Relatório Experimental: Predição de Preços de Diamantes

Isabella Rodrigues de Oliveira[1]

[1]Faculdade de Computação e Informática – Universidade Presbiteriana Mackenzie (UPM)
São Paulo – SP – Brasil

10357696@mackenzista.com.br

Resumo. *Este trabalho aborda a precificação de diamantes através do desenvolvimento de modelos de regressão com Aprendizado de Máquina. O dataset Diamonds foi pré-processado realizando a remoção de outliers, codificação de variáveis categóricas e normalização. Três etapas foram realizadas: (I) comparação preliminar de seis regressores resultou em Multilayer Perceptron como baseline; (II) ajuste fino de hiperparâmetros do baseline; (III) comparação do baseline com algoritmos utilizados em trabalhos semelhantes. A última etapa resultou em um regressor CATBoost com menor erro dentre os modelos testados deste e de dois outros trabalhos. Assim, provando a efetividade de modelos ensemble e baseados em árvores. Código disponível em [R. de Oliveira 2025].*

1. Introdução

A precificação de diamantes é uma tarefa de alta complexidade, tradicionalmente dependente da avaliação de especialistas sobre um conjunto de atributos, como os ”4 Cs”: quilate (*carat*), lapidação (*cut*), cor (*color*) e pureza (*clarity*). A relação não-linear entre essas características torna o processo suscetível a inconsistências. Neste cenário, o Aprendizado de Máquina (AM) surge como uma ferramenta assistente utilizando algoritmos para analisar dados em larga escala e construir modelos de predição que trazem maior objetividade e precisão para o mercado.

O presente trabalho tem como objetivo desenvolver e avaliar modelos de regressão para prever o preço de diamantes. Utilizando um *dataset* público, será conduzido um processo de análise e tratamento de dados que abrange desde a codificação de variáveis categóricas e normalização até a remoção de *outliers* que possam distorcer os resultados. Um modelo *baseline* foi construído por meio do treino e avaliação de diferentes modelos como: Regressão de Vetores de Suporte (SVR), Árvore de Decisão, K-Vizinhos mais Próximos (KNN), Perceptron Multicamadas (MLP), Regressão Linear e AdaBoost. Em seguida, o *baseline* foi refinado e comparado com modelos obtidos por trabalhos semelhantes. Consequentemente, alcançou-se um conjunto de modelos finais robustos para a tarefa de precificação de diamantes.

2. Fundamentação Teórica

Esta seção é composta de duas subseções, cuja primeira discorre sobre trabalhos relacionados e resultados obtidos por outros autores e a segunda apresenta, brevemente, a teoria dos algoritmos selecionados.

2.1. Trabalhos Relacionados

No trabalho [Alsuraihi et al. 2020], os autores aplicam *one hot encoding* para transformar três variáveis categóricas em valores nominais, separam 80% do *dataset* para treino, 20% para teste e treinaram os algoritmos de regressão linear, regressão de floresta aleatória, regressão polinomial, gradiente descendente e redes neurais. Os autores concluem, por meio do erro médio quadrático, que o método de floresta aleatória demonstra melhor resultado.

Em [Mihir et al. 2021], foi proposto o desenvolvimento de uma Interface Gráfica do Usuário (GUI) para predição de preço e seleção de modelo baseado nos treinos com os algoritmos de regressão linear, regressão com vetores de suporte, árvores de decisão, regressão de floresta aleatória, K-vizinhos mais próximos, regressão CatBoost, regressão de Huber, regressão com árvore extra, regressão passivo agressiva, regressão Bayesiana e regressão XGBoost. No pré-processamento, os autores aplicaram apenas a codificação dos atributos categóricos e a divisão do *dataset* em 70% treino e 30% teste. Os autores demonstraram que regressão CATBoost, regressão XGBoost e regressão de floresta aleatória tiveram resultados superiores aos outros métodos.

Os autores de [Sharma et al. 2021] realizaram uma análise comparativa de oito algoritmos supervisionados, sendo eles: regressão linear, regressão lasso, regressão ridge, árvore de decisão, regressão de floresta aleatória, ElasticNet, regressão AdaBoost e regressão Gradient-Boosting. O *dataset* foi dividido em 80% treino e 20% teste de maneira estratificada usando o atributo quilate como referência. Os melhores modelos foram os de regressão de floresta aleatória e regressão de árvore de decisão segundo a métrica de erro médio quadrático.

2.2. Algoritmos Selecionados

Inicialmente, seis algoritmos de AM foram selecionados para a definição de um *baseline*, sendo eles: SVR, Árvore de Decisão, KNN, MLP, Regressão Linear e AdaBoost. Os parágrafos a seguir descrevem brevemente cada método.

O primeiro método, SVR, estende os princípios de uma máquina de vetores de suporte (SVM) que busca justapor uma linha de regressão conhecida como hiperplano no espaço dos dados. A equação desse hiperplano é depois usada para predizer novos valores.

Árvores de Decisão são modelos de aprendizado supervisionado que criam regras simples para predizer um determinado valor. Elas separas os dados baseadas em atributos nos quais cada nó da árvore toma uma decisão até um nó folha que contenha a predição final.

O método KNN não se preocupa em construir um modelo. Nesse caso, cada instância é gravada na memória e, ao receber uma instância de dados nova, o modelo calcula sua distância dos dados salvos e prediz o valor final por meio de uma média dos valores mais próximos.

Um MLP é um modelo simples de rede neural totalmente conectada composto de uma camada de entrada, uma ou mais camadas escondidas e uma camada de saída, sendo cada camada composta de um determinado número de neurônios. O valor recebido por

um neurônio é propagado para todos os neurônios da camada subsequente por meio da seguinte fórmula:

$$a = g\left(z\right)$$

Onde a é o valor de saída resultante da aplicação da função de ativação $g()$ na soma ponderada z, representada pela fórmula

$$z = \sum_{i=1}^{n}(w_i x_i) + b$$

Onde b é o termo de *bias*, w_i é o peso associado ao i-ésimo neurônio da camada anterior e x_i é o valor propagado pelo i-ésimo neurônio da camada anterior.

No caso da Regressão Linear, a mesma busca traçar uma reta que correlaciona uma variável dependente a uma ou mais variáveis independentes. Sua predição é dita pela seguinte equação:

$$\hat{y} = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$$

Onde y é o rótulo predito, x são os dados de entrada, $\theta_{1,2,n}$ são os coeficientes de x e θ_0 é o coeficiente linear da reta.

Por últiom, o princípio central do método AdaBoost consiste em criar uma sequência de preditores fracos em versões modificadas dos dados. As predições de todos são combinadas por meio de um voto ponderado para produzir uma predição final [Scikit-Learn 2025].

3. Descrição do Problema

A avaliação e precificação de pedras de diamante possuem desafios significativos para geminologistas devido à variação das características das pedras. O presente trabalho visa, por meio do uso de AM, facilitar a predição dos preços de pedras de diamante baseadas em características como: quilate, proporções, clareza, dentre outros.

4. Aspectos Éticos

Por consequência da adoção de modelos de linguagem em larga escala e uso de dados pessoais no meio digital, leis foram criadas para proteger indivíduos e suas informações. Como exemplo, temos a Lei Geral de Proteção de Dados (LGPD) no Brasil e a *General Data Protection Regulation* (GDPR) na União Europeia. Porém, o presente projeto não trabalha com dados pessoais de indivíduos e, portanto, não demanda preocupações relacionadas às regulações citadas anteriormente ou tratamentos de anonimização dos dados.

No que tange ao uso do sistema proposto, é importante notar que os artefatos a serem gerados possuirão limitações inerentes. Na possível implementação do modelo em um sistema de predição, é importante que sua documentação seja transparente quanto aos algoritmos e *dataset* utilizados para a compreensão do usuário, de forma que a interpretação final seja feita de forma consciente.

5. Dataset

O conjunto de dados escolhido para o projeto é o conjunto *Diamonds*, disponível na plataforma Kaggle [Agrawal 2025]. Esse *dataset* é composto de 53.940 instâncias e 10 variáveis, são elas: *price, carat, cut, color, clarity, x, y, z, depth* e *table*.

A variável *price* representa o preço em dólares americanos ($326–$18,823), *carat* representa o peso do diamante (0.2–5.01), *cut* representa a qualidade do diamante *(Fair, Good, Very Good, Premium* e *Ideal)*, *color* representa a cor do diamante (de J a D em ordem de qualidade ascendente), *clarity* representa a clareza do diamante (I1, SI2, SI1, VS2, VS1, VVS2, VVS1 e IF), *x* representa a altura em mm (0–10.74), *y* representa a largura em mm (0–58.9), *z* representa a profundidade em mm (0–31.8), *depth* representa a porcentagem da profundidade total com fórmula: $2z/(x+y)$ (43–79), e *table* representa a largura da mesa no topo do diamante em relação ao ponto mais largo (43–95).

Para suavizar o processo de ajuste dos modelos de predição, algumas técnicas de pré-processamento são comumente adotadas. A remoção de dados incongruentes é importante para auxiliar o processo de interpretação dos algoritmos de AM; isso inclui dados corrompidos e *outliers*. A análise de correlações identifica quais atributos estão fortemente correlacionados com o atributo objetivo (preço) e pode justificar a remoção ou tratamento adicional de uma coluna inteira de atributos. O processo de codificação de atributos categóricos consiste em transformar dados textuais e, possivelmente, sem relações hierárquicas em dados numéricos, sendo vital para o seu uso em algoritmos de AM.

5.1. Análise e Pré-Processamento

Primeiramente, foram utilizadas as seguintes bibliotecas: Pandas [pandas development team 2020] para trabalhar com dados tabulares, NumPy [Harris et al. 2020] para manipular dados em formato de arranjos e realizar operações numéricas eficientemente, Matplotlib [Hunter 2007] para criação de gráficos e scikit-learn [Buitinck et al. 2013] que oferece ferramentas para análise de dados e modelagem preditiva.

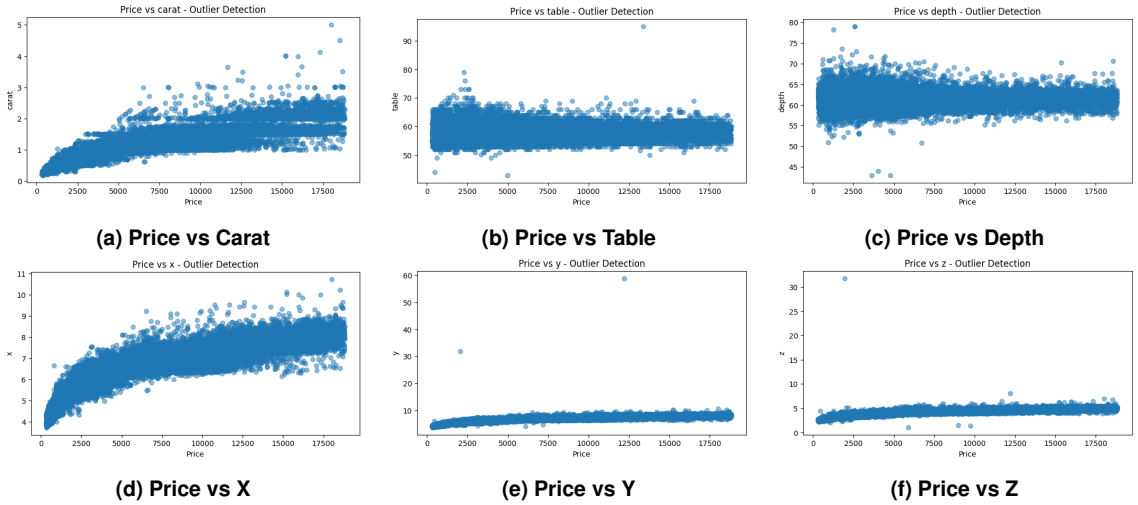
Iniciando o pré-processamento de dados, foi identificado que a primeira coluna da tabela continha dados referentes ao número das instâncias, o que se mostrou irrelevante para a análise e, portanto, a coluna foi removida. Após isso, por meio do método *describe* foi descoberta a existência de valores mínimos iguais a 0 nas colunas *x, y* e *z*, visíveis na Tabela 5.1. O que não faz sentido, visto que esses atributos dizem respeito às dimensões do diamante, portanto, tais instâncias também foram removidas.

	carat	*depth*	*table*	*price*	x	y	z
Média	0.798	61.749	57.457	3932.799	5.731	5.735	3.539
DP	0.474	1.433	2.234	3989.439	1.122	1.142	0.705
Mínimo	0.2	43.0	43.0	326.0	0.0	0.0	0.0
Máximo	5.01	79.0	95.0	18823.0	10.740	58.90	31.80

Tabela 1. Informações resultantes do método *describe* do *dataframe* Pandas.

Posteriormente, foi realizada a codificação de dados categóricos (sendo as variáveis *cut*, *clarity* e *color*) em dados numéricos a fim de estarem aptos à interpretação

por algoritmos de AM. Após isso, análises gráficas determinaram a existência de *outliers* a serem removidos (Figura 1). O número de instâncias do dataset após todas as remoções foi de 53910.



(a) Price vs Carat **(b) Price vs Table** **(c) Price vs Depth**

(d) Price vs X **(e) Price vs Y** **(f) Price vs Z**

Figura 1. Análise gráfica dos atributos numéricos para identificação de *outliers*.

Com a análise de correlações, chegamos à conclusão de que a variável mais influente no preço é a *carat*. Logo, o dataset foi separado em "baldes"de acordo com a variável *carat* a fim de termos uma separação estratificada do conjunto em 80% treino e 20% teste.

6. Metodologia

Para desenvolver e avaliar os diversos métodos de predição, foi utilizada a fórmula de Erro Quadrático Médio (RMSE), que informa a quantia de erros presentes no sistema desenvolvido e é descrita pela Equação $\text{RMSE}(y, \hat{y}) = \sqrt{\frac{\sum_{i=0}^{N-1}(y_i - \hat{y}_i)^2}{N}}$ [Géron 2017]. Dentre os algoritmos avaliados, estão: SVR, Árvore de Decisão, KNN, MLP, Regressão Linear e AdaBoost.

O algoritmo com o melhor desempenho dentre os mencionados anteriormente foi usado como *baseline* e comparado com os melhores algoritmos dos trabalhos relacionados: XGBoost, CATBoost e Regressão de Floresta Aleatória.

Espera-se que, por meio do "*input*" do usuário, o modelo retorne uma predição que sirva de base para a precificação de um diamante. Espera-se também que o modelo demonstre bom ajuste aos dados (sem *overfitting* ou *underfitting*) e que, dada a métrica de erro, seu desempenho seja satisfatório.

7. Resultados

Seis modelos foram treinados e validados usando *cross-validation* em cinco *folds*. De acordo com os resultados na tabela 2, o regressor MLP com três camadas escondidas, sendo a quantidade de neurônios respectivamente 50, 50 e 20, função de ativação unidade linear retificada e otimizador Adam demonstrou melhor resultado preliminar.

Para refinar o resultado do modelo MLP, uma busca em grade foi executada em seguida, variando a quantidade de camadas escondidas, número de neurônios por camada,

Modelo	RMSE
MLP Regressor	-605,35
Decision Tree	-758,07
KNN	-822,17
AdaBoost Regressor	-1166,90
Linear Regression	-1214,76
SVR	-4056,58

Tabela 2. Valores de RMSE negativos por modelo de acordo com *cross-validation* de cinco *folds*, onde quão mais próximo de zero melhor o desempenho.

função de ativação, otimizador e taxa de aprendizado. Certas combinações resultaram em erros de convergência, com a MLP original ainda demonstrando melhor resultado de RMSE igual a 591,69. Resultados do conjunto de teste estão dispostos na tabela 3.

Modelo	RMSE
MLP Regressor	591,69
Decision Tree	718,04
KNN	764,18
AdaBoost Regressor	1174,09
Linear Regression	1198,09
SVR	4054,83

Tabela 3. Resultados de RMSE no conjunto de testes por modelo

Esses resultados preliminares oferecem um *baseline* com o modelo MLP que, em seguida, foi comparado com os melhores regressores encontrados pelos trabalhos [Sharma et al. 2021], [Mihir et al. 2021] e [Alsuraihi et al. 2020]. A tabela 4 demonstra que o regressor CATBoost oferece melhor desempenho com RMSE igual a 509,55. Apesar de não alcançar o resultado dos modelos utilizados em outros trabalhos, o MLP *baseline* continua competitivo tendo uma diferença de erro de apenas 53,87 quando comparado com o próximo melhor modelo.

Modelo	RMSE
CatBoost	509,55
XGB	528,34
Random Forest	529,88
MLP Regressor	583,65

Tabela 4. Comparação do baseline obtido com os regressores usados nos trabalhos de [Sharma et al. 2021], [Mihir et al. 2021] e [Alsuraihi et al. 2020]

.

Nota-se que o valor obtido pelo CATBoost do presente trabalho foi mais favorável que os melhores modelos encontrados nos trabalhos [Sharma et al. 2021] e [Mihir et al. 2021], respectivamente: regressão de floresta aleatória, que obteve RMSE igual a 581,90 e o regressor CATBoost, que obteve RMSE igual a 525.81. Acredita-se que esse resultado superior se deve a técnicas de pré-processamento mais sofisticadas que foram utilizadas no presente trabalho, como remoção de outliers e dados inválidos e divisão

estratificada dos conjuntos de treino e teste. Ademais, o trabalho [Alsuraihi et al. 2020] obteve valor de RMSE igual a 241,00 para o método de florestas aleatórias, no entanto os autores não explicitaram os parâmetros do modelo e a única diferença de pré-processamento é que os mesmos utilizaram *one-hot encoding*. Dessa forma, não foi possível replicar os resultados.

8. Conclusão

O presente trabalho desenvolveu e avaliou modelos de regressão baseados em AM para a tarefa de precificação de diamantes. Foram aplicadas várias técnicas de pré-processamento de dados, como a identificação e remoção de *outliers* e instâncias inválidas, separação estratificada do conjunto de dados e codificação de atributos categóricos. Realizou-se três rodadas de testes, a primeira estabeleceu o MLP como melhor candidato, a segunda buscou refiná-lo por meio de uma busca em grade de hiperparâmetros e a terceira comparou o MLP *baseline* com algoritmos usados em outros trabalhos.

A última etapa de testes resultou em um modelo CATBoost com menor erro dentre todos os modelos avaliados e, também, quando comparado com os melhores modelos de dois trabalhos relacionados. Em suma, o modelo de regressão CatBoost, treinado com o *dataset* pré-processado de diamantes, representa uma solução robusta e competitiva para a precificação de diamantes. Para trabalhos futuros, sugere-se o uso de outras técnicas de pré-processamento, como enriquecimento de dados e criação de novos atributos representativos ou remoção de atributos altamente correlacionados e repetitivos. Propõe-se, também, o refinamento de hiperparâmetros do regressor CATBoost.

Referências

Agrawal, S. (2025). Kaggle: diamonds dataset. `https://www.kaggle.com/datasets/shivam2503/diamonds?resource=download`.

Alsuraihi, W., Al-hazmi, E., Bawazeer, K., and Alghamdi, H. (2020). Machine learning algorithms for diamond price prediction. In *Proceedings of the 2020 2nd International Conference on Image, Video and Signal Processing*, IVSP '20, page 150–154, New York, NY, USA. Association for Computing Machinery.

Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., VanderPlas, J., Joly, A., Holt, B., and Varoquaux, G. (2013). API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122.

Géron, A. (2017). *Hands-On Machine Learning with Scikit-Learn & TensorFlow CONCEPTS, TOOLS, AND TECHNIQUES TO BUILD INTELLIGENT SYSTEMS*. O'Reilly, Boston - Farnham - Sebastopol - Tokyo - Beijing.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., and Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, 585(7825):357–362.

Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95.

Mihir, H., Patel, M. I., Jani, S., and Gajjar, R. (2021). Diamond price prediction using machine learning. In *2021 2nd International Conference on Communication, Computing and Industry 4.0 (C2I4)*, pages 1–5.

pandas development team, T. (2020). pandas-dev/pandas: Pandas. https://doi.org/10.5281/zenodo.3509134.

R. de Oliveira, I. (2025). Projeto_ai. `https://github.com/bellaofficial23/Projeto_AI/tree/main`.

Scikit-Learn (2025). 1.10.2. regression. `https://scikit-learn.org/stable/modules/tree.html#tree-regression`.

Sharma, G., Tripathi, V., Mahajan, M., and Kumar Srivastava, A. (2021). Comparative analysis of supervised models for diamond price prediction. In *2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pages 1019–1022.