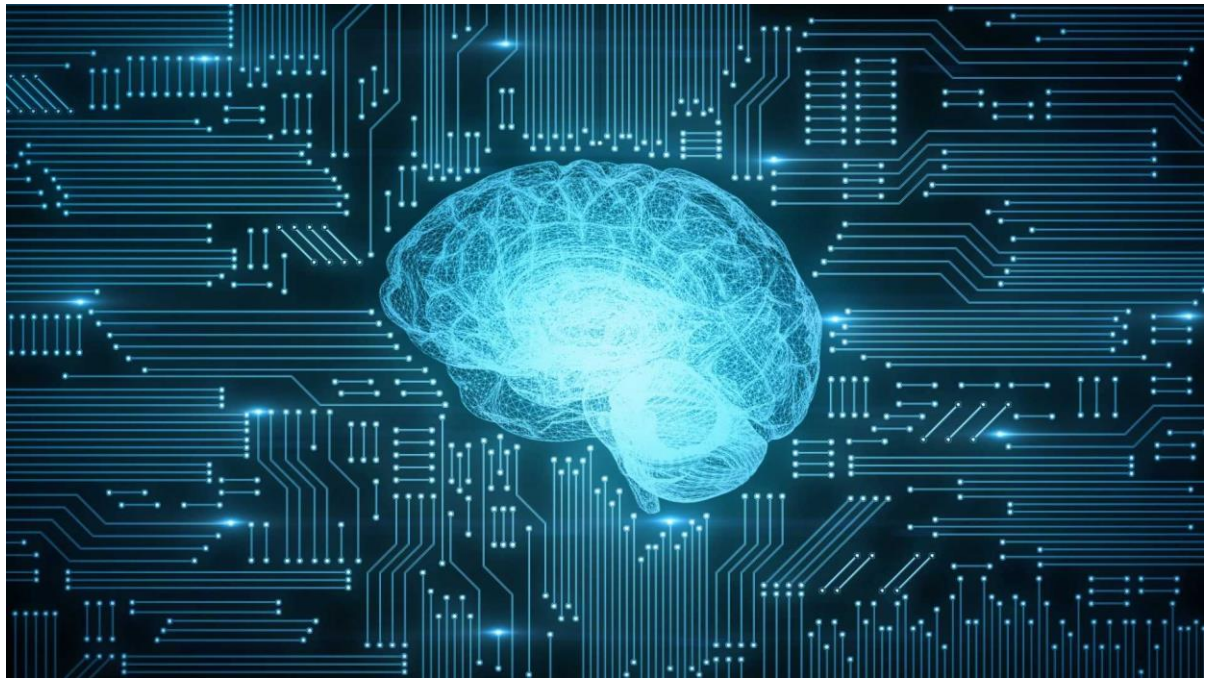
FGV EESP
ESCOLA DE ECONOMIA DE SÃO PAULO

INTELIGÊNCIA COMPUTACIONAL



PROF. Dr. EDSON C. KITANI

2022

Sumário da Aula

- Teoria do Classificador de Bayes
- Custo da Decisão (Duda et al. Cap. 2)
- K-NN e Parzen Window (Duda et al. Cap. 2)
- Espaços vetoriais de alta Dimensão (Hastie et al. – Cap. 2, Duda Cap. 3.7)

Classificadores de Bayes

CLASSIFICADOR NAIVE BAYES

O classificador NB considera que as características de um vetor aleatório são descorrelacionadas entre as classes. Mesmo se essas características dependam entre si ou pela existência de outras características, ele considera que as características contribuem independentemente para a probabilidade de cada classe. E é essa a característica que dá o nome “naïve”.

A parte de Bayes considera que as quantidades de interesse são governadas pela distribuição de probabilidades e que as decisões ótimas podem ser feitas pelo raciocínio sobre essas probabilidades $P(\omega_j|\mathbf{x})$ juntamente com os dados observados $(\mathcal{D})$.

$$P(\omega_i|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_i, \mathcal{D})P(\omega_i|\mathcal{D})}{\sum_{j=1}^{C} p(x|\omega_j, \mathcal{D})P(\omega_j)}$$

O problema é sempre estimar $p(\mathbf{x}|\omega_j, \mathcal{D})$ justamente pelo número de amostras limitado.

$$P(x_1, \ldots, x_n|\omega_j) = \prod_i P(x_i|\omega_j)$$

CLASSIFICADOR NAIVE BAYES - CATEGÓRICO

Com Bayes é possível superar o problema anterior simplificando a hipótese e usando o conhecimento à priori que temos a partir da base de dados

$$\omega_{NB} = argmax_{\omega_j \in \Omega} P(\omega_j) \prod_i P(a_i | v_j)$$

Por exemplo, suponha que tenhamos a seguinte tarefa de classificar a seguinte entrada:

Clima = ensolarado, Temperatura = fria, Umidade = alta, Vento = forte

Observe que nesta tarefa de classificação estamos lidando com um conjunto de fatores que vão gerar uma decisão binomial, Jogar ou Não Jogar.

Classificador de Bayes

CLASSIFICADOR NAIVE BAYES - CATEGÓRICO

$$\omega_{NB} = argmax_{\omega_j \in \Omega} P(\omega_j) \prod_i P(a_i | v_j)$$

Calculando $\omega_{NB} = \underset{\omega_j \in \Omega}{argmax} P(\omega_j)$

$$P(Clima = ensolarado|\omega_j).P(Temperatura = fria|\omega_j).P(Umidade = Alta|\omega_j).P(Vento = forte|\omega_j)$$

Amostras	Clima	Temperatura	Umidade	Vento	Jogar Volei
1	Ensolarado	Quente	Alta	Fraco	Não
2	Ensolarado	Quente	Alta	Forte	Não
3	Nublado	Quente	Alta	Fraco	Sim
4	Chovendo	Agradável	Alta	Fraco	Sim
5	Chovendo	Fria	Normal	Fraco	Sim
6	Chovendo	Fria	Normal	Forte	Não
7	Nublado	Fria	Normal	Forte	Sim
8	Ensolarado	Agradável	Alta	Fraco	Não
9	Ensolarado	Fria	Normal	Fraco	Sim
10	Chovendo	Agradável	Normal	Fraco	Sim
11	Ensolarado	Agradável	Normal	Forte	Sim
12	Nublado	Agradável	Alta	Forte	Sim
13	Nublado	Quente	Normal	Fraco	Sim
14	Chovendo	Agradável	Alta	Forte	Não

Calcule a probabilidade para jogar e não jogar.

$$P(Clima = ensolarado|c_j) = ?$$
$$P(Temperatura = fria|c_j) = ?$$
$$P(Umidade = Alta|c_j) = ?$$
$$P(Vento = forte|c_j) = ?$$

Classificador de Bayes

Amostras	Clima	Temperatura	Umidade	Vento	Jogar Volei
1	Ensolarado	Quente	Alta	Fraco	Não
2	Ensolarado	Quente	Alta	Forte	Não
3	Nublado	Quente	Alta	Fraco	Sim
4	Chovendo	Agradável	Alta	Fraco	Sim
5	Chovendo	Fria	Normal	Fraco	Sim
6	Chovendo	Fria	Normal	Forte	Não
7	Nublado	Fria	Normal	Forte	Sim
8	Ensolarado	Agradável	Alta	Fraco	Não
9	Ensolarado	Fria	Normal	Fraco	Sim
10	Chovendo	Agradável	Normal	Fraco	Sim
11	Ensolarado	Agradável	Normal	Forte	Sim
12	Nublado	Agradável	Alta	Forte	Sim
13	Nublado	Quente	Normal	Fraco	Sim
14	Chovendo	Agradável	Alta	Forte	Não

Condição	Joga	Não Joga
Ensolarado	2/9	3/5
Nublado	4/9	0
Chovendo	3/9	2/5
Quente	2/9	2/5
Agradável	4/9	2/5
Fria	3/9	1/5
Hum. Alta	3/9	4/5
Hum. Normal	6/9	2/5
Vento Forte	3/9	3/5
Vento Fraco	6/9	2/5

$$P(Jogar|X) = P(Clima = ensolarado|c_j).P(Temperatura = fria|c_j).P(Umidade = Alta|c_j).P(Vento = forte|c_j)$$

Classificador de Bayes

Calculando a probabilidade de cada evento dado o conjunto das observações, temos:

$$P(Joga|\mathbf{X}) = P(Jogar).P(Clima = Ensol.|Joga).P(Temp = Fria|Joga).P(Umid. = Alta|Joga).P(Vento = Forte|Joga)$$

$$\frac{9}{14} \times \frac{2}{9} \times \frac{3}{9} \times \frac{3}{9} \times \frac{3}{9} = 0{,}0053$$

$$P(\neg Joga|\mathbf{X}) = P(\neg Jogar).P(Clima = Ensol.|\neg Joga).P(Temp = Fria|\neg Joga).P(Umid. = Alta|\neg Joga).P(Vento = Forte|\neg Joga) = 0.0206.$$

$$\frac{5}{14} \times \frac{3}{5} \times \frac{1}{5} \times \frac{4}{5} \times \frac{3}{5} = 0{,}0206$$

Normalizando as duas probabilidades tal que:

$$P(Jogar) = \frac{P(\omega_1)}{\sum_{i=1}^{N} P(\omega_i|\mathbf{X})} = 0{,}205 \;\; e \quad P(\neg Jogar) = \frac{P(\omega_2)}{\sum_{i=1}^{N} P(\omega_i|\mathbf{X})} = 0{,}795$$

$$P(Jogar) = \frac{0{,}0053}{0{,}0053 + 0{,}0206} \qquad P(\neg Jogar) = \frac{0{,}0206}{0.0053 + 0{,}0206}$$

Logo, decidimos que não haverá jogo com um probabilidade de 79,5%, considerando as condições climáticas.

Algoritmo do Classificador Naive Bayes Categórico no SKLEARN

O pacote Scikit Learn implementa 4 variações do algoritmo Naive Bayes. Um deles é o CategoricalNB, que é usado para aplicações categóricas, tal como no exemplo do jogo de vôlei ou em aplicações com textos, como por exemplo, detecção de spam, ou análise de opiniões numa pesquisa.

```
modelo.CategoricalNB(alpha, fit_prior, class_prior, min_categories)
```

alpha = 1 (default)

é um parâmetro de suavização que é usado para evitar que tenhamos algum termo do produtório com probabilidade zero. Isso pode ocorrer quando alguma instância apresentar algum valor x_i não presente duran te o treinamento. O valor default é 1, mas ele aceita qualquer valor float. À medida que o alpha aumenta, ele leva a probabilidade da distribuição de cada classe para 0,5 tornando a distribuição desse fator uniforme.

Esse parâmetro é importante, pois durante os testes podem aparecer amostras com fatores que não existem no treinamento, e ele evita que o resultado do produtório dê probabilidade zero. O algoritmo usa o método de suavização de Laplace*, adicionando o fator α na contagem.

$$P(x_i = t|\omega_k, \alpha) = \frac{(\#x_i = t \in \omega_k) + \alpha}{(\#\ amostras \in \omega_k) + \alpha(dimensionalidade)}$$

*https://nlp.stanford.edu/IR-book/pdf/irbookonlinereading.pdf –page 297

Algoritmo do Classificador Naive Bayes - SKLEARN

fit_prior = True (default) faz o algoritmo usar a probabilidade à priori da classe, do contrário ele usará a distribuição uniforme.

class_prior = Nome (default) é o parâmetro na qual você ajusta a sua própria probabilidade da classe.

min_categories = None (default) define o número de características de cada classe, do contrário ele usa o número definido pelo conjunto de treinamento.

Algoritmo do Classificador Naive Bayes - SKLEARN

A biblioteca do CategoricalNB tem também alguns métodos. O principais são:

fit(X, y) - treina o modelo com a a base de dados X.

predict(X) - executa testes com amostras

predict_proba(X) - retorna o estimativa da probabilidade do teste com a amostra X

predict_log_proba(X) - retorna a estimativa no espaço de logarítmico para evitar problemas com valores muito pequenos.

score(X, Y) - retorna a acurácia média sobre as amostras de teste e rótulos.

CLASSIFICADOR NAIVE BAYES MULTINOMINAL

No modelo anterior utilizamos um classificador baseado em Bayes para uma tarefa de classificação binomial (Jogar ou Não Jogar). Entretanto, é muito comum tarefas de classificação que apresentam múltiplos resultados binomiais, tais como no processamento de linguagem natural (NLP).

$$\hat{\theta}_{yi} = \frac{(\sum x \in T) + \alpha}{\sum_{i=1}^{n} N_{yi}}$$

Onde $(\sum x \in T)$ é a contagem das ocorrências de x da classe y no conjunto de treino T, e $\sum_{i=1}^{n} N_{yi}$ é a contagem de todas as ocorrências da classe y.

CLASSIFICADOR NAIVE BAYES GAUSSIANO

Quando os atributos do nosso conjunto de dados é numérico precisamos alterar o método de estimação da probabilidade das classes $P(x_i|\omega_k)$. A classificação de uma nova amostra será dará pelo rótulo da classe que retornar a maior probabilidade.

$$\omega_{NB} = argmax_{\omega_j \in \Omega} P(x_i|\omega_k)$$

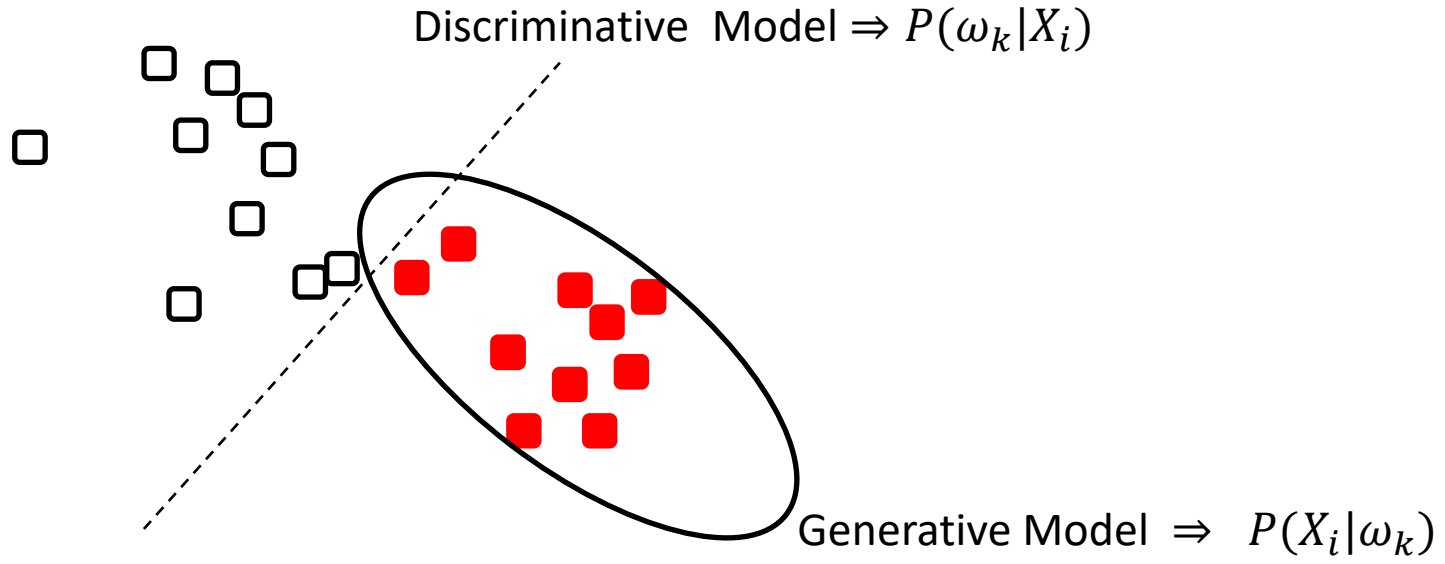
$$P(x_i|\omega_k) = \frac{1}{\sqrt{2\pi\sigma_y^2}} exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right)$$

Classificador de Bayes

O Naive Bayes é um classificador conhecido também como **Generative Model**, pois ele estima as novas amostras baseado nas probabilidades a priori obtidas pelas classes do conjunto de treinamento.

$$P(X|c_i) = \prod_{j=1}^{N} p(x_j|c_{i)}, \qquad i = 1, 2, \dots C$$

Computa para cada observação a probabilidade da classe. $P\left(X_1, \dots, X_N \middle| c_j\right)$.



Classificador de Bayes

Exercícios:

a) Usando a base de dados do jogo de Vôlei, calcule a probabilidade de haver jogo se forem observadas as seguintes condições: Clima = Nublado, Temperatura = Quente, Umidade = Normal, Vento = Forte

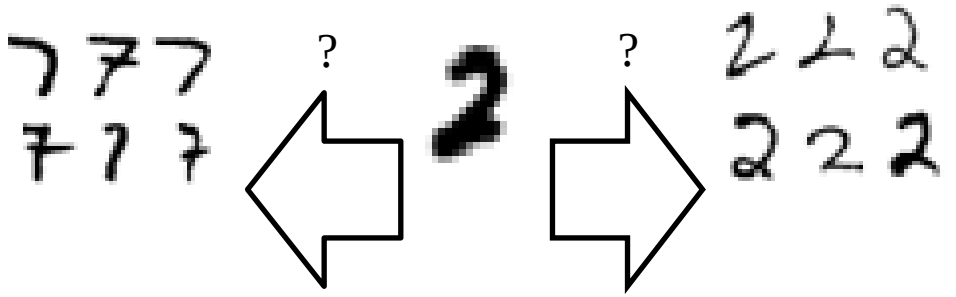
b) Modifique o programa da aula 1 (NaiveBayes.jpynb) para que ele forneça a estimativa da probabilidade de P($\omega = jogo$).

c) Faça o valor alpha = 0 e rode novamente o programa.

d) Analise os resultados.

Multinomial Naive Bayes Classifier

Considere que temos o banco de dados número manuscritos MNIST. Desenvolva um classificador Bayesiano para solucionar o problema abaixo.

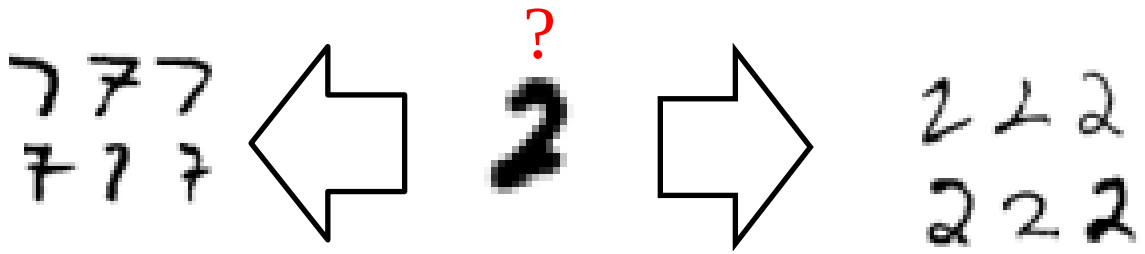


Amostra do banco MNIST.

Multinomial Naive Bayes Classifier

Considere que temos um banco de dados número manuscritos que são usados para treinar um classificador Naive Bayes Multinomial. Cada dígito é convertido para uma imagem *pixelada* de 28X28 bits. A dimensionalidade desse espaço será $\mathbb{R}^{784}$.

O número de combinações possíveis, considerando que cada pixel pode assumir 0 ou 1, seria de $2^{784} = 101{,}7 \times 10^{234}$.



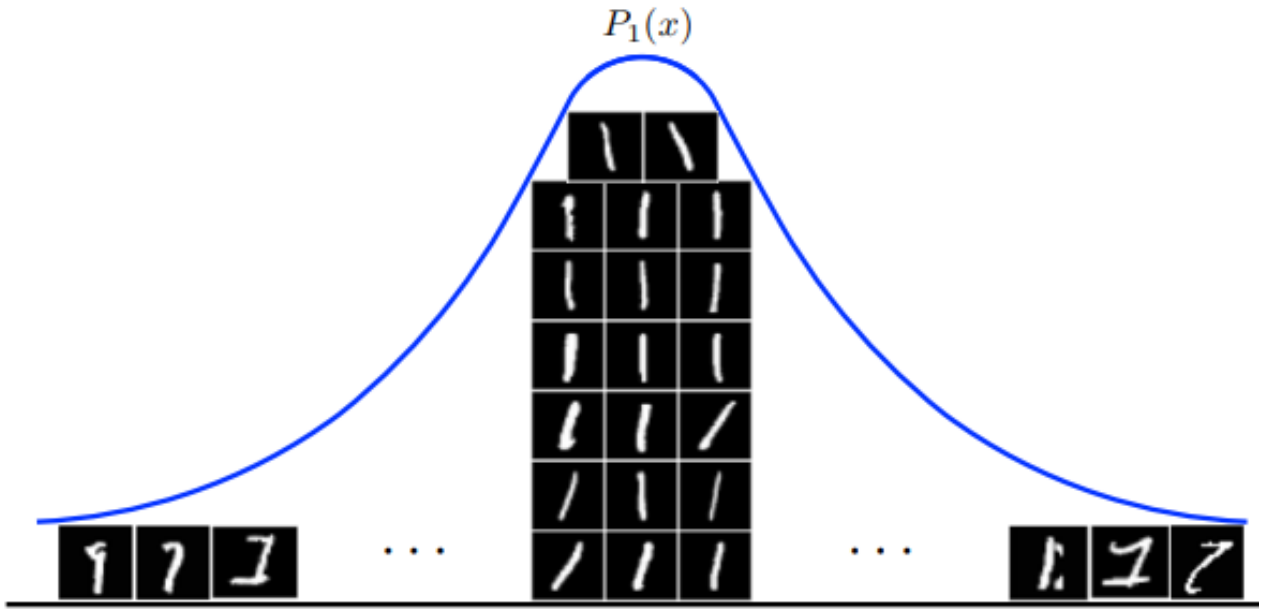
Para cada dígito $j \in \{0, 1, \ldots 9\}$ a probabilidade $P_j(x_i) = P_j(x_1)P_j(x_2) \ldots P_j(x_{784})$

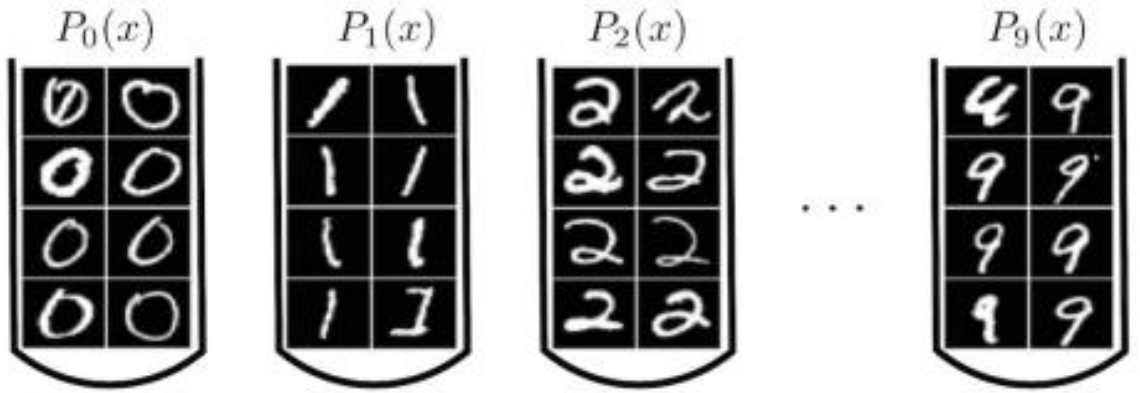
Na prática temos apenas poucas observações de cada classe, que neste caso são 10 classes. A ideia é então estimar a probabilidade condicionada à classe para cada pixel.

$$P(X|c_i) = \prod_{j=1}^{n} p(x_j|c_{i)}$$

Multinomial Naive Bayes Classifier

Exemplo da distribuição de $P_1(x)$ considerando as nossas incertezas sobre como o número 1 pode ser representado de modos diferentes. Note que há uma grande ambiguidade em alguns dígitos e que podem criar erros de interpretação. E a mesma interpretação da probabilidade à priori pode ser observado nos outros números também.





Gaussian Bayes Classifier

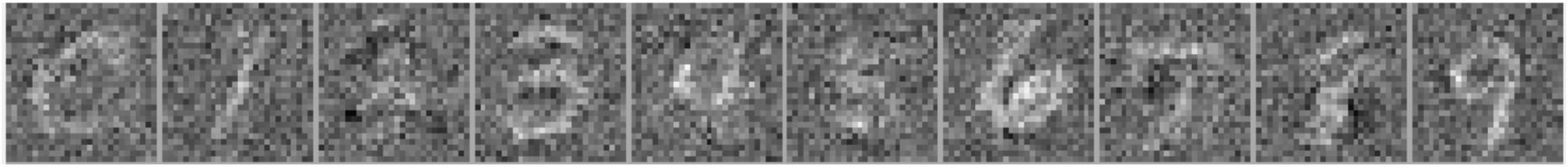
O modelo de Bayes Gaussiano considera que cada pixel está correlacionada multivariadamente para formar a imagem de um dígito, segundo uma média μ e uma covariância σ.

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right), \qquad x \in \mathbb{R}.$$

Portanto, dado uma imagem $x \in \mathbb{R}^{784}$, a classificação será definida pelo MAP dado abaixo:

$$c(\mathrm{X}) = argmax_j\left(c_j . P_j\,(\mathrm{X})\right)$$

Umesh Vazirani – CS 70 - 2015



- ✓ **Naive Bayes Multinomial:** Assume que $P_j(\mathrm{X})$ é um produto independente de cada pixel na formação da imagem.
- ✓ **Modelo Gaussiano:** Assume que $P_j(\mathrm{X})$ é um distribuição Gaussiana multivariada. Isso permite modelar uma correlação entre os valores dos *pixels* (0-255).

Resumo

O classificador de Bayes é um abordagem simples e probabilística, mas que pode ser usada antes de qualquer outra mais complexa, principalmente para compreender a estrutura dos dados. Use o conceito do *Occan Razor e Kiss (Keep it Simple)*

- ✓ Sempre temos a probabilidade da classe $P(\omega_j), j = 1, 2, \ldots C$ em aplicações supervisionadas.
- ✓ A partir do conjunto de treinamento podemos aprender o modelo probabilístico $P(\mathrm{X}|y = \omega_j)$.
- ✓ Usando a regra de Bayes $P(\omega_j|\mathrm{X}) = \frac{P(\mathrm{X}|y=\omega_j).P(\omega_j)}{\sum_{i=1}^{C} P(\mathrm{X}|y=\omega_j).P(\omega_j)}$ podemos determinar a probabilidade à posteriori.

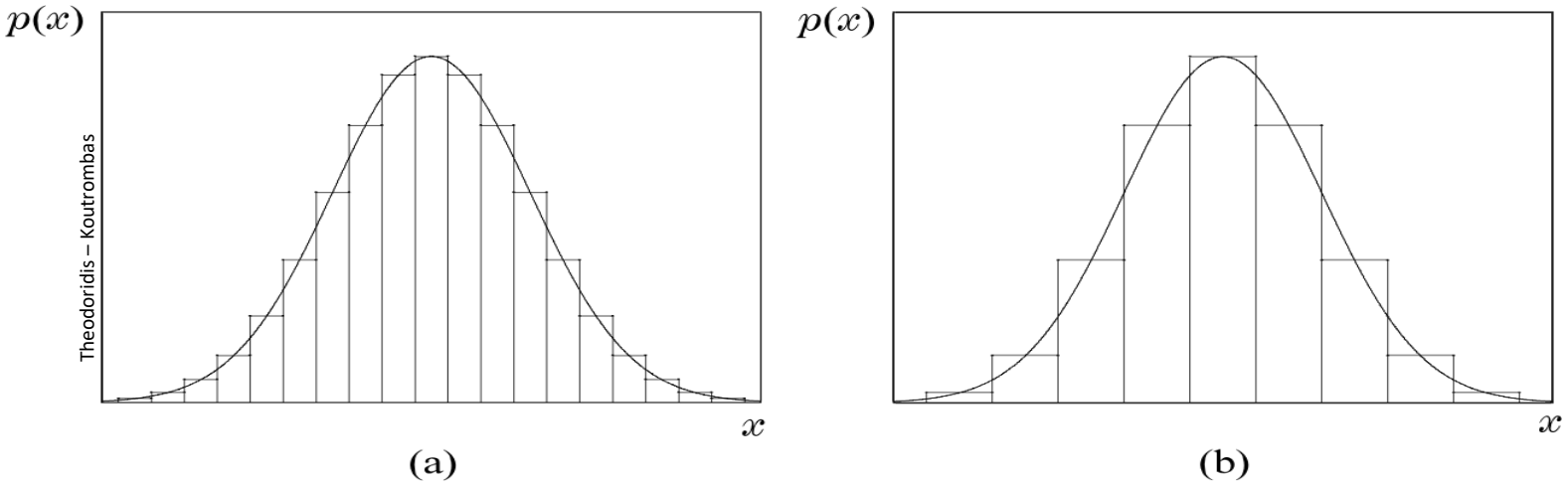
K-NN

Estimação não Paramétrica

No classificador de Bayes Gaussiano é sempre assumido que há uma função densidade de probabilidade dependente de parâmetros tais como: classe, média e covariância.

Entretanto, nem sempre é possível estimar a forma da função global de densidade de distribuição dos dados. Assim, para essas situações, cuja função global de densidade são desconhecidas, podemos utilizar técnicas de estimação paramétrica local, também conhecidas como Estimação não Paramétrica.

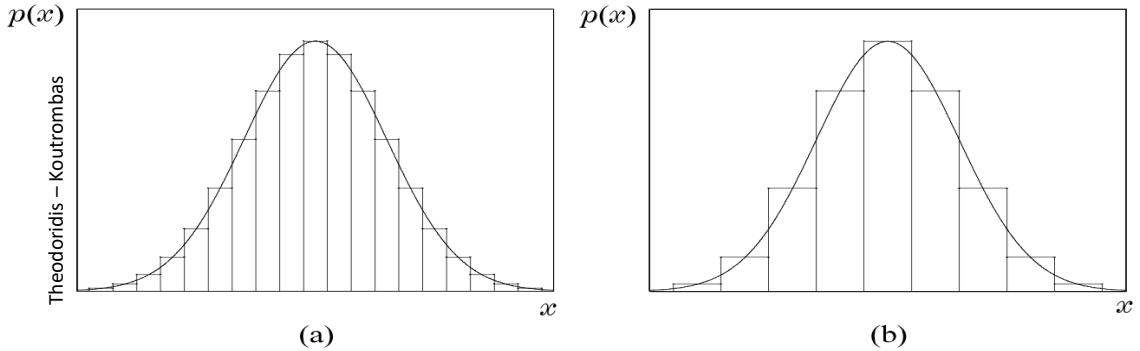
Existem duas técnicas de estimação bastante conhecidas que são a Janela de Parzen e K-NN (K-Nearest Neighbor), e são uma variação do conceito de histogramas.



Theodoridis – Koutrombas

Estimação não Paramétrica

No histograma da figura (a) observa-se que a largura h dos bins são menores do que em (b). Isso aumenta o grau de aproximação da estimação em direção à forma real da distribuição real dos dados.



Theodoridis – Koutrombas

Então, a probabilidade P(X) de uma amostra X ser alocada num bin é estimada para cada bin.

$$P \approx \frac{k_N}{N}$$

Onde k_N é número de amostras dentro de cada bin e N o número total de amostras, então:

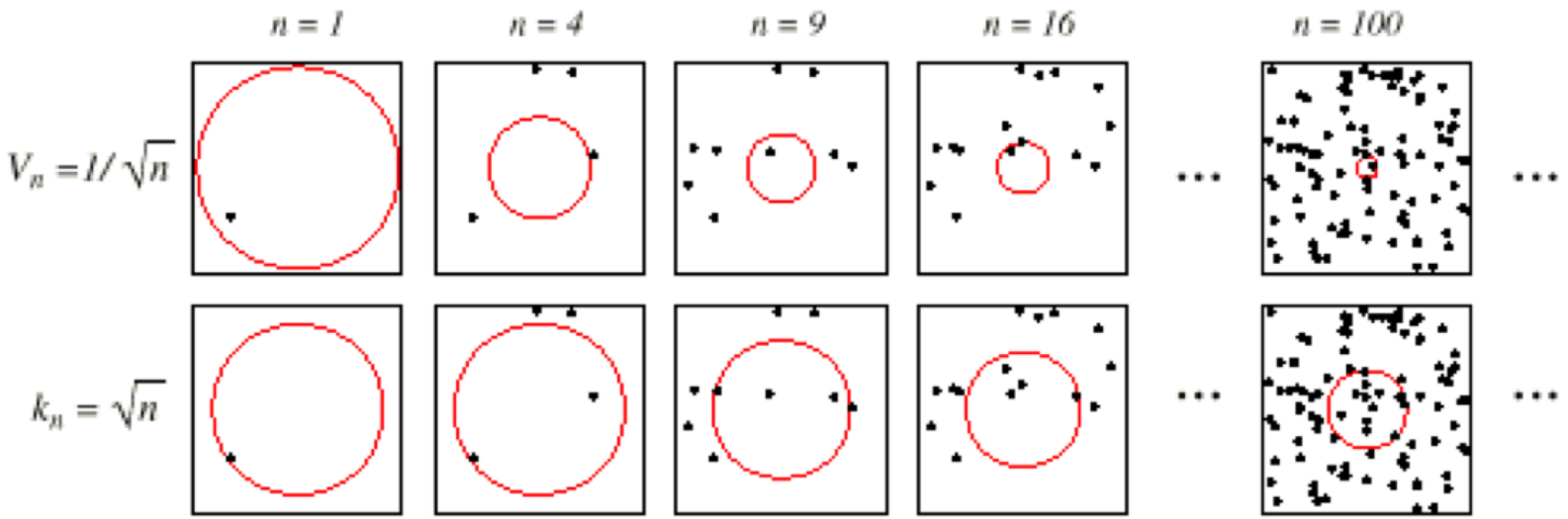
$$\hat{p}(\hat{X}) = \approx \frac{1}{h}\frac{k_N}{N} \quad e \quad \left| X - \hat{X} \leq \frac{h}{2} \right|,$$

é uma aproximação para cada bin, onde $\hat{X}$ é o centro do bin. Assim, considera-se que um ponto X está dentro do bin se sua distância até o centro do bin seja menor do que $^{h}/_{2}$. Lembrar que o espaço de X será dividido por bins de comprimento h.

K-NN

Abaixo estão ilustrados dois métodos para estimar a densidade de probabilidade num ponto. O primeiro caso, começamos com um volume grande centrado num ponto de interesse e depois vamos reduzindo de acordo com a função $V_n = \frac{1}{\sqrt{n}}$. Nos exemplos abaixo o ponto está no centro do quadrado.
Outro método é reduzir o volume em função dos dados contidos no volume, por exemplo $k_n = \sqrt{n}$.

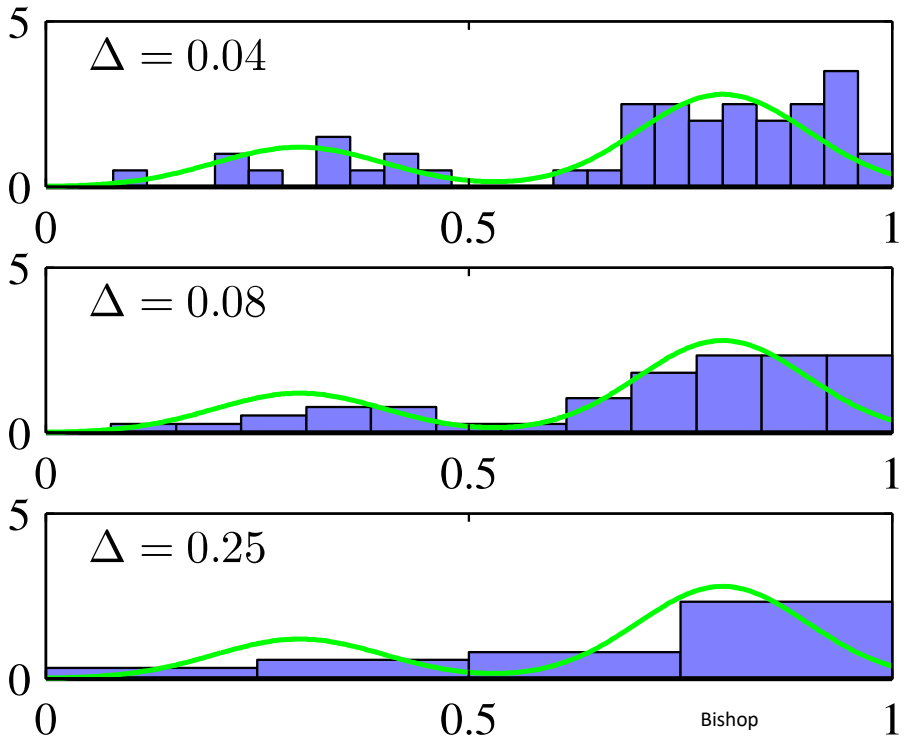
Atenção, o raio do circulo é apenas uma referência.



Duda, Hart, Stork

Estimação não Paramétrica

Figure 2.24 An illustration of the histogram approach to density estimation, in which a data set of 50 data points is generated from the distribution shown by the green curve. Histogram density estimates, based on (2.241), with a common bin width Δ are shown for various values of Δ.



Onde k_N representa o número de amostras de N que pertencem a uma região $\mathcal{R}$ de dimensão $\Delta = h$, e supondo que temos uma amostra x que pertence a um certo bin do histograma, podemos escrever que:

$$\hat{p}(X) \approx \frac{1}{h}\frac{k_N}{N}$$

Janela de Parzen

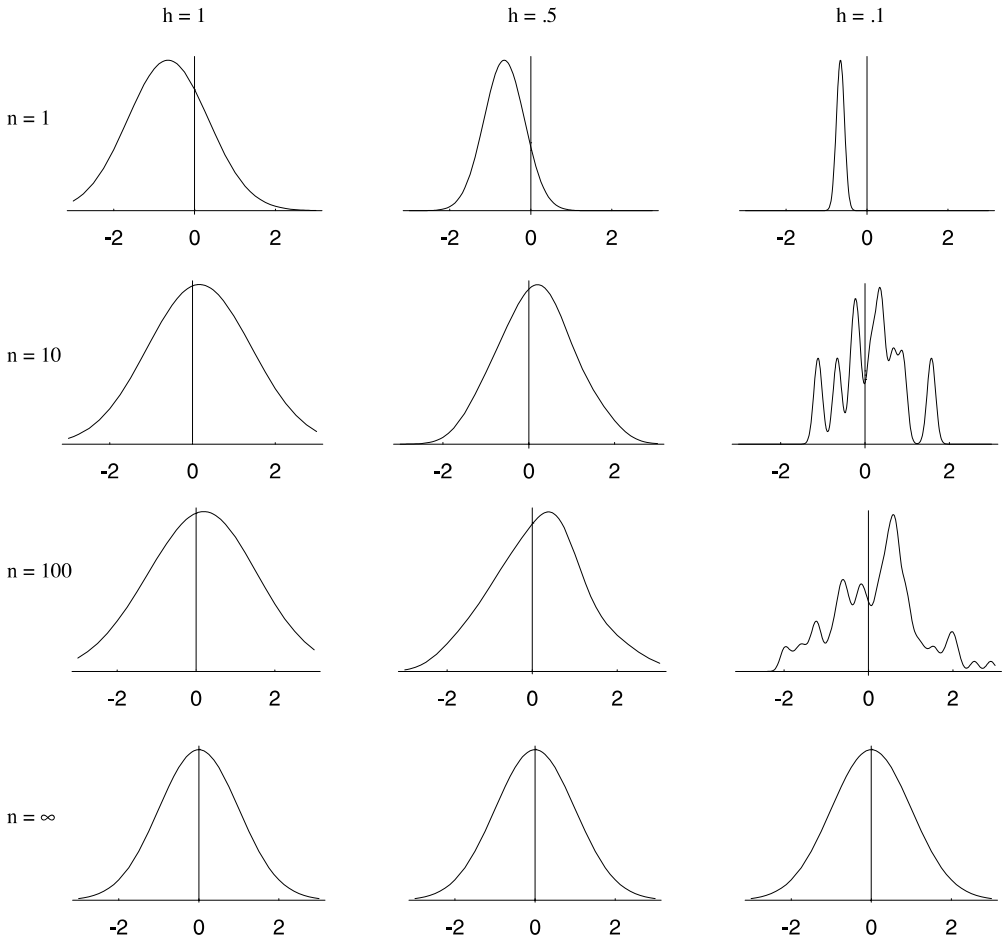


Figure 4.5: Parzen-window estimates of a univariate normal density using different window widths and numbers of samples. The vertical axes have been scaled to best show the structure in each graph. Note particularly that the $n = \infty$ estimates are the same (and match the true generating function), regardless of window width h.

Duda, Hart, Stork

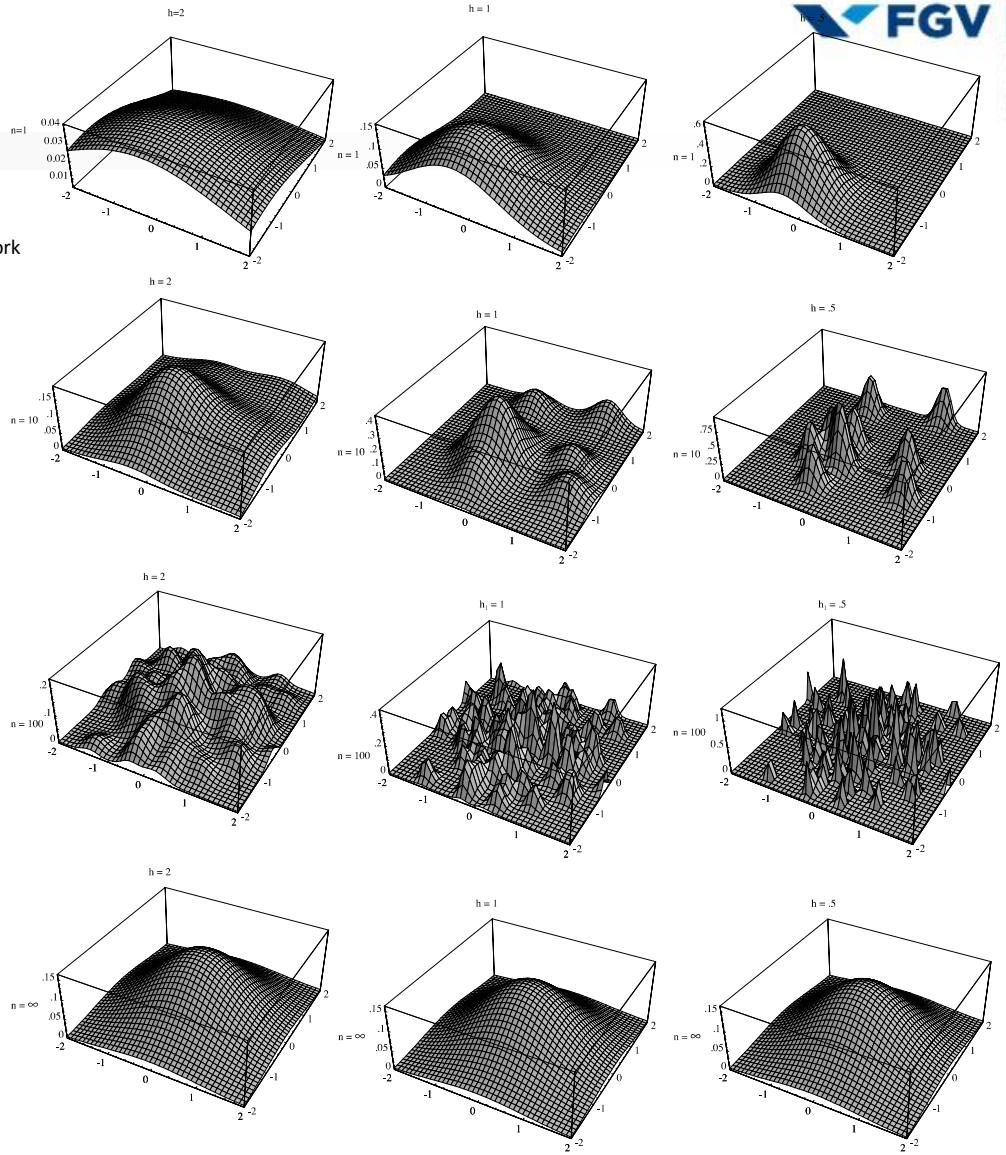


Figure 4.6: Parzen-window estimates of a bivariate normal density using different window widths and numbers of samples. The vertical axes have been scaled to best show the structure in each graph. Note particularly that the $n = \infty$ estimates are the same (and match the true generating distribution), regardless of window width h.

Janela de Parzen – Demonstração da Convergência

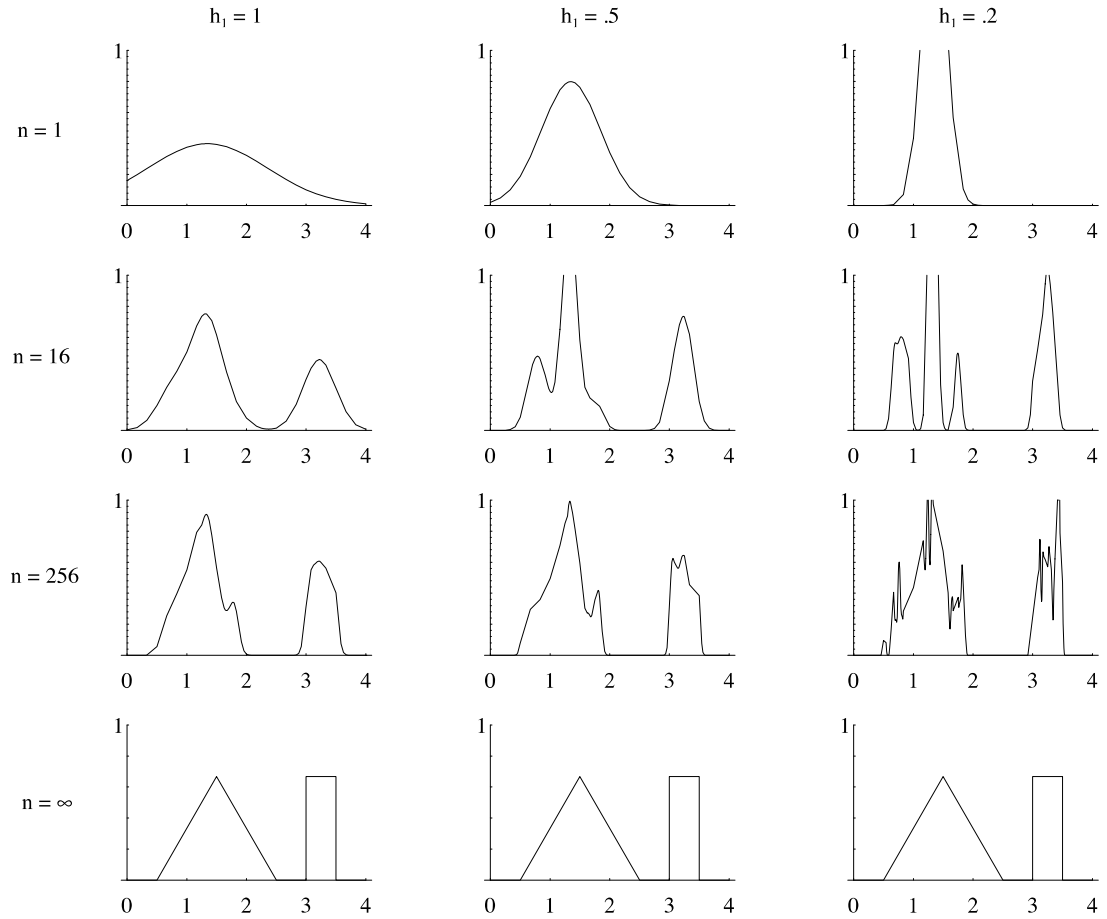
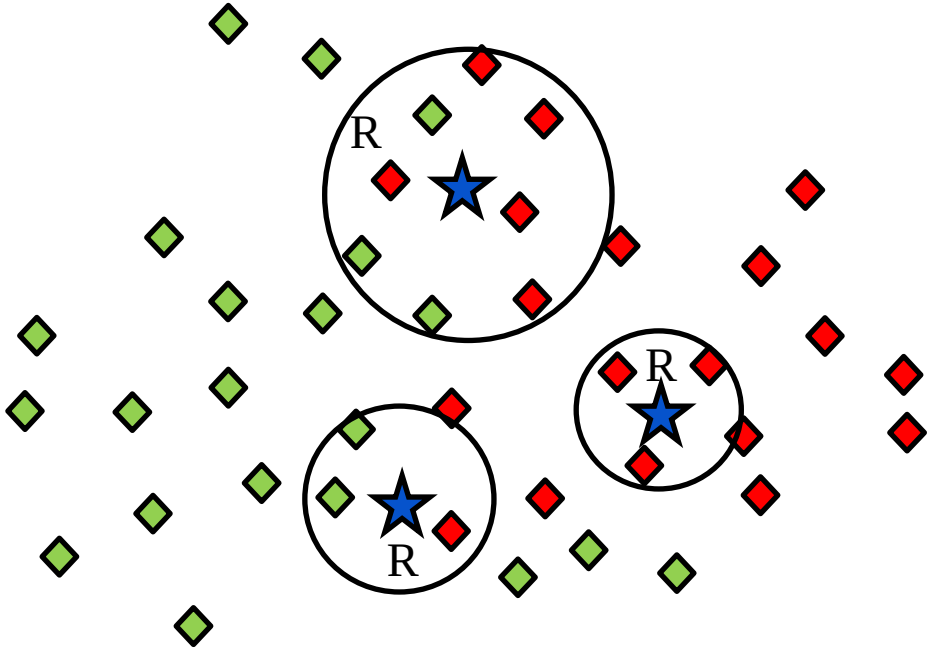
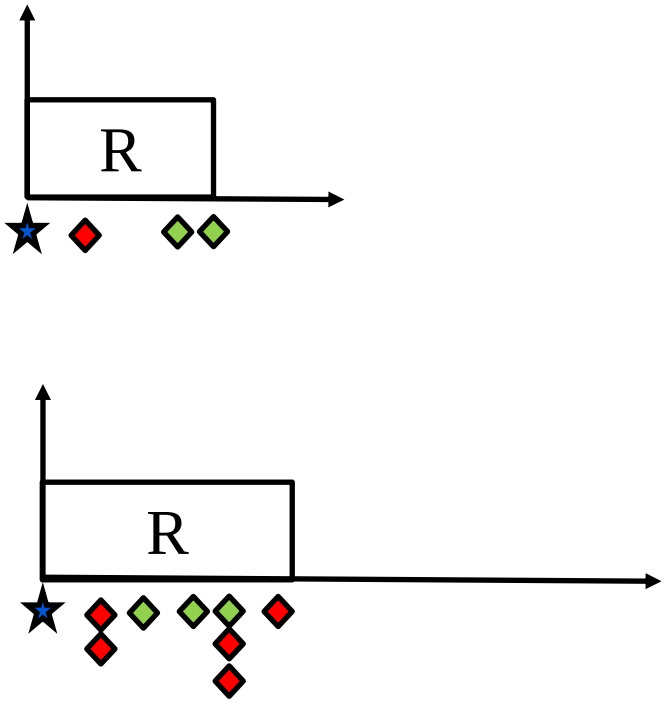


Figure 4.7: Parzen-window estimates of a bimodal distribution using different window widths and numbers of samples. Note particularly that the $n = \infty$ estimates are the same (and match the true generating distribution), regardless of window width h.

Duda, Hart, Stork

Classificador k-NN e Parzen



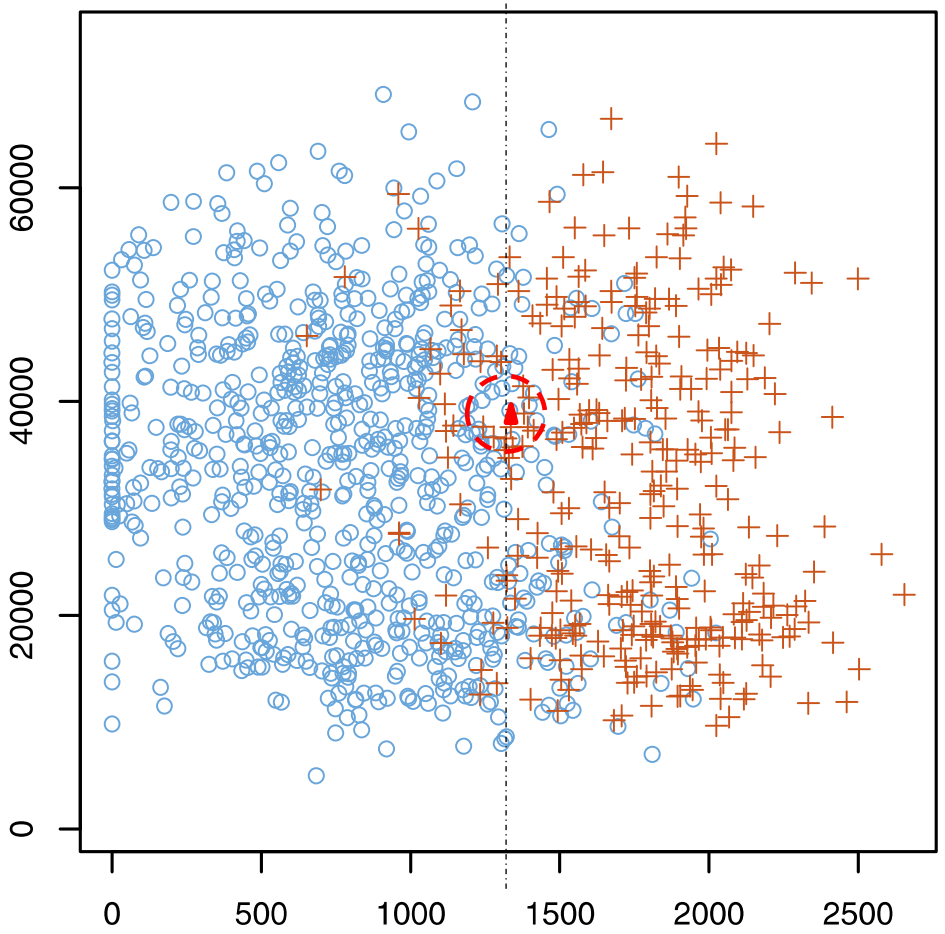


K-NN – Quantidade de amostra fixo.

Parzen – Volume da hiperesfera fixo.

Estimação não Paramétrica

Classificação do triângulo em vermelho.



Introduction to Statistical Learning – Casella et al.

Grade de Voronoi

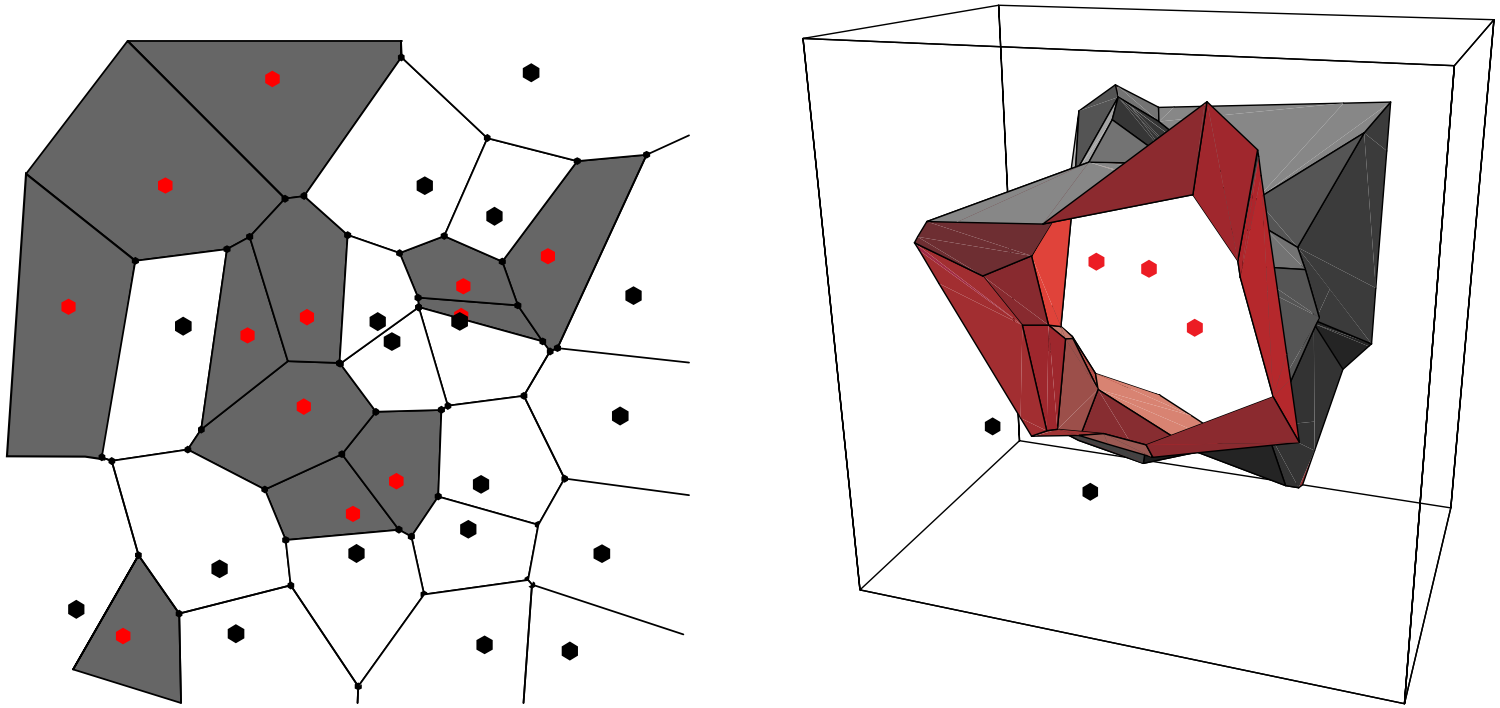


Figure 4.13: In two dimensions, the nearest-neighbor algorithm leads to a partitioning of the input space into Voronoi cells, each labelled by the category of the training point it contains. In three dimensions, the cells are three-dimensional, and the decision boundary resembles the surface of a crystal.

Duda, Hart, Stork

Cálculo de Volumes em $\mathbb{R}^n$

O volume de um hipercubo de lado l pode ser determinado por:

$$v(l) = l^n$$

O volume de uma hiperesfera de raio r pode ser determinado por:

$$v(r) = 2r \ \ para\, n = 1$$

$$v(r) = \ \pi r^2 \ para\, n = 2$$

$$v(r) = \ \frac{4}{3}\pi r^3 \ para\, n = 3$$

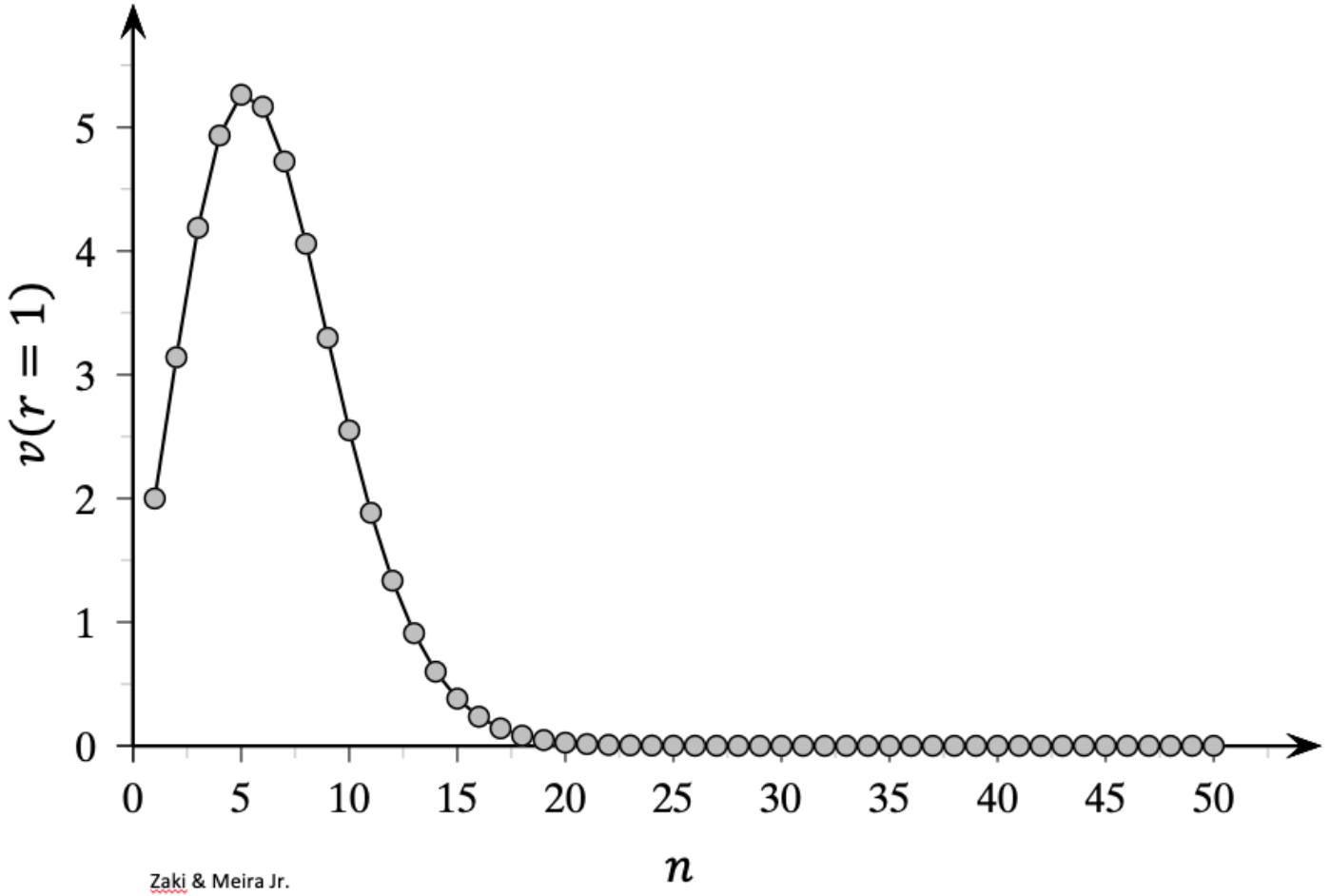
Para um caso geral:

$$v(r) = K_d r^n = \left(\frac{\pi^{n/2}}{\Gamma\left(\frac{n}{2} + 1\right)} \right) r^n$$

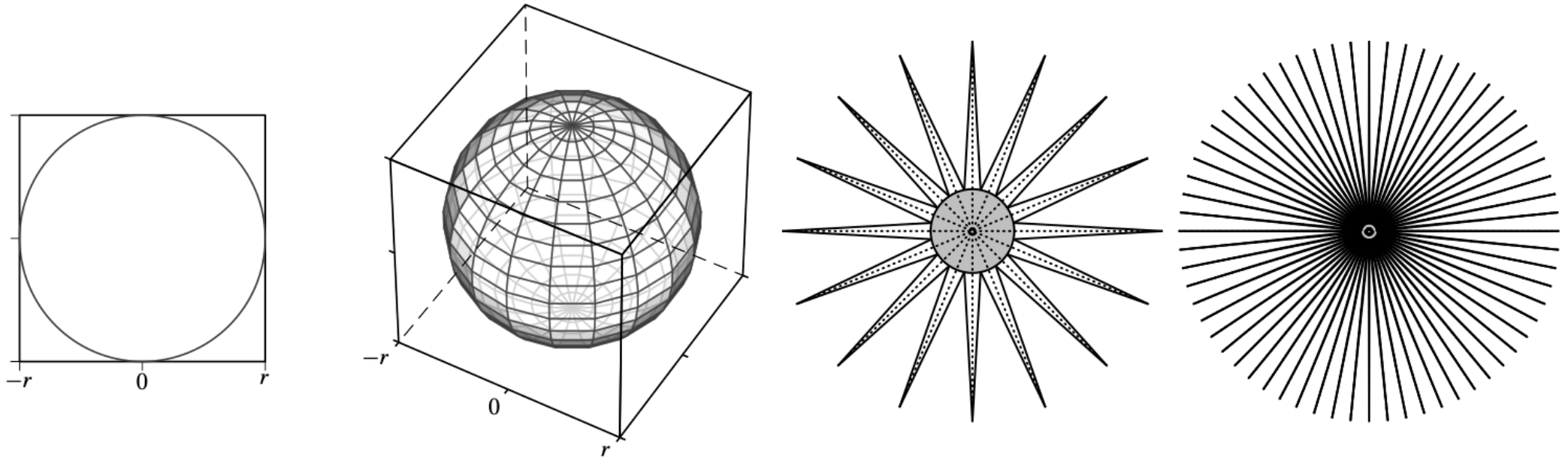
Onde a função $\Gamma\left(\frac{n}{2} + 1\right) = \begin{cases} \left(\frac{n}{2}\right)! & se\ n\ é\ par \\ \sqrt{n}\left(\frac{n!!}{2^{(n+1)/2}}\right) & se\ n\ é\ impar \end{cases}$

$$n!! = \prod_{i=0}^{k} (n - 2i) = n(n-2)(n-4)$$

Volume da Hiperesfera em função de $\mathbb{R}^n$



Hiperesfera inscrita num Hipercubo



O número de arestas do hipercubo é de 2^n e o número de diagonais é de $2^{(n-1)}$

Praga da Dimensionalidade

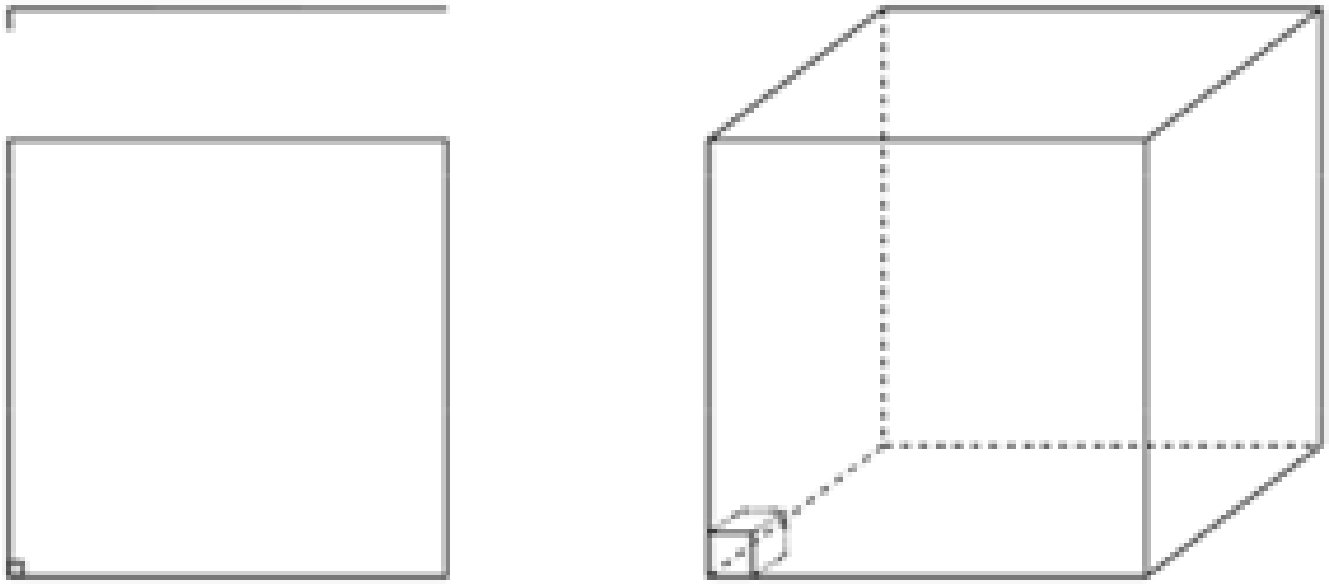
A dimensionalidade do espaço de entrada é um aspecto muito relevante e que afeta todas as metodologias multivariadas. A Maldição da Dimensionalidade foi um termo cunhado por Richard Bellman durante os estudos sobre controles adaptativos.

Bellman postulou que, dado um hipercubo de 10 dimensões, na qual cada lado *Li=1, i=1, 2,…,10*, e se cada lado do cubo for dividido em espaçamentos iguais de um décimo, ter-se-á um total de pontos de 10^{10}. Caso o número de lados do hipercubo seja de 20 lados com os mesmos espaçamentos no plano Cartesiano, obtém-se 10^{20} pontos. O crescimento do número de pontos é exponencial.

E quando se tem uma estrutura de dados multidimensional desconhecida, todos os pontos são importantes para aprender a descrevê-lo. Logo, a Maldição da Dimensionalidade encontra-se na limitação de se obter um número significativo de amostras que preencham o volume da estrutura de dados e que permita uma estimação adequada da superfície. A ausência de amostras em certas regiões fará com que os estimadores preencham os espaços de forma aleatória, degenerando a estrutura de dados original.

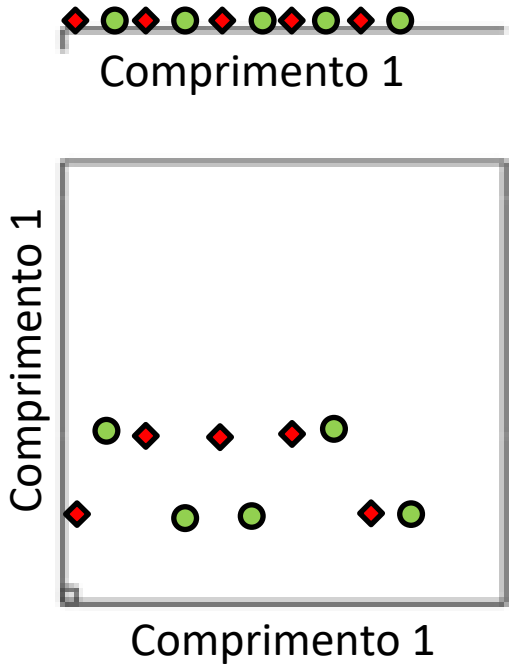
Praga da Dimensionalidade

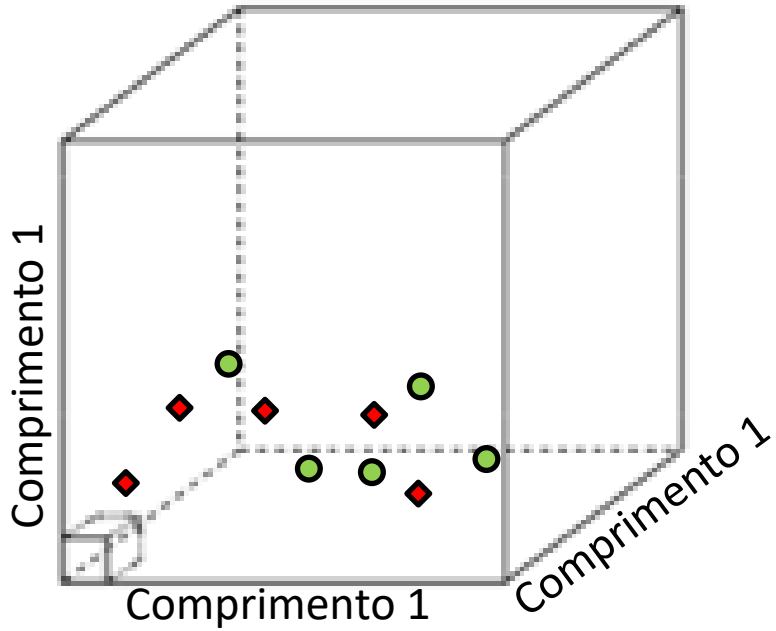
Conforme a dimensionalidade aumenta, o número de amostras necessárias para representar o espaço de entrada aumenta consideravelmente. Considere o seguinte exemplo. Suponha que num segmento de reta tenhamos 10 pontos distribuídos igualmente no dois intervalos. Considere então que o espaçamento seja de $d_{xi,x1+1} = {}^{1}/_{10}$. Então teremos 10 pontos representando essa reta. Agora imagine o mesmo para um plano, um cubo, ou qualquer outra dimensão, mas mantendo o mesmo espaçamento. Veja o quanto o número cresce.



Praga da Dimensionalidade

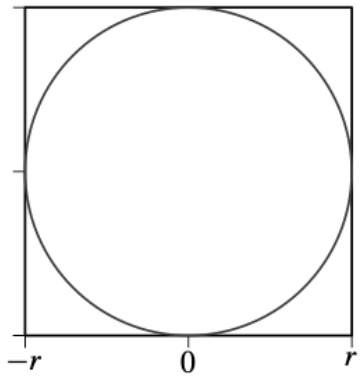
Outra forma de ver o problema da dimensionalidade é quando temos poucas amostras e a dimensionalidade dessas amostras vai aumentando.

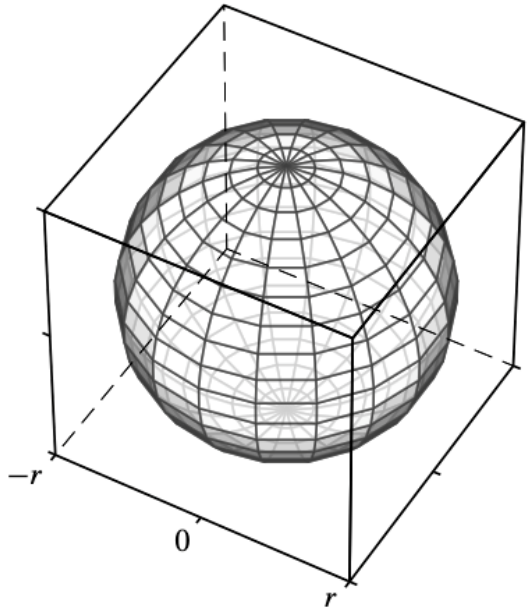


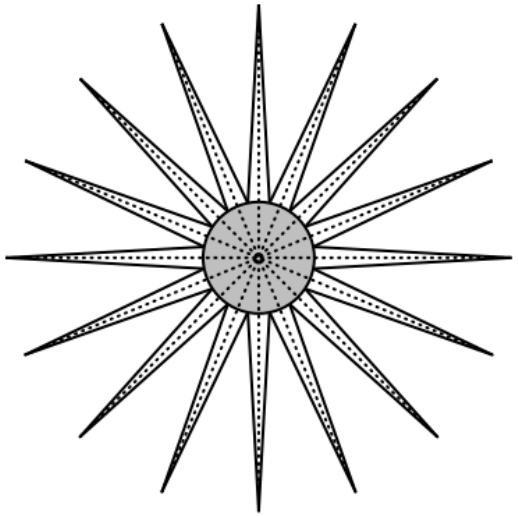


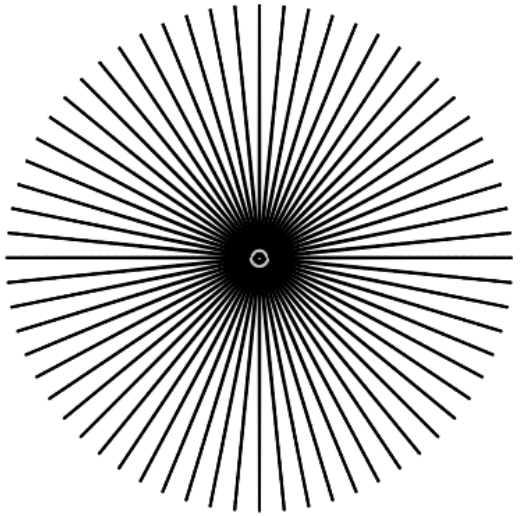
Praga da Dimensionalidade

Amostras que que caem fora do circunferência no espaço de duas dimensões, irão para os vértices do hipercubo. Assim, algoritmos como K-NN podem sofrer, pois os k vizinhos mais próximos vão se distanciando a medida que a dimensionalidade aumenta.









Teoria de Decisão

Teoria de Decisão Bayesiana

Esta abordagem é baseado na quantificação do equilíbrio entre várias decisões de classificação usando a probabilidade e o custo que acompanha cada decisão $\Longrightarrow \ \hat{y} = argmax(P(y|\mathbf{x})$.

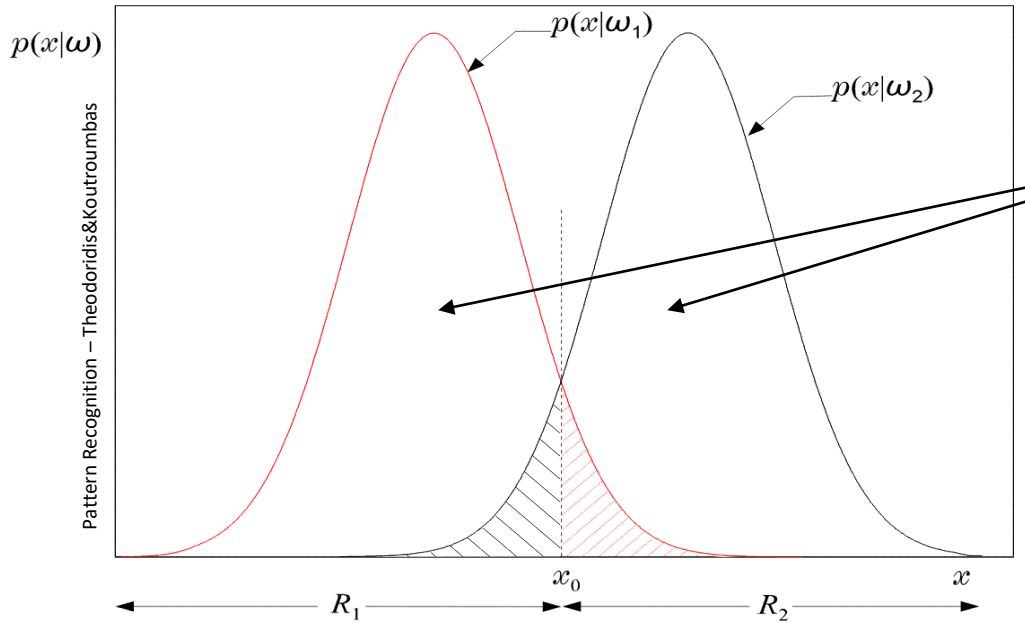
Seja $C = \{c_1, .., c_d\}$ um conjunto finito de d classes que são os estados da natureza e seja $\mathrm{A} = \{\alpha_1, \dots, \alpha_a\}$ o conjunto finito de a ações possíveis. A função $f(\alpha_i|c_j)$ fornece o custo da ação α_i quando o estado da natureza é c_j.

Agora considere um vetor aleatório $\mathbf{x}$ e seja $p(\mathbf{x}|c_j)$ a função densidade de probabilidade condicional do $\mathbf{x}$ associado ao estado da natureza c_j. Por Bayes podemos determinar a probabilidade a posteriori de $P(c_j|\mathbf{x})$, ou seja determinar a que classe pertence a amostra, dado o conhecimento prévio que temos das classes e das amostras anteriores.

$$P(c_j|\mathbf{x}) = \frac{p(\mathbf{x}|c_j)P(c_j)}{\sum_{j=1}^{d} p(\mathbf{x}|c_j)\,P(c_j)}$$

Minimização da Probabilidade de Erro

Exemplo de duas regiões R1 e R2 formadas pelo Classificador Bayesiano para duas classes.



$Juntas\ formam\ p(x)$

$$2Pe = \int_{-\infty}^{x_0} p(x|c_2)dx + \int_{x_0}^{\infty} p(x|c_1)dx$$

Na qual Pe é o erro de decidir por c_1 quando $\mathbf{x} \in c_2$ e vice versa. Ou seja, a menor probabilidade de erro é dado por $P(c_1|\mathbf{x}) = P(c_2|\mathbf{x})$, que define a fronteira de decisão x0.

Minimização do Risco Médio

A classificação baseado na minimização da probabilidade do erro pode não trazer os melhores resultados, dado que ele dá a mesma importância para todos os erros. Como alternativa, temos o conceito de associar uma penalidade para cada erro.

Considere um problema de C classes e seja $R_j, j = 1, 2, \dots C$ as regiões das características associadas à classe c_j. Considere uma amostra $\mathbf{x}$ que pertence a classe c_k, mas que o classificador define como $\mathbf{x} \in R_i \mid i \neq k$. Então, é visível que o classificador cometeu um erro.

Vamos definir um termo $\lambda_{k,i}$ que é o custo pelo erro de classificação. Dessa maneira, podemos construir uma matriz $\mathcal{L}$ de dimensão $k \times i$ que corresponderá ao custo de cada decisão. Define-se como o risco ou perda associado à classe c_k

$$r_k = \sum_{i=1}^{C} \lambda_{k,i} \int_{R_i} p(\mathrm{x}|c_k) d\mathbf{x}$$

Como o objetivo é escolher a região que tenha o risco médio minimizado.

$$r = \sum_{k=1}^{C} r_k P(c_k) = \sum_{k=1}^{C} \sum_{k=1}^{C} \lambda_{k,i} \int_{R_i} p(\mathbf{x}|c_k) P(c_k) dx$$

Minimização do Risco Médio

$$r_k = \sum_{i=1}^{C} \int_{R_i} \left(\sum_{k=1}^{C} \lambda_{k,i} p(\mathbf{x}|c_k) P(c_k) \right) d\mathbf{x}$$

A regra de decisão de menor risco é a região do espaço de características cuja integral acima seja a menor possível.

$$x \in R_i \; se \sum_{k=1}^{C} \lambda_{k,i} p(\mathbf{x}|c_k) P(c_k) < \sum_{k=1}^{C} \lambda_{k,j} p(\mathbf{x}|c_k) P(c_k) \qquad \forall j \neq i$$

Para um exemplo com duas classes, considerando a probabilidade a priori:

$$l_1 = \lambda_{11} p(x|c_1) P(c_1) + \lambda_{21} p(x|c_2) P(c_2)$$
$$l_2 = \lambda_{12} p(x|c_1) P(c_1) + \lambda_{22} p(x|c_2) P(c_2)$$

Que associa x para $c_1 se\ l_1 < l_2$. Para o caso de duas classes temos:

$$\mathbf{x} \in c_1 \; se \; l_{12} \equiv \frac{p(x|c_1)}{p(x|c_2)} > \frac{P(c_2)(\lambda_{21} - \lambda_{22})}{P(C_1)(\lambda_{12} - \lambda_{11})}$$

Minimização do Risco Médio

Vamos considerar um a matriz de perda $L = \begin{pmatrix} 0 & \lambda_{12} \\ \lambda_{21} & 0 \end{pmatrix}$. Observe que os riscos $\lambda_{12}\ e\ \lambda_{21}$ definirão a escolha do classificador. Se $\lambda_{21} > \lambda_{12}$ então os padrões serão associados à classe c_2. Observe que o risco para $\lambda_{11}\ e\ \lambda_{22} = 0$, pois considera-se que é a decisão correta.

Como regra geral para a classificação pelo risco de Bayes:

$$\mathbf{x} \in c_1\ se\ l_{12} \equiv \frac{p(\mathbf{x}|c_1)}{p(\mathbf{x}|c_2)} > \frac{P(c_2)(\lambda_{21} - \lambda_{22})}{P(c_1)(\lambda_{12} - \lambda_{11})}$$

E a fronteira de decisão estará definida quando:

$$\frac{p(\mathbf{x}|c_1)}{p(\mathbf{x}|c_2)} = \frac{P(c_2)(\lambda_{21} - \lambda_{22})}{P(c_1)(\lambda_{12} - \lambda_{11})}$$

Minimização do Risco Médio

Exercício:

Considere uma distribuição binária cuja densidade condicional para um padrão de entrada é dado por:

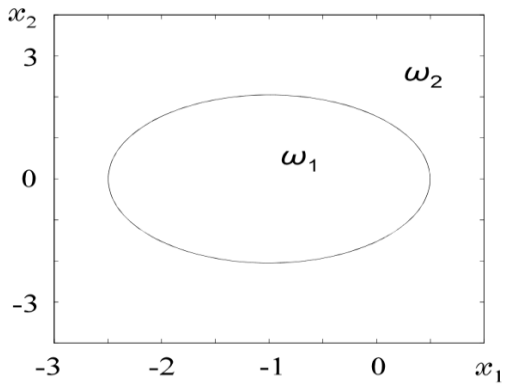
$$p(\mathbf{x}|\boldsymbol{c_1}) = \frac{1}{\sqrt{20\pi}} e^{\left(-\boldsymbol{x^2}/\boldsymbol{20}\right)}$$

$$p(\mathbf{x}|c_2) = \frac{1}{\sqrt{12\pi}} e^{\left(-(x-6)^2/12\right)}$$

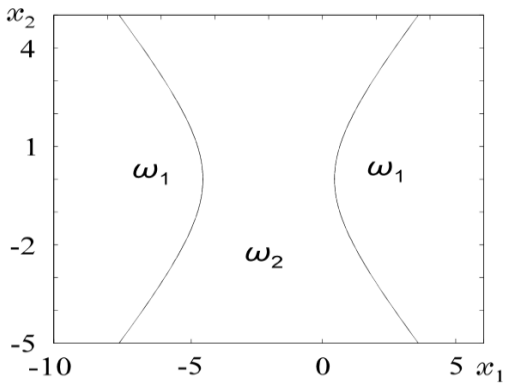
Sendo as médias $\mu_1 = 0$ e $\mu_2 = 6$, as variâncias $\sigma_1^2 = 6\ e\ \sigma_2^2 = 10$, a matriz de risco $\mathcal{L} = \begin{bmatrix} 0 & \sqrt{3} \\ \sqrt{5} & 0 \end{bmatrix}$ e $P(c_1) = P(c_2) = \frac{1}{2}$, determine a expressão do risco e as fronteiras de decisão.

Classificador Bayesiano

A fronteiras de decisão nos classificadores são quadráticos porque:



(a)



(b)