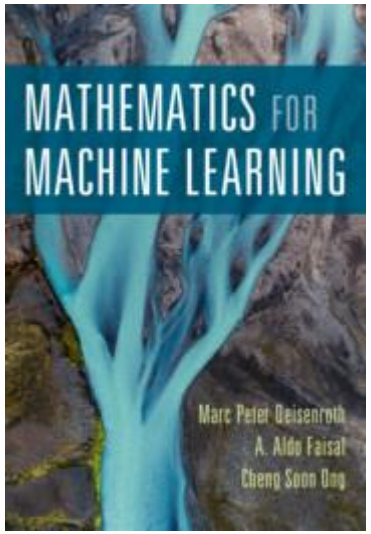
PROGRAMA DE PÓS-GRADUÇÃO EM CIÊNCIA DA COMPUTAÇÃO
TÓPICOS ESPECIAIS EM FUNDAMENTOS DE COMPUTAÇÃO – MATEMÁTICA E ESTATÍSTICA PARA CIÊNCIA DE DADOS
Prof. Dr. Rommel Melgaço Barbosa

Seminários

Quando os modelos encontram os dados



André Riccioppo

Gabriel Almeida

Hudson Romualdo

Junho/2024

UFG
UNIVERSIDADE
FEDERAL DE GOIÁS

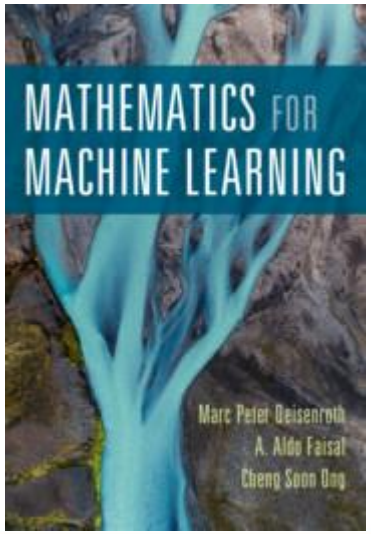
Introdução

Capítulo 8:

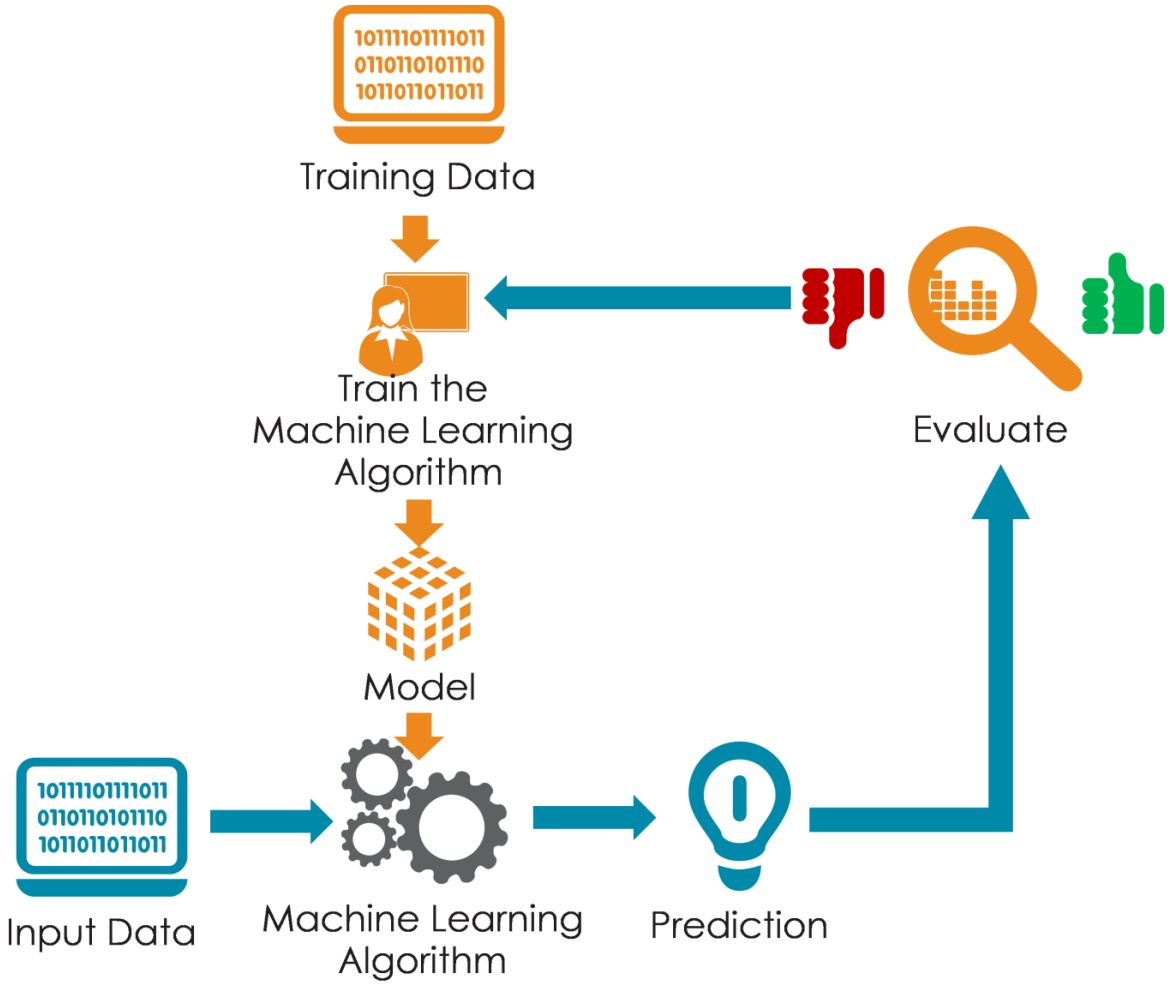
- Dados
- Modelos
- Aprendizado

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

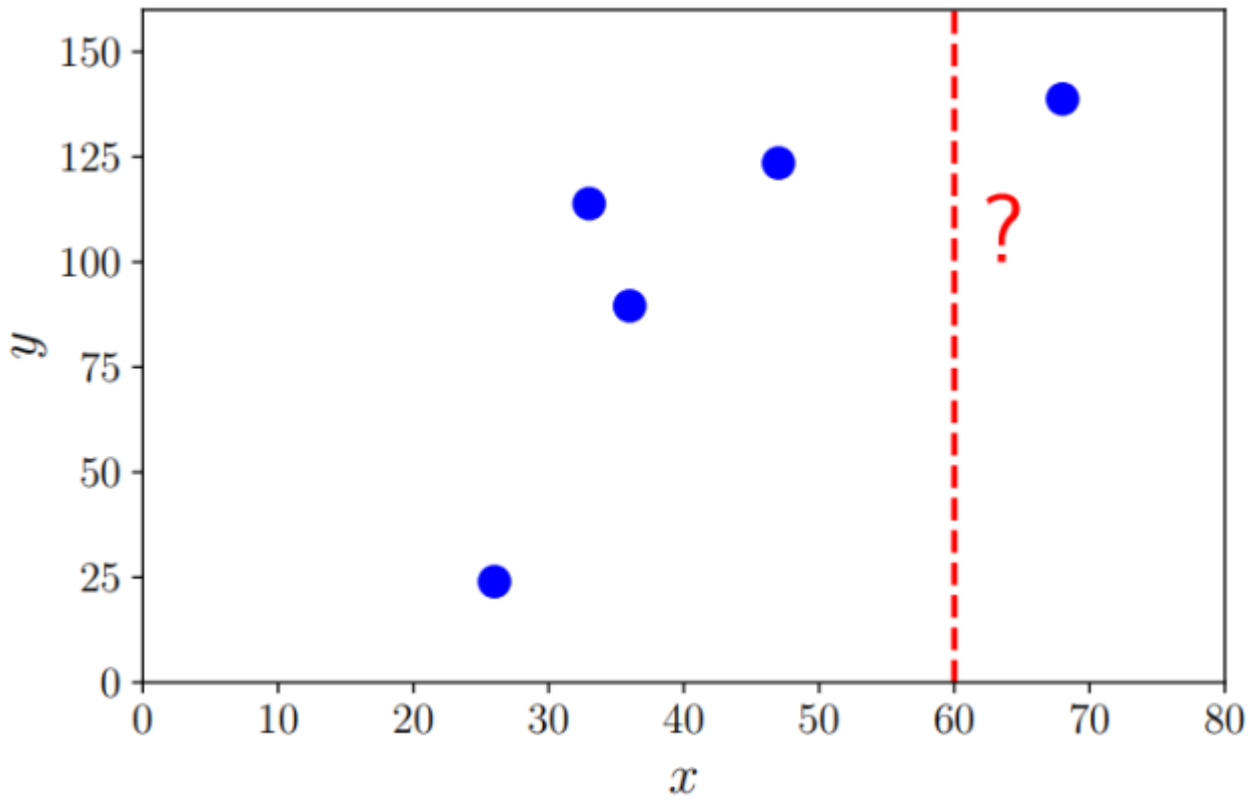
Dados como Vetores

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: $d = 1, \dots, D$.

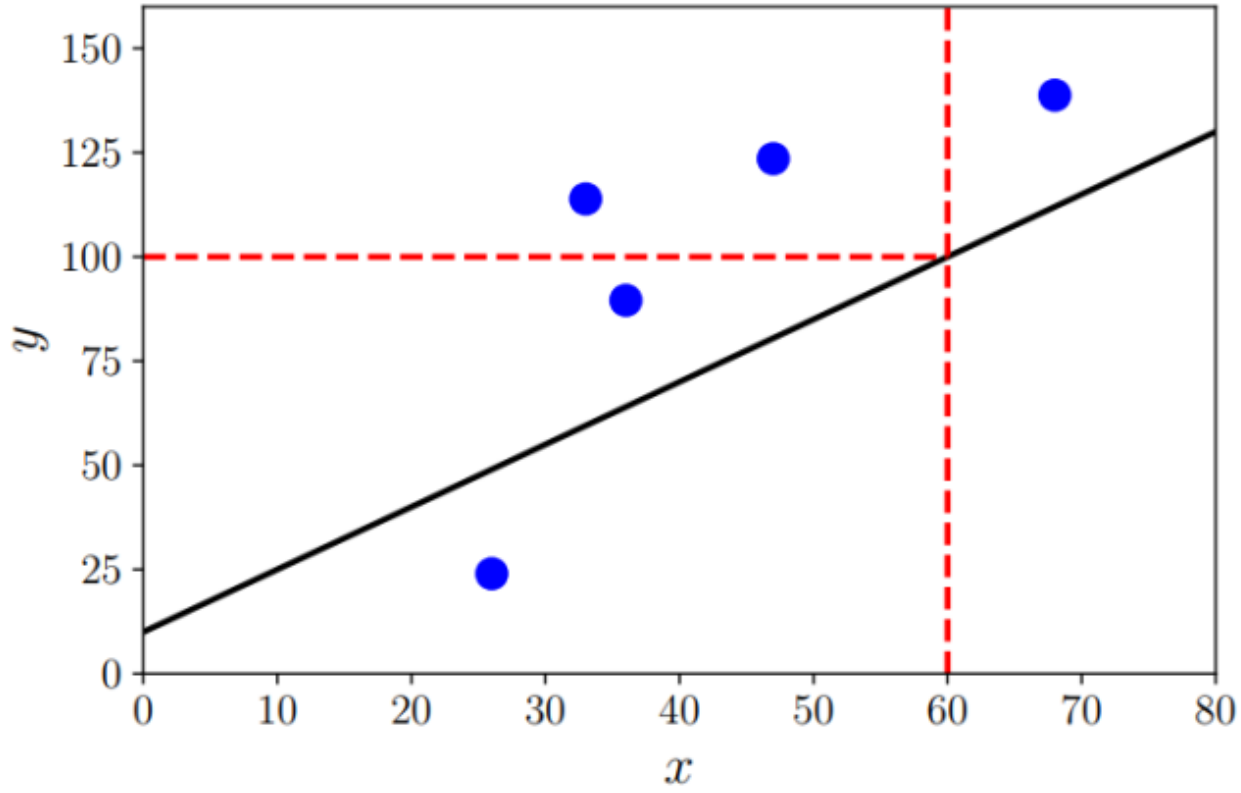


- *Target*:$\boldsymbol{y}_n$
- *Input*: $\boldsymbol{x}_n$.
- *Dataset:* $(\boldsymbol{x}_1, \boldsymbol{y}_1)$,..., $(\boldsymbol{x}_n, \boldsymbol{y}_n)$, ... $(\boldsymbol{x}_N, \boldsymbol{y}_N)$
- Conjunto de *datapoints* $\boldsymbol{x}_n$, ... $\boldsymbol{x}_N$: $\boldsymbol{X} \in \mathbb{R}^{N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

Modelos como funções

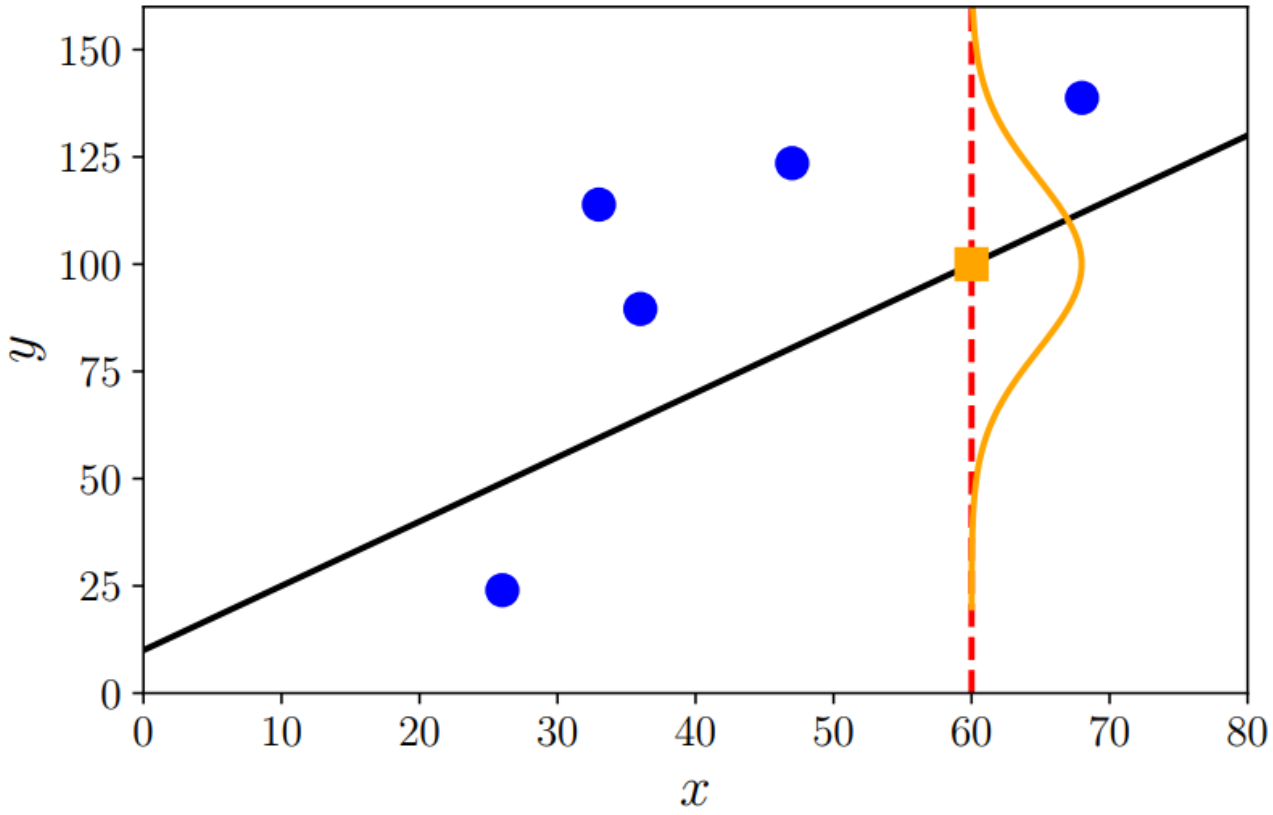
Um preditor (modelo treinado) é uma função quando, ao receber uma determinada entrada (no nosso caso, um vetor de características), produz um saída.

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



Modelos como distribuições de probabilidade

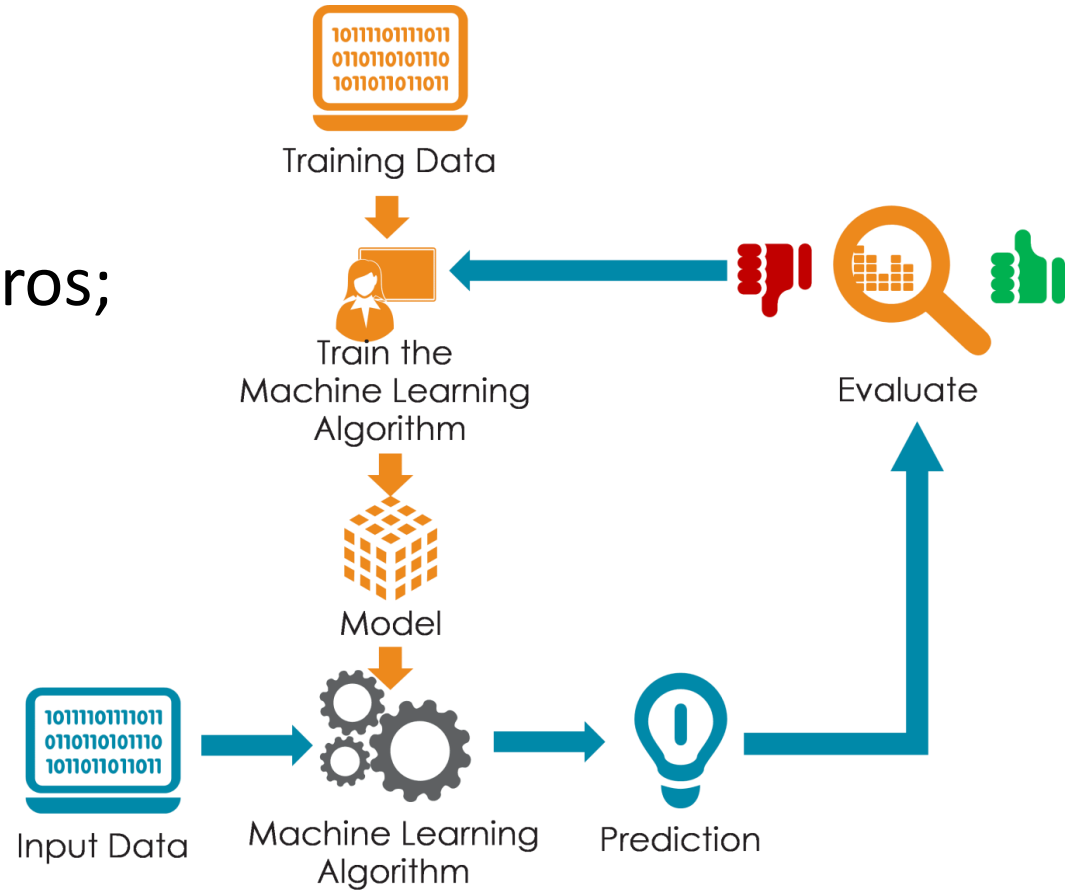
Função (diagonal sólida preta) e sua incerteza preditiva em $x = 60$ (representada como uma Gaussiana)



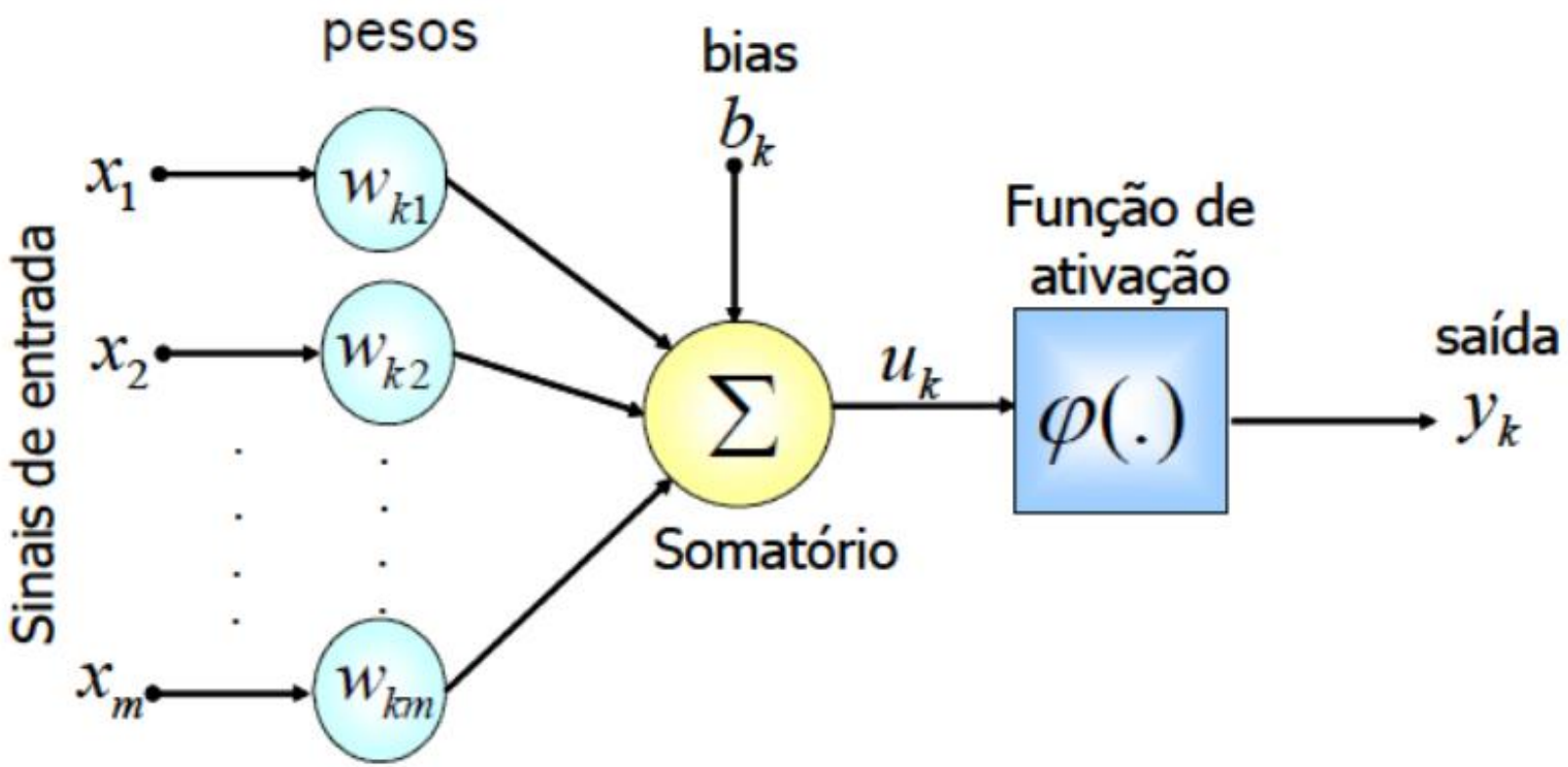
Aprender é encontrar parâmetros

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.



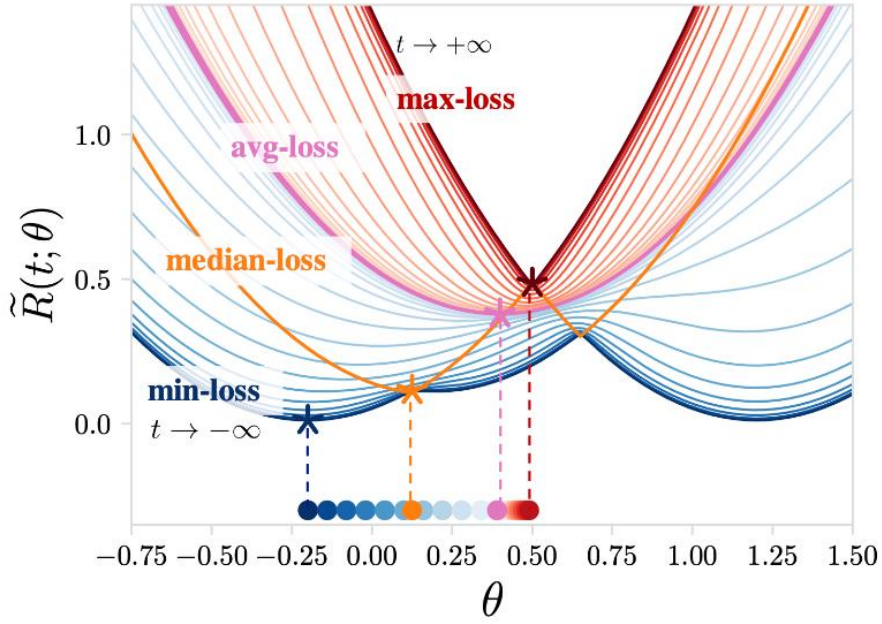
Parâmetros x Hiperparâmetros



Minimização de Risco Empírico

Como modelos de aprendizado de máquina realizam predições?

Como os modelos "aprendem“ com dados?



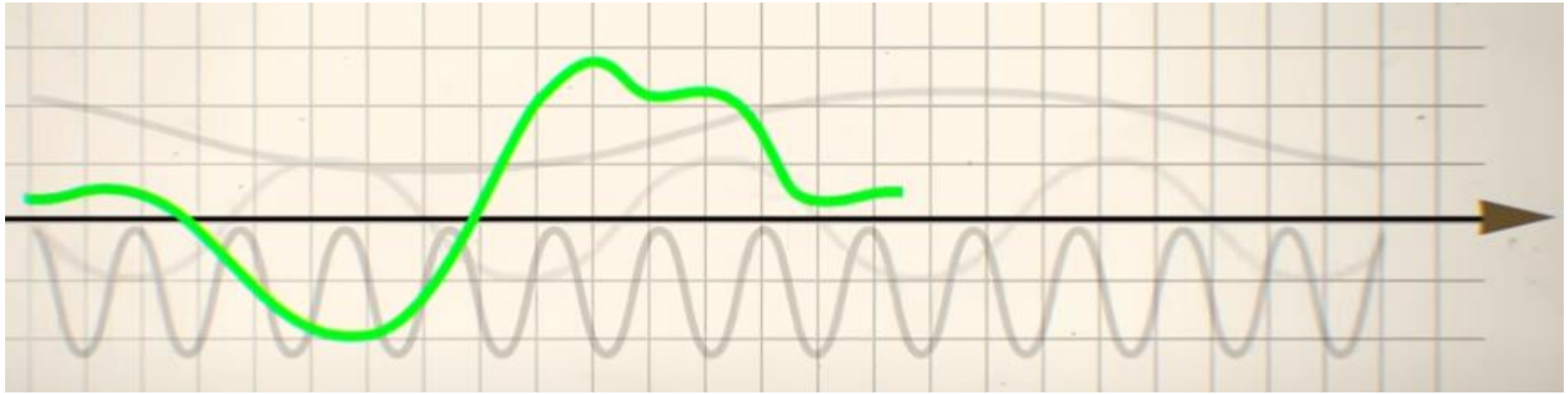
Minimização de Risco Empírico

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?

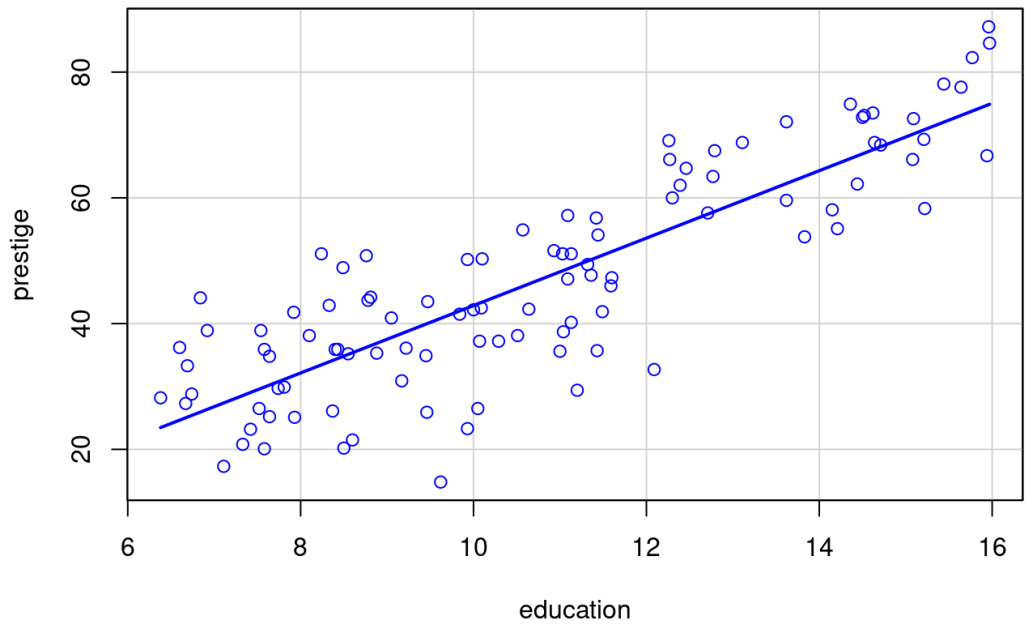
Classe de hipóteses de funções

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.

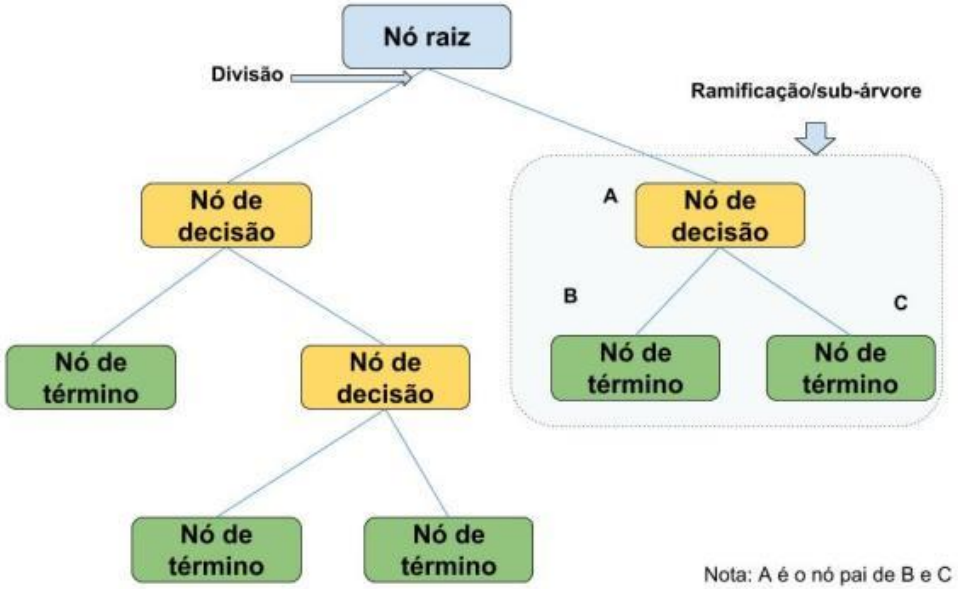


Exemplos de Classes de Hipóteses

Regressão Linear

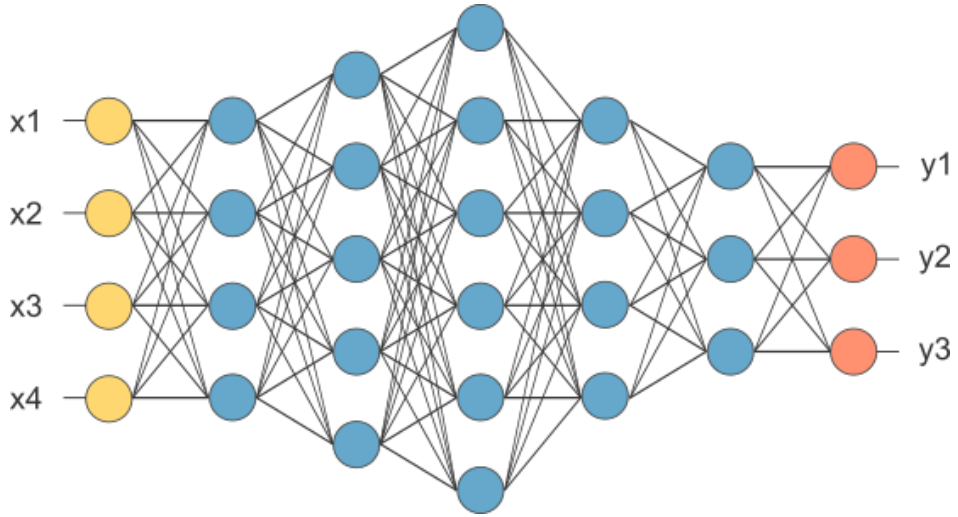


Árvores de Decisão

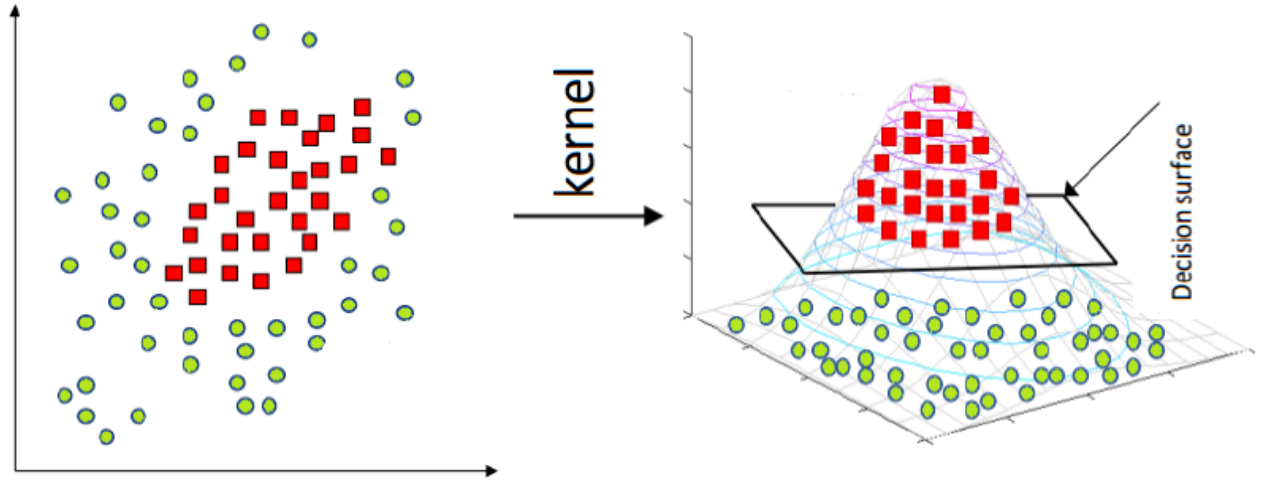


Exemplos de Classes de Hipóteses

Regressão Linear



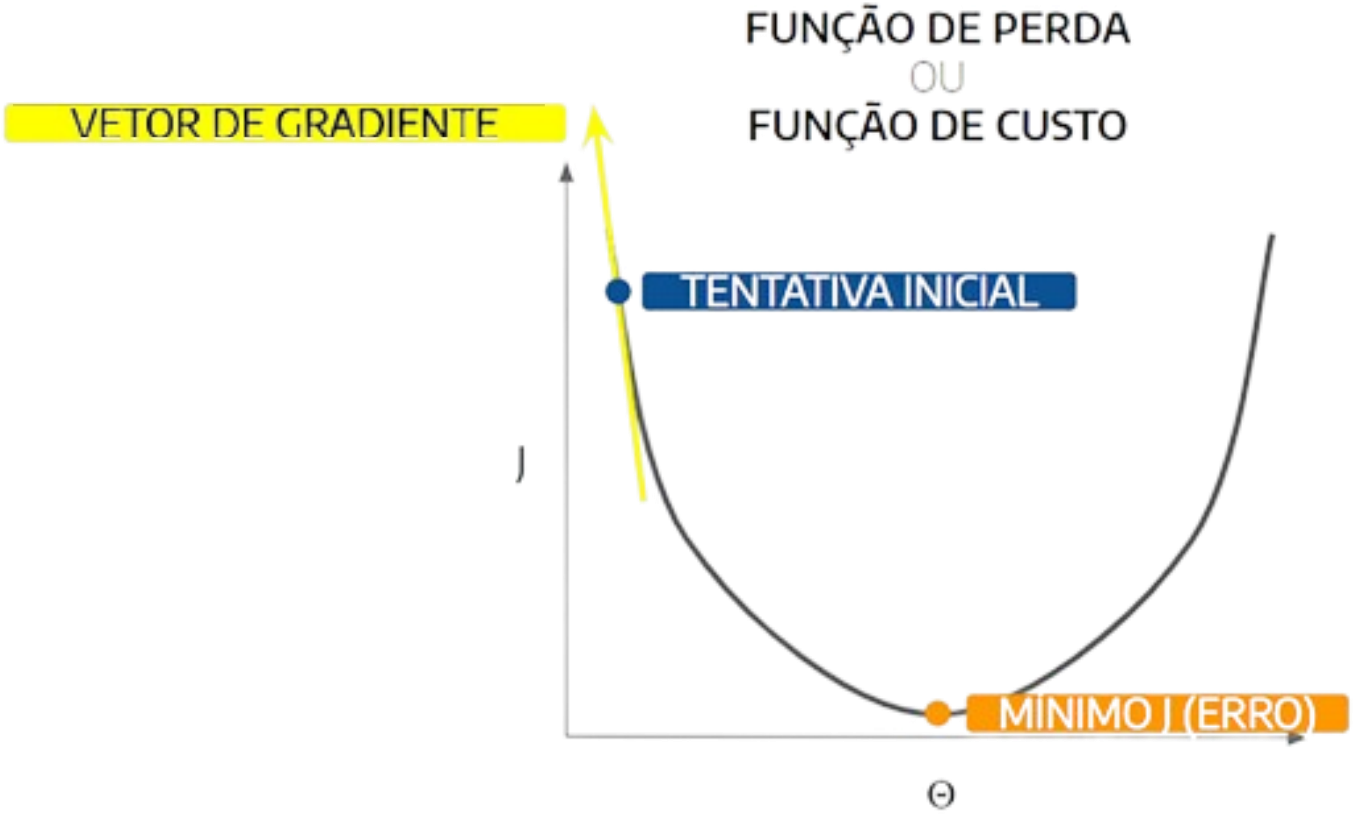
Máquinas de Vetores de Suporte (SVM)



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*)

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

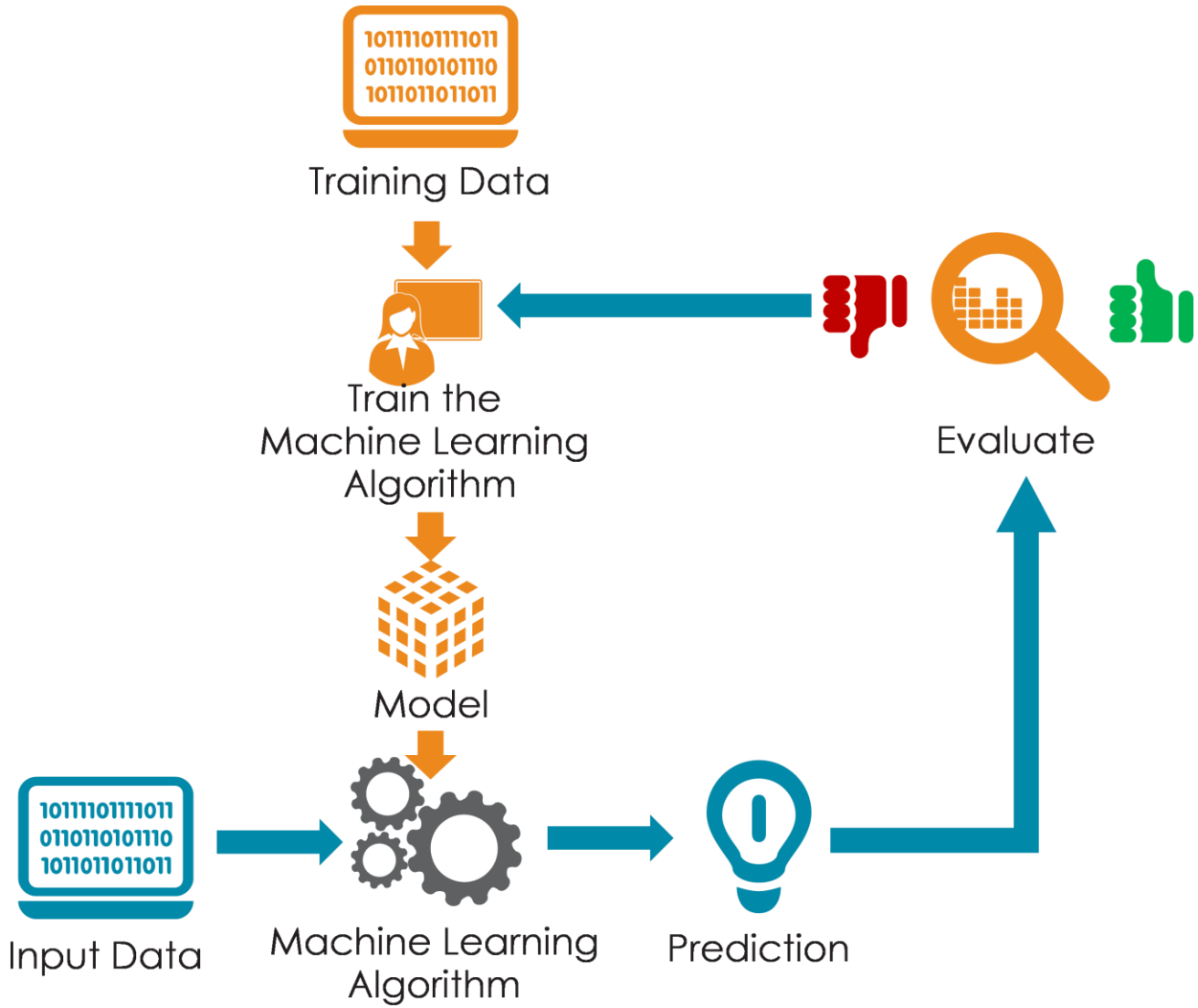
Equacionando o Risco Empírico

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

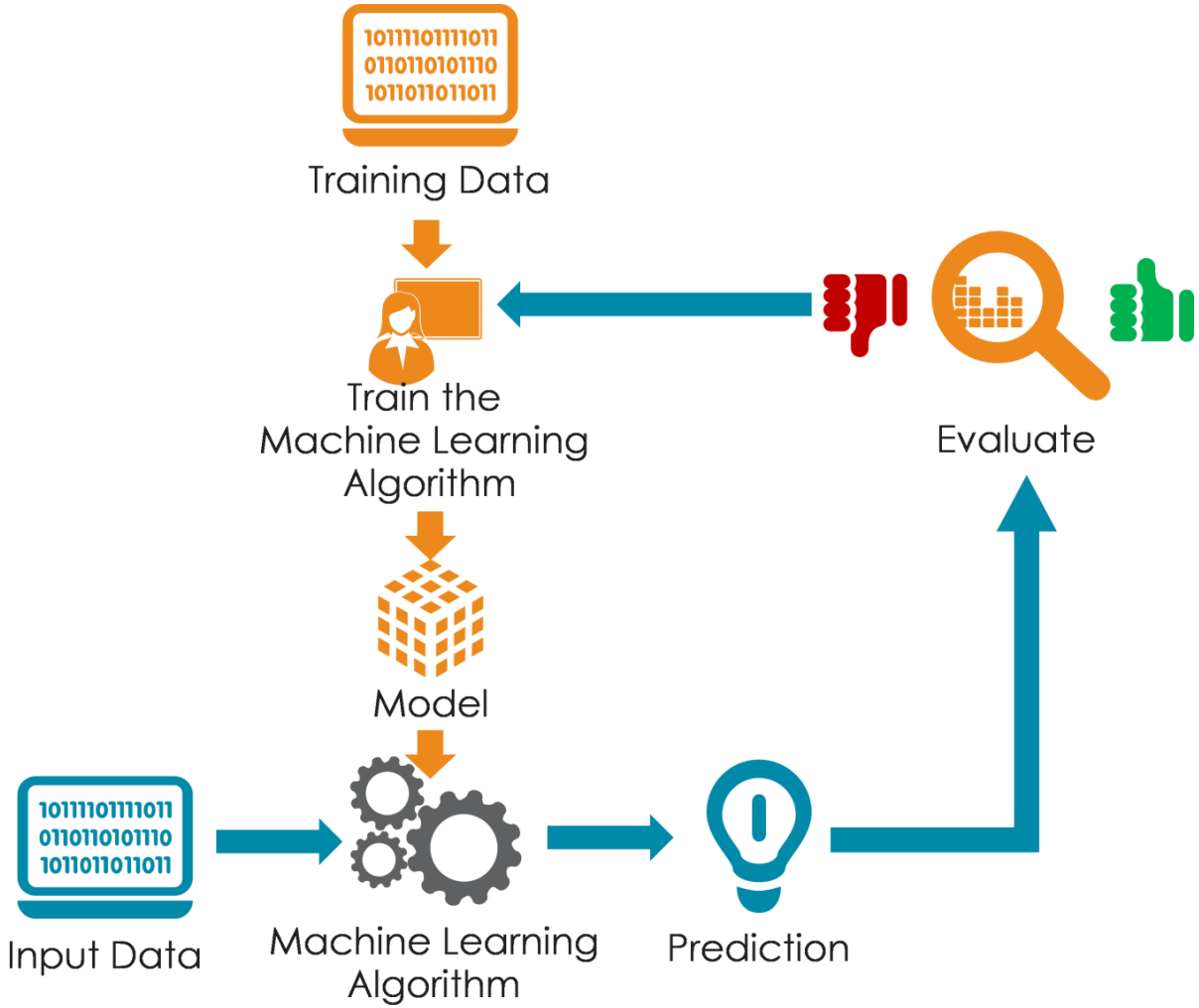
- f é o preditor;
- X é a matriz de exemplos;
- y é o vetor de rótulos;
- $\ell(y_n, \hat{y}_n)$ é a função de perda;
- N é o número de exemplos;
- $\hat{y}_n = f(x_n, \theta)$ é a predição do modelo para o exemplo x_n.

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.

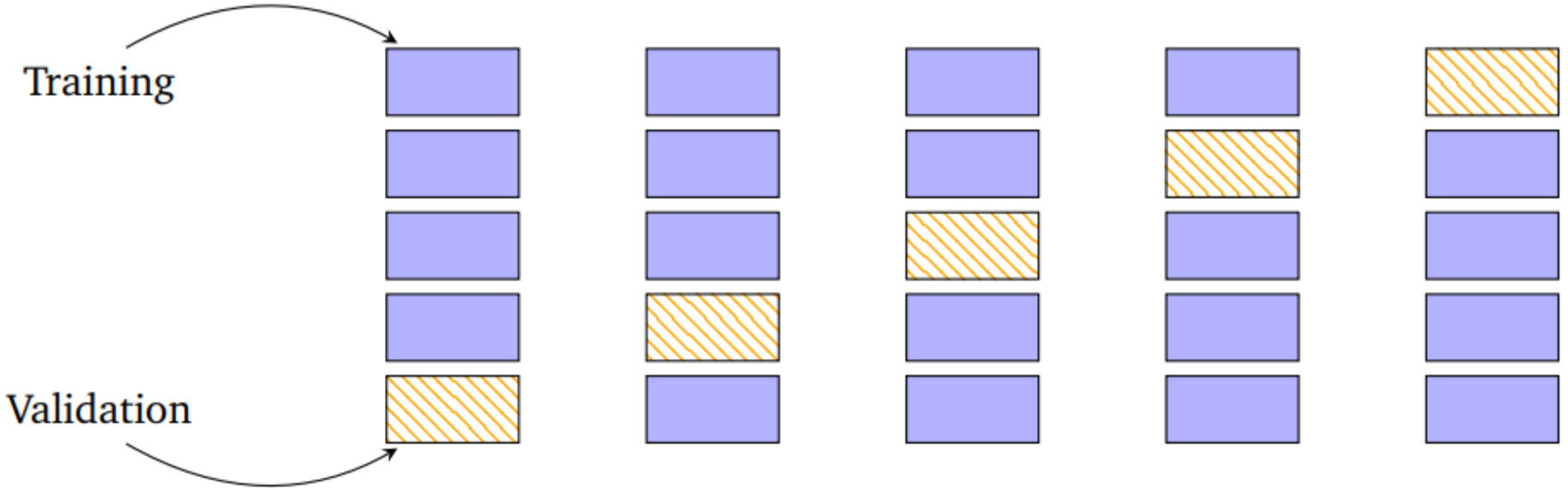
Minimização de Risco Empírico

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*

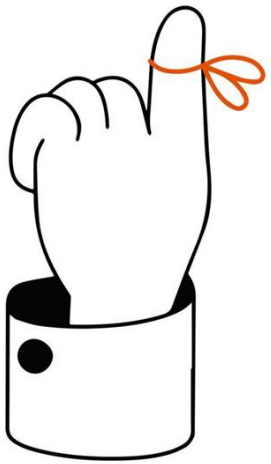


Validação cruzada *K-Fold*. *Folds* de treinamento (azul) e *fold* de validação (laranja listrado).

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



Estimativa de parâmetros e Modelagem Probabilística e Inferência

André M. Riccioppo

Estimativa de parâmetros

- EMV (Estimação de Máxima Verossimilhança)
- Verossimilhança: função a partir dos parâmetros que permite encontrar um modelo que se adeque bem aos dados
- Problema focado no logaritmo negativo da função
  - Desejamos maximizar a verossimilhança
  - Otimização numérica tende a estudar minimização de funções

Estimação de Máxima Verossimilhança

- $\mathcal{L}_x(\theta) = -\log p(x|\theta)$
  - x : variável randômica
  - $p(x|\theta)$: família de densidades de probabilidades parametrizadas por (θ)
- Parâmetro θ está variando e o dado x está fixo.
  - Comum descartar referência a x e escrever como função apenas de θ, quando a variável randômica está livre de contexto.
- A interpretação da densidade de $p(x|\theta)$ se dá usando um valor fixo de θ
  - Escolhemos as configurações que mais “provavelmente” geraram os dados
- $\mathcal{L}(\theta)$ diz o quão provável é uma configuração de θ para as observações x

Estimação de Máxima Verossimilhança

- Cenário de aprendizado supervisionado, onde temos

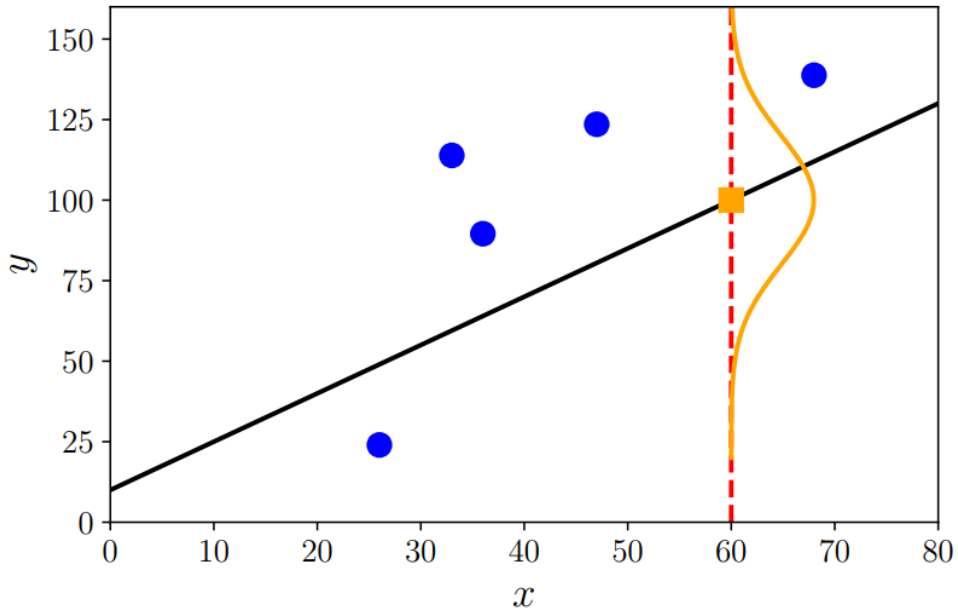
$$(x_1, y_1), \ldots, (x_N, y_N),\ x_n \in \mathbb{R}^D \text{ e } y_n \in \mathbb{R}$$

- Queremos um preditor que a partir de um vetor com características x_n produza uma predição y_n.

- Exemplo: especificação da probabilidade condicionada dos rótulos dada uma distribuição Gaussiana em que a verossimilhança para cada par (x_n, y_n) pode ser especificada como $\mathrm{p}(y_n|x_n, \theta) = \mathcal{N}(y_n|x_n^T\theta, \sigma^2)$

Estimação de Máxima Verossimilhança

- A figura mostra a incerteza preditiva em x=60.



Estimação de Máxima Verossimilhança

- Assumimos que os exemplos são independentes e identicamente distribuídos.
  - Independentes: probabilidade envolvendo ($\mathcal{Y} = \{y_1, \ldots, y_N\}$ e $\mathcal{X} = \{x_1, \ldots, x_N\}$) pode ser fatorizada em produto de probabilidades de cada exemplo individual $p(\mathcal{Y}|\mathcal{X}, \theta) = \prod_{n=1}^{N} p(y_n|x_n, \theta)$
  - Identicamente distribuídos: cada termo no produto desta fórmula faz parte da mesma distribuição e todos eles compartilham os mesmos parâmetros

Estimação de Máxima Verossimilhança

- Para otimização, mais fácil calcular funções decompostas em somas de funções mais simples.
- Considerando $\log(ab) = \log(a) + \log(b)$

$$\mathcal{L}(\theta) = -\log p(\mathcal{Y}|\mathcal{X}, \theta) = -\sum_{n=1}^{N} \log p(y_n|x_n, \theta)$$

- Continuando o raciocínio sobre a verossimilhança Gaussiana, a log-verossimilhança negativa pode ser reescrita como:

$$\mathcal{L}(\theta) = -\sum_{n=1}^{N} \log p(y_n|x_n, \theta) = -\sum_{n=1}^{N} \log \mathcal{N}(y_n|x_n^T\theta, \sigma^2) =$$

$$-\sum_{n=1}^{N} \log \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{\left(y_n - x_n^T\theta\right)^2}{2\sigma^2}\right) =$$

$$-\sum_{n=1}^{N} \log \exp\left(-\frac{\left(y_n - x_n^T\theta\right)^2}{2\sigma^2}\right) - \sum_{n=1}^{N} \log \frac{1}{\sqrt{2\pi\sigma^2}}$$

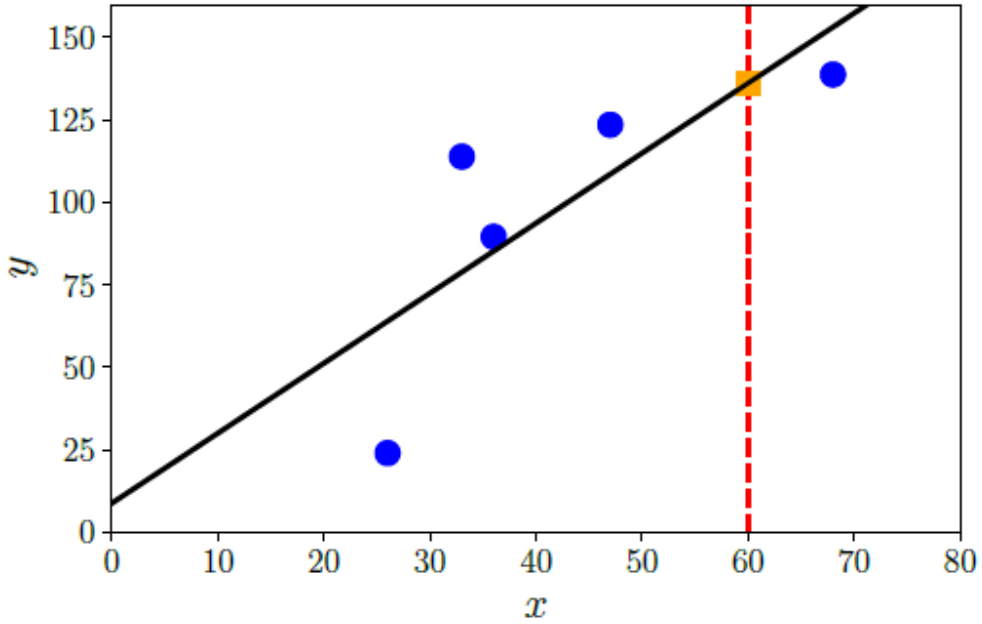
Estimação de Máxima Verossimilhança

$$\mathcal{L}(\theta) = (\dots) = \frac{1}{2\sigma^2}\sum_{n=1}^{N}(y_n - x_n^T\theta)^2 - \sum_{n=1}^{N}\log\frac{1}{\sqrt{2\pi\sigma^2}}$$

- Como σ é conhecido, o segundo termo da equação é constante.
- Portanto, minimizar $\mathcal{L}(\theta)$ corresponde à estimação de máxima verossimilhança.

Estimação de Máxima Verossimilhança

- A figura abaixo mostra um conjunto de dados de regressão e uma função que é induzida por parâmetros de verossimilhança máxima



Estimação de Máxima a Posteriori

- Se tivermos conhecimento anterior sobre a distribuição de parâmetros θ, podemos multiplicar um termo adicional à verossimilhança: a distribuição a priori dos parâmetros $\mathrm{p}(\theta)$

- O Teorema de Bayes traz ferramenta para atualizar uma distribuição de probabilidades de variáveis aleatórias.
  - Podemos calcular a distribuição posterior $\mathrm{p}(\theta|x)$ partindo da distribuição a priori $\mathrm{p}(\theta)$ e da função de verossimilhança $\mathrm{p}(x|\theta)$ que vincula os parâmetros e os dados observados

$$\mathrm{p}(\theta|x) = \frac{\mathrm{p}(x|\theta)\mathrm{p}(\theta)}{\mathrm{p}(x)}$$

Estimação de Máxima a Posteriori

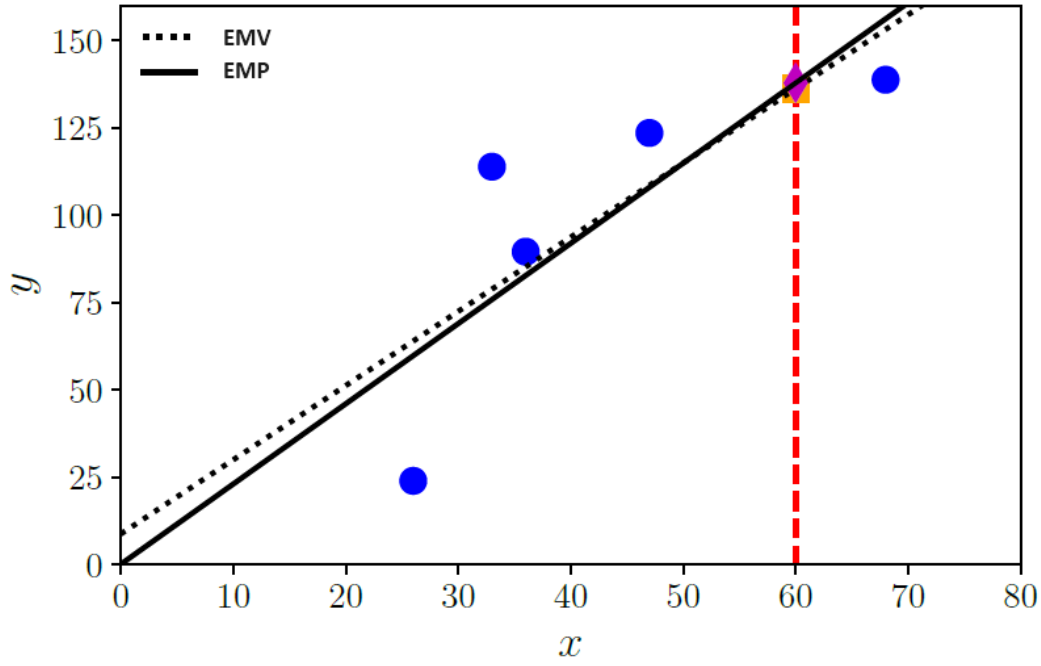
- Queremos encontrar θ que maximiza essa distribuição posterior.
- Como $p(x)$ não depende de θ, podemos ignorar o valor do denominador para a otimização, obtendo:

$$p(\theta|x) \propto p(x|\theta)p(\theta)$$

- Estimativa da máxima a posteriori (EMP): Ao invés de estimar o mínimo da log-verossimilhança negativa, vamos estimar o mínimo da log-posterior negativa.

Estimação de Máxima a Posteriori

- A figura abaixo mostra o efeito de adicionar uma priori Gaussiana de média 0 no conjunto de dados da figura anterior



Estimação de Máxima a Posteriori

- A ideia de incluir conhecimento a priori de onde estão os melhores parâmetros é amplamente utilizada em aprendizagem de máquina
- Visão alternativa é a ideia de regularização, que introduz um termo adicional com um viés que aproxima os parâmetros à origem
- Estimação de máxima a posteriori pode ser vista como ponte entre abordagens probabilísticas e não probabilísticas

Ajuste do Modelo

- Temos um conjunto de dados e queremos ajustar um modelo parametrizado a esses dados.
  - Ajuste = otimizar ou aprender os parâmetros do modelo minimizando uma função de perda (ex.: log-verossimilhança negativa)
  - EMV e EMP são dois algoritmos usados para isso

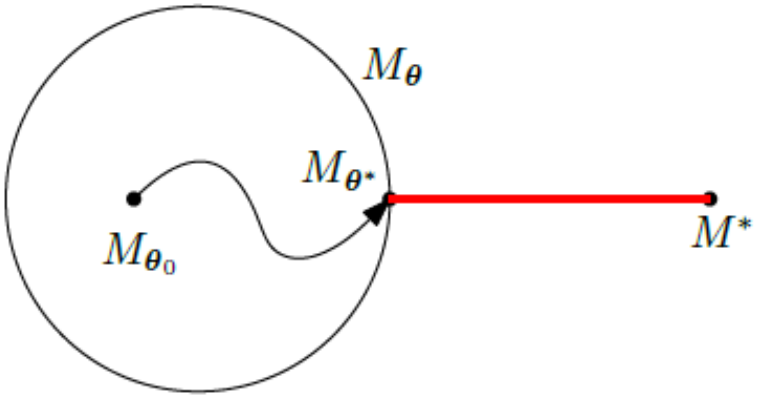
- A parametrização do modelo define a classe de modelos M_θ com a qual vamos operar.

Ajuste do Modelo

- Exemplo: em uma regressão linear, define que a relação entre entradas x e observações y seja y = ax + b.
  - $\theta := \{a, b\}$ são os parâmetros do modelo
  - Nesse caso, família de funções afins

- Se temos dados de um modelo M^* desconhecido, otimizamos θ de forma que M_θ seja o mais próximo possível de M^*.
  - Proximidade é definida pela função objetiva que vamos otimizar

Ajuste do Modelo

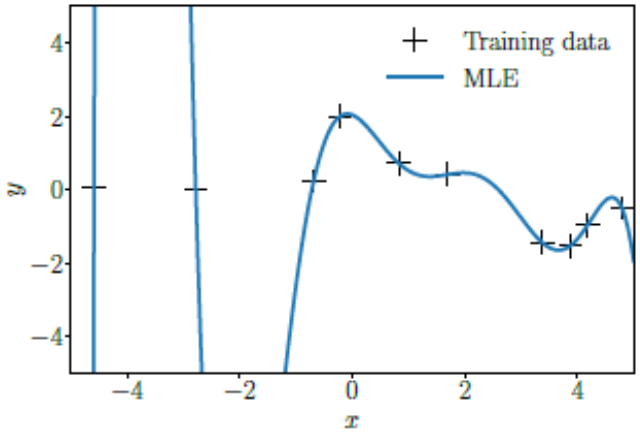


Ajuste do Modelo

- Depois da otimização, podemos distinguir três casos diferentes:
  - Overfit
  - Underfit
  - Ajuste adequado

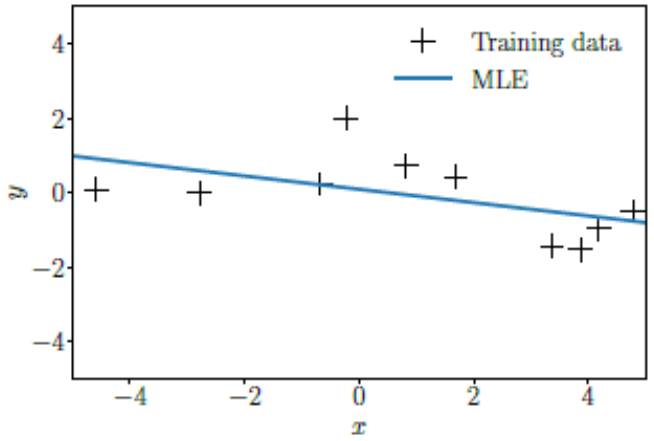
Ajuste do Modelo
Overfit

- A classe de modelos parametrizada é muito rica para modelar o conjunto de dados gerado por M^*
- Exemplo:
  - M^*: gerado por função linear
  - M_θ: definido por classe de polinômios de sétima ordem



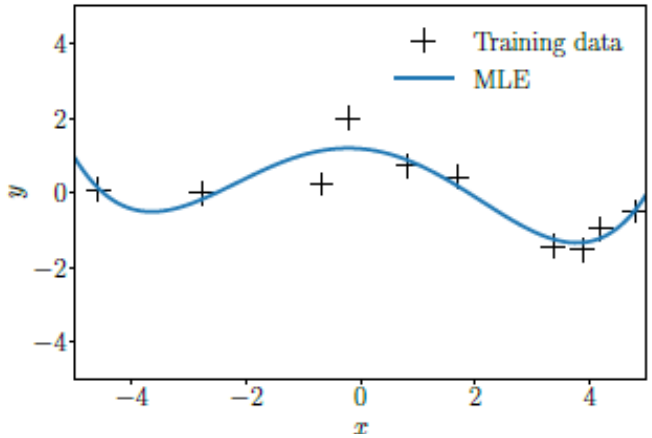
Ajuste do Modelo Underfit

- A classe de modelos parametrizada não é rica o suficiente para modelar o conjunto de dados gerado por M^*
- Exemplo:
  - M^*: gerado por função sinusoidal
  - M_θ: parametriza linhas retas



Ajuste do Modelo
Ajuste adequado

- A classe de modelos parametrizada é adequada e rica o suficiente para descrever o conjunto de dados



Modelagem Probabilística e Inferência

- Em aprendizado de máquina geralmente queremos interpretar os dados para predizer eventos futuros ou auxiliar na tomada de decisão.
  - Então construímos modelos que descrevem o processo generativo dos dados

- Exemplo: descrever saídas de um experimento de jogar moeda em 2 passos.
  - 1) Definimos um parâmetro μ, descrevendo a probabilidade de “cara” como parâmetro de distribuição de Bernoulli.
  - 2) Obtemos amostra de uma experiência $\mathrm{x} \in$ {cara, coroa} da distribuição de Bernoulli $p(x|\mu) = Ber(\mu)$

Modelagem Probabilística e Inferência

- O parâmetro μ dá origem a um conjunto de dados X e depende da moeda utilizada.
- μ não é conhecido anteriormente e não pode ser observado diretamente, então precisamos de mecanismos para aprender sobre ele a partir de observações obtidas de experimentos.
- A modelagem probabilística pode ser usada para isso.

Modelos Probabilísticos

- Modelos probabilísticos representam aspectos incertos de um experimento como distribuições probabilísticas.

- Benefício: ferramentas da teoria da probabilidade para modelagem inferência, predição e seleção de modelos.

Modelos Probabilísticos

- Distribuição conjunta das variáveis observadas x e dos parâmetros ocultos θ são importantes e encapsulam as seguintes informações:
  - Verossimilhança e probabilidade a priori
  - A verossimilhança marginal $p(x)$ pode ser calculada usando a distribuição conjunta e integrando seus parâmetros
  - A distribuição posterior, que pode ser obtida dividindo a distribuição conjunta pela verossimilhança marginal.
- O modelo probabilístico é especificado pela distribuição conjunta de todas suas variáveis aleatórias.

Inferência Bayesiana

- Uma das principais tarefas em aprendizado de máquina é descobrir o valor das variáveis ocultas θ associadas ao modelo e aos dados a partir das variáveis observadas x.

- Usando a EVM ou a EMP, o algoritmo de estimação de parâmetros vai resolver um problema de otimização e obter um valor único que seria o melhor θ.

  - Ou seja, a distribuição preditiva será dada por $p(x|\theta^*)$ onde $\theta*$ é usado na função de verossimilhança.

Inferência Bayesiana

- Ao focar apenas em algumas estatísticas da distribuição posterior, perdemos informações que poderiam ser críticas em um sistema que utilize a predição para tomar decisões.

- Trabalhar a distribuição posterior completa pode ser útil e levar a decisões mais robustas.

    - Para isso se usa a inferência Bayesiana

Inferência Bayesiana

- Aplicando o teorema de Bayes obtemos a distribuição posterior dos parâmetros θ dado X, a partir de:
  - Conjunto de dados: X
  - Distribuição a priori dos parâmetros θ: $p(\theta)$
  - Verossimilhança de X dados os parâmetros θ: $p(X|\theta)$
  - Evidência ou probabilidade marginal dos dados: $p(X)$

$$p(\theta|X) = \frac{p(X|\theta)p(\theta)}{p(X)}, \qquad p(X) = \int p(X|\theta)p(\theta)d(\theta)$$

- Inverte-se a relação entre θ e os dados X (dados pela verossimilhança)

Inferência Bayesiana

- Implicação de ter uma distribuição posterior é que ela pode ser usada para propagar a incerteza dos parâmetros aos dados, eliminando a dependência dos parâmetros θ, que foram marginalizados ou integrados.

- A equação mostra que a predição é uma média sobre todos os valores plausíveis do parâmetro θ, encapsulada pela probabilidade $p(\theta)$

Inferência Bayesiana

- Comparação com estimação de parâmetros:
  - A estimação de parâmetros por EMV ou EMP fornece um ponto estimado θ^* dos parâmetros e o problema computacional a ser resolvido é uma otimização. A inferência Bayesiana fornece uma distribuição posterior e o problema computacional a ser resolvido é uma integração.
  - Predições com pontos estimados são diretas. Predições no framework Bayesiano exigem resolução de outro problema de integração.
  - Inferência Bayesiana permite incorporar conhecimento anterior. Isso não é feito facilmente no contexto de estimação de parâmetros.

Inferência Bayesiana

- Propagação de incerteza dos parâmetros pode ser importante para sistemas de tomada de decisão.
  - Permite avaliar e explorar riscos presentes nesse contexto.
- Existem desafios decorrentes dos problemas de integração. Podem existir integrais que não são tratadas de forma analítica, não permitindo calcular a distribuição posterior, predições ou verossimilhança máxima de forma fechada.
- Nesses casos podem ser usadas:
  - Aproximações estocásticas: método de Monte Carlo via Cadeias de Markov
  - Aproximações determinísticas: aproximações de Laplace, inferência variacional ou propagação de expectativas.

Inferência Bayesiana

- Aplicação em grande variedade de problemas:
  - Modelagem de tópicos em grande escala
  - Predição de cliques
  - Aprendizado por reforço em sistemas de controle
  - Sistemas de ranqueamento online
  - Sistemas de recomendação em larga escala

Modelos de Variáveis Latentes

- Na prática, pode ser útil ter variáveis latentes adicionais z como parte do modelo.
- Essas variáveis latentes são diferentes dos parâmetros θ, pois não parametrizam o modelo explicitamente.
- Podem:
  - Participar da descrição do modelo de dados, contribuindo para a capacidade de se interpretar esse modelo.
  - Simplificar a estrutura do modelo, com a utilização de um número menor de parâmetros

Modelos de Variáveis Latentes

- Exemplo de utilização:
  - Principal Component Analysis (PCA) para redução de dimensionalidade
  - Estimação de densidade com modelos de mistura Gaussiana
  - Modelagem de séries temporais com modelos ocultos de Markov ou sistemas dinâmicos
  - Meta aprendizagem e generalização de tarefas

- Por outro lado, apesar de facilitar a estrutura do modelo e do processo generativo, o aprendizado é mais difícil

Modelos de Variáveis Latentes

- Podem ser usados para aprendizado e inferência de parâmetros em um procedimento de dois passos:

  1. Calcula-se a verossimilhança do modelo $p(x|\theta)$, que não depende de variáveis latentes.

  2. Usa-se essa verossimilhança para estimação de parâmetros ou inferência Bayesiana

Modelos de Variáveis Latentes

- Como a função de verossimilhança $p(x|\theta)$ é preditiva dos dados a partir dos parâmetros do modelo, precisamos marginalizar as variáveis latentes:

$$p(x|\theta) = \int p(x|z,\theta)p(z)dz$$

- onde
  - $p(x|z,\theta)$ é conhecido
  - $p(z)$ é o priori das variáveis latentes
- A verossimilhança não deve depender das variáveis latentes z, mas é função apenas dos dados x e dos parâmetros θ do modelo.

Modelos de Variáveis Latentes

- A partir dessa fórmula podemos:
  - Estimar parâmetros usando a verossimilhança máxima
  - Usar estimação de máxima a posteriori diretamente nos parâmetros θ
  - Realizar a inferência Bayesiana da verossimilhança adicionando um priori $p(\theta)$ aos parâmetros e usando o teorema de Bayes para obter uma distribuição posterior usando os parâmetros do modelo, conforme a fórmula:

$$p(\theta|X) = \frac{p(X|\theta)p(\theta)}{p(X)}$$

Essa distribuição posterior pode ser usada para predições no framework de inferência Bayesiana

Modelos de Variáveis Latentes

- Desafio é que se exige a marginalização de variáveis latentes, mas ela não pode ser tratada analiticamente e deve se recorrer a aproximações.

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

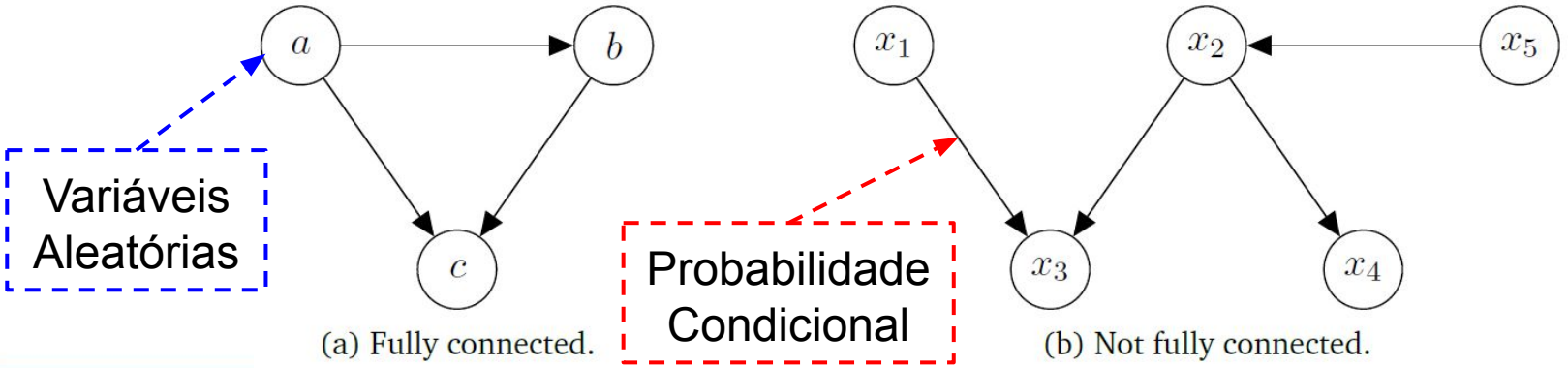
Introdução

- 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

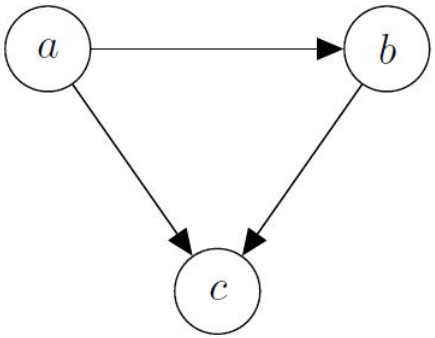
- Visualização da estrutura de um modelo probabilístico
- Usados para projetar novos tipos de modelos estatísticos
- Ilustra propriedades, como independência condicional
- Inferência e aprendizado em modelos estatísticos podem ser expressas em termos de manipulações gráficas



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**



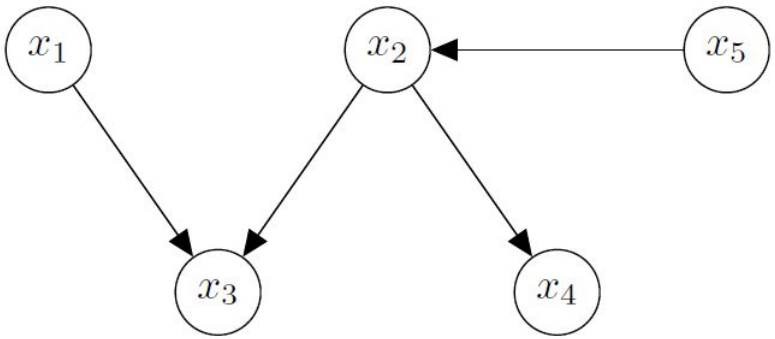
(a) Fully connected.

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

$$p(x1, x2, x3, x4, x5) = p(x1)\, p(x5)\, p(x2 \mid x5)\, p(x3 \mid x1, x2)\, p(x4 \mid x2)$$



(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:

$$\mathcal{A} \perp\!\!\!\perp \mathcal{B} \,|\, \mathcal{C}$$

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

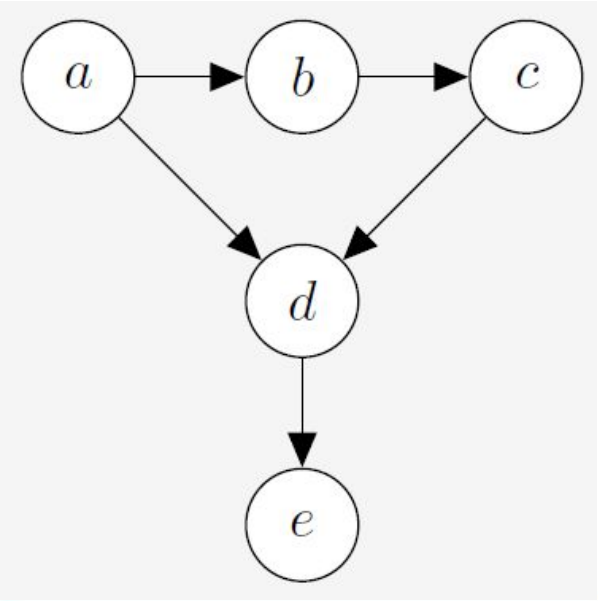
Independência condicional

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

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



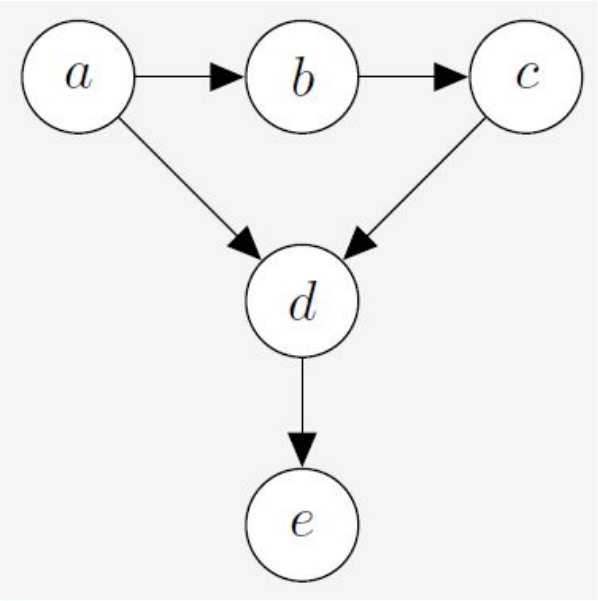
Independência condicional

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

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



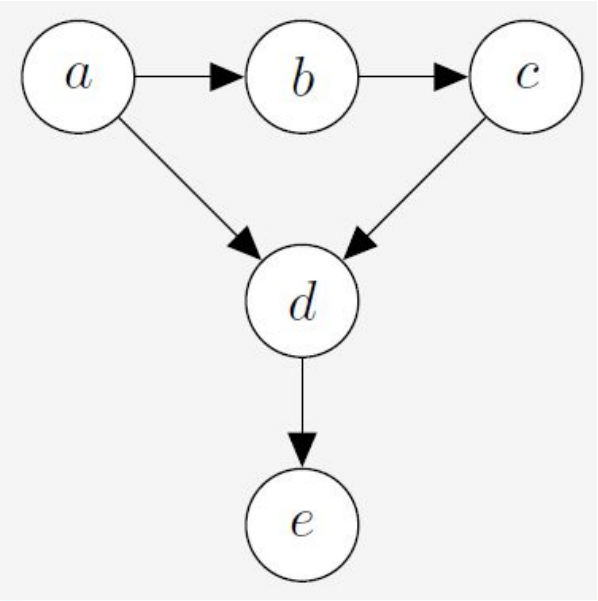
Independência condicional

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

$b \not\perp d \mid c$



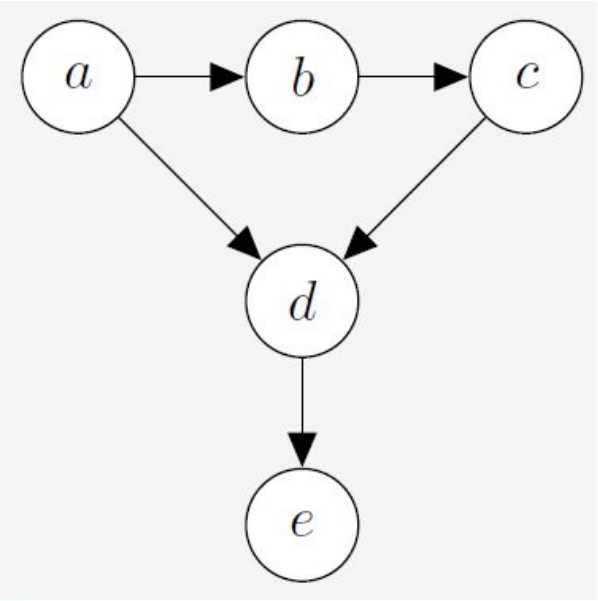
Independência condicional

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

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



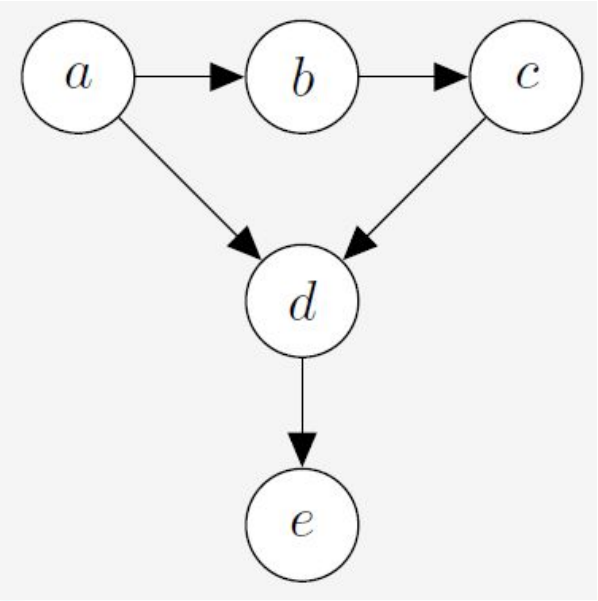
Independência condicional

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

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



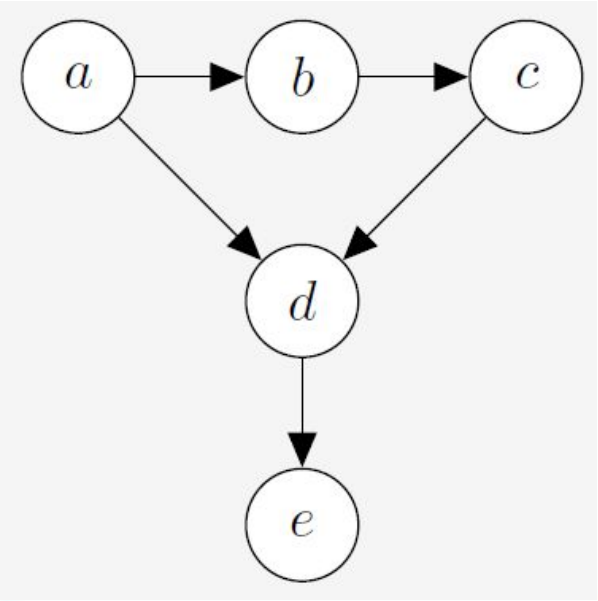
Independência condicional

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

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



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

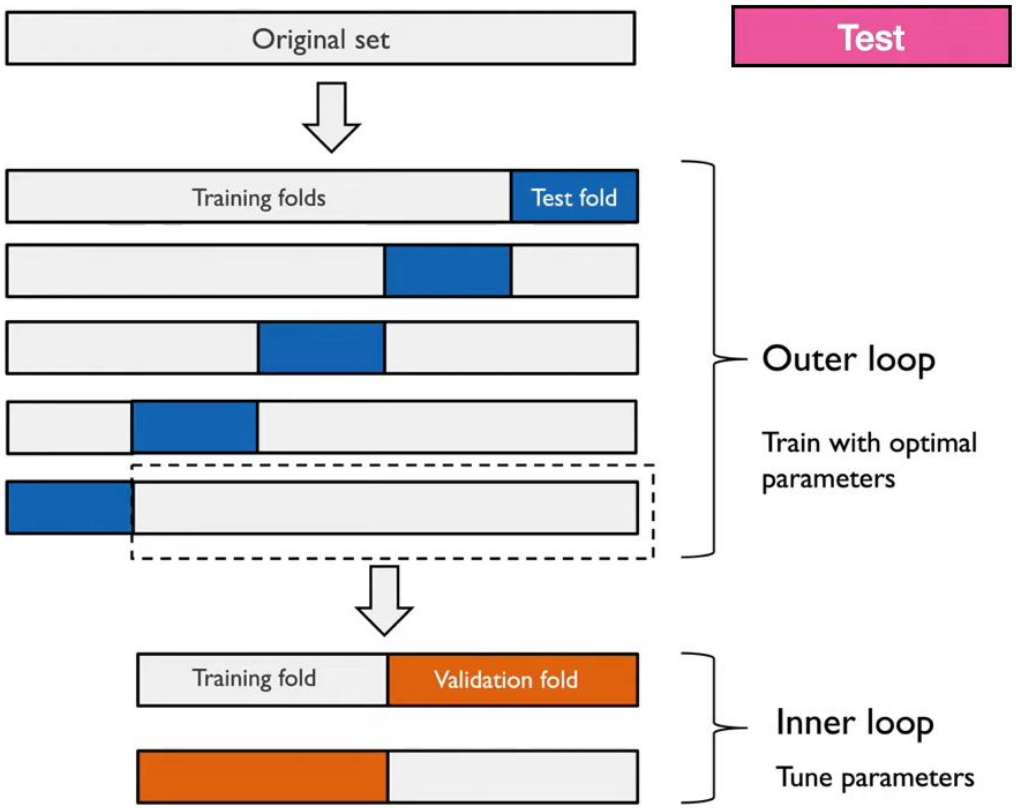
- Um problema comum em ML é a avaliação do modelo
- Durante o treinamento o modelo usa apenas os dados de treinamento para avaliar seu desempenho
- O desempenho nos dados de treinamento não é interessante
- Estimação máxima da verossimilhança pode levar ao *overfitting*
- O interessante é o desempenho 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

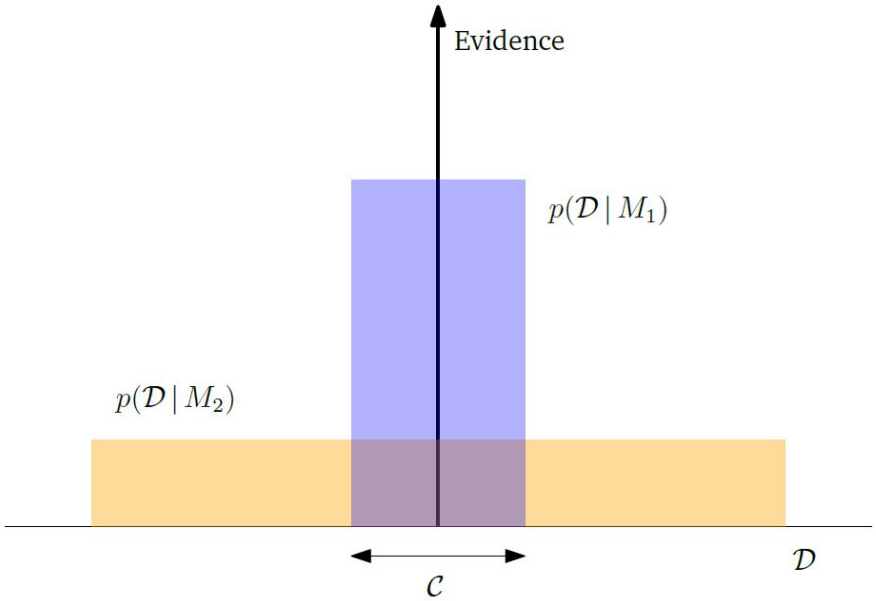
Seleção de modelos Bayesianos

- 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

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

- 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