



UNIVERSIDADE FEDERAL DO RIO DE JANEIRO



1920 | 2020



Universidade Federal
do Rio de Janeiro

Escola Politécnica

REDES NEURAIS

Multi Layer Perceptron - Regressão

Ana Thais Castro

anathaiscastro@poli.ufrj.br

Jonas Degrave

jonasdegrave@poli.ufrj.br

Índice

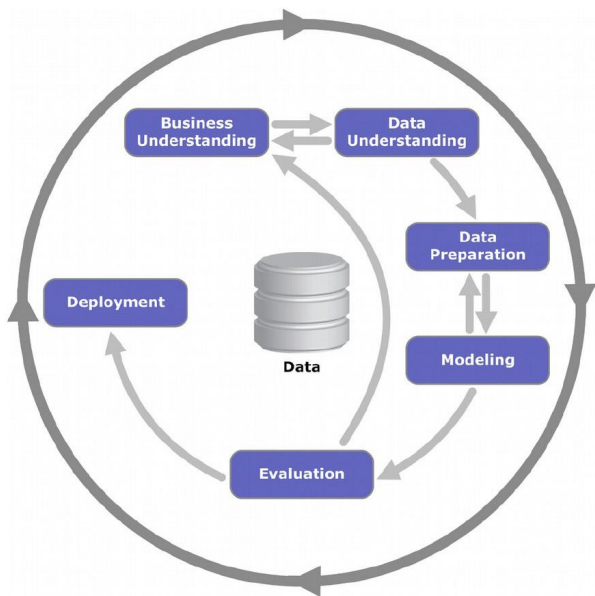
- **Introdução**
- **Metodologia**
 - Análise Exploratória
 - Pré-Processamento
 - Modelo de Regressão MLP
 - Otimização de Hiperparâmetros
- **Resultados e Conclusão**



Figura 1: Diamantes. (Fonte: Alrosa)

Introdução

O objetivo deste trabalho é explorar o projeto de hiper-parâmetros para **avaliar modelos de regressão MLP** usando um dataset para predição de preço de diamantes.



Business Understanding

- A **tarefa de aprendizado** predição do preço pode ser útil tanto do ponto de vista de quem vende quanto de quem compra, para que seja possível estimar o valor adequado a partir de algumas características de um diamante.
- Considerando a análise de resultados conhecidos de outros artigos, emprega-se como **critério de sucesso** o coeficiente de determinação (R^2 Score) comparativo aos já encontrados como benchmarking.

Figura 2: Etapas CRISP-DM. (Fonte: Udacity)

Introdução

Data Understanding

Dataset: Diamonds.csv	
Registros	54 000
Variáveis	10
Tipo	Regressão
Objetivo	Estimar Preço

x: length (mm)

y: width (mm)

z: depth (mm)

Price

in USD\$, regression target variable

Table

width of top relative to widest point

Depth %

depth (z) relative to widest point

Volume

engineered feature ($V = x * y * z$)

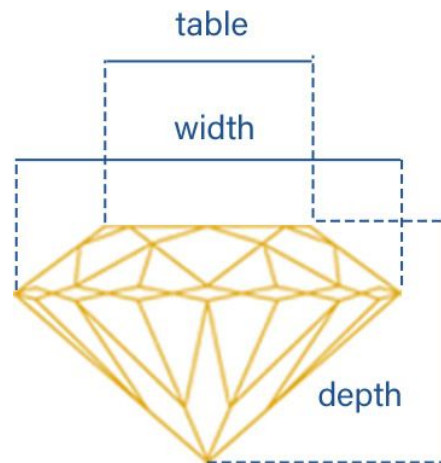


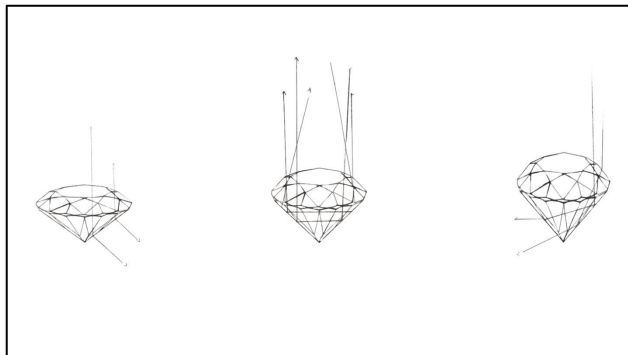
Figura 3: Dimensões de Diamantes.
(Fonte: Lumera Diamonds)

Introdução

Data Understanding

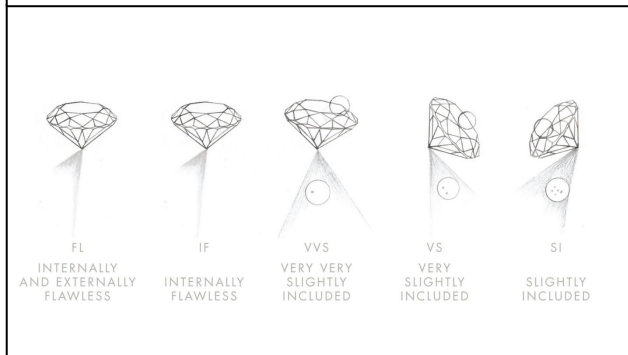
Cut (5)

1. Fair
2. Good
3. Very Good
4. Premium
5. Ideal



Clarity (8)

1. I1
2. SI2
3. SI1
4. VS2
5. VS1
6. VVS2
7. VVS1
8. IF



Color (7)

1. J (Best)
2. I
3. H
4. G
5. F
6. E
7. D (Worst)

Carat

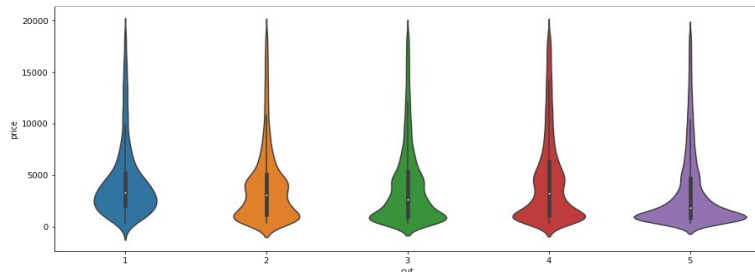
- 0.20
- [...]
- 5.00

5 carat = 1 gram

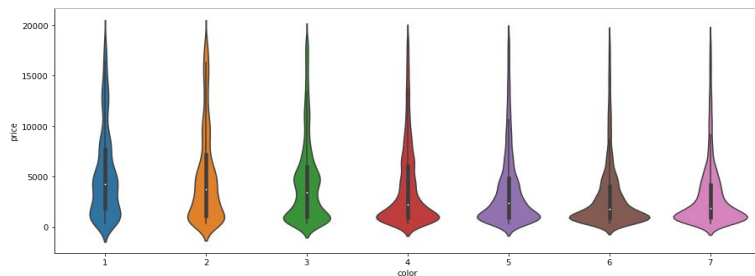
Figura 4: 4C's dos Diamantes - Cut, Clarity, Color, Carat. (Fonte: De Beers)

Análise Exploratória e Pré-Processamento

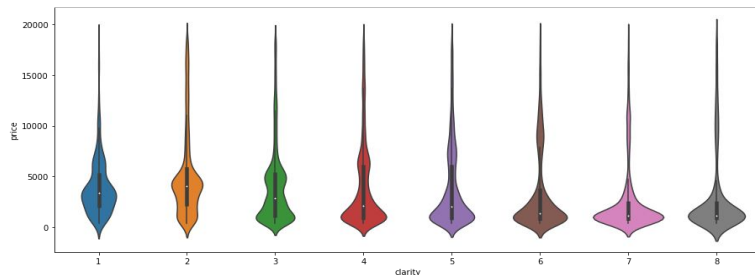
Cut



Color



Clarity

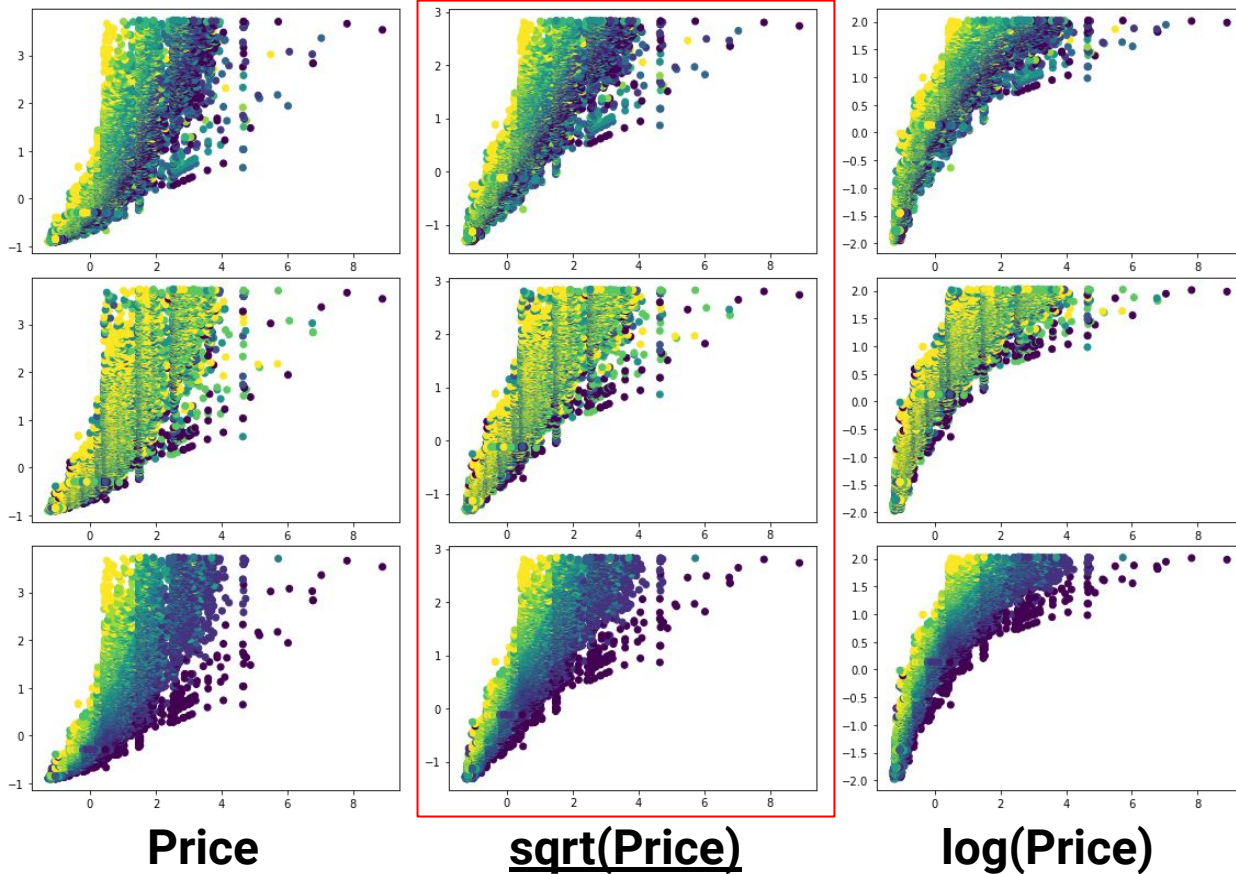


Violin Plot: Cut, Color, Clarity

- Apresenta a distribuição de registros por preço em cada categoria.
- Existe hierarquia “melhor” - “pior” para as variáveis categóricas.
- Label Encoding vs One Hot:
Atribuição de valores numéricos sequenciais às variáveis categóricas.

Análise Exploratória e Pré-Processamento

Scatter Plot:
Price vs Carat

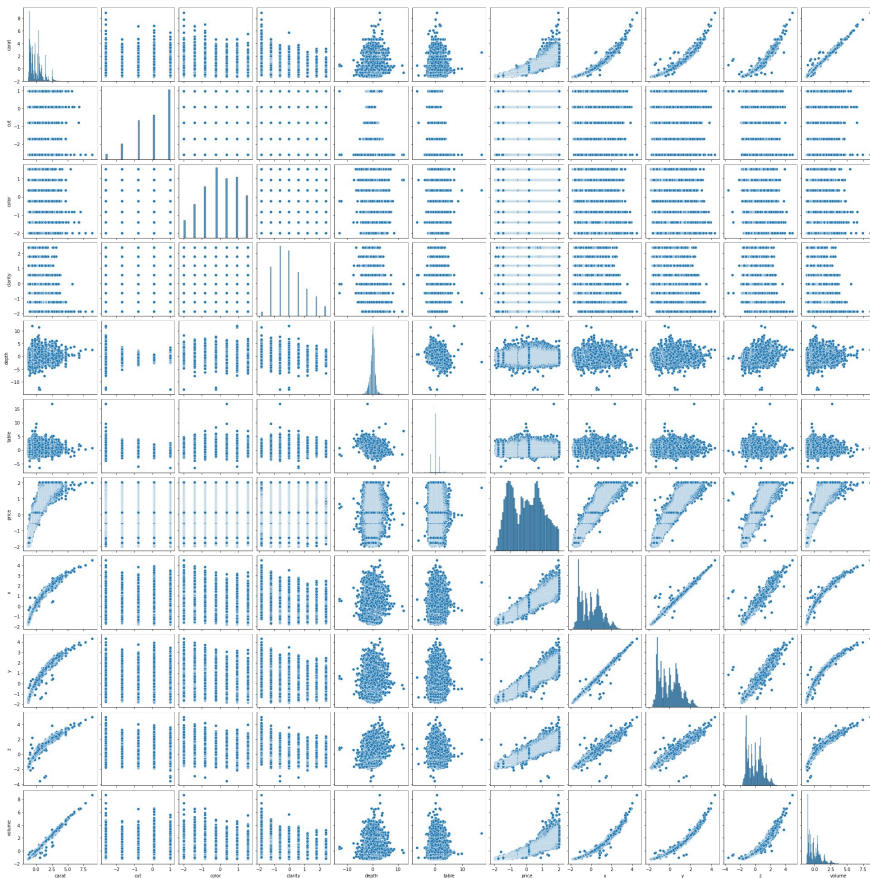


- Color

- Cut

- Clarity

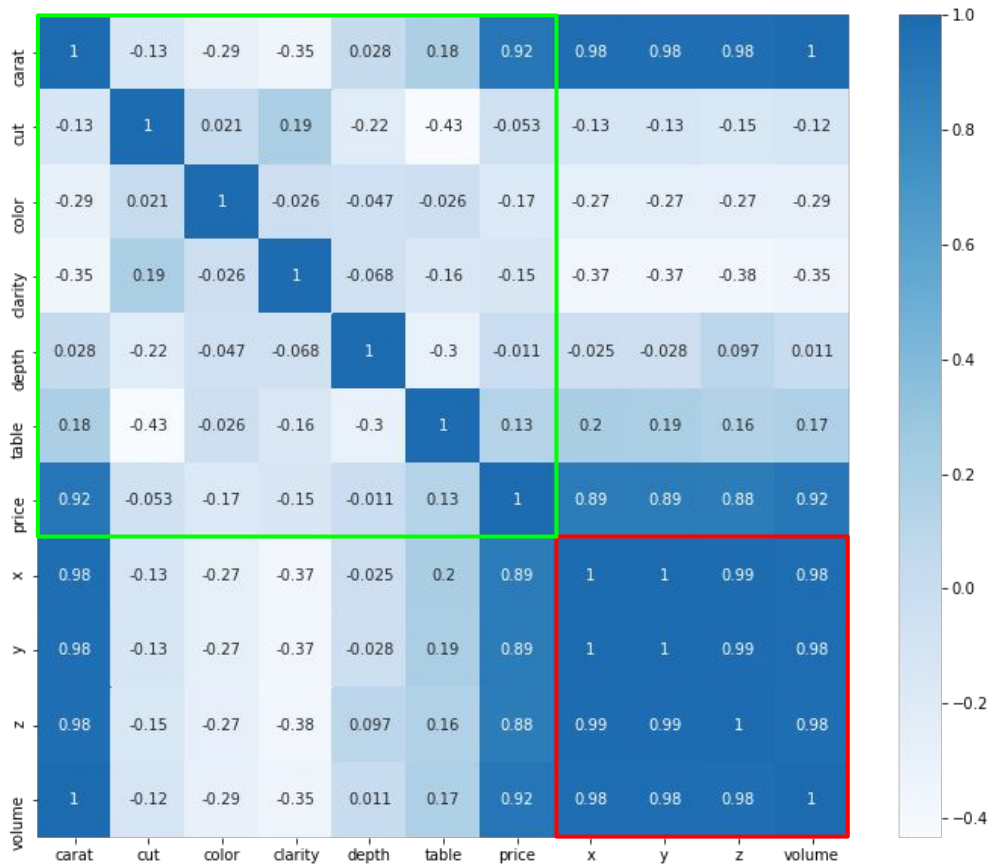
Análise Exploratória e Pré-Processamento



Pair Plot

- Data Selection: Remoção de outliers discrepantes identificados no pair plot.
- Data Cleaning: Eliminação de registros com valores nulos ou inexistentes.

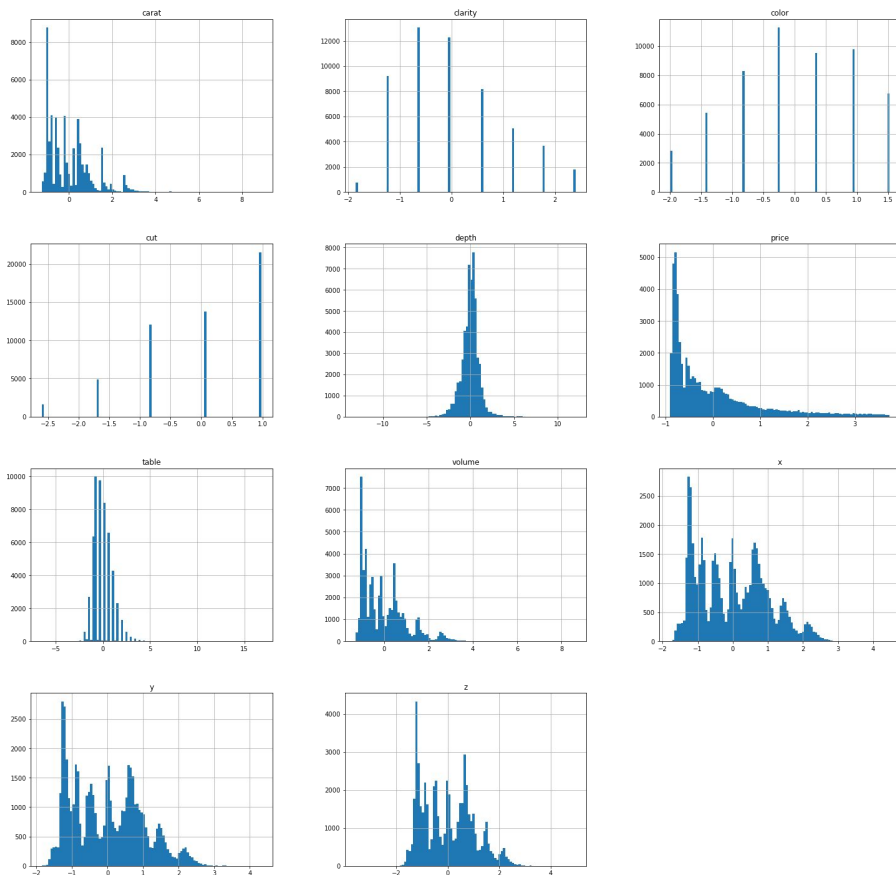
Análise Exploratória e Pré-Processamento



Correlation Heatmap Plot

- Feature Engineering
- Redução de Dimensionalidade:
Variáveis 'x', 'y' e 'z' eliminadas por
haver alta correlação com 'volume'.

Análise Exploratória e Pré-Processamento



Histogram Plot e Normalização

- Subtrair da média para centrar na origem e dividir pelo desvio padrão para normalização:

$$\text{dataNorm} = (x - \bar{x}) / \sigma$$

- Resultado: $\sigma = 1$ nas variáveis
- Melhor treinabilidade e convergência do modelo.

Modelo de Regressão MLP

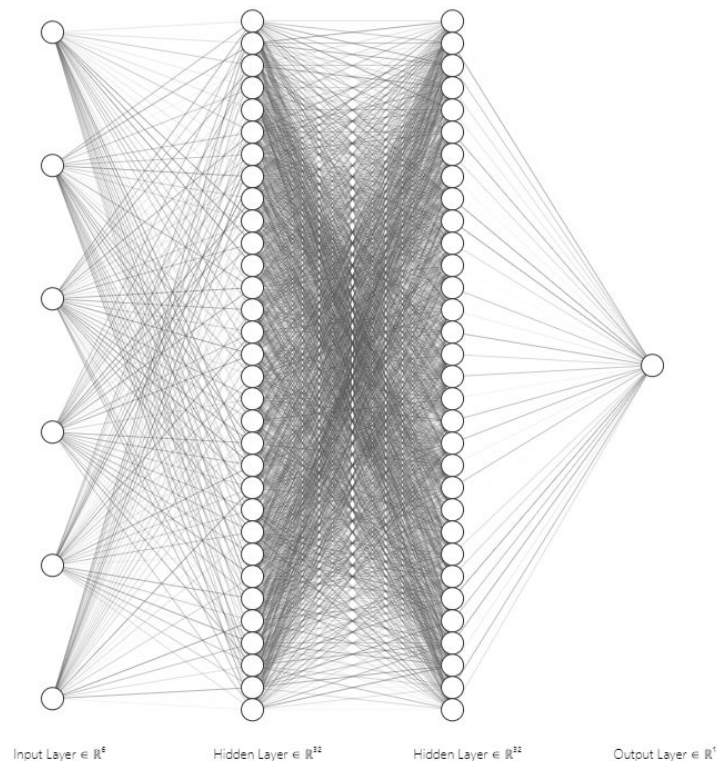


Figura 5: Rede MLP modelada. ([Fonte](#))

Modelo Base: `sklearn.neural_network.MLPRegressor()`

- 6 entradas e 1 saída
- 2 hidden layers de 100 neurônios 'tanh'
- Regularização L2: $\alpha = 0.001$
- Taxa aprendizado constante: 0.001
- Solver: Adam
- Validação Cruzada com k-fold (k=10)

```
MLPRegressor(activation='tanh', alpha=0.001, batch_size='auto', beta_1=0.9,
              beta_2=0.999, early_stopping=False, epsilon=1e-08,
              hidden_layer_sizes=(100, 100), learning_rate='constant',
              learning_rate_init=0.001, max_fun=15000, max_iter=1000,
              momentum=0.9, n_iter_no_change=10, nesterovs_momentum=True,
              power_t=0.5, random_state=None, shuffle=True, solver='adam',
              tol=1e-06, validation_fraction=0.1, verbose=True,
              warm_start=False)
```

Metodologia | Otim. Hiperparâmetros

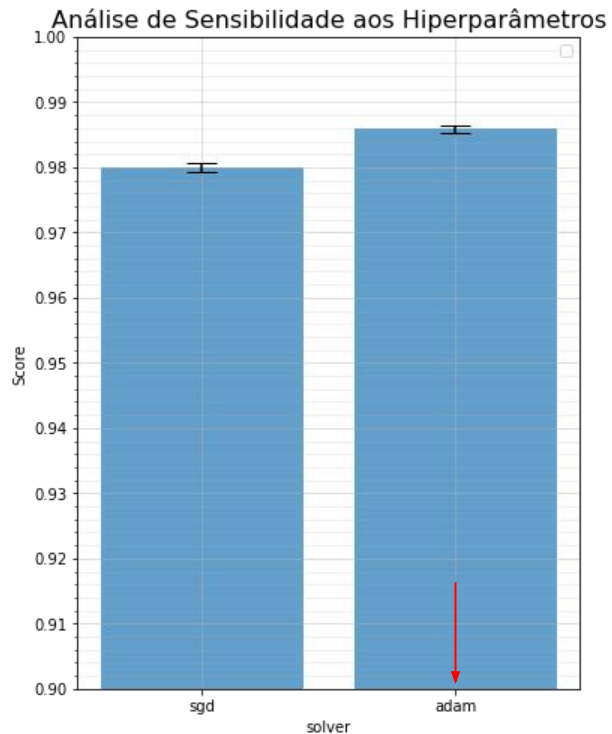
Otimização de Hiperparâmetros: `sklearn.model_selection.GridSearchCV()`

- 7 tipos de hiperparâmetros em estudo
- Combinatória: 622 080 opções com 6.2 milhões de modelos k-fold (impossível computar)
- Sem Combinatória: 71 buscas com 710 modelos com k-fold (computacionalmente viável)

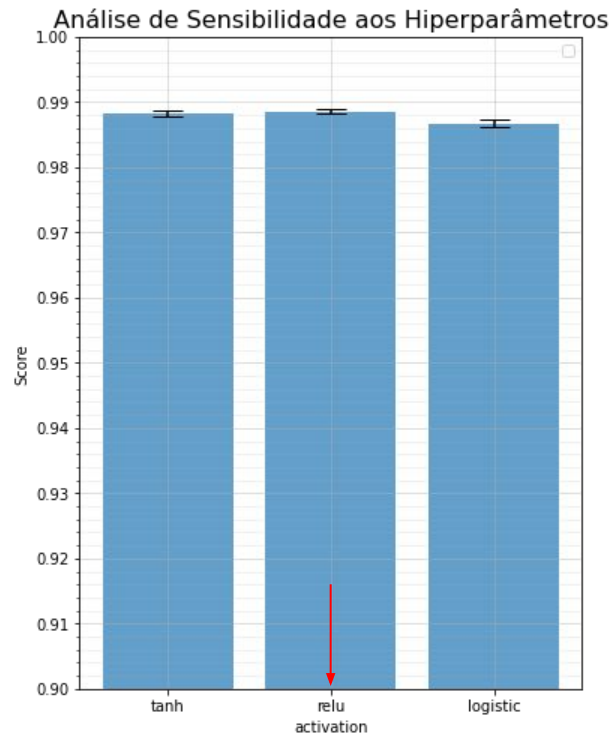
```
param_grid = [  
    {'solver': ['sgd', 'adam'], 'max_iter': [5000], 'tol': [1e-04], 'hidden_layer_sizes':[(32,32)]},  
    {'learning_rate_init': [0.1, 0.01, 0.001, 0.0001, 0.00001]},  
    {'activation': ['tanh', 'relu', 'logistic']},  
    {'early_stopping': [True], 'validation_fraction': [0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6]},  
    {'alpha': [0.0001, 0.001, 0.01, 0.1, 1, 0.00001, 0.000001, 10, 100]},  
    {'tol': [0.0001, 0.001, 0.01, 0.1, 0.00001, 0.000001, 0.0000001, 0.00000001]},  
    {'hidden_layer_sizes':[(1), (2), (4), (8), (16), (32), (64), (128), (256), (512)]},  
    {'hidden_layer_sizes':[(1,1), (2,2), (4,4), (8,8), (16,16), (32,32), (64,64), (128,128), (256,256), (512,512)]},  
    {'hidden_layer_sizes':[(1,1,1), (2,2,2), (4,4,4), (8,8,8), (16,16,16), (32,32,32), (64,64,64), (128,128,128)]},  
    {'hidden_layer_sizes':[(1,1,1,1), (2,2,2,2), (4,4,4,4), (8,8,8,8), (16,16,16,16), (32,32,32,32), (64,64,64,64), (128,128,128,128)]}  
]
```

Resultados | Otim. Hiperparâmetros

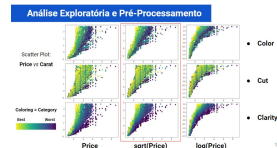
Solver



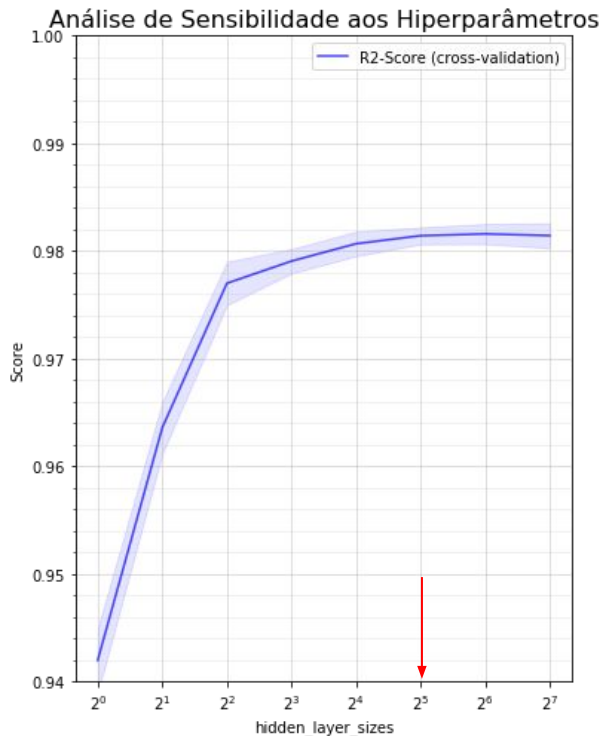
Função de Ativação



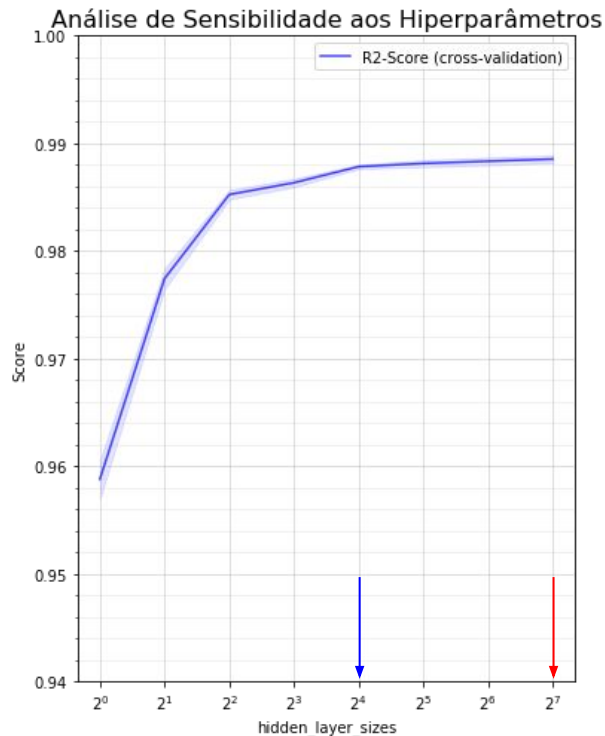
Resultados | Otim. Hiperparâmetros



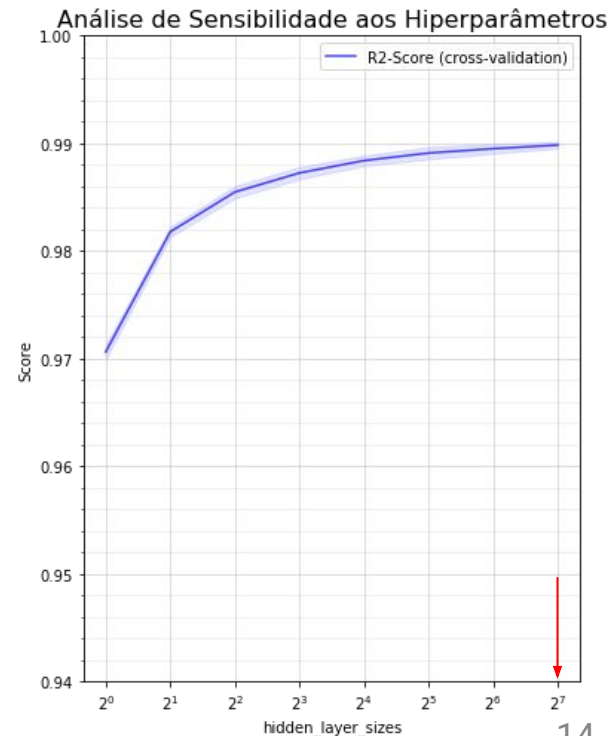
y=Price



y=sqrt(Price)

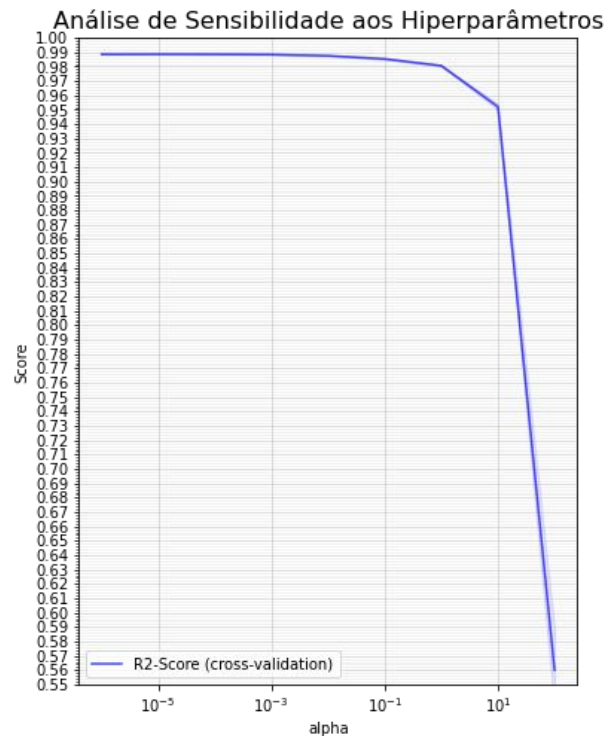
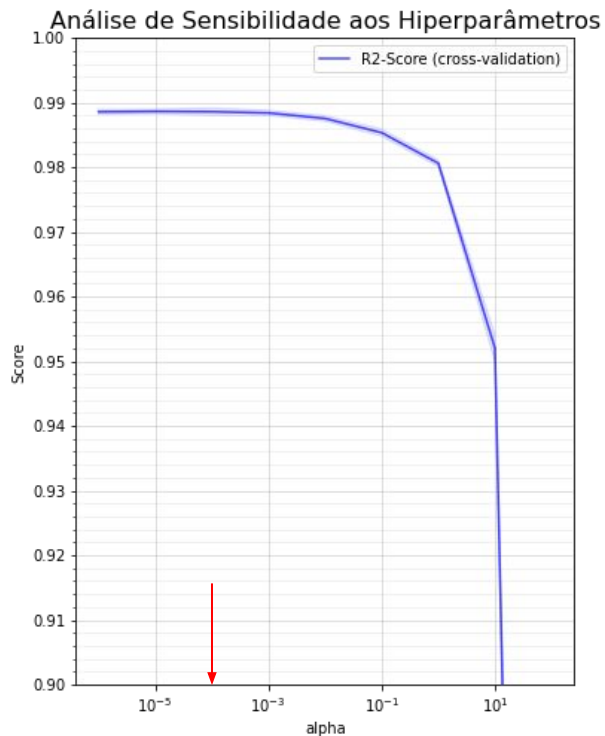


y=log(Price)



Resultados | Otim. Hiperparâmetros

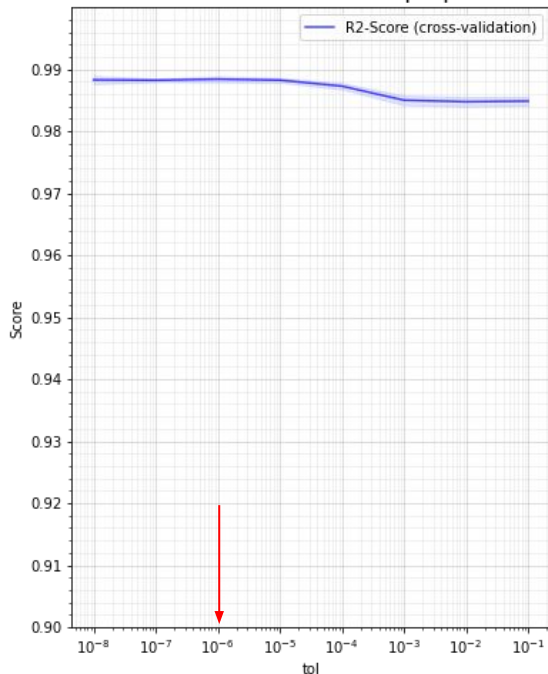
Alpha



Resultados | Otim. Hiperparâmetros

Tol

Análise de Sensibilidade aos Hiperparâmetros



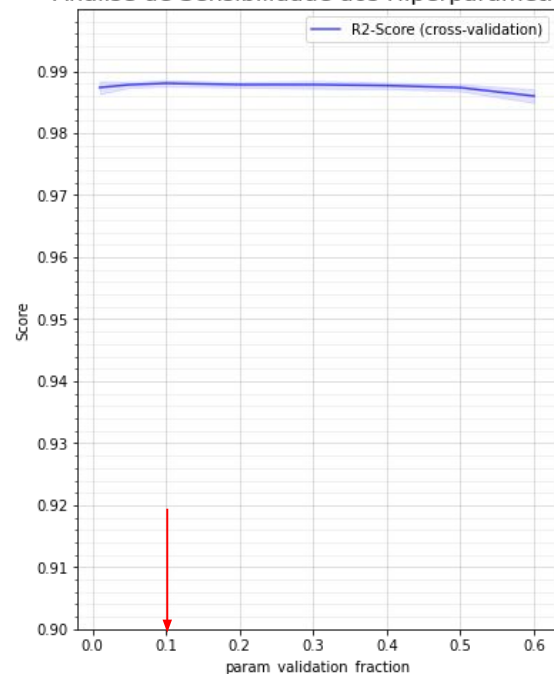
Learning Rate

Análise de Sensibilidade aos Hiperparâmetros



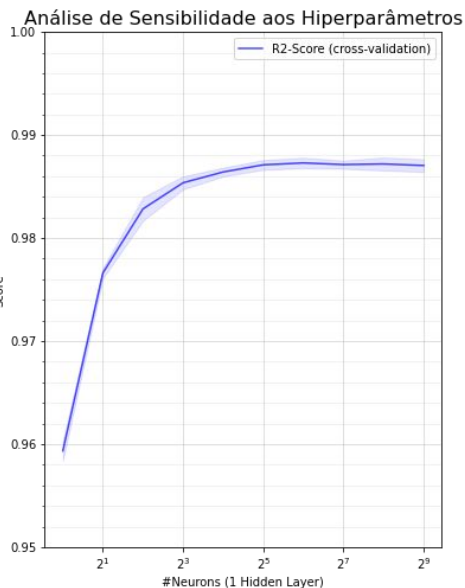
Early Stopping

Análise de Sensibilidade aos Hiperparâmetros

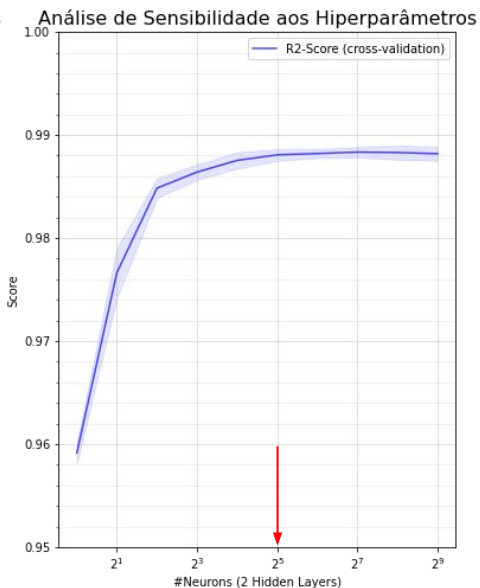


Resultados | Otim. Hiperparâmetros

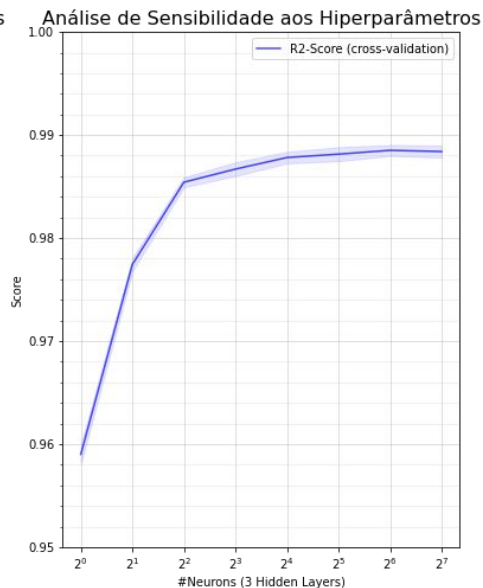
1 Hidden Layer



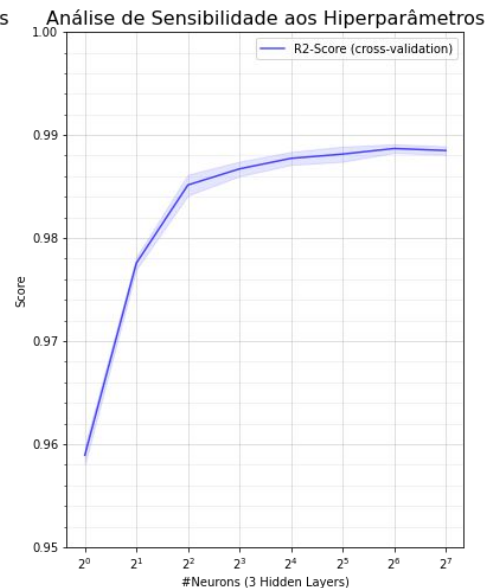
2 Hidden Layer



3 Hidden Layer



4 Hidden Layer



Conclusão

Metodologia:

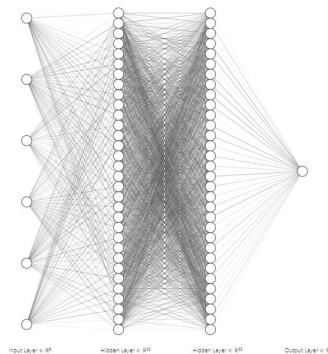
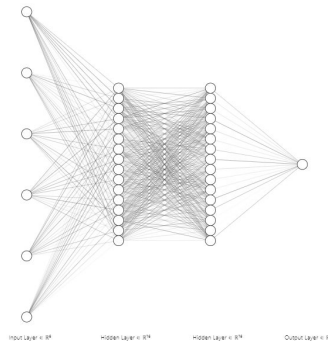
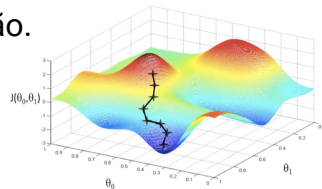
- A análise de sensibilidade da função objetivo à variação dos hiperparâmetros pode ser uma ferramenta poderosa para a correta configuração das redes Multi-Layer Perceptron (MLP).

Características das redes MLP:

- MLP's são sensíveis à normalização e pré-processamento correto dos dados.
- MLP's com camadas ocultas têm função custo não-convexa com múltiplos mínimos locais. Diferentes inicializações dos pesos podem convergir para soluções não ótimas diferentes.
- MLP's têm performance altamente dependente do refinamento de hiperparâmetros.

Trabalhos Futuros:

- Algoritmizar o processo de ajuste de hiperparâmetros para reduzir a subjetividade de inspeção.
- Possíveis caminhos são o Gradiente Descendente e Algoritmos Genéticos.
- Publicar uma nota técnica / artigo.



Referências

Kaggle, ***"Diamonds - Analyze diamonds by their cut, clarity, price, and other attributes"*** (2017)

Disponível em: <https://www.kaggle.com/shivam2503/diamonds>

SciKit-Learn.org, ***"User Guide"*** (2020)

Disponível em: https://scikit-learn.org/stable/user_guide.html

MARMOLEJOS, J. M. P., ***"Implementing Data Mining Methods to Predict Diamond Prices"*** (2018)

ZHANG, C. Y., et al., ***"State-of-the-Art Diamond Price Predictions using Neural Networks"*** (2017)

LECUN, Y., et al., ***"Efficient Backprop"*** (1998)

ESPÍNDOLA, R., ***"EEL817 Redes Neurais - Notas de Aula"*** (2020)



UNIVERSIDADE FEDERAL DO RIO DE JANEIRO



1920 | 2020



Universidade Federal
do Rio de Janeiro

Escola Politécnica

Obrigado!

Ana Thais Castro

anathaiscastro@poli.ufrj.br

Jonas Degrave

jonasdegrave@poli.ufrj.br