

Mãos à obra!

Vamos começar importando o algoritmo `AgglomerativeClustering` da biblioteca `sklearn` :

```
from sklearn.cluster import AgglomerativeClustering
```

Algoritmo importado, vamos criar um modelo e, seguindo o K-Means, vamos falar que queremos agrupar em 17 grupos. Modelo criado, basta falar para ele aprender e retornar os grupos (`fit_predict`):

```
modelo = AgglomerativeClustering(n_clusters=17)
grupos = modelo.fit_predict(generos_escalados)
```

Vamos rodar o algoritmo `TSNE` para visualizar os dados. Para isso, vamos utilizar a mesma abordagem que utilizamos anteriormente. Criar um objeto a partir da classe e pedir para ele transformar nossos gêneros:

```
tsne = TSNE()
visualizacao = tsne.fit_transform(generos_escalados)
```

Bacana! Agora, basta plotar com um gráfico de dispersão passando como parâmetro de cores os grupos criados pelo `AgglomerativeClustering` :

```
sns.scatterplot(x=visualizacao[:, 0],
                y=visualizacao[:, 1],
                hue=grupos)
```

Porém, quando utilizamos o agrupamento hierárquico, uma forma comum de visualização é através do dendrograma. Para plotar o dendrograma, precisamos da matriz de distâncias dos dados e da função que plota o dendrograma propriamente dito. Vamos importá-las da biblioteca `scipy` :

```
from scipy.cluster.hierarchy import dendrogram, linkage
```

Antes de plotar o dendrograma, vamos agrupar novamente os dados com o K-Means para ter uma visualização dos centróides mais próxima a célula do dendrograma (e para sobrescrever as variáveis que usamos com o mesmo nome):

```
modelo = KMeans(n_clusters=17)
modelo.fit(generos_escalados)

grupos = pd.DataFrame(modelo.cluster_centers_,
                      columns=generos.columns)

grupos.transpose().plot.bar(subplots=True,
                             figsize=(25, 50),
                             sharex=False,
                             rot=0)
```

Agora, basta criarmos a matriz de distâncias baseada nos grupos do K-Means e plotar o dendrograma:

```
matriz_de_distancia = linkage(grupos)

dendrograma = dendrogram(matriz_de_distancia)
```