

# FAST – Bootcamp Machine Learning

Experimentos e Avaliação  
de Regressores

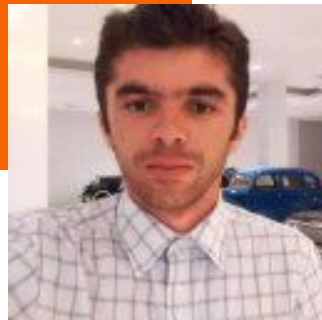
Pedro D. Marrero Fernandez  
pdmf@cesar.school





**Pedro D. Marrero**

Pedro é formado em Ciência da Computação e com Mestrado pela Universidade de Oriente, Cuba, e Doutorado em Ciência da Computação pela Universidade Federal de Pernambuco (UFPE), Brasil. Colabora com diversos laboratórios de pesquisa internacional como: CAMBIA Lab, CALTECH, USA; Unitat de Gràfics i Visió per Ordinador i Intel·ligència Artificial, UIB, Espanha e VIISAR Lab, UFPE, Brasil. Suas principais contribuições e experiências são nas áreas de Computer Vision, Machine Learning, Deep Learning and Computational Photography.



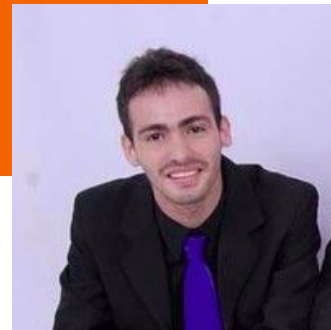
**Antônio J. Pinheiro**

Antônio Janael Pinheiro é engenheiro de software do Centro de Estudos e Sistemas Avançados do Recife (CESAR). Ele recebeu graduação em Redes de Computadores pela Universidade Federal do Ceará em 2013, mestrado e doutorado em Ciência da Computação pela Universidade Federal de Pernambuco, em 2016 e 2020, respectivamente, e especialização em inteligência computacional aplicada pela Universidade Federal Rural de Pernambuco em 2021.



**Blenda Guedes**

Blenda é Cientista de Dados no C.E.S.A.R. e estudante de mestrado em Computação Aplicada na UFRPE. Como amante da Astronomia, sua pesquisa atual combina aprendizado de máquina e modelagem de ondas gravitacionais. Tem experiência, principalmente, com análise de séries temporais.



**Emory R V Freitas**

Emory Raphael Viana Freitas é engenheiro de software do Centro de Estudos e Sistemas Avançados do Recife (CESAR). Ele recebeu graduação em Ciência da Computação pela Universidade Federal do Amazonas em 2011, mestrado em Ciência da Computação pela Universidade Federal de Amazonas, em 2015.












[House Prices - Advanced Regression Techniques](#)

# kaggle



# Kaggle Leaderboard

## House Prices - Advanced Regression Techniques

3621	muggle tan		0.16286	6	4d
3622	John daniela		0.16293	3	22d
3623	ide hiemstra		0.16295	2	1mo
3624	Fast-ML-CESAR		0.16297	1	1s
Your First Entry 					
Welcome to the leaderboard!					
3625	recha_wine		0.16302	1	2mo
3626	Tsvetan Ivanov		0.16305	1	1mo

# Problema



	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	LotConfig	LandSlope	Neighborhood	Condition1	Condition2	BldgType	HouseStyle
0	1	60	RL	65.0	8450	Pave	NaN	Reg	Lvl	AllPub	Inside	Gtl	CollgCr	Norm	Norm	1Fam	2Story
1	2	20	RL	80.0	9600	Pave	NaN	Reg	Lvl	AllPub	FR2	Gtl	Veenker	Feedr	Norm	1Fam	1Story
2	3	60	RL	68.0	11250	Pave	NaN	IR1	Lvl	AllPub	Inside	Gtl	CollgCr	Norm	Norm	1Fam	2Story
3	4	70	RL	60.0	9550	Pave	NaN	IR1	Lvl	AllPub	Corner	Gtl	Crawfor	Norm	Norm	1Fam	2Story
4	5	60	RL	84.0	14260	Pave	NaN	IR1	Lvl	AllPub	FR2	Gtl	NoRidge	Norm	Norm	1Fam	2Story
5	6	50	RL	85.0	14115	Pave	NaN	IR1	Lvl	AllPub	Inside	Gtl	Mitchel	Norm	Norm	1Fam	1.5Fin
6	7	20	RL	75.0	10084	Pave	NaN	Reg	Lvl	AllPub	Inside	Gtl	Somerst	Norm	Norm	1Fam	1Story
7	8	60	RL	NaN	10382	Pave	NaN	IR1	Lvl	AllPub	Corner	Gtl	NWAmes	PosN	Norm	1Fam	2Story
8	9	50	RM	51.0	6120	Pave	NaN	Reg	Lvl	AllPub	Inside	Gtl	OldTown	Artery	Norm	1Fam	1.5Fin



# Objetivo



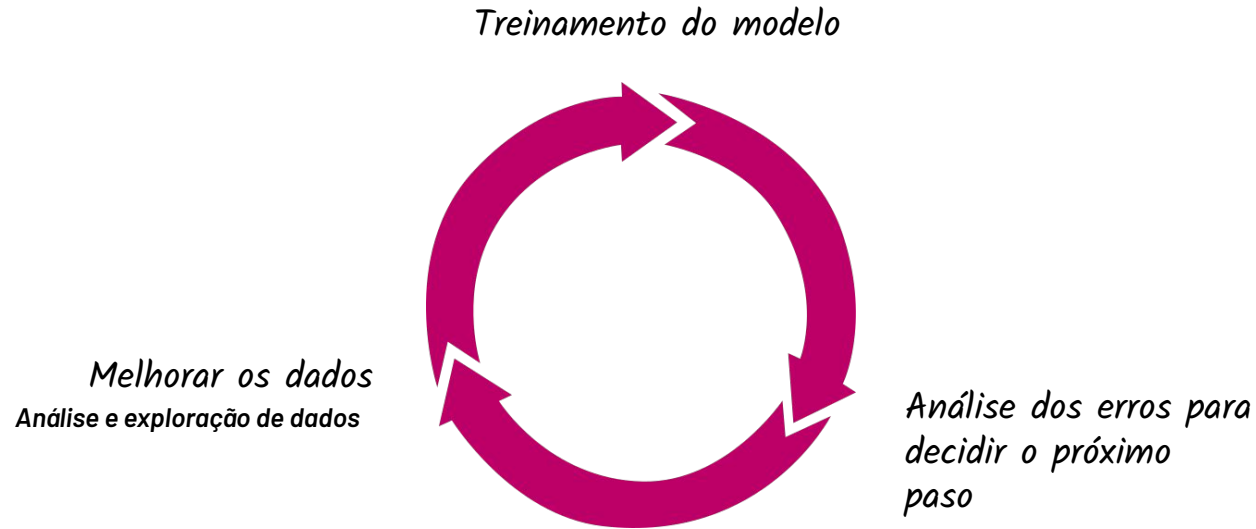
*"AI began with an ancient wish to  
forge the gods."*

- Pamela McCorduck,  
*Machines Who Think*, 1979

**Experimentos e Avaliação de Regressores**  
**Competir e competir**



# Workflow iterativo

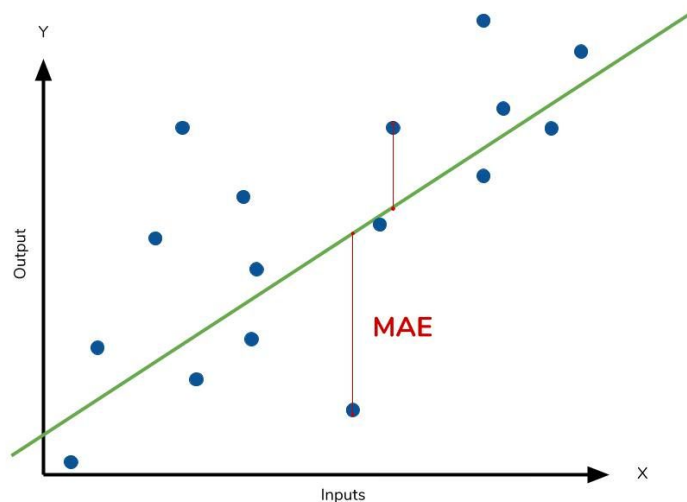


# MAE (Mean Absolute Error) – Erro Médio Absoluto

$$MAE = \frac{1}{n} \sum |y - \hat{y}|$$

Diagram illustrating the MAE formula components:

- $\frac{1}{n}$ : Divide by the total number of data points
- $\sum$ : Sum of
- $y$ : Actual output value
- $\hat{y}$ : Predicted output value
- $|y - \hat{y}|$ : The absolute value of the residual



- Descreve a magnitude típica dos resíduos em relação à medida sendo predita
- **MAE pequeno indica que o modelo é muito bom**
- **MAE grande sugere que seu modelo pode ter problema em certas áreas**
  - **Não** indica *under/overperformance*
- Cada resíduo contribui proporcionalmente
  - Erros grandes contribuirão linearmente





# MAE (Mean Absolute Error) – Erro Médio Absoluto

## `sklearn.metrics.mean_absolute_error`

```
sklearn.metrics.mean_absolute_error(y_true, y_pred, *, sample_weight=None, multioutput='uniform_average')
```

[\[source\]](#)

Mean absolute error regression loss.

Read more in the [User Guide](#).

**Parameters:**    **y\_true** : *array-like of shape (n\_samples,) or (n\_samples, n\_outputs)*

Ground truth (correct) target values.

**y\_pred** : *array-like of shape (n\_samples,) or (n\_samples, n\_outputs)*

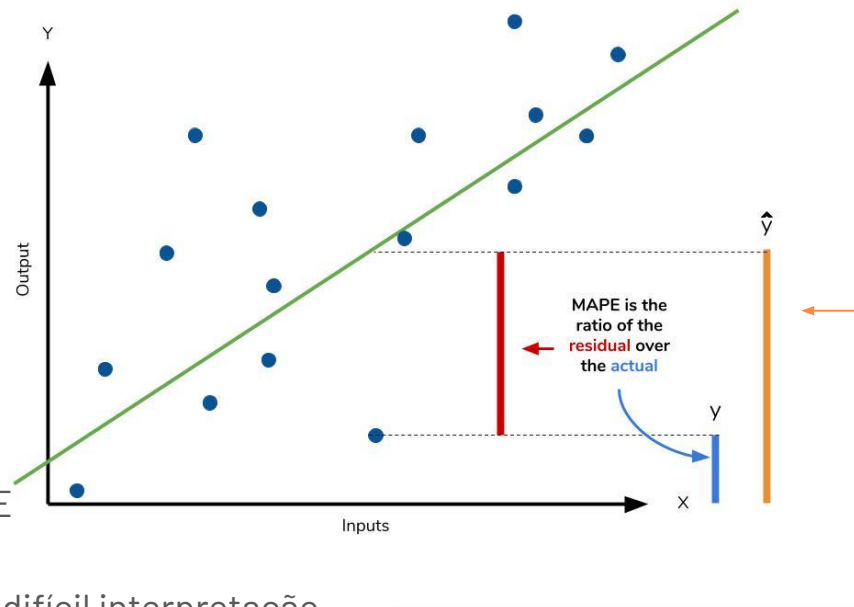
Estimated target values.



# MAPE - Erro Médio Absoluto Percentual

$$MAPE = \frac{100\%}{n} \sum \left| \frac{\overbrace{y - \hat{y}}^{\text{The residual}}}{\underbrace{y}_{\text{Each residual is scaled against the actual value}}} \right|$$

Multiplying by 100% converts to percentage



- Equação representa o percentual equivalente ao MAE
- **Percentuais são de fácil interpretação**
- Individualmente - e em alguns casos -, podem ser de difícil interpretação
- Erros com valor predito maior do que o valor real são mais penalizados



# MAPE - Erro Médio Absoluto Percentual

## `sklearn.metrics.mean_absolute_percentage_error`

```
sklearn.metrics.mean_absolute_percentage_error(y_true, y_pred, sample_weight=None, multioutput='uniform_average')
```

[\[source\]](#)

Mean absolute percentage error regression loss.

Note here that we do not represent the output as a percentage in range  $[0, 100]$ . Instead, we represent it in range  $[0, 1/\text{eps}]$ .

Read more in the [User Guide](#).

*New in version 0.24.*

**Parameters:**    **y\_true** : *array-like of shape (n\_samples,) or (n\_samples, n\_outputs)*

Ground truth (correct) target values.

**y\_pred** : *array-like of shape (n\_samples,) or (n\_samples, n\_outputs)*

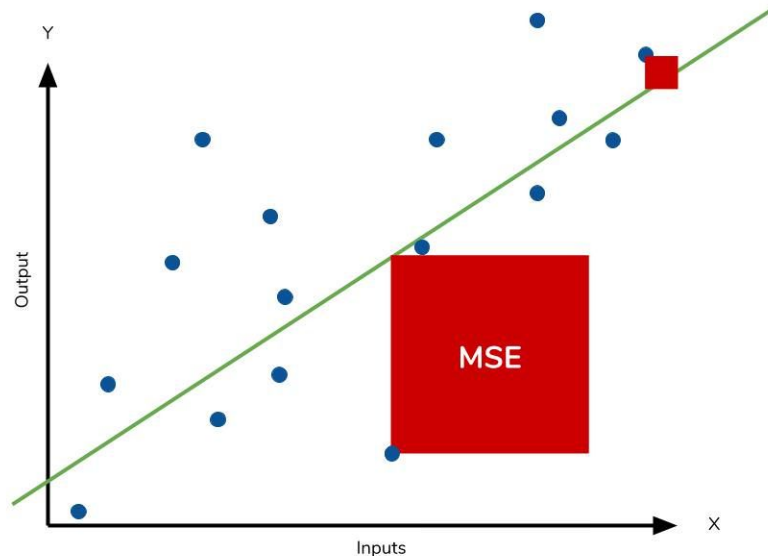
Estimated target values.



# MSE (Mean Square Error) – Erro Médio Quadrático

$$MSE = \frac{1}{n} \sum \underbrace{\left( y - \hat{y} \right)^2}_{\substack{\text{The square of the difference} \\ \text{between actual and} \\ \text{predicted}}}$$

- Similar ao MAE, porém eleva o resíduo ao quadrado
  - *Outliers* irão contribuir muito mais  
= Erros maiores serão mais penalizados
- **Magnitude não é a mesma da saída**
  - Não pode ser comparado ao MAE



# RMSE (Root Mean Square Error)

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{j=1}^n (y_j - \hat{y}_j)^2}$$

- Raiz do Erro Médio Quadrático - Raiz-quadrada do MSE
  - Torna interpretação mais fácil pois **pode ser comparada com a magnitude das saídas**
  - Similarmente, pune mais erros maiores e é afetada por *outliers*
- Representa o desvio-padrão amostral das diferenças entre os valores preditos e os reais
  - Mede quão amplamente os resíduos estão espalhados
- MAE, MSE e RMSE variam teoricamente de 0 ao infinito
  - Todas tem interpretação difícil quando valores são altos



# (R)MSE – (Root) Mean Square Error

## `sklearn.metrics.mean_squared_error`

```
sklearn.metrics.mean_squared_error(y_true, y_pred, *, sample_weight=None, multioutput='uniform_average', squared=True)
```

[\[source\]](#)

Mean squared error regression loss.

Read more in the [User Guide](#).

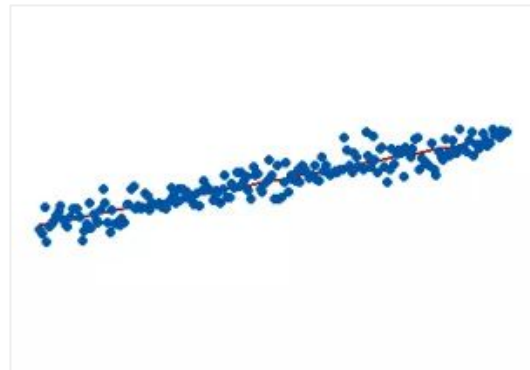
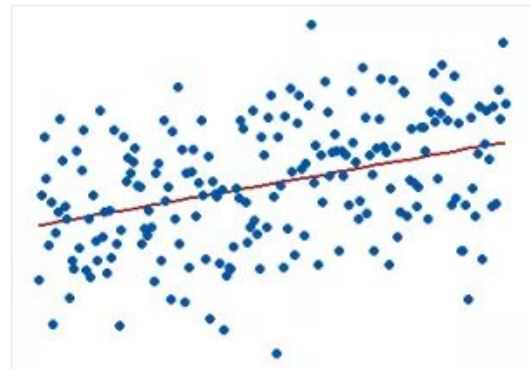
<b>Parameters:</b>	<b><code>y_true</code> :</b> <i>array-like of shape (n_samples,) or (n_samples, n_outputs)</i> Ground truth (correct) target values.
	<b><code>y_pred</code> :</b> <i>array-like of shape (n_samples,) or (n_samples, n_outputs)</i> Estimated target values.



# R<sup>2</sup> (R-Squared) – Coeficiente de Determinação

Expressa quanto da variação dos dados de saída (variável dependente) é explicada pela variável de entrada (independente)

- R<sup>2</sup> é igual ao quadrado de R, a correlação
- Quanto maior o R<sup>2</sup>, melhor o modelo se ajusta à amostra
- Máximo de 1 – Expressa normalmente em percentual
  - Ex.:.  $R^2 = 0,8234$  – significa que 82,34% da variação da variável dependente consegue ser explicada pelos regressores presentes no modelo.





**Hands On!**



# Exemplo: Avaliação de Regressores em Python

```
from sklearn.metrics import mean_squared_error, r2_score
```

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

```
RMSE = mean_squared_error(y_train, y_pred, squared=False)
```

```
R2 = r2_score(y_train, y_pred)
```



# House Prices - Advanced Regression Techniques

Vamos voltar ao exercício anterior e avaliá-lo de forma mais completa.

- Avalie todas as versões implementadas utilizando os erros agora conhecidos e o  $R^2$ 
  - *mean\_squared\_error...* e *r2\_score*

# Para continuar...

- Kevin P. Murphy. **Machine Learning: A Probabilistic Perspective (Adaptive Computation and Machine Learning series)**. 1a Edição: The MIT Press, 2012.
  - Capítulo 7 - Linear regression - 7.1 a 7.3
- Joel Grus. **Data Science do Zero: Primeiras regras com o Python**. 1ª Edição: Alta Books, 2016.
  - Capítulo 14 - Regressão Linear Simples



C . E . S . A . R

Pessoas impulsionando inovação.  
Inovação impulsionando negócios.

**NOSSO CONTATO**

[cesar.org.br](http://cesar.org.br)

[cesar.school](http://cesar.school)





C . E . S . A . R

school