Imagine que eu tenha construído uma tabela com os resultados das partidas de futebol de meu time, informando se eu fiquei feliz ou não com os resultados:

RESULTADO DO JOGO	FIQUEI FELIZ?
vitória	sim
empate	sim
vitória	sim
derrota	não
derrota	não
empate	sim
vitória	sim
vitória	não
vitória	sim
derrota	sim
empate	não
derrota	não
empate	não
vitória	sim
vitória	sim
derrota	não
empate	sim

A partir desses dados, podemos contabilizar quantas vezes cada resultado aparece, construindo uma tabela de frequências:

Tabela de Frequências		
Resultado do jogo	*Feliz = sim*	*Feliz = não*
vitória	6	1
empate	3	2
derrota	1	4
Total	**10**	**7**

P(sim) = 10/17 = 0,5882353

P(não) = 7/17 = 0,4117647

P(vitória) = 7/17 = 0,4117647

P(empate) = 5/17 = 0,2941176

P(derrota) = 5/17 = 0,2941176

P(A/B) = P(B/A)*P(A) / P(B)

P(sim/vitória) = P(vitória/sim)*P(sim)/ P(vitória)

P(vitória/sim) = 6/10 = 0,6

P(sim/vitória) = 0,6*0,5882353/ 0,4117647= **0,8571**

Quando existe mais de uma feature:

$$P(C_k|X) = \prod_{i=1}^{n} P(x_i|C_k)$$

$$P(C_i|x_1, x_2, \ldots, x_n) = \frac{P(x_1, x_2, \ldots, x_n|C_i).P(C_i)}{P(x_1, x_2, \ldots, x_n)} \text{ for } 1 \leq i \leq k$$

Explicando melhor:

P(A/B) = P(B/A)*P(A) / P(B)

P(A/B) = P(B/A)*P(A) / P(B)

P(A/(x1, x2, x3)) = P(x1/A)*P(x2/A)*P(x3/A)*P(A) / P(x1)*P(x2)*P(x3)

Funções em Programação:

- Quando os dados das variáveis preditoras são discretos, devemos utilizar a função Multinomial – Distribuição de Poisson.
- Quando os dados das variáveis preditoras são discretos e binários, devemos utilizar a função Bernoulli - Distribuição de Bernoulli.
- Quando os dados das variáveis preditoras são contínuos com distribuição normal, devemos utilizar a função Gaussiana – Distribuição Gaussiana.
- Quando os dados das variáveis preditoras são contínuos e não seguem uma distribuição normal, devemos utilizar a Estimativa de densidade de kernel.

Na linguagem R uma das funções que utiliza o algoritmo Naive Bayes é a “naive_bayes”, do pacote “naivebayes”. Com esta função, através do ajuste de seus parâmetros é possível definir qual destes métodos será utilizado. Segue tabela com detalhamento dos parâmetros:

usekernel	*poisson*	Método – Variáveis numéricas
FALSE	FALSE	Distribuição Gaussiana
TRUE	FALSE	Estimativa de densidade kernel
FALSE	TRUE	Distribuição Gaussiana para variáveis com decimais e Distribuição de Poisson para variáveis com inteiros
TRUE	TRUE	Estimativa de densidade kernel para variáveis decimais e Distribuição de Poisson para variáveis com inteiros

Variáveis do tipo caractere, fator e lógico utilizam a Distribuição de Bernoulli.

As funções Multinomial (Poisson) e Bernoulli realizam os cálculos das probabilidades conforme vimos no exemplo anterior. Já a função Gaussian faz um cálculo um pouco diferente, pois ela pega cada variável preditora, calcula sua média e seu desvio padrão, e a partir disso calcula as probabilidades considerando uma distribuição gaussiana.

Exemplo:

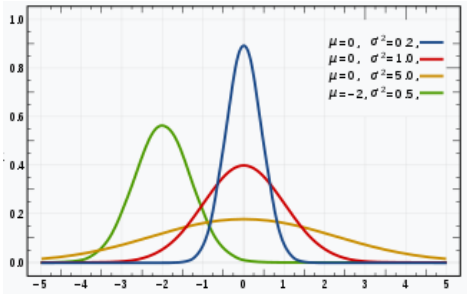
Altura	Sexo
1.46	F
1.59	F
1.85	M
1.73	M
1.66	M
1.60	F
1.74	F
1.93	M
1.80	M
1.71	F
1.68	M

P(A/B) = P(B/A)*P(A) / P(B)

P(M/1.82) = P(1.82/ M)*P(M) / P(1.82)

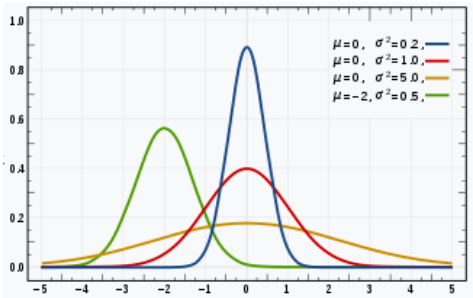
P(1.82/M) =

$$f(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$



P(1.82) =

$$f(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$



O problema de probabilidade zero na tabela de frequências

Agora vamos falar sobre o problema de probabilidade zero nas funções Multinomial (Poisson) e Bernoulli.

Imagine a seguinte classificação de mensagens de e-mail:

Frases	Spam
clique no link abaixo	sim
confira essa foto incrível	sim
vamos na casa do Marcelo	não
cadastre uma nova senha	não
poker online grátis	sim
...	...

Tabela de Frequências		
Palavra	*Spam = sim*	*Spam = não*
clique	68	3
no	42	115
link	13	5
abaixo	7	1
confira	9	0
vamos	2	76
...	...	...

P(sim/“confira essa foto louca”) = P(sim/(confira, essa, foto, louca))

P(A/B) = P(B/A)*P(A) / P(B)

P(A/(x1, x2, x3, x4)) = P(x1/A)*P(x2/A)*P(x3/A)* P(x4/A)*P(A) / P(x1)*P(x2)*P(x3) *P(x4)

Repare que quando uma palavra não existe na tabela de frequências, isso resultaria em probabilidade zero.

O método utilizado para contornar essa situação é a **suavização de Laplace**. Essa suavização é dada pela fórmula:

$$\hat{\theta}_i = \frac{x_i + \alpha}{N + \alpha d} \qquad (i = 1, \ldots, d),$$

Onde Theta-i é o novo parâmetro de cálculo de probabilidade (suavizado), xi são as observações desse parâmetro, alfa é o suavizador (default = 1), N é o total de ocorrências (tabela de frequência) dos parâmetros e d é o total de parâmetros.

Utilizando como base a tabela de frequências do primeiro exercício:

Tabela de Frequências		
Resultado do jogo	*Feliz = sim*	*Feliz = não*
vitória	6	1
empate	3	2
derrota	1	4
Total	**10**	**7**

Cada probabilidade precisaria ser ajustada. Por exemplo, a probabilidade de vitória, em vez de ser 7/17 = 0,41, ficaria (supondo alfa = 1):

P(vitória) = θ1 = (7+1)/(17+1*3) = 0,40.

P(empate) = θ2 = (5+1)/(17+1*3) = 0,30.

P(derrota) = θ3 = (5+1)/(17+1*3) = 0,30.

Essa fórmula mostra que, quando um dado novo i que nunca apareceu antes precisa ser testado no modelo, em vez de receber probabilidade zero, acaba recebendo a probabilidade de:

θi = 1/(N + d)

Quando alfa = 0, o cálculo elimina o fator suavização:

θi = xi/N