



64%

ATIVIDADES
2 de 6

DISCORD
ALURA

FÓRUM DO
CURSO

VOLTAR
PARA
DASHBOARD

MODO
NOTURNO



15.4k xp

a

02

Normalização

Transcrição

Continuando nosso estudo, agora analisaremos as vendas de nossa cafeteria fictícia muito conhecida no Brasil, a Cafelura.

Então, criaremos uma célula de tipo texto e, para o título, digitaremos "#Cafelura - Análise de vendas.

Criaremos uma variável chamada `cafelura` para a leitura do arquivo `csv`. Depois, visualizaremos os primeiros registros com o `head`.

```
cafelura = pd.read_csv('cafelura.csv')  
cafelura.head()
```

COPIAR CÓDIGO

O documento terá as colunas `mês` e `vendas`. Conferiremos como o `mês` estará tipado.

```
cafelura.dtypes
```

O `mês` estará do tipo `object`, e alteraremos isso para `datetime64`. Depois de executar, faremos a prova real com outro `cafelura.dtypes`.



64%

ATIVIDADES
2 de 6

DISCORD
ALURA

FÓRUM DO
CURSO

VOLTAR
PARA
DASHBOARD

MODOS
NOTURNOS



15.4k xp

a

```
cafelura['mes'] = pd.to_datetime(cafelura['mes'])
```

Vamos conferir se temos dados nulos e quantos serão os dados nesse arquivo.

```
print('Quantidade de linhas e colunas:', cafelura.shape)
print('Quantidade de dados nulos:', cafelura.isna().sum().sum())
```

COPIAR CÓDIGO

Haverá 24 linhas, 2 colunas e nenhum dado nulo.

Agora faremos um primeiro plot. Mas, em vez de fazer o `diff`, nesse caso faremos o plot apenas de um gráfico. Usaremos a função `plotar` e preencheremos de acordo com a sugestão do próprio programa `plotar(titulo, labelx, labely, x, y, dataset)`. É importante lembrar que o `DataSet` não será uma string.

Queremos plotar as vendas da Cafelura de 2017 e 2018.

```
plotar('Vendas da Cafelura de 2017 e 2018', 'Tempo', 'Vendas', 'mes', 'vendas', cafelura)
```

Vamos rodar esse código, e teremos criado um gráfico interessante que mostrará que a empresa está crescendo. Porém, há alguns pontos diferentes, por exemplo, alguns em que parece ter ocorrido mais vendas. Teremos que conferir se poderemos normalizar esse gráfico.

A normalização é uma prática comum do trabalho com Time Series. Geralmente, a quantidade de dias úteis do mês, de dias da semana ou até dias úteis são técnicas utilizadas para normalizar séries temporais.



64%

ATIVIDADES
2 de 6

DISCORD
ALURA

FÓRUM DO
CURSO

VOLTAR
PARA
DASHBOARD

MODOS
NOTURNOS



15.4k xp

a

Já teremos disponibilizado no material do nosso curso um arquivo chamado

"dias_final_de_semana.csv", que reúne os dias de finais de semana que tivemos a cada mês dos anos de 2017 e 2018. Normalizaremos cada mês nos baseando nos dias de finais de semana e veremos como será gerado outro gráfico diferente.

Para começar, carregaremos nosso novo DataFrame.

```
quantidade_de_dias_de_fds = pd.read_csv('dias_final_de_semana.csv')
```

Vamos rodar e visualizar a quantidade de dias com `quantidade_de_dias_de_fds.head()`.

Teremos a coluna `quantidade_de_dias`. Então, pegaremos a `quantidade_de_dias_de_fds` e dentro dessa coluna, estabeleceremos que sejam trazidos apenas os valores dentro de uma array.

```
quantidade_de_dias_de_fds['quantidade_de_dias'].values
```

Nos será devolvido a quantidade de dias de finais de semana que tivemos no mês 1, no 2 e assim respectivamente. Agora que teremos esses valores, dividiremos o número das vendas pela quantidade de finais de semana que tivemos. Em seguida, já visualizaremos os primeiros registros.

```
cafelura['vendas_normalizadas'] = cafelura['vendas']/quantidade_de_dias_de_fds['quantidade_de_dias']  
cafelura.head()
```

COPIAR CÓDIGO



Executaremos e teremos, então, o mês, o valor das vendas e o valor das vendas normalizadas.



64%

ATIVIDADES
2 de 6

DISCORD
ALURA

FÓRUM DO
CURSO

VOLTAR
PARA
DASHBOARD

MODOS
NOTURNOS



15.4k xp

a

Faremos um plot das vendas normalizadas, portanto. Chamaremos a função plotar e completaremos os campos. `plotar(titulo, labelx, labely, x, y, dataset)`

```
plotar('Vendas normalizadas da Cafelura de 2017 a 2018', 'Tempo', 'Vendas normalizadas',  
'mes', 'vendas_normalizadas', cafelura)
```

Rodaremos e por termos normalizado as vendas dividindo pela quantidade de finais de semana que temos, nosso gráfico será totalmente diferente do anterior, mostrando apenas uma curva de crescimento. Vamos criar um plot para comparar ambos os gráficos.

Nesse caso não teremos 3, apenas 2 no subplot. Queremos que todo o espaço seja ocupado e que o primeiro plot seja o primeiro gráfico.

```
plt.figure(figsize=(12,8))  
ax=plt.subplot(2,1,1)  
ax.set_title('Vendas Cafelura 2017 e 2018', fontsize=18)  
sns.lineplot(x='mes', y='vendas', data=cafelura)  
ax=plt.subplot(2,1,2)  
ax.set_title('Vendas normalizadas Cafelura 2017 e 2018', fontsize=18)  
sns.lineplot(x='mes', y='vendas_normalizadas', data=cafelura)  
ax=ax
```

COPIAR CÓDIGO

Então teremos as duas formas das vendas apresentadas nos gráficos, tanto as vendas quanto as vendas normalizadas de acordo com a quantidade de dias de finais de semana.



64%

ATIVIDADES
2 de 6

DISCORD
ALURA

FÓRUM DO
CURSO

VOLTAR
PARA
DASHBOARD

**MODO
NOTURNO**



15.4k xp

a