

Relatório Técnico: Implementação e Análise do Algoritmo de Regressão Linear

Resumo

O projeto visa implementar um modelo preditivo para estimar a taxa de engajamento de influenciadores em diversas regiões do mundo. A previsão será realizada com base em variáveis independentes que estão fortemente correlacionadas com a variável dependente, ou em casos onde as variáveis não apresentam correlação entre si. O objetivo principal é criar um sistema eficiente de previsão, usando técnicas de regressão e análise de dados.

Metodologia

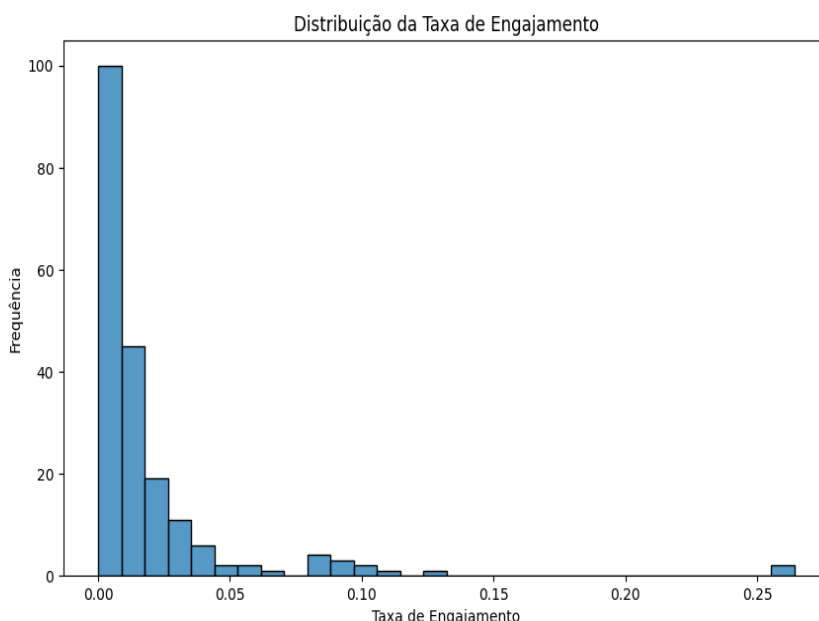
O projeto utilizou três variações de modelos de regressão linear para prever a taxa de engajamento dos influenciadores: regressão linear simples, Ridge e Lasso. O processo de modelagem incluiu a análise e o pré-processamento dos dados, onde foram realizadas etapas de remoção de outliers para garantir a qualidade dos dados e minimizar a influência de valores extremos no modelo. Além disso, as variáveis independentes foram normalizadas para garantir que todas as features tivessem uma escala similar, o que favorece o desempenho dos modelos de regressão. A seleção dos modelos foi feita com base na eficiência de cada um para lidar com dados altamente correlacionados e na capacidade de regularização de Ridge e Lasso para evitar overfitting.

Análise Exploratória

Conhecendo os Dados

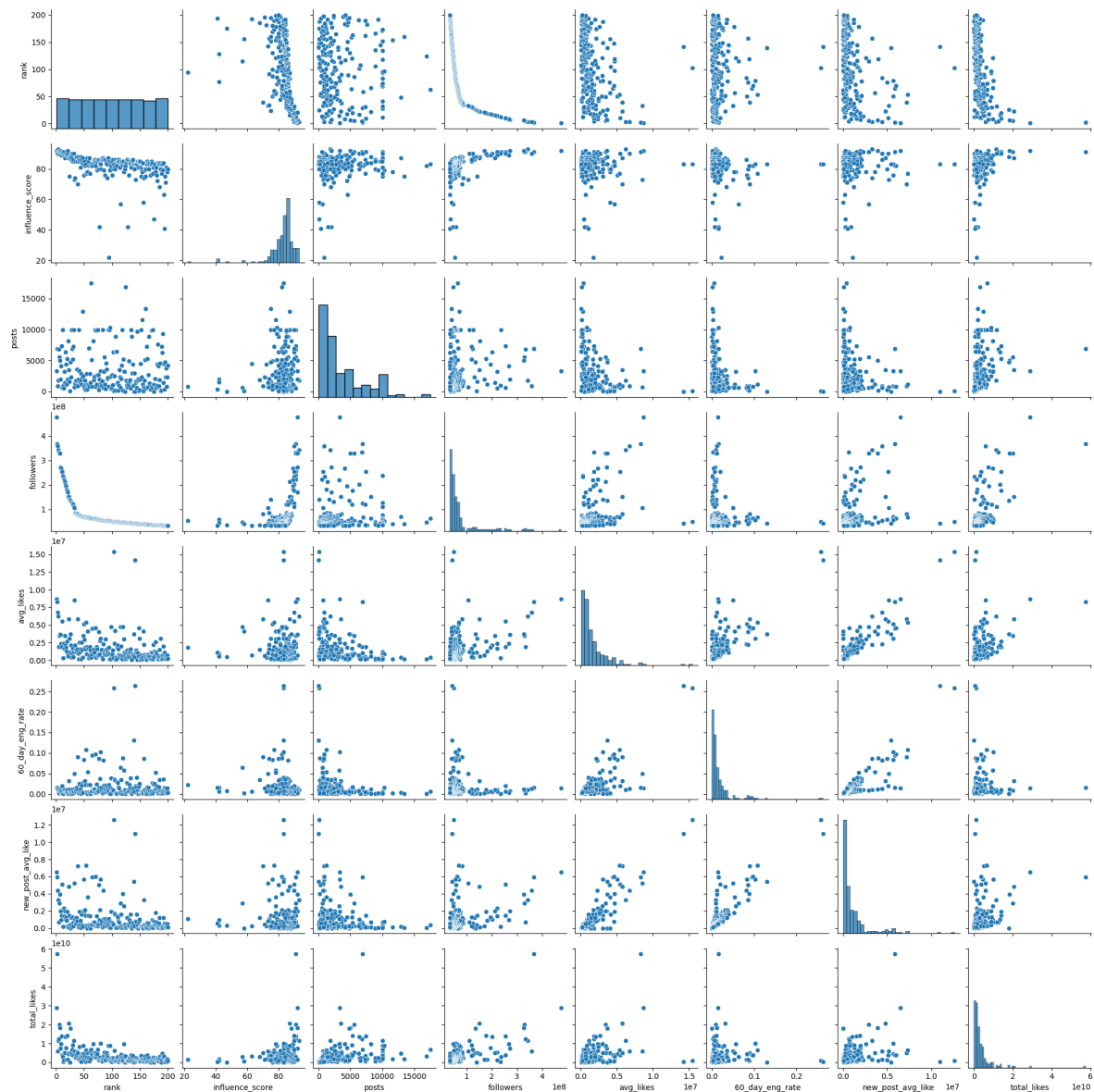
1° Analisamos a distribuição da nossa variável dependente

Percebe-mos alguns outliers com taxa de engajamento acima de 25%, no qual é muito alto comparado ao resto dos dados. Devemos tratar isso.



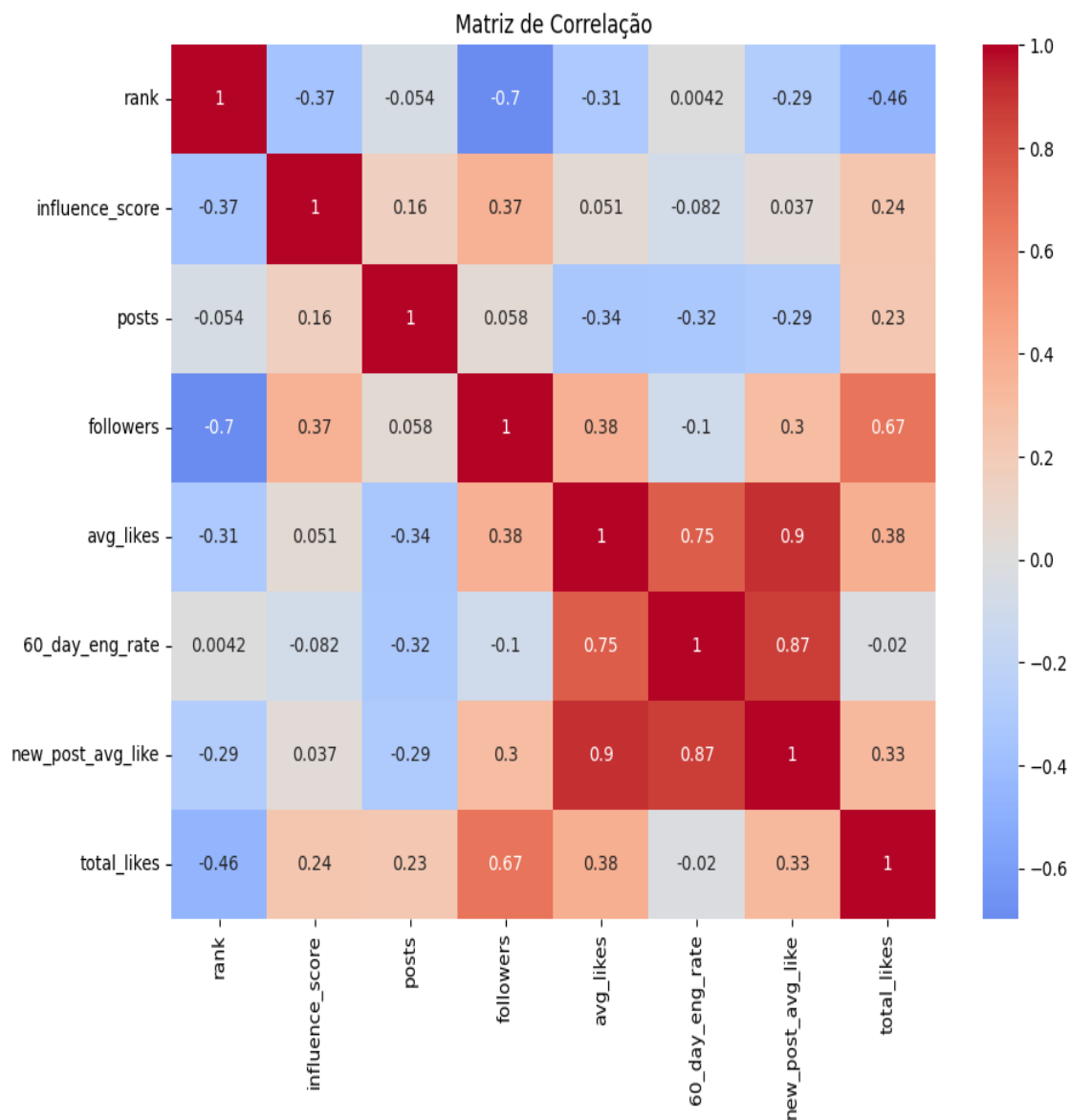
2º Analisamos a distribuição e relação de cada uma das variáveis

Percebe-se que a variável independente rank segue uma distribuição quase constante, e tem uma relação não linear com a variável dependente. A variável independente new_post_avg_like, segue uma distribuição bastante semelhante ao da nossa variável dependente, e de uma forma não tão clara, parece se relacionar com a variável dependente de forma linear. E também, a variável new_post_avg_like parece se relacionar de forma quase linear com avg_likes. Outra observação, é praticamente todas as variáveis exceto rank, possuem outliers que devem ser tratados para melhorar o desempenho do nosso modelo.



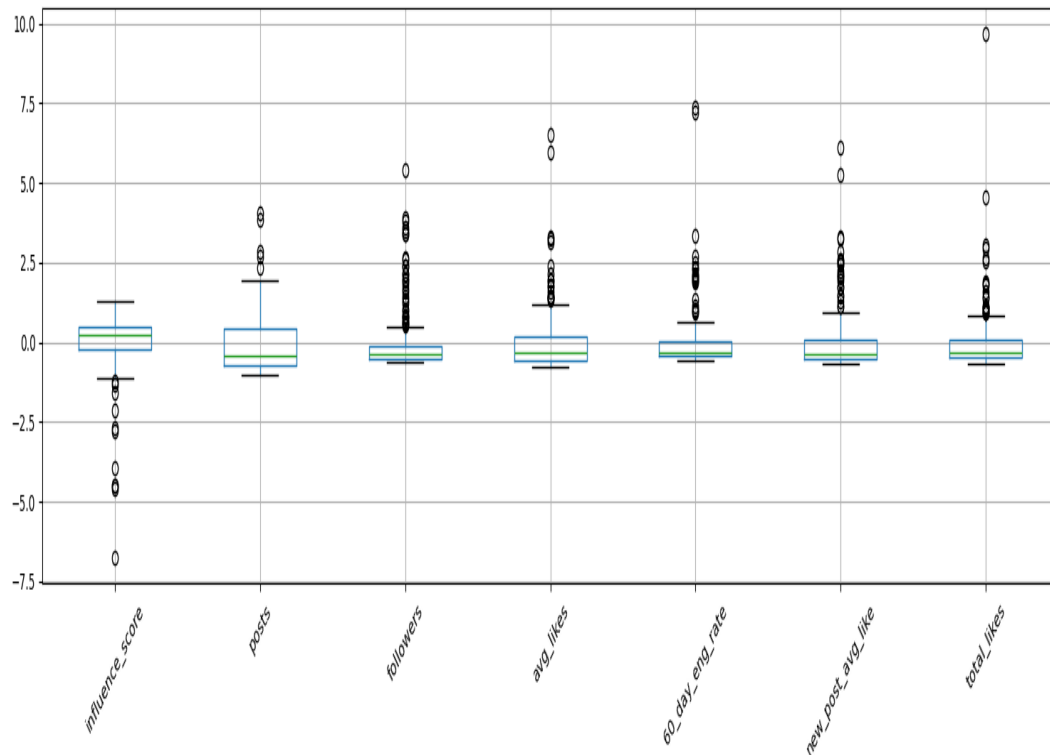
3º Analisamos a matriz de correlação

Como esperado a variável rank possui correlação quase nula com a variável dependente o que nos permite remover tranquilamente essa variável, da mesma forma as variáveis independentes channel_info e country foram removidas pelo mesmo motivo em testes anteriores. E percebemos uma forte correlação entre a variável dependente 60_day_eng_rate, e duas variáveis independentes new_post_avg_like e avg_likes. A correlação entre as variáveis independentes new_post_avg_like e avg_likes também está alta, devemos reduzi-la no pré-processamento. A variável followers também está um pouco correlacionada com total_likes, mas nada muito alarmante.



4° Distribuição das Features antes tratamento de Outliers

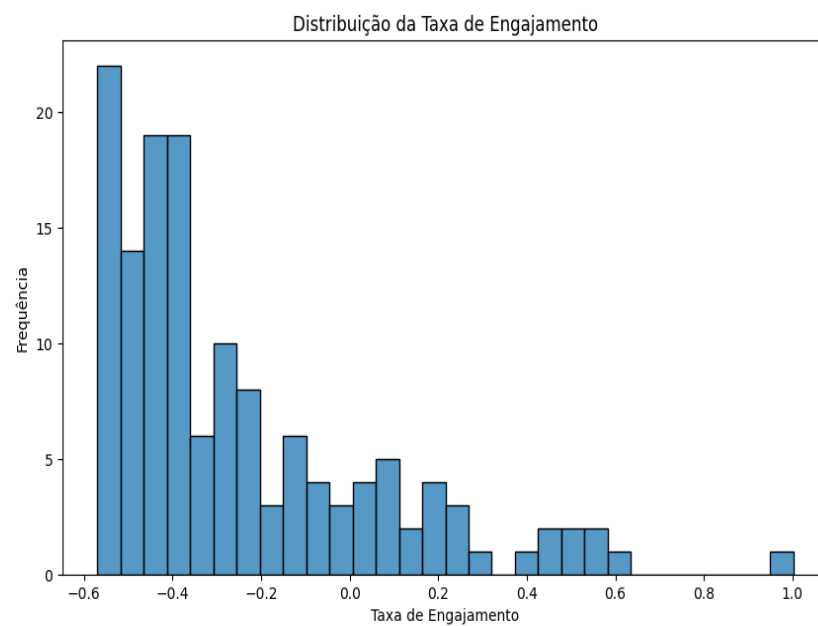
Percebe-se uma vasta quantidade de outliers em métricas como followers, avg_likes, new_post_avg_like e na variável dependente 60_day_eng_rate. Esses outliers devem ser tratados da melhor forma possível para que o modelo seja menos enviesado pelo overfitting.



Dados Após Pré-processamento

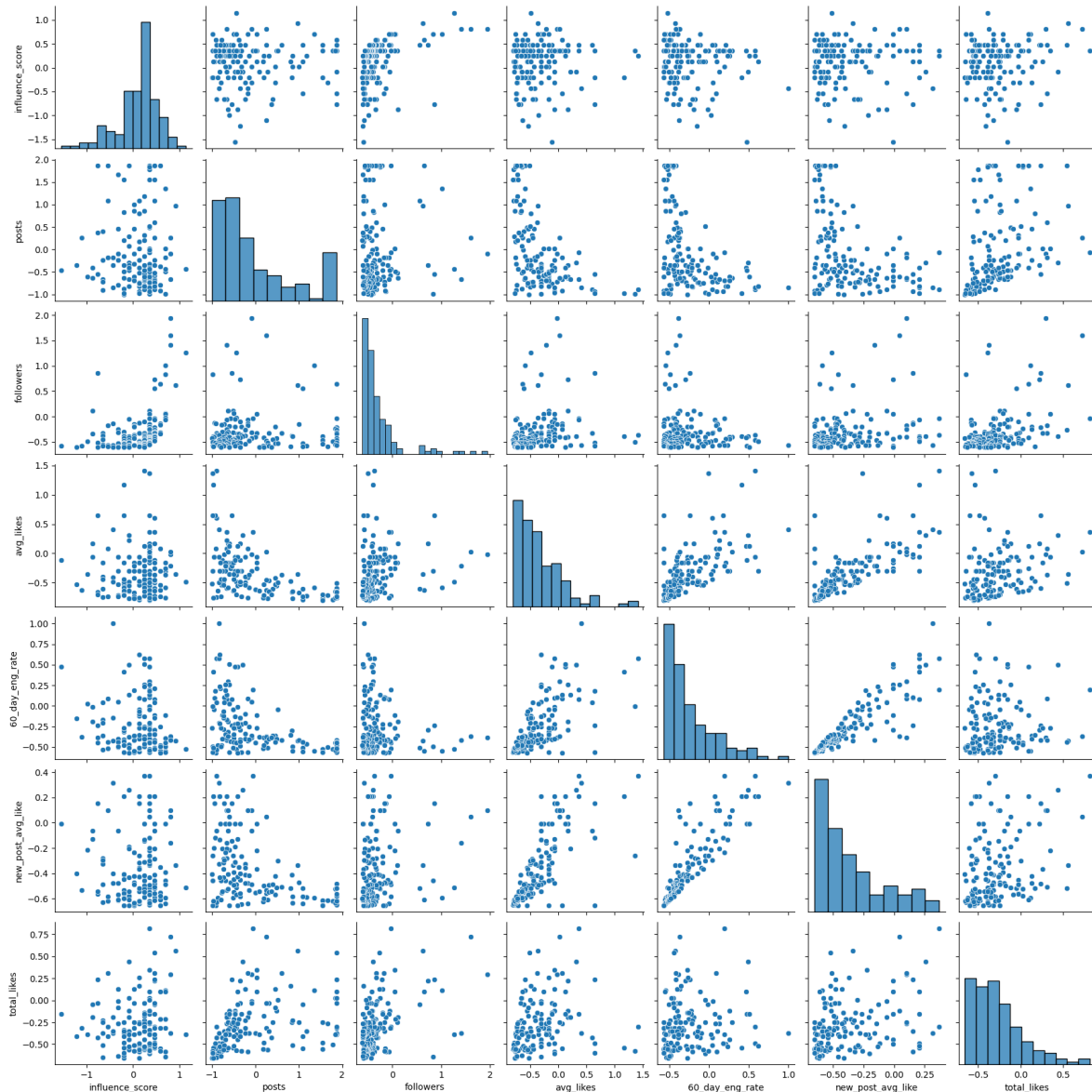
1° Nova distribuição da variável dependente

Após a remoção de outliers e a normalização dos dados a nova distribuição ficou mais clara. O pico inicial de dados foi reduzido, e a distribuição ficou mais suave nos evidenciando que temos muito mais dados com taxa de engajamento de porcentagem baixa do que de porcentagem alta.



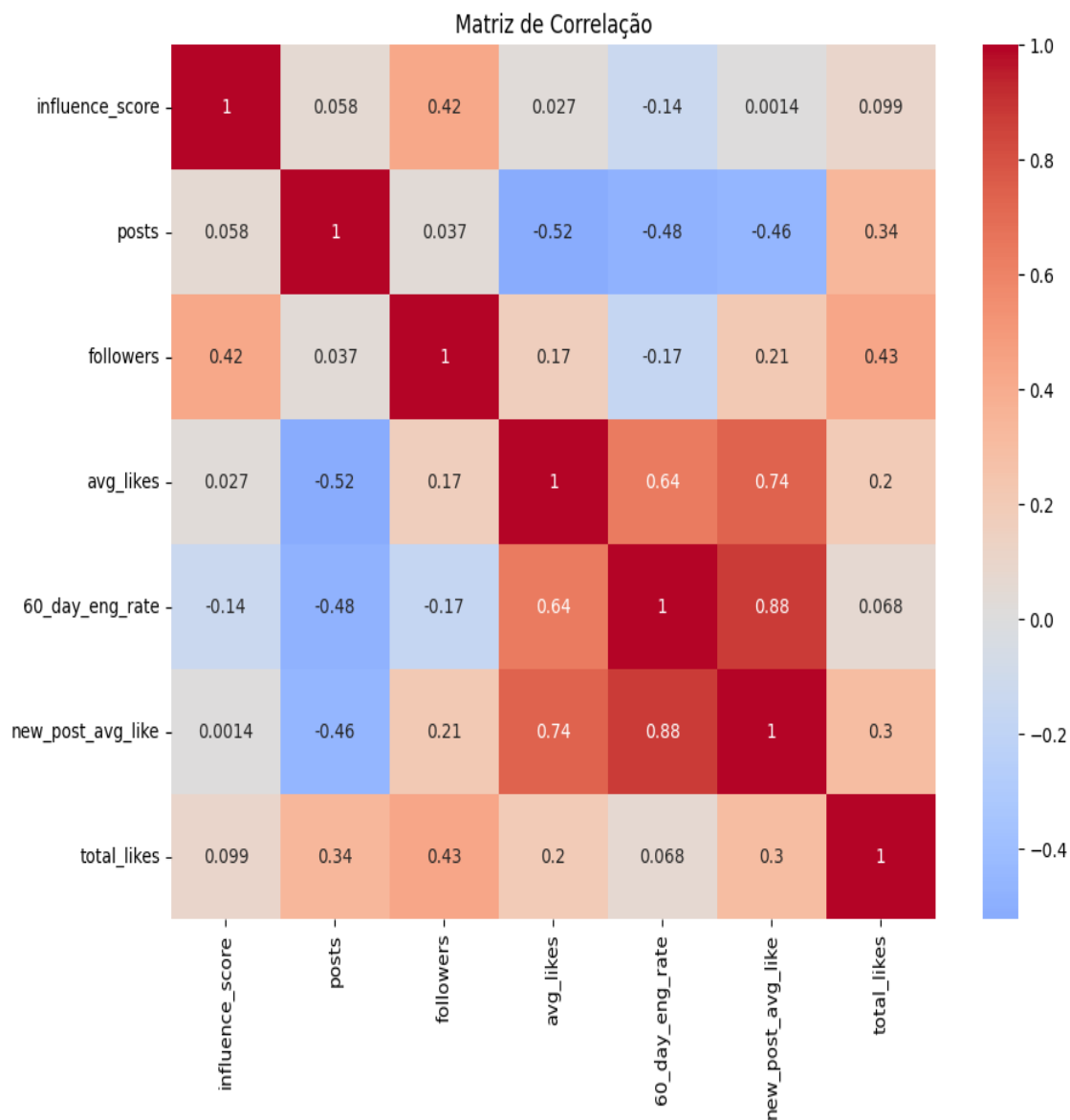
2° Nova distribuição e relação entre variáveis

Conseguimos evidenciar mais ainda a relação linear entre a variável independente `new_post_avg_like` e a variável dependente `60_day_eng_rate`, o que é ótimo para precisão do nosso modelo. Percebe-se também, que quase todas as distribuições de variáveis seguem o mesmo padrão com muitos valores baixos e menos valores altos, exceto pela variável independente `influence_score`, que segue uma distribuição mais central. As variáveis `rank`, `country` e `channel_info` foram removidas por terem relação quase nula com a variável dependente.



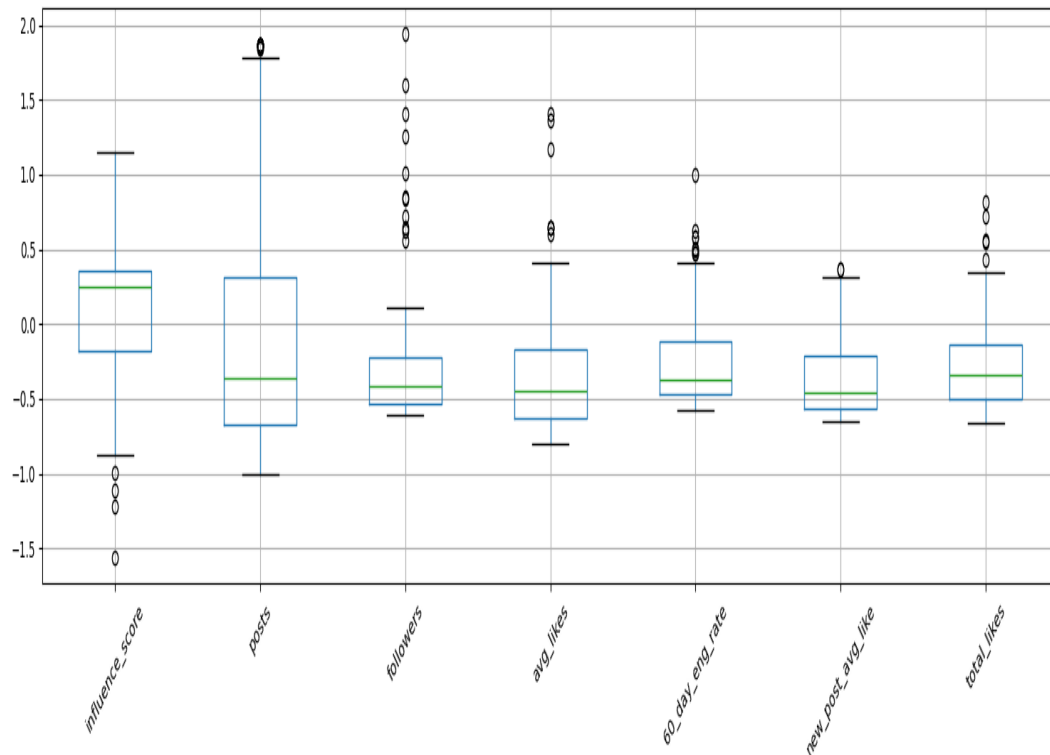
3ª Nova Matriz de Correlação

Percebe-se que a relação entre a variável dependente `60_day_eng_rate` e as variáveis independentes `new_post_avg_like` e `avg_likes` permanecem da mesma forma. Entretanto, conseguimos reduzir a correlação entre `new_post_avg_like` e `avg_likes` para um nível aceitável, abaixo de 0.75. Decidimos por não usar técnicas de redução de dimensionalidade em variáveis independentes com correlação abaixo de 0.8, e também optamos por não fundir as duas variáveis porque `avg_likes` também está com uma correlação considerável com a variável dependente. A variável independente `total_likes` que possuía uma correlação considerável com `followers` não possui mais, a correlação entre as duas foi bastante reduzida. Dessa forma, os dados ficaram melhores para treinar o modelo.



4° Distribuição das Features após tratamento de Outliers

Aqui já temos uma redução considerável na quantidade de outliers em todas as variáveis independentes citadas anteriormente, principalmente na `new_post_avg_like` que apenas sobrou um outlier. Dessa forma, nosso modelo não será tão afetado pelos outliers no data set.



Implementação do Algoritmo

O algoritmo foi implementado usando 3 modelos diferentes da biblioteca `sklearn.linear_model` sendo eles o modelo regular de regressão linear, o de Lasso e o de Ridge. Eles foram treinados com os mesmos dados para comparação posteriormente. Depois de alguns testes decidimos que 0.75% é o valor ideal para treinamento do nosso modelo. E claro, todos os modelos foram treinados com dados preprocessados.

Validação e Ajuste de Hiperparâmetros

Escolha das variáveis

As variáveis independentes foram escolhidas por dois principais motivos, 1º forte correlação com a variável dependente, 2º baixa correlação entre as outras variáveis independentes. Dessa forma temos, que as variáveis independentes como `influence_score`, `followers` e `total_likes`, atendem o primeiro critério, enquanto as `avg_likes` e `new_post_arg_like` atendem o segundo e por fim a variável independente `post` que é um meio termo entre o primeiro e segundo. Variáveis como `rank`, `country` e `channel_info` possuíam uma correlação com a variável dependente muito baixa, por isso foram removidas e como elas não eram correlacionadas também, técnicas de redução de dimensionalidade não iriam ajudar.

Validação dos Modelos

Para validar os modelos foram usadas métricas de validação comuns como R^2 , MSE e MAE. E para evitar o máximo de overfitting usamos validação cruzada com o R^2 , e calculamos a média e o desvio padrão, da validação cruzada, para análise.

Otimização dos Modelos

Para otimizar todos os modelos, além de ajustar os hiperparâmetros nós deveríamos otimizar a base de dados, e assim fizemos. A base de dados foi melhorada com os seguintes passos:

- Garantimos que todos os dados estão normalizados.

- Evitamos a multicolineariedade entre variáveis independentes muito correlacionadas.

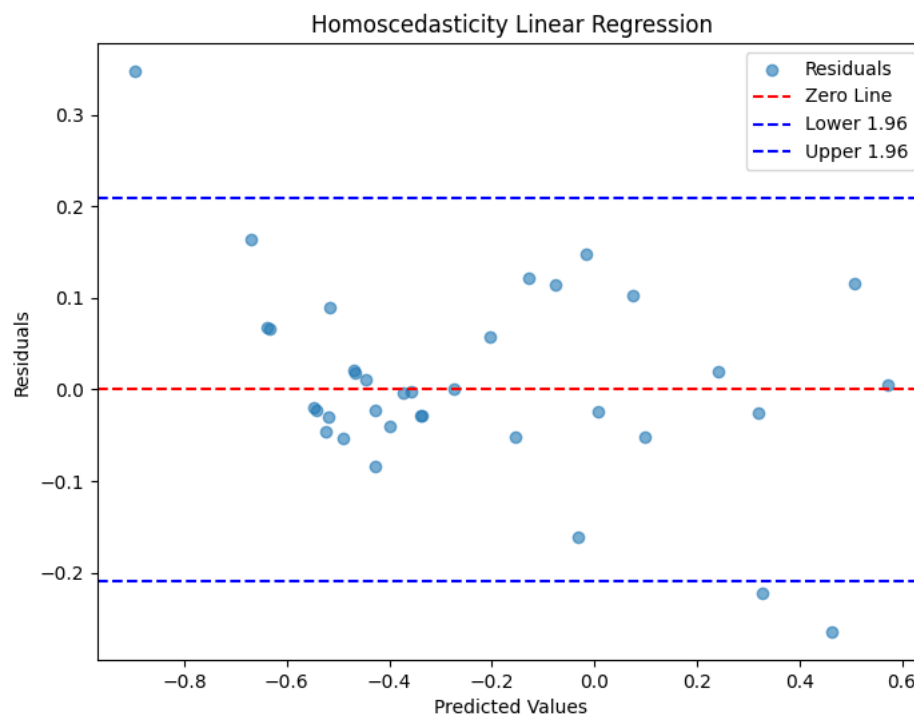
Entretando, permitimos que avg_like e new_post_avg_like pois ambas compartilhavam uma correlação de alto valor com a variável dependente.

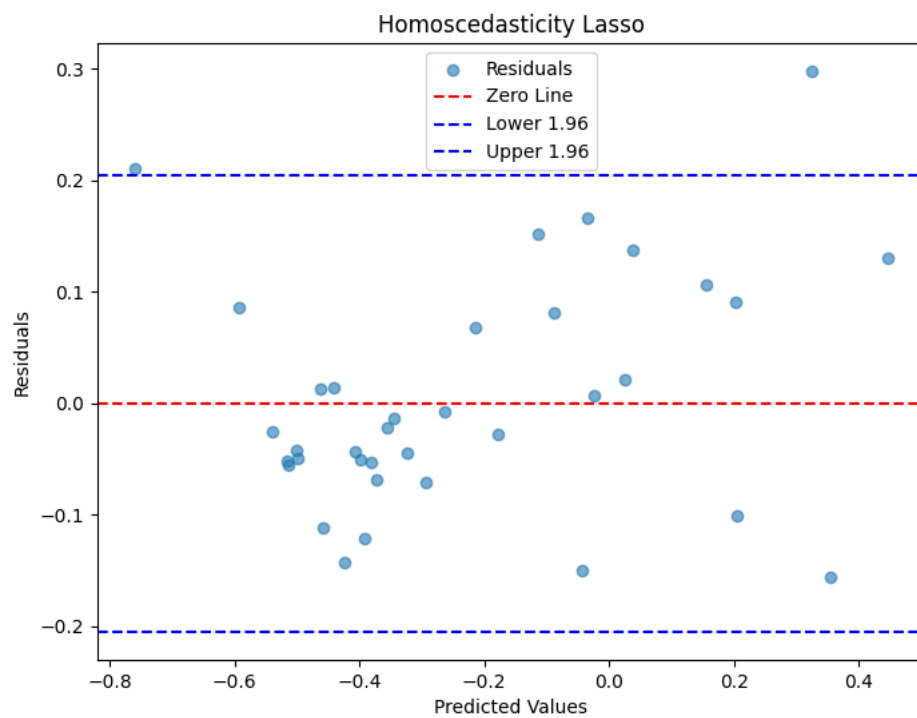
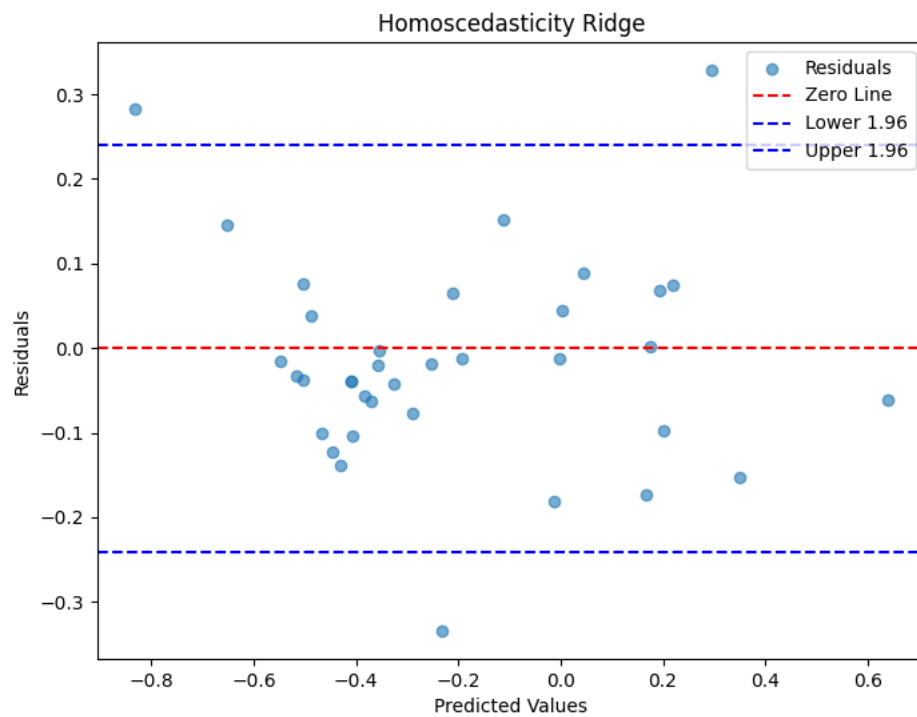
- Ajustamos o hiperparâmetro alpha de Ridge e Lasso.

Para o modelo Ridge um hiperparâmetro $\alpha = 1$ consideramos ideal, penalizando moderadamente os coeficientes do modelo. E para o modelo de Lasso, $\alpha = 0.01$ já é o suficiente para zerar alguns coeficientes e e fornecer boas métricas de desvio padrão e média do R^2 da validação cruzada.

- Evitamos que os dados sofram de heteroscedasticidade.

Percebe-se que temos pouquíssimos dados que saem do intervalo de confiança dos modelos, e ainda assim os valores que saem do intervalo de confiança não são extrapolantes, dessa forma eles podem ser aceitos.





Resultados

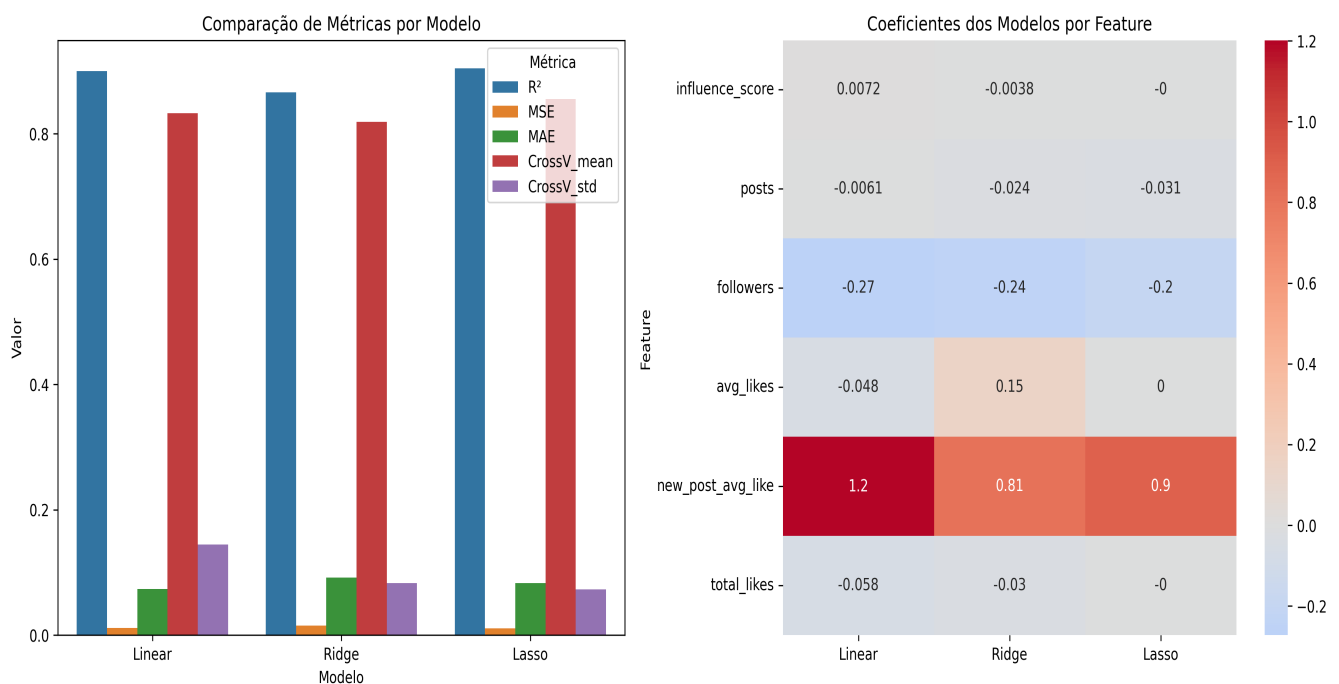
Métricas e Coeficientes de Correlação

Métricas

Conseguimos uma boa porcentagem em cada uma das métricas utilizadas para validar o modelo, o R^2 em cada um dos modelos está em aproximadamente 0.90, no qual é um ótimo resultado para o nosso data set, também utilizamos a validação cruzada para o R^2 que ficou entre 0.8 à 0.86 nos modelos, o que é um ótimo resultado. O desvio padrão do R^2 também foi realizado com validação cruzada resultando em um valor entre 0.07 à 0.15 nos modelos, consideramos um valor aceitável. As métricas estão dispostas no gráfico da esquerda.

Coeficientes

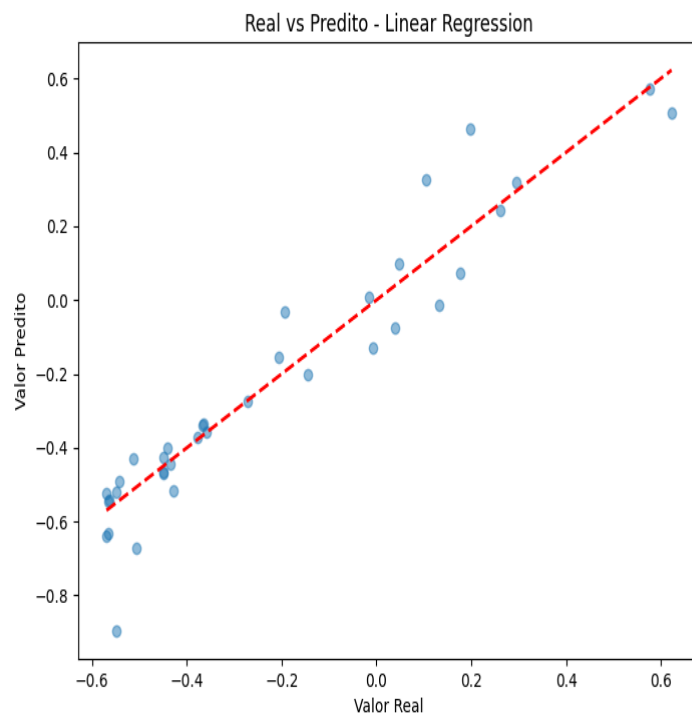
Nota-se que a variável independente de longe mais importante nos nossos modelos é a `new_post_avg_like`, em todos os modelos ela apresenta um alto valor de correlação com a variável dependente, ou seja ela é a principal variável do nosso modelo. Outra variável que apresenta uma correlação inversa com o nosso target é a `followers`, tendo um coeficiente de correlação negativo em todos os modelos. Uma observação que o modelo Lasso deixa bem evidente, é que não são necessárias variáveis como `influence_score`, `total_likes` e `avg_likes` para o desempenho do nosso modelo. Se à removermos teremos um modelo tão bom quanto. Deixando opcional a escolha de deixar as variáveis ou remove-las.



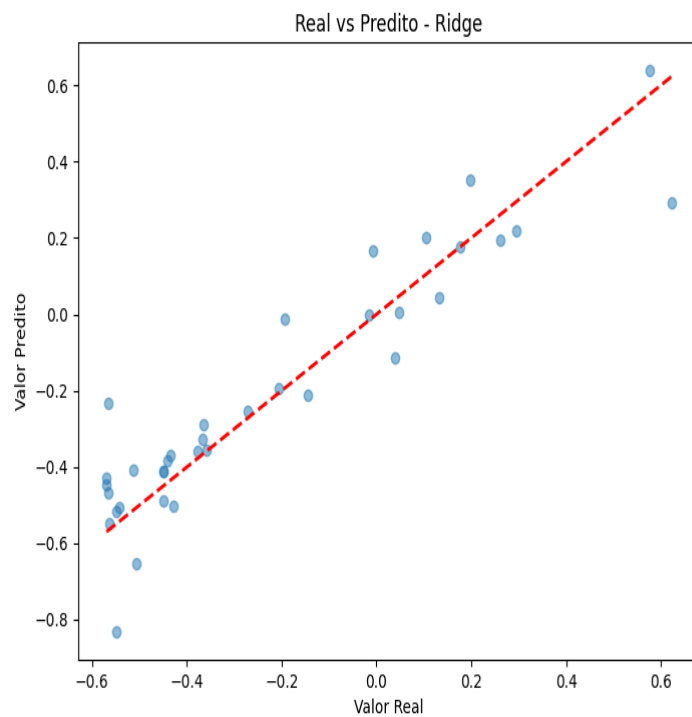
Visualização dos modelos

Ambos os 3 modelos apresentam uma abordagem linear em relação aos dados, isso é evidenciado por cada um dos gráficos a seguir:

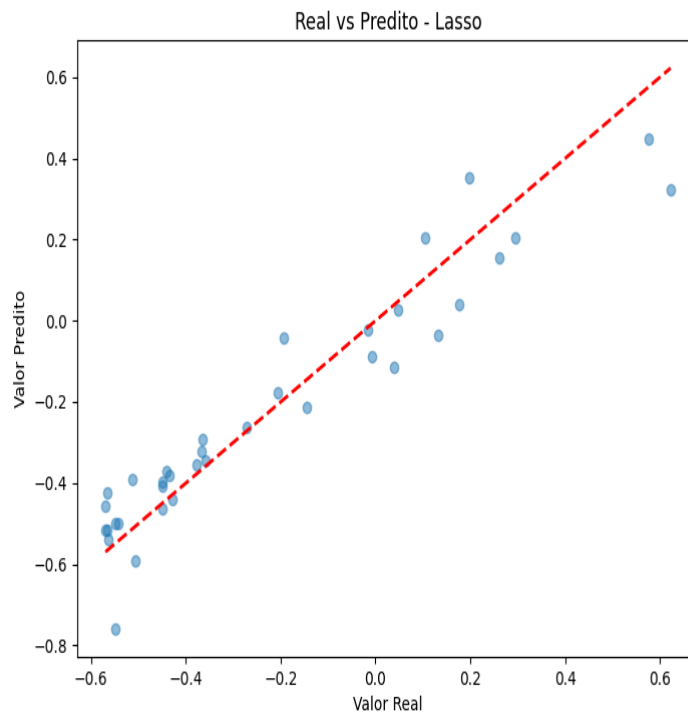
Modelo de Regressão Linear



Modelo de Ridge



Modelo de Lasso



Discussão

A partir dos resultados encontrados nós podemos simplificar o modelo e se quiséssemos, mas como não precisamos de um modelo muito simples, que por sua vez pode não capturar as complexidades dos dados. Usamos 75% dos dados para treinar os modelos, o que nos resultou em apenas 100 valores aproximadamente, com um conjunto maior de dados talvez conseguiríamos uma precisão maior. Enquanto ao desempenho do modelo achamos que ambos os três ficaram bons para prever a taxa de engajamento uma assertividade de 0.85% em média na validação cruzada do R^2 indicando um bom desempenho do modelo. Testamos com um mais simples também, a única coisa necessária a se fazer é dar um drop nas variáveis `avg_likes`, `total_likes` e `influence_score`, caso precisássemos de simplicidade poderíamos fazer. Reconheço também, que o código não está tão claro, algum tempo de refatoração melhoraria e muito.

Conclusão e trabalhos futuros

Principais Aprendizados

Os modelos foram eficazes para prever engajamento identificando características mais importantes. No entanto, pode ser simplificado.

Propostas de Melhoria

- Inclusão de features temporais: Pode ajudar o modelo a capturar padrões mais detalhados
- Experimentação com modelos não-lineares: pode melhorar a capacidade de prever resultados em cenários mais complexos

Analisar subgrupos específicos: Dividir o dataset em segmentos menores pode revelar padrões que o modelo pode não ter achado.