Aprendendo Analise de dados/Mexer no jupyter notebook

**Import** pandas **as** pd: para importar alguma biblioteca

Biblioteca boa para ver gráficos com python **plotly**

Pd.DataFrame(): serve para criar uma tabela dentro do pandas

Para facilitar declara uma variável e poe o pd.dataframe() que já vai criar a tabela

pd**.read\_excel ():** para ler uma arquivo

df.to\_xxx( serve para salvar um arquivo desse modo

display(xxx): é como se fosse um print só que estilizada

xxx**.describe()**  serve para mostrar as informações resumidas

xxx**.sum()** : serve para somar a tabela inteira

xxx**.astype(**xxx):Serve para mudar o tipo da variável, ex: mudar str para float.

xxxx**.groupby(‘**xxx**’)**: e para agrupar uma coluna ou alguma coisa

quando se adiciona vários [[]] e aonde vai pegar varias colunas

xxx**.head():**Seve para mostrar as 5 primeiras linhas, ou ate onde você quer q mostre

xxx**.tail()**: Serve para mostrar as 5 ultimas linhas do DataFrame

xxx**.sample(**x**)**: Pega o número que você escolher de linhas aleatórias do seu data frame

xxx**.shape**:serve para mostrar quantas linhas e quantas colunas

xxx**.plot.pie()**:Serve para plotar gráfico de pizza

to\_numeric(xxx[“xxx”], erros = “coerce”(Serve para se houver error deixa ela vazia))

xxx**.qcut**:Permite criar categorias a partir de valores contínuos (ou de muitas categorias). Basta você partir de um princípio de ordenamento e selecionar a quantidade de categorias final que deseja

**.T**: Serve para inverter as linhas pelas colunas

xxx.**describe():**Ele vai mostrar um resumo das colunas numéricas sobre a tabela

para pegar uma coluno em expecifico, deve-se colocar entre [ ] e o nome identido da coluna

.loc [linhas,colunas ]: - Pegar 1 linha

- Pegar linhas de acordo com alguma condição

- Pegar linhas e colunas específicas

- Pegar 1 valor específico

Dois pontos “:” significa todas as linhas ou colunas

xxx**.append(**xxx**)**: ele adiciona outra variável

xxx**.drop(**xxx ou **colums=** x**)**: serve para excluir uma coluna ou linha

xxx**.dropna(how=’all’)**:exclui as linhas com valores vazios que são completamente vazios e se quiser colunas so colocar **axis = 1**

**xxx.dropna(how”any”,axis = 0)** Exclui linhas que tem algum valor vazio vazias

xxx**.dropna():** ele exclui aonde existe qualquer linha em branco,não precisa ser a linha toda, mas qq espaço

xxx**.isnull()**:Serve para buscar os valores faltantes ou Nulos no dataframe

xxx**.dropna(inplace=True)**:Serve para pegar os espaços em branco e excluir da variavel

xxx[xxx]**.fillna(**x**)**: serve para preencher as colunas vazias com alguma coisa

xxx**.mean()**: serve para calcular a media de uma linha ou coluna

xxx**.fill():**serve para preencher um valor que esta logo a cima de cada coluna correspondente

xxx**info:**ele traz informações sobre a tabela

xxx**.value\_counts():** serve para contar todos os valores de uma coluna as transações da coluna

(normalize = True): serve para mostrar os valores em percentual

xxx**.count()**:Serve para contar as linhas de uma coluna que não estão em branco

xxx**.size**: Serve para contar todas as linhas de uma coluna,incluindo as linhas em branco

normalize=True: ele vai normalizar os valores, em vez de contar a quantidade ele vai mostrar o que aquilo representa sobre o total

xxx**.merge(**xxx): ela mescla uma tabela com a outra, junta as tabelas

xxx**.plot.bar():** serve para plotar qualquer gráfico

xxx**.min():**Serve para buscar o menor valor da variável

xxx**.max():**Serve para buscar o maior valor da variável

xxx**.std():**Serve para buscar o desvio padrão dos valores

xxx.**.hist()**:Serve para abrir um historiograma

xxx**, sheet\_name=’**nome da aba **‘:**serve para selecionar uma aba expecifica do excel

xxx**, sheet\_name=’**xxx’, header=1:Serve para tirar o cabeçalho da primeira linha

xxx,**skipfooter=**xx:Serve para skipar uma certa quantidade de linhas

xxx**.split(‘**x**’)**:Serve para dividir o conteúdo de uma string

xxx**.nunique()**: Serve para mostrar valores únicos que existem em cada coluna

Pyautogui: biblioteca para comandar seu teclado ou mouse

.hotkey(“escreve as teclas que você quer apertar usando virgula”) : serve para usar os comandos

.press( aperta uma tecla)

.write( escreve um texto)

.PAUSE( serve para da um tempo no codigo)

.click( serve para clicar com o mouse, clicks = 2( serve para dar dois clicks) )

Dica e usar o sleep para dar tempo de colocar o mouse no local

.position() mostra a posição do seu mouse para usar no click

Pyperclip.copy( serve para copiar algo

~:Serve para pegarmos o valor negativo de uma sequência de True ou False

xxx**.sort\_values()**:Serve para ordernar um dataframe por COLUNAS

-**by=**nome da coluna

-ascending=False: Serve para pegarmos os itens em ordem decrescente

&:Significa E, e | significa Ou

atalhos

Esc' - Saí do modo edição da célula atual

1. 'Enter' - Edita a célula selecionada

2. 'Shift + Enter' - Executa a célula selecionada

3. 'A' - Adiciona uma nova célula acima da célula atual (não pode estar em modo de edição)

4. 'B' - Adiciona uma nova célula abaixo da célula atual (não pode estar em modo de edição)

5. 'X' - Deleta a célula selecionada (não pode estar em modo de edição)

6. 'Z' - Desfaz a deleção de uma célula (não pode estar em modo de edição)

7. 'Ctrl + Z' - Desfaz as últimas edições

8. 'Ctrl + Shift + Z' - Refaz o último comando de desfazer

9. 'H' - Abre um arquivo de ajuda com todos os atalhos (não pode estar em modo de edição)

10. 'Tab' - Faz sugestões para completar o código

11. '?função' - Mostra a documentação de uma dada função

%matplotlib inline: comando para plotar(desenhar) os gráficos em todas as versões do Jupyter Notebook  
sns.kdeplot(**XXX**):Serve para plotar a ferramenta de gráficos

Plotly.express as px gráfico

Px.histogram cria o gráfico padrão

Px.histogram( xxx, x=(coluna x), color = xxx(comparar com outra cor)

Gráfico.show() mostra o grafico