

Aula 3 - Analisando campanhas de marketing com pandas


Objetivos de ensino

Nesta aula, você revisará os conceitos básicos do Pandas, incluindo:

- Importação de conjuntos de dados
- Análise exploratória e
- Plotagem de dados.

Recursos utilizados

1. Baixe o dataset a seguir e salve no diretório do seu projeto.

 marketing.csv 803.8KB

1. Importação do Dataset

- Instruções
 - a. importe `pandas` no seu ambiente com o alias `pd`
 - b. carregue o dataset `marketing.csv` em um DataFrame.

```
# Importe o pandas no ambiente python import ----- # carregamento do  
arquivo marketing.csv marketing = ---.-----.(-----)
```

2. Examinando os dados

Agora que você importou os dados, você irá examiná-los para garantir que são válidos. Embora possa ser tentador mergulhar na análise, ao validar seu conjunto de dados imediatamente, você garante que os insights obtidos sejam verdadeiros e acionáveis. Garantir a validade do seu conjunto de dados deve sempre ser o primeiro passo ao iniciar a análise de um novo conjunto de dados.

O DataFrame `marketing` está disponível em seu ambiente de trabalho.

- **Instruções**

- Imprima as primeira 5 linhas do DataFrame `marketing`.
- Imprima um resumo estatístico de todas as colunas do DataFrame `marketing`. Utilize `describe` e o parâmetro `include='all'`.
- Imprima tipos de dados das colunas e a quantidade de valores não-nulos por coluna do DataFrame `marketing`. Utilize `info`

```
# a) Imprima as primeiras 5 linhas do DataFrame marketing print(____.____())  
# b) Imprima um resumo estatístico de todas as colunas do DataFrame  
marketing print(____.____()) # c) Imprima tipos de dados das colunas e a  
quantidade de valores não-nulos por coluna print(____.____())
```

3. Tipos de dados e mesclagem

Tipos de dados de uma coluna

```
# Print a data type of a single column  
print(marketing['converted'].dtype)
```

```
dtype('object')
```

Mudando tipo de dados de uma coluna

```
# Change the data type of a column  
marketing['converted'] = marketing['converted']\  
                        .astype('bool')  
print(marketing['converted'].dtype)
```

```
dtype('bool')
```

Criando novas colunas do tipo Bool

```
marketing['is_house_ads'] = np.where(
    marketing['marketing_channel'] == 'House Ads',
    True, False
)

print(marketing.is_house_ads.head(3))
```

```
0    True
1   False
2     True
Name: is_house_ads, dtype: bool
```

Mapeando valores às colunas existentes

```
channel_dict = {"House Ads": 1, "Instagram": 2,
               "Facebook": 3, "Email": 4, "Push": 5}
marketing['channel_code'] = marketing['marketing_channel']\
    .map(channel_dict)

print(marketing['channel_code'].head(3))
```

```
0    1
1    1
2    1
Name: channel_code, dtype: int64
```

Colunas do tipo Data

```
# Read date columns using parse_dates
marketing = pd.read_csv('marketing.csv',
                       parse_dates=['date_served',
                                   'date_subscribed',
                                   'date_canceled'])

# Or
# Convert already existing column to datetime column
marketing['date_served'] = pd.to_datetime(
    marketing['date_served']
)
```

```
marketing['day_served'] = marketing['date_served']\
    .dt.dayofweek
```

Atualizando o tipo de dados de uma coluna

Frequentemente você vai querer modificar o tipo de dados das colunas para atender às suas necessidades. Pandas e Python têm funcionalidades especiais para manipular tipos de dados que podem tornar sua análise mais fácil e robusta, como concatenar strings e somar valores booleanos. No entanto, essa funcionalidade só é possível se as colunas estiverem armazenadas com o tipo de dado pretendido.

Neste exercício, você irá modificar o tipo de dados da coluna `is_retained`.

- Instruções

- Imprima o tipo de dados da coluna `is_retained`.

- b. Atualize o tipo de dados da coluna `is_retained` para booleano e então verifique seu tipo de dados novamente.

```
# Verifique o tipo de dado da coluna is_retained
print(marketing['is_retained'].dtype) # Converta is_retained para booleano
marketing['is_retained'] = marketing['is_retained'].astype('bool') #
Verifique novamente o tipo de dado da coluna is_retained
print(marketing['is_retained'].dtype)
```

Adicionando novas colunas

Adicionar novas colunas que derivam informações de dados existentes ou com base em conhecimento do domínio é conhecido como *Engenharia de Features*. Mesmo em conjuntos de dados relativamente simples, sempre existem novas características que podem ser extraídas para criar uma análise mais aprofundada.

Uma das habilidades mais importantes que um cientista de dados precisa aprender é como identificar oportunidades para engenharia de features.

Neste exercício, você irá adicionar duas colunas ao `marketing`:

- `channel_code`: representa o valor numérico do canal de inscrição
- `is_correct_lang`: indica se o anúncio foi mostrado ao usuário em seu idioma preferido

- **Instruções**

- a. Add a new column, `channel_code`, which maps the values in the `subscribing_channel` column to a numeric scale using the `channel_dict` dictionary.

```
# Mapping for channels channel_dict = {"House Ads": 1, "Instagram": 2,
"Facebook": 3, "Email": 4, "Push": 5} # Map the channel to a channel code
marketing['channel_code'] =
marketing['subscribing_channel'].map(_____)
```

- b.

- Importe `numpy` com o alias `np`.
- Adicione uma nova coluna, `is_correct_lang`, que será `'Yes'` se o anúncio foi mostrado ao usuário em seu idioma preferido, `'No'` caso contrário. Você pode usar as colunas `'language_preferred'` e `'language_displayed'`.

```
# Import numpy ____ # Add the new column is_correct_lang
marketing['is_correct_lang'] = ____
```

Colunas de data

Atualmente, as colunas de data no DataFrame `marketing` estão sendo incorretamente lidas como objetos. Precisamos converter essas colunas para o tipo data para poder usar as funcionalidades robustas de manipulação e formatação de datas do Python e pandas.

Neste exercício, você praticará a leitura do CSV com as colunas de data adequadas e criará uma coluna para o dia da semana.

Instruções

- Importe `pandas` no ambiente com o alias `pd`.
- Leia o arquivo `'marketing.csv'` no seu ambiente identificando corretamente as colunas de data `'date_served'`, `'date_subscribed'` e `'date_canceled'` dentro da chamada do `read_csv()`.
- Crie uma coluna de dia da semana a partir de `'date_subscribed'` usando `dt.dayofweek`.

```
# Import pandas into the environment ____ # Import marketing.csv with date
columns marketing = ____ # Add a DoW column marketing['Dow'] = ____
```

3. Análise exploratória inicial

Como os usuários visualizam propagandas ou ativos de marketing?

```
# Aggregate unique users that see ads by date
daily_users = marketing.groupby(['date_served'])\
                        ['user_id'].nunique()

print(daily_users)
```

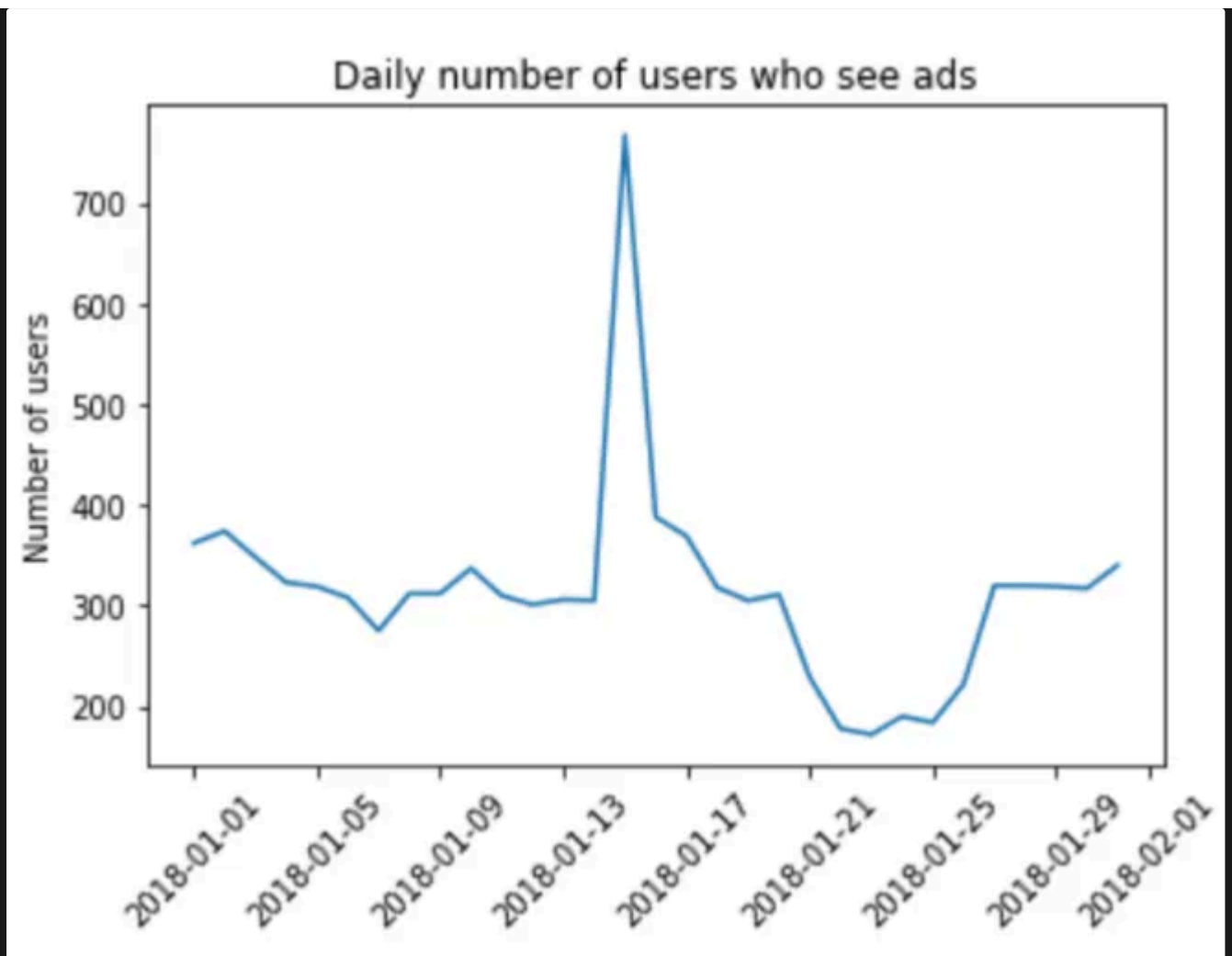
```
date_served
2018-01-01    362
2018-01-02    374
2018-01-03    348
...
Name: user_id, dtype: int64
```

Visualização dos resultados

```
import matplotlib.pyplot as plt

# Plot
daily_users.plot()

# Annotate
plt.title('Daily number of users who see ads')
plt.xlabel('Date')
plt.ylabel('Number of users')
plt.xticks(rotation = 45)
plt.show()
```



Alcance diário de marketing por canal

Neste exercício, você determinará quantos usuários estão visualizando os ativos de marketing a cada dia. Isso é crucial para entender a eficácia de nossos esforços de marketing durante o último mês.

Você usará `.groupby()` e `.nunique()` para calcular os usuários diários.

Instruções

- Agrupe o DataFrame `marketing` por `'date_served'` e conte o número de IDs de usuário únicos presentes na coluna `user_id`.

```
# Group by date_served and count number of unique user_id's
daily_users = marketing.groupby('date_served')['user_id'].nunique()
# Print head of daily_users
print(daily_users.head())
```

Visualizando o alcance diário de marketing

No exercício anterior, você criou um DataFrame `daily_users`, que continha o número de usuários que se inscreveram a cada dia. Embora isso tenha sido um ótimo primeiro passo, é difícil interpretar tendências diárias olhando para uma tabela. Para facilitar para você e seus stakeholders de negócios perceberem tendências de