

## Intensivão de Python 09 a 12/05/2022

### Aula 4 - Machine Learning e Data Science

Inteligência artificial, machine learning. Análise de dados, plotagem de gráficos.

Fazer download do material, gabarito das 4 aulas, fazer os exercícios para praticar os conceitos mostrados.

Na aula de hoje será um projeto de ciência de dados, quando vem de forma desafio, prever vendas em um determinado mês quando investido um determinado valor de marketing. Ciência de dados para fazer previsão de retorno.

Primeiro passo em todo desafio, construir o passo a passo em português das coisas que vc vai fazer pra resolver o desafio.

Ciência de dados é o projeto como o todo e a inteligência artificial é uma etapa do projeto de ciência de dados.

Passo 1: Entendimento do desafio;

Passo 2: Entendimento da área da empresa;

Passo 3: Extração/Obtenção de dados;

Passo 4: Ajuste de dados (tratamento /limpeza);

Passo 5: Análise exploratória;

Passo 6: Modelagem + Algoritmos (aqui que entra a inteligência artificial, se necessário);

Passo 7: Interpretação de resultados;

#importar a base de dados para o Python

Código do Python:

```
import pandas as pd
```

```
tabela = pd.read_csv("advertising.csv")
```

display(tabela)      formatação deve se preocupar apenas no final. No exemplo, os valores de investimento estão em base diferente da coluna vendas que está em milhões.

#visualizar a base de dados e fazer os ajustes necessários na base de dados

#análise exploratória entender como sua base de dados está se comportando

`tabela.corr()` com este comando calcula-se a correlação entre os dados contidos na tabela. A correlação varia de 0 a 1, se o valor está próximo de 1, então os dois valores comparados estão muito ligados, aumentando linearmente. Se o valor próximo de 0, os valores estão bem discrepantes

o objetivo é entender como o valor investido em TV está relacionado com o valor de vendas, como valor investido em radio está relacionado com o valor de vendas e como o valor investido em jornal está impactando no valor das vendas

para criar gráficos em Python, temos matplotlib e Seaborn como bibliotecas e também scikit-learn, sendo este ultimo para criar a inteligencia artificial

no caso, criaremos um gráfico como mapa de calor para visualizar a correlação dos valores investidos correlacionados com os valores de vendas

```
import seaborn as sns
```

```
import matplotlib.pyplot as plt
```

```
sns.heatmap(tabela.corr(), annot=True, cmap="Wistia")
```

 este comando cria o gráfico

```
plt.show()
```

 este comando exibe o gráfico, e sempre exibirá o último gráfico criado

#criar a inteligência artificial e fazer as previsões

como o youtube sabe recomendar vídeos pra vc, toda inteligencia artificial precisa passar pelo aprendizado, precisa treinar ela, utilizando a base de dados

precisamos treinar a inteligencia artificial e, é necessário testar pra saber se ela treinou, se ela aprendeu e conseguiu fazer boas previsões

Dar um pedaço grande de treino e deixar poucos dados para teste. Não devemos passar a base dados toda para o aprendizado de máquina, passamos apenas um pedaço e o outro pedaço, ficará para teste. Entender se a inteligencia artificial está funcionando corretamente

No caso de variáveis que vc quer usar para prever, por convenção chamamos de Y. No caso do projeto, previsão de vendas, a variável vendas será chamada de Y e os valores investidos serão chamados de X. Os valores usados para fazer a previsão chamamos de X

Deixar de 10-30% para teste e 70-90% para treino, uma recomendação, porém deve variar de acordo com cada necessidade

Como devemos passar parâmetros para a inteligencia artificial, devemos considerar os valores para a previsão, chamado de X

```
y = tabela["Vendas"]
```

```
x = tabela[["TV", "Radio", "Jornal"]]
```

```
from sklearn.model_selection import train_test_split
```

```
x_treino, x_teste, y_treino, y_teste = train_test_split(x, y, test_size=0.3)
```

```
from sklearn.linear_model import LinearRegression
from sklearn.ensemble import RandomForestRegressor
```

```
modelo_regressaolinear = LinearRegression()
modelo_arvorededecisao = RandomForestRegressor()
```

```
modelo_regressaolinear.fit(x_treino, y_treino)
modelo_arvorededecisao.fit(x_treino, y_treino)
```

Dois modelos foram utilizados de inteligencia artificial, foi utilizado regressão linear e o outro foi arvore de decisão

- 1 - Regressão Linear;
- 2 - Árvore de Decisão;

Existem centenas de opções de equações matemáticas para aplicar na inteligencia artificial

Para saber se a inteligencia artificial está respondendo bem, faz um calculo do r2 o qual compara com os valores de teste e a previsão da inteligencia artificial

No caso, fizemos comparação das duas AI, digitando os comandos abaixo:

```
from sklearn.metrics import r2_score
print(r2_score(y_teste, previsao_regressaolinear))
print(r2_score(y_teste, previsao_arvorededecisao))
resultado da regressão linear foi de 0.90
resultado da arvore de decisão foi de 0.96
```

Efetuada download dos arquivos das 4 aulas