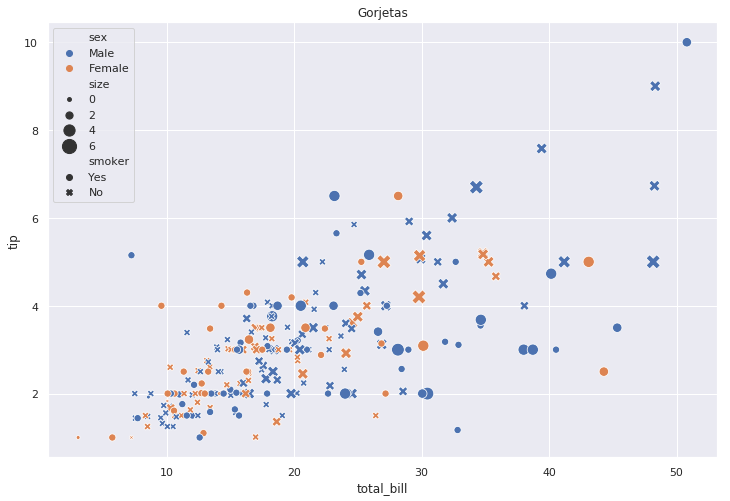
1 plt.figure(figsize=(12,8))

2 plt.title('Gorjetas')

3 sns.scatterplot(x="total\_bill", y="tip", size = 'size', hue = 'sex', style = 'smoker', sizes=(15, 200), data=tips)

4 plt.show()

Download do código sem numeração no link a seguir: [Download código](https://leadfortaleza.com.br/ead2pcd/conteudo/tmp/myopenolat_1_102915159613561/aula/code/codet2p9s18.txt)

Figura 11 – Gráfico de dispersão com múltiplos atributos

As linhas 1 e 2 configuram o tamanho e título da imagem, como já feito anteriormente. Na linha 3, a função scatterplot() recebeu a coluna ‘total\_bill’ para o parâmetro x; a coluna ‘tip’ para o parâmetro y; a coluna ‘size’ para o parâmetro size; a coluna ‘sex’ para o parâmetro hue; e a coluna ‘smoker’ para o parâmetro style. Ou seja, o valor da conta foi mapeado para a posição horizontal, a gorjeta para a posição vertical, o tamanho da mesa para o tamanho do ponto, o sexo do cliente para a cor, e se ele é fumante ou não para a forma. Assim, quase todas as variáveis do conjunto de dados foram representadas em uma única visualização.

O que você achou do resultado? Consegue identificar padrões? Embora seja tentador, raramente, é uma boa prática colocar informação em excesso em uma única visualização, pois ela se torna poluída e confusa, como foi explicado anteriormente. Portanto, ao mostrar tudo, você estará, na verdade, focando em nada, tornando a visualização pouco útil. Por isso, atente-se sempre aos resultados. Pesquise e experimente, pois cada caso é um caso!

Chegamos ao fim do conteúdo! Você estudou o relacionamento entre variáveis quantitativas, conheceu o conceito de [correlação](https://leadfortaleza.com.br/ead/glossary/Correlacao) entre variáveis, um dos conceitos mais importantes em análise de dados. Além disso, aprendeu a criar gráficos de dispersão e suas diversas variações, além de interpretar e analisar o resultado desses gráficos.

Agora você possui ferramentas bastante poderosas para explorar conjuntos de dados e descobrir relações entre variáveis. Assim, sempre que possível, procure exercitar esse conhecimento, explorando dados e testando técnicas de visualização que você aprendeu. Ah, lembre-se sempre de se divertir enquanto isso!

Espero que você tenha aprendido bastante! Até a próxima.

### Referências

* Bussab, Wilton de O.; Morettin, Pedro A. **Estatística Básica**. 6ª ed. [S.l.]: Saraiva. 2010.
* Matplotlib. <https://matplotlib.org/index.html>. Acessado pela última vez em Maio de 2020.
* Seaborn. <https://seaborn.pydata.org/tutorial.html>. Acessado pela última vez em Maio de 2020.
* WILKE, Claus O. **Fundamentals of Data Visualization**. O’Reilly, 2019.