

A close-up photograph of a red dart with a gold-colored barrel hitting the bullseye of a dartboard. The dartboard has a green and white pattern. The background is a soft, out-of-focus yellow and green. The image is partially obscured by a white geometric shape on the right side of the slide.

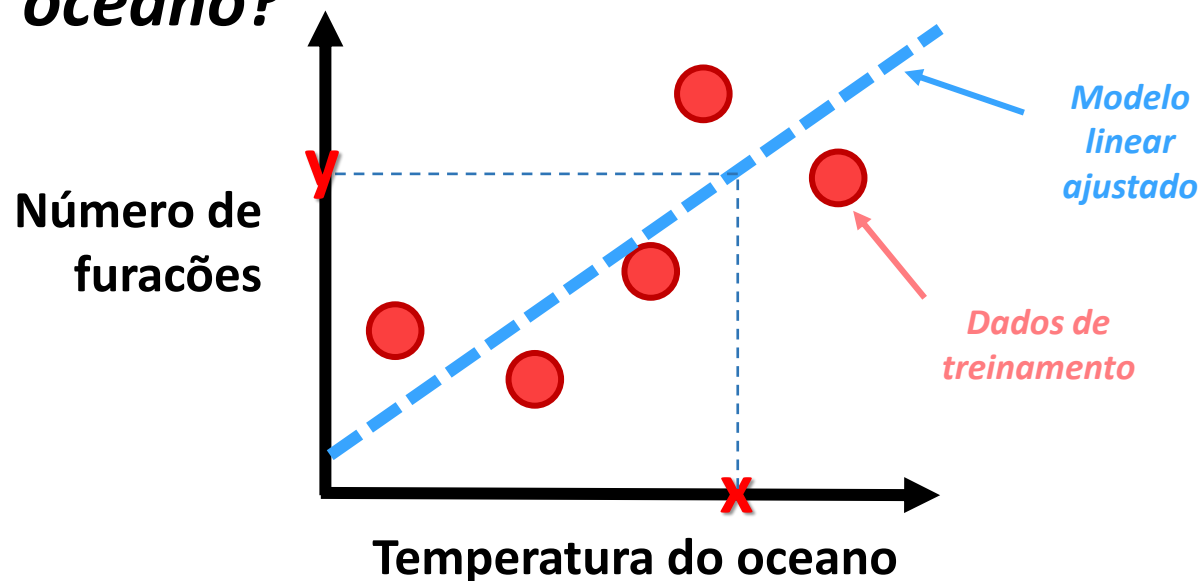
Introdução ao aprendizado de máquina:

Validação de Modelos

Thales A. P. West, Ph.D.

Validação de modelos

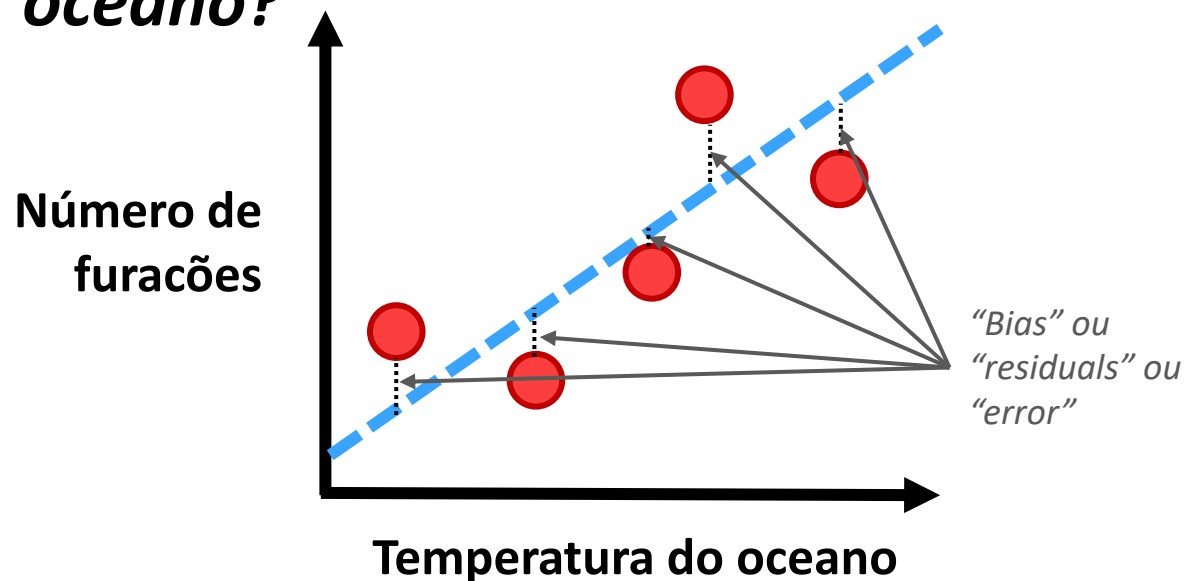
Podemos prever furacões com base na temperatura do oceano?



*Podemos usar a **linha ajustada** para prever o número de furacões (**y**) com base na temperatura do oceano (**x**)*

Validação de modelos

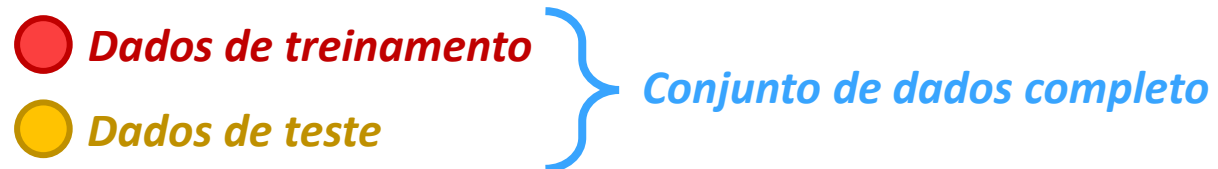
Podemos prever furacões com base na temperatura do oceano?



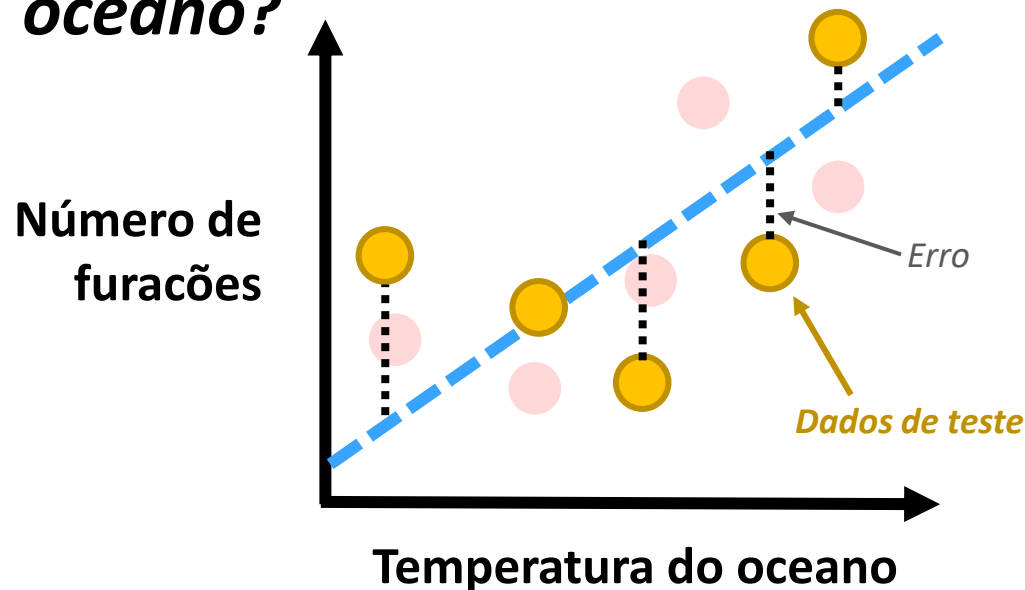
Na análise de regressão, “resíduos” é o termo mais popular...

No aprendizado de máquina, “bias” ou simplesmente “error” são os termos mais populares

Validação de modelos



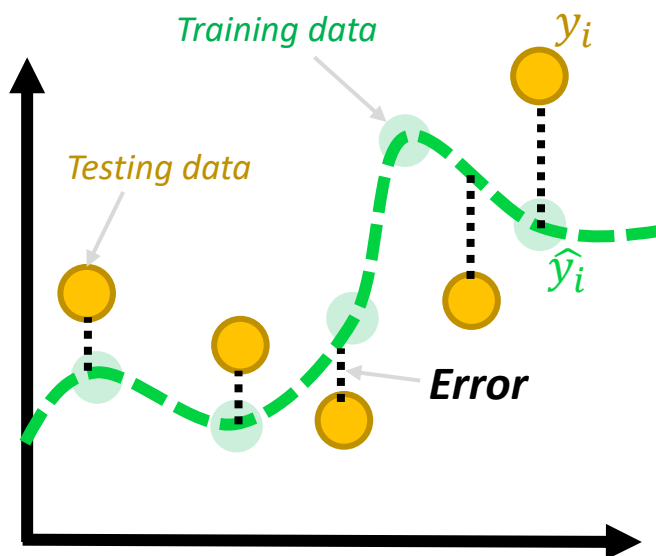
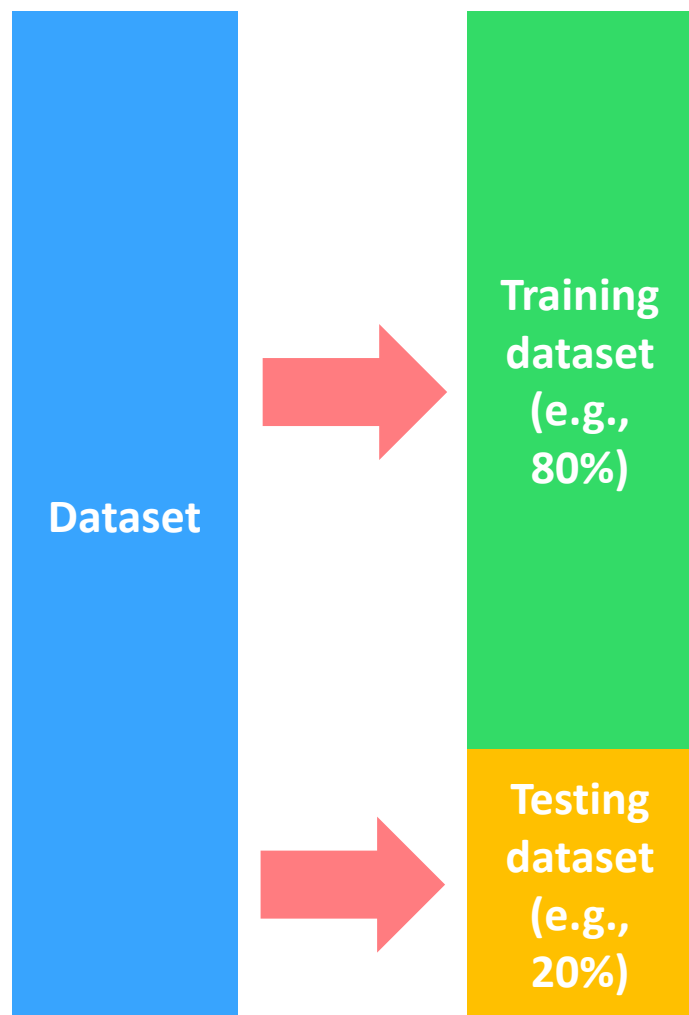
Podemos prever furacões com base na temperatura do oceano?



A **validação** envolve um conjunto de técnicas usada para avaliar o desempenho de um modelo

Envolve o particionamento dos dados em **dados de treinamento** e **dados de teste**

Método: Holdout Validation



Soma dos Resíduos Quadrados (SSR)

$$SSR = \sum (y_i - \hat{y}_i)^2$$

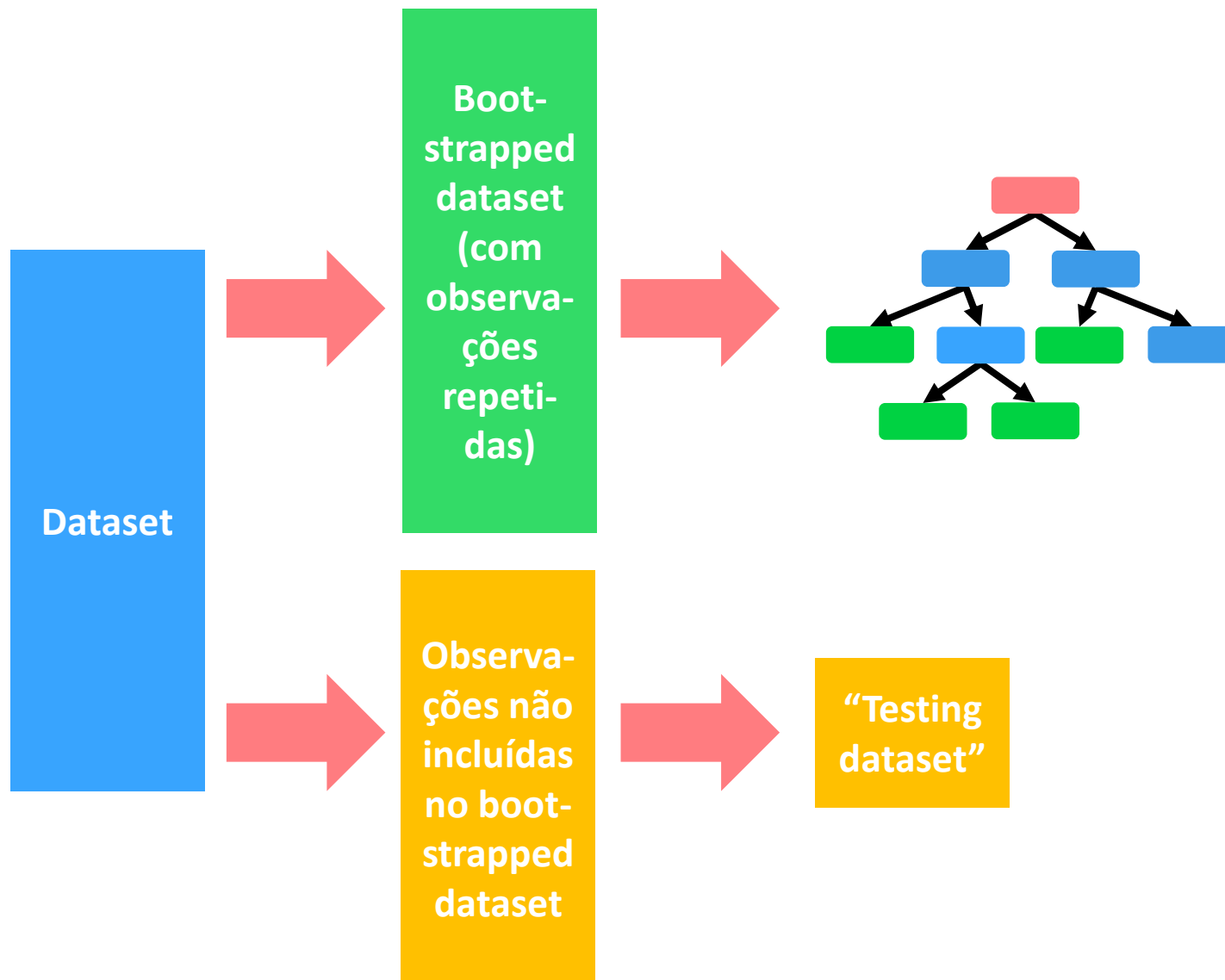
Erro Quadrático Médio (MSE)

$$MSE = \frac{\sum (y_i - \hat{y}_i)^2}{n}$$

Raiz do Erro Quadrático Médio (RMSE)

$$RMSE = \sqrt{\frac{\sum (y_i - \hat{y}_i)^2}{n}}$$

Método: Out-of-Bag (OOB) evaluation

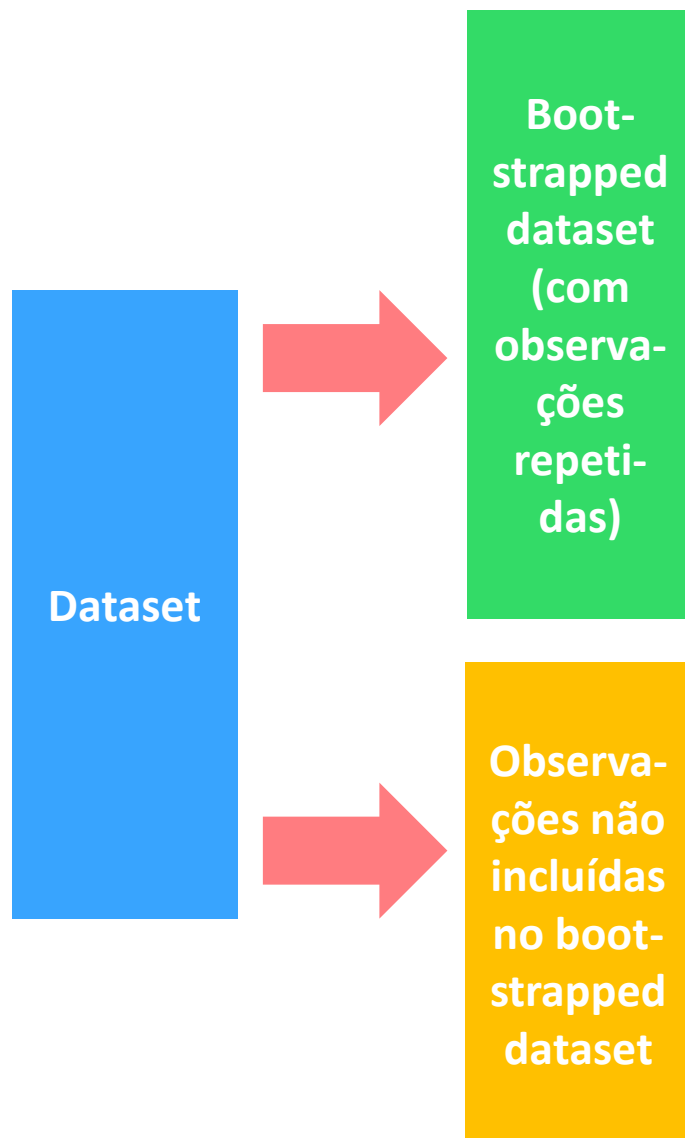


No contexto de Random Forests, cada árvore é treinada em uma amostra bootstrap dos dados (amostrada aleatoriamente com substituição do conjunto de dados original)

Como cada árvore é treinada em um subconjunto dos dados, haverá observações que não incluídas ("out of bag" ou OOB) na amostra bootstrap de cada árvore. As amostras deixadas de fora são então usadas como um conjunto de validação para estimar o desempenho da árvore de decisão. O erro OOB é o erro calculado baseado nessas observações

A estimativa final de erro OOB para o modelo de floresta aleatória é normalmente obtida pela média dos erros OOB de todas as árvores individuais

Método: Out-of-Bag (OOB) evaluation



Para qualquer observação, a probabilidade dela *não ser escolhida* em um único sorteio é:

$$P(\text{não escolhida}) = 1 - \frac{1}{n}$$

A probabilidade de essa observação *nunca ser escolhida* em **n sorteios** é:

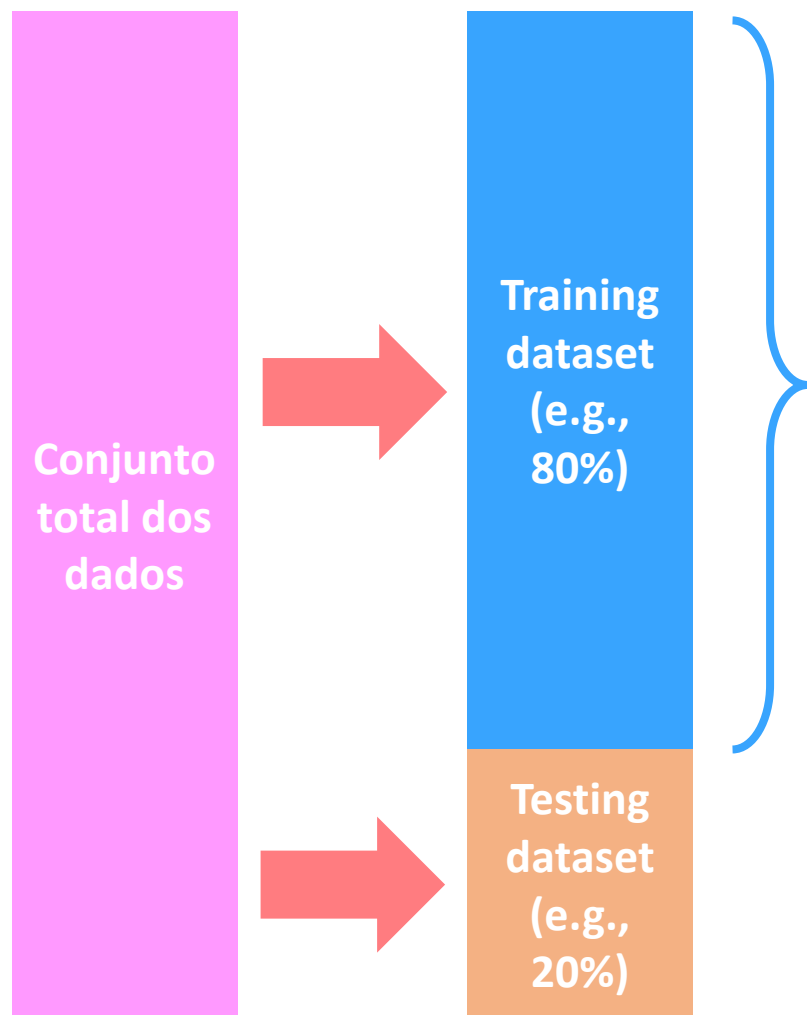
$$P(\text{nunca escolhida}) = \left(1 - \frac{1}{n}\right)^n$$

Resultado assintótico (quando $n \rightarrow \infty$):

$$\lim_{n \rightarrow \infty} \left(1 - \frac{1}{n}\right)^n = \frac{1}{e} \approx 0.368$$

Ou seja, em média, aproximadamente **36,8%** das observações **não são incluídas** em uma amostra bootstrap e podem ser usadas para calcular o erro **OOB**

Validação Cruzada (Cross-Validation)



Cross-validation é uma técnica usada para avaliar o desempenho e a generalização de um modelo

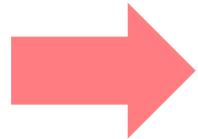
Envolve particionar os dados de treinamento em vários subconjuntos (ou “folds”) e treinar iterativamente o modelo em um subconjunto enquanto o valida nos restantes

Esse processo ajuda a avaliar o desempenho do modelo em dados “não vistos”, ajustar calibração de parametros e reduz o risco de sobreajuste (“overfitting”)

Método: Leave-One-Out Cross-Validation (LOOCV)

*Dados de
treinamento*

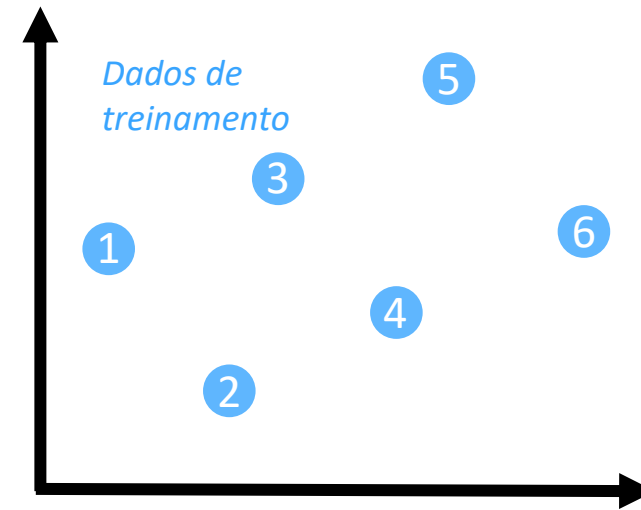
1
2
3
4
5
6
7
8
9
10
11
12



1
2
3
4
5
6
7
8
9
10
11
12

Dado de validação #1

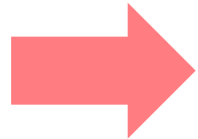
*Subconjunto de
treinamento #1*



Método: Leave-One-Out Cross-Validation (LOOCV)

*Dados de
treinamento*

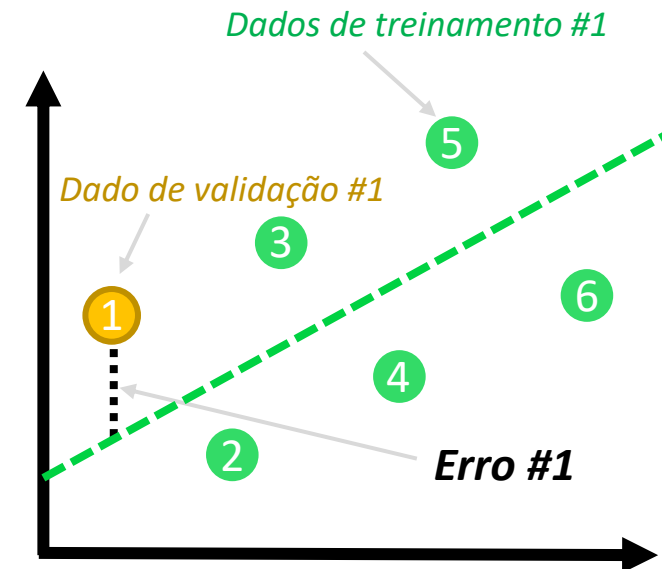
1
2
3
4
5
6
7
8
9
10
11
12



1
2
3
4
5
6
7
8
9
10
11
12

Dado de validação #1

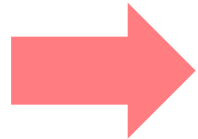
*Subconjunto de
treinamento #1*



Método: Leave-One-Out Cross-Validation (LOOCV)

*Dados de
treinamento*

1
2
3
4
5
6
7
8
9
10
11
12

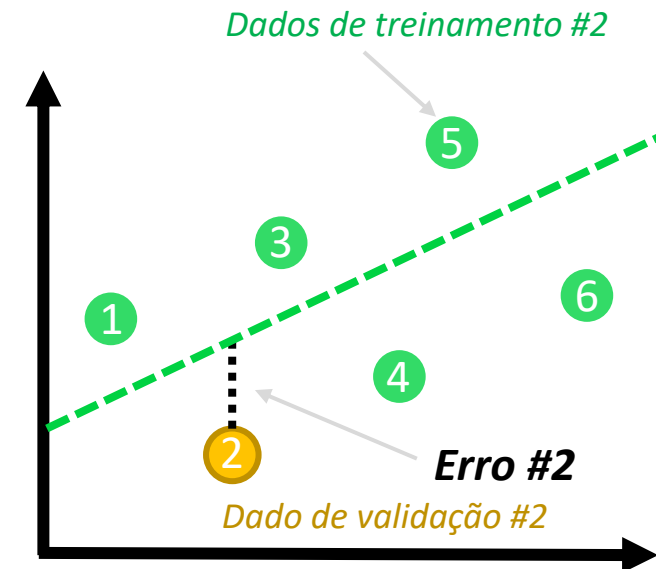


1
2
3
4
5
6
7
8
9
10
11
12

Subconjunto de treinamento #2

Dado de validação #2

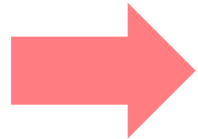
*Subconjunto de
treinamento #2*



Método: Leave-One-Out Cross-Validation (LOOCV)

*Dados de
treinamento*

1
2
3
4
5
6
7
8
9
10
11
12

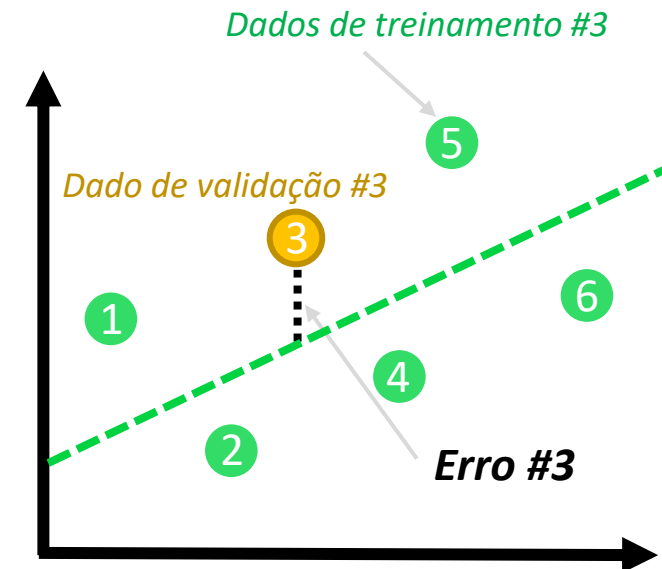


1
2
3
4
5
6
7
8
9
10
11
12

Subconjunto de treinamento #3

Dado de validação #3

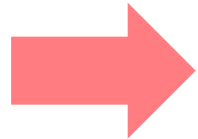
*Subconjunto de
treinamento #3*



Método: Leave-One-Out Cross-Validation (LOOCV)

Dados de
treinamento

1
2
3
4
5
6
7
8
9
10
11
12



1
2
3
4
5
6
7
8
9
10
11
12

Subconjunto de
treinamento #12

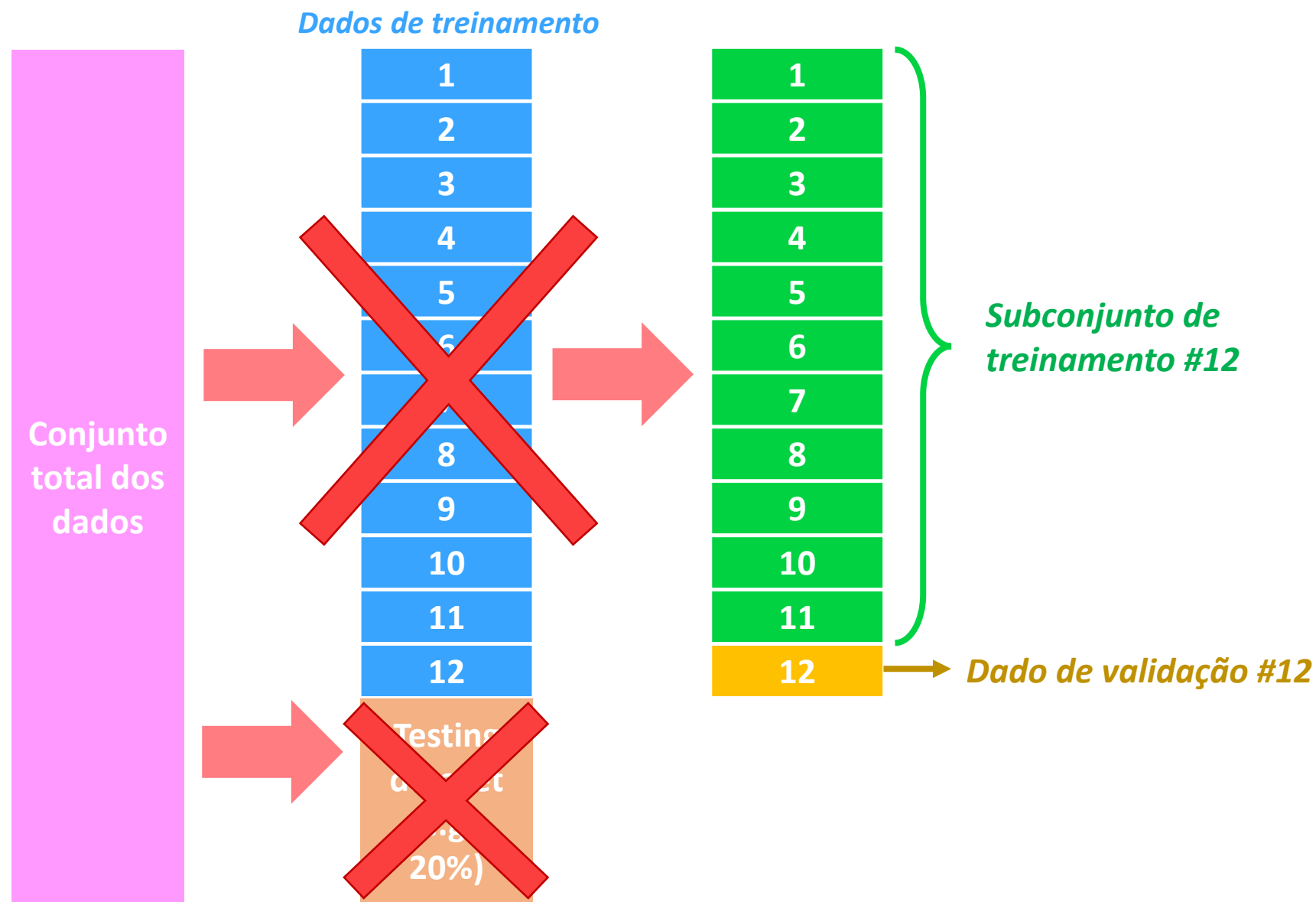
Dado de validação #12

Um erro geral é calculado com base na média dos erros calculados cada vez que **um dado de validação** foi deixado de fora dos **subconjuntos de treinamento** (#1 a #12, em nosso exemplo)

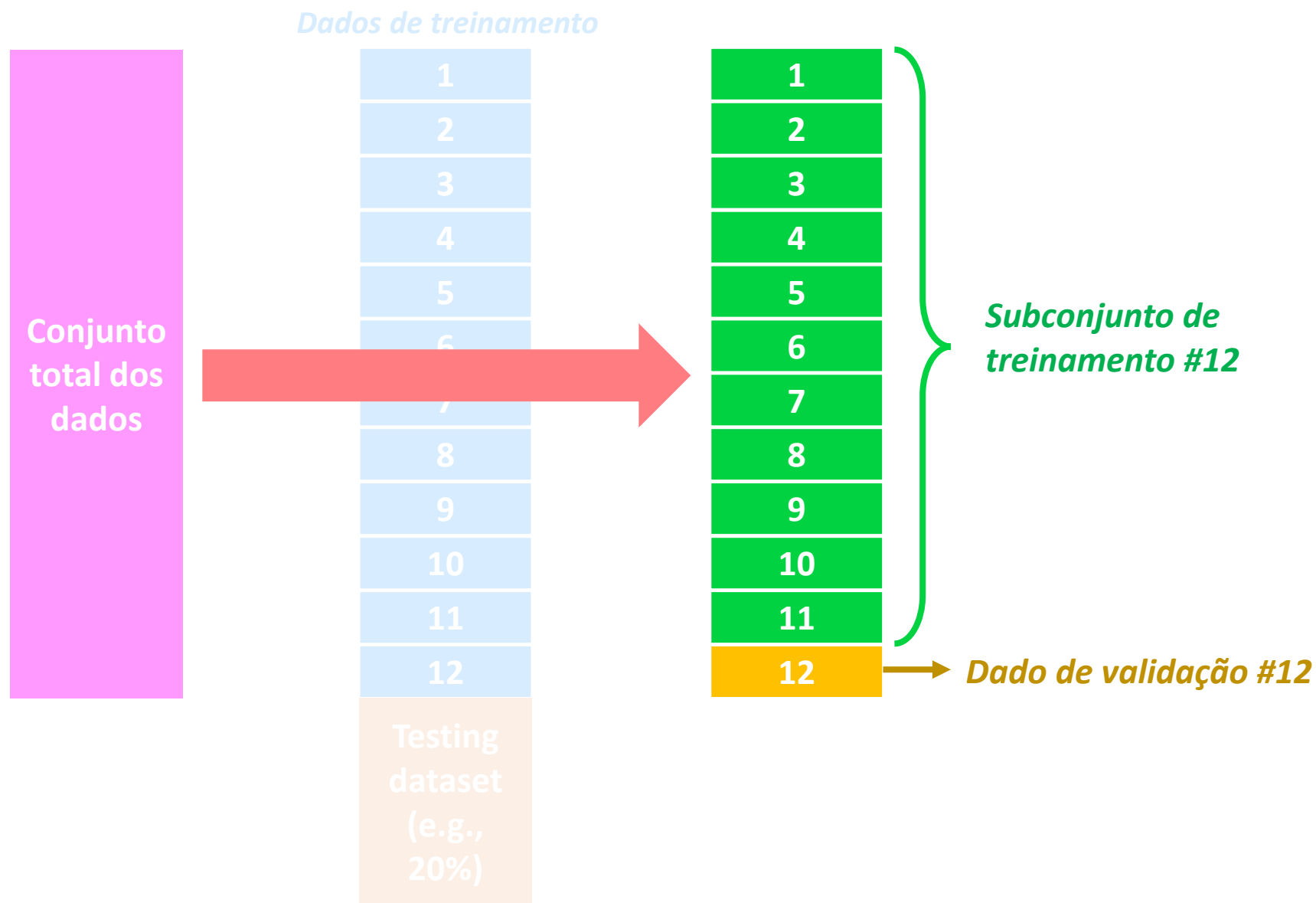
O modelo final a ser usado para fazer previsões é o modelo treinado em todo o **conjunto de dados de treinamento**

Obs.: Nos casos em que o conjunto de dados é muito pequeno, o LOOCV pode ser aplicado ao **conjunto total dos dados**. O modelo final também seria o modelo treinado com o **conjunto total dos dados**

Método: Leave-One-Out Cross-Validation (LOOCV)



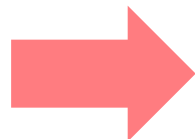
Método: Leave-One-Out Cross-Validation (LOOCV)



Método: K-Fold Cross-Validation

*Dados de
treinamento*

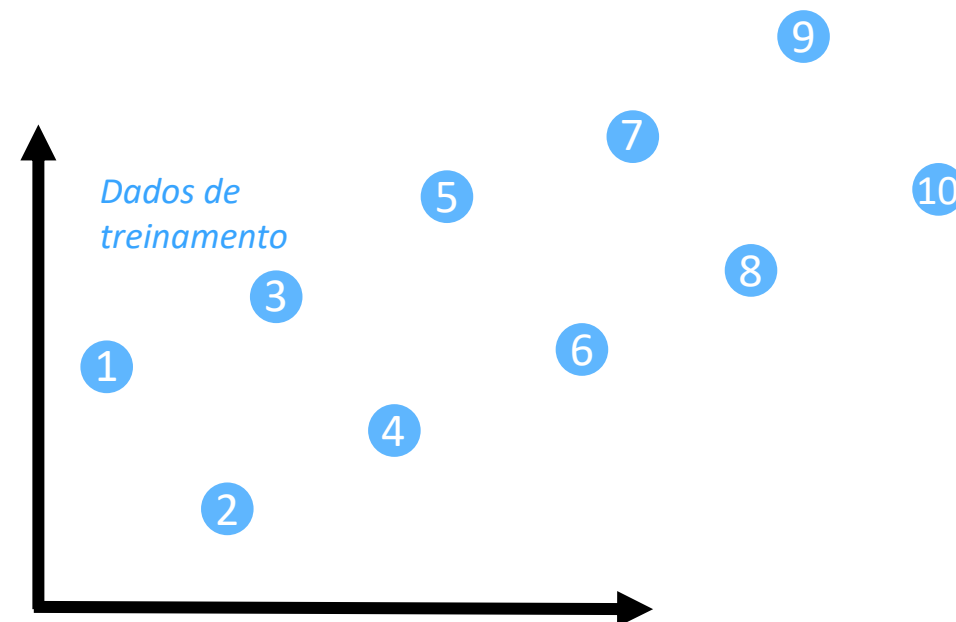
1
2
3
4
5
6
7
8
9
10
11
12



1
2
3
4
5
6
7
8
9
10
11
12

Fold de validação #1

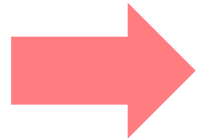
*Subconjunto de
treinamento #1*



Método: K-Fold Cross-Validation

*Dados de
treinamento*

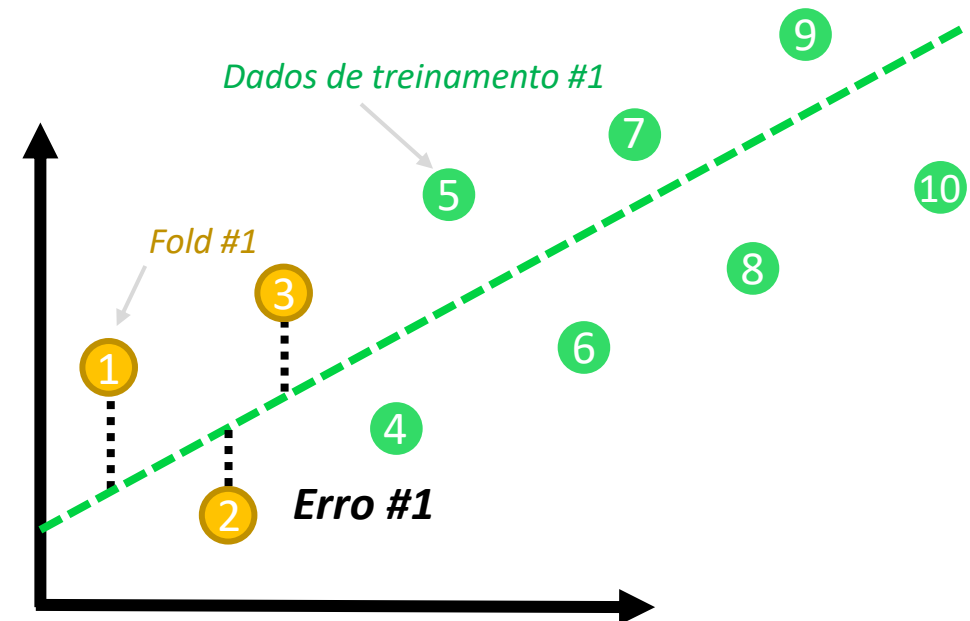
1
2
3
4
5
6
7
8
9
10
11
12



1
2
3
4
5
6
7
8
9
10
11
12

Fold de validação #1

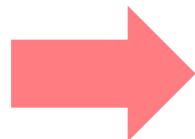
*Subconjunto de
treinamento #1*



Método: K-Fold Cross-Validation

*Whole
training data*

1
2
3
4
5
6
7
8
9
10
11
12

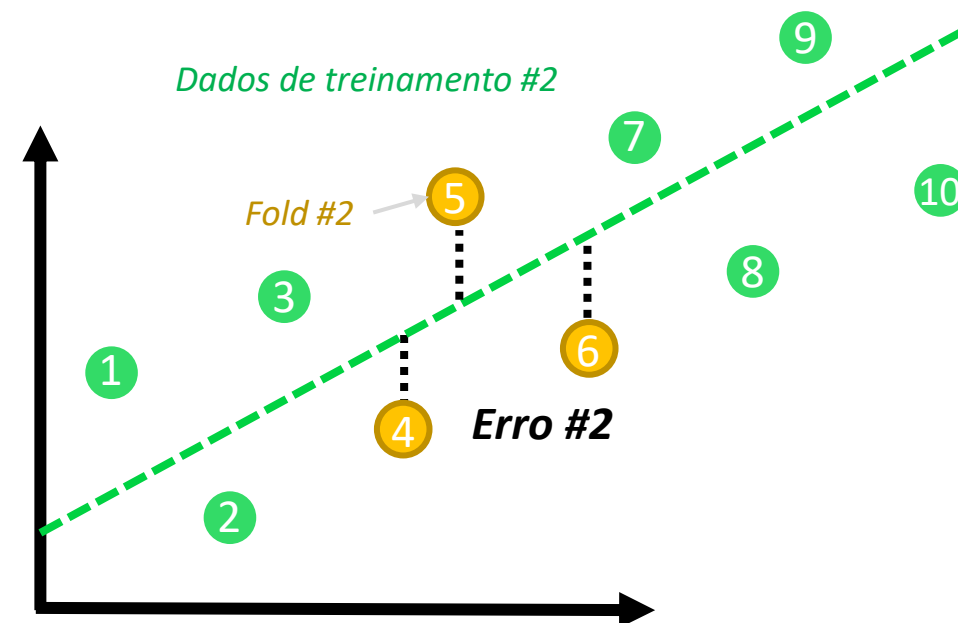


1
2
3
4
5
6
7
8
9
10
11
12

*Subconjunto de
treinamento #2*

Fold de validação #2

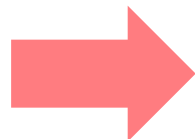
*Subconjunto de
treinamento #2*



Método: K-Fold Cross-Validation

*Dados de
treinamento*

1
2
3
4
5
6
7
8
9
10
11
12

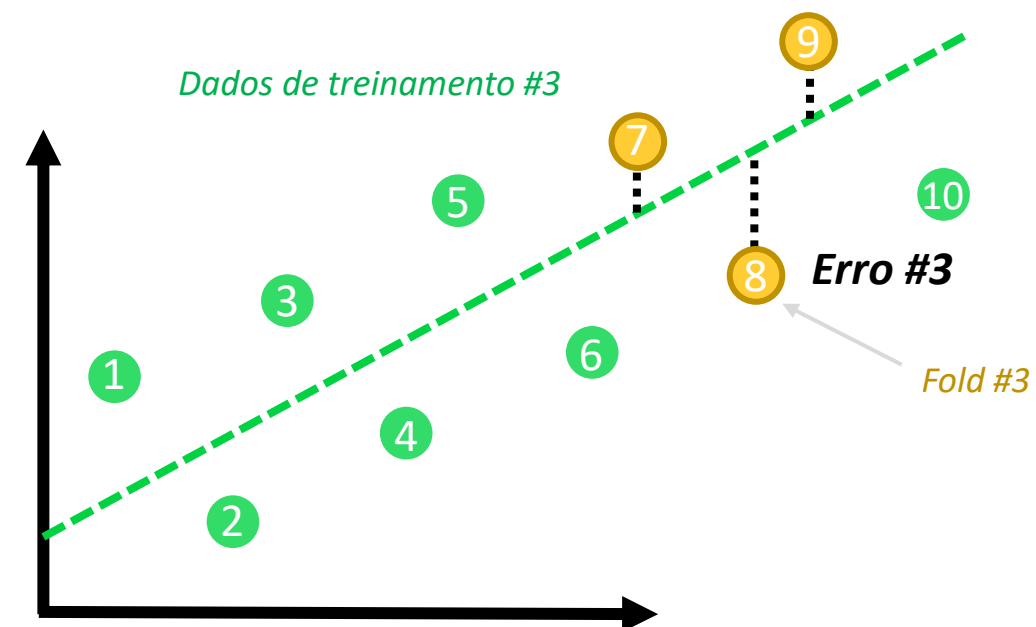


1
2
3
4
5
6
7
8
9
10
11
12

*Subconjunto de
treinamento #3*

Fold de validação #3

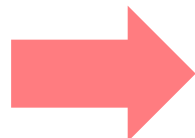
*Subconjunto de
treinamento #3*



Método: K-Fold Cross-Validation

Dados de
treinamento

1
2
3
4
5
6
7
8
9
10
11
12



1
2
3
4
5
6
7
8
9
10
11
12

Subconjunto de
treinamento #4

Fold de validação #4

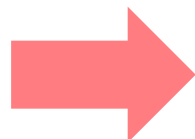
Neste exemplo, os **folds** são compostos por 3 observações cada, já que temos 12 observações em todo, temos **4 folds** ($K = 4$)

Novamente, um erro geral é calculado com base na média dos erros de previsão que foram calculados cada vez que um **fold** foi deixado de fora dos **subconjuntos de treinamento**

Método: K-Fold Cross-Validation

*Dados de
treinamento*

1
2
3
4
5
6
7
8
9
10
11
12



1
2
3
4
5
6
7
8
9
10
11
12

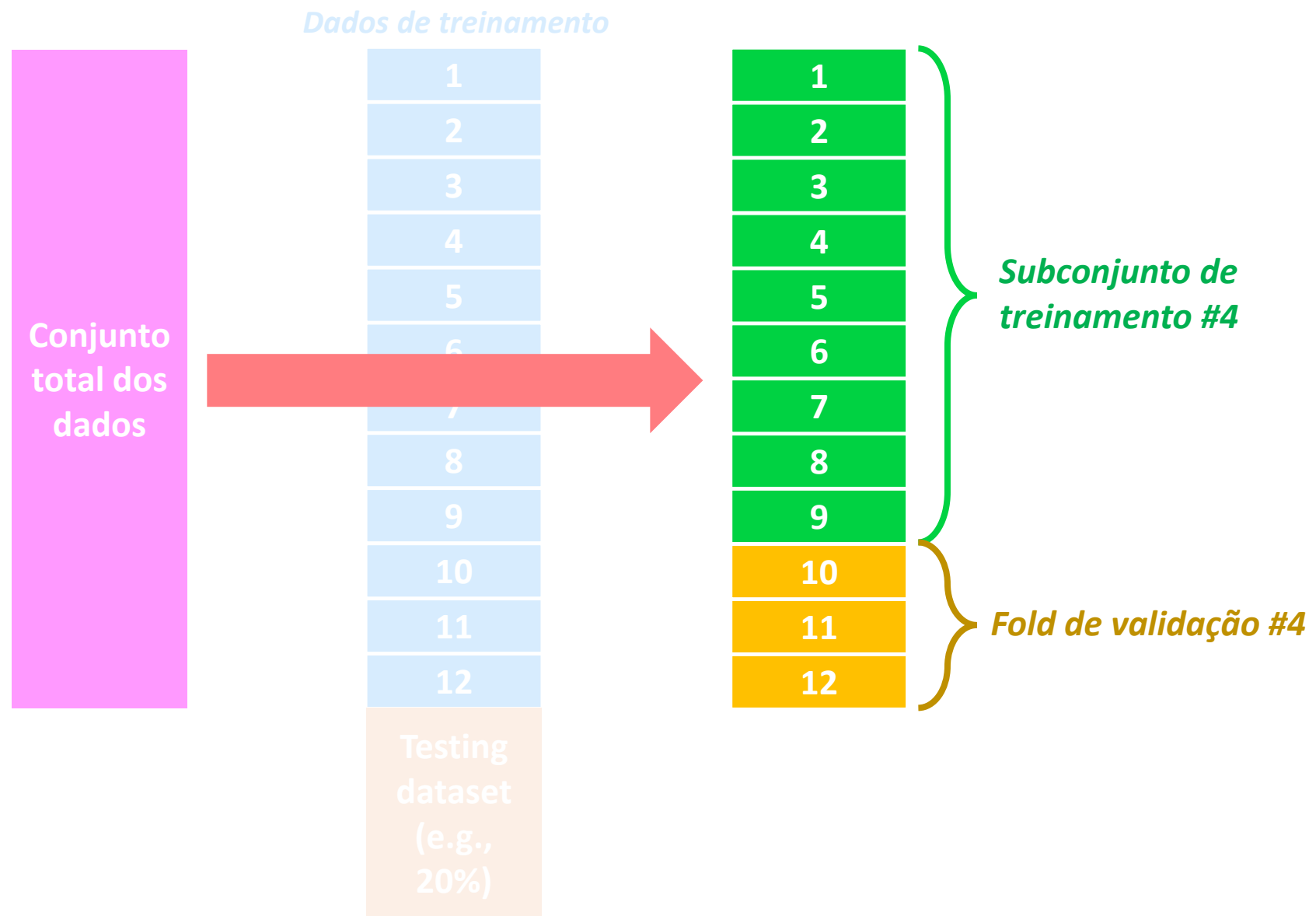
*Subconjunto de
treinamento #4*

Fold de validação #4

*Como no caso do LOOCV, o modelo usado no final para fazer previsões é o modelo treinado em todo o conjunto de **dados de treinamento***

*Obs.: Novamente, nos casos em que o conjunto de dados é muito pequeno, o método K-Fold pode ser aplicado ao **conjunto total dos dados**. O modelo usado também seria o modelo treinado com **conjunto total dos dados***

Método: K-Fold Cross-Validation



Utilização da validação cruzada (cross-validation)

Seleção de modelo e ajuste de hiperparâmetros: A validação cruzada ajuda a escolher o melhor modelo e otimizar os hiperparâmetros sem tocar no conjunto de dados de teste. O conjunto de teste deve ser usado idealmente apenas uma vez no final, para obter uma estimativa imparcial do desempenho do modelo no “mundo real”

Evitar o sobreajuste aos dados de treinamento: se você treinar um modelo em todo o conjunto de treinamento sem validação cruzada, poderá escolher um modelo que se ajuste bem aos dados de treinamento, mas generalize mal. A validação cruzada ajuda:

- Dividindo os dados de treinamento em vários subconjuntos
- Treinando o modelo em diferentes subconjuntos e validando os dados restantes
- Fornecer uma estimativa mais robusta de como o modelo funcionaria em dados não vistos

Aproveitar ao máximo os dados limitados: quando os dados são escassos, a validação cruzada permite um uso mais eficiente dos dados disponíveis. Em vez de reservar uma parte dos dados para teste (o que reduz o tamanho do conjunto usado para treinamento), a validação cruzada permite várias execuções de treinamento e validação no mesmo conjunto de dados

Estimar a variabilidade do desempenho do modelo: A validação cruzada fornece uma distribuição de métricas de desempenho (por exemplo, precisão, RMSE) em vez de apenas um único número. Isso ajuda a entender o quão estável e robusto é um modelo em diferentes subconjuntos de treinamento

Conjunto de teste permanece intocado até o final: Se você usar o conjunto de teste muito cedo (por exemplo, para ajustar hiperparâmetros), corre o risco de sobreajustá-lo. No momento em que você realmente avalia o modelo no conjunto de testes, ele pode não generalizar bem para outros novos dados invisíveis (além dos dados no próprio conjunto de dados de teste)

Métricas de validação para classificação binária

Classificação do modelo	Status real	Interpretação
Positivo (1)	Positivo (1)	= Verdadeiro positivo
Positivo (1)	Negativo (0)	= Falso positivo
Negativo (0)	Negativo (0)	= Verdadeiro negativo
Negativo (0)	Positivo (1)	= Falso negativo

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total Predictions}}$$

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

(True Positive Rate or Sensitivity)

$$\text{Specificity} = \frac{\text{True Negatives}}{\text{True Negatives} + \text{False Positives}}$$

(True Negative Rate)

Essas métricas podem ser combinadas em outras métricas, como **F1**, Receiver Operating Characteristic (**ROC**) e a Área Sob a Curva ROC (**AUC**)

Métricas de validação para classificação binária

Accuracy: a proporção de instâncias classificadas corretamente (verdadeiros positivos e verdadeiros negativos) em relação ao número total de instâncias

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total Predictions}}$$

Precision: Mede a proporção de previsões positivas verdadeiras entre todas as previsões positivas feitas pelo modelo. *Foco na exatidão das previsões positivas*

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

Recall (Sensitivity): mede a proporção de positivos reais que foram identificados corretamente pelo modelo. *Foco em capturar todas as instâncias positivas*

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

(True Positive Rate or Sensitivity)

Specificity: mede a proporção de negativos reais que foram corretamente identificados pelo modelo. *Foco em capturar todas as instâncias negativas*

$$\text{Specificity} = \frac{\text{True Negatives}}{\text{True Negatives} + \text{False Positives}}$$

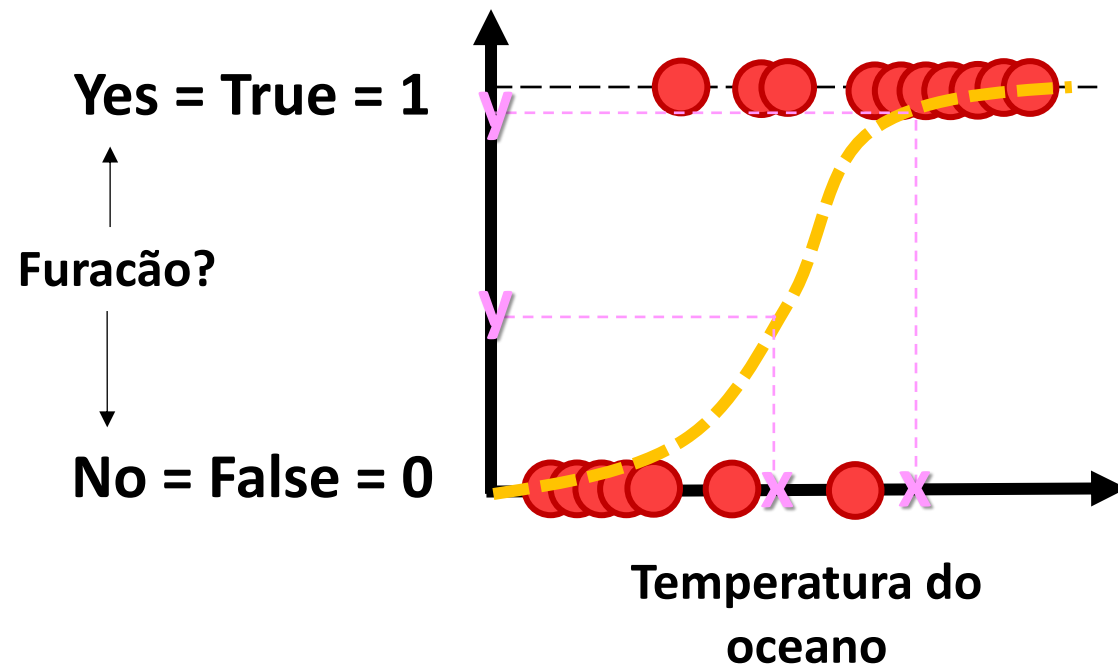
(True Negative Rate)

Essas métricas podem ser combinadas em outras métricas, como **F1, Receiver Operating Characteristic (**ROC**) e a Área Sob a Curva ROC (**AUC**)**



Voltando às regressões logísticas para classificação

Podemos prever furacões com base na temperatura do oceano?

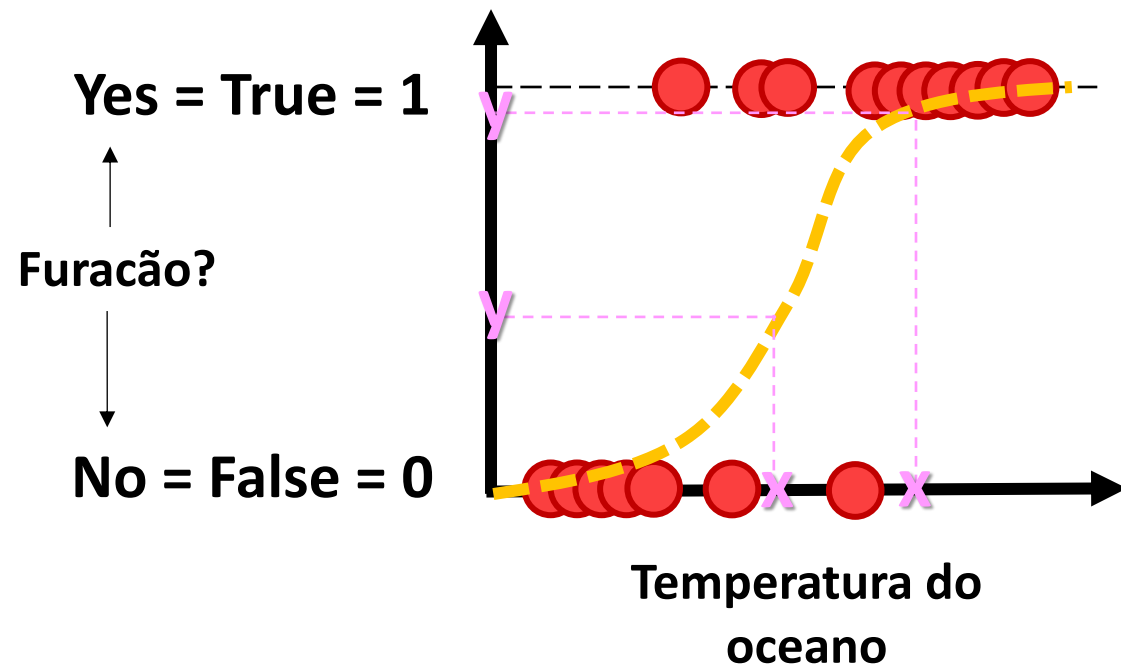


Com altas temperaturas do oceano, temos uma grande chance de ter um furacão

... Com temperaturas oceânicas mais baixas, temos uma chance menor de ter um furacão

Voltando às regressões logísticas para classificação

Podemos prever furacões com base na temperatura do oceano?



*Mas e se quisermos usar as probabilidades estimadas da regressão logística para classificar o risco de furacões em duas categorias: **furacão** e **sem furacão**?*

Voltando às regressões logísticas para classificação

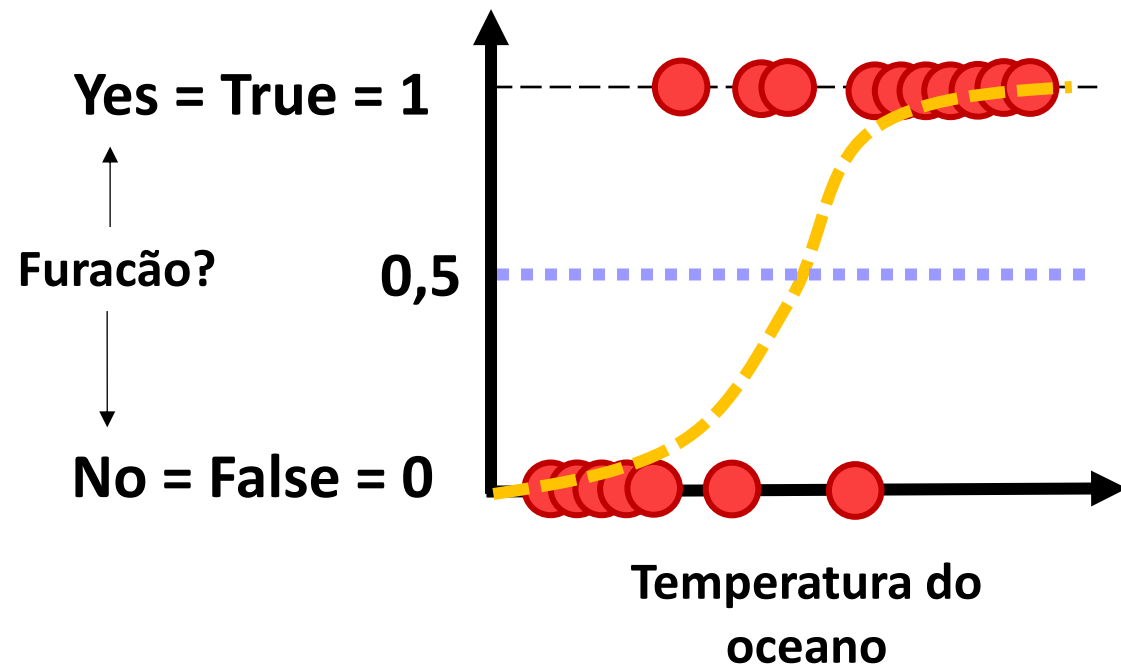
$$\ln \left(\frac{P}{1 - P} \right) = \alpha + \beta x + \varepsilon$$

- Se $p \geq 0,5$: classificado como classe 1 (**furacão**, em nosso exemplo)
- Se $p < 0,5$: classificado como classe 0 (**sem furacão**, em nosso exemplo)

O limite de probabilidade usado para classificação na regressão logística é normalmente definido em 0,5 por padrão. Isso significa que, se a probabilidade prevista de uma observação pertencente à categoria positiva (classe 1) for 0,5 ou maior, a observação será classificada nessa categoria; caso contrário, é classificado na categoria negativa (classe 0)

Voltando às regressões logísticas para classificação

Receiver Operating Characteristic (ROC) & Area Under the ROC Curve (AUC)

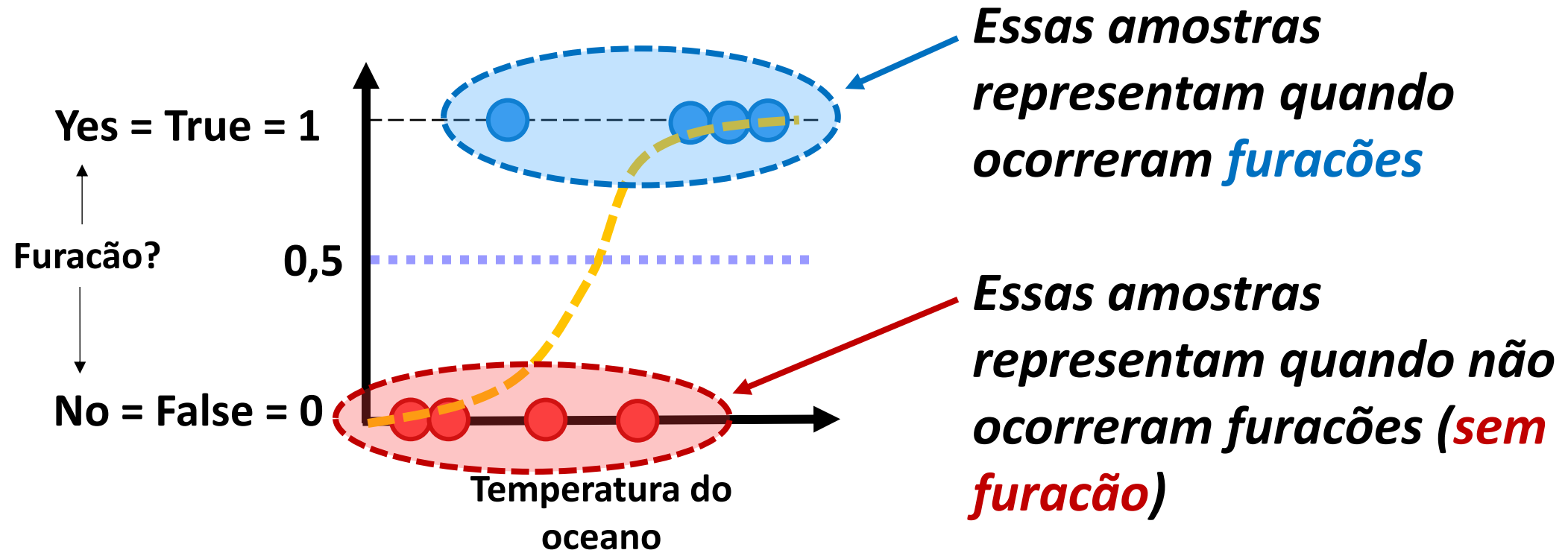


*Graficamente, o que temos é uma **linha limite** cortando a curva logística com probabilidade = 0,5*

Vamos remover algumas observações do nosso gráfico e alterar algumas cores para facilitar a interpretação...

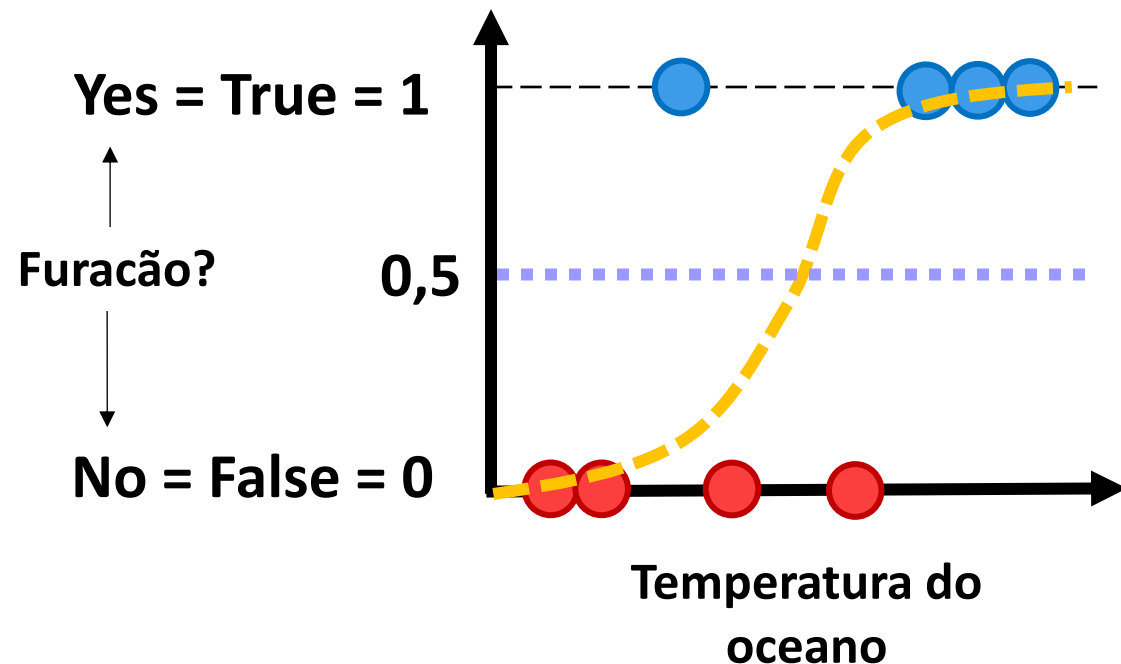
Voltando às regressões logísticas para classificação

Receiver Operating Characteristic (ROC) & Area Under the ROC Curve (AUC)



Voltando às regressões logísticas para classificação

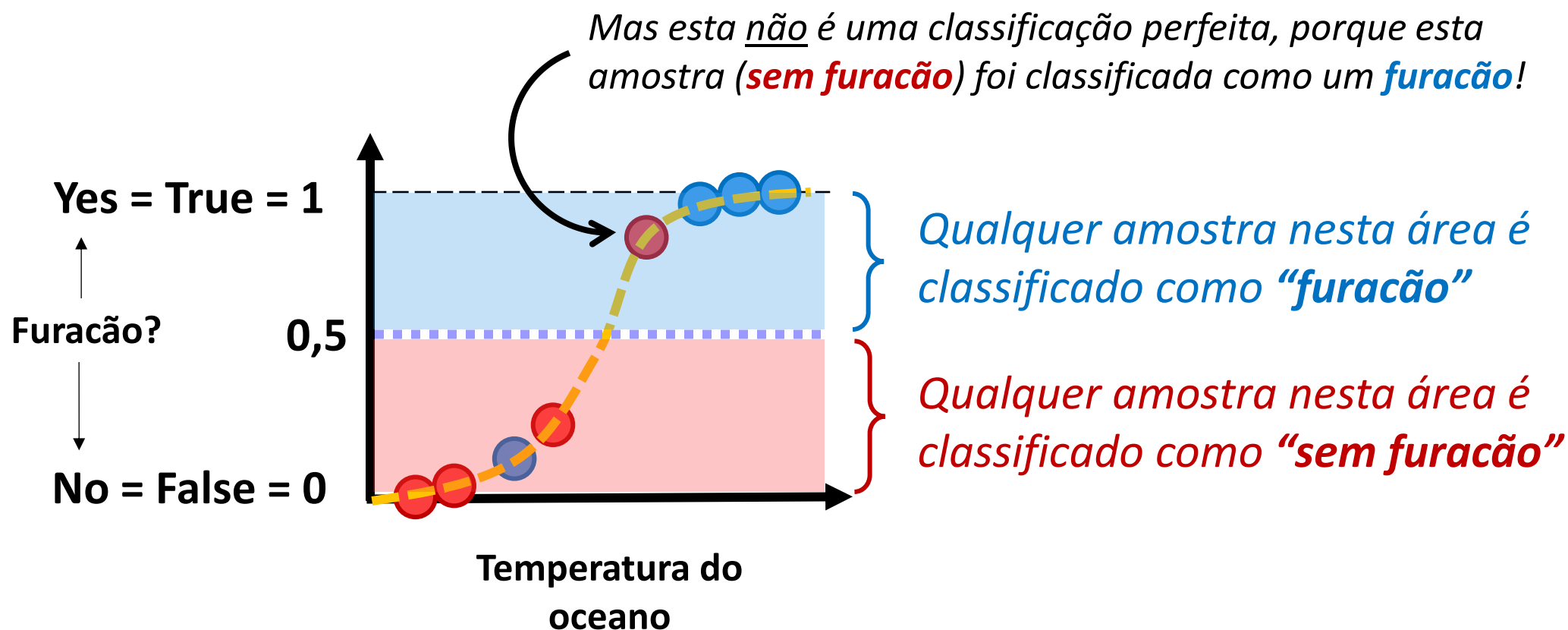
Receiver Operating Characteristic (ROC) & Area Under the ROC Curve (AUC)



*Vamos colocar essas amostras em cima da **linha** e interpretar qual resultado nosso **modelo logístico** previu para elas em termos de ocorrência de furacões, dado nosso **limite de probabilidade de 0,5** ...*

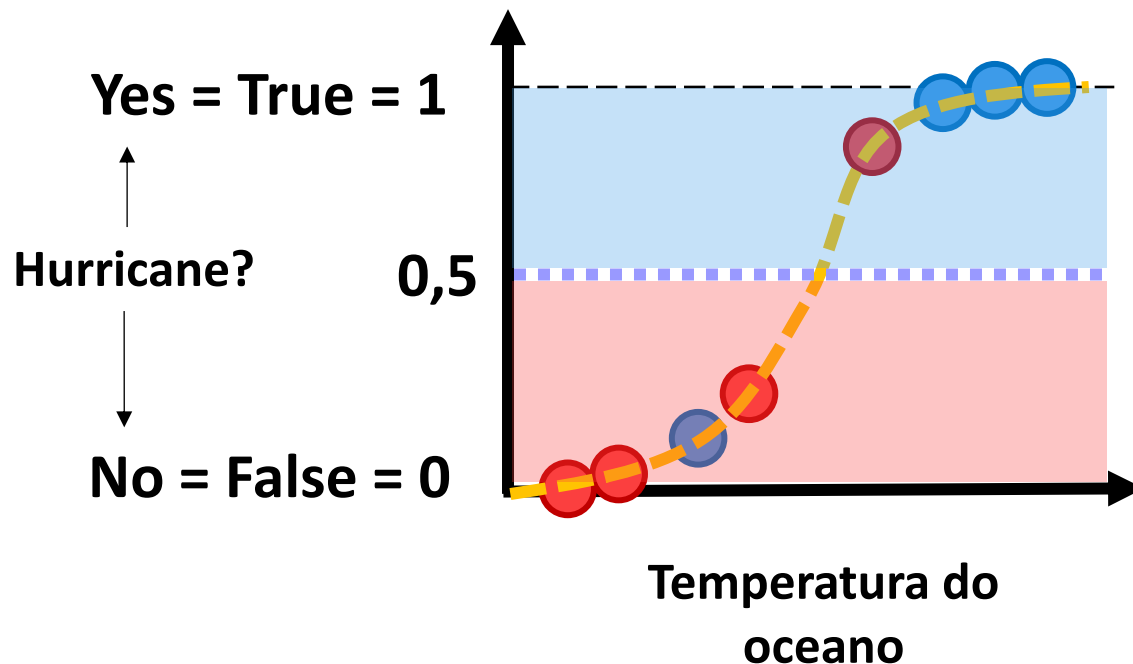
Voltando às regressões logísticas para classificação

Receiver Operating Characteristic (ROC) & Area Under the ROC Curve (AUC)



Voltando às métricas de classificação binárias

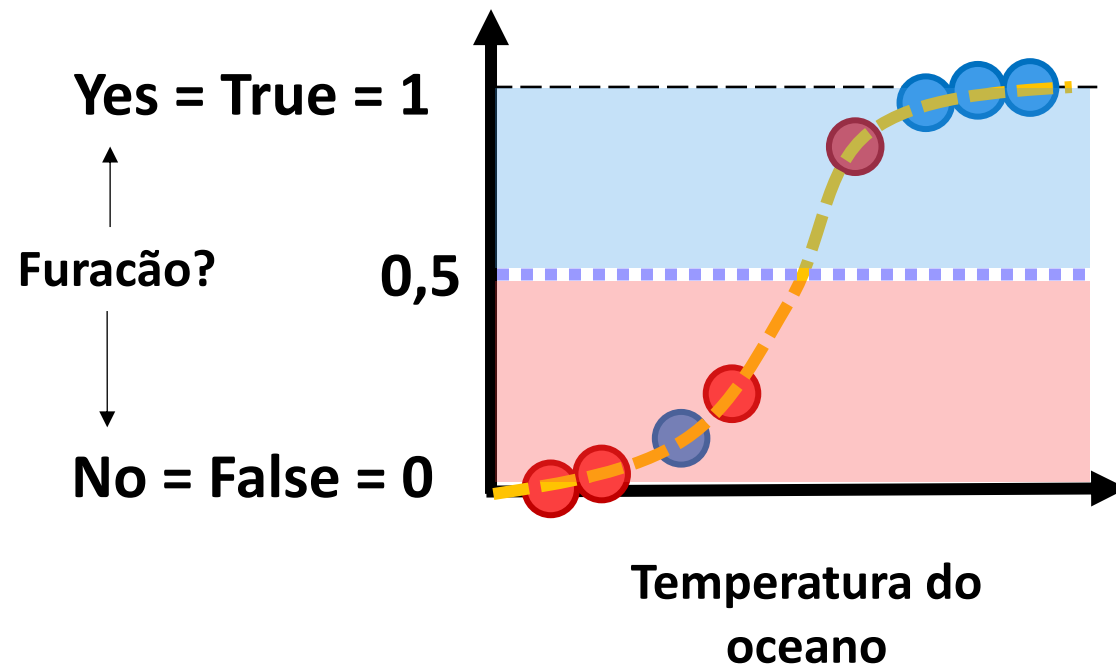
Receiver Operating Characteristic (ROC) & Area Under the ROC Curve (AUC)



Classificação do modelo	Status real	Interpretação
Positivo (1)	Positivo (1)	= Verdadeiro positivo
Positivo (1)	Negativo (0)	= Falso positivo
Negativo (0)	Negativo (0)	= Verdadeiro negativo
Negativo (0)	Positivo (1)	= Falso negativo

Voltando às métricas de classificação binárias

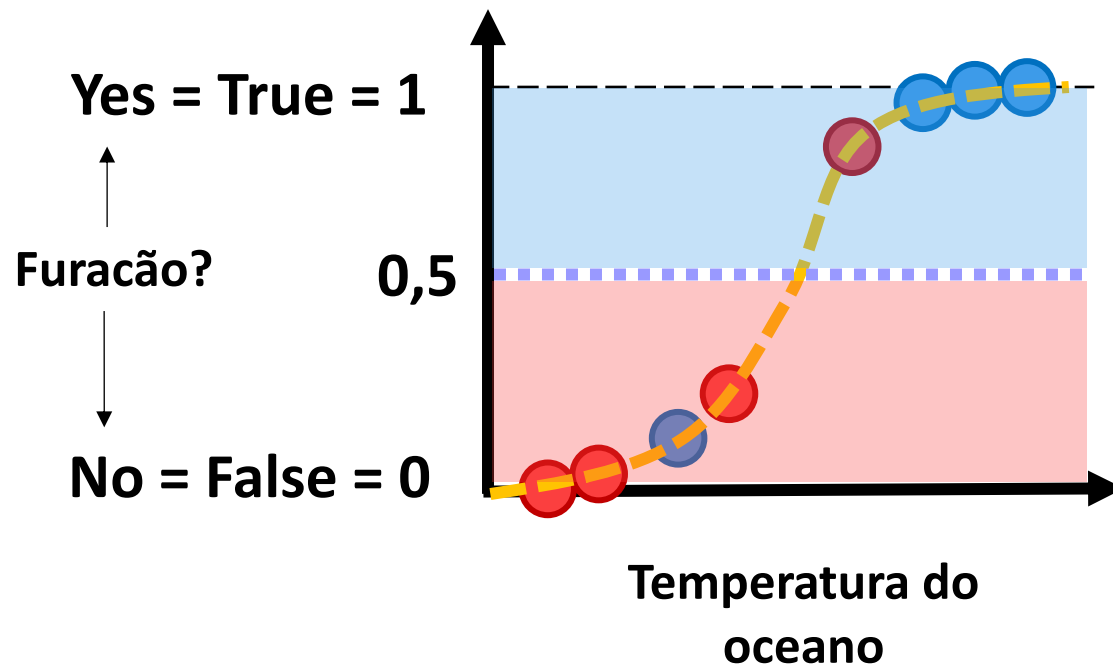
Receiver Operating Characteristic (ROC) & Area Under the ROC Curve (AUC)



Classificação do modelo	Status real	Interpretação
Furacão (1)	Furacão (1)	= True positive
Furacão (1)	Sem furacão (0)	= False positive
Sem furacão (0)	Sem furacão (0)	= True negative
Sem furacão (0)	Furacão (1)	= False negative

Voltando às métricas de classificação binárias

Receiver Operating Characteristic (ROC) & Area Under the ROC Curve (AUC)



$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total Predictions}}$$

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

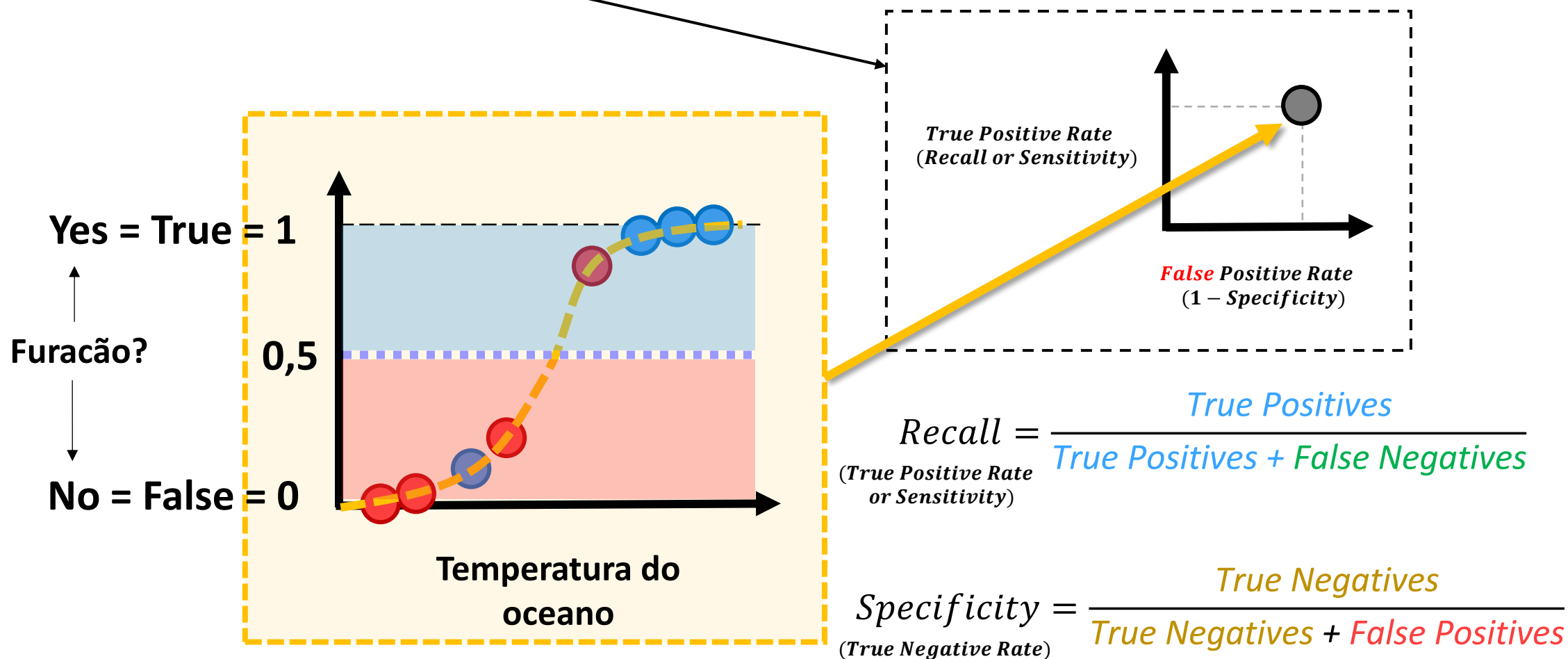
(True Positive Rate or Sensitivity)

$$\text{Specificity} = \frac{\text{True Negatives}}{\text{True Negatives} + \text{False Positives}}$$

(True Negative Rate)

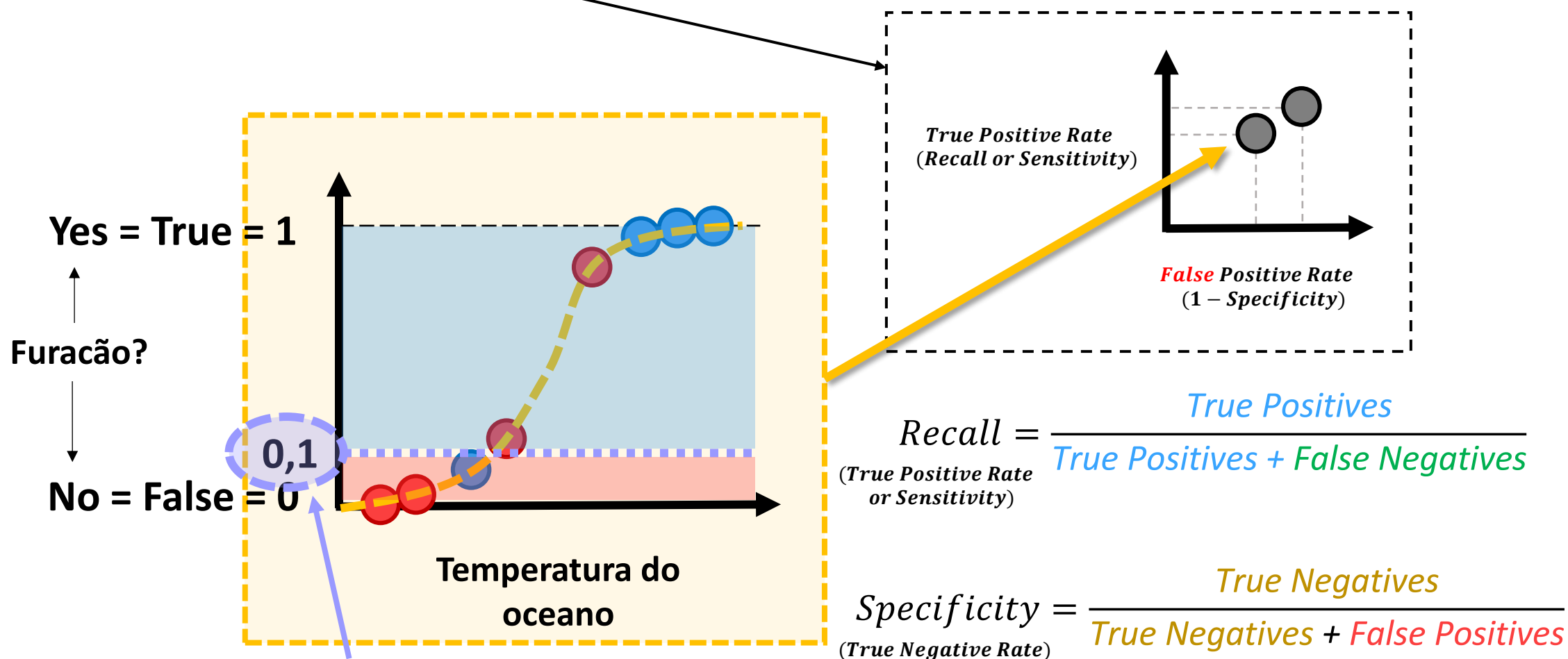
Calculando e plotando valores ROC

Receiver Operating Characteristic (ROC) & Area Under the ROC Curve (AUC)



Calculando e plotando valores ROC

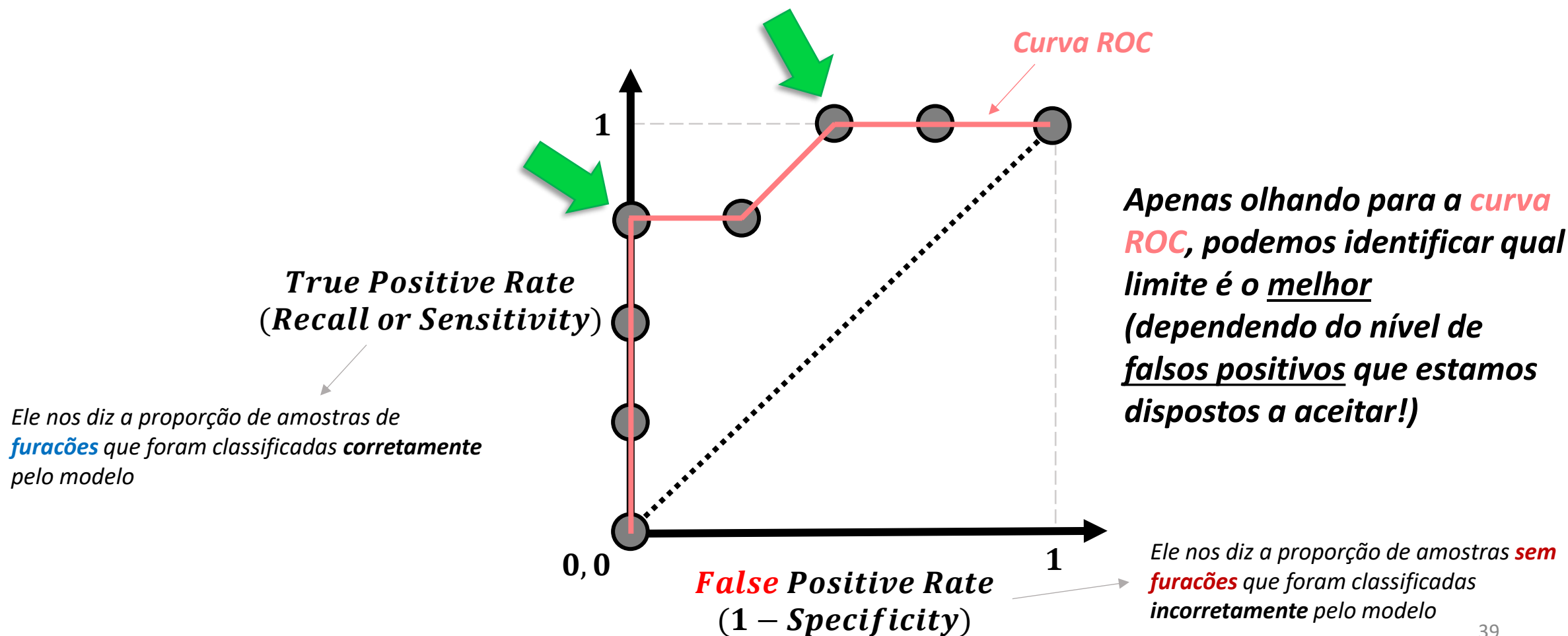
Receiver Operating Characteristic (ROC) & Area Under the ROC Curve (AUC)



E continuamos alterando os *limites* e plotando os novos resultados ...

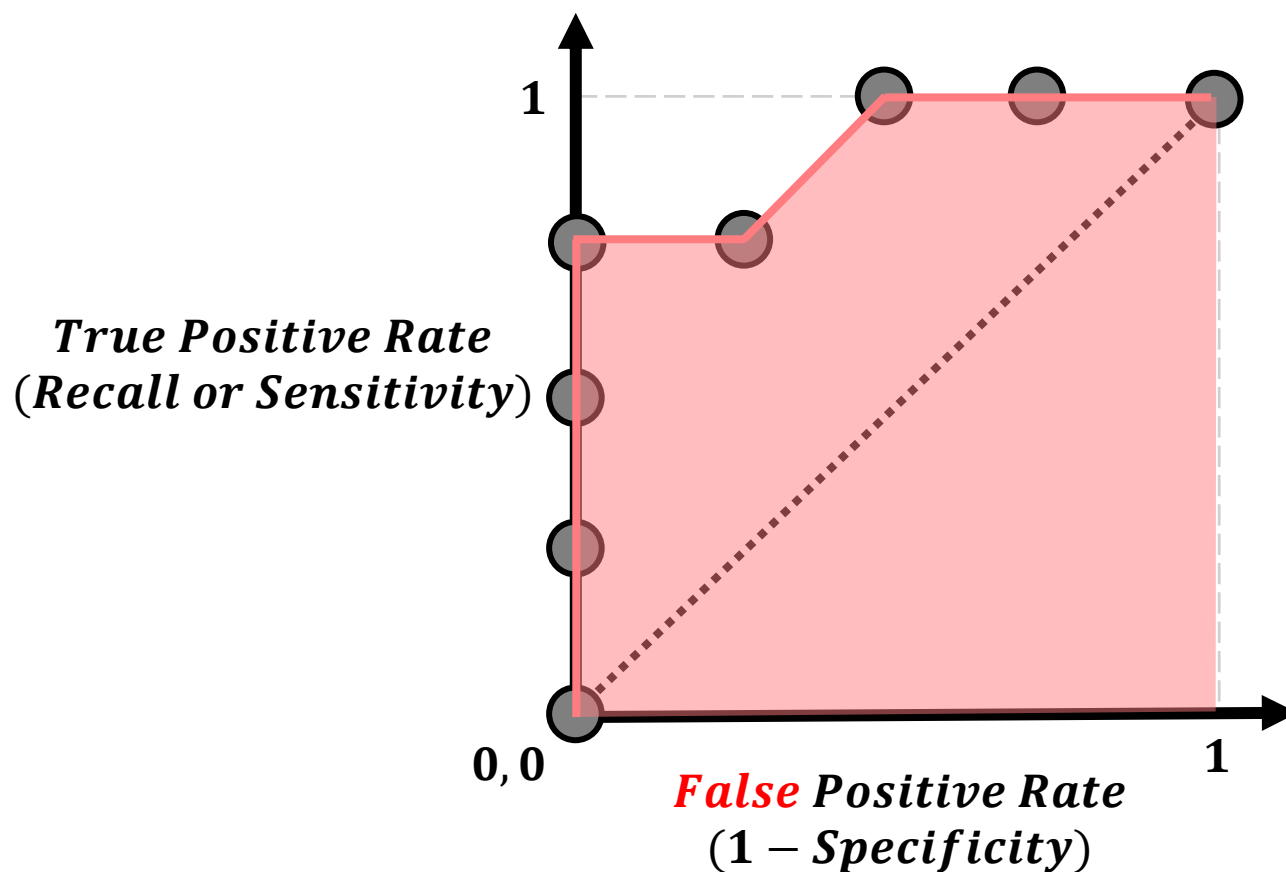
Calculando e plotando valores ROC

Receiver Operating Characteristic (ROC) & Area Under the ROC Curve (AUC)



Calculando e plotando a AUC

*Receiver Operating Characteristic (ROC) & Area Under the ROC Curve (**AUC**)*

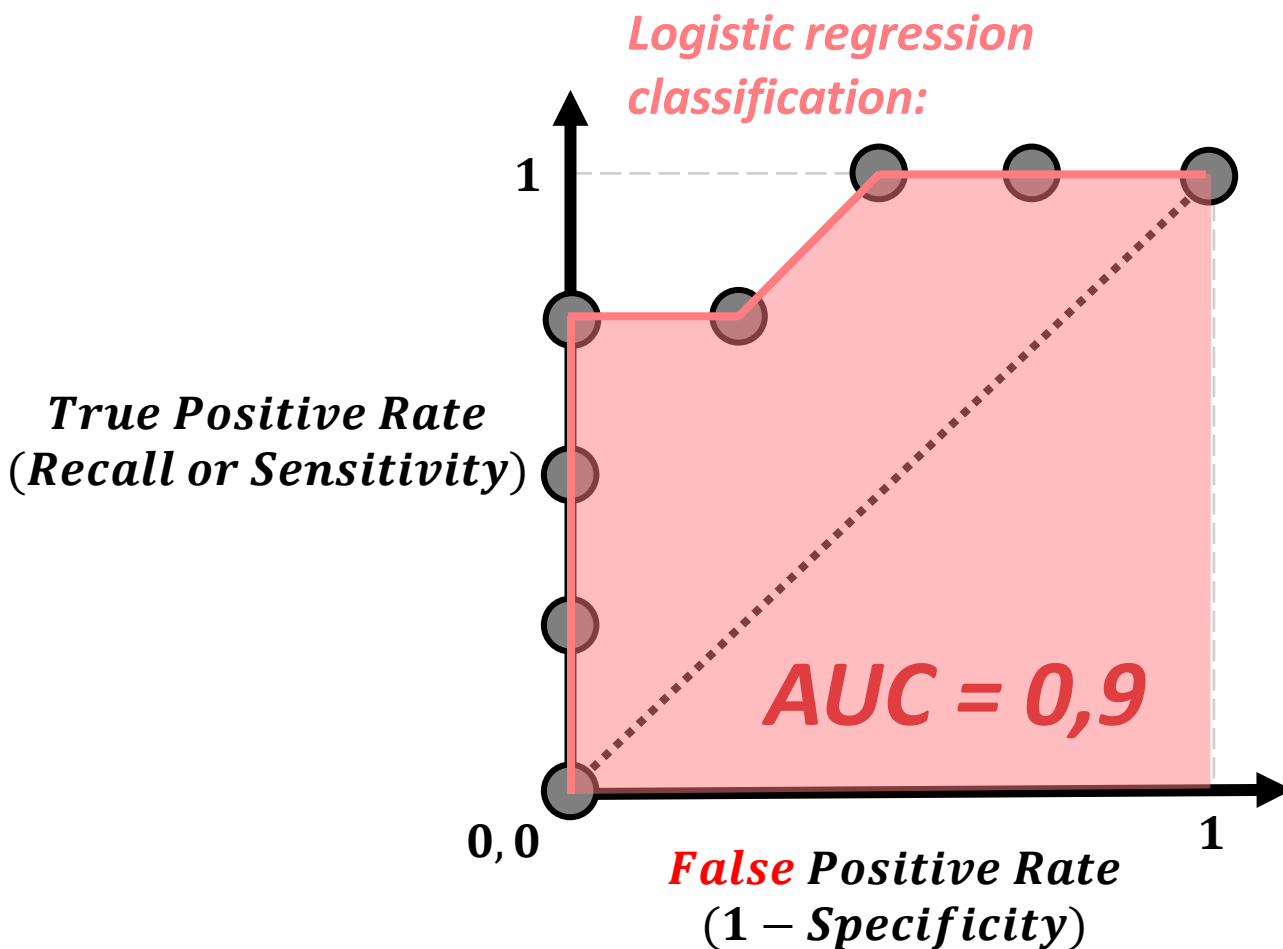


AUC = 0,9

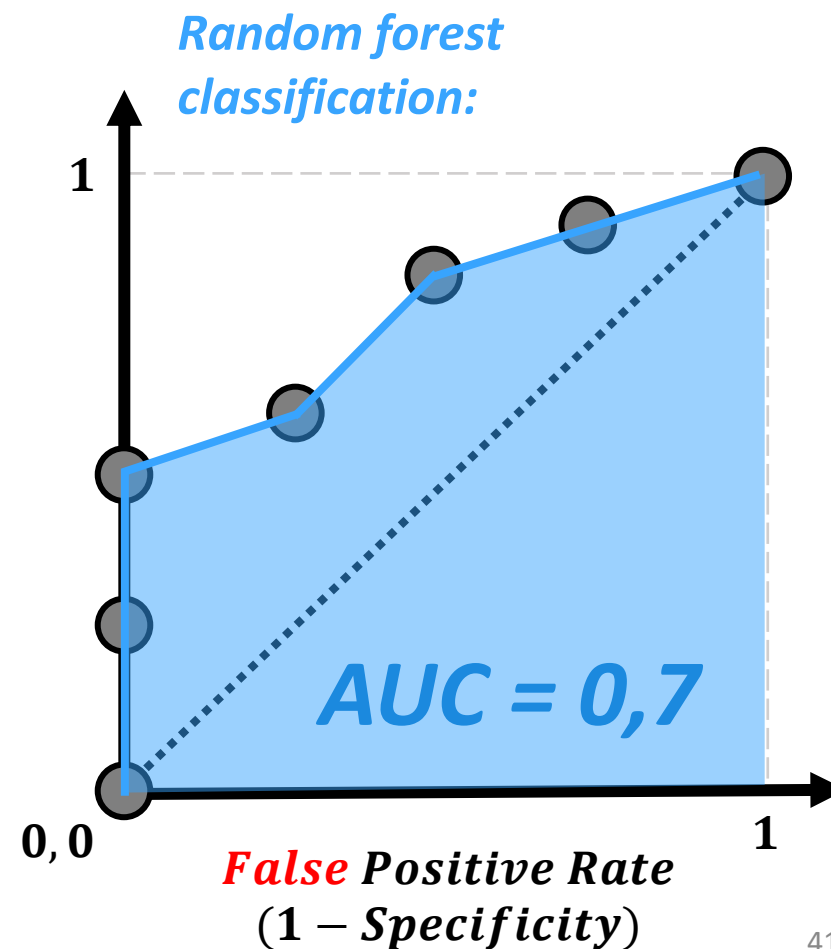
*A AUC facilita a comparação de uma **curva ROC** com outra (por exemplo, de um modelo de classificação diferente)*

Calculando e plotando a AUC

Receiver Operating Characteristic (ROC) & Area Under the ROC Curve (**AUC**)

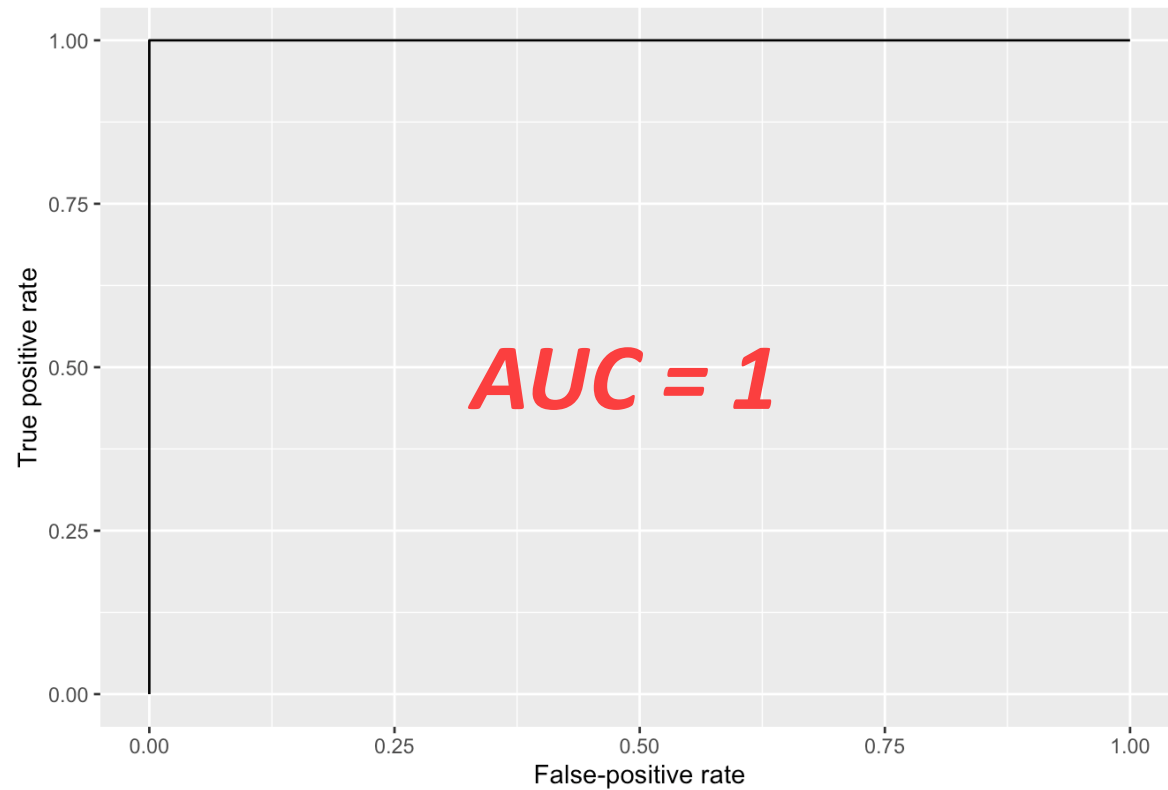


VS.

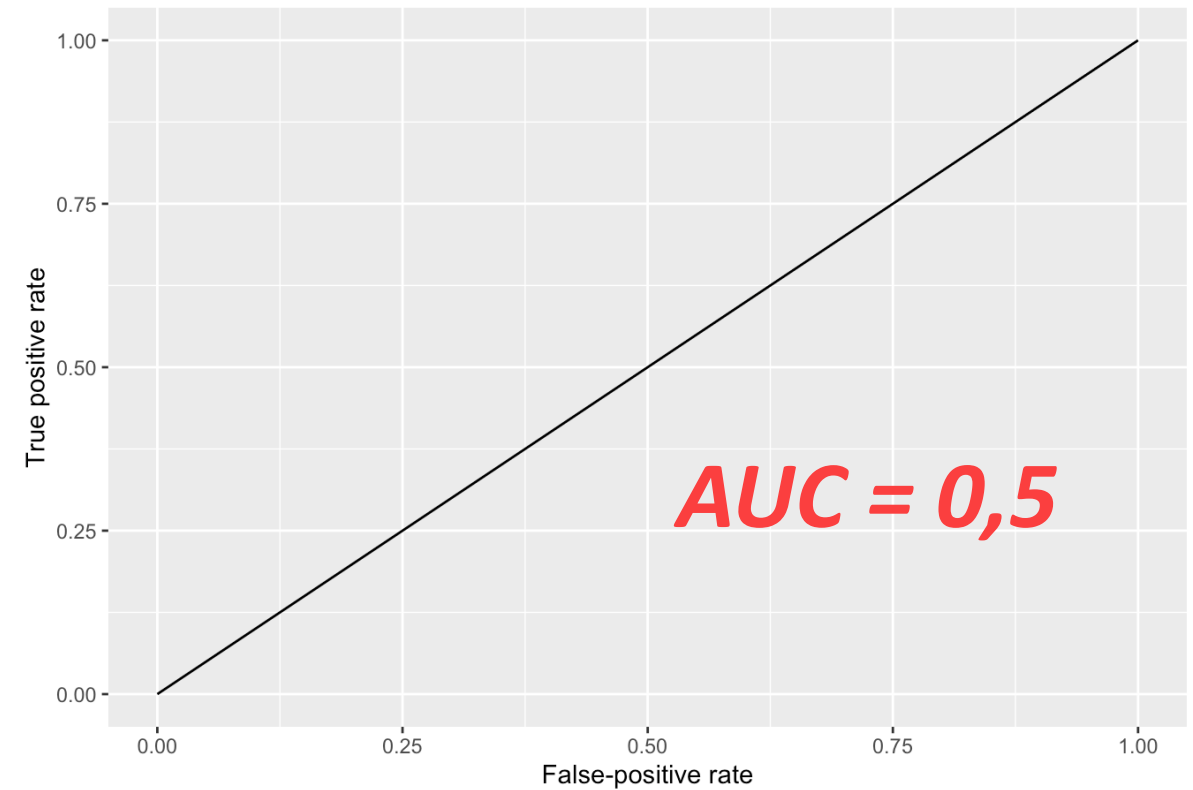


Modelo perfeito vs. Modelo “aleatório”

ROC curve for a perfect model



ROC curve for a guessing model



Resumo

- *Existem vários métodos de validação de modelo, alguns se aplicam a modelos de previsão (numéricos), outros se aplicam a modelos de classificação*
- ***Validação cruzada** é um termo geral usado para descrever uma "família" de métodos de validação baseados no particionamento dos dados em vários subconjuntos, por exemplo, K-fold e LOOCV*
- ***ROC** e **AUC** não são métodos de validação cruzada, mas são métodos muito populares para avaliar o desempenho de modelos de classificação binária*
- *Embora alguns métodos de validação possam parecer complexos, geralmente eles podem ser calculados com uma simples linha de código em Python ou R*