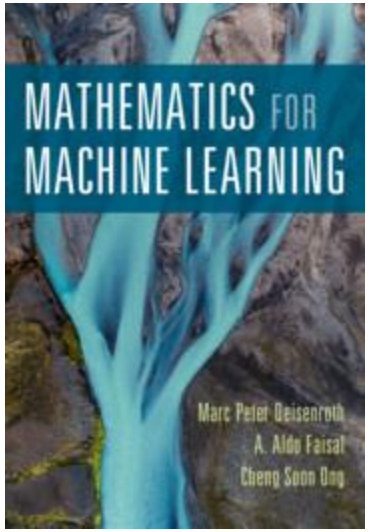
PROGRAMA DE PÓS-GRADUÇÃO EM CIÊNCIA DA COMPUTAÇÃO
TÓPICOS ESPECIAIS EM FUNDAMENTOS DE COMPUTAÇÃO – MATEMÁTICA E ESTATÍSTICA PARA CIÊNCIA DE DADOS
Prof. Dr. Rommel Melgaço Barbosa

Seminários

Quando os modelos encontram os dados



André Riccioppo

Gabriel Almeida

Hudson Romualdo

Junho/2024

UFG
UNIVERSIDADE FEDERAL DE GOIÁS

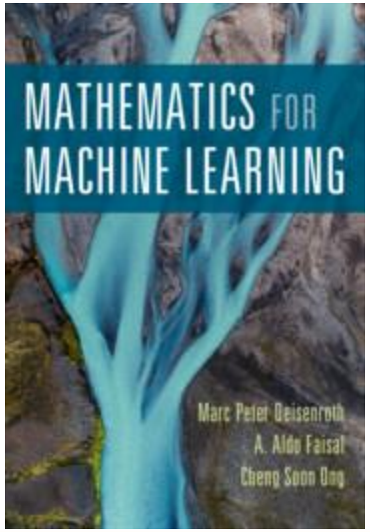
Introdução

Capítulo 8:

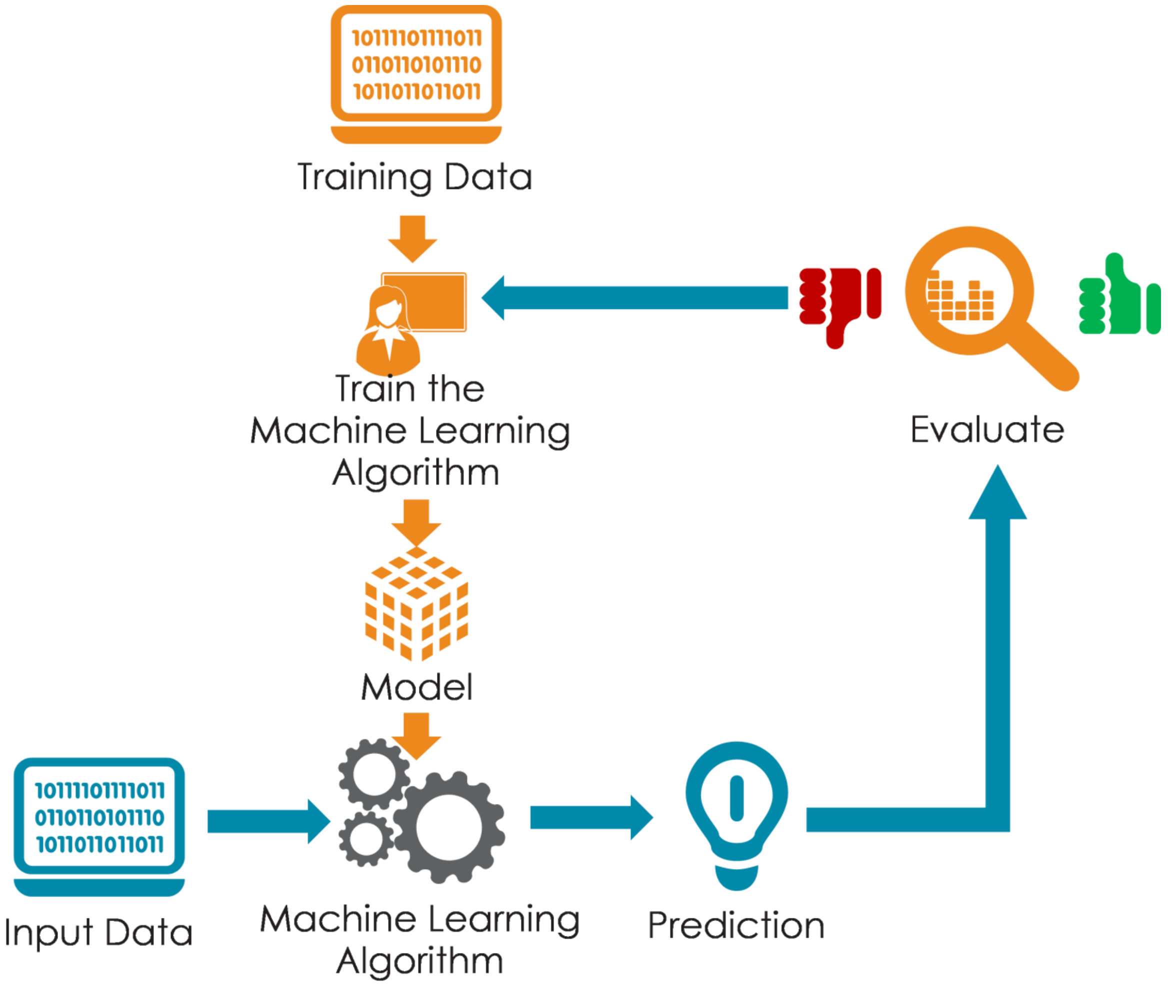
- Dados
- Modelos
- Aprendizado

Preparação para os capítulos seguintes:

- Regressão (Capítulo 9)
- Redução de Dimensionalidade (Capítulo 10)
- Estimação de Densidade (Capítulo 11)
- Classificação (Capítulo 12)



Dados, modelos e aprendizagem



Dados, modelos e aprendizagem

Name	Gender	Degree	Postcode	Age	Annual salary
Aditya	M	MSc	W21BG	36	89563
Bob	M	PhD	EC1A1BA	47	123543
Chloé	F	BEcon	SW1A1BH	26	23989
Daisuke	M	BSc	SE207AT	68	138769
Elisabeth	F	MBA	SE10AA	33	113888

Tabela 1 – Dados de recursos humanos que não estão em um formato numérico

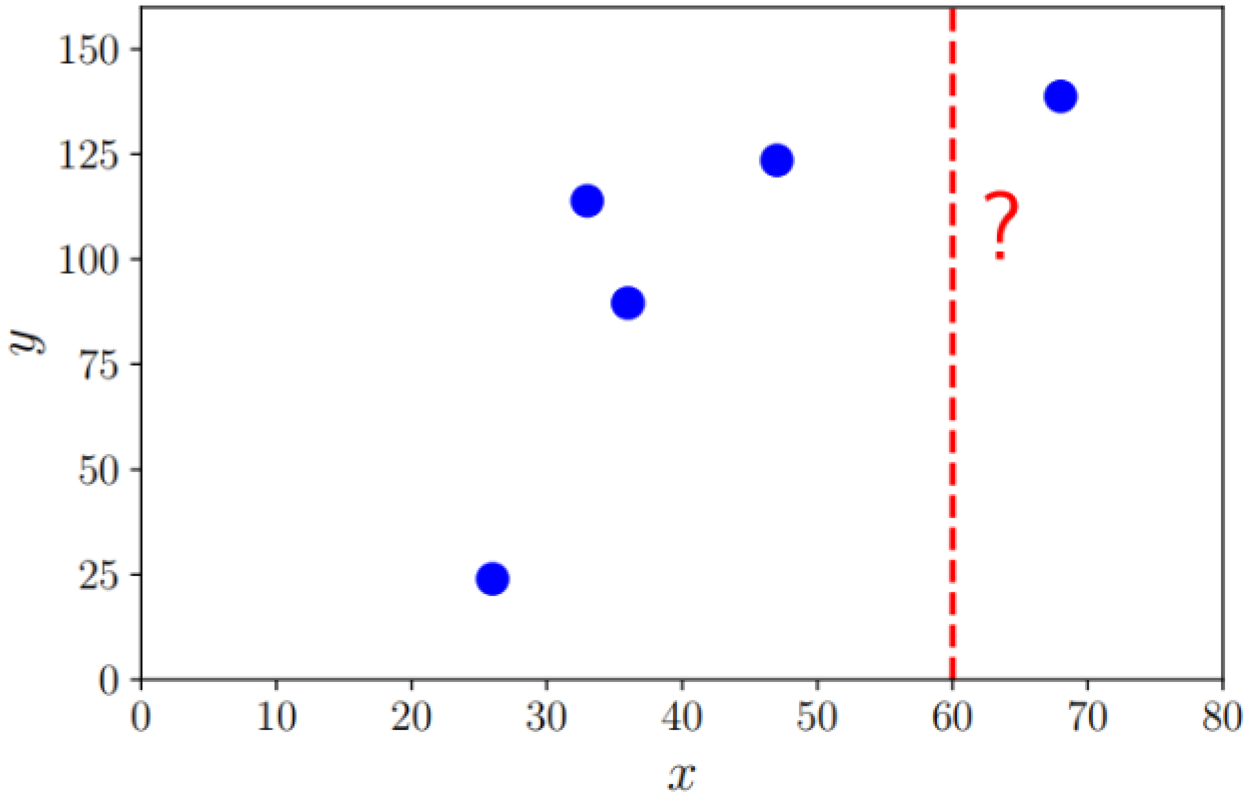
Dados como Vetores

Gender ID	Degree	Latitude (in degrees)	Longitude (in degrees)	Age	Annual Salary (in thousands)
-1	2	51.5073	0.1290	36	89.563
-1	3	51.5074	0.1275	47	123.543
+1	1	51.5071	0.1278	26	23.989
-1	1	51.5075	0.1281	68	138.769
+1	2	51.5074	0.1278	33	113.888

Tabela 2 – Dados de recursos humanos que estão em um formato numérico

Gender ID	Degree	Latitude (in degrees)	Longitude (in degrees)	Age	Annual Salary (in thousands)
-1	2	51.5073	0.1290	36	89.563
-1	3	51.5074	0.1275	47	123.543
+1	1	51.5071	0.1278	26	23.989
-1	1	51.5075	0.1281	68	138.769
+1	2	51.5074	0.1278	33	113.888

- Conjunto de dados: N
- Exemplos de dados: $n = 1, \dots, N$.
- Cada exemplo (*datapoint)* é um vetor: $\boldsymbol{x}_n$.
- Cada característica indexamos: $d = 1, \dots, D$.

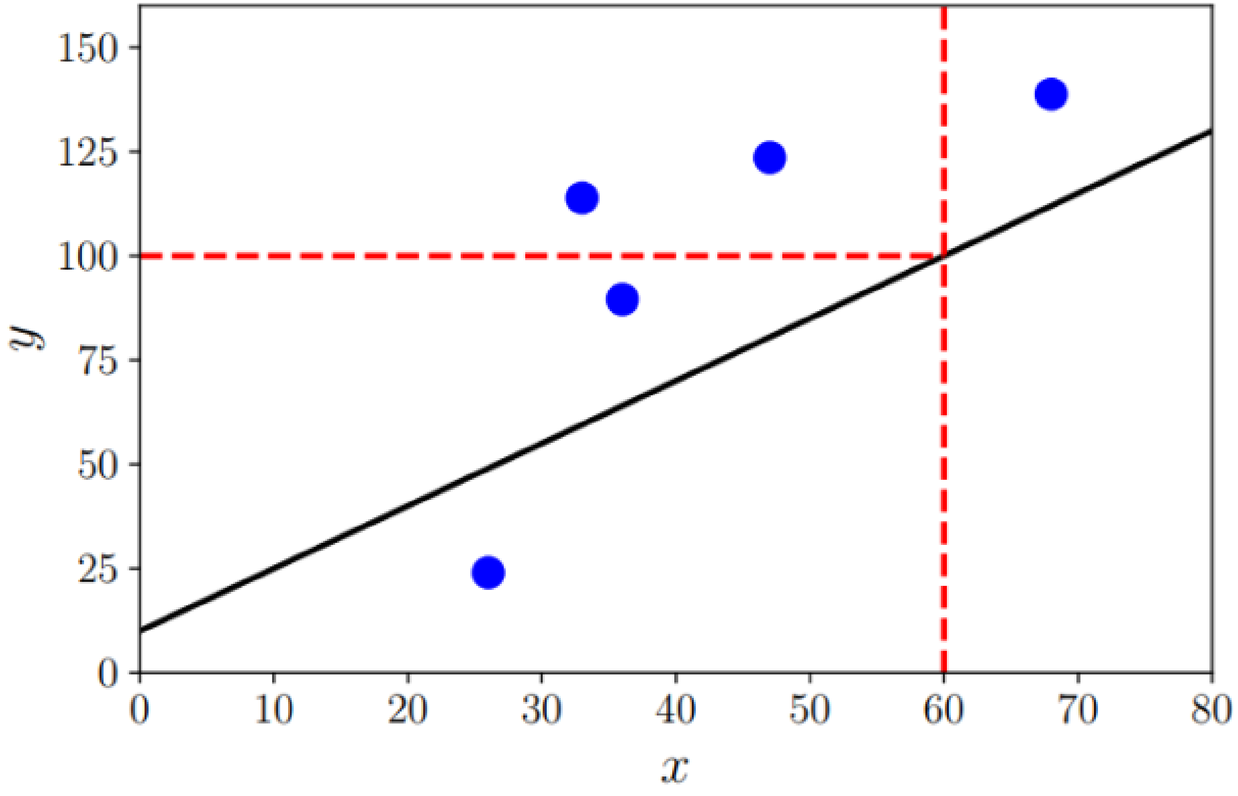


- *Target*: $\boldsymbol{y}_n$
- *Input*: $\boldsymbol{x}_n$.
- *Dataset*: $(\boldsymbol{x}_1, \boldsymbol{y}_1), \ldots, (\boldsymbol{x}_n, \boldsymbol{y}_n), \ldots (\boldsymbol{x}_N, \boldsymbol{y}_N)$
- Conjunto de *datapoints* $\boldsymbol{x}_n, \ldots \boldsymbol{x}_N$: $\boldsymbol{X} \in \mathbb{R}^{N \times D}$.

Modelos como funções

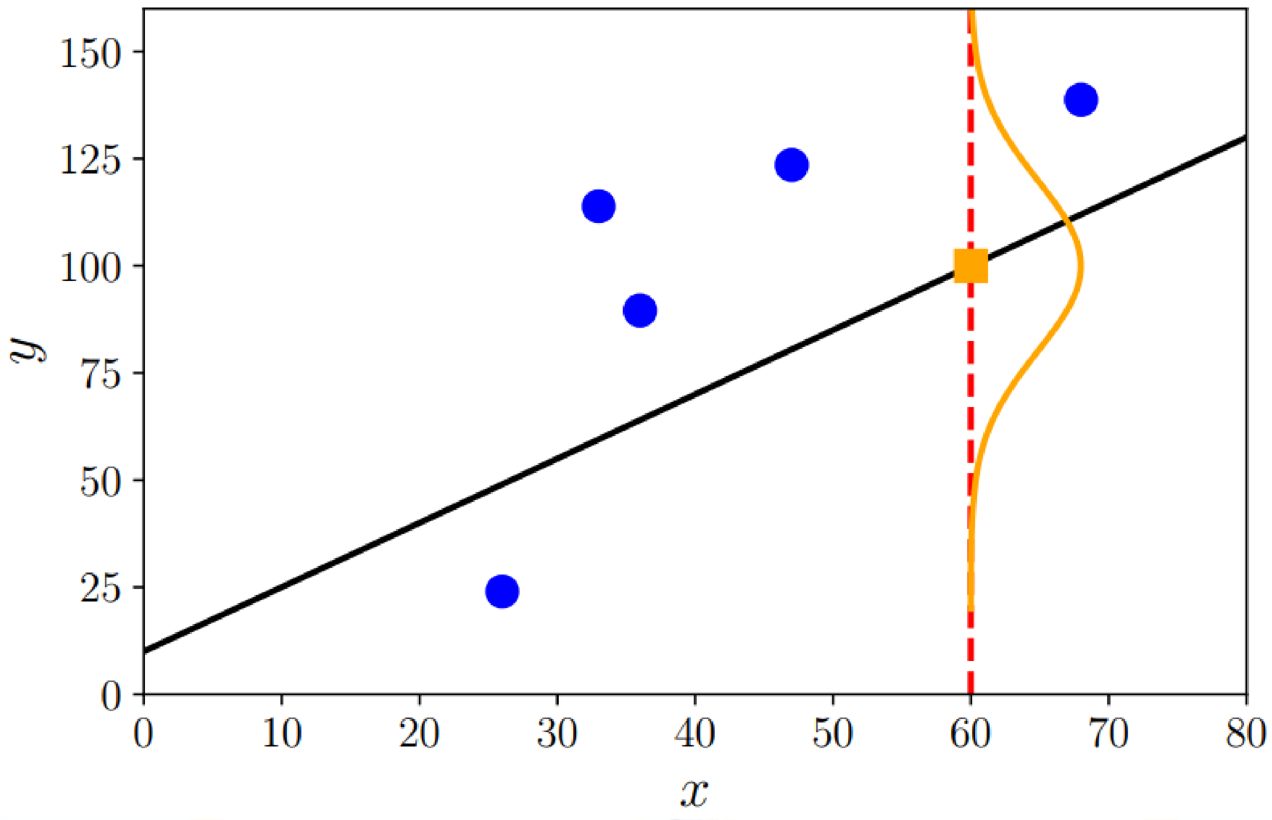
Um preditor (modelo treinado) é uma função quando, ao receber uma determinada entrada (no nosso caso, um vetor de características), produz um saída.

$$f : \mathbb{R}^D \to \mathbb{R}.$$



Modelos como distribuições de probabilidade

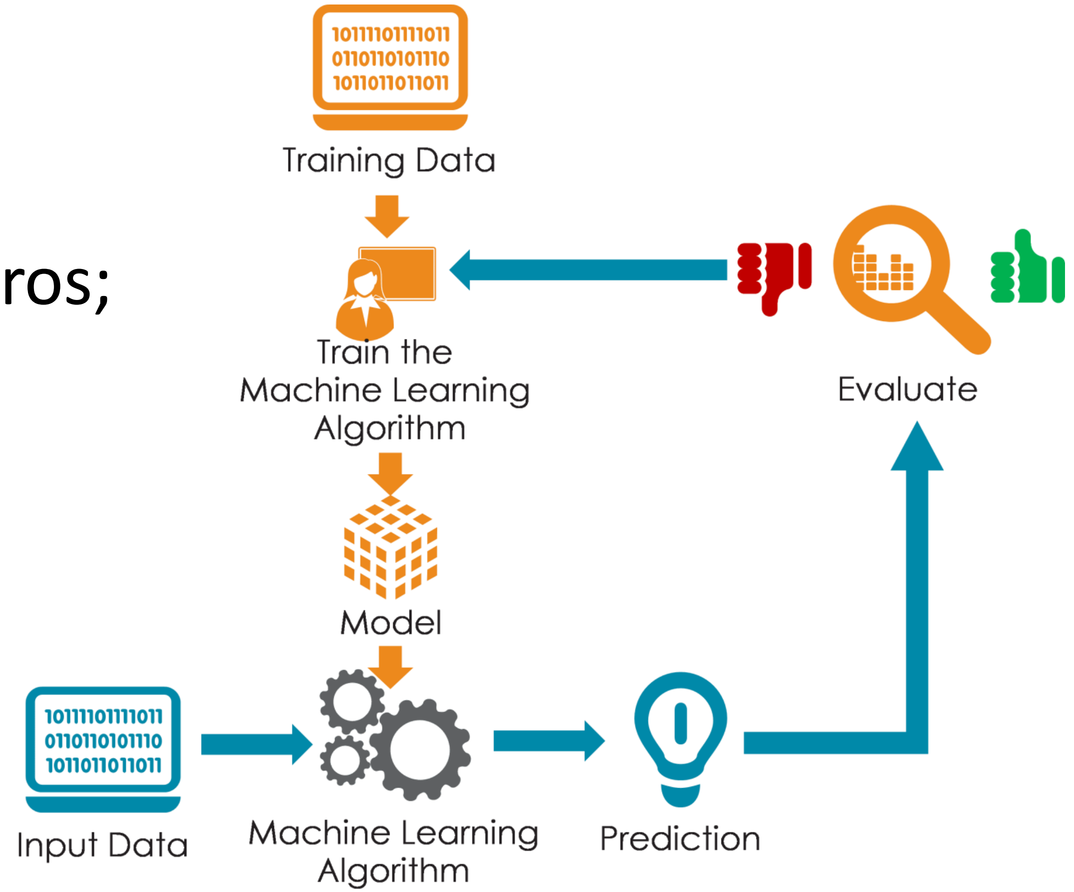
Função (diagonal sólida preta) e sua incerteza preditiva em $x = 60$ (representada como uma Gaussiana)



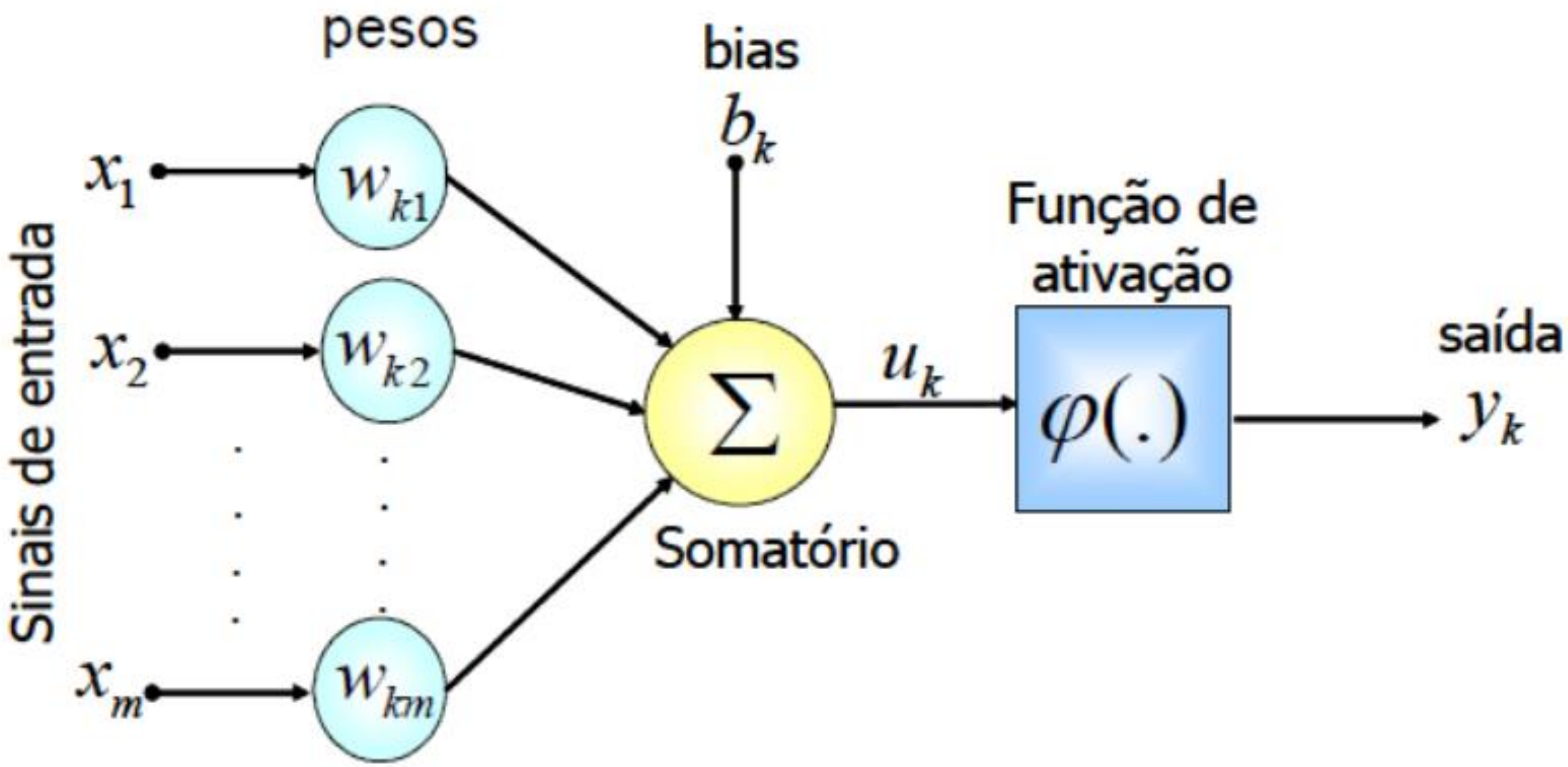
Aprender é encontrar parâmetros

Existem três fases algorítmicas distintas ao discutir algoritmos de aprendizagem de máquina:

- Seleção de modelo;
  - Ajuste de hiperparâmetros
- Treinamento ou estimativa de parâmetros;
- Predição ou inferência.



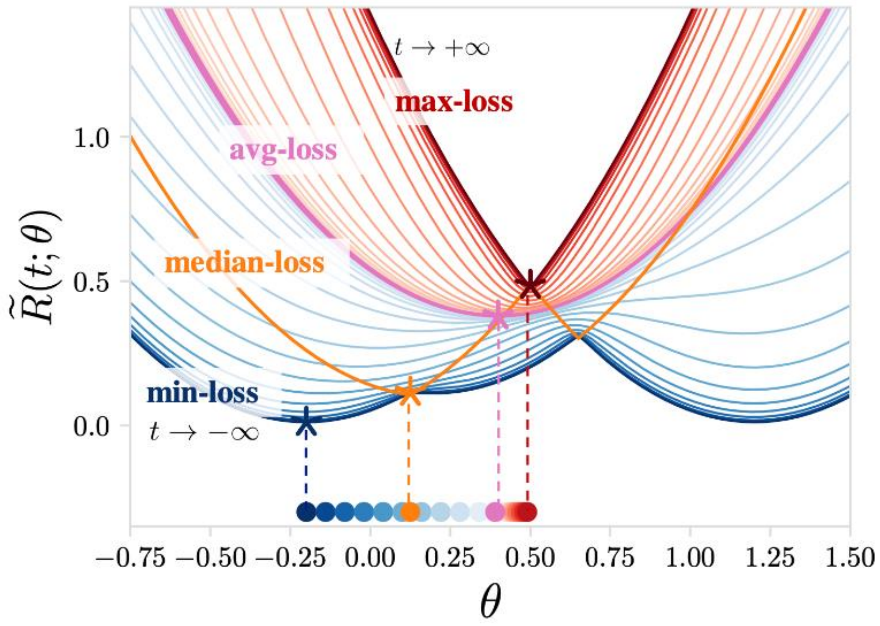
Parâmetros x Hiperparâmetros



Minimização de Risco Empírico

Como modelos de aprendizado de máquina realizam predições?

Como os modelos "aprendem“ com dados?



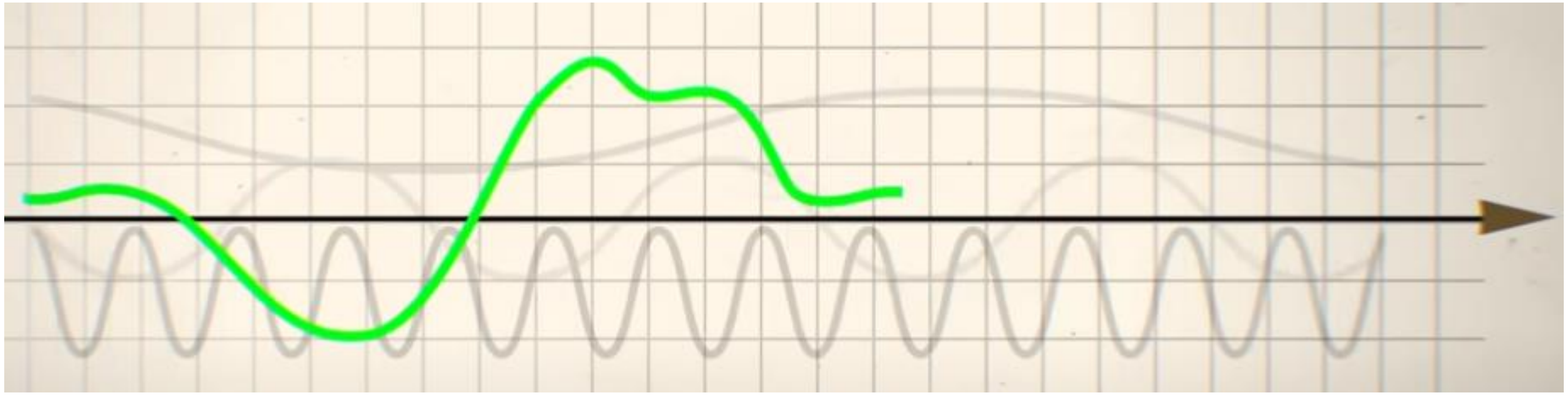
Minimização de Risco Empírico

Existem quatro escolhas principais de design que estão associadas às perguntas abaixo e que serão apresentadas detalhadamente nas subseções a seguir:

1. Qual é a classe de hipóteses que permitimos que o preditor assuma?
2. Como medimos o desempenho do preditor nos dados de treinamento?
3. Como construímos preditores apenas a partir de dados de treinamento que têm bom desempenho em dados de teste não vistos?
4. Qual é o procedimento de busca no espaço dos modelos?

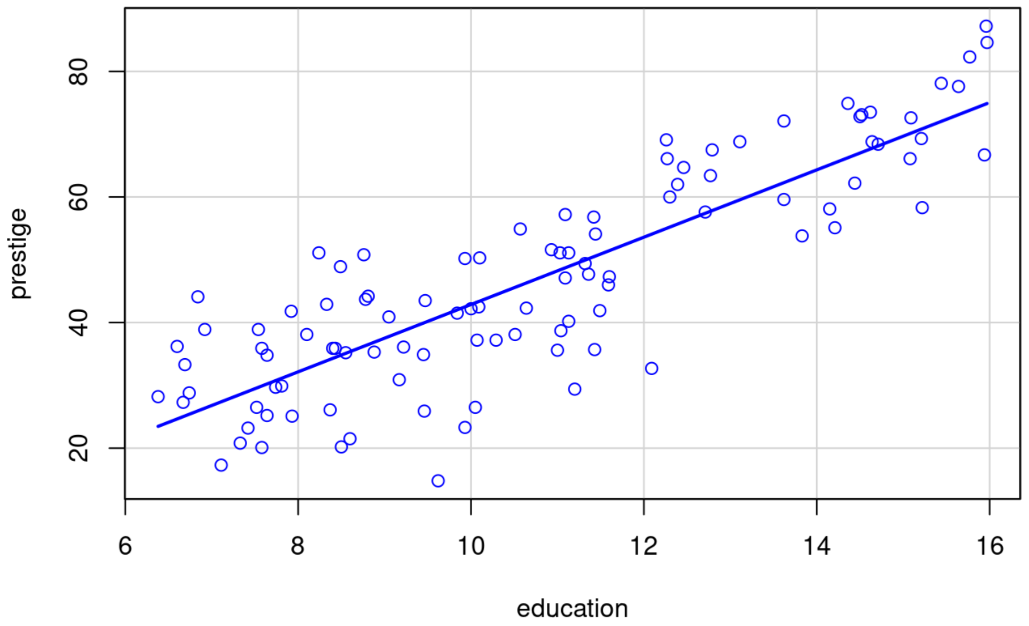
Classe de hipóteses de funções

É o conjunto de todas as funções possíveis que um modelo pode aprender para mapear entradas para saídas, dadas uma arquitetura e um conjunto de parâmetros.

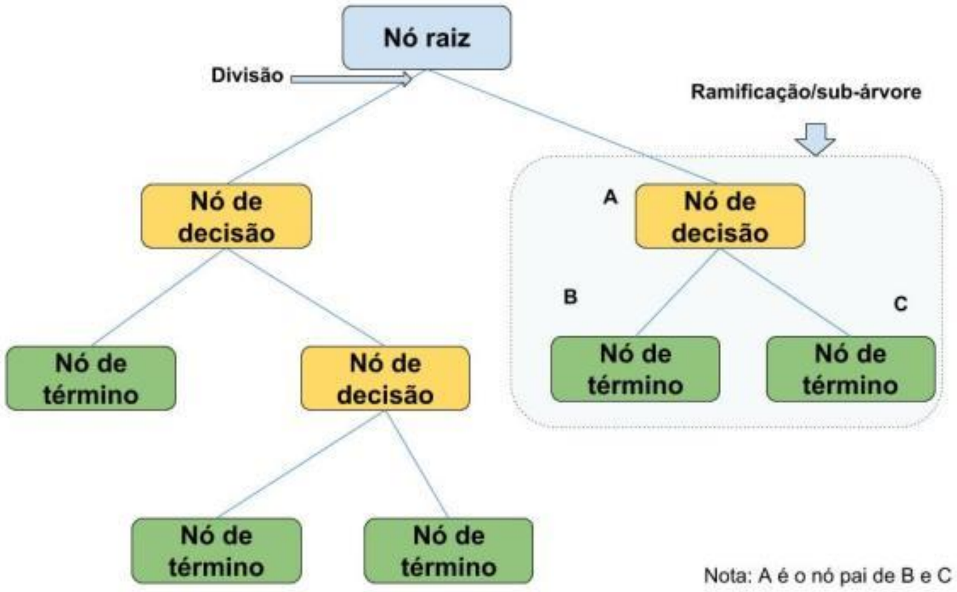


Exemplos de Classes de Hipóteses

Regressão Linear

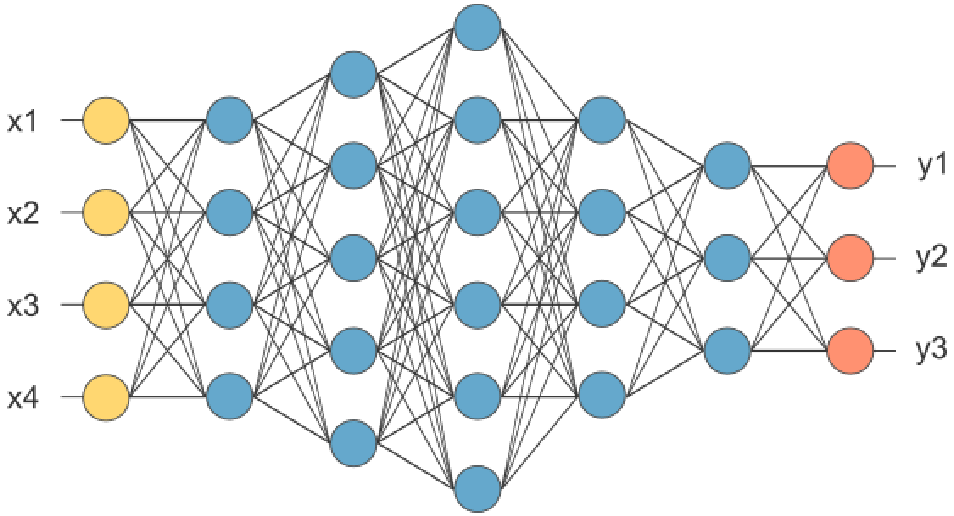


Árvores de Decisão

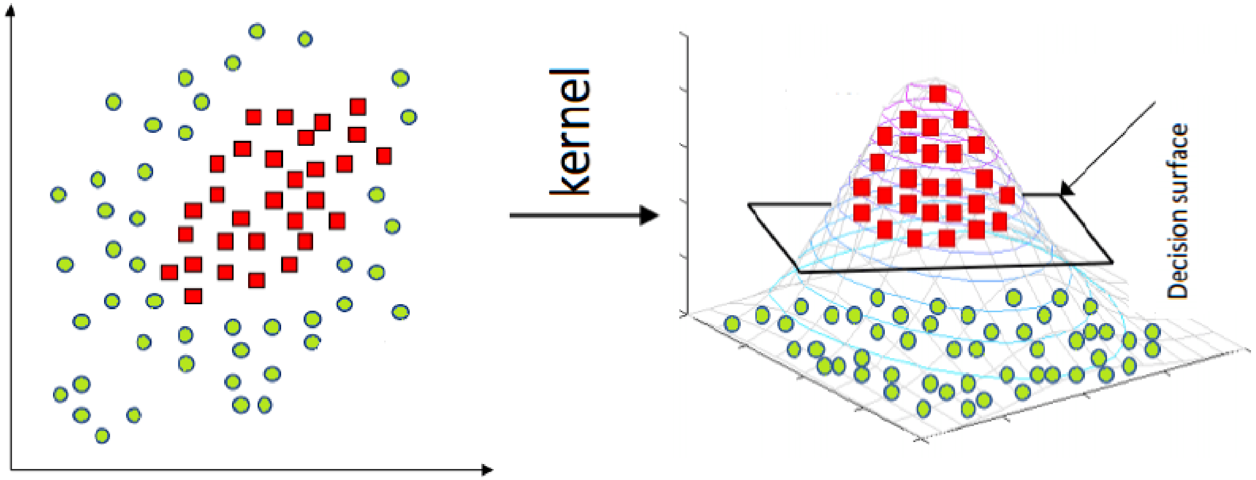


Exemplos de Classes de Hipóteses

Regressão Linear



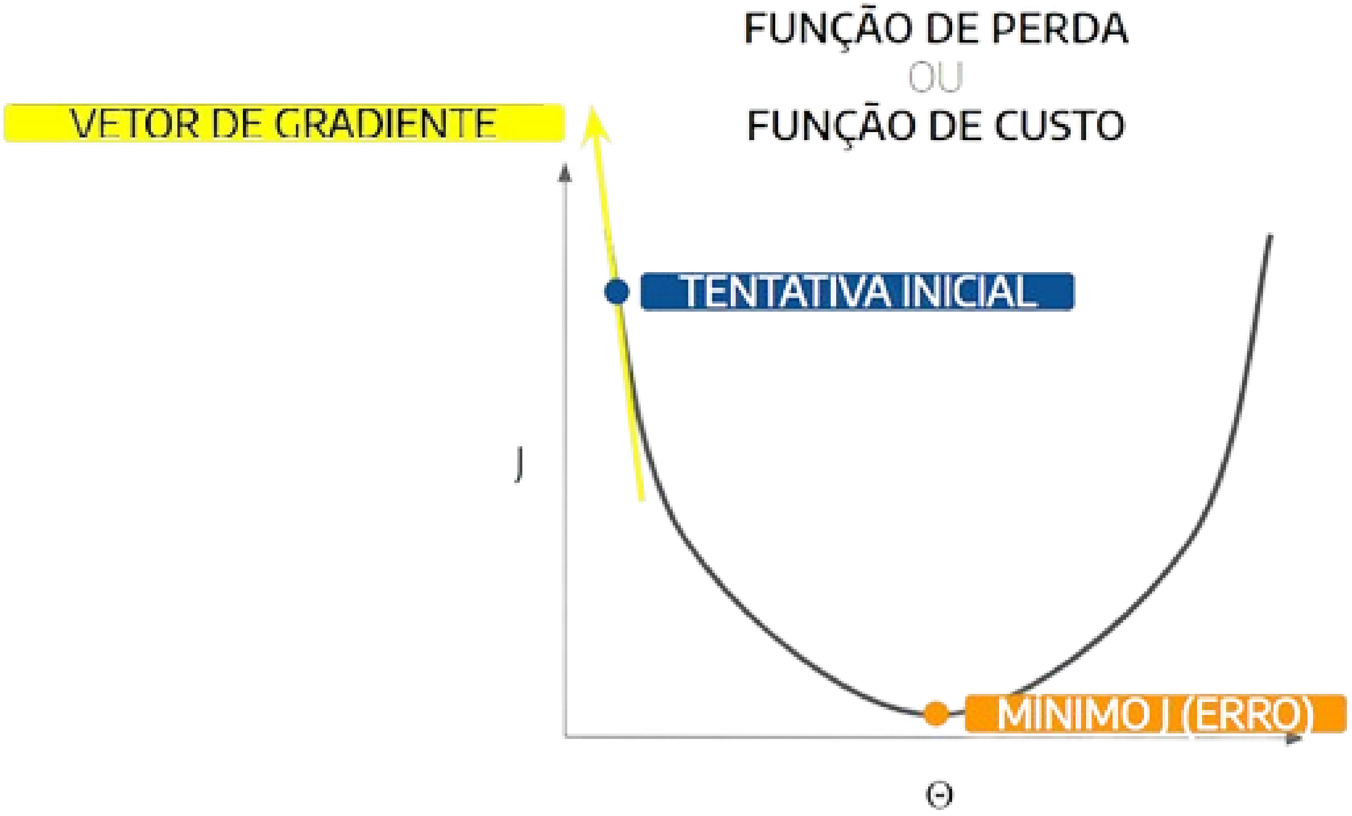
Máquinas de Vetores de Suporte (SVM)



Minimização de Risco Empírico

1. Qual é a classe de hipóteses que permitimos que o preditor assuma?
2. Como medimos o desempenho do preditor nos dados de treinamento?
3. Como construímos preditores apenas a partir de dados de treinamento que têm bom desempenho em dados de teste não vistos?
4. Qual é o procedimento de busca no espaço dos modelos?

Função de perda para treinamento



Exemplos de Funções de Perda

Erro Quadrático Médio (*Mean Square Error, MSE*)

$$\ell(y, \hat{y}) = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

Erro Absoluto Médio (*Mean Absolute Error, MAE*)

$$\ell(y, \hat{y}) = \frac{1}{N} \sum_{i=1}^{N} |y_i - \hat{y}_i|$$

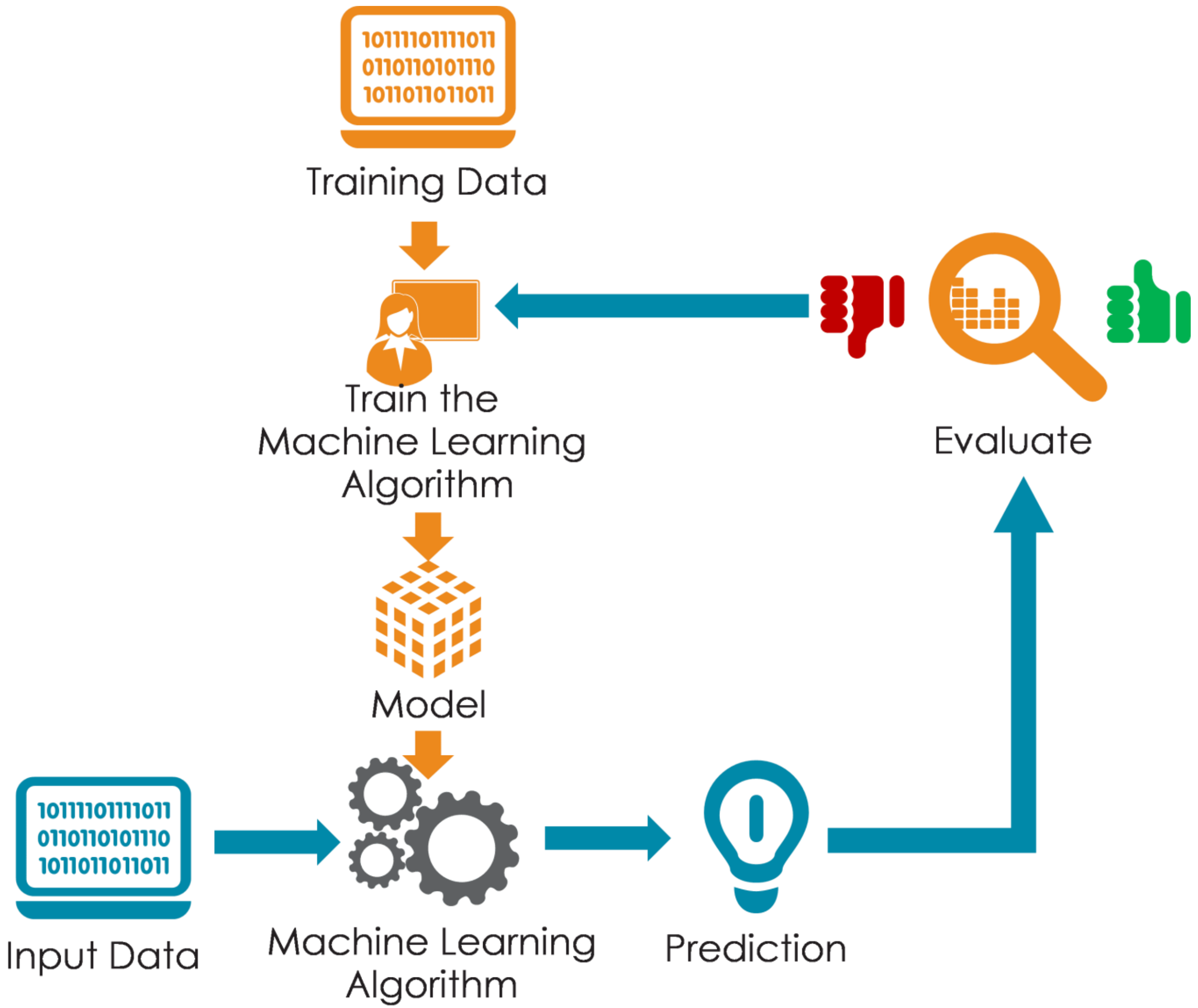
Equacionando o Risco Empírico

$$R_{\text{emp}}(f, X, y) = \frac{1}{N} \sum_{n=1}^{N} \ell(y_n, \hat{y}_n)$$

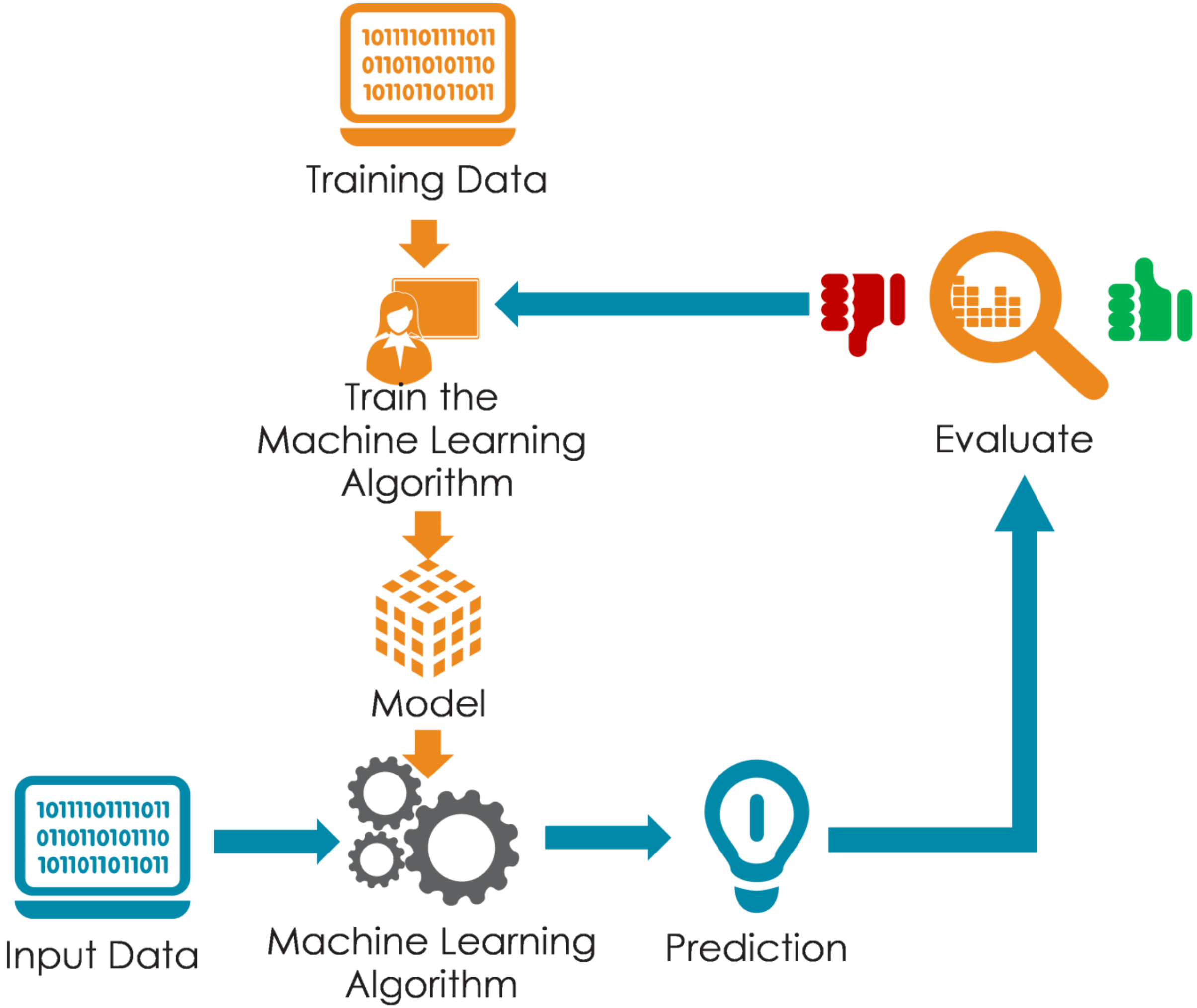
- f é o preditor;
- X é a matriz de exemplos;
- y é o vetor de rótulos;
- $\ell(y_n, \hat{y}_n)$ é a função de perda;
- N é o número de exemplos;
- $\hat{y}_n = f(x_n, \theta)$ é a predição do modelo para o exemplo x_n.

Minimização de Risco Empírico

1. Qual é a classe de hipóteses que permitimos que o preditor assuma?
2. Como medimos o desempenho do preditor nos dados de treinamento?
3. Como construímos preditores apenas a partir de dados de treinamento que têm bom desempenho em dados de teste não vistos?
4. Qual é o procedimento de busca no espaço dos modelos?


Training Data
Train the Machine Learning Algorithm
Evaluate
Model
Input Data
Machine Learning Algorithm
Prediction

Regularização para reduzir *Overfitting*



Regularização L2 (Ridge Regression)

$$\ell_{\text{reg}}(y, \hat{y}, \theta) = \ell(y, \hat{y}) + \lambda||\theta||_2^2$$

- $\ell(y, \hat{y})$ é a função de perda original,
- λ é o hiperparâmetro de regularização,
- $||\theta||_2^2$ é a norma $L2$ (quadrado da norma Euclidiana) dos parâmetros do modelo θ.

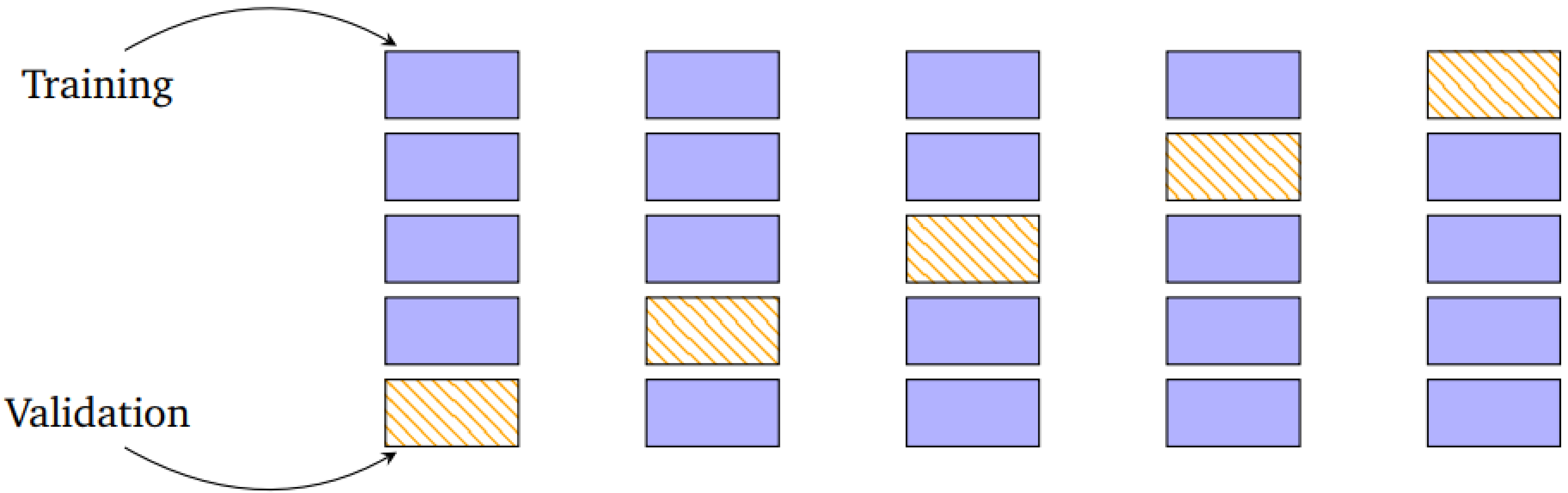
Minimização de Risco Empírico

1. Qual é a classe de hipóteses que permitimos que o preditor assuma?
2. Como medimos o desempenho do preditor nos dados de treinamento?
3. Como construímos preditores apenas a partir de dados de treinamento que têm bom desempenho em dados de teste não vistos?
4. Qual é o procedimento de busca no espaço dos modelos?

Validação cruzada para avaliar o desempenho da generalização

K-Fold Cross-Validation

Exemplo de janelamento *K-Fold*



Validação cruzada *K-Fold*. *Folds* de treinamento (azul) e *fold* de validação (laranja listrado).

O que aprendemos?

A técnica de minimização do risco empírico é central no aprendizado de máquina e envolve os seguintes elementos:

- Classe de Hipóteses de Funções
- Funções de Perda
- Regularização
- Validação Cruzada

Modelos probabilísticos gráficos
(*Directed Graphical Models*)

Seleção de modelos
(*Model selection*)

Aluno: Gabriel Almeida

Modelos probabilísticos gráficos

Roteiro

- Representações gráficas
  - Definições, propriedades e exemplos
- Independência condicional
  - Propriedades e exemplos
- Seleção de modelos
  - Definições e propriedades
  - *Nested Cross-Validation*
  - Seleção de modelos Bayesianos

Representação gráfica de modelos probabilísticos

Representações gráficas

Introdução

- Modelos probabilísticos gráficos são representações gráficas que representam modelos probabilísticos
- Permite a visualização gráfica das dependências entre variáveis aleatórias
- Auxilia a identificação de fatores que dependem de um subconjunto de variáveis aleatórias
- Também representam visualmente algoritmos de inferência, e.g. Programação Dinâmica, Monte Carlo via Cadeias de Markov

Representações gráficas

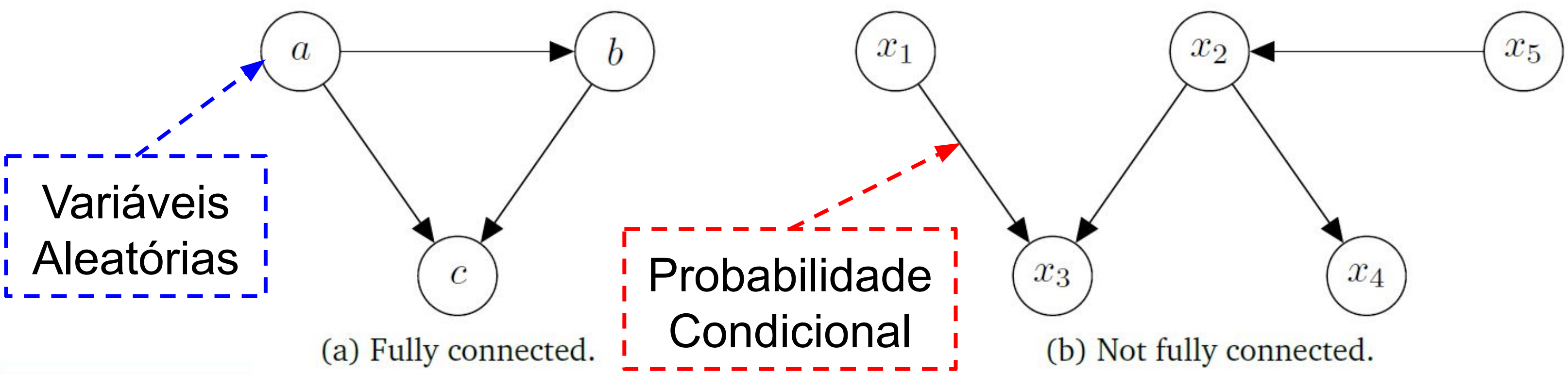
Introdução

- Distribuição conjunta contém informações sobre verossimilhança e posteriori, porém, não nos informa nada sobre as propriedades estruturais do modelo probabilístico.
- **Exemplo:**
  - A distribuição conjunta não nos diz nada sobre as relações de independência do modelo probabilístico
- Neste ponto que os modelos gráficos se tornam interessantes, pois se baseiam nos conceitos de independência condicional

Representações gráficas

Propriedades

- Visualização da estrutura de um modelo probabilístico
- Usados para projetar novos tipos de modelos estatísticos
- Ilustra propriedades, como independência condicional
- Inferência e aprendizado em modelos estatísticos podem ser expressas em termos de manipulações gráficas



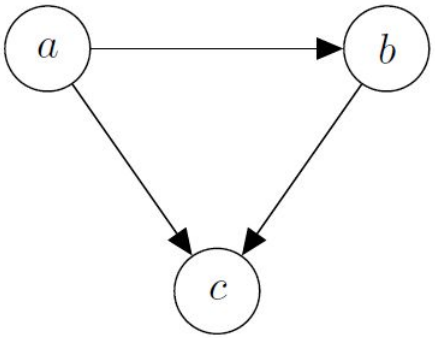
(a) Fully connected.

(b) Not fully connected.

Representações gráficas

Introdução

- **Exemplo 1** - dada a distribuição, gerar a representação gráfica
  - Dada a distribuição conjunta
    - **p(a, b, c) = p(c | a, b) p(b | a) p(a)**
  - A fatoração da distribuição conjunta nos diz que **c** depende de **a** e **b**, **b** depende de **a** e **a** não depende de **b** nem de **c**



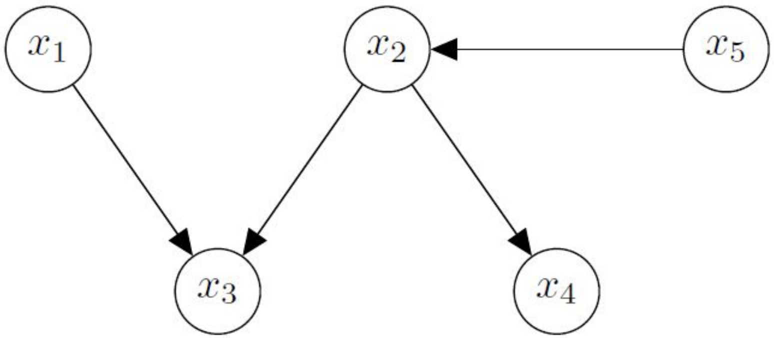
(a) Fully connected.

Representações gráficas

Introdução

- **Exemplo 2** - dada a representação gráfica extrair a distribuição
  - Olhando para o grafo encontramos que
    - A distribuição p(x1, … x5) é um conjunto de 5 condicionais
    - Cada condicional depende apenas dos pais do nó no grafo

p(x1, x2, x3, x4, x5) = p(x1) p(x5) p(x2 | x5) p(x3 | x1, x2) p(x4 | x2)



(b) Not fully connected.

Independência condicional entre variáveis

Independência condicional

Definições

- Apenas observando o grafo conseguimos encontrar propriedades de independência condicional, para isso as d-separações são fundamentais

- Dado um grafo onde A, B e C são nós não intersectantes, queremos verificar se A é condicionalmente independente de B dado C, denotado:

$$\mathcal{A} \perp\!\!\!\perp \mathcal{B} \mid \mathcal{C}$$

Independência condicional

Propriedades

- Para verificar se A é condicionalmente independente de B dado C

$$\mathcal{A} \perp\!\!\!\perp \mathcal{B} \mid \mathcal{C}$$

- Considera todos os caminhos possíveis ignorando a direção das arestas
- Qualquer caminho é considerado bloqueado se incluir algum nó onde:

  1. As setas no caminho se encontram cabeça-cauda ou cauda-cauda no nó e o nó está no conjunto C
  2. As setas se encontram cabeça-cabeça no nó, e o nó e nenhum de seus descendentes estão em C

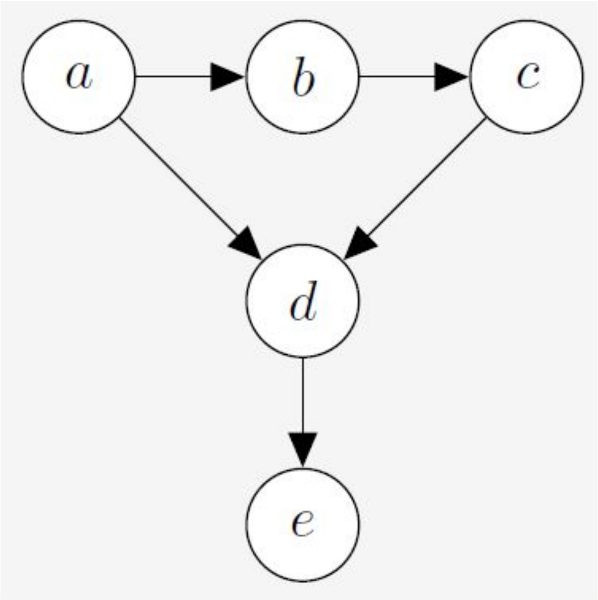
Independência condicional

Propriedades

1. As setas no caminho se encontram cabeça-cauda ou cauda-cauda no nó e o nó está no conjunto C
2. As setas se encontram cabeça-cabeça no nó, e o nó e nenhum de seus descendentes estão em C

- Exemplos

$$b \perp\!\!\!\perp d \mid a, c$$



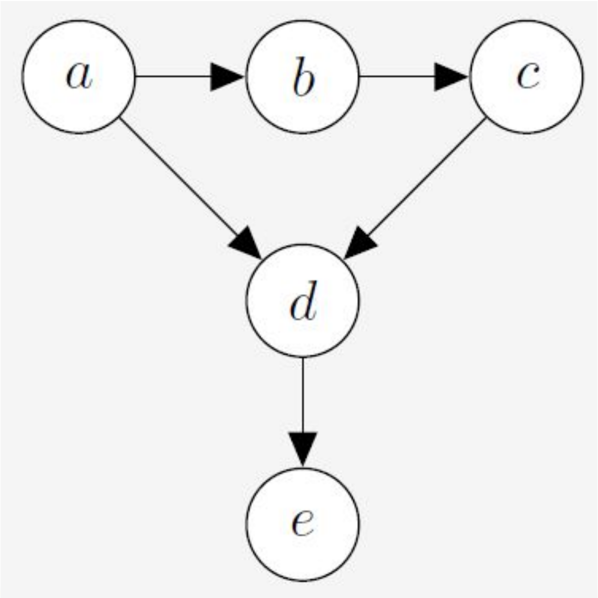
Independência condicional

Propriedades

1. As setas no caminho se encontram cabeça-cauda ou cauda-cauda no nó e o nó está no conjunto C
2. As setas se encontram cabeça-cabeça no nó, e o nó e nenhum de seus descendentes estão em C

- Exemplos

$$a \perp\!\!\!\perp c \mid b$$



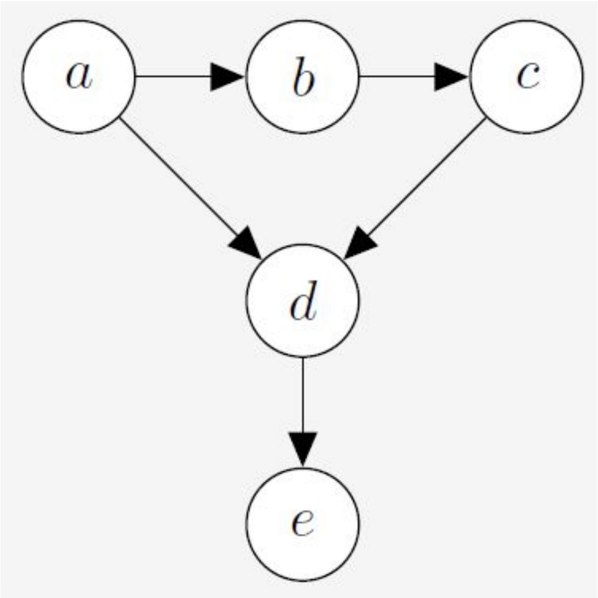
Independência condicional

Propriedades

1. As setas no caminho se encontram cabeça-cauda ou cauda-cauda no nó e o nó está no conjunto C
2. As setas se encontram cabeça-cabeça no nó, e o nó e nenhum de seus descendentes estão em C

- Exemplos

$$b \not\perp\!\!\!\perp d \mid c$$



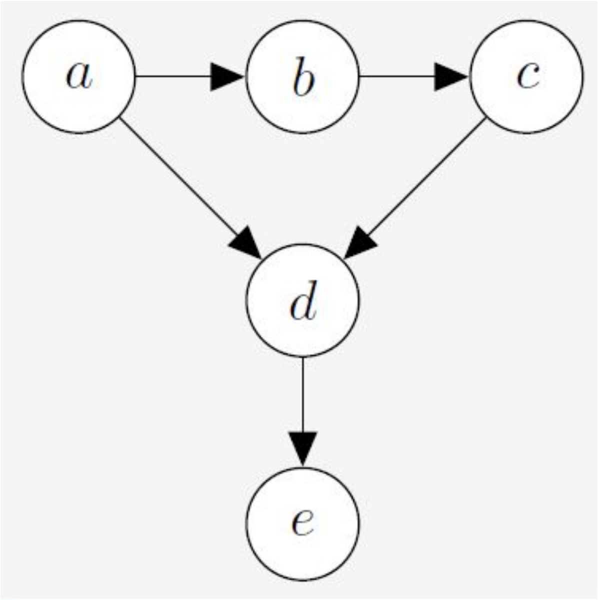
Independência condicional

Propriedades

1. As setas no caminho se encontram cabeça-cauda ou cauda-cauda no nó e o nó está no conjunto C
2. As setas se encontram cabeça-cabeça no nó, e o nó e nenhum de seus descendentes estão em C

- Exemplos

$$a \not\perp\!\!\!\perp c \mid b, e$$



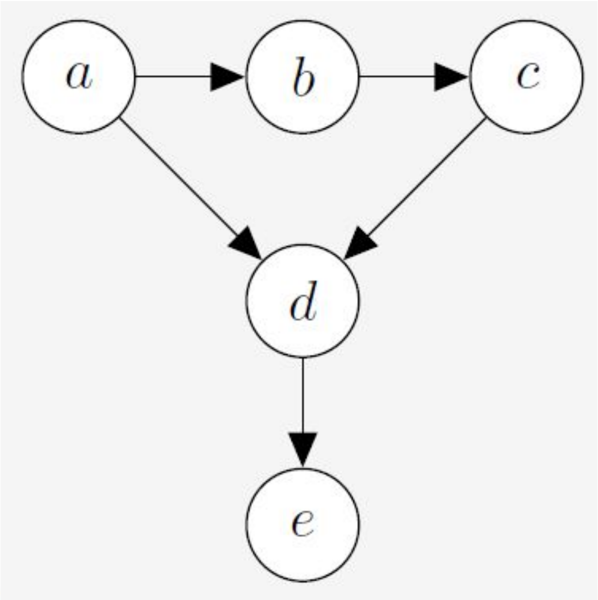
Independência condicional

Propriedades

1. As setas no caminho se encontram cabeça-cauda ou cauda-cauda no nó e o nó está no conjunto C
2. As setas se encontram cabeça-cabeça no nó, e o nó e nenhum de seus descendentes estão em C

- Exemplos

$$a \not\perp\!\!\!\perp c \mid b, e$$



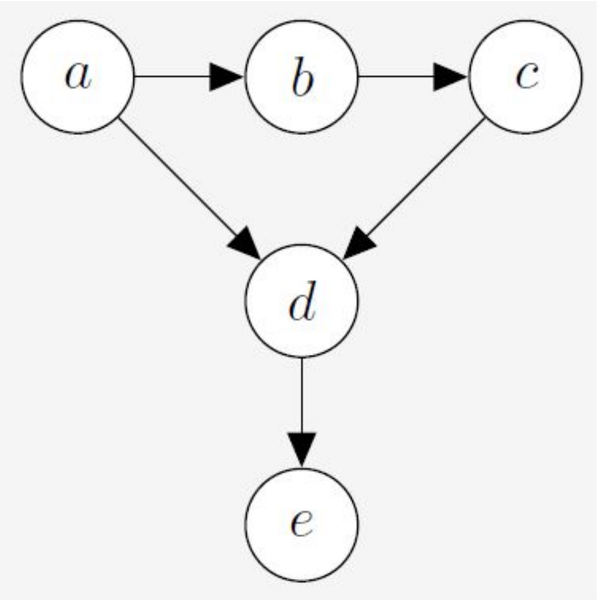
Independência condicional

Propriedades

1. As setas no caminho se encontram cabeça-cauda ou cauda-cauda no nó e o nó está no conjunto C
2. As setas se encontram cabeça-cabeça no nó, e o nó e nenhum de seus descendentes estão em C

- Exemplos

$$a \not\perp\!\!\!\perp c \mid b, e$$



Seleção de modelos

Seleção de modelos

Definições

- Em ML precisamos tomar decisões de modelagem de alto nível que influenciam criticamente o desempenho do modelo
- Tais escolhas (e.g. forma da verossimilhança) influenciam os parâmetros livres no modelo, sua flexibilidade e expressividade
- Modelos mais complexos são mais flexíveis e podem ser usado para descrever mais dados
- Exemplo
  - Funções de primeiro grau resolvem equações do tipo f(x) = y
  - Funções de segundo grau descrevem relações quadráticas

Seleção de modelos

Definições

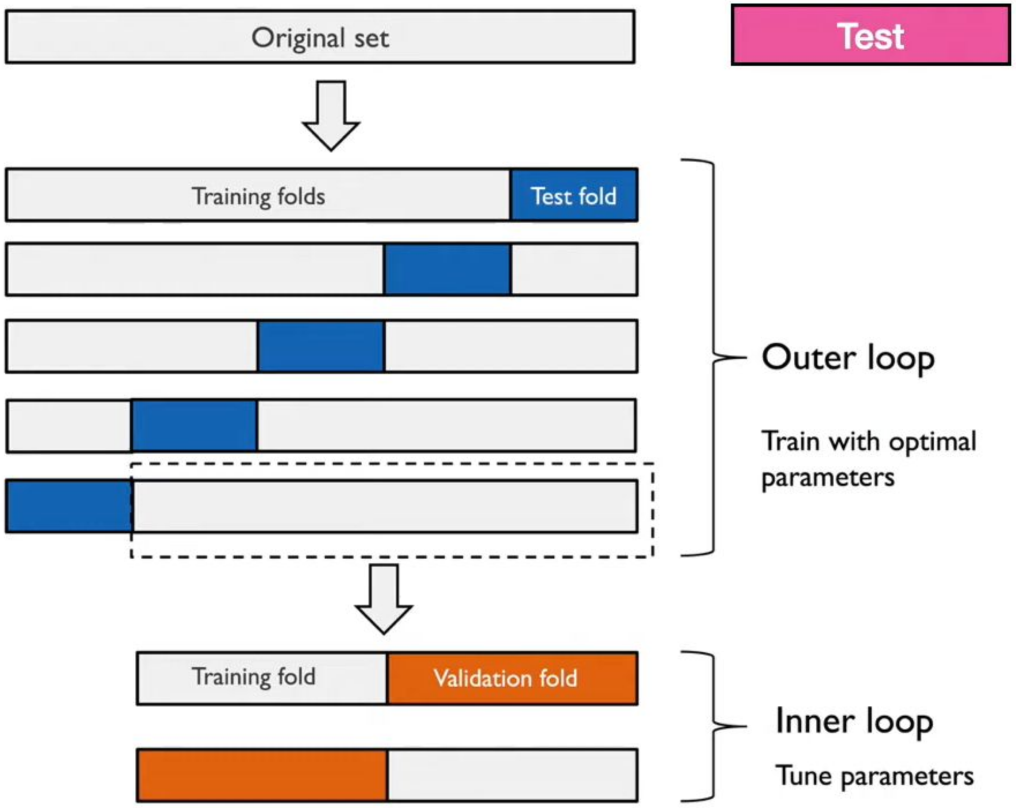
- Um problema comum em ML é a avaliação do modelo
- Durante o treinamento o modelo usa apenas os dados de treinamento para avaliar seu desempenho
- O desempenho nos dados de treinamento não é interessante
- Estimação máxima da verossimilhança pode levar ao *overfitting*
- O interessante é o desempenho do modelo no conjunto de testes
- Avaliando assim a generalização do modelo para os dados de testes não conhecidos durante o treinamento
- Seleção de modelos preocupa justamente com esse problema

Nested Cross-Validation

Seleção de modelos

Nested Cross-Validation

- Realiza em cada divisão uma rodada adicional de *Cross-Validation*
- ***Nested Cross-Validation*** Seleciona modelos/algoritmos
- ***Cross-Validation*** seleciona hiper parâmetros



Seleção de modelos

Nested Cross-Validation

- O loop interno estima o erro de generalização de um modelo usando o erro empírico no conjunto de validação, onde:
  - $\mathbf{R}(\mathcal{V} \mid M)$ é o risco empírico (e.g. root mean square error - RMSE) do conjunto de validação V, para o modelo M
- Para cada modelo o cálculo é realizado e é escolhido o modelo com melhor desempenho

$$\mathbb{E}_{\mathcal{V}}[\mathbf{R}(\mathcal{V} \mid M)] \approx \frac{1}{K} \sum_{k=1}^{K} \mathbf{R}(\mathcal{V}^{(k)} \mid M),$$

Seleção de modelos Bayesianos

Seleção de modelos

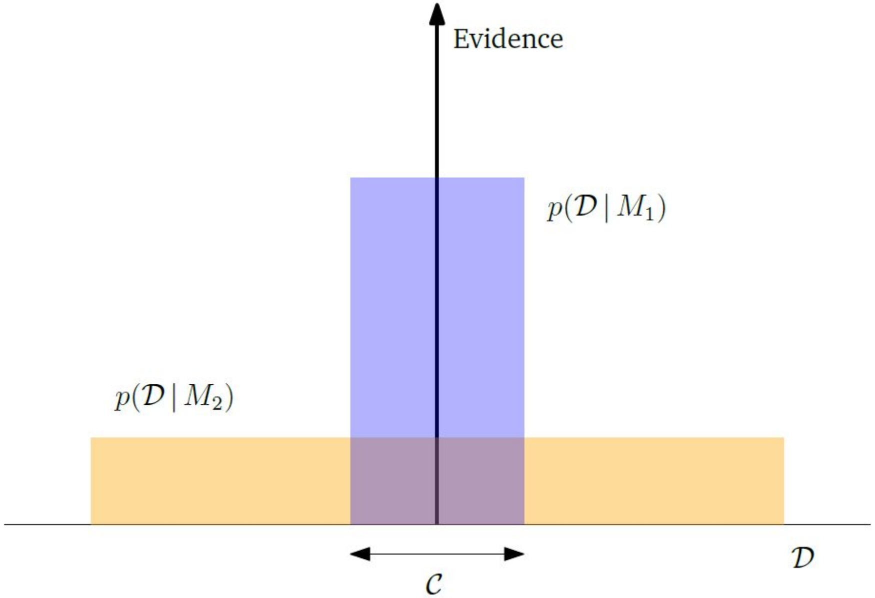
Seleção de modelos Bayesianos

- Existem diversas abordagens para seleção de modelos, em geral, todas tentam equilibrar a complexidade ao ajuste dos dados
- Navalha de Occam: Modelos mais simples são menos propensos ao *overfitting*, o objetivo é o modelo mais simples que adere aos dados
- Aplicações de probabilidade Bayesiana incorporam uma “Navalha de Occam automática”

Seleção de modelos

Seleção de modelos Bayesianos

- Seja D o espaço de todos os conjunto de dados
- Se estamos interessados na prob. Posterior p(Mi | D) podemos usar o teorema de Bayes assumindo uma priori uniforme p(M), que vão ter recompensa conforme a predição dos dados, p(D | Mi) (evidência)



Seleção de modelos

Seleção de modelos Bayesianos

- Seja M um conjunto de modelos, onde M_k possui os parâmetros θ_k
- Na seleção de modelos bayesianos colocamos uma priori p(M) em M
- O processo generativo permite gerar dados a partir do modelo

$$M_k \sim p(M)$$
$$\boldsymbol{\theta}_k \sim p(\boldsymbol{\theta} \mid M_k)$$
$$\mathcal{D} \sim p(\mathcal{D} \mid \boldsymbol{\theta}_k)$$

- Nos permitindo calcular a distribuição posteriori dos modelos como

$$p(M_k \mid \mathcal{D}) \propto p(M_k) p(\mathcal{D} \mid M_k)$$

Seleção de modelos

Comparando modelos a partir de fatores Bayesianos

- Considerando dois modelos M1 e M2 podemos comparar os modelos sobre um dado conjunto de dados D usando fatores bayesianos
- Se computarmos os posteriores p(M1 | D) e p(M2 | D), podemos calcular os fatores bayesianos da seguinte forma:

$$\underbrace{\frac{p(M_1 \mid \mathcal{D})}{p(M_2 \mid \mathcal{D})}}_{\text{posterior odds}} = \frac{\frac{p(\mathcal{D} \mid M_1)p(M_1)}{p(\mathcal{D})}}{\frac{p(\mathcal{D} \mid M_2)p(M_2)}{p(\mathcal{D})}} = \underbrace{\frac{p(M_1)}{p(M_2)}}_{\text{prior odds}} \underbrace{\frac{p(\mathcal{D} \mid M_1)}{p(\mathcal{D} \mid M_2)}}_{\text{Bayes factor}}.$$

- O *prior odds* mede quanto a "priori" favore M1 em relação a M2
- *Bayes factor* mede quão bem o modelo M1 prevê D comparado a M2

Aplicações

- **Representações gráficas**
  - AI/ML
    - Redes bayesianas: Modelagem de dependências probabilísticas
    - Modelos de Markov: Tarefas de previsão e classificação
  - Processamento de Linguagem Natural
    - Modelos de cadeias de Markov ocultas: reconhecimento de fala e sentimentos
- **Seleção de modelos**
  - AI/ML
    - Seleção de algoritmos
    - Otimização de hiper parâmetros
  - Engenharia de sistemas complexos
    - Análise e previsão de sistemas dinâmicos - redes de telefonia móvel
    - Controle de processo - escolha de modelos para processos industriais

Conclusão

- Parte 1 - Hudson Romualdo (hudson_romualdo@discente.ufg.br)
  - 8.1 Data, Models, and Learning
  - 8.2 Empirical Risk Minimization
- Parte 2 - André Riccioppo (andre.riccioppo@discente.ufg.br)
  - 8.3 Parameter Estimation
  - 8.4 Probabilistic Modeling and Inference
- Parte 3 - Gabriel Almeida (gabrielmatheus05@discente.ufg.br)
  - 8.5 Directed Graphical Models
  - 8.6 Model Selection