Aula 06 – Naïve Bayes

1001524 – Aprendizado de Máquina I
2023/1 - Turmas A, B e C
Prof. Dr. Murilo Naldi

naldi@dc.ufscar.br

Agradecimentos

- Parte do material utilizado nesta aula foi cedido pelos professores André C.P.L.F de Carvalho e Ricardo J.G.B. Campello e, por esse motivo, o crédito deste material é deles
- Parte do material utilizado nesta aula foi disponibilizado por M. Kumar no endereço:
  - www-users.cs.umn.edu/~kumar/dmbook/index.php
- Agradecimentos a Intel Software e a

  Intel IA Academy pelo material

  disponibilizado e recursos didáticos

Aulas Anteriores

- Visualização e estatísticas de resumo
- Pré-processamento
- Classificação e Validação de resultados
- Árvores de Decisão

Conteúdo

- Resumo
  - Revisão probabilidade
  - Regra de Bayes
  - Generalização
  - Bayes Ingênuo
  - Aplicações em classificação

Incerteza e Probabilidade

- Variável aleatória
  - Representa objeto ou evento (fatos, ações, crenças,...) que depende de aleatoriedade
  - Elas podem possuir domínios:
    - Booleano: como Feliz com domínio <verdadeiro, falso>
    - Discreto: como Clima com domínio <ensolarado, nublado, chuvoso>
    - Contínuo: como Peso com domínio nos números reais positivos

Variável aleatória

- Geralmente, escrevemos o nome da variável aleatória com a primeira letra maiúscula
- Adicionalmente, podemos representar eventos utilizando constantes, geralmente em letras minúsculas
- Exemplo:

$$\mathit{CéuAzul} = \text{verdadeiro} \Rightarrow \mathit{céuAzul}$$

$$\mathit{CéuAzul} = \text{falso} \Rightarrow \neg\mathit{céuAzul}$$

Axiomas de probabilidade

- Todas as probabilidades estão entre 0 e 1

$$0 \leq P(a) \leq 1$$

- Proposições verdadeiras tem probabilidade 1 e as falsas possuem 0

$$P(verdadeiro) = 1 \text{ e } P(falso) = 0$$

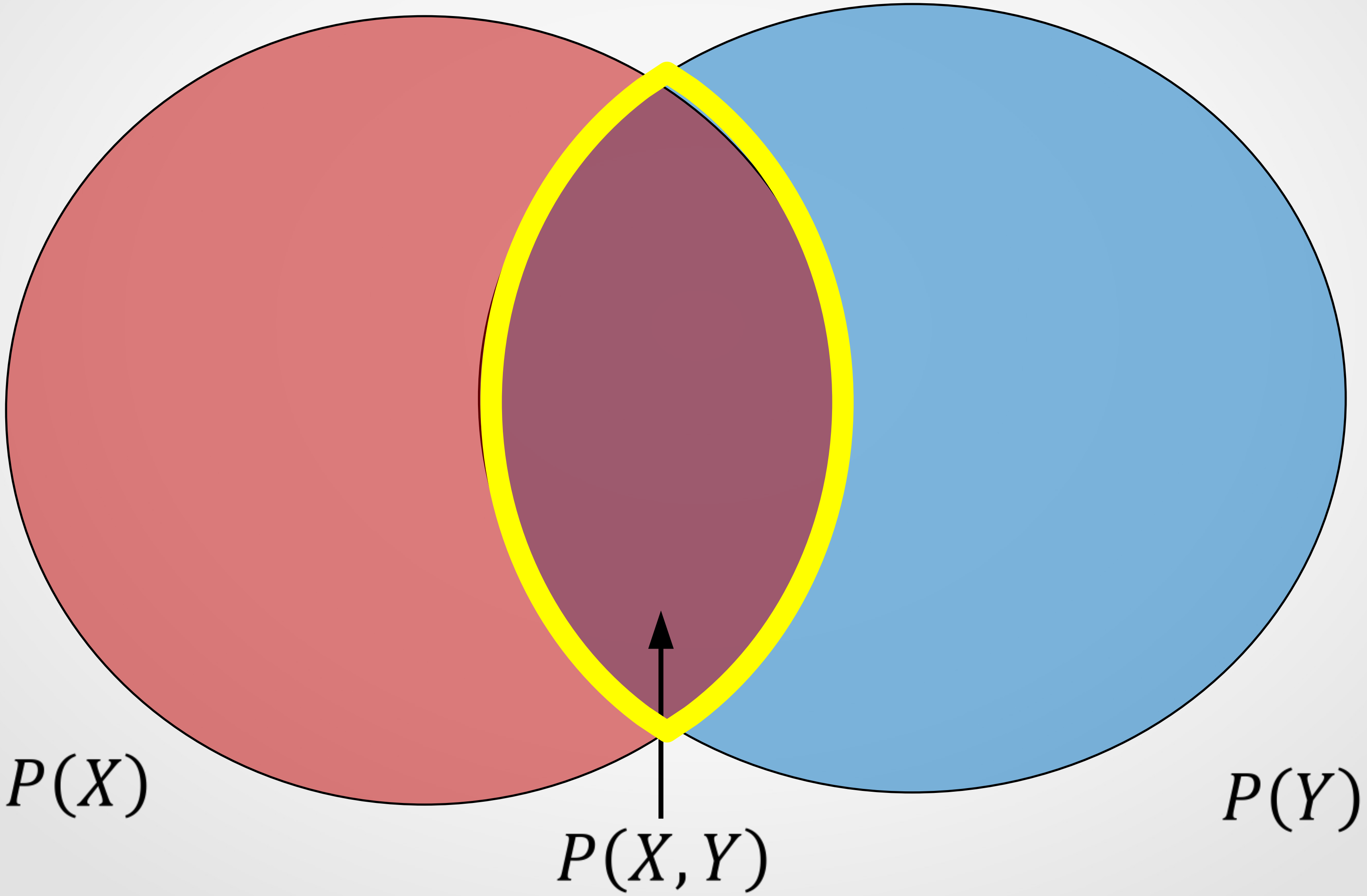
- A probabilidade de uma disjunção é:

$$P(a \vee b) = P(a) + P(b) - P(a \wedge b)$$

Probabilidade a *priori*

- Ou probabilidade *incondicional*
- Consiste na probabilidade de uma proposição ocorrer na ausência de quaisquer outras informações
- Pode ser *simples* ou *conjunta*
- Exemplos:
  - $P(\mathit{Clima} = \mathit{ensolarado}) = 0{,}5$
  - $P(\textit{cárie} \wedge \mathit{dor}) = 0{,}1$

Probabilidade a *priori* conjunta



Distribuição de probabilidade

- São as probabilidades *a priori* de uma determinada variável aleatória
- Exemplo:
  - P(*Clima*) = <0,5, 0,2, 0,3>

significa que *Clima* possui probabilidade *a priori* 0,5 de ser *ensolarado*, 0,2 de estar *nublado* e 0,3 de estar *chuvoso*.

Distribuição de probabilidade

- A distribuição de probabilidade conjunta total é quando temos a probabilidade *a priori de cada combinação do conjunto de variáveis aleatórias*

Inferência

- Podemos utilizar a distribuição de probabilidade conjunta total como um modelo do ambiente

	atrasado		***¬atrasado***	
	gasolina	***¬gasolina***	***gasolina***	***¬gasolina***
engarrafamento	0,19	0,14	0,13	0,01
¬engarrafamento	0,06	0,07	0,25	0,15

Marginalização

- Ou totalização consiste em calcular a probabilidade de um acontecimento a partir da base de conhecimento

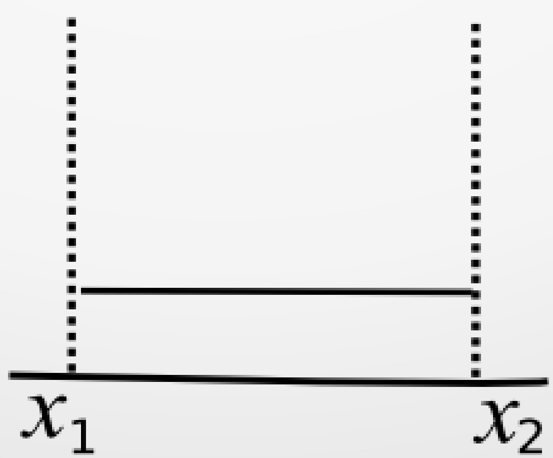
$$P(Y) = \sum_{z \in Z} P(Y, z)$$

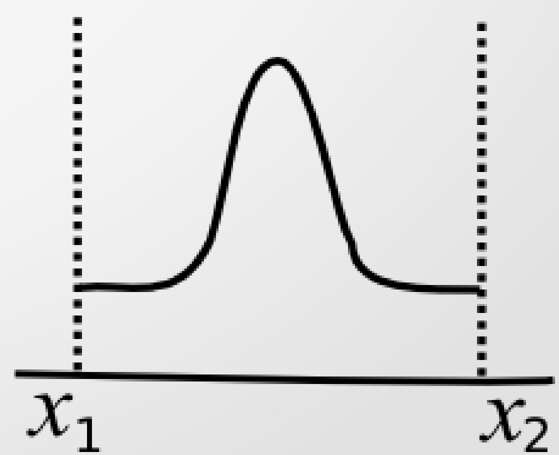
- Exemplo: a probabilidade de ocorrer um *engarrafamento* é de 0,47 e não ocorrer é 0,53

engarrafamento	**0,19**	**0,14**	**0,13**	**0,01**

Funções de densidade de probabilidade

- Variáveis contínuas não podem ser representadas em uma tabela pois seus valores são infinitos
- Para isso, utilizamos funções de densidade de probabilidade
- Exemplos:
  - uniforme
  - normal





Exemplo

- Podemos definir que uma variável aleatória X que define a temperatura em uma manhã de outubro em São Carlos está *distribuída uniformemente* entre 18 e 26 ^{O}C

$$P(X=x) = U[18,26](x)$$

- Qual a probabilidade de fazer 20,5^{O}C? Não é possível calcular a probabilidade em um ponto (zero). Então, usamos uma pequena região em torno dele.

$$\lim_{dx \to 0} P(20{,}5 \leq X \leq 20{,}5 + dx) / dx = 0{,}125$$

Probabilidade condicional

- Conhecimento a respeito de variáveis aleatórias podem influenciar na probabilidade de outras
- Por exemplo, saber que um paciente possui dor de dente aumenta a chance deste paciente ter cárie

$$P(cárie|dordedente) = 0,9$$

- Chamamos isso de probabilidade condicional

Probabilidade condicional

- Probabilidades condicionais podem ser calculadas a partir de probabilidades incondicionais

$$P(a|b) = \frac{P(a \land b)}{P(b)}$$

Condicionamento

- Outra forma de obter a probabilidade *a priori* de uma variável é a partir de variáveis condicionais.
- Esse processo é conhecido como condicionamento.

$$P(Y) = \sum_{z} P(Y|z) P(z)$$

Independência

- Ou independência absoluta
- Uma variável é independente de outra quando sua probabilidade condicional em relação à última é igual a sua probabilidade *a priori*, ou seja:

$$P(A|B) = P(A)$$

- Exemplo:
  - Imagine que adicionamos uma quarta variável a nossa base de conhecimento, a variável booleana *Apaixonado*

apaixonado	***atrasado***		***¬atrasado***	
	gasolina	***¬gasolina***	***gasolina***	***¬gasolina***
engarrafamento	0,145	0,07	0,06	0,005
¬engarrafamento	0,03	0,035	0,125	0,075

¬apaixonado	***atrasado***		***¬atrasado***	
	gasolina	***¬gasolina***	***gasolina***	***¬gasolina***
engarrafamento	0,145	0,07	0,06	0,005
¬engarrafamento	0,03	0,035	0,125	0,075

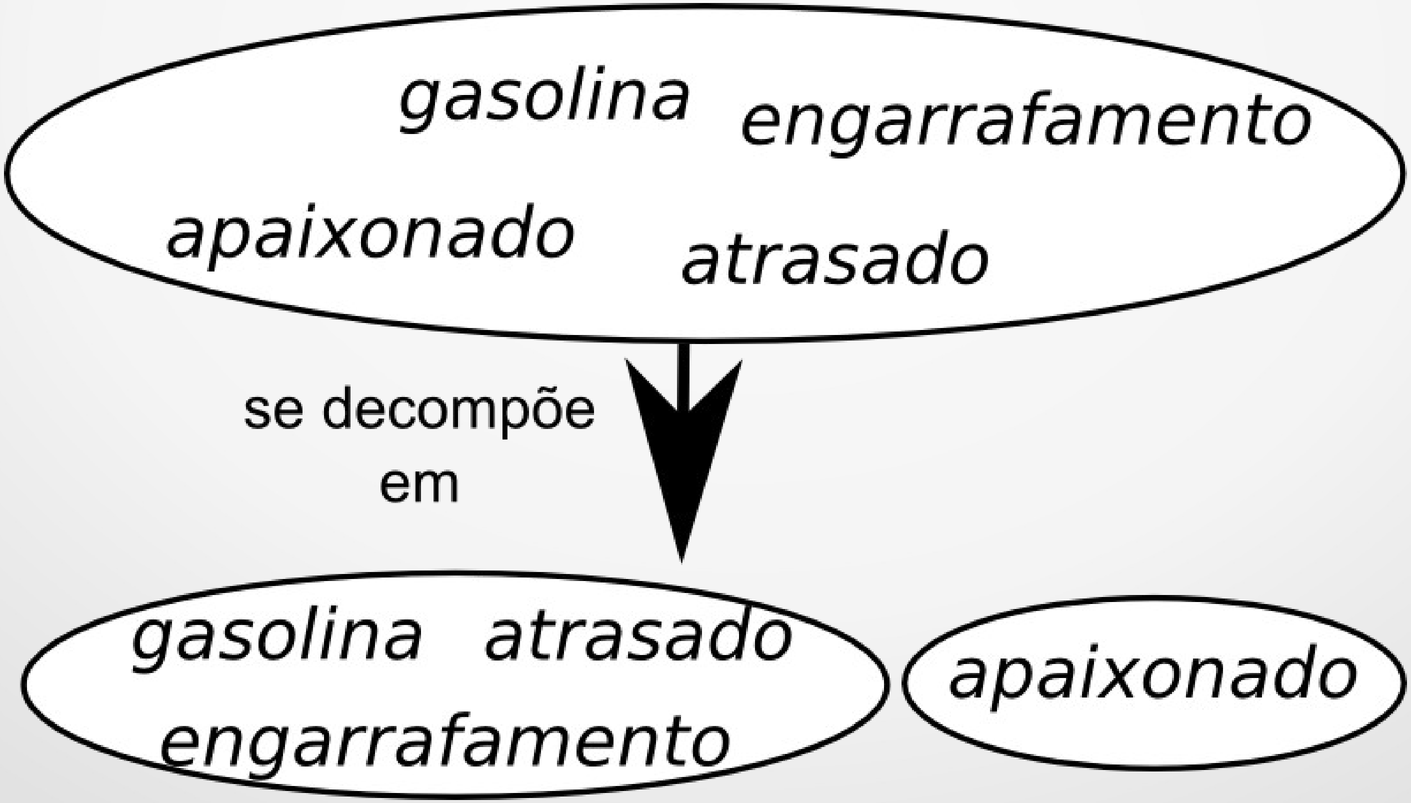
Independência

- Por exemplo, verificando a independência de Engarrafamento e Apaixonado temos:

$$P(engarrafamento|apaixonado) = P(engarrafamento \wedge apaixonado) / P(apaixonado)$$

Fatoração

- Consiste em reduzir a quantidade de informações necessárias para especificar a distribuição do conjunto total



Fatoração

- A fatoração é uma característica muito importante.
  - O número de valores de probabilidades condicionais é combinatorial em relação ao número de variáveis dependentes
  - Independência e fatoração permitem ganho computacional, pois tratam variáveis independentemente.

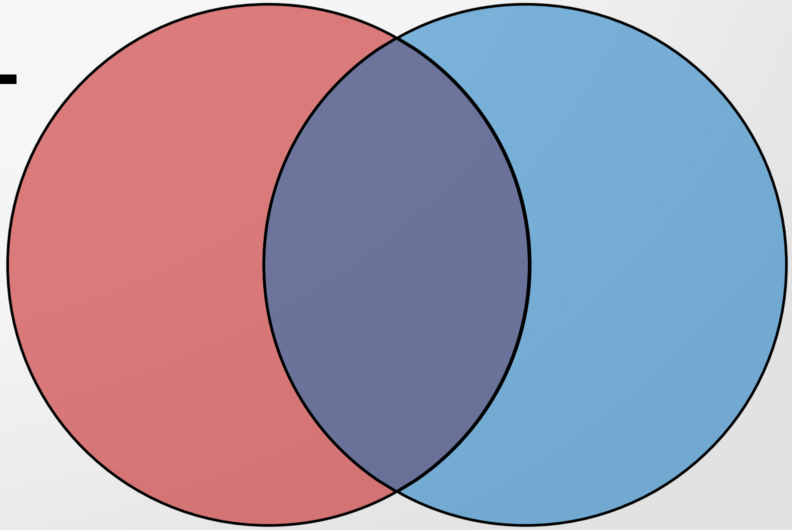
Relações entre probabilidades

- Considerando as relações entre probabilidades

$$P(X,Y) = P(Y|X) * P(X) = P(X|Y) * P(Y)$$

- Se invertemos as probabilidades condicionais, temos que:

$$P(Y|X) = \frac{P(X|Y) * P(Y)}{P(X)}$$



Regra de Bayes

- Para variáveis multivaloradas temos que a regra de Bayes pode ser escrita como:

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)}$$

- A regra de Bayes é a base de todos os sistemas modernos de inferência probabilística.

Regra de Bayes

- Ou em sua forma condicionada com uma evidência prática *e*

$$P(Y|X,e)=\frac{P(X|Y,e)P(Y|e)}{P(X|e)}$$

Aplicação

- Muito utilizado na prática quando se tem um efeito conhecido e é desejado descobrir a causa.

$$P(causa \mid efeito) = \frac{P(efeito \mid causa)\,P(causa)}{P(efeito)}$$

Exemplo de uso

- Um agente de diagnóstico sabe que meningite causa rigidez no pescoço 50% das vezes.
- Ele também sabe que a probabilidade *a priori* de um paciente ter meningite é de 1/50.000 e de qualquer paciente ter rigidez no pescoço é 1/20.
- Qual a probabilidade que um paciente tenha rigidez no pescoço causada por meningite?

Exemplo de uso

- Sendo s a proposição que o paciente tem rigidez no pescoço e m a proposição de que o paciente possui meningite, temos:
  - $P(s|m) = 0,5$
  - $P(m)=1/50.000$
  - $P(s)=1/20$
  - $P(m|s)=P(s|m)P(m)/P(s)=$

    $0,5 \times 1/50.000 \times 20 = 0,0002$

Combinação de evidências

- Vimos que a regra de Bayes é útil para responder consultas condicionadas sobre uma única peça de evidência.
- Contudo, e se existirem mais de uma variável de evidência?
- É preciso conhecer a dependência entre cada dupla de variáveis!

Combinação de evidências

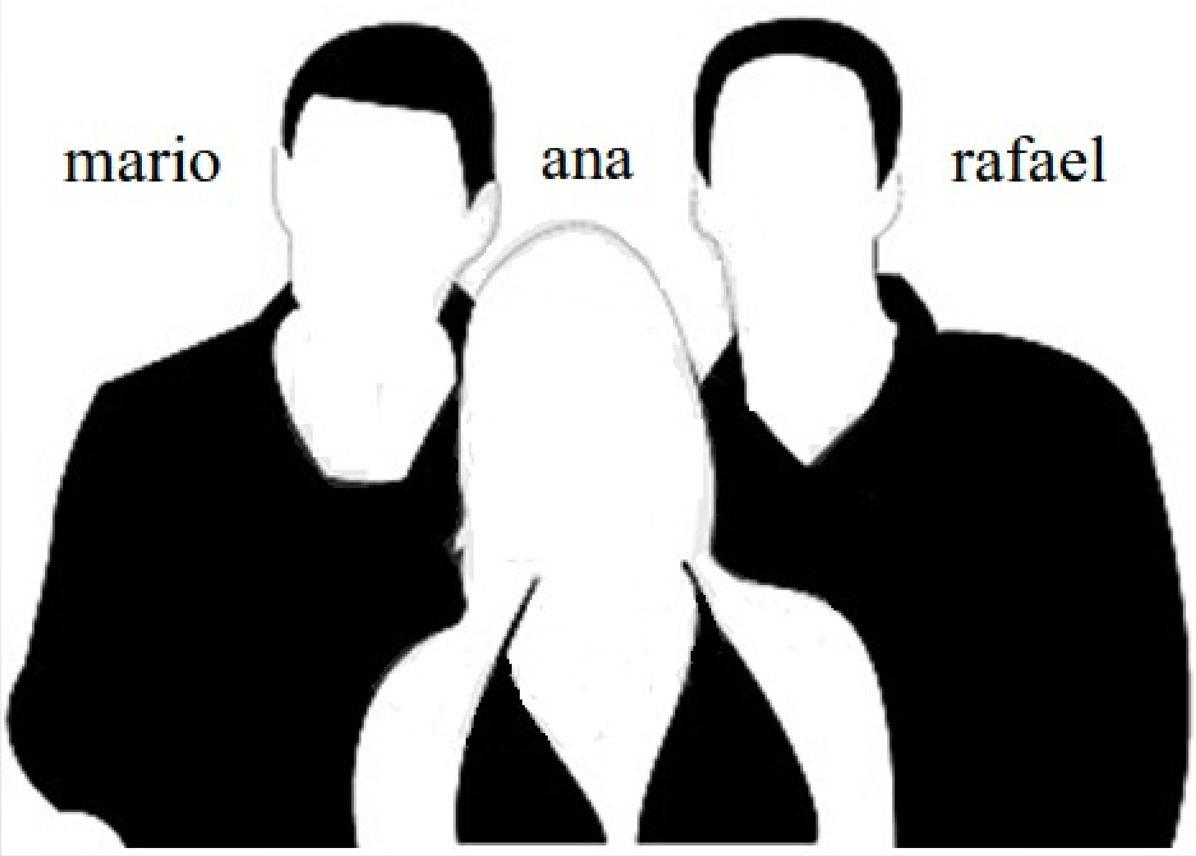
- Se houver *n* variáveis de evidência binárias, o número de combinações possíveis de variáveis aleatória será 2^n.
- Portanto, é preciso encontrar asserções sobre o domínio que permitam simplificar as expressões.
- Usar um método aproximado!

Combinação de evidências

- Utilizar independência pode ser a solução.
  - Porém, pode ser difícil encontrar!
- Duas variáveis podem ser independentes quando condicionadas aos valores de uma terceira.
  - Esse é o conceito de independência condicional!
  - Menos raro que absoluta.

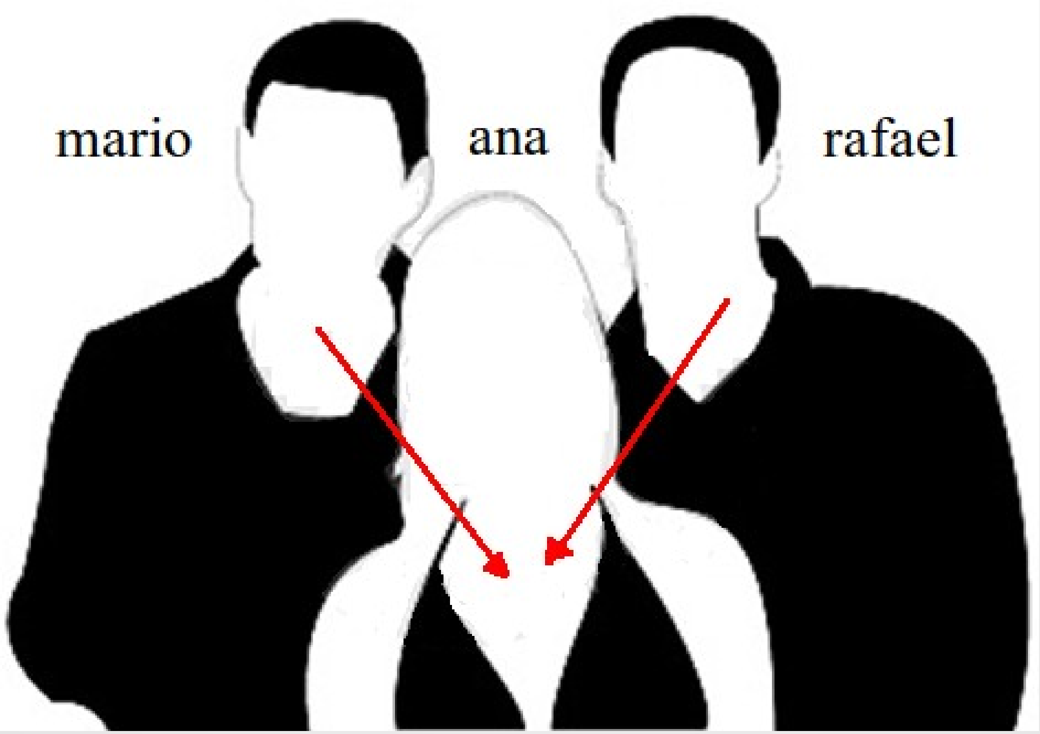
Independência condicional

- Considere a probabilidade de três pessoas irem no TUSCA.



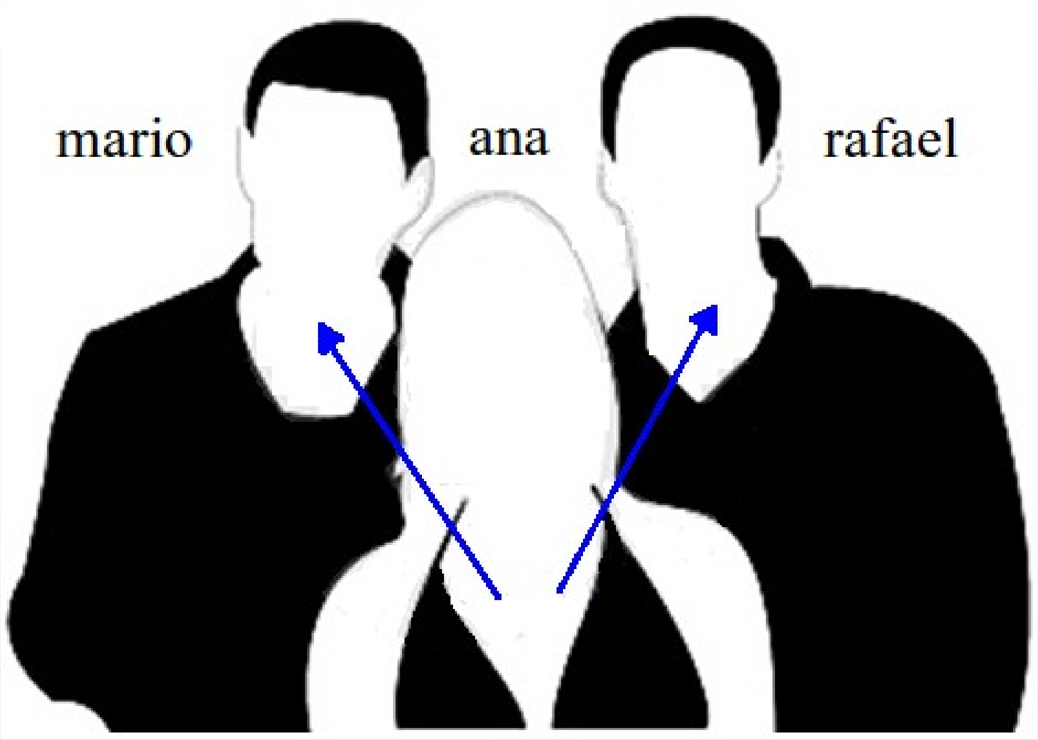
Independência condicional

- Ana é amiga de Mario e Rafael, que não se conhecem.
- Eles influenciam na decisão de Ana



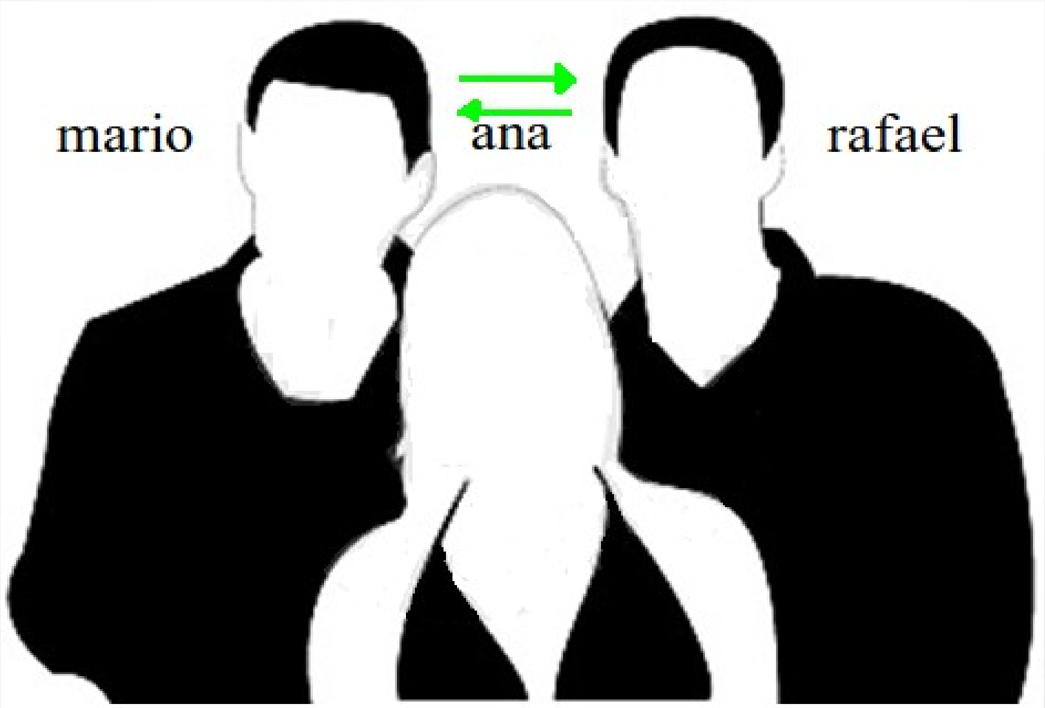
Independência condicional

- Em contrapartida, a probabilidade de Ana ir (ou deixar de ir) influencia na decisão dos dois.



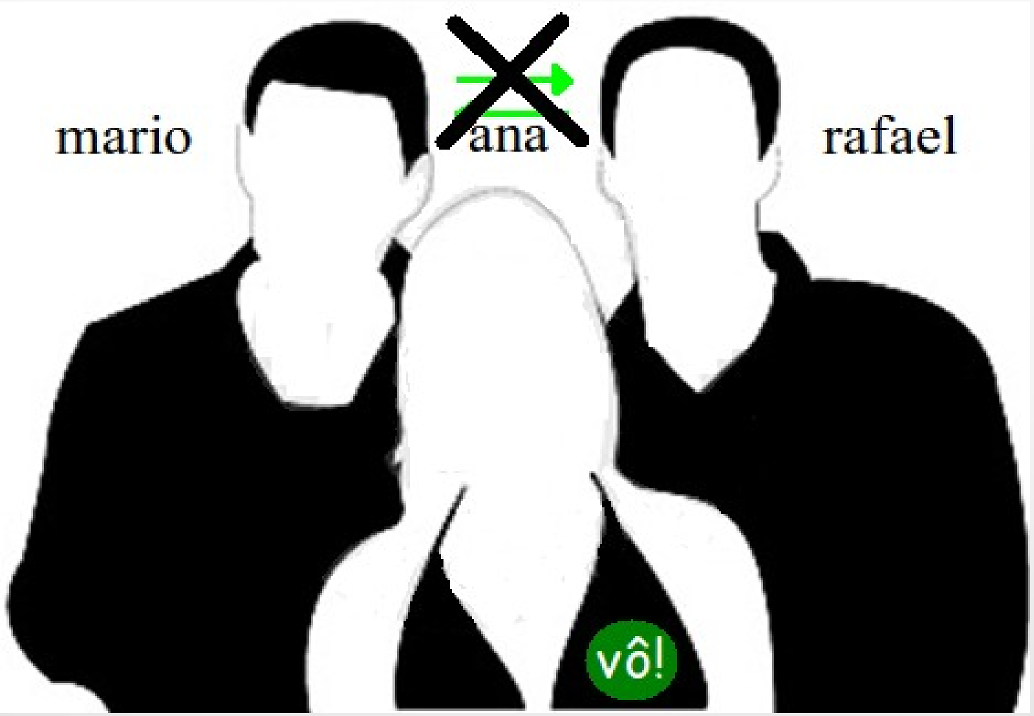
Independência condicional

- Enquanto houver incerteza a respeito de Ana, Mario pode influenciar (ou não) a decisão de Rafael e vice-versa



Independência condicional

- Se Ana decide ir (ou não), as probabilidades de *Mario* e *Rafael* são independentes
- Isso é independência condicional!



Independência condicional

- Duas variáveis são independentes condicionalmente se, dada uma terceira variável Z, temos:

$$P(X,Y|Z) = P(X|Z)P(Y|Z)$$

- Sendo assim, no exemplo dado temos:

$$P(\mathit{Mario},\mathit{Rafael}|\mathit{Ana}) = P(\mathit{Mario}|\mathit{Ana})P(\mathit{Rafael}|\mathit{Ana})$$

Fatoração

- Asserções de independência condicional também permitem a decomposição da distribuição conjunta total em itens menores

- Exemplo:

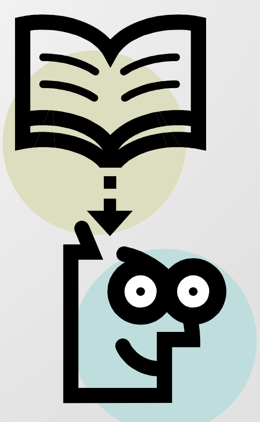
$$P(\textit{Mario, Rafael,Ana}) =$$

$$P(\textit{Mario, Rafael}|\textit{Ana})P(\textit{Ana}) =$$

$$P(\textit{Mario}|\textit{Ana})P(\textit{Rafael}|\textit{Ana})P(\textit{Ana})$$

Redução de complexidade

- A decomposição em problemas menores permite a redução significativa da complexidade de sistemas probabilísticos.
- Se existirem *n* variáveis independentes condicionalmente dado um evento, a complexidade do tamanho do sistema é reduzido de $O(2^n)$ para $O(n)$!



Causa X Múltiplos Efeitos

- Podemos modelar o problema em que uma variável aleatória influencia de maneira direta um conjunto de variáveis aleatórias condicionalmente independentes.
- Neste modelo, chamamos a variável que influencia de causa e as influenciadas de efeitos.

Regra da cadeia ingênua

- Assumindo independência condicional entre os efeitos, $P(Efeito_i|Efeito_{i+1}, \ldots, Efeito_d, Causa) = P(Efeito_i|Causa)$, a distribuição conjunta pode ser escrita como:

$$P(Causa, Efeito_1, \ldots, Efeito_d) = \mathrm{P}(Causa)\prod_{i=1}^{d} P(Efeito_i|Causa)$$

Regra da cadeia ingênua

Ingênuo!!!

- Assumindo independência condicional entre os efeitos, $P(Efeito_i | Efeito_{i+1}, \ldots, Efeito_d, Causa) = P(Efeito_i | Causa)$, a distribuição conjunta pode ser escrita como:

$$P(Causa, Efeito_1, \ldots, Efeito_d) = P(Causa) \prod_{i=1}^{d} P(Efeito_i | Causa)$$

Classificação usando *Bayes*

- Para classificação usando Regra de Bayes, utilizamos um modelo probabilístico condicional.
  - Consideramos um objeto $\mathbf{x}=(x_1,\ldots,x_d)$ a ser classificado, composto por *d* características (*variáveis independentes*)
  - É possível calcular a probabilidade deste objeto pertencer a classe C_k ela como:

$$P(C_k|\mathbf{x}) = P(C_k|x_1,\ldots,x_d)$$

Voltando a Regra de Bayes

$$P\left(C_k \middle| \mathbf{x}\right) = \frac{P\left(C_k\right) P\left(\mathbf{x} \middle| C_k\right)}{P\left(\mathbf{x}\right)}$$

$$Posteriori = \frac{Verossimilhança \times Priori}{Evidência}$$

Regra de Bayes para Classificação

$$P(C_k|\mathbf{x}) = \frac{P(C_k)P(\mathbf{x}|C_k)}{\cancel{P(\mathbf{x})}}$$

Denominador é constante e não depende das classes C_k!

$$Posteriori = \frac{Verossimilhança \times Priori}{\cancel{Evidência}}$$

Naïve Bayes para Classificação

Denotando proporcionalidade por α

$$P\left(C_k \middle| \mathbf{x}\right) \alpha\, P\left(C_k\right) P\left(\mathbf{x} \middle| C_k\right)$$

Numerador é a probabilidade conjunta!

$$P\left(C_k \middle| x_1, \ldots, x_d\right) \alpha\, P\left(C_k, x_1, \ldots, x_d\right)$$

Assumindo independência condicional entre os atributos do objeto, temos:

$$= P\left(C_k\right) \prod_{i=1}^{d} P\left(x_i \middle| C_k\right)$$

Naïve Bayes para Classificação

- Portanto, o modelo probabilístico para a classe C_k é dado por:

$$P(C_k | x_1, \ldots, x_d) =$$

$$= \frac{1}{Z} P(C_k) \prod_{i=1}^{d} P(x_i | C_k)$$

onde Z é P(**x**), constante e dependente de **x**.

Construindo Classificador

- O classificador Bayesiano utiliza o modelo probabilístico combinado com alguma regra de decisão que defina qual classe é preferida dentre as *K* hipóteses testadas
  - Assim, a classe de um objeto **x** é dada por:

$$\hat{y} = \underset{k \in \{1, \ldots, K\}}{\operatorname{argmax}} P(C_k) \prod_{i=1}^{d} P(x_i | C_k)$$

O truque do log

- Ao multiplicar muitos valores (*d*) abaixo de 0 faz com que haja instabilidade computacional (*underflow*).
  - Para evitar esse problema, pode-se aplicar o truque do log

$$\log\left(P\left(C_k\right)\right)\prod_{i=1}^{d}\log\left(P\left(x_i|C_k\right)\right)$$

Exemplo: predizendo tênis com NB

Dia	Clima	Temperatura	Umidade	Vento	Jogar Tênis
1	ensolarado	quente	alta	fraco	não
2	ensolarado	quente	alta	forte	não
3	nublado	quente	alta	fraco	sim
4	chuva	suave	alta	fraco	sim
5	chuva	fria	normal	fraca	sim
6	chuva	fria	normal	forte	não
7	nublado	fria	normal	forte	sim
8	ensolarado	suave	alta	fraco	não
9	ensolarado	fria	normal	fraco	sim
10	chuva	suave	normal	fraco	sim
11	ensolarado	suave	normal	forte	sim
12	nublado	suave	alta	forte	sim
13	nublado	quente	normal	fraco	sim
14	chuva	suave	alta	forte	não

Treinando modelo NB para tênis

- P(Jogar = sim) = 9/14   P(Jogar = não) = 5/14

Clima	**Jogar = sim**	**Jogar = não**
ensolarado	2/9	3/5
nublado	4/9	0/5
chuva	3/9	2/5

Temperatura	**Jogar = sim**	**Jogar = não**
quente	2/9	2/5
suave	4/9	2/5
fria	3/9	1/5

Umidade	**Jogar = sim**	**Jogar = não**
alta	3/9	4/5
normal	6/9	1/5

Vento	**Jogar = sim**	**Jogar = não**
forte	3/9	3/5
fraco	6/9	2/5

Usando modelo

- Considerando as probabilidades dos atributos temos

Feature	Jogar=sim	Jogar = não
Clima = ensolarado	2/9	3/5
Temperatura = fria	3/9	1/5
Umidade = alta	3/9	4/5
Vento = forte	3/9	3/5
Priori	**9/14**	**5/14**
Posteriori	**0.0053**	**0.0206**

- P(Jogar=sim|ensolarado,fria,alta,forte) = P(ensolarado|sim) * P(fria|sim) * P(alta|sim) * P(forte|sim) * P(sim)
- P(Jogar=não|ensolarado,fria,alta,forte) = P(ensolarado|não) * P(fria|não) * P(alta|não) * P(forte|não) * P(não)

Problema

- Categorias que não possuem entradas para todos os atributos resultam em 0 para probabilidades condicionais
  - $P(C|\mathbf{x}) = P(x_1|C) * P(x_2|C) * P(C)$

Problema

- Categorias que não possuem entradas para todos os atributos resultam em 0 para probabilidades condicionais
  - $P(C|\mathbf{x}) = P(x_1|C) * P(x_2|C) * P(C) = 0$

0

Suavização de Laplace

- Categorias que não possuem entradas para todos os atributos resultam em 0 para probabilidades condicionais
  - $P(C|\mathbf{x}) = P(x_1|C) * P(x_2|C) * P(C) = 0$

0

- Solução
  - Somar 1 no numerador e denominador das categorias vazias, onde *d* é o número de atributos no conjunto de treino

$$P(C|x) = \frac{contagem(x, C) + 1}{contagem(C) + d + 1}$$

Tipos diferentes de Naïve *Bayes*

- Diferentes tipos de dados requerem diferentes tipos de distribuição de probabilidade para gerar o modelo

Modelo de NB	Tipo de dado
Bernoulli	Binário (V/F)
Multinomial	Discreto (e.g. contagens)
Gaussiano	Contínuo

Combinando tipos diferentes de atributos

- Problema
  - Os dados contêm diferente tipos de atributos
    - Ou seja, contínuos e categóricos/discretos

Combinando tipos diferentes de atributos

- Problema
  - Os dados contêm diferente tipos de atributos
    - Ou seja, contínuos e categóricos/discretos
- Possíveis soluções
  - Opção 1: dividir o atributo contínuo em intervalos e ajustar a um modelo multinomial
  - Opção 2: ajustar um modelo gaussiano aos atributos contínuos e categóricos e combinar

Aplicando NB com *holdout* Iris

- Código

```
import pandas as pd
#Importa o método de Naïve Bayes com distribuição de Bernouilli do Sklearn
from sklearn.Naïve_bayes import BernoulliNB
# Localização do arquivo
filepath = 'data/Iris_Data.csv'
# Importando os dados
data = pd.read_csv(filepath)
#Colocando os dados em ordem aleatória
randomdata = (data.sample(n=150, replace=False))
#Aplicando hold out
traindata = randomdata.iloc[:135,:]
testdata = randomdata.iloc[135:,:]
#Cria uma instância de classe
BNB = BernoulliNB(alpha=1.0)
#Ajusta o modelo NB aos dados de treino
BNB = BNB.fit(traindata.iloc[:,0:4], traindata.iloc[:,4])
#Classe real
print(testdata.iloc[:,4])
#Classe predita
print(BNB.predict(testdata.iloc[:,0:4]))
```

Aplicando NB com *holdout* Iris

- Código

```
import pandas as pd
#Importa o método de Naïve Bayes com distribuição de Bernouilli do Sklearn
from sklearn.Naïve_bayes import BernoulliNB
# Localização do arquivo
filepath = 'data/Iris_Data.csv'
# Importando os dados
data = pd.read_csv(filepath)
#Colocando os dados em ordem aleatória
randomdata = (data.sample(n=150, replace=False))
#Aplicando hold out
traindata = randomdata.iloc[:135,:]
testdata = randomdata.iloc[135:,:]
#Cria uma instância de classe
BNB = BernoulliNB(alpha=1.0)
#Ajusta o modelo NB aos dados de treino
BNB = BNB.fit(traindata.iloc[:,0:4], traindata.iloc[:,4])
#Classe real
print(testdata.iloc[:,4])
#Classe predita
print(BNB.predict(testdata.iloc[:,0:4]))
```

Parâmetro de suavização de Laplace

Aplicando NB com *holdout* Iris

- Saída = Classes

```
19             Iris-setosa
72         Iris-versicolor
16             Iris-setosa
66         Iris-versicolor
39             Iris-setosa
99         Iris-versicolor
105         Iris-virginica
13             Iris-setosa
64         Iris-versicolor
34             Iris-setosa
96         Iris-versicolor
138         Iris-virginica
20             Iris-setosa
79         Iris-versicolor
22             Iris-setosa
```

- Saída Predita

```
['Iris-virginica'
'Iris-virginica'
'Iris-virginica'
'Iris-virginica'
 'Iris-virginica'
'Iris-virginica'
'Iris-virginica'
'Iris-virginica'
 'Iris-virginica'
'Iris-virginica'
'Iris-virginica'
'Iris-virginica'
 'Iris-virginica'
'Iris-virginica'
'Iris-virginica']
```

Aplicando NB com *holdout* Iris

- Código

```
import pandas as pd
#Importa o método de Naïve Bayes com distribuição de Bernouilli do Sklearn
from sklearn.Naïve_bayes import BernoulliNB
# Localização do arquivo
filepath = 'data/Iris_Data.csv'
# Importando os dados
data = pd.read_csv(filepath)
#Colocando os dados em ordem aleatória
randomdata = (data.sample(n=150, replace=False))
#Aplicando hold out
traindata = randomdata.iloc[:135,:]
testdata = randomdata.iloc[135:,:]
#Cria uma instância de classe
BNB = BernoulliNB(alpha=1.0)
#Ajusta o modelo NB aos dados de treino
BNB = BNB.fit(traindata.iloc[:,0:4], traindata.iloc[:,4])
#Classe real
print(testdata.iloc[:,4])
#Classe predita
print(BNB.predict(testdata.iloc[:,0:4]))
```

Modelo com distribuição de Bernoulli é para atributos binários!

Aplicando NB com *holdout* Iris

- Código

#Importa o método de Naïve Bayes com distribuição Gaussiana do Sklearn

from sklearn.Naïve_bayes import GaussianNB

#Cria uma instância de classe

BNB = GaussianNB()

#Ajusta o modelo NB aos dados de treino

BNB = BNB.fit(traindata.iloc[:,0:4], traindata.iloc[:,4])

#Classe real

print(testdata.iloc[:,4])

#Classe predita

print(BNB.predict(testdata.iloc[:,0:4]))

Aplicando NB com *holdout* Iris

- Saída = Classes

```
19          Iris-setosa
72      Iris-versicolor
16          Iris-setosa
66      Iris-versicolor
39          Iris-setosa
99      Iris-versicolor
105      Iris-virginica
13          Iris-setosa
64      Iris-versicolor
34          Iris-setosa
96      Iris-versicolor
138      Iris-virginica
20          Iris-setosa
79      Iris-versicolor
22          Iris-setosa
```

- Saída Predita

```
['Iris-setosa'
'Iris-versicolor'
'Iris-setosa'
'Iris-versicolor'
 'Iris-setosa'
'Iris-versicolor'
'Iris-virginica'
'Iris-setosa'
 'Iris-versicolor'
'Iris-setosa'
'Iris-versicolor'
'Iris-virginica'
 'Iris-setosa'
'Iris-versicolor'
'Iris-setosa']
```

Computação distribuída com NB

- Existem bibliotecas para computação de modelos de Naïve Bayes com grandes conjuntos de dados em paralelo
  - Usa parâmetros limitados e probabilidades de log como método de sumarização
- Scikit-Learn contém implementações de “ajuste parcial” para métodos que podem ser aplicados em paralelo em cálculos fora de núcleo