## Tópico 2 – Gráficos de Barras Agrupadas

**OBJETIVOS**

* Aprender a criar gráficos de barras agrupadas para comparar diferentes valores em categorias;
* Entender como se utiliza paletas de cores da biblioteca [seaborn](https://leadfortaleza.com.br/ead/glossary/seaborn" \t "_blank) para configurar o design de gráficos de barras.

Muitas vezes queremos não apenas conhecer valores quantitativos associados a uma categoria, mas comparar valores entre categorias distintas ou comparar diferentes valores dentro da mesma categoria. Tome o conjunto de dados de filmes que estivemos explorando como exemplo. Quantos filmes foram feitos de cada categoria em cada ano? Qual foi a média de orçamento por gênero em cada ano? Essas informações podem ser visualizadas com gráficos de barras agrupadas. Assim, neste tópico, você aprenderá a criar esses gráficos utilizando [seaborn](https://leadfortaleza.com.br/ead/glossary/seaborn) e também entenderá como deve utilizar as paletas de cores do [seaborn](https://leadfortaleza.com.br/ead/glossary/seaborn) para controlar as cores das visualizações quando a cor for um fator importante.

Em gráficos de barras simples, cada posição ao longo de um eixo representa uma categoria e uma barra é desenhada mapeando um valor para essa categoria. Em gráficos de barras agrupadas, as categorias também estão dispostas ao longo de um eixo, mas existe mais de uma barra para cada categoria. Essas barras em cada categoria podem ser distinguidas pela cor e representam outro atributo categórico.

Para gerar esse tipo de gráfico no [seaborn](https://leadfortaleza.com.br/ead/glossary/seaborn), é muito simples. Basta atribuir uma categoria ao parâmetro [hue](https://leadfortaleza.com.br/ead/glossary/hue) nas funções que você já utilizou. [Hue](https://leadfortaleza.com.br/ead/glossary/hue) significa matiz de uma cor. Ou seja, a nova categoria será mapeada na cor das barras. Confira o exemplo de código a seguir e o resultado de sua execução.

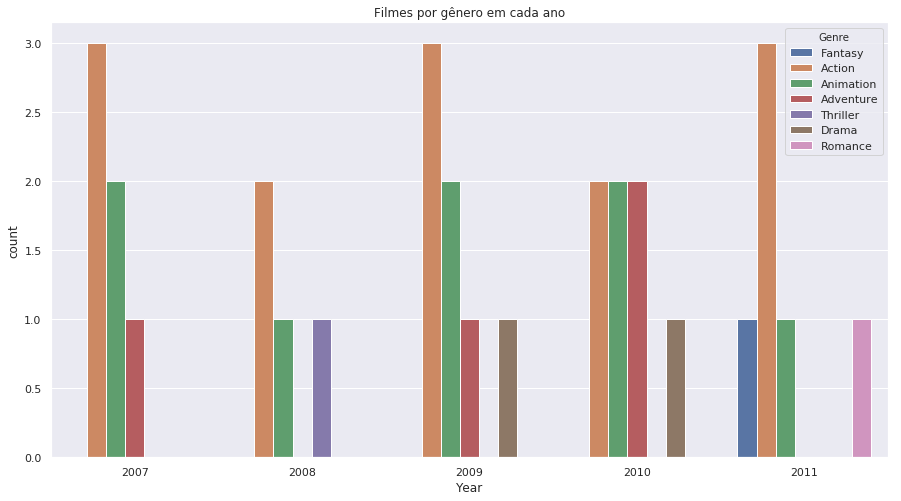
1 plt.figure(figsize=(15,8))

2 plt.title('Filmes por gênero em cada ano')

3 sns.countplot(x="Year", hue= 'Genre', data=filmes)

4 plt.show()

Download do código sem numeração no link a seguir: [Download código](https://leadfortaleza.com.br/ead2pcd/conteudo/tmp/myopenolat_1_103154023698583/aula/code/codet2p2s21.txt)

Figura 10 – Filmes por gênero em cada ano

E então, você achou simples? As linhas 1 e 2 configuram o tamanho e o título da figura, como já feito anteriormente. Na linha 3 é chamada a função countplot() de maneira muito similar ao que já foi feito, passando o valor ‘Year’ para o parâmetro x. Ou seja, os anos serão mapeados na posição do eixo x. A novidade é o parâmetro [hue](https://leadfortaleza.com.br/ead/glossary/hue), que recebe a coluna ‘Genre’, significando que o gênero dos filmes será mapeado nas cores das barras.

Note algumas particularidades dos dados: só existe um filme de fantasia e um de romance nesses dados, e são de 2011. O filme de suspense é de 2008. A quantidade de filmes de ação varia de dois a três a cada ano. E se você quiser visualizar a média de bilheteria por gênero em cada ano?

O parâmetro [hue](https://leadfortaleza.com.br/ead/glossary/hue) também pode ser utilizado na função barplot(). Então para criar essa visualização é análogo ao que foi feito com a função countplot(). Confira no exemplo:

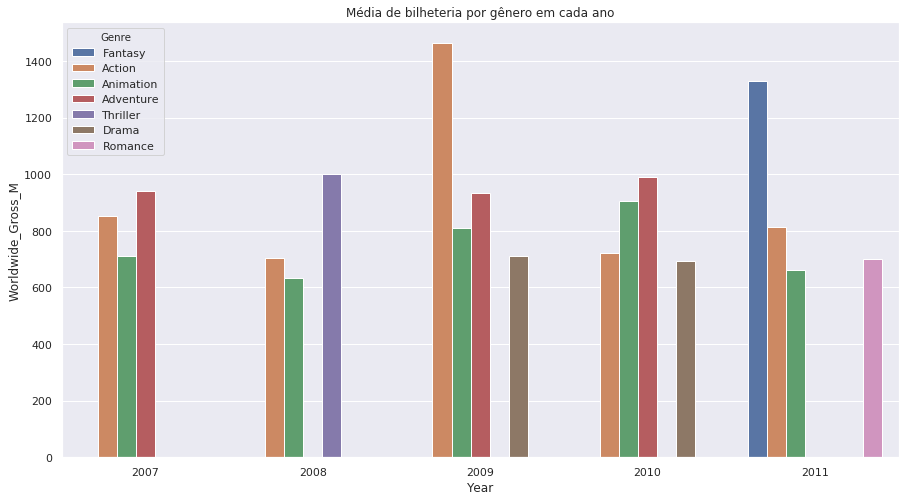
1 plt.figure(figsize=(15,8))

2 plt.title('Média de bilheteria por gênero em cada ano')

3 sns.barplot(x="Year", y = 'Worldwide\_Gross\_M' , ci = None, hue= 'Genre', data=filmes)

4 plt.show()

Download do código sem numeração no link a seguir: [Download código](https://leadfortaleza.com.br/ead2pcd/conteudo/tmp/myopenolat_1_103154023698583/aula/code/codet2p3s22.txt)

Figura 11 – Média de bilheteria por gênero em cada ano

A mudança nesse código em relação ao anterior ocorre somente na linha 3. Note que, na linha 3, a função barplot() é chamada passando como parâmetro x o ‘Year’ e o parâmetro y como o ‘Worldwide\_Gross\_M’, de forma similar ao já feito anteriormente. A grande diferença é que é passado o atributo ‘Genre’ para o parâmetro [hue](https://leadfortaleza.com.br/ead/glossary/hue).

Essa visualização esclarece mais o comportamento das bilheterias. Lembre que no geral a bilheteria de filmes de fantasia era maior do que os de ação. Mas isso ocorreu porque só existe um filme de fantasia, que obteve uma boa bilheteria. Já os filmes de ação, que são maioria nesse conjunto de dados, tiveram um desempenho variável ao longo dos anos, mas tiveram em média uma grande bilheteria no ano de 2009. Que outras considerações você pode fazer a partir dessa visualização?

Note que quando você utilizou a barras simples, a cor era apenas uma opção estética. Em barras agrupadas, a cor possui valor informativo, pois é usada para distinguir categorias. É preciso escolher cuidadosamente as cores utilizadas, para que a informação seja passada de forma harmoniosa. Alterar o conjunto de cores utilizado, ou paleta de cores, é simples no [seaborn](https://leadfortaleza.com.br/ead/glossary/seaborn), bastando alterar o valor do parâmetro palette. O [seaborn](https://leadfortaleza.com.br/ead/glossary/seaborn) disponibiliza diversas paletas de cores, que podem ser usadas tanto para distinguir como para ressaltar características de interesse. Confira no exemplo a seguir como alterar a paleta de cores da visualização:

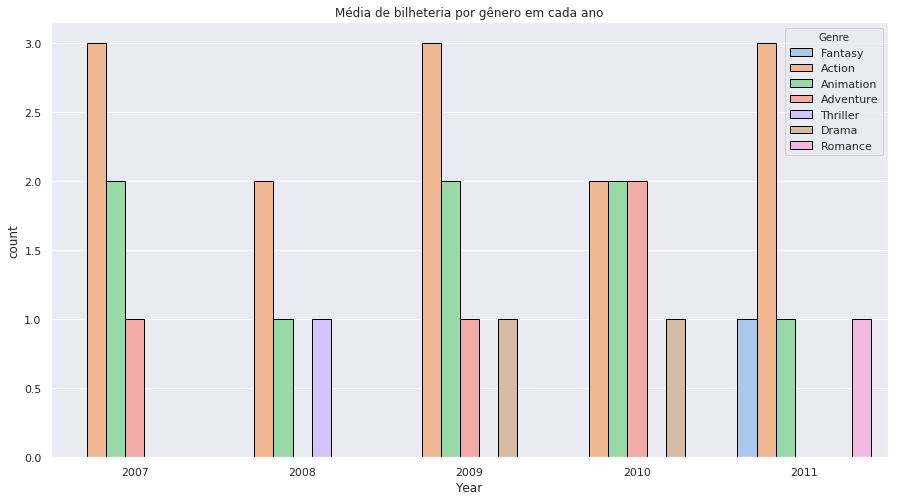
1 plt.figure(figsize=(15,8))

2 plt.title('Média de bilheteria por gênero em cada ano')

3 sns.countplot(x="Year", hue= 'Genre', palette = "pastel", edgecolor="0.4", data=filmes)

4 plt.show()

Download do código sem numeração no link a seguir: [Download código](https://leadfortaleza.com.br/ead2pcd/conteudo/tmp/myopenolat_1_103154023698583/aula/code/codet2p4s23.txt)

Figura 12 – Alterando a [paleta](https://leadfortaleza.com.br/ead/glossary/paleta) de cores

Na linha 2, foi passado o valor ‘pastel’ para o parâmetro palette, resultando em tons pastéis. Você pode conhecer as diversas opções de paleta consultando este link: [https://](https://seaborn.pydata.org/tutorial/color_palettes.html)[seaborn](https://leadfortaleza.com.br/ead/glossary/seaborn).pydata.org/tutorial/color\_palettes.html. Experimente utilizar outras paletas.

Também foi passado o valor 0.4 para o atributo edgecolor, que configura a cor do contorno das barras. Quanto mais próximo de 0, mais preto fica e quanto mais próximo de 1, mais branco. Teste alterar esse valor.

Você notou que existem muitas formas de alterar o design das visualizações criadas com o [seaborn](https://leadfortaleza.com.br/ead/glossary/seaborn)? Conhecer todas as opções possíveis não seria possível em uma única aula, mas você pode explorar essa grande variedade consultando a documentação oficial da biblioteca, disponível neste link: [https://](https://seaborn.pydata.org/api.html)[seaborn](https://leadfortaleza.com.br/ead/glossary/seaborn).pydata.org/api.html.

Por último, você pode desejar salvar os gráficos criados. No Jupyter Notebook você pode simplesmente clicar com o botão direito do mouse, ou usar o botão Aplicação, e escolher a opção “Salvar imagem como…”, assim como você pode salvar qualquer imagem na Web. Porém, você também pode salvar a imagem através do código, automatizando o processo. Basta utilizar a função do [matplotlib](https://leadfortaleza.com.br/ead/glossary/matplotlib) savefig(). Essa função pode receber vários argumentos para configurar o arquivo criado, como extensão, resolução da imagem e outros. Confira, no exemplo de código a seguir, como salvar a última imagem criada em um arquivo com extensão PNG. O código exibe a imagem no programa e, ao mesmo tempo, salva a imagem na mesma pasta do notebook.

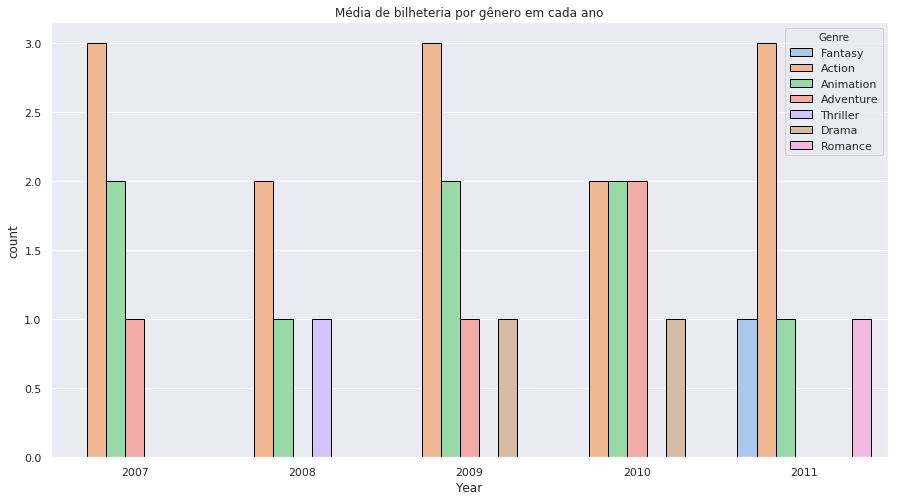
1 plt.figure(figsize=(15,8))

2 plt.title('Média de bilheteria por gênero em cada ano')

3 sns.countplot(x="Year", hue= 'Genre', palette = "pastel", edgecolor="0", data=filmes)

4 plt.savefig('Média de bilheteria por gênero em cada ano.png')

Download do código sem numeração no link a seguir: [Download código](https://leadfortaleza.com.br/ead2pcd/conteudo/tmp/myopenolat_1_103154023698583/aula/code/codet2p5s24.txt)



Note que o único argumento passado foi o nome do arquivo. O arquivo é salvo diretamente na pasta onde está o Jupyter Notebook. Se você desejar, pode especificar no parâmetro da função o caminho completo para salvar o arquivo em qualquer pasta da sua máquina.

Caro aluno, você chegou ao final da aula sobre visualizar quantidades categóricas. Você agora está apto a utilizar comandos básicos das bibliotecas [matplotlib](https://leadfortaleza.com.br/ead/glossary/matplotlib) e seaborn para criar diferentes gráficos de barras sobre diferentes tipos de dados e modificar a aparência desses gráficos para criar melhores visualizações. Existem muitas opções gráficas não estudadas aqui e recomendo que você explore a documentação das bibliotecas para tirar o melhor proveito delas. Além disso, é importante lembrar que as visualizações devem revelar informações sobre os dados, e às vezes é necessária alguma manipulação nesses dados. Por isso, também é importante conhecer as opções existentes na biblioteca pandas para trabalhar com dados e ampliar as suas possibilidades de análise.

Por fim, gráficos de barras são visualizações básicas e muito utilizadas para representar quantidades de dados categóricos, mas existem outros tipos que podem ser úteis em casos específicos, como gráficos de pontos e mapas de calor. Percebeu como são muitas as possibilidades de visualização de dados? Aconselho você a sempre estudar outras possibilidades e experimentá-las em diferentes casos.

Espero que você tenha aprendido e experimentado bastante com este conteúdo. Até a próxima!

### Referências

* WILKE, Claus O. Fundamentals of Data Visualization. O’Reilly, 2019.
* Matplotlib. [https://](https://matplotlib.org/index.html)[matplotlib](https://leadfortaleza.com.br/ead/glossary/matplotlib).org/index.html. Acessado pela última vez em Maio de 2020.
* Seaborn. [https://](https://seaborn.pydata.org/tutorial.html)[seaborn](https://leadfortaleza.com.br/ead/glossary/seaborn).pydata.org/tutorial.html. Acessado pela última vez em Maio de 2020.