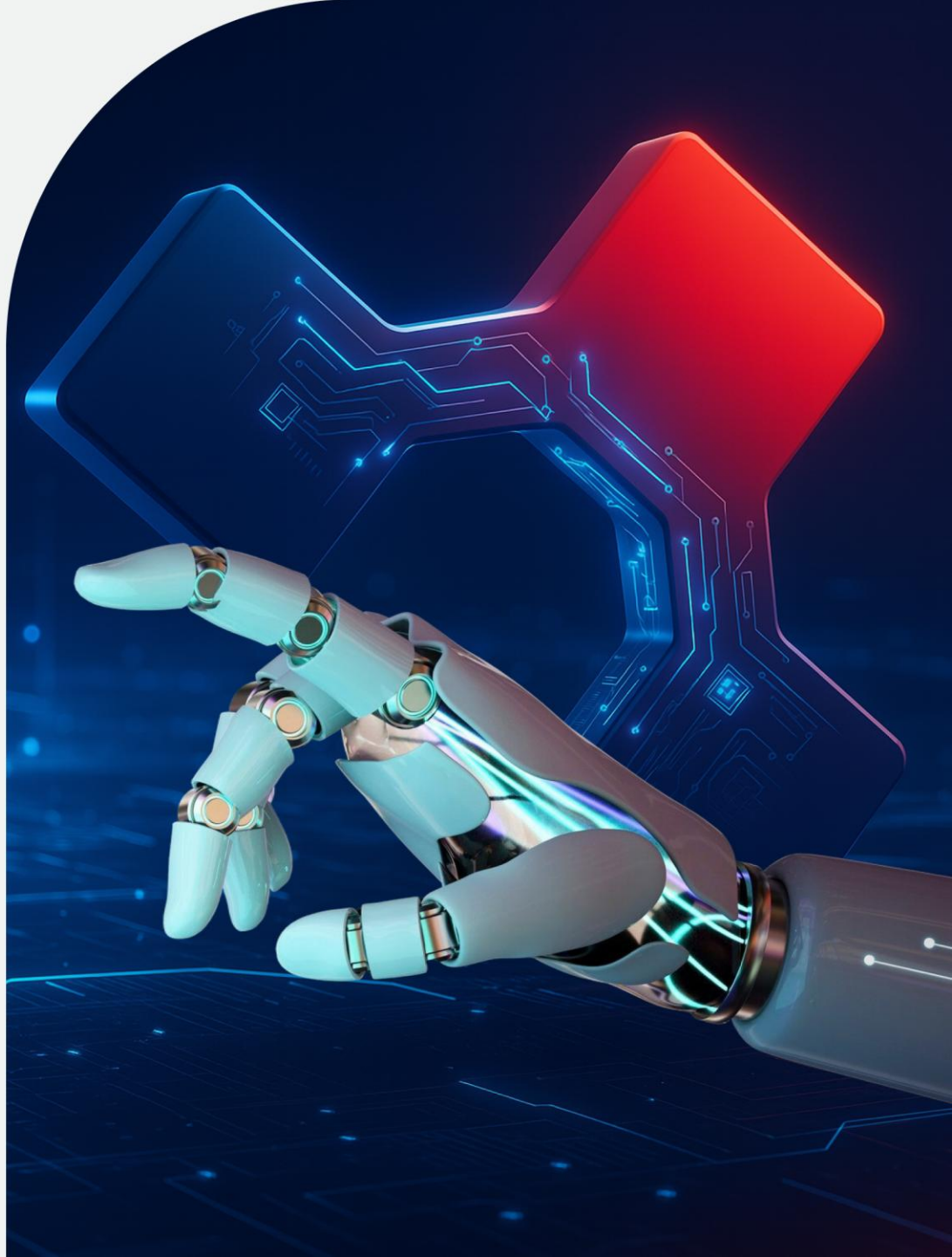
NCIA

Núcleo de Capacitação em Inteligência Artificial

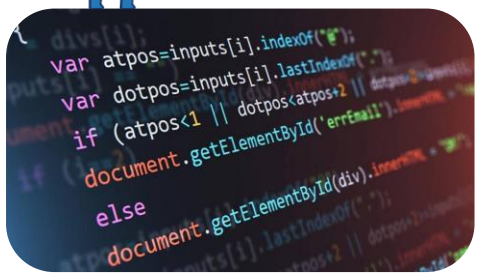


NCIA | FOXCONN FPF tech

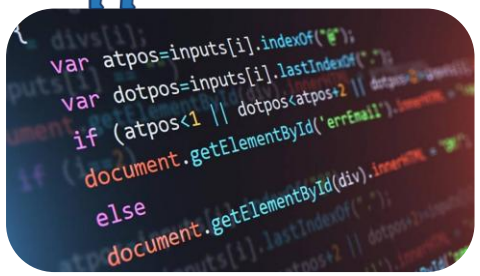
Main Challenges of Machine Learning

Insufficient Quantity of Training Data, Nonrepresentative Training Data, Poor-Quality Data, Irrelevant Features, Overfitting the Training Data, Underfitting the Training Data

Algoritmos Ruins

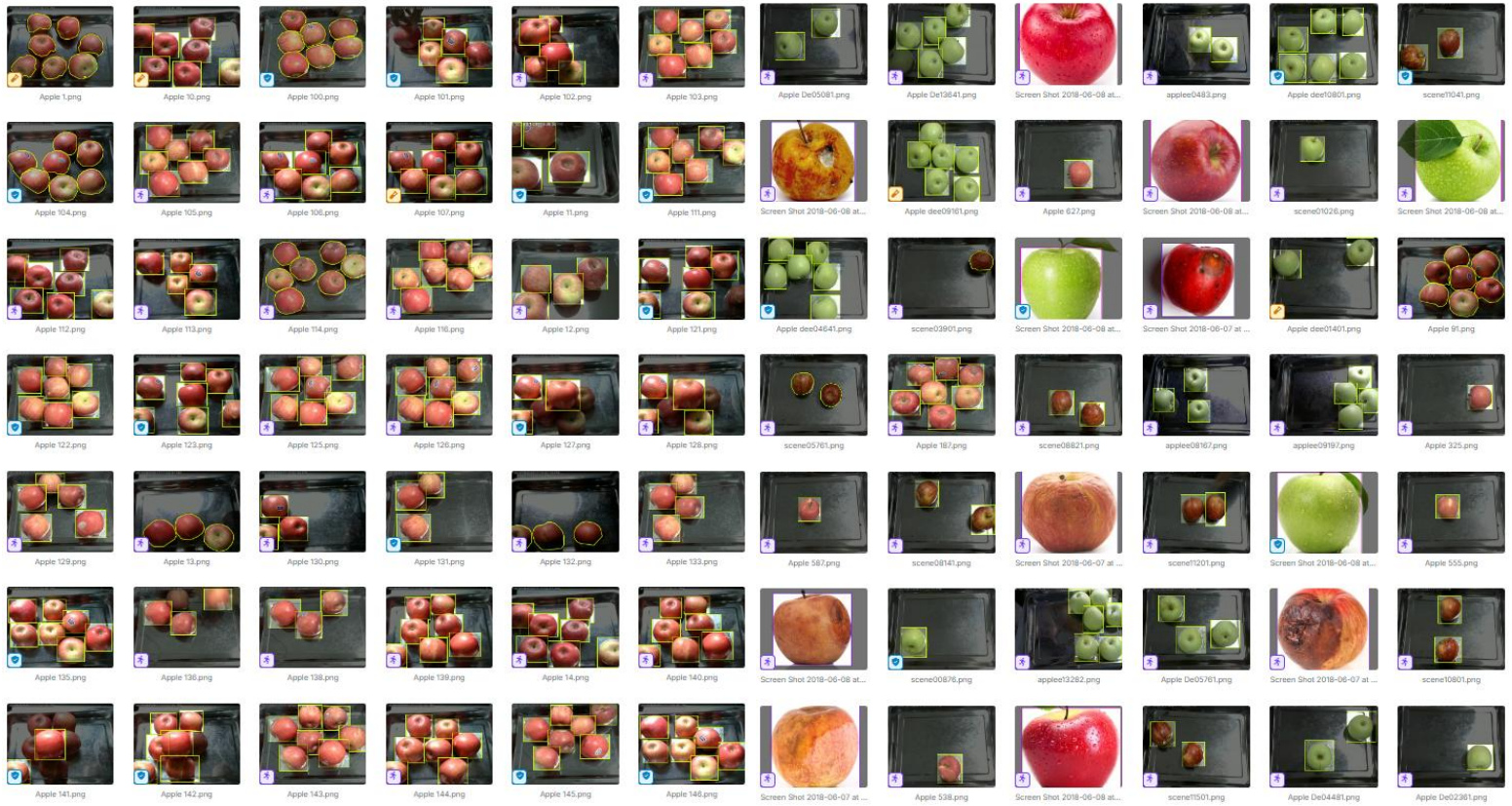


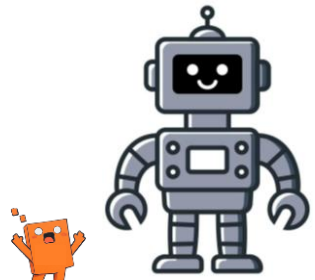
Algoritmos Ruins



Dados Ruins

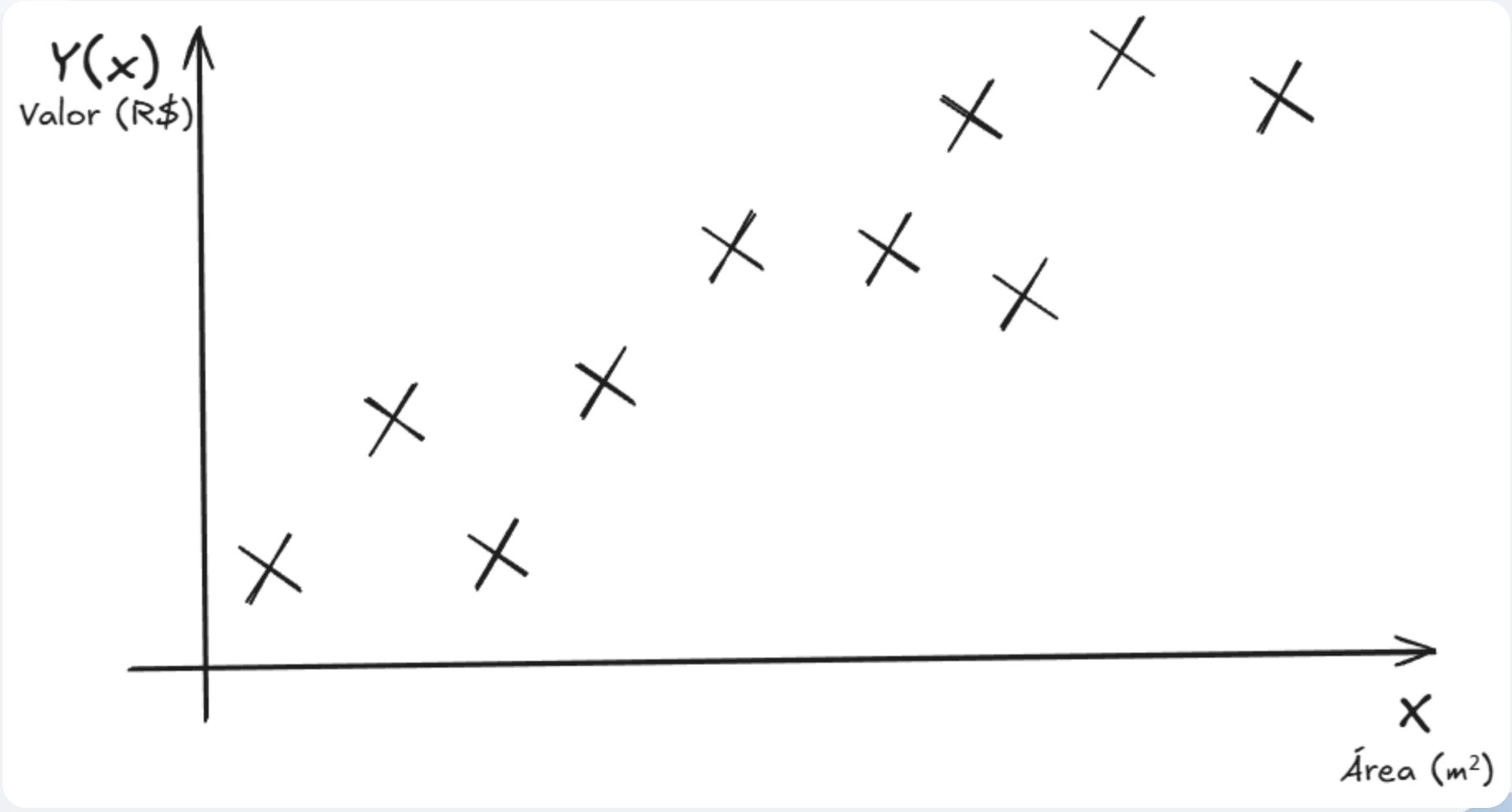
Insufficient Quantity of Training Data

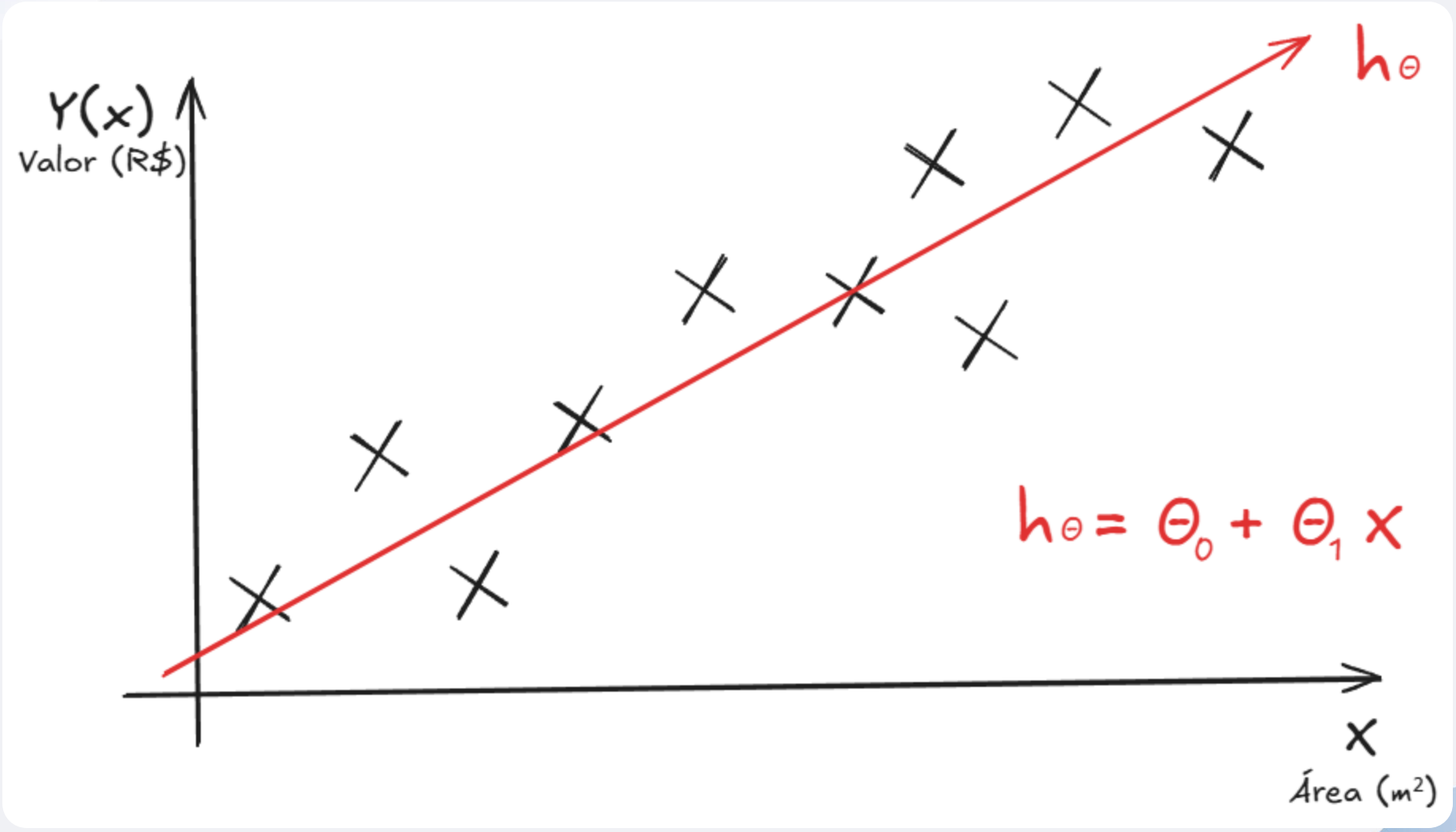


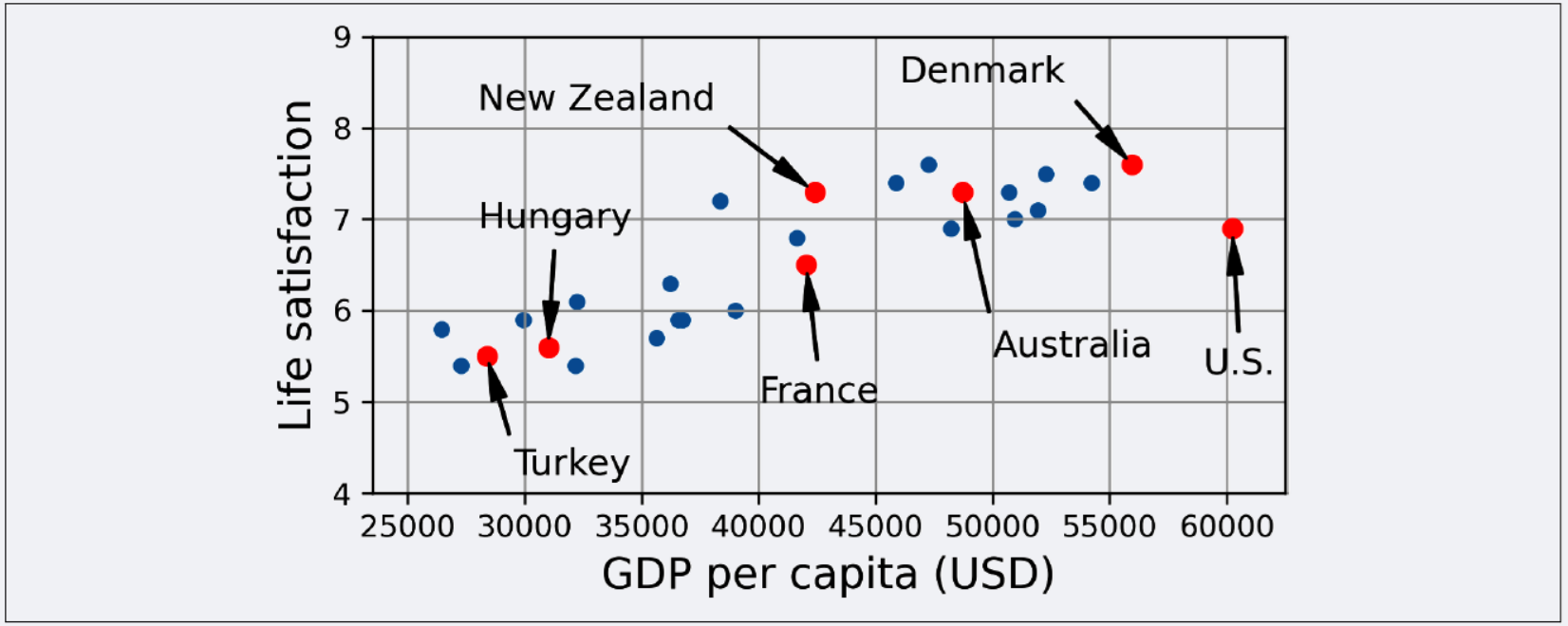


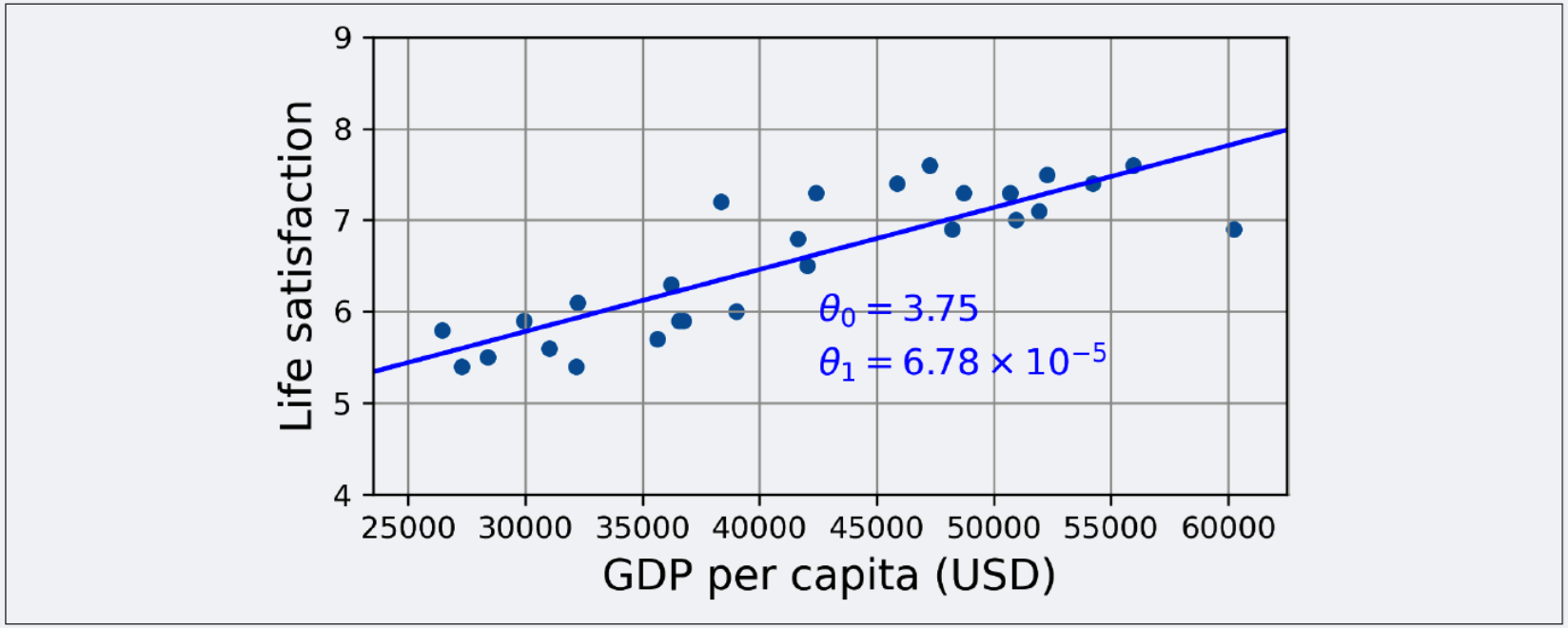
NCIA | FOXCONN FPFtech

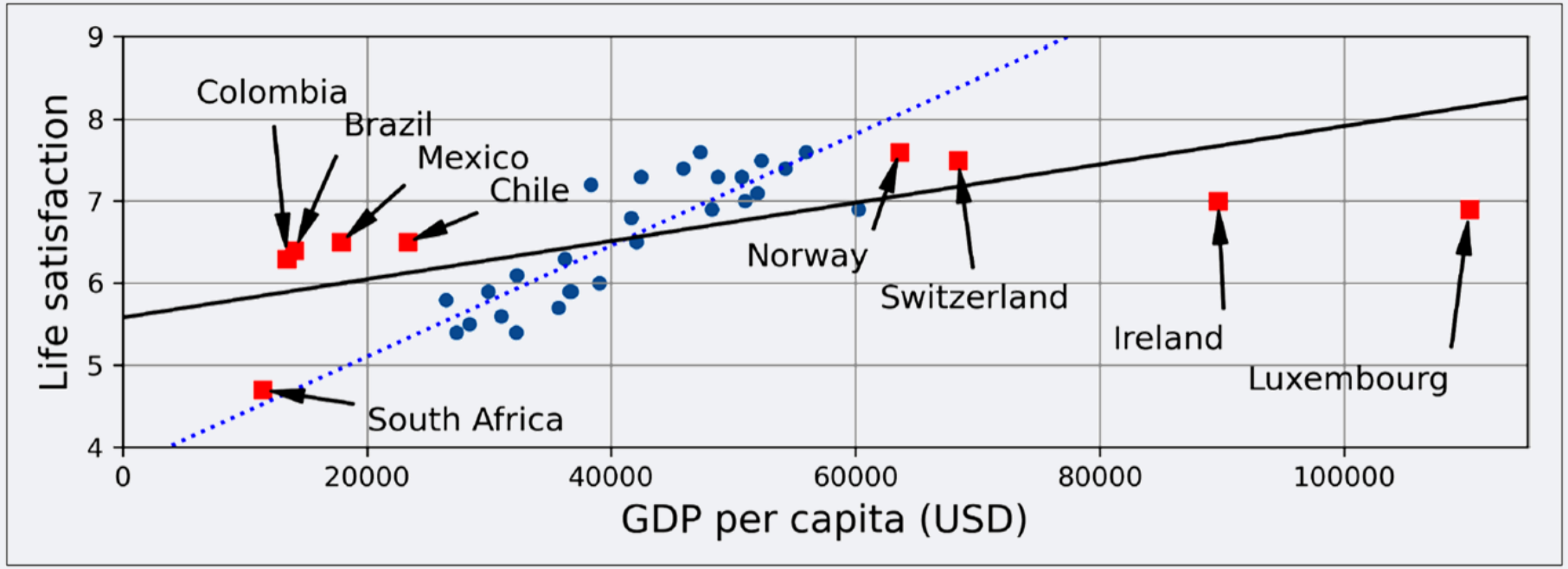
Nonrepresentative Training Data


Y(x)
Valor (R$)
Área (m²)


Y(x)
Valor (R$)
h_θ
$h_\theta = \theta_0 + \theta_1 x$
X
Área (m^2)


Life satisfaction
GDP per capita (USD)
New Zealand
Denmark
Hungary
France
Australia
U.S.
Turkey
9
8
7
6
5
4
25000
30000
35000
40000
45000
50000
55000
60000


Life satisfaction
GDP per capita (USD)
$\theta_0 = 3.75$
$\theta_1 = 6.78 \times 10^{-5}$
9
8
7
6
5
4
25000
30000
35000
40000
45000
50000
55000
60000


Life satisfaction
GDP per capita (USD)
Colombia
Brazil
Mexico
Chile
Norway
Switzerland
Ireland
Luxembourg
South Africa
9
8
7
6
5
4
0
20000
40000
60000
80000
100000

Generalização em aprendizado de máquina é a capacidade essencial de um modelo de performar bem em dados novos e nunca vistos, após ter sido treinado em um conjunto de dados limitado.

Um modelo que generaliza bem não apenas "decorou" os exemplos de treinamento, mas aprendeu os padrões verdadeiros e subjacentes, permitindo-lhe fazer previsões precisas e úteis em situações do mundo real.

Para que um modelo de Machine Learning generalize bem para novos dados, é crucial que seu conjunto de treinamento seja representativo para aquele universo de dados.

Um modelo que previa a satisfação com a vida a partir do PIB: ao omitir países muito ricos e muito pobres, apresentou uma forma linear aparentemente satisfatória mas quando submetido às novas amostras mostrou-se inadequado.

A falta de representatividade pode ocorrer por ruído de amostragem (amostras pequenas) ou por viés de amostragem (método de coleta falho), que é um problema sério mesmo em amostras grandes.

Poor-Quality Data

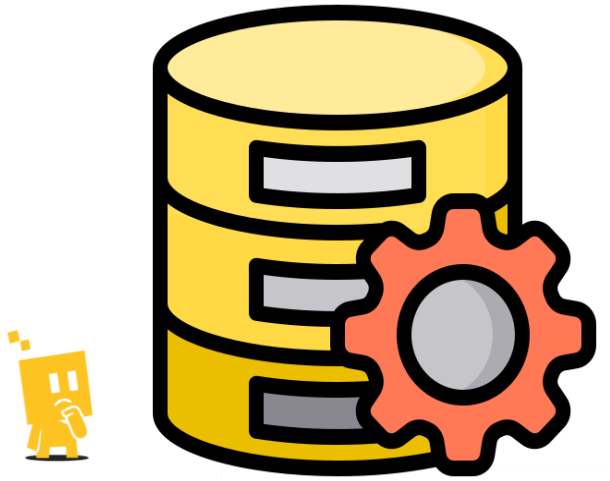
Problema: Dados com erros, outliers, ruído, ausentes...
Solução: Pré-processamento de dados

Irrelevant Features

Um modelo só consegue aprender se os dados de treino contiverem atributos relevantes.

Engenharia de Atributos: É o processo de selecionar e criar o melhor conjunto de atributos (features) para o modelo, sendo uma parte crítica para o sucesso de um projeto de Machine Learning.

Etapas Principais:
Seleção de Atributos: Escolher os atributos mais úteis entre os que já existem.
Extração de Atributos: Combinar atributos existentes para criar um novo mais informativo.
Criação de Atributos: Coletar novos dados para gerar atributos que ainda não existem.

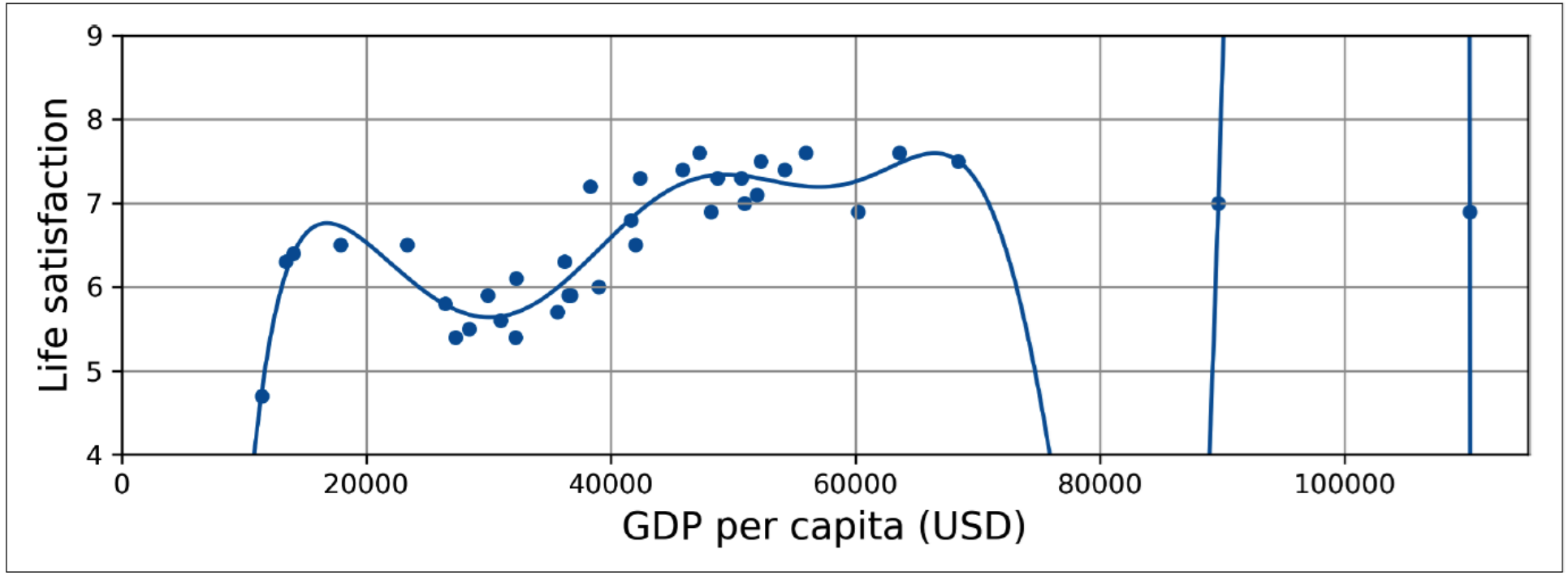


Overfitting the Training Data

Overfitting (Sobreajuste): O Risco de Aprender Demais

Ocorre quando o modelo se ajusta perfeitamente aos dados de treino, mas falha ao fazer previsões com novos dados. Ele **memoriza o ruído** em vez de aprender o padrão real.

Exemplo Prático: O modelo "aprende" que todos os países com a letra "W" no nome têm alta satisfação com a vida (ex: New Zealand, Sweden). Obviamente, essa é uma regra falsa que não se aplicará a outros casos.


Life satisfaction
9
8
7
6
5
4
0
20000
40000
60000
80000
100000
GDP per capita (USD)

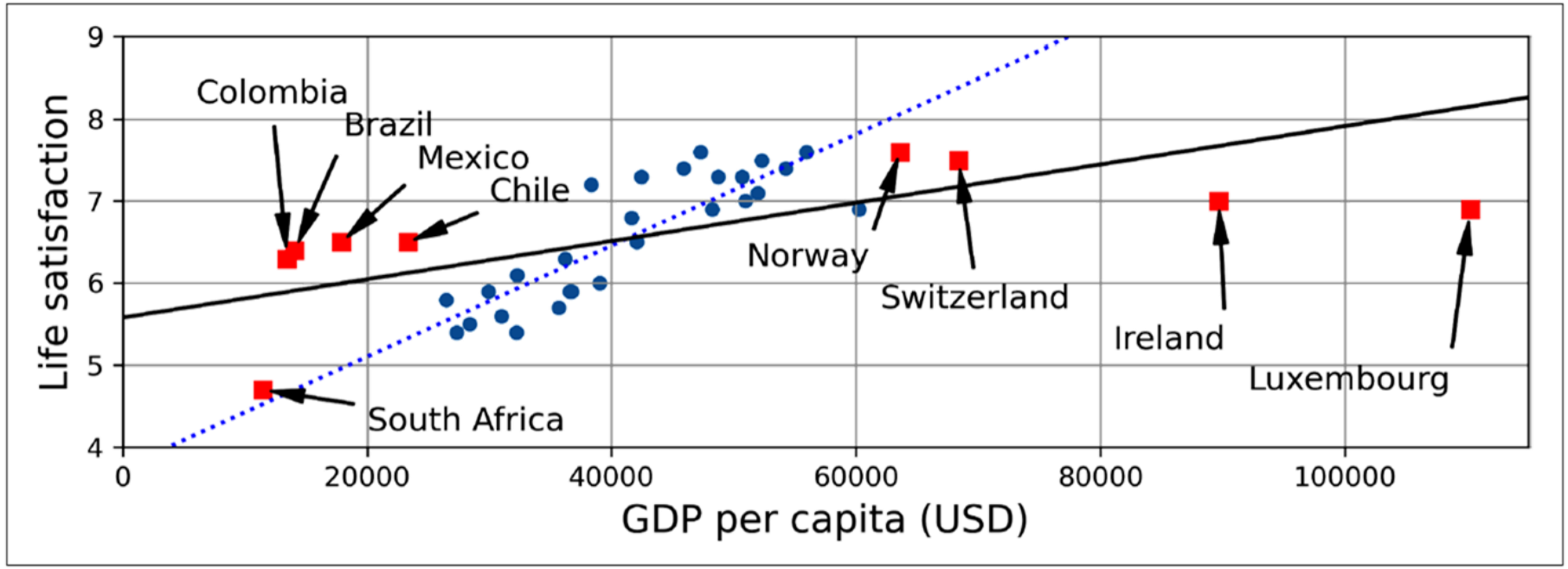
Underfitting - Subajuste dos dados de treinamento

Underfitting (Subajuste): Quando o Modelo é Simples Demais

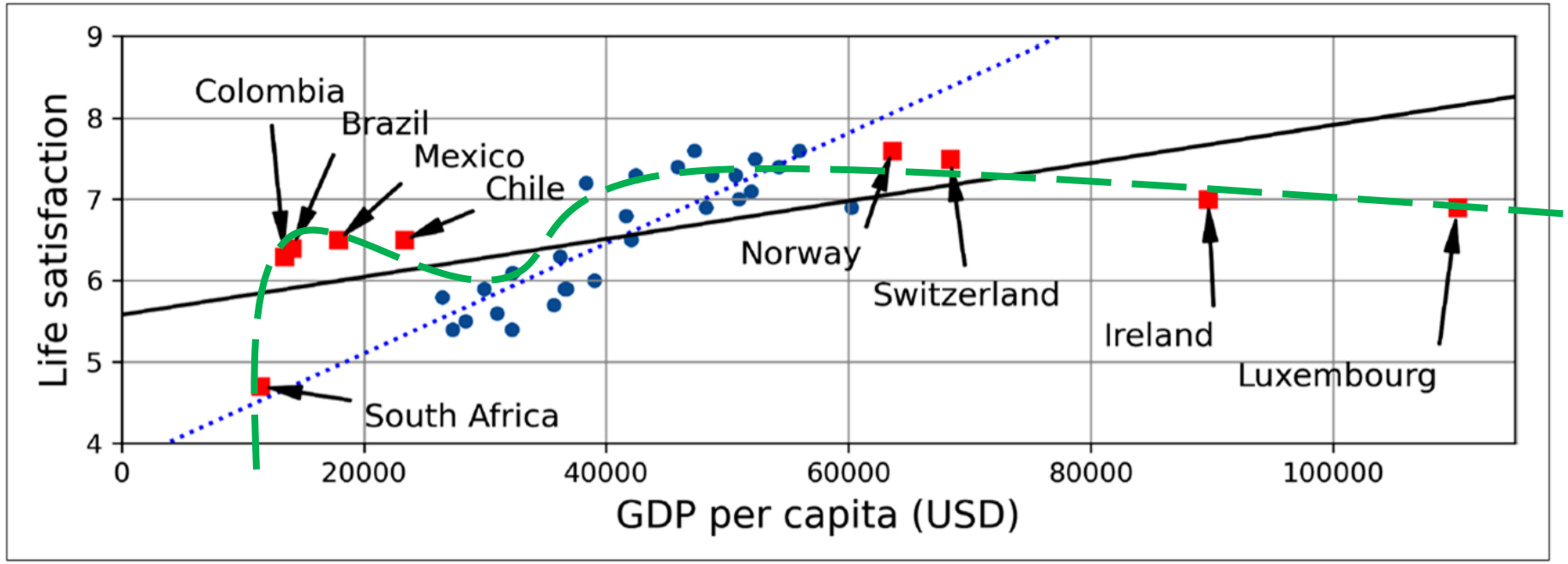
É o oposto de overfitting. Ocorre quando o seu modelo é **muito simples** para aprender a estrutura e os padrões reais presentes nos dados.

Exemplo Prático: Usar um modelo de linha reta (linear) para descrever algo complexo como a satisfação com a vida. A realidade não é uma linha reta.

Regressão Linear não seria uma abordagem suficiente para esse universo de dados



Regressão Linear não seria uma abordagem suficiente para esse universo de dados



Testing and Validating

Hyperparameter Tuning and Model Selection, Data Mismatch

NCIA | FOXCONN FPFtech

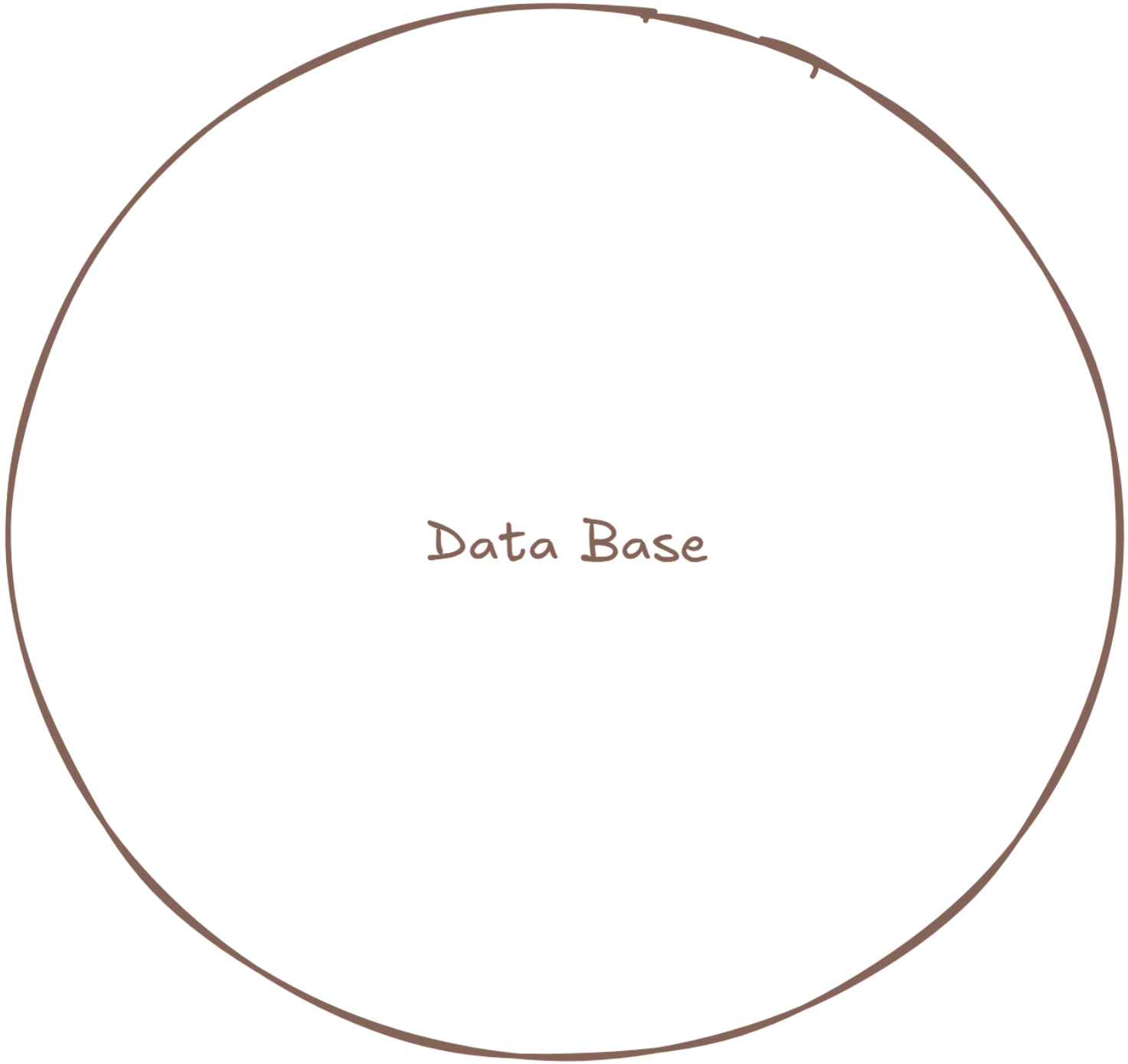
Testando e validando

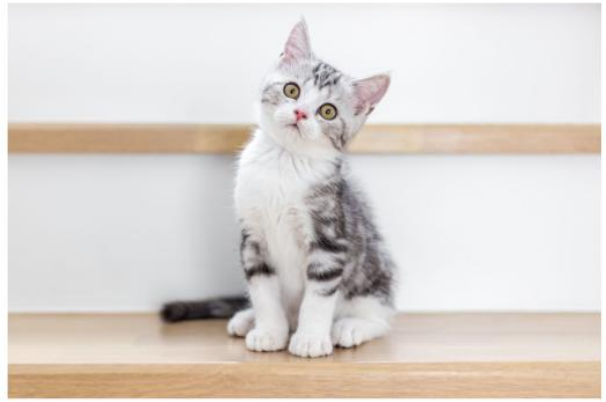
Para saber se um modelo é bom sem arriscar em produção, divida seus dados em conjunto de treino e conjunto de teste.

Treine o modelo com o conjunto de treino; avalie o desempenho com o conjunto de teste.

O erro no teste (erro de generalização) mostra como o modelo se sairá com dados que nunca viu.

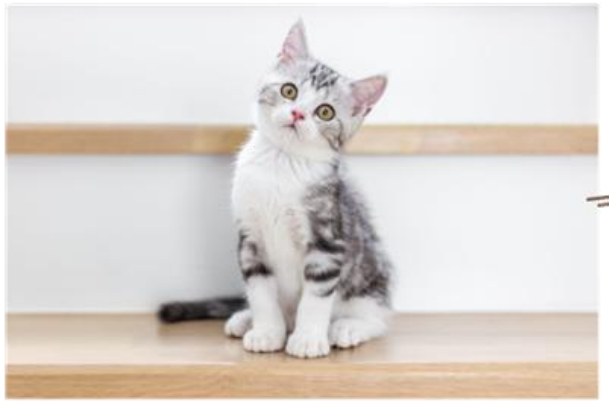
Se o modelo vai bem no treino, mas mal no teste, ele está com **overfitting**.


Data Base



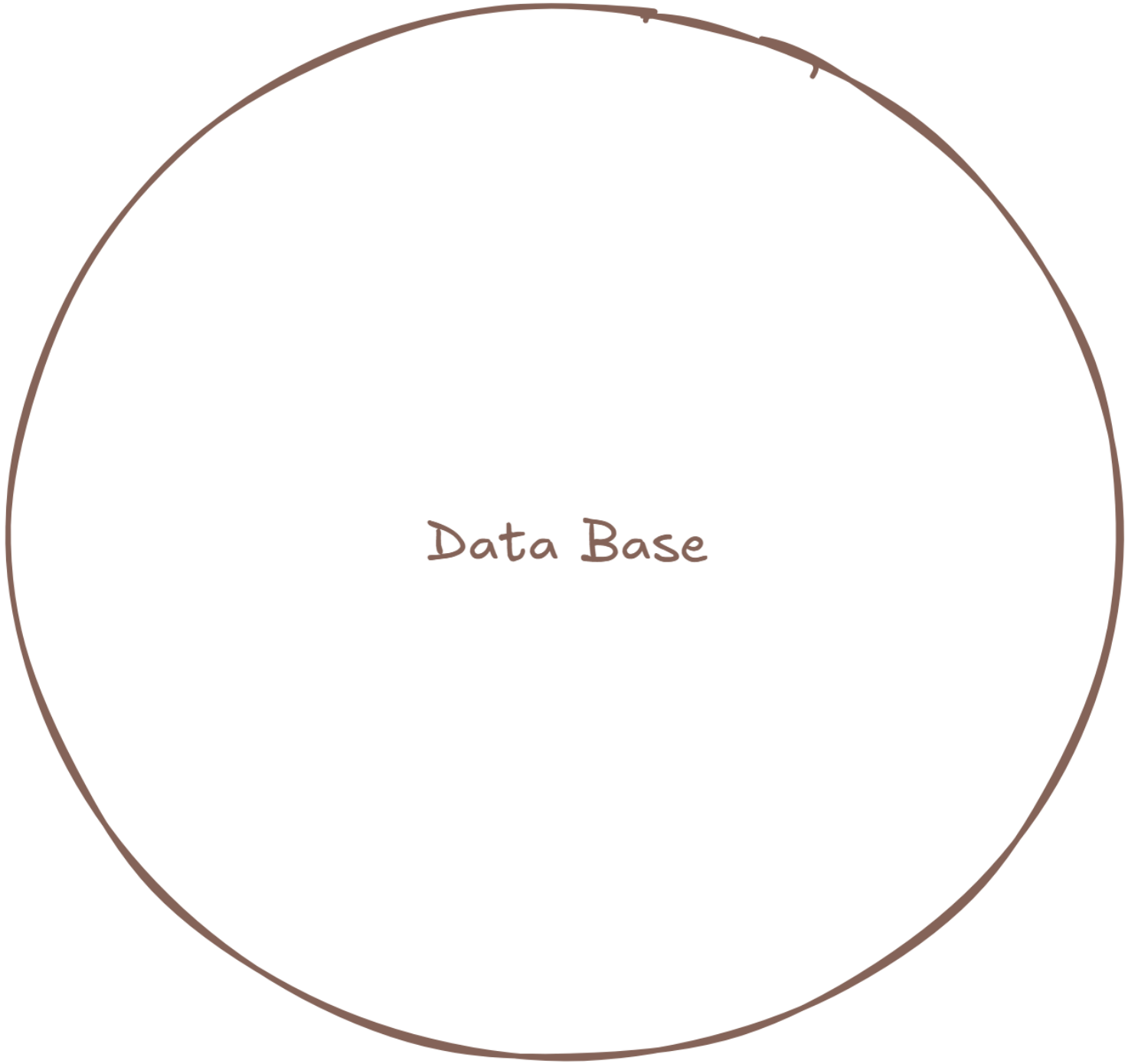
Data Base

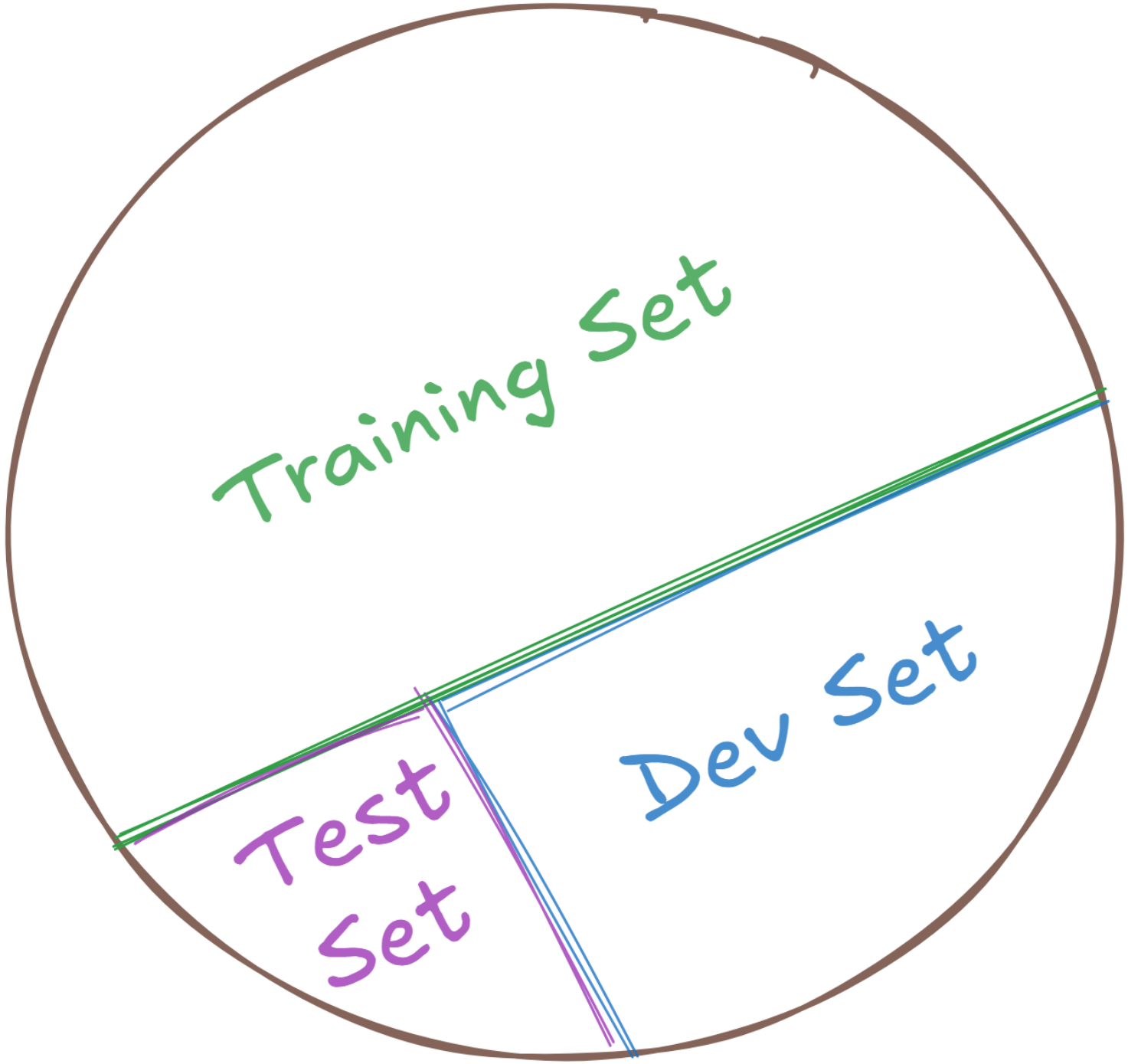
class_index x_center
y_center width height

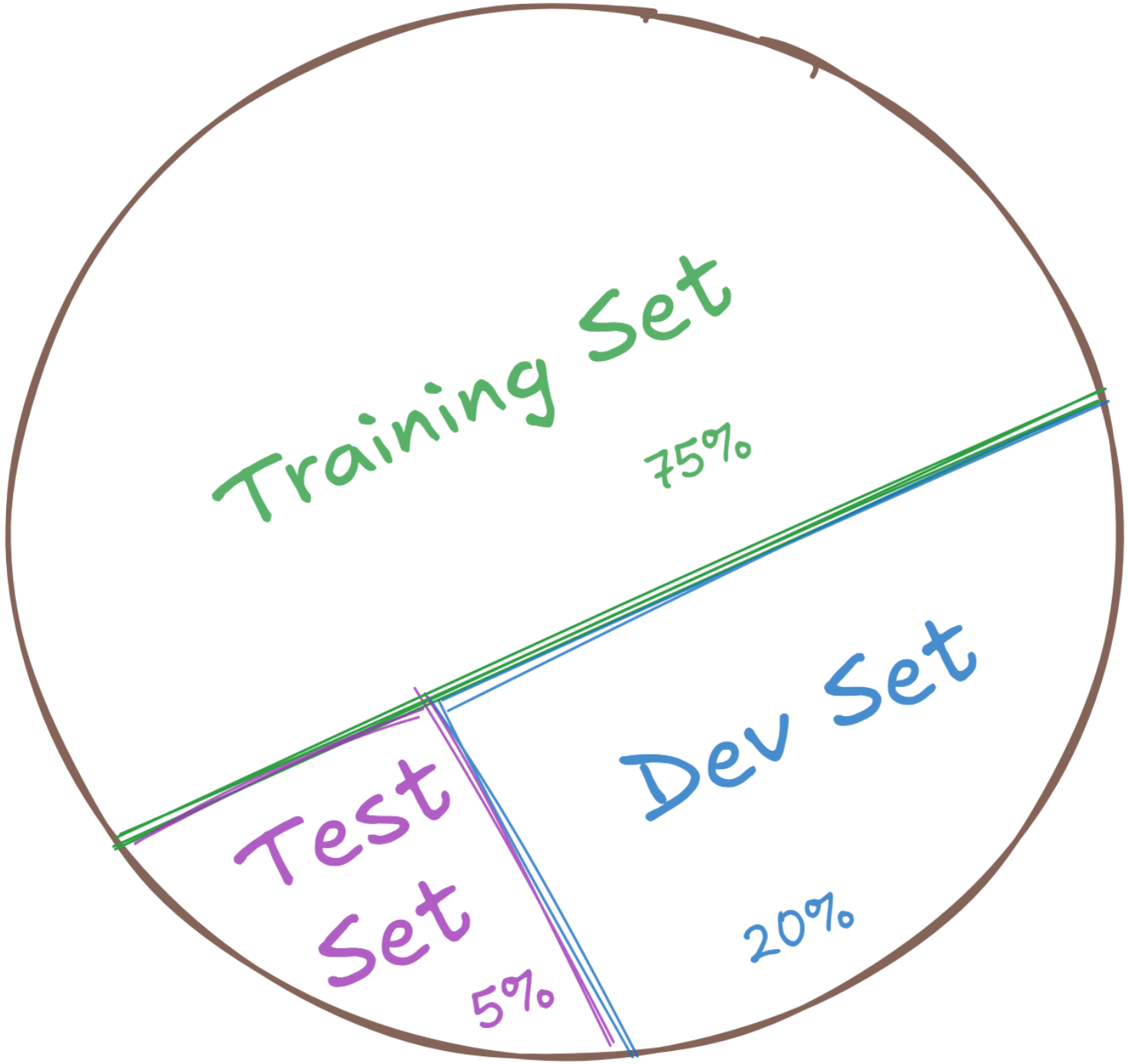


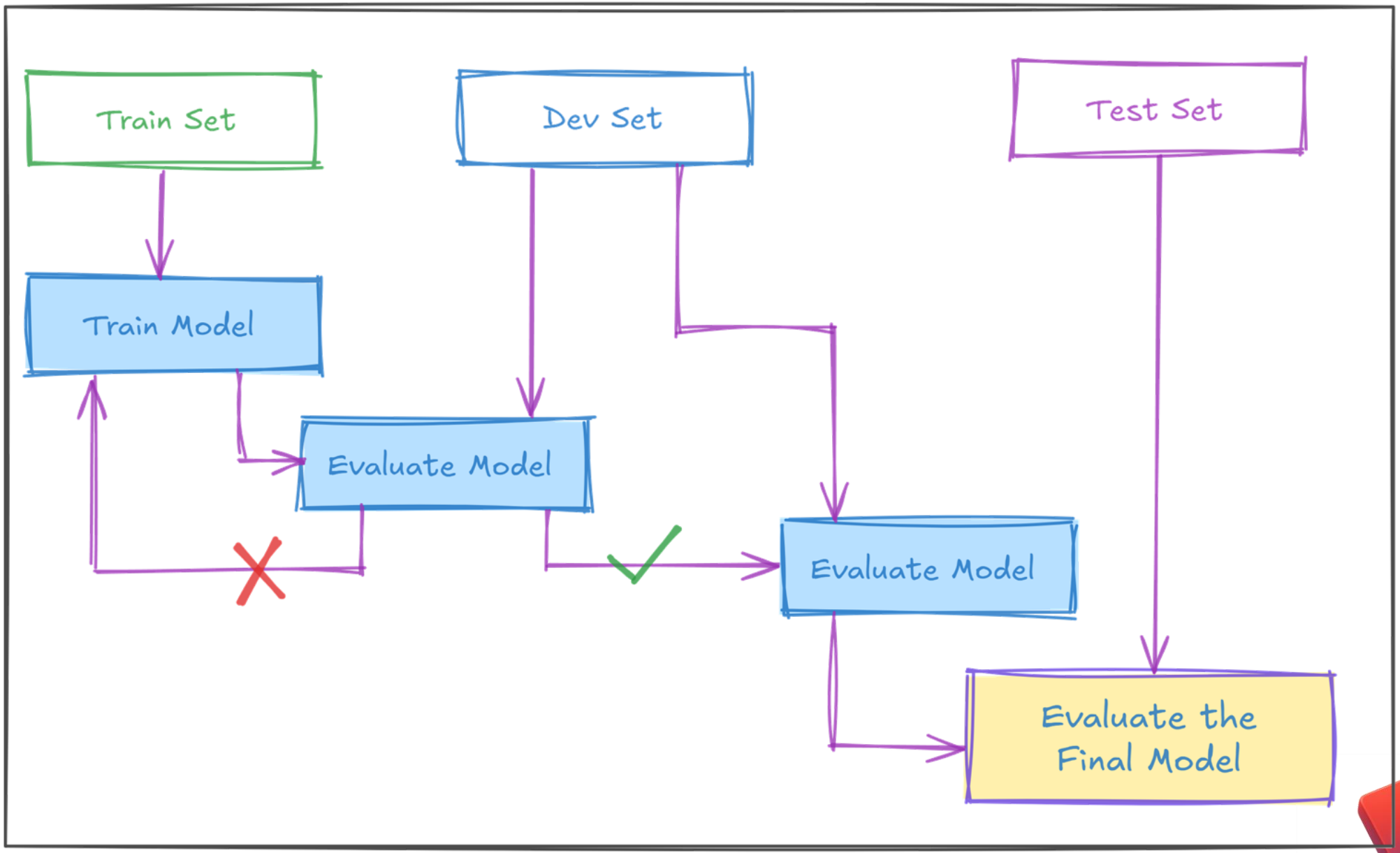
Data Base

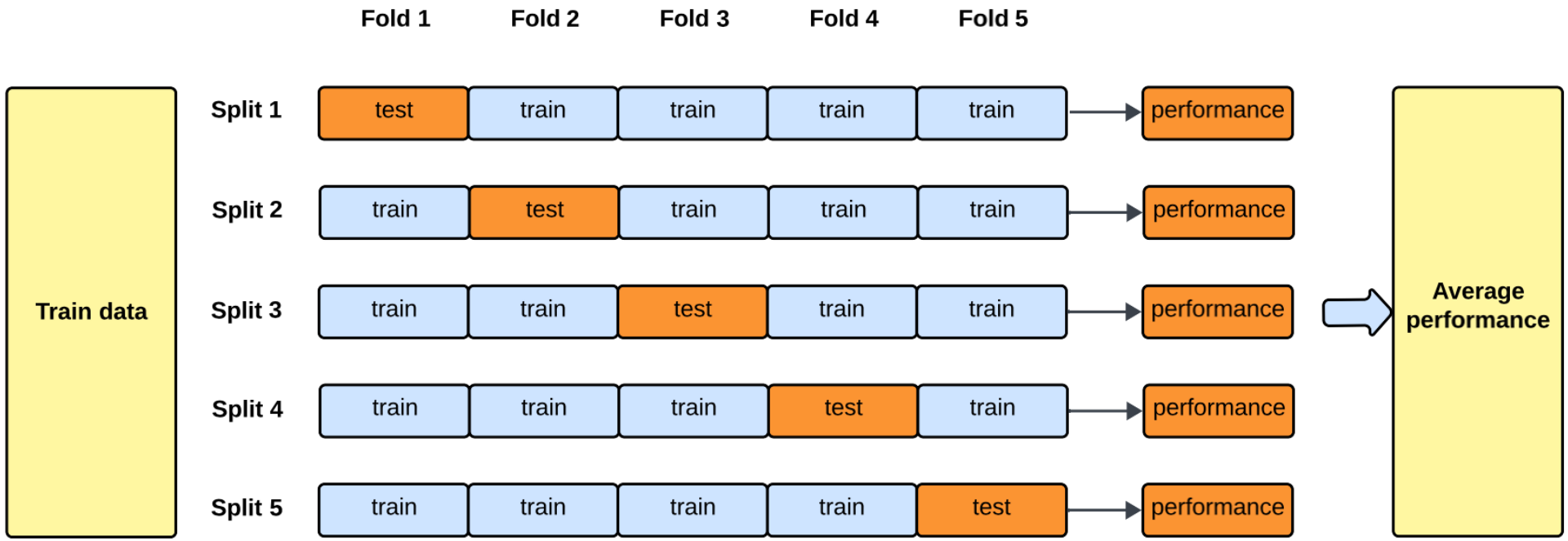
class_index x_center
y_center width height


Data Base


Training Set
Test Set
Dev Set


Training Set
75%
Dev Set
20%
Test Set
5%


Train Set
Dev Set
Test Set
Train Model
Evaluate Model
Evaluate Model
Evaluate the Final Model


Fold 1
Fold 2
Fold 3
Fold 4
Fold 5
Train data
Split 1
Split 2
Split 3
Split 4
Split 5
test
train
performance
Average performance

FPFtech