COMPONENTE CURRICULAR:	Projeto aplicado II
NOME COMPLETO DO ALUNO:	Mariana Simões Rubio; Patrícia Corrêa França; Valdiney Atílio Pedro.
RA:	10424388; 10423533; 10424616.

APLICANDO CONHECIMENTO A2 – Definição do produto analítico

Link Github: https://github.com/valdineyatilio/ProjetoAplicado-II

1. Definir quais bibliotecas (pacotes) na linguagem devem ser usadas:

   A linguagem utilizada foi python com as seguintes bibliotecas:

   - Pandas: Manipulação e análise de dados (DataFrames).
   - Matplotlib: Visualização de dados (gráficos).
   - Seaborn: Visualização de dados com gráficos mais avançados e estilizados.
   - Scikit-learn: Ferramentas para pré-processamento de dados, modelos de machine learning e avaliação de performance.

2. Análise exploratória da base de dados escolhida:

O objetivo dessa análise é de entender a estrutura e as características da base de dados.

O Algoritmo da análise exploratória da base de dados se encontra no arquivo AnaliseExploratoria.py e AnaliseExploratoria.ipynb como alternativas de visualização e execução dos códigos referente ao que foi requisitado a apresentar.

Os passos utilizados foram:

- Carregamento dos dados;
- Visualização das primeiras linhas do data frame;
- Verificar o nome das colunas;
- Verificação de tipos de dados;

- Verificação das informações das colunas;
- Resumo estatísticos das variáveis numéricas;
- Visualizações gráficas;
- Histograma das visualizações;
- Gráfico de dispersão das visualizações;
- Gráficos de barras por número de vídeos por categoria;
- Identificação dos valores ausentes;
- Identificação de outliers usando z-score;
- Boxplot para identificar outliers;
- Removendo outliers;
- Análise de correlação;
- Mapa de calor da correlação.

3. Tratamento da base de dados (preparação e treinamento):

O Algoritmo do tratamento da base de dados se encontra no arquivo TratamentoDeDados.py e TratamentoDeDados.ipynb como alternativas de visualização e execução dos códigos referente ao que foi requisitado a apresentar.

Os passos utilizados foram:

- Carregar os dados;
- Limpeza dos dados;
- Transformação dos dados;
- Dividindo os dados;
- Exibindo as primeiras linhas dos conjuntos de treinamento e teste.

4. Definir e descrever as bases teóricas dos métodos:

a. Redes Neurais Artificiais (RNA):

Teoria: As RNAs são inspiradas no funcionamento do cérebro humano e são compostas por camadas de neurônios artificiais. Elas são capazes de aprender padrões complexos a partir dos dados.

Aplicação: utilizando uma rede neural para prever os ganhos anuais mais altos com base em variáveis como assinantes, visualizações de vídeos e uploads.

b. Regressor MLP (Perceptron Multicamadas):

Teoria: O MLP é um tipo de rede neural feedforward que consiste em uma camada de entrada, uma ou mais camadas ocultas e uma camada de saída. Ele utiliza a retropropagação para ajustar os pesos durante o treinamento.

Aplicação: Configuração do MLPRegressor com duas camadas ocultas (100 e 50 neurônios) e utilização do algoritmo de gradiente descendente estocástico (SGD) para a otimização.

5. Definição e descrição de como será calculada a acurácia

a. Erro Quadrático Médio (MSE):

Teoria: O MSE é uma métrica que mede a média dos quadrados dos erros, ou seja, a diferença entre os valores previstos e os valores reais. É uma medida comum de acurácia para modelos de regressão.

Aplicação: Cálculo do MSE para avaliar a performance do modelo de rede neural. Um MSE menor indica um modelo mais preciso.

b. Definição e descrição das bases teóricas dos métodos

Análise de Variância (ANOVA):

Teoria: A ANOVA é uma técnica estatística usada para comparar as médias de três ou mais grupos para determinar se pelo menos um grupo é significativamente diferente dos outros. Ela se baseia na decomposição da variabilidade total em variabilidade entre grupos e variabilidade dentro dos grupos.

Aplicação: Utilizando a ANOVA para comparar as visualizações de vídeos entre diferentes categorias do YouTube. Isso ajuda a identificar se há diferenças significativas nas visualizações médias entre as categorias.

c. Estatística F e Valor p:

Teoria: A estatística F é a razão entre a variabilidade entre os grupos e a variabilidade dentro dos grupos. Um valor F alto indica que há uma diferença significativa entre as médias dos grupos. O valor p é a probabilidade de obter um valor F tão extremo quanto o observado, assumindo que a hipótese nula (de que todas as médias são iguais) é verdadeira.

Aplicação: Calcular a estatística F e o valor p para determinar se há diferenças significativas nas visualizações de vídeos entre as categorias. Um valor p menor que um nível de significância (geralmente 0.05) indica que há uma diferença significativa.

d. Definição e descrição das bases teóricas dos métodos
   Algoritmo Apriori:

Teoria: O algoritmo Apriori é um método clássico de mineração de dados usado para extrair padrões frequentes e regras de associação em grandes bases de dados. Ele funciona iterativamente, identificando conjuntos de itens frequentes e gerando regras de associação a partir desses conjuntos.

Aplicação: Utilizar o algoritmo Apriori para encontrar associações entre países e altos ganhos anuais no YouTube. Isso pode ajudar a identificar padrões interessantes sobre quais países têm maior probabilidade de gerar altos ganhos.

e. Regras de Associação:

Teoria: As regras de associação são usadas para descobrir relações interessantes entre variáveis em grandes bases de dados. Elas são expressas na forma “se A, então B”, onde A e B são conjuntos de itens. As métricas comuns para avaliar a força das regras incluem suporte, confiança e lift.

Aplicação: Utilizar a função association_rules para gerar regras de associação a partir dos conjuntos de itens frequentes identificados pelo algoritmo Apriori. As regras geradas podem revelar insights sobre a relação entre o país de origem e a probabilidade de altos ganhos.

f. Métricas de Avaliação (Lift, Suporte, Confiança):

Teoria:

Suporte: A proporção de registros no conjunto de dados que contém ambos os itens A e B.

Confiança: A proporção de registros que contém o item B entre aqueles que contêm o item A.

Lift: A razão entre a confiança da regra e a expectativa de encontrar o item B, dado que o item A está presente. Um lift maior que 1 indica uma associação positiva entre A e B.

Aplicação: utilizando essas métricas para avaliar a força das regras de associação geradas. Regras com alto suporte, confiança e lift são consideradas mais significativas.

acuracia = accuracy_score(y_test, y_pred)

Métricas para Avaliar a Acurácia

Métricas importantes:

Precisão (Precision): precisao = precision_score(y_test, y_pred, average='weighted', zero_division=1)

Recall (Sensibilidade): recall = recall_score(y_test, y_pred, average='weighted', zero_division=1)

F1-Score: f1 = f1_score(y_test, y_pred, average='weighted', zero_division=1)

AUC-ROC: O código verifica se há mais de uma classe em y_test antes de calcular o AUC-ROC:

```
if len(set(y_test)) > 1:
    y_test_binarized = label_binarize(y_test, classes=classes)
    y_pred_proba = modelo.predict_proba(X_test)
    roc_auc = roc_auc_score(y_test_binarized, y_pred_proba, multi_class='ovr')
```

Processo de Validação

O código utiliza a técnica de divisão treino-teste para validar o modelo:

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

6. Conclusão

A partir do código apresentado, a acurácia do modelo é calculada juntamente com métricas adicionais, como precisão, recall, F1-score e AUC-ROC, proporcionando uma análise mais abrangente do desempenho do modelo. Além disso, a aplicação da técnica de validação de divisão treino-teste garante a robustez dos resultados, permitindo avaliar a generalização do modelo em dados desconhecidos. Essa abordagem combinada fortalece a confiabilidade das conclusões obtidas, oferecendo uma visão completa tanto dos acertos quanto dos erros do modelo, contribuindo para decisões informadas sobre o seu desempenho.