

EA991 - Laboratório de Aprendizado de Máquina

Fundamentos de aprendizado de
máquina

Prof. Denis G. Fantinato
Prof. Levy Boccato





Introdução e definições

“Machine learning is the science (and art) of programming computers so they can learn from data”.
(Aurélien Géron, 2017)

“Field of study that gives computers the ability to learn without being explicitly programmed”.
(Arthur Samuel, 1959)

- **Ponto central:** *machine learning* (ML) está associado à busca por modelos computacionais que se mostrem capazes de aprimorar sua habilidade em realizar uma determinada tarefa à medida que são expostos aos dados disponíveis.
 - Logo, são essencialmente modelos guiados por dados, ou *data-driven*.



Fatores que alavancaram ML

- O volume de dados disponíveis, e das mais variadas naturezas, como texto, imagem e áudio, atingiu escalas sem precedentes.
- Maior disponibilidade de infraestrutura computacional para armazenamento e processamento: GPUs (*graphics processing units*), TPUs (*tensor processing units*), *clusters*, *cloud computing*.
- Boas estratégias e técnicas para realizar o treinamento, i.e., o ajuste dos parâmetros do modelo escolhido.
- Existência de *frameworks* para o desenvolvimento eficiente de soluções de ML e *deep learning* (DL).





Onde aplicar ML?

- Problemas em que as abordagens tradicionais falham ou se tornam inviáveis (e.g., devido à complexidade de se definir as regras de tomada de decisão).
- Ambientes em constante mudança: os modelos podem continuar aprendendo e se adaptando aos novos desafios impostos pela dinâmica do ambiente.
- Quando temos que extrair informações a partir de vastas coleções de dados.



Em que consiste o aprendizado?

1

Estrutura

Está relacionada a como pode ser caracterizado o mapeamento entrada-saída gerado pelo modelo.

2

Critério

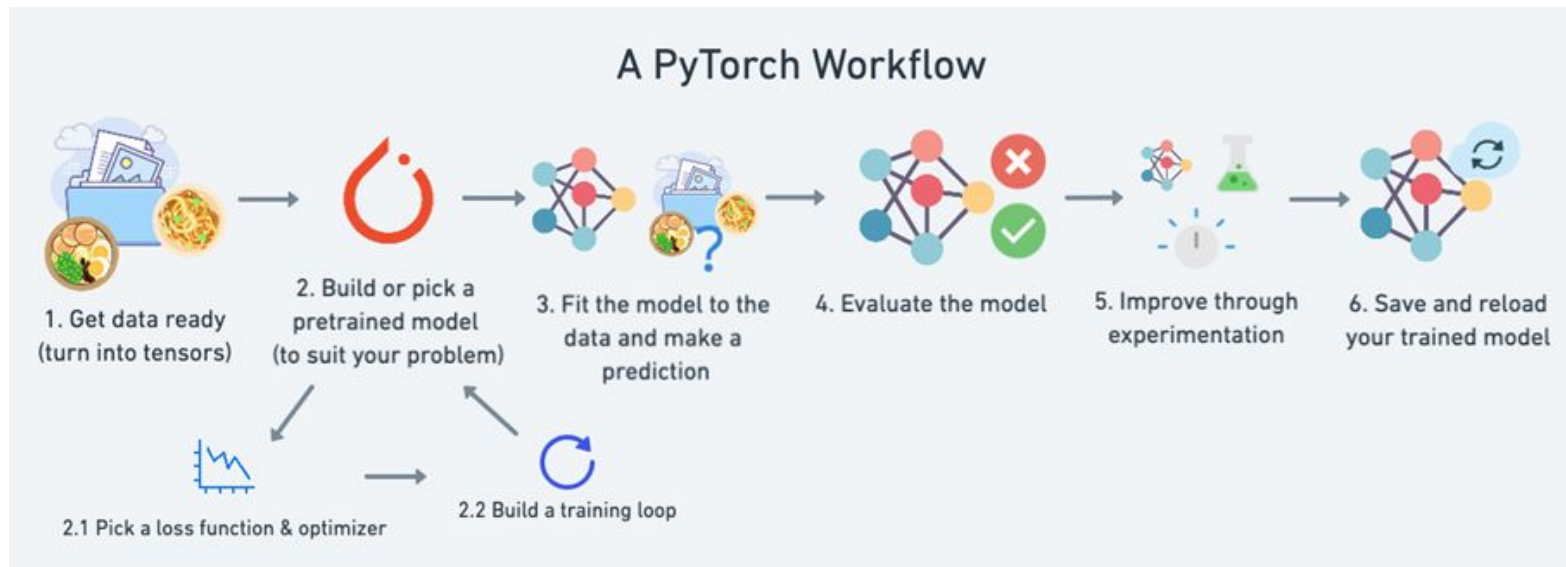
Expressa o que se deseja atingir durante o aprendizado, revelando o grau de sucesso alcançado pelo modelo na realização da tarefa.

3

Algoritmo

Responsável por resolver o problema de otimização atrelado ao critério, isto é, por encontrar os valores ótimos dos parâmetros do modelo.

Em que consiste o aprendizado?



Fluxo de trabalho típico para aplicação de ML / DL a problemas práticos.

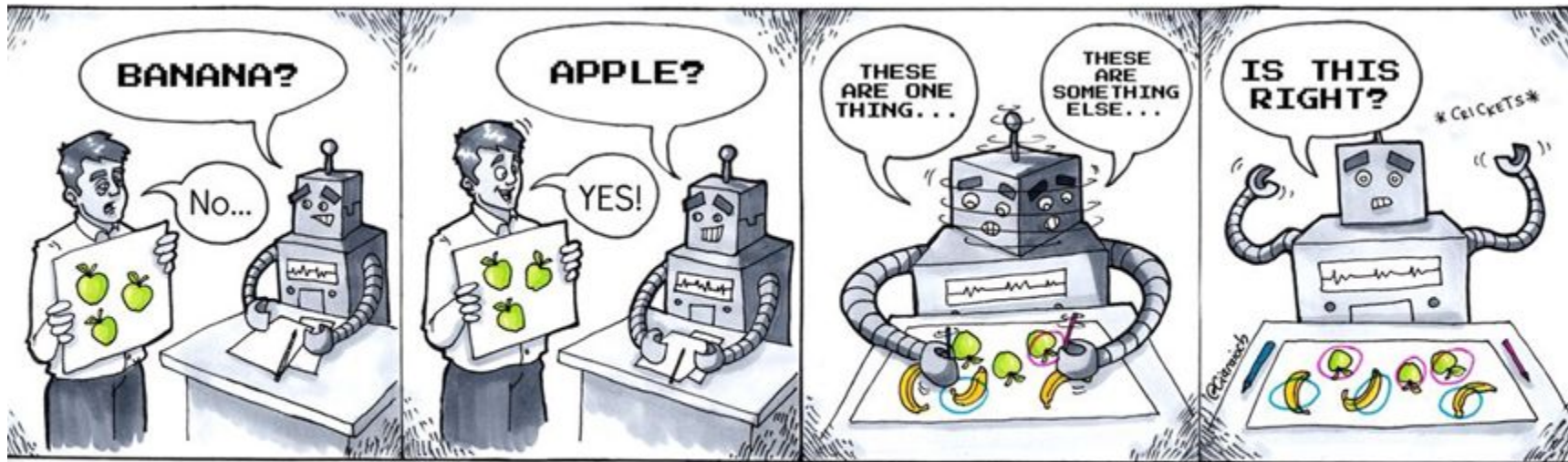
https://www.learnpytorch.io/01_pytorch_workflow/



Paradigmas de aprendizado

- **Supervisionado:** para cada amostra (ou padrão) de treinamento disponível, existe uma resposta desejada que é conhecida. Neste caso, dizemos que os dados são rotulados.
 - Sendo assim, é possível definir uma função de perda (*loss function*) que expresse uma noção de erro entre a resposta que o modelo gerou e a resposta esperada para cada amostra.
- **Não-supervisionado:** não há uma saída desejada associada a cada padrão, de modo que os dados são não-rotulados.
 - Neste cenário, desejamos que o modelo seja capaz de capturar, representar ou evidenciar propriedades ou regularidades existentes no conjunto de dados.
- **Por reforço:** embora não seja conhecida a saída correta para cada amostra, como ocorre no caso supervisionado, temos uma realimentação sobre a qualidade da saída gerada pelo modelo na forma de um sinal de recompensa ou punição.

Paradigmas de aprendizaje



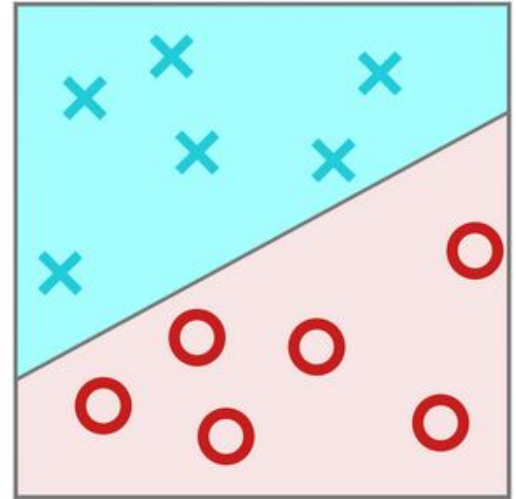
Supervised Learning

Unsupervised Learning

Tarefas de aprendizado supervisionado

- **Classificação:**

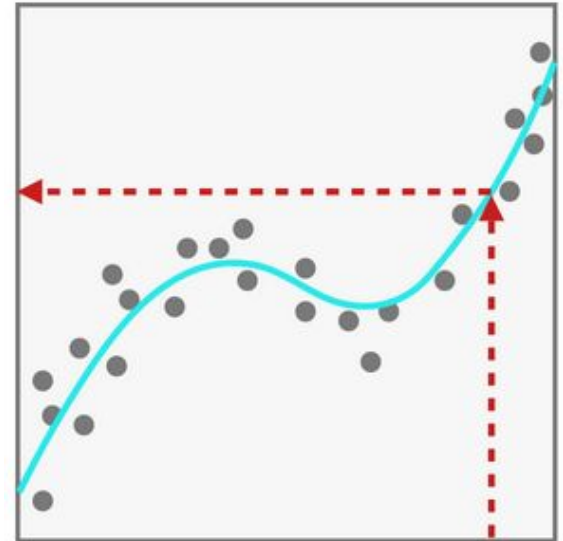
- É a atividade de designar objetos a algumas classes ou categorias pré-existentes.
- Para cada amostra de entrada, um classificador deve produzir como resposta o identificador da classe à qual ela pertence.
 - Ou seja, ele gera um particionamento do espaço dos dados em regiões de decisão para cada classe.



Tarefas de aprendizado supervisionado

- **Regressão:**

- Consiste em estimar a relação matemática que há entre uma variável dependente e um conjunto de variáveis independentes.
- Em outras palavras, envolve a busca por um mecanismo de estimação numérica do valor de uma informação a partir de um conjunto de variáveis de entrada.






Aprendizado supervisionado





- **Missão:** dado um conjunto de amostras rotuladas, desejamos projetar um modelo que consiga aprender o mapeamento correto das entradas nas respectivas saídas.
- **Alvo:** ser capaz de estimar ou prever a saída com o menor erro possível para **qualquer** amostra de entrada.
 - O modelo deve ser competente em sua tarefa mesmo quando lidar com amostras que não havia visto durante o seu treinamento.
 - Ou seja, ele deve **generalizar** bem.

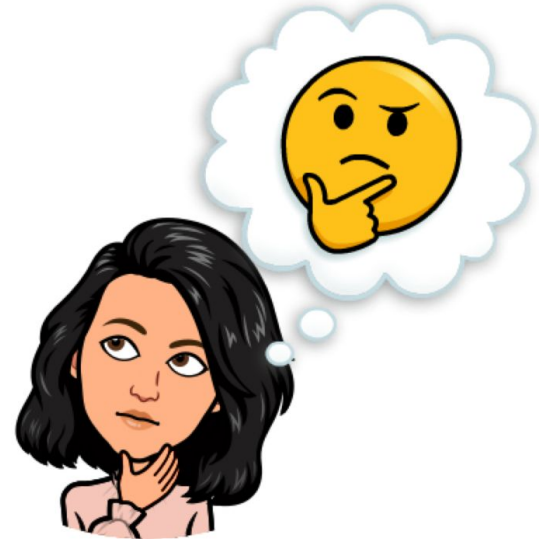
Generalização

Dados de
treinamento

	GATO
	GATO
	CÃO
	CÃO

Novas amostras para
predição

1		?
2		?
3		?
4		?





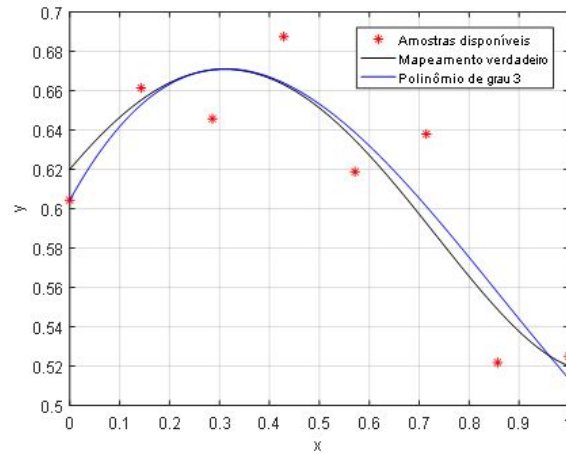
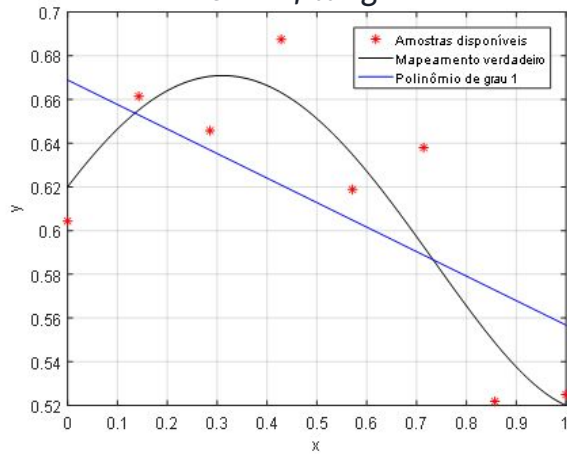
Generalização

- Situações indesejadas:

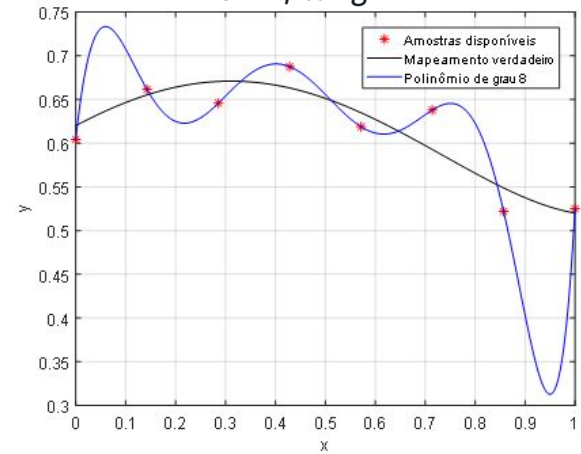
- Sobreajuste (*overfitting*): a aproximação gerada pelo modelo se contorceu de forma excessiva a fim de reduzir ao máximo o erro junto aos dados de treinamento. Contudo, embora o erro de treinamento seja baixo, o modelo comete erros significativos quando recebe amostras inéditas na entrada.
- Sub-ajuste (*underfitting*): o modelo não é capaz de aprender o mapeamento verdadeiro, não atingindo bom desempenho nem mesmo nos dados utilizados no treinamento. Isto pode ocorrer pelo fato de o grau de flexibilidade do modelo ser insuficiente diante da complexidade do mapeamento a ser aproximado, ou, também, por problemas de convergência do processo de treinamento.

Generalização

Underfitting



Overfitting





Medindo a capacidade de generalização

- **Procedimento:** parte dos dados disponíveis é reservada antes do treinamento e não participa do ajuste do modelo, sendo utilizada posteriormente para avaliar o seu desempenho justamente diante de amostras inéditas - **Conjunto de teste**.
- **Etapa de teste:** trata-se de uma “simulação” do uso de um modelo pronto na prática, como se fosse um produto (e.g., um aplicativo) lançado no mercado que começasse a ser utilizado por usuários com os seus próprios dados (portanto, inéditos).
- **Cuidado:** decisões a respeito do modelo (por exemplo, os seus (hiper)parâmetros) e/ou do processo de treinamento não podem se basear no conjunto de teste, pois isso geraria um viés.

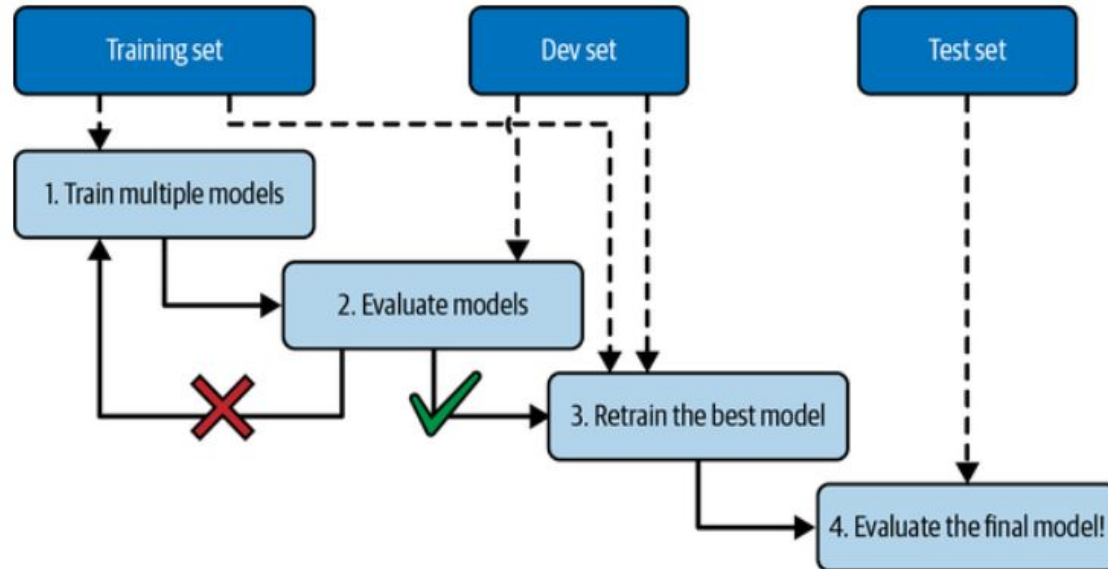


Validação cruzada

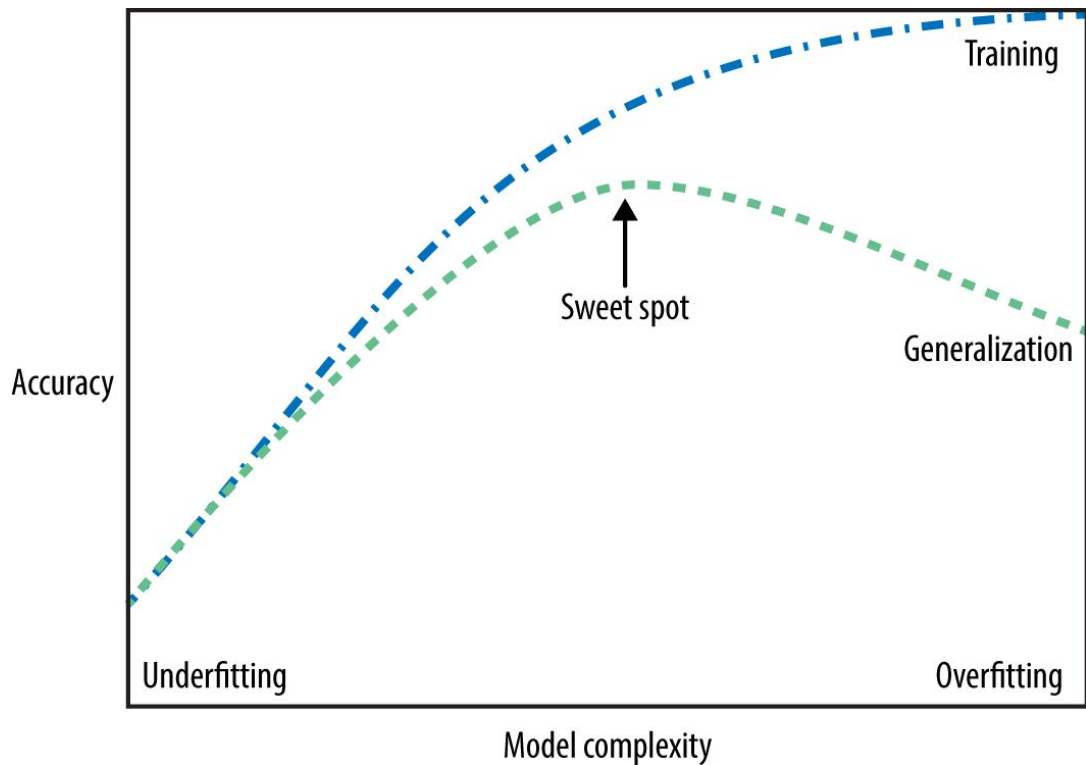
- Consiste em dividir o conjunto de amostras originalmente separado para treinamento em dois:
 - O primeiro conjunto será efetivamente empregado no ajuste dos parâmetros do modelo – conjunto de treinamento.
 - O segundo conjunto será utilizado para monitorar a capacidade de generalização do modelo – **conjunto de validação**.
- Durante o treinamento do modelo, monitora-se também o seu desempenho junto aos dados de validação (por exemplo, ao longo das iterações/épocas, no caso de processos iterativos, ou, então, em função de escolhas possíveis de hiperparâmetros).
- **Premissa:** observar o desempenho do modelo junto aos dados de validação fornece um indicativo antecipado de como ele se comporta quando exposto a amostras não vistas no treinamento. Em outras palavras, o desempenho de validação é interpretado como uma estimativa do erro de generalização.

Validação cruzada

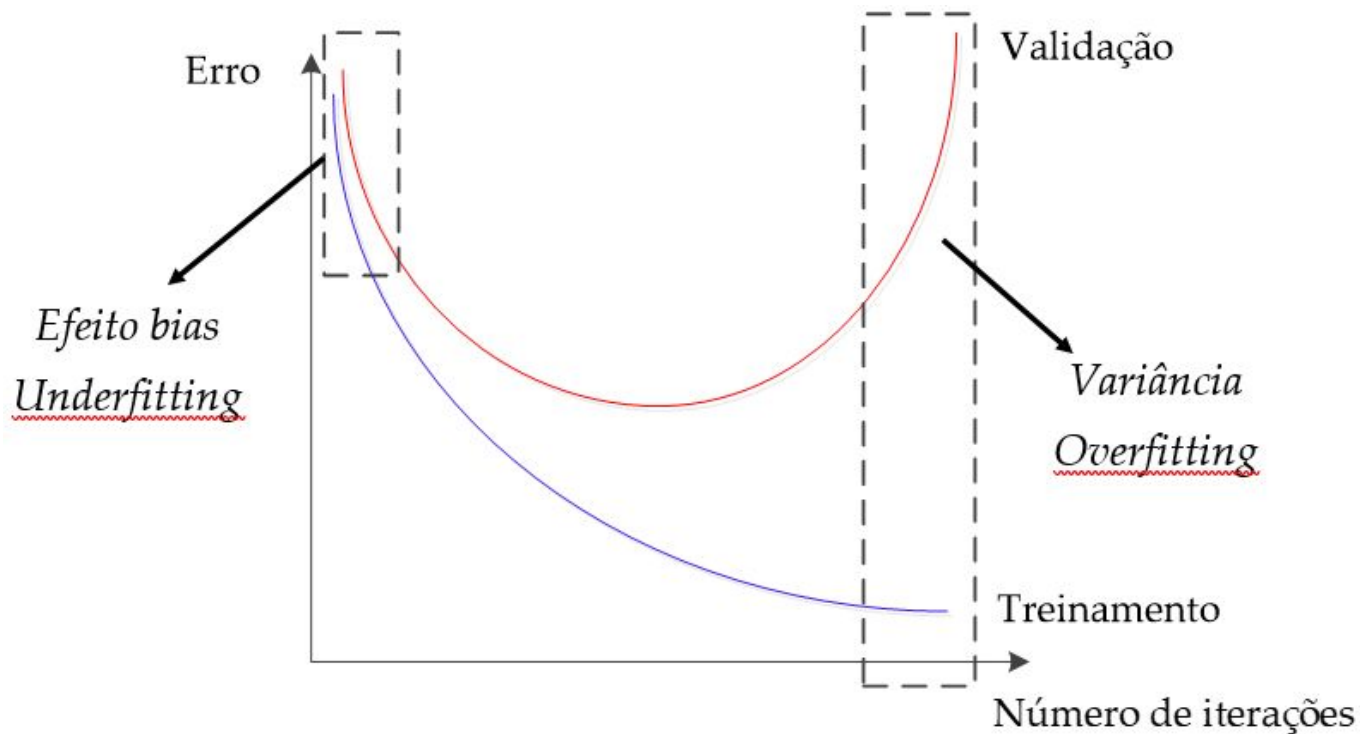
- Procedimento com *holdout*



Dilema viés-variância



Dilema viés-variância



Aplicação de aprendizado supervisionado

What question do I want to answer?



Formulation to supervised machine learning problem



7. Test data predictions, visualizations, possible deployment



6. Evaluation, model selection



5. Feature selection, Model building

Supervised ML process

1. Data collection



2. Data cleaning, splitting



3. Data exploration (EDA)



4. Preprocessing, Feature engineering





Desafios de ML / DL

- Interpretabilidade: os padrões aprendidos por um modelo muitas vezes não são interpretáveis. Além disso, o modo pelo qual o modelo toma as suas decisões nem sempre é perfeitamente explicável para o ser humano.
- “Fome” por dados: em cenários com escassez de amostras, explorar o potencial das abordagens *data-driven* pode ser difícil.
- Representatividade dos dados: para que seja possível generalizar bem, é crucial que os dados de treinamento sejam representativos em relação ao universo de padrões que podem se manifestar na entrada.
 - Em certo sentido, os conjuntos de treinamento, validação e teste devem ser compatíveis.



Desafios de ML / DL

- Qualidade dos dados: se as amostras estiverem cheias de erros e ruídos (e.g., devido a instrumentos de medição de baixa qualidade), será mais difícil para o modelo aprender a detectar os padrões subjacentes.
 - **Exemplos:**
 - Se algumas amostras claramente são *outliers*, pode ser benéfico simplesmente descartá-las ou tentar ajustar os erros manualmente.
 - Se algumas amostras contêm atributos faltantes (e.g., 5% dos clientes não informaram a idade), é preciso decidir se aquele atributo será ignorado, ou se essas instâncias serão ignoradas, ou se tais lacunas serão preenchidas por algum esquema de imputação (e.g., com a mediana das idades), ou, ainda, se duas versões de modelo serão criadas, uma usando o atributo, outra sem aquele atributo.



Desafios de ML / DL

- Qualidade dos atributos: um modelo será capaz de aprender bem se os dados de treinamento tiverem atributos suficientemente relevantes e não muitos atributos irrelevantes.
 - Uma parte crítica de ML consiste em obter um conjunto de bons atributos para descrever os dados e guiar o treinamento. Essa busca compreende a etapa de engenharia de atributos (*feature engineering*), e envolve:
 - Seleção de características: dentre os atributos disponíveis, selecionar apenas os que se mostram mais úteis para o treinamento.
 - Extração de características: aplicar alguma transformação sobre os dados originais a fim de obter atributos mais informativos e úteis para a solução do problema.