

Uma Brevíssima Introdução à Aprendizagem de Máquina

Bedrooms	Sq. feet	Neighborhood	Sale price
3	2000	Normaltown	\$250,000
2	800	Hipsterton	\$300,000
2	850	Normaltown	\$150,000
1	550	Normaltown	\$78,000
4	2000	Skid Row	\$150,000

Bedrooms	Sq. feet	Neighborhood	Sale price
3	2000	Hipsterton	???

<https://medium.com/@ageitgey/machine-learning-is-fun-80ea3ec3c471>

Solução do programador...

```
def estimate_house_sales_price(num_of_bedrooms, sqft, neighborhood):  
    price = 0  
  
    # In my area, the average house costs $200 per sqft  
    price_per_sqft = 200  
  
    if neighborhood == "hipsterton":  
        # but some areas cost a bit more  
        price_per_sqft = 400  
  
    elif neighborhood == "skid row":  
        # and some areas cost less  
        price_per_sqft = 100  
  
    # start with a base price estimate based on how big the place is  
    price = price_per_sqft * sqft  
  
    # now adjust our estimate based on the number of bedrooms  
    if num_of_bedrooms == 0:  
        # Studio apartments are cheap  
        price = price - 20000  
    else:  
        # places with more bedrooms are usually  
        # more valuable  
        price = price + (num_of_bedrooms * 1000)  
  
    return price
```

Solução mais inteligente...

```
def estimate_house_sales_price(num_of_bedrooms, sqft, neighborhood):  
    price = 0  
  
    # a little pinch of this  
    price += num_of_bedrooms * .841231951398213  
  
    # and a big pinch of that  
    price += sqft * 1231.1231231  
  
    # maybe a handful of this  
    price += neighborhood * 2.3242341421  
  
    # and finally, just a little extra salt for good measure  
    price += 201.23432095  
  
    return price
```

Solução de Aprendizagem de Máquina

"Computador, é o seguinte: eu sei que tem uma relação linear entre o preço do imóvel e essas variáveis! Me traz a melhor relação possível com esses dados, te vira!"

Aprendizagem de Máquina

**Não diz pro computador
como fazer, mas sim o que
fazer!**

Aprendizagem de Máquina

Modelo é uma relação matemática entre variáveis

Aprendizagem de Máquina é construir modelos com base nos dados! (J. Grus)

Aprendizagem de Máquina

Supervisionada: dados já possuem a variável a ser prevista

Ex. Regressão Linear, Árvore de Decisão

Não supervisionada: dados não possuem variável a ser prevista, padrão tem que ser descoberto

Ex. Clusterização

Aprendizagem Supervisionada

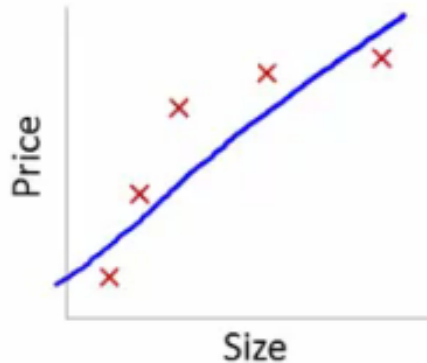
Modelos de Regressão: variável a ser prevista é numérica

Ex. Regressão Linear

Modelos de Classificação: variável a ser prevista é categórica

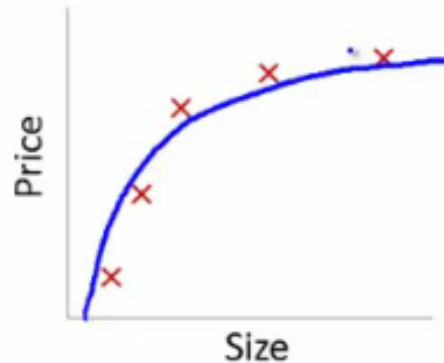
Ex. Árvore de Decisão

Sobreaajuste e Sub-Ajuste



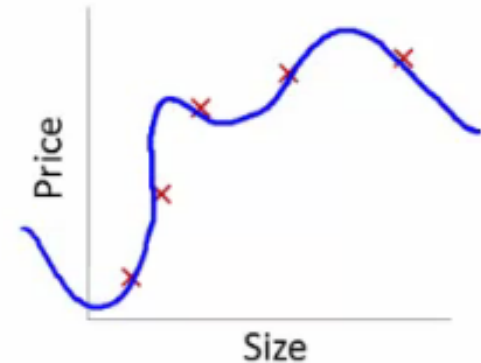
$$\theta_0 + \theta_1 x$$

High bias
(underfit)



$$\theta_0 + \theta_1 x + \theta_2 x^2$$

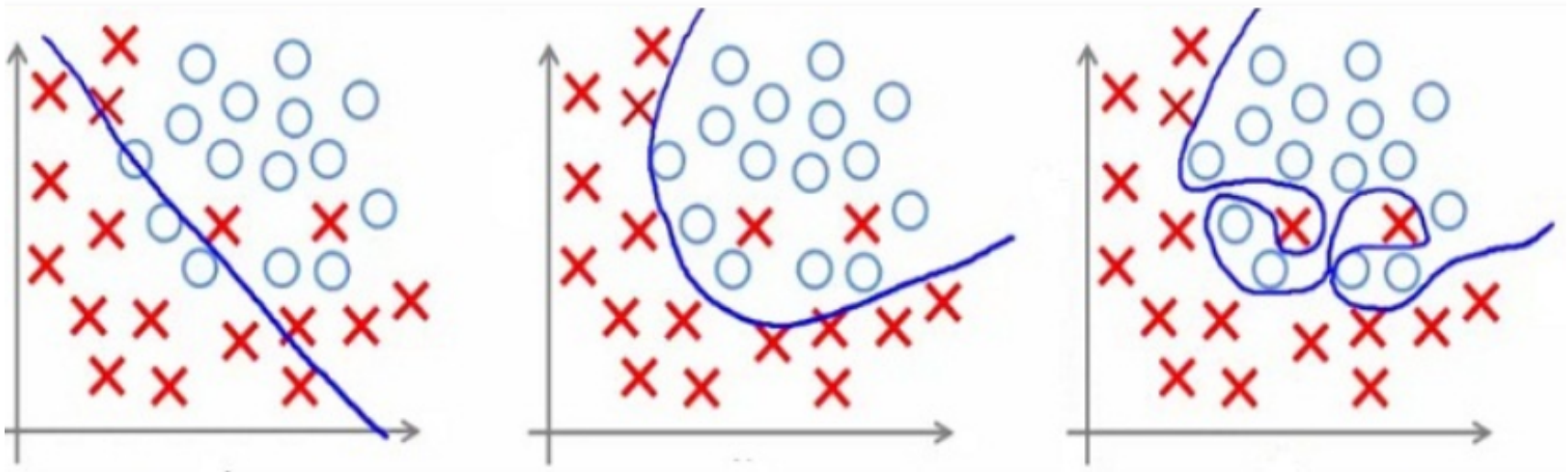
"Just right"



$$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$$

High variance
(overfit)

Sobreajuste e Sub-Ajuste



Under-fitting

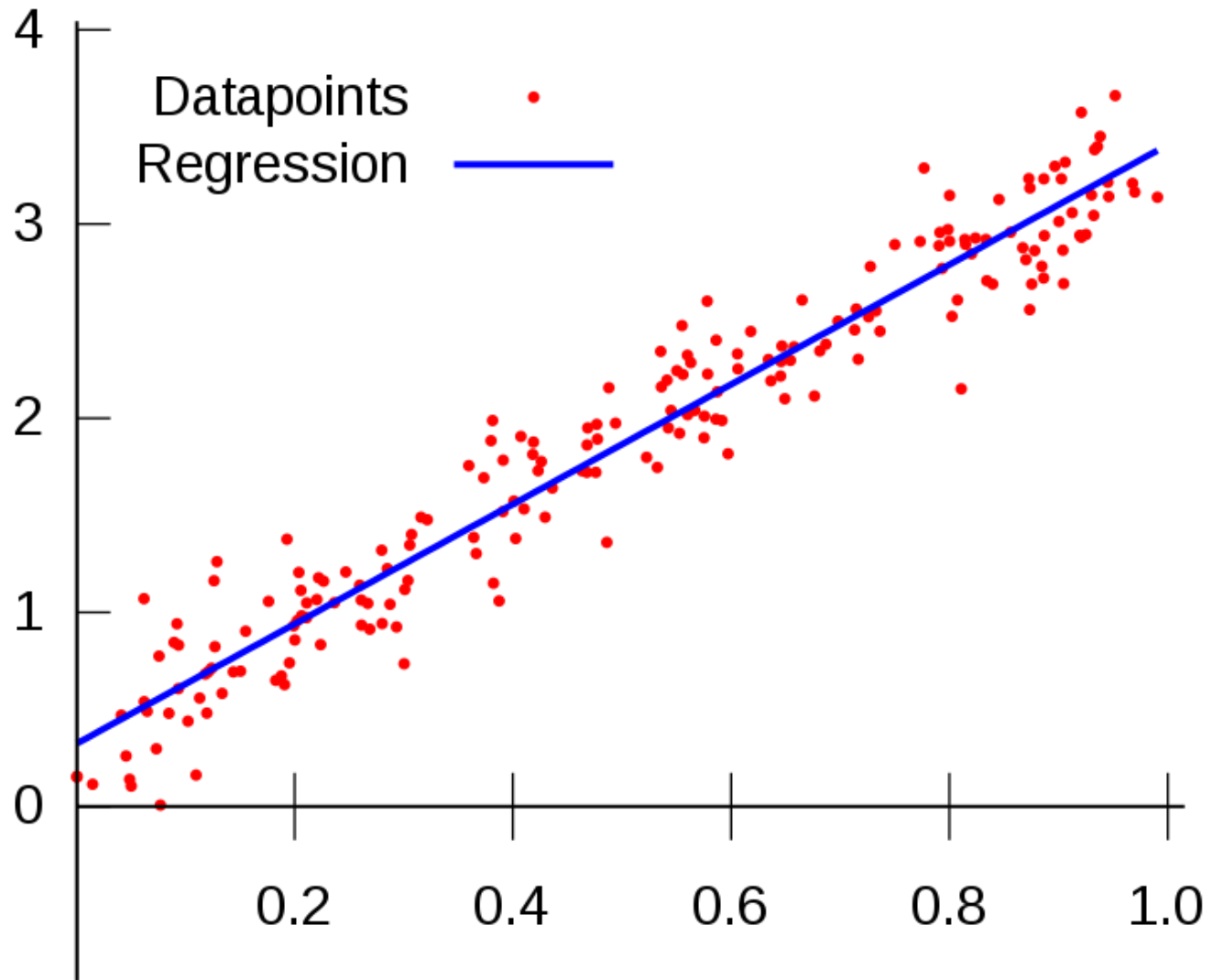
(too simple to
explain the
variance)

Appropriate-fitting

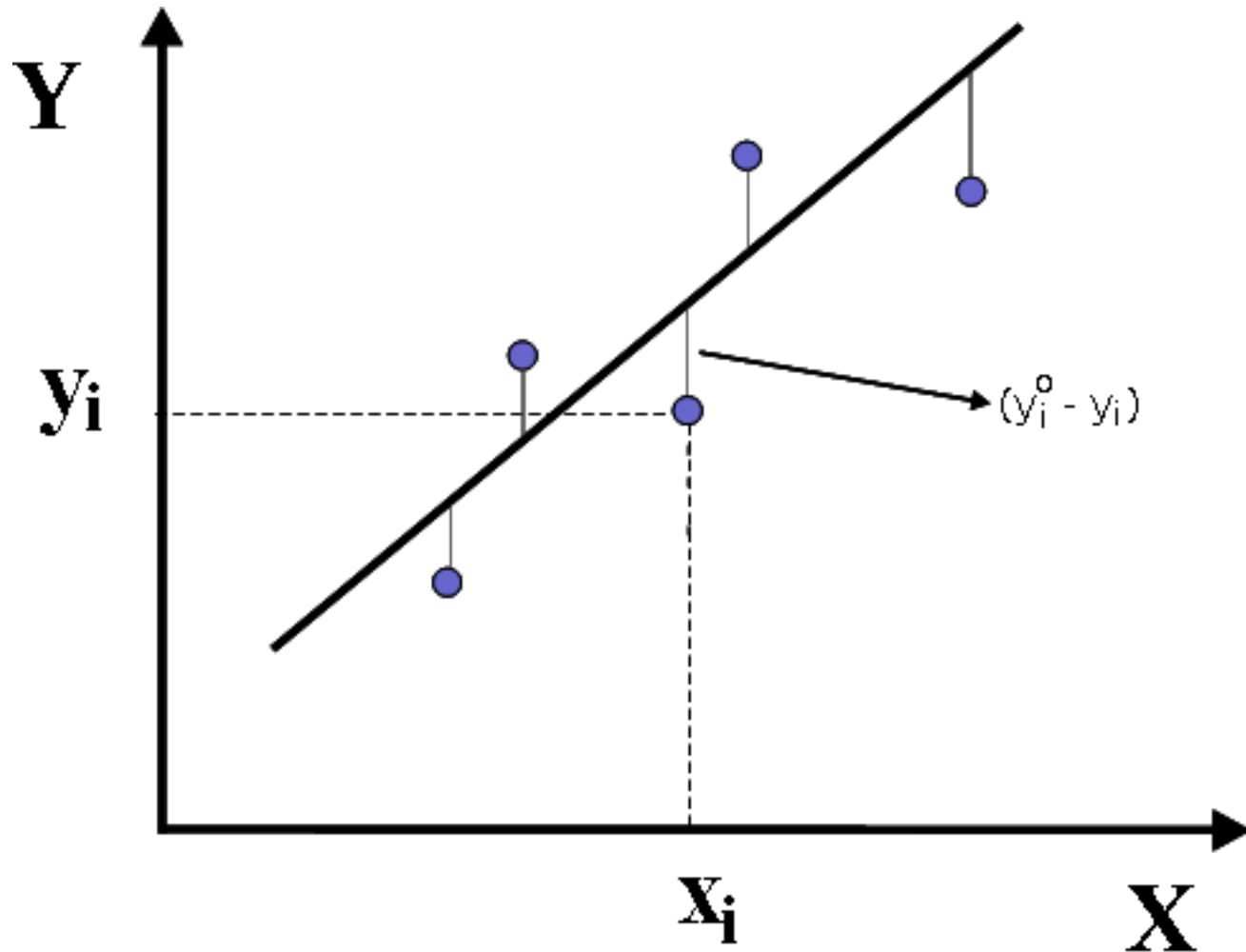
Over-fitting

(forcefitting -- too
good to be true)

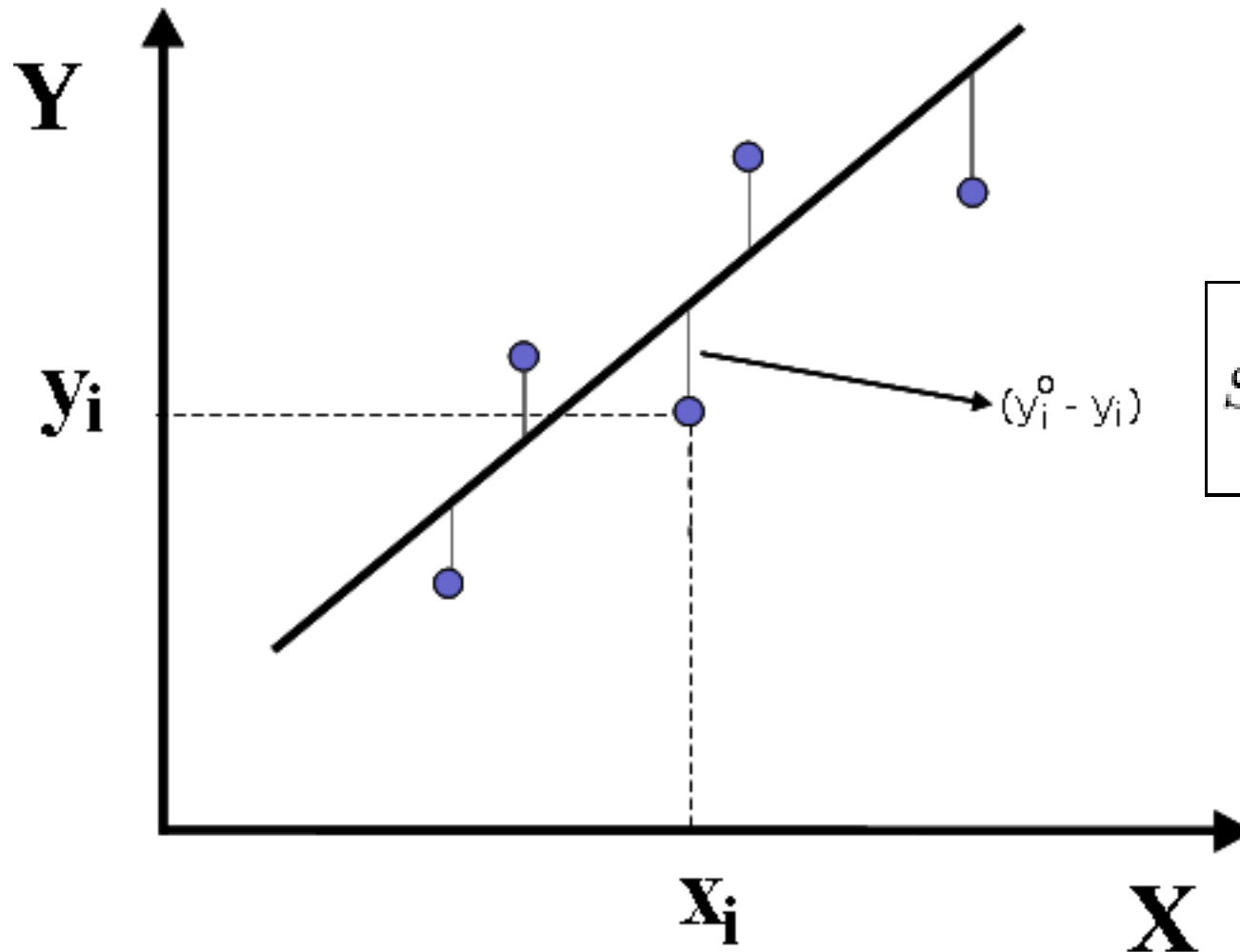
Regressão Linear



Como funciona o ajuste

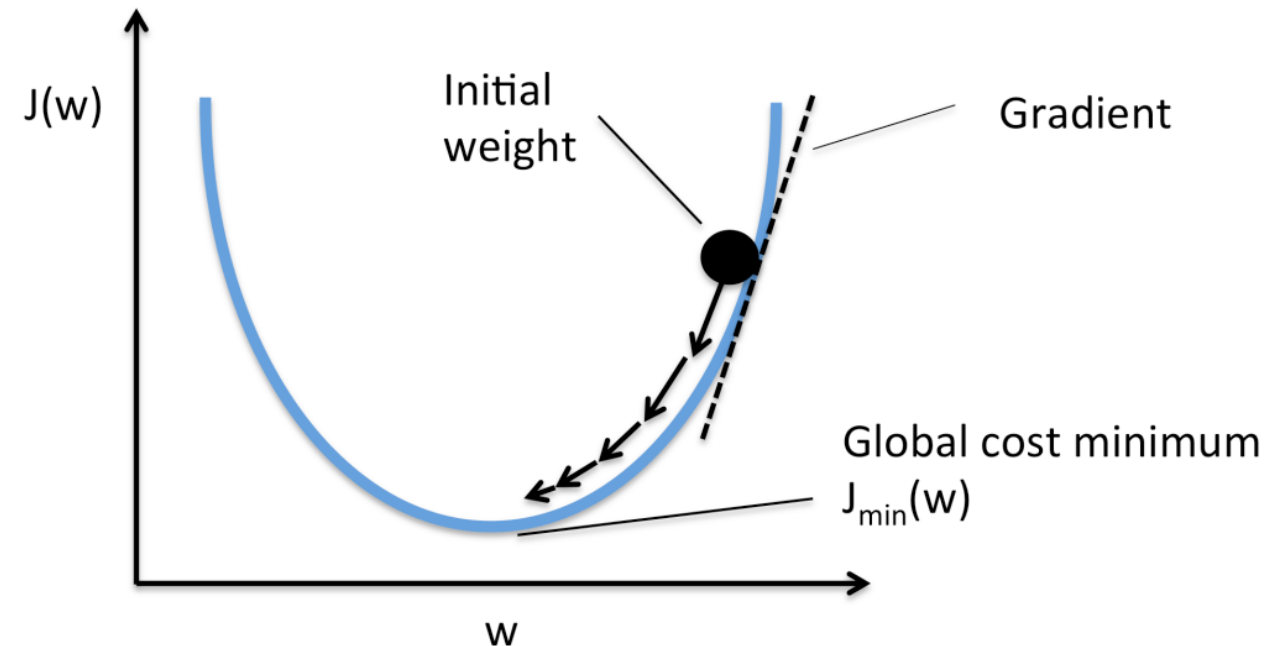


Como funciona o ajuste



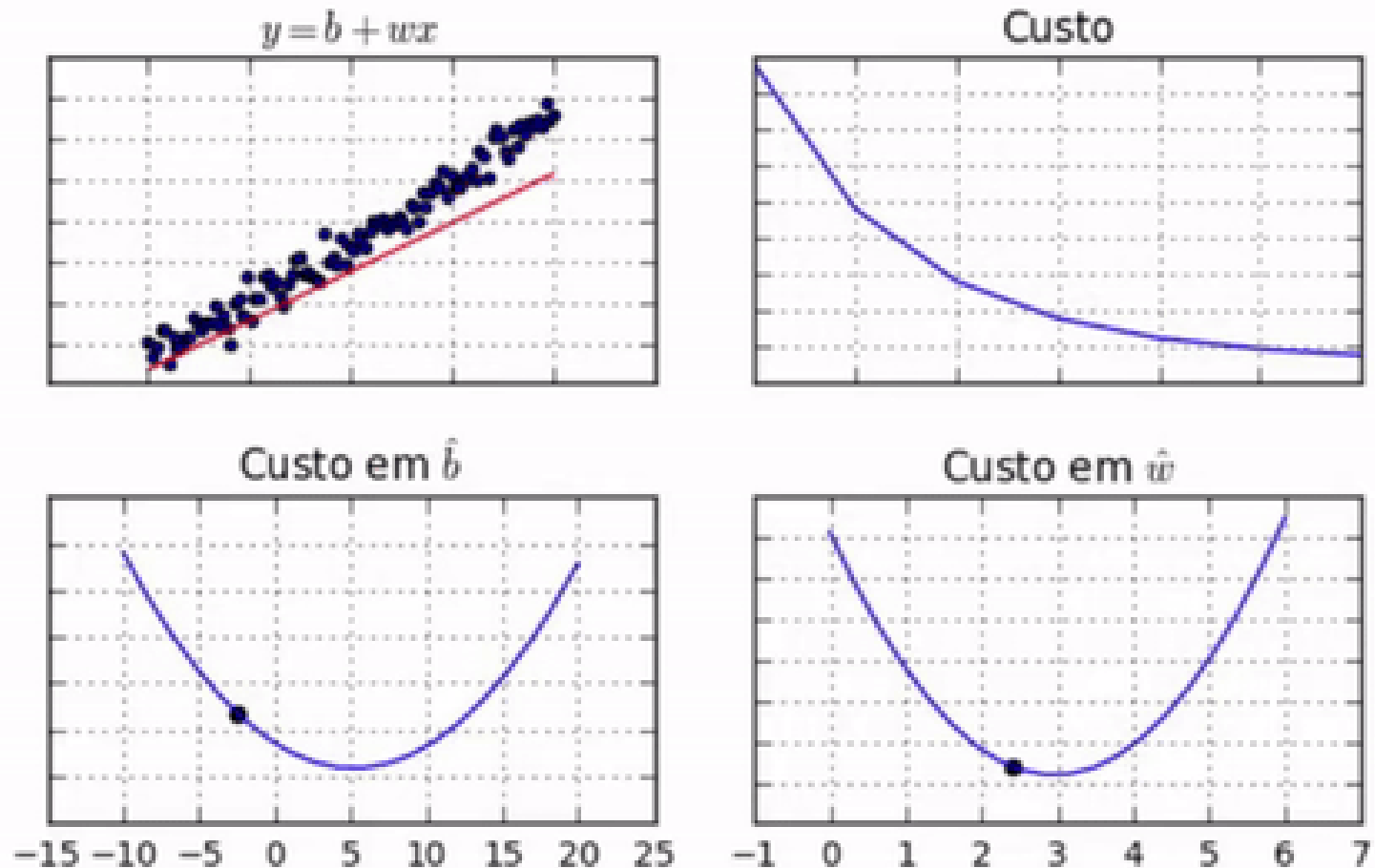
$$S \equiv \sum_{i=1}^N (y_i^o - y_i)^2$$

Como funciona o ajuste

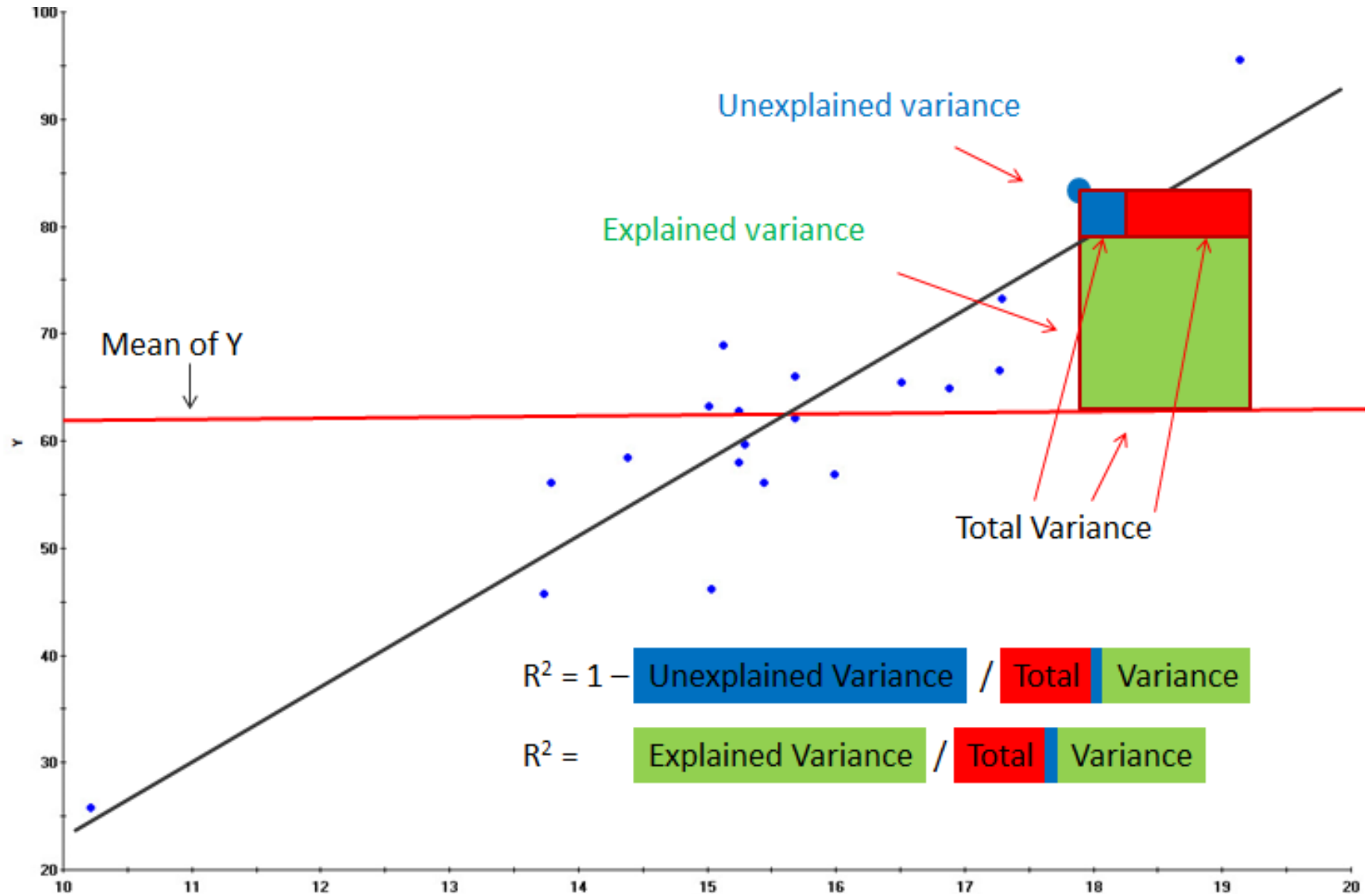


$$S \equiv \sum_{i=1}^N (y_i^o - y_i)^2$$

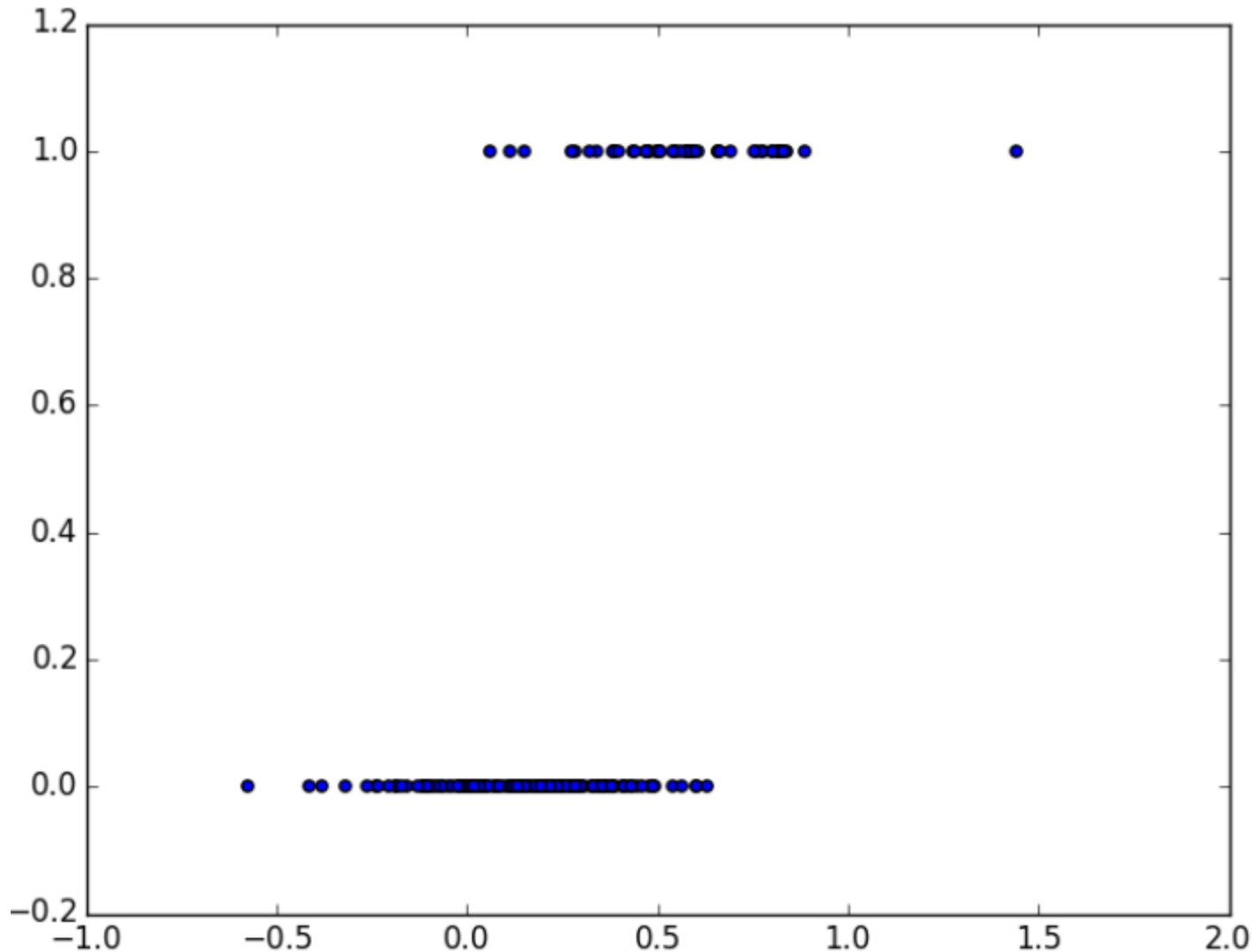
Como funciona o ajuste



Métrica R^2

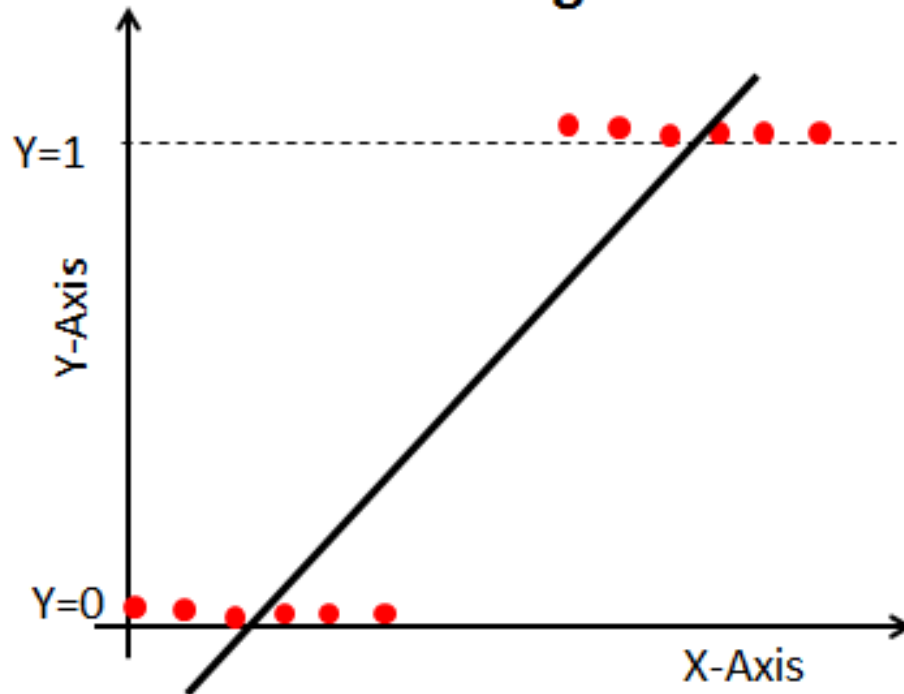


Regressão Logística

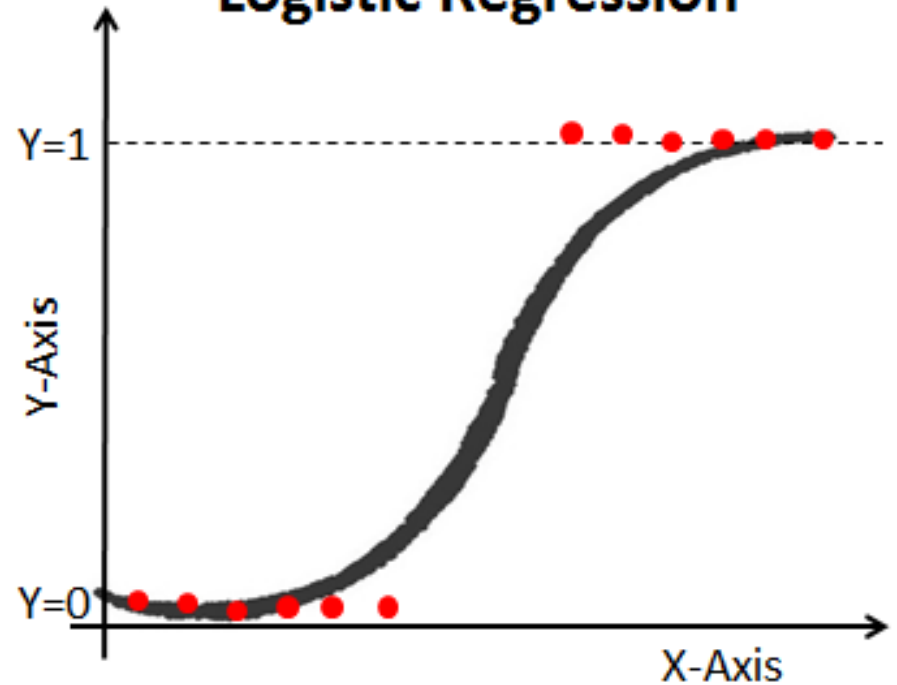


Regressão Logística

Linear Regression



Logistic Regression



Regressão Logística

```
def logistic(x):  
    return 1.0 / (1 + math.exp(-x))
```

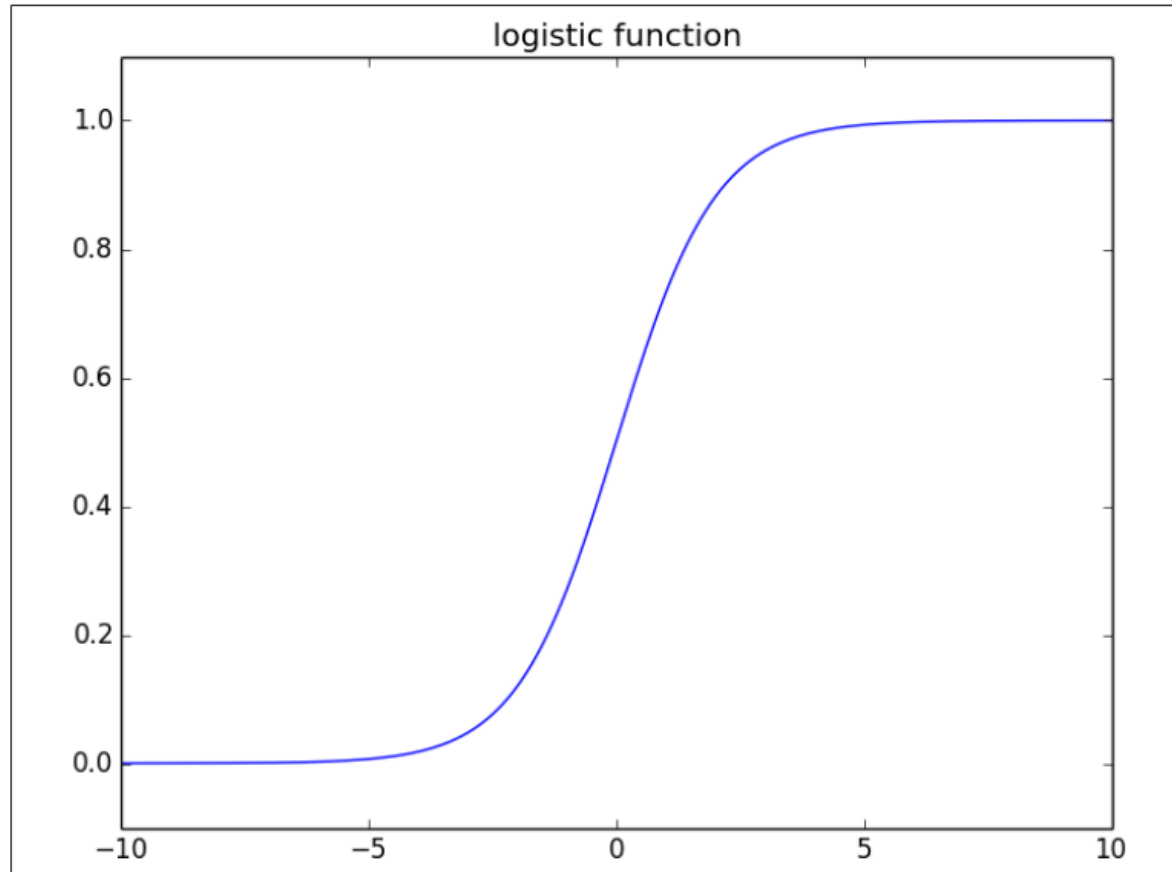


Figure 16-3. The logistic function

Métricas de Modelos de Classificação (Regressão Logística, Árvore de Decisão)

- Acurácia
- Precisão
- Abrangência
- F1
- e muitas outras...

Matriz de Confusão

Classe Predit

Positivo

Negativo

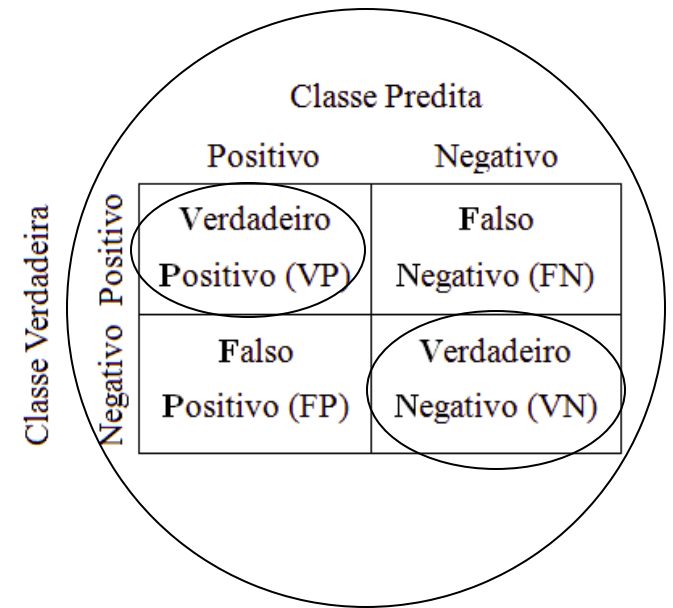
Classe Verdadeira

Positivo
Negativo

Verdadeiro Positivo (VP)	Falso Negativo (FN)
Falso Positivo (FP)	Verdadeiro Negativo (VN)

Acurácia

$$\text{Acurácia} = \frac{\text{Verdadeiros Positivos (TP)} + \text{Verdadeiros Negativos (VN)}}{\text{Total}}$$

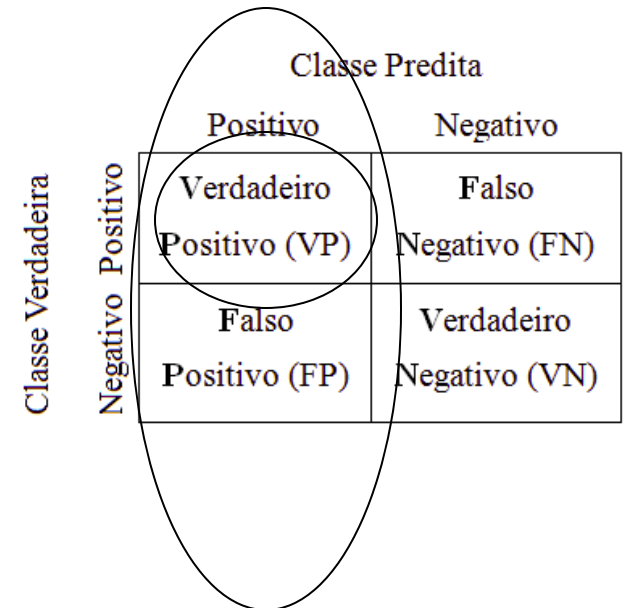


A confusion matrix diagram for Accuracy. It features a large circle enclosing a 2x2 table. The table's columns are labeled 'Positivo' and 'Negativo' under the heading 'Classe Predita'. The rows are labeled 'Positivo' and 'Negativo' under the heading 'Classe Verdadeira'. The cells contain: Top-Left: 'Verdadeiro Positivo (VP)', Top-Right: 'Falso Negativo (FN)', Bottom-Left: 'Falso Positivo (FP)', Bottom-Right: 'Verdadeiro Negativo (VN)'. The top-left and bottom-right cells are circled, representing the correct classifications (TP and TN) used in the accuracy formula.

	Classe Predita	
	Positivo	Negativo
Positivo	Verdadeiro Positivo (VP)	Falso Negativo (FN)
Negativo	Falso Positivo (FP)	Verdadeiro Negativo (VN)

Precisão

$$\text{Precisão} = \frac{\text{Verdadeiros Positivos (TP)}}{\text{Verdadeiros Positivos (TP)} + \text{Falsos Positivos (FP)}}$$



A confusion matrix diagram for Precision. It features a 2x2 table with the same structure as the one above. The columns are 'Positivo' and 'Negativo' under 'Classe Predita', and the rows are 'Positivo' and 'Negativo' under 'Classe Verdadeira'. The cells contain: Top-Left: 'Verdadeiro Positivo (VP)', Top-Right: 'Falso Negativo (FN)', Bottom-Left: 'Falso Positivo (FP)', Bottom-Right: 'Verdadeiro Negativo (VN)'. The top-left cell is circled, representing the True Positives (TP). The bottom-left cell (FP) is also circled, and a large oval encompasses both the TP and FP cells, representing the total number of predicted positives (TP + FP) used in the precision formula.

	Classe Predita	
	Positivo	Negativo
Positivo	Verdadeiro Positivo (VP)	Falso Negativo (FN)
Negativo	Falso Positivo (FP)	Verdadeiro Negativo (VN)

Recall (abrangência)

$$\text{Recall} = \frac{\text{Verdadeiros Positivos (TP)}}{\text{Verdadeiros Positivos (TP)} + \text{Falsos Negativos (FN)}}$$

		Classe Predita	
		Positivo	Negativo
Classe Verdadeira	Positivo	Verdadeiro Positivo (VP)	Falso Negativo (FN)
	Negativo	Falso Positivo (FP)	Verdadeiro Negativo (VN)

F1: Combina precisão e recall

$$F1 = \frac{2 * \text{precisão} * \text{recall}}{\text{precisão} + \text{recall}}$$