Instituto Federal de Brasília
Campus Taguatinga

Lista de Exercícios 2 (16/14/2025)

Ciência de Dados - 2025/1

Dr. Prof. Raimundo C. S. Vasconcelos
Dr. Prof. Fabiano C. Fernandes

Tales Lima de Oliveira
tales.oliveira@estudante.ifb.edu.br

Disponibilidade do Código

Os `códigos-fonte` e `scripts` desenvolvidos para esta atividade estão disponíveis publicamente e podem ser acessados através do GitHub e do Google Colab.

Código 0: Caminho para os arquivos.

```
###### IMPORTAR DADOS DO DRIVE ######
from google.colab import drive
drive.mount('/content/drive')
path_ecom = '/content/drive/MyDrive/ecommerce_purchases.csv'
path_conc = '/content/drive/MyDrive/conceito_enade_2021.xlsx'

###### IMPORTAR DADOS DA PASTA LOCAL ######
path_ecom = 'data/ecommerce_purchases.csv'
path_conc = 'data/conceito_enade_2021.xlsx'
```

1. Exercício - Compras de Ecommerce

Código 1.1: Importe pandas e leia o arquivo csv Ecommerce Purchases e configure-o para um DataFrame chamado ecom.

```
import pandas as pd
df_ecom = pd.read_csv(path_ecom)
```

Código 1.2: Verifique o **head** do DataFrame.

```
df_ecom.head()
```

Código 1.3: Quantas linhas e colunas existem?

```
rows, columns = df_ecom.shape
print(rows, columns)
```

Código 1.4: Qual é o preço de compra médio?

```
average_price = df_ecom['Purchase Price'].mean()
print(f"${average_price:.2f}")
```

Código 1.5: Quais foram os preços de compra mais altos e mais baixos?

```
max_price = df_ecom['Purchase Price'].max()
min_price = df_ecom['Purchase Price'].min()

print(f"Max: ${max_price:.2f}")
print(f"Min: ${min_price:.2f}")
```

Código 1.6: Quantas pessoas têm Inglês **(en)** como sua língua de escolha no site?

```
num_english_users = (df_ecom['Language'] == 'en').sum()
print(num_english_users)
```

Código 1.7: Quantas pessoas têm o cargo de **Lawyer** (Advogado)?

```
num_lawyers = df_ecom['Job'].str.contains(r'\blawyer\b', case=False, na=False).sum()
# num_lawyers = df_ecom['Job'].str.contains('Lawyer').sum()
print(num_lawyers)
```

Código 1.8: Quantas pessoas fizeram a compra durante a AM e PM?

```
am_pm_counts = df_ecom['AM or PM'].value_counts()
print(am_pm_counts)
```

Código 1.9: Quais são os 5 títulos de trabalho mais comuns?

```
top_5_job = df_ecom['Job'].value_counts().head(5)
print(top_5_job)
```

Código 1.10: Alguém fez uma compra que veio do Lot: **90 WT**, qual foi o preço de compra para esta transação?

```
price_for_lot_90wt = df_ecom[df_ecom['Lot'] == '90 WT']['Purchase Price'].iloc[0]
print(f"${price_for_lot_90wt:.2f}")
```

Código 1.11: Qual é o email da pessoa com o seguinte número do cartão de crédito: **4926535242672853**

```
email_for_card = df_ecom[df_ecom['Credit Card'] == 4926535242672853]['Email'].iloc[0]
print(email_for_card)
```

Código 1.12: Quantas pessoas têm o American Express como seu fornecedor de cartão de crédito e fizeram uma compra acima de US $95?

```
num_american_express_above_95 = df_ecom[(df_ecom['CC Provider'] == 'American Express') &
↪  (df_ecom['Purchase Price'] > 95)].shape[0]
print(num_american_express_above_95)
```

Código 1.13: *Difícil:* Quantas pessoas tem um cartão de crédito que expira em 2025?

```
df_ecom['CC Exp Date'] = df_ecom['CC Exp Date'].astype(str)
num_expiring_2025 = df_ecom[df_ecom['CC Exp Date'].str.endswith('25')].shape[0]
print(num_expiring_2025)
```

Código 1.14: *Difícil:* Quais são os 5 principais provedores de e-mail?

```
df_ecom['Email Domain'] = df_ecom['Email'].str.split('@').str[1]
top_5_email_providers = df_ecom['Email Domain'].value_counts().head(5)
print(top_5_email_providers)
```

2. Exercício - Compras de Ecommerce

Código 2.1: Faça uma análise estatística e realize as plotagens que achar pertinente na exploração dos dados.

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
df_conc = pd.read_excel(path_conc)

for c in df_conc.columns:
  print(f"Column: {c}")
```

```
df_group = df_conc[df_conc['Grau Acadêmico'] == 'Bacharelado'].groupby('Sigla da IES*')['Nº de Concluintes Participantes'].count().sort_values(ascending=False)

fig, ax = plt.subplots(layout='constrained')
sns.barplot(
    x = df_group.head(10).index,
    y = df_group.head(10).values
)

plt.title('Nº de Concluintes Participantes (Bacharelado)')
plt.show()
```

```
df_conc.groupby('Modalidade de Ensino')['Nota Padronizada - CE'].mean().plot(kind='bar')
plt.title('Nota média por Modalidade de Ensino')
plt.ylabel('Nota Padronizada - CE')
plt.xticks(rotation=0)
plt.show()
```

```
df_conc.groupby('Categoria Administrativa')['Nota Bruta - FG'].mean().plot(kind="bar")
plt.ylabel('Nota Bruta - FG')
plt.xlabel('Categoria Administrativa')
plt.title('Categoria Administrativa x Nota Bruta - FG')
plt.xticks(rotation=45, ha='right')
plt.show()
```

Código 2.2: Construa um modelo de clusterização e analise os resultados encontrados.

```
import matplotlib.pyplot as plt
import matplotlib.patches as mpatches

import seaborn as sns
import pandas as pd

from sklearn.preprocessing import StandardScaler
from sklearn.cluster import DBSCAN
```

```
# Seleção e limpeza dos dados
features = df_conc[['Nota Padronizada - FG', 'Nota Padronizada - CE', 'Nº de Concluintes Participantes']]\
    .head(500)\
    .sort_values('Nº de Concluintes Participantes', ascending=True)\
    .dropna()
```

```
# Normalização
scaler = StandardScaler()
X_scaled = scaler.fit_transform(features)
```

```python
# DBSCAN
dbscan = DBSCAN(eps=0.5, min_samples=5)
clusters = dbscan.fit_predict(X_scaled)
features['Cluster'] = clusters
```

Como interpretar este gráfico?

Insight 1: Relação positiva

- Há uma correlação positiva clara entre Nota Padronizada - FG e Nota Padronizada - CE.
- Conforme uma aumenta, a outra tende a aumentar também.

Insight 2: Agrupamentos distintos

- O DBSCAN conseguiu identificar pelo menos dois grupos principais com comportamentos semelhantes:
    1. Grupo mais à direita: alunos com notas altas em ambas as provas.
    2. Grupo mais à esquerda: alunos com notas um pouco mais baixas.

Insight 3: Outliers

- Os pontos classificados como `-1` são outliers.
- Podem representar cursos ou instituições com desempenho muito diferente dos demais (para cima ou para baixo), ou com poucos concluintes.

Insight 4: Concluintes

- O tamanho das bolhas mostra que há variação significativa no número de participantes.
- Pontos grandes indicam instituições/cursos mais relevantes numericamente.
- Eles ajudam a destacar quais grupos representam mais pessoas e podem ser mais relevantes para análises políticas ou institucionais.