Aprendizado de Máquina II

Redes Bayesianas

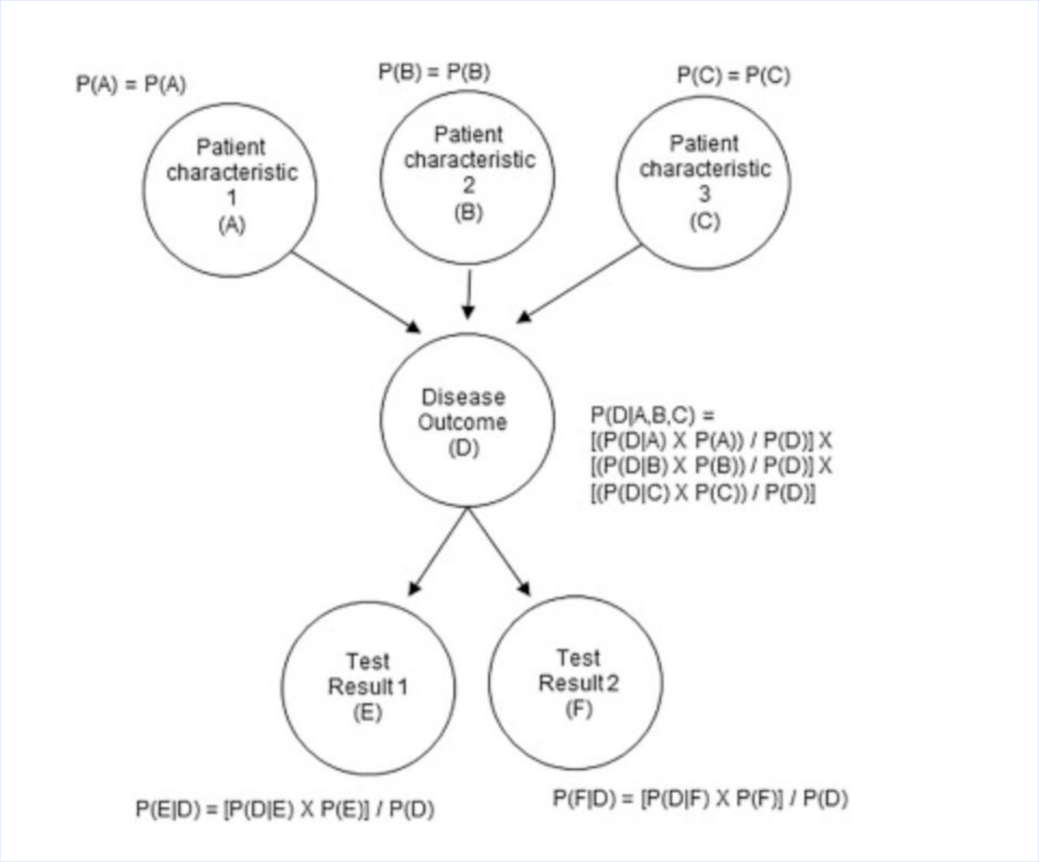
Profª. Carine Webber

Especialização em Ciência de Dados

2024

UCS
UNIVERSIDADE
DE CAXIAS DO SUL

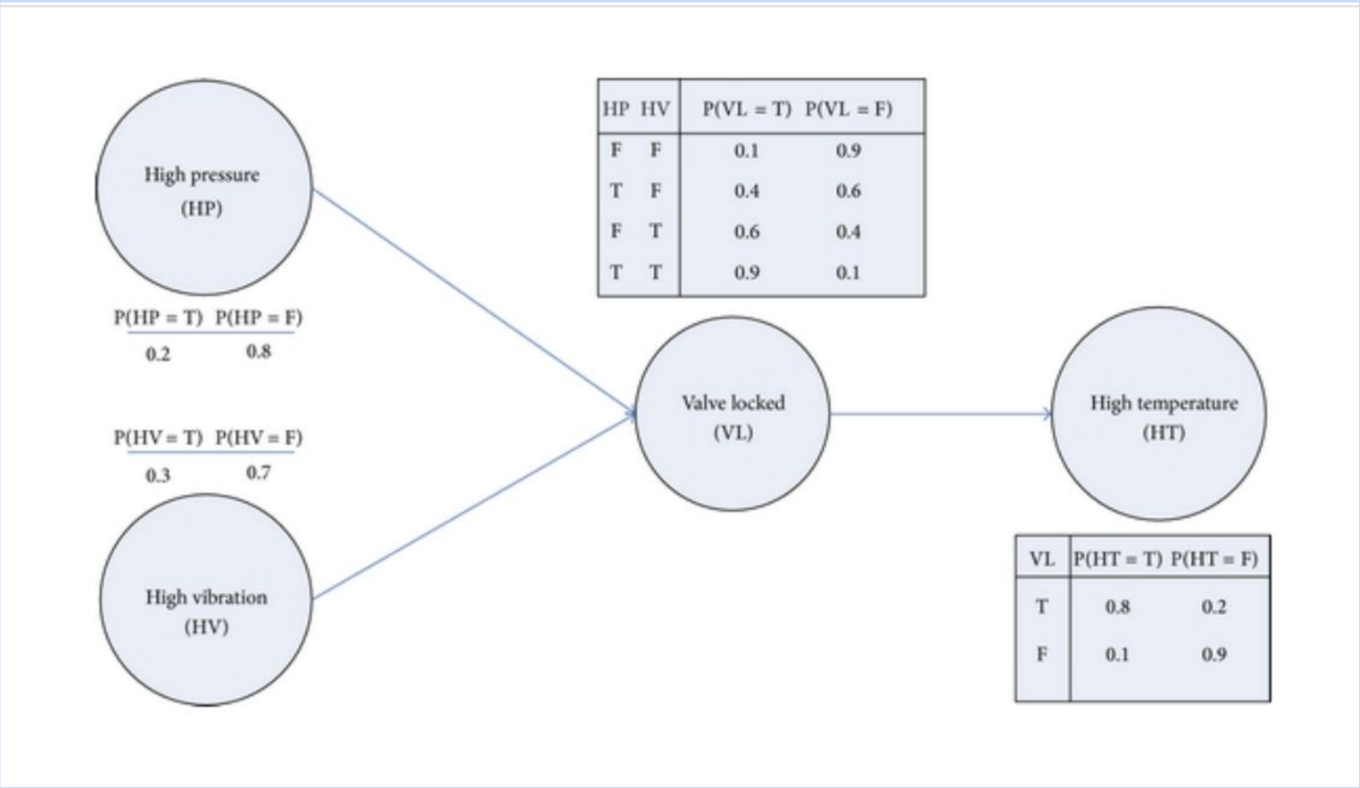
Exemplo na Medicina



https://www.valueinhealthjournal.com/article/S1098-3015(19)30057-9/fulltext

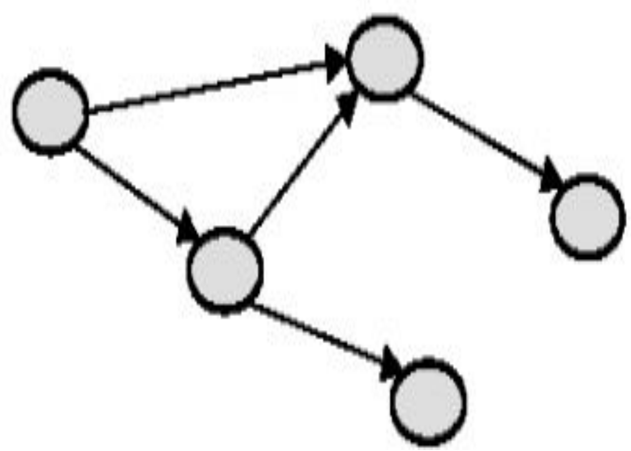
Exemplo na Engenharia

https://www.hindawi.com/journals/mpe/2014/210714/



Roteiro da Aula

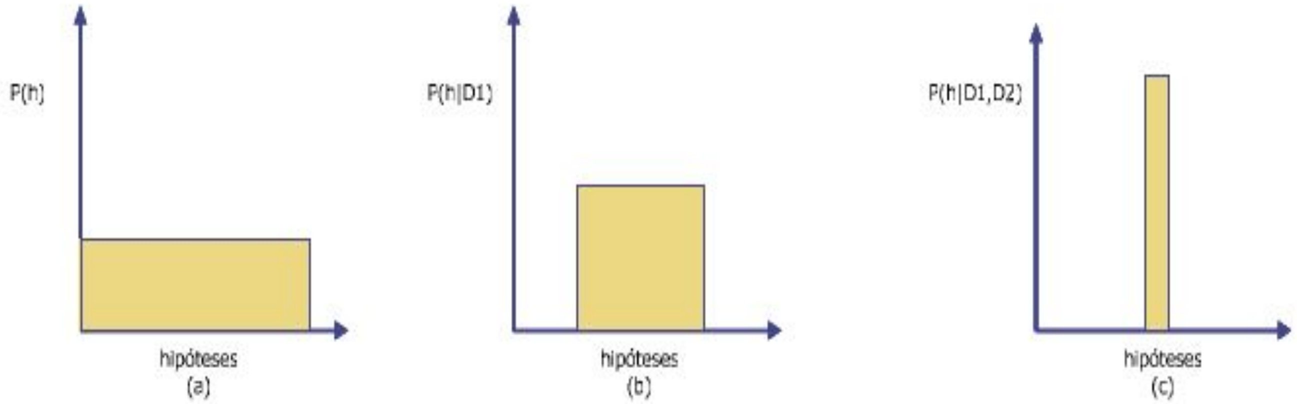
- Teoria das Probabilidades
- Teorema de Bayes
- Aprendizagem Bayesiana
- Implementação no ScikitLearn
- Avaliação do desempenho
- Atividades



Teoria das Probabilidades

- Desenvolvida no séc. XVIII por Thomas Bayes.

- Idéia principal: entender a freqüência com que eventos ocorrem e raciocinar sobre a freqüência de futuras combinações de eventos.



Teorema de Bayes

- **Definição:**
  O Teorema de Bayes é uma fórmula matemática que nos permite calcular a probabilidade de um evento com base em informações condicionais.
- **Fórmula:**

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

- **Explicação:**
  - $P(A|B)$: Probabilidade de A dado que B ocorreu.
  - $P(B|A)$: Probabilidade de B dado que A ocorreu.
  - $P(A)$: Probabilidade de A (a priori).
  - $P(B)$: Probabilidade de B (a priori).
- **Por que é útil?**
  Ele ajuda a atualizar nossas crenças com base em novas evidências.

Exemplo

Problema:

Um teste detecta uma doença com:

- 95% de acertos para quem está doente ($P(Pos|D) = 0.95$).
- 5% de falsos positivos ($P(Pos|\neg D) = 0.05$).
- Apenas 1% da população está doente ($P(D) = 0.01$).

Pergunta: Se o teste der positivo, qual a probabilidade de a pessoa estar realmente doente ($P(D|Pos)$)?

Solução com o Teorema de Bayes:

1. Aplique a fórmula:

$$P(D|Pos) = \frac{P(Pos|D)P(D)}{P(Pos)}$$

2. Calcule $P(Pos)$:

$$P(Pos) = P(Pos|D)P(D) + P(Pos|\neg D)P(\neg D)$$

3. Substitua os valores e descubra a resposta ($P(D|Pos)$).

Demonstração do Raciocínio

1. **Cálculo de $P(Pos)$:**

$$P(Pos) = P(Pos|D)P(D) + P(Pos|\neg D)P(\neg D)$$

   Substituindo os valores:

$$P(Pos) = (0.95 \cdot 0.01) + (0.05 \cdot 0.99) = 0.059$$

2. **Cálculo de $P(D|Pos)$:**

$$P(D|Pos) = \frac{P(Pos|D)P(D)}{P(Pos)}$$

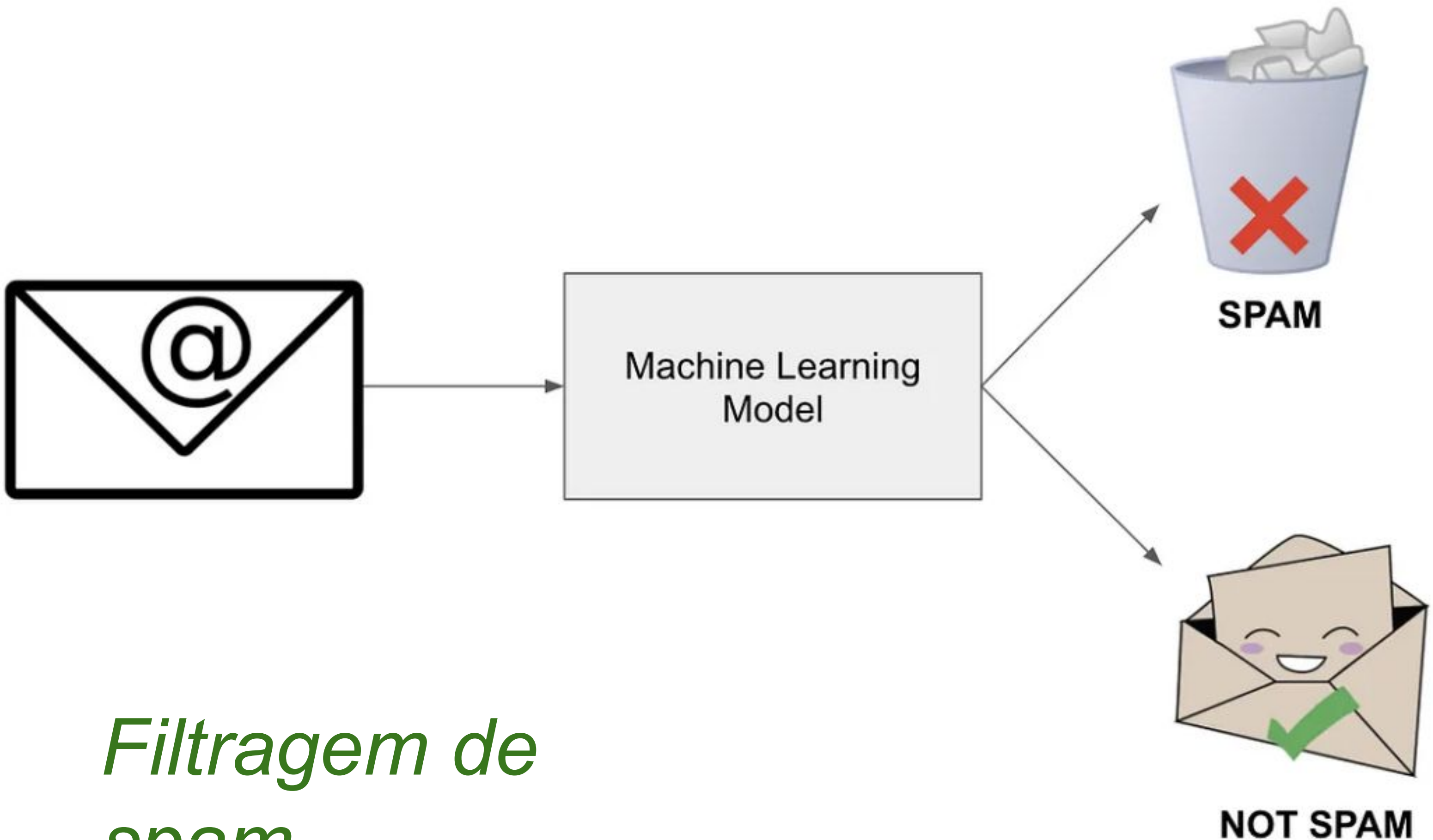
   Substituindo os valores:

$$P(D|Pos) = \frac{0.95 \cdot 0.01}{0.059} \approx 0.161$$

Resultados Finais:

- $P(Pos) = 0.059$ (ou seja, a probabilidade de um teste positivo).
- $P(D|Pos) \approx 16.1\%$ (a probabilidade de estar realmente doente dado que o teste foi positivo).

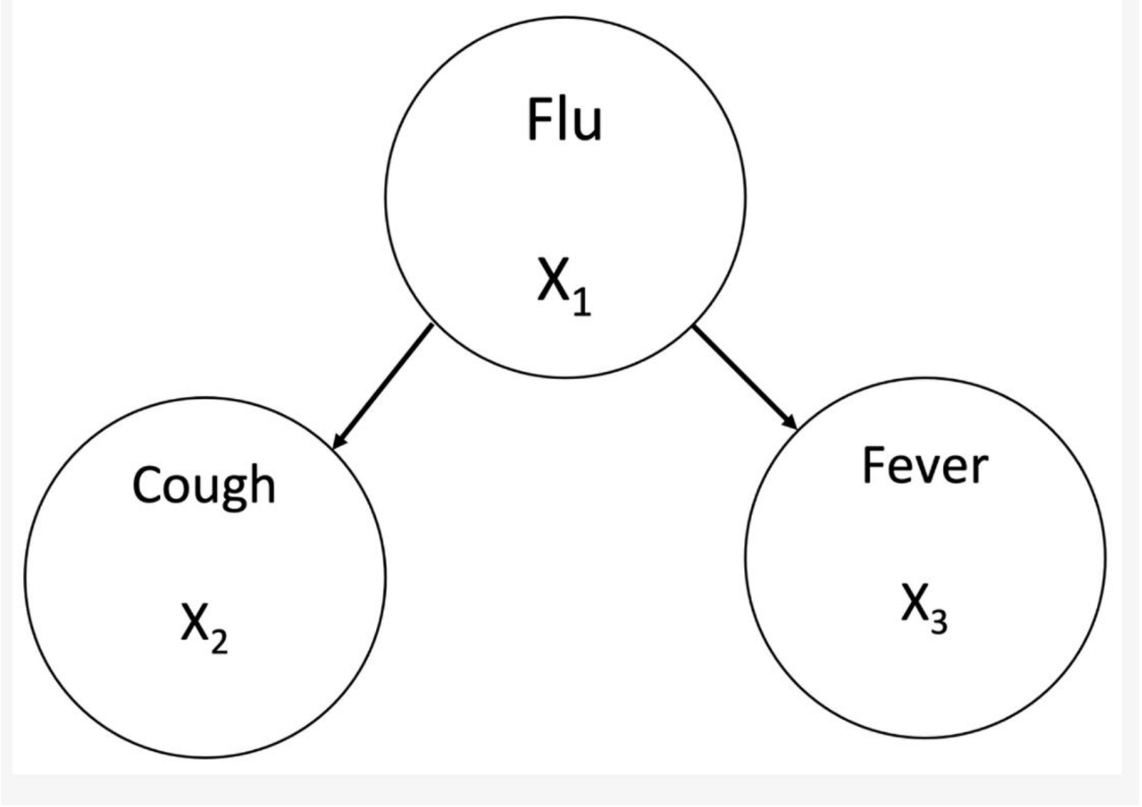
Exemplos



Filtragem de spam

Exemplos

Diagnóstico médico



Raciocínio Bayesiano

- Teorema de Bayes:

$$P(H|E) = \frac{P(E|H) \cdot P(H)}{P(E)}$$

  - Onde:
    - P(H | E) é a probabilidade que H seja verdade dada a evidência E;
    - P(E | H) é a probabilidade de se observar a evidência E, quando H é verdadeira;
    - P(H) é a probabilidade que H seja verdadeira de modo global;
    - P(E) é um

fator de normalização:

$$P(E) = P(E|H) \cdot P(H) + P(E|\neg H) \cdot P(\neg H)$$

Raciocínio Bayesiano

- Exemplo:
  - H = "João tem malária"; E="João tem febre alta"

$$P(malaria \mid febre) = \frac{P(febre \mid malaria) \cdot P(malaria)}{P(febre)}$$

$$P(febre) = P(febre \mid malaria) \cdot P(malaria) + P(febre \mid \neg malaria) \cdot P(\neg malaria)$$

Raciocínio Bayesiano

- Exemplo:
  - Supõe-se: $P(malaria) = 0.0001$

    $P(febre \mid malaria) = 0.75$

    $P(febre \mid \neg malaria) = 0.14$

$$P(febre) = (0.75) \cdot (0.0001) + (0.14) \cdot (0.9999) \approx 0.14006$$

$$P(malaria \mid febre) = \frac{(0.75) \cdot (0.0001)}{0.14006} \approx 0.0005354$$

Aplicação da Regra de Bayes: Diagnóstico Médico

Seja

M=doença meningite

S= rigidez no pescoço

- Um médico sabe:

P(S/M)=0.5

P(M)=1/50000

P(S)=1/20

$$\mathbf{P}(M/S) = \frac{\mathbf{P}(S/M)\mathbf{P}(M)}{\mathbf{P}(S)}$$

$$= \frac{0,5*(1/50000)}{1/20} = 0,002$$

A probabilidade de uma pessoa ter meningite dado que ela está com rigidez no pescoço é 0,02% (1 em 5000).

Teoria das Probabilidades

- Distribuição de probabilidades:

	DorDeDente		¬DorDeDente	
	Choque	¬Choque	Choque	¬Choque
Cárie	0.108	0.012	0.072	0.008
¬Cárie	0.016	0.064	0.144	0.576

$$P(Cárie \mid DorDeDente) = \frac{P(Cárie \wedge DorDeDente)}{P(DorDeDente)}$$

$$= \frac{0.108 + 0.012}{0.108 + 0.012 + 0.016 + 0.064} = 0.6$$

Teoria das Probabilidades

- Probabilidade Incondicional (a *priori*):
  - Grau de confiança associada a uma proposição na ausência de qualquer outra informação.
  - Utilizada somente quando não existem outras informações.
  - P(*Tempo=ensolarado*) = 0.7
  - P(*Tempo=chuvoso*) = 0.2
  - P(*Tempo=frio*) = 0.08
  - P(*Tempo=neve*) = 0.02

Teoria das Probabilidades

- Probabilidade Condicional (a *posteriori*):
  - P(evento|evidência) = "a probabilidade de evento, dado que tudo que se conhece é a evidência."
  - Teorema de Bayes para uma doença e um sintoma:

$$P(d|s) = \frac{P(s|d) \cdot P(d)}{P(s)}$$

Método Naive Bayes ou Bayesiano Ingênuo

- **O que é?**
  O Naive Bayes é um algoritmo baseado no Teorema de Bayes, com a suposição de que as variáveis são **condicionalmente independentes**.
- **Como funciona?**
  Ele calcula a probabilidade de cada classe (C_k) e escolhe a mais provável dado um conjunto de características ($x_1, x_2, \ldots, x_n$).
- **Por que 'Naive'?**
  A palavra "ingênuo" (naive) refere-se à suposição simplificadora de que todas as variáveis são independentes.

Fórmula geral Naive Bayes

Para classificar uma instância com características $x_1, x_2, \ldots, x_n$:

$$P(C_k|x_1, x_2, \ldots, x_n) \propto P(C_k) \prod_{i=1}^{n} P(x_i|C_k)$$

Passos básicos:

1. **Probabilidade da classe:** Calcular $P(C_k)$.
2. **Probabilidade condicional:** Calcular $P(x_i|C_k)$ para cada característica.
3. **Multiplicação:** Combinar as probabilidades e determinar a maior.

(A força do Naive Bayes está em sua simplicidade e eficiência.)

Implementação no Scikit Learn

- **Definição:**
  Um algoritmo probabilístico baseado no teorema de Bayes, assumindo independência condicional entre as características.
- **Principais Tipos no scikit-learn:**
  1. `GaussianNB`: Para dados contínuos (assume distribuição normal).
  2. `MultinomialNB`: Para dados discretos (e.g., contagem de palavras em NLP).
  3. `BernoulliNB`: Para dados binários ou booleanos.
  4. `ComplementNB`: Variante do MultinomialNB para lidar com desbalanceamento.
  5. `CategoricalNB`: Para variáveis categóricas.
- **Vantagens:** Simples, rápido e eficaz para classificações básicas.
- **Limitações:** Assunção de independência condicional nem sempre é realista.

GaussianNB

- **Quando Usar:**
  - Para **dados contínuos** que seguem (ou aproximadamente seguem) uma **distribuição normal (gaussiana)**.
  - Exemplos:
    - Previsão de notas de alunos com base em idade, horas de estudo, etc.
    - Classificação de espécies com base em medições contínuas, como o dataset Iris.
- **Características:**
  - Calcula probabilidades condicionais assumindo uma distribuição normal para cada característica.
  - Pode ser usado em problemas onde as variáveis não seguem exatamente uma distribuição normal, mas funciona melhor se a suposição for razoável.
- **Hiperparâmetro Principal:**
  - `var_smoothing`: Um pequeno valor é adicionado à variância para evitar problemas numéricos com variâncias próximas de zero.

Hiperparâmetros GaussianNB

2.1 GaussianNB

- `var_smoothing`:
    - **Descrição:** Fração adicionada à variância de cada característica para melhorar a estabilidade numérica.
    - **Valor padrão:** `1e-9`.
    - **Impacto:** Valores maiores suavizam os cálculos, prevenindo divisões por zero em dados com variâncias muito pequenas.

MultinomialNB

- **Quando Usar:**
    - Para **dados discretos**, como contagens ou frequências.
    - Exemplos:
        - Problemas de **classificação de texto** (como spam ou análise de sentimentos).
        - Modelagem de ocorrências, como número de cliques em anúncios online.
- **Características:**
    - Assume que as características são contagens ou valores positivos.
    - Ideal para **Bag-of-Words** ou **TF-IDF** em problemas de NLP.
- **Hiperparâmetros Principais:**
    - `alpha`: Suavização de Laplace para evitar probabilidades zero (valor padrão: 1.0).
    - `fit_prior`: Define se as probabilidades a priori devem ser aprendidas dos dados.

Hiperparâmetros

2.2 MultinomialNB

- `alpha`:
  - **Descrição:** Parâmetro de suavização de Laplace/Lidstone.
  - **Valor padrão:** `1.0`.
  - **Impacto:** Evita probabilidades zero para palavras ausentes no treinamento (em problemas de texto, por exemplo).
  - **Observação:** Valores menores podem levar a overfitting.
- `fit_prior`:
  - **Descrição:** Define se as probabilidades a priori devem ser aprendidas dos dados ou assumidas uniformes.
  - **Valor padrão:** `True`.
  - **Impacto:** Desabilitar (`False`) pode ser útil para classes equilibradas.

BernoulliNB

- **Quando Usar:**
  - Para **dados binários** (0 ou 1) ou quando as características podem ser tratadas como presença/ausência.
  - Exemplos:
    - Classificação de texto com indicadores binários (se uma palavra aparece ou não em um documento).
    - Dados booleanos em diagnósticos médicos.
- **Características:**
  - Funciona como uma versão binária do MultinomialNB.
  - Útil quando você quer diferenciar entre presença/ausência, em vez de frequências.
- **Hiperparâmetros Principais:**
  - `alpha`: Suavização de Laplace.
  - `binarize`: Define o limiar para binarizar os valores das características. Exemplo: $binarize = 0.5$ transforma todos os valores > 0.5 em 1 e $<= 0.5$ em 0.

Hiperparâmetros

2.3 BernoulliNB

- `alpha`:
  - Igual ao `MultinomialNB`.
- `binarize`:
  - **Descrição:** Limite usado para binarizar as características (valores maiores que este limite serão tratados como 1, e menores ou iguais, como 0).
  - **Valor padrão:** `0.0`.
  - **Impacto:** Importante para dados que não são binários por natureza. Um valor de `None` desabilita a binarização.
- `fit_prior`:
  - Igual ao `MultinomialNB`.

ComplementNB

- **Quando Usar:**
  - Uma variante do MultinomialNB para lidar com **dados desbalanceados**.
  - Exemplos:
    - Classificação de texto quando algumas classes têm muitos mais exemplos do que outras.
- **Características:**
  - Calcula probabilidades complementares, priorizando classes menos representadas.
  - Melhora a precisão em classes minoritárias sem prejudicar classes majoritárias.
- **Hiperparâmetros Principais:**
  - `alpha`: Suavização.
  - `norm`: Se as probabilidades complementares devem ser normalizadas (valor padrão: True).

Hiperparâmetros

2.4 ComplementNB

- `alpha`:
  - Igual ao `MultinomialNB`.
- `norm`:
  - **Descrição:** Se as probabilidades complementares devem ser normalizadas.
  - **Valor padrão:** `True`.
  - **Impacto:** Normalização ajuda a evitar problemas de escala.

2.5 CategoricalNB

- `alpha`:
  - Igual ao `MultinomialNB`.

CategoricalNB

- **Quando Usar:**
  - Para **variáveis categóricas** em vez de contínuas ou discretas.
  - Exemplos:
    - Classificação de clientes com base em atributos categóricos como "idade", "localização" e "preferências".
- **Características:**
  - Trata cada característica como uma categoria nominal.
  - Requer que os dados sejam pré-processados em categorias ou rótulos numéricos.
- **Hiperparâmetros Principais:**
  - `alpha`: Suavização.

Hiperparâmetros

Modelo	Hiperparâmetro	Descrição	Valor Padrão
GaussianNB	`var_smoothing`	Fração adicionada à variância	`1e-9`
MultinomialNB	`alpha`	Suavização de Laplace	`1.0`
	`fit_prior`	Considerar probabilidades a priori	`True`
BernoulliNB	`alpha`	Suavização de Laplace	`1.0`
	`binarize`	Limite para binarização das características	`0.0`
	`fit_prior`	Considerar probabilidades a priori	`True`
ComplementNB	`alpha`	Suavização de Laplace	`1.0`
	`norm`	Normalizar probabilidades complementares	`True`
CategoricalNB	`alpha`	Suavização de Laplace	`1.0`

Redes Bayesianas

- RB foram introduzidas por Judea Pearl (1988).
- RB surgiram como um método para lidar com informações incertas e probabilísticas.
- RB foram geradas do casamento entre a Teoria das Probabilidades e a Teoria dos Grafos.
- Aplica-se a problemas que envolvam processos de decisão, tais como classificação, reconhecimento de padrões, reconhecimento de sequencias, direcionamento de robôs, modelagem de CAD,dentre outras.
- Exemplo: Softwares de filtragem de SPAMS.

Redes Bayesianas

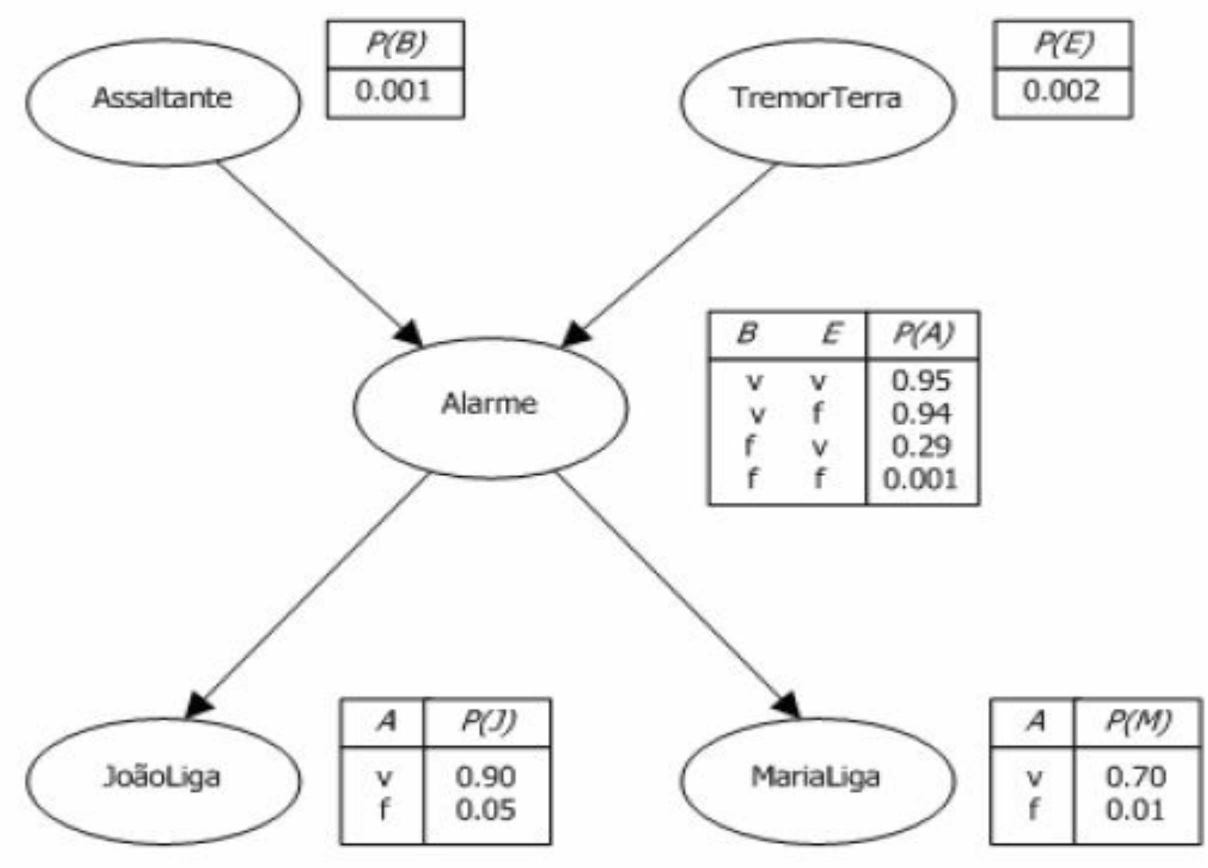
- Um conjunto de variáveis representadas por nós;

- Um conjunto de setas conectando pares de nós.
  - Um nó X é dito pai de Y, se existe uma seta com origem em X e chegada em Y;

- Cada nó X possui uma distribuição de probabilidades que quantifica os efeitos dos seus pais sobre este nó;

- Todos os nós e setas devem formar um grafo dirigido acíclico.

Redes Bayesianas-Exemplo



P(B)
0.001

P(E)
0.002

B	E	P(A)
v	v	0.95
v	f	0.94
f	v	0.29
f	f	0.001

A	P(J)
v	0.90
f	0.05

A	P(M)
v	0.70
f	0.01

Redes Bayesianas-Exemplo

- Qual a probabilidade de ocorrer a situação em que o alarme tenha tocado (a), mas não ocorreu nem tremor (e) e nem invasão de assaltante (b), e João (j) e Maria (m) ligam para avisar ?

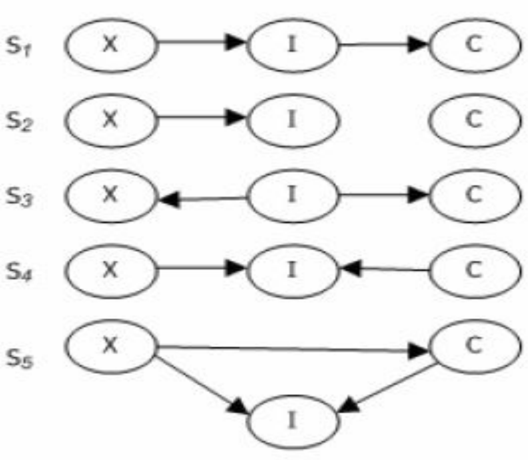
$$P(j \wedge m \wedge a \wedge \neg b \wedge \neg e)$$

$$= P(j|a) P(m|a) P(a|\neg b \wedge \neg e) P(\neg b) P(\neg e)$$

$$= 0.90 * 0.70 * 0.001 * 0.999 * 0.098 = 0.00062$$

Aprendizagem de Redes Bayesianas

- Busca e pontuação
  - Pontuar as redes candidatas através de uma métrica de pontuação.
  - Realizar uma busca heurística para encontrar a estrutura ótima.
  - Exemplo:



Engenharia do conhecimento para Redes Bayesianas

1. Escolher um conjunto de variáveis relevantes que descrevam o domínio

2. Ordem de inclusão dos nós na rede

   (a). causas como “raízes” da rede

   (b). variáveis que elas influenciam

   (c). folhas, que não influenciam diretamente nenhuma outra variável.

3. Enquanto houver variáveis a representar:

   (a). escolher uma variável X_i e adicionar um nó para ela na rede

   (b). estabelecer Pais(X_i) dentre os nós que já estão na rede, satisfazendo a propriedade de dependência condicional

   (c). definir a tabela de probabilidade condicional para X_i

Bibliografia

- Russel, S, & Norvig, P. (1995). Artificial Intelligence: a Modern Approach (AIMA) Prentice-Hall. Pages 436-458, 588-593

- Mitchell, T. & (1997). Machine Learning, McGraw-Hill. Cap.6

- Fayyad et al. (1996). Advances in knowledge discovery and data mining, AAAI Press/MIT Press. Cap.11

- Pearl, J. (1988) Probabilistic Reasoning in Inteligent Systems. San Mateo, CA: Morgan Kauffman.