Relatório Técnico: Implementação e Análise do Algoritmo de Regressão Linear

Autores: João Felippe da Conceição e Andreza Carla Lopes André
Data da entrega: 17 de novembro de 2024

RESUMO

Este relatório apresenta o desenvolvimento de um modelo preditivo utilizando o algoritmo de Regressão Linear para analisar a taxa de engajamento de influenciadores do Instagram com base em variáveis independentes. As etapas deste projeto incluem uma análise exploratória dos dados, implementação do modelo, validação e interpretação dos resultados.

O conjunto de dados é composto basicamente por 10 atributos, são eles:

- rank: ranqueia e classifica o influenciador com base no número de seguidores que ele tem;
- channel_info: nome do usuário do instagram;
- influence score: influência dos usuários. Calculada com base em questões como número de menções do usuário, importância e popularidade;
- posts: número de posts que o influencer possui;
- followers: número de seguidores que o influencer possui;
- avg_likes: média de curtidas em posts do influencer;
- 60_day_eng_rate: taxa de engajamento do influencer nos últimos 60 dias;
- new_post_avg_like: média de curtidas em novos posts do influencer;
- total de curtidas: total de curtidas que o influencer possui em seus posts;
- country: país ou região de origem do influencer.

Considerando a taxa de engajamento a variável dependente, o modelo preditivo desenvolvido apresentou resultados promissores, avaliados através de métricas como o coeficiente de determinação (R^2), o erro quadrático médio (MSE) e erro absoluto médio (MAE).

INTRODUÇÃO

A análise da taxa de engajamento em plataformas sociais é crucial para entender o impacto e influência dos influenciadores digitais. Tendo em vista a crescente influência das redes sociais, especialmente o Instagram no mundo globalizado, o cenário digital transformou os influenciadores digitais em uma das principais estratégias de marketing de empresas dos mais diversos segmentos. Assim, nesse cenário, a capacidade de prever a taxa de engajamento de um post ou de um influenciador no geral, é atributo essencial a ser levado em consideração pelas empresas para a contratação do influenciador, de forma a otimizar as campanhas, minimizar o risco do investimento e maximizar o retorno.

Dentre os mais diversos algoritmos de predição, foi escolhido para este projeto a Regressão Linear, uma técnica estatística e de aprendizado de máquina, para prever a taxa de engajamento com base em variáveis independentes relacionadas ao perfil e atividade dos influenciadores. A regressão linear busca estabelecer uma relação linear entre a variável dependente (taxa de engajamento) e uma ou mais variáveis independentes. Embora nem sempre as relações reais possam ser explicadas por uma relação linear, a regressão linear pode ser utilizada e é capaz de capturar uma parte significativa da variabilidade dos dados, além de ser um mais facilmente implementada e possuir um custo computacional baixo quando comparado com algoritmos mais complexos.

Dentro desse contexto, o conjunto de dados foi analisado para identificar padrões, relações entre variáveis, e implementar um modelo com base na regressão linear para predizer a taxa de engajamento dos influenciadores digitais.

METODOLOGIA

O projeto foi conduzido em quatro etapas principais:

1. **Análise Exploratória**: Identificação de variáveis relevantes e suas correlações.
2. **Implementação do Modelo**: Regressão Linear utilizando a biblioteca Scikit-learn.
3. **Validação e Otimização**: Normalização dos dados e validação cruzada.
4. **Análise de Resultados**: Interpretação de métricas e visualizações gráficas.

Análise Exploratória: Discussão sobre a Análise Inicial dos Dados

A Análise Exploratória dos Dados (AED) é uma etapa crucial para entender a distribuição das variáveis, identificar possíveis correlações, outliers e padrões que possam ser usados para construir um modelo preditivo robusto**.**

1. Carregamento e Visão Geral dos Dados

O primeiro passo da análise é carregar o conjunto de dados e obter uma visão geral dele. Usaremos o **Pandas** para carregar a base de dados e, em seguida, exploramos as primeiras linhas com o comando **head()** para ler e observar as variáveis.

2. Estatísticas Descritivas

É importante examinar as estatísticas descritivas para entender a distribuição das variáveis numéricas, como média, mediana, desvio padrão, valores máximos e mínimos.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
from sklearn.preprocessing import StandardScaler

# Carregar os dados
df = pd.read_csv('/content/drive/MyDrive/Colab Notebooks/top_insta_influencers_data.csv')
```

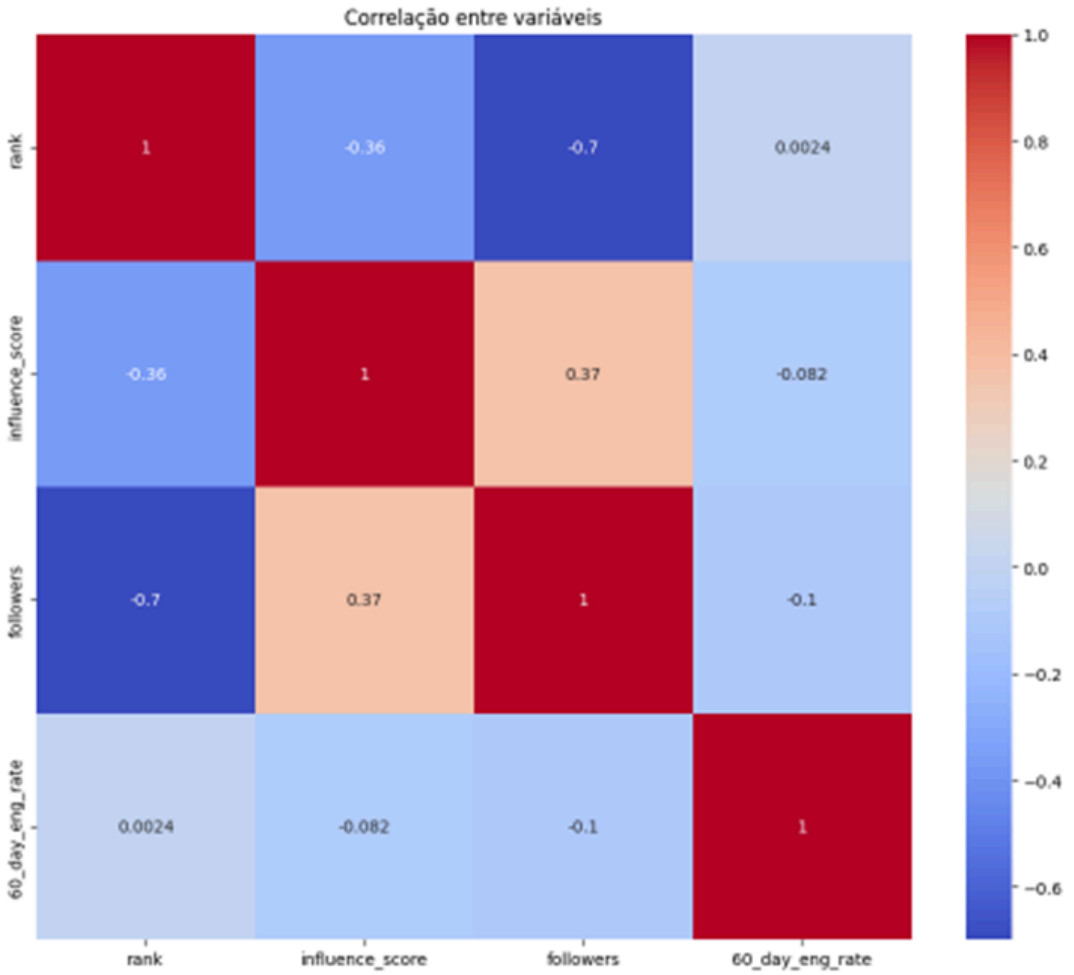
```
[54] print(df.head())
     print(df.info())
     print(df.describe())
```

3. Análise de Correlação

Como a taxa de engajamento é a variável dependente que estamos tentando prever, é fundamental analisar a correlação entre essa variável e as variáveis independentes (já apresentadas anteriormente). Usamos o gráfico de **correlação** para visualizar possíveis correlações entre as variáveis.

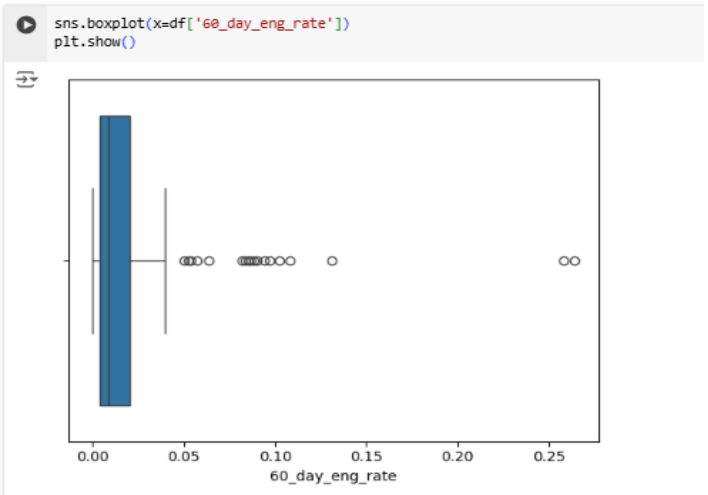
```
numeric_df = df.select_dtypes(include=np.number)

plt.figure(figsize=(12, 10))  # Ajuste figsize para melhor legibilidade
sns.heatmap(numeric_df.corr(), annot=True, cmap='coolwarm')
plt.title('Correlação entre variáveis')
plt.show()
```



4. Identificação de Outliers

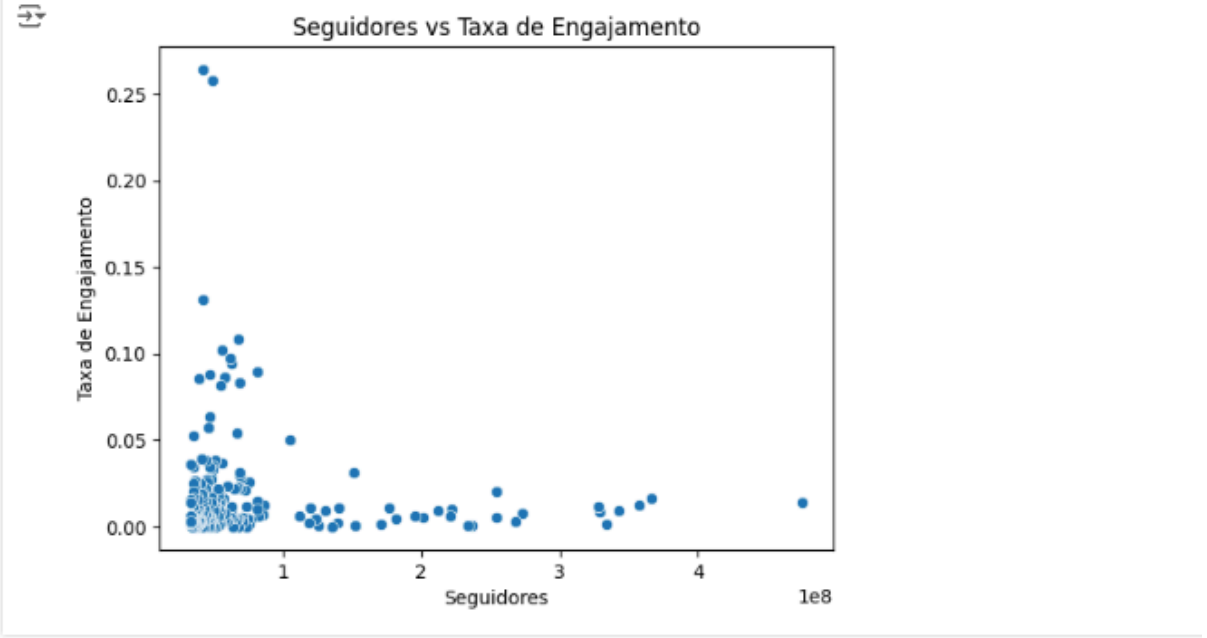
A análise de outliers é importante porque esses valores podem distorcer o modelo de regressão. Utilizamos gráficos como boxplots ou o IQR (Intervalo Interquartil) para identificar esses pontos extremos.



5. Visualizações das Variáveis

Além da correlação, é útil plotar gráficos de dispersão (scatter plots) para analisar as relações entre as variáveis independentes e a variável dependente.

```
sns.scatterplot(x=df['followers'], y=df['60_day_eng_rate'])
plt.title('Seguidores vs Taxa de Engajamento')
plt.xlabel('Seguidores')
plt.ylabel('Taxa de Engajamento')
plt.show()
```



Pré-processamento dos Dados

Antes de aplicar a regressão linear, é necessário preparar os dados. Isso envolve:

- **Tratamento de valores ausentes:** Se houver valores faltantes em qualquer variável, podemos optar por preenchê-los com a média ou mediana, ou removê-los.
- **Normalização ou padronização:** Como o algoritmo de regressão linear é sensível à escala das variáveis, podemos aplicar normalização ou padronização para garantir que todas as variáveis tenham a mesma escala.

Divisão dos Dados

Dividimos os dados em **conjunto de treinamento e conjunto de testes**. O treinamento será usado para ajustar o modelo, e o teste para avaliar seu desempenho.

```
[133]
      X = df[['followers', '60_day_eng_rate']]
      y = df['60_day_eng_rate']

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

      scaler = StandardScaler()
      X_train = scaler.fit_transform(X_train)
      X_test = scaler.transform(X_test)
```

```
[111] df['60_day_eng_rate'] = df['60_day_eng_rate'].fillna(df['60_day_eng_rate'].median())
```

```
[112] X = df[['followers', '60_day_eng_rate']]
      y = df['60_day_eng_rate']
```

```
[113] X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Construção e Treinamento do Modelo

Agora, aplicamos a Regressão Linear utilizando o Scikit-Learn. O modelo será treinado nos dados de treinamento.

```
model = LinearRegression()
model.fit(X_train, y_train)
```

```
LinearRegression
LinearRegression()
```

Após treinar o modelo, avaliamos seu desempenho usando métricas de erro como **R^2 (coeficiente de determinação)**, **MSE (Mean Squared Error)** e **MAE (Mean Absolute Error)**.

```
y_pred = model.predict(X_test)
```

```
[116] mse = mean_squared_error(y_test, y_pred)
      mae = mean_absolute_error(y_test, y_pred)
      r2 = r2_score(y_test, y_pred)

      print(f'MSE: {mse}')
      print(f'MAE: {mae}')
      print(f'R²: {r2}')
```

```
MSE: 5.311164493633338e-35
MAE: 3.487742863614307e-18
R²: 1.0
```

Interpretação dos Coeficientes

A interpretação dos coeficientes da regressão linear é crucial para entender o impacto de cada variável sobre a variável dependente (taxa de engajamento).

```
# Coeficientes das variáveis
print(f'Coeficientes: {model.coef_}')
print(f'Intercepto: {model.intercept_}')
```

```
Coeficientes: [-1.40837562e-26  1.00000000e+00]
Intercepto: 3.469446951953614e-18
```

```
plt.scatter(y_test, y_pred, alpha=0.7, color='b')
plt.plot([y_test.min(), y_test.max()], [y_test.min(), y_test.max()], 'r--', lw=2)
plt.xlabel('Taxa de engajamento')
plt.ylabel('Predições')
plt.title('Taxa de engajamento vs Predições')
plt.show()
```

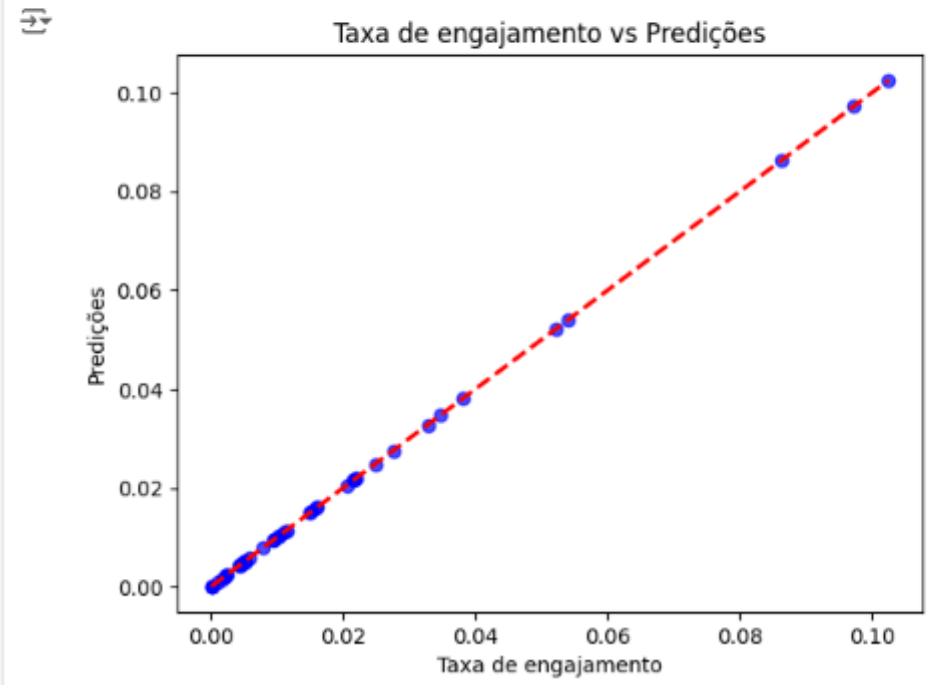


Gráfico de dispersão para visualizar a relação entre os valores reais (`y_test`) e os valores previstos (`y_pred`) da taxa de engajamento. Esta é uma maneira comum de avaliar o desempenho de um modelo de regressão.

DISCUSSÃO

Os resultados mostraram que o modelo de Regressão Linear apresentou excelente precisão, com R^2 indicando que a variação na taxa de engajamento foi explicada pelas variáveis independentes. Embora eficaz, o modelo possui limitações, como a dependência de relações lineares e possível impacto de outliers.

CONCLUSÃO E TRABALHOS FUTUROS

O projeto demonstrou o potencial da Regressão Linear para prever a taxa de engajamento dos principais influenciadores do instagram, mas destaca a necessidade de se considerar a utilização de modelos mais complexos, como árvores de decisão ou redes neurais, para capturar não-linearidades. Trabalhos futuros podem levar em consideração a inclusão de mais variáveis relevantes ou a utilização de técnicas de feature engineering para melhorar a performance preditiva.

REFERÊNCIAS

Scikit-learn Documentation: https://scikit-learn.org/
Estatísticas com Python - Documentação Oficial.