6. Nonparametric techniques
Motivation

Problem:
how to decide on a suitable model
(e.g. which type of Gaussian)

Idea:
just use the original data
(lazy learning)
Idea 1: each data point represents a piece of probability

\[ P(x) \]

a Parzen Window Method
Idea 2: ignore probabilities just measure distance to training data

- Consider two class problem

k-nearest neighbor classifier
Idea 2:
ignore probabilities
just measure distance to training data

• Consider two class problem

k-nearest neighbor classifier
6.1. Density Estimation
Goal

• Determine probability density $P(x)$

• Given: training data $x_1, x_2, ..., x_n$

• Consider region $R$

Q: should I continue from here as a white board lecture?
Estimate probability $P$ inside a region

- Probability of $x$ being in $R$

\[
P = \int_{R} P(x)dx
\]

- Suppose $k$ training vectors are inside $R$ from a total of $n$ training vectors

\[
P \approx \frac{k}{n}
\]

What's an estimate for $P$?
Consider \( n \rightarrow \infty \)

Sequences \( V_n \) (volume of region) and \( k_n \)

Resulting sequence of probabilities:

\[
P_n(x) = \frac{k_n}{nV_n}
\]
Expanding Number of Samples/Shrinking Volume in k-Nearest-Neighbour Estimation

\[ k_n = \sqrt{n} \]

From: Duda+Hart: Pattern Classification
Necessary conditions for convergence

Convergence means \( P_n(x) \to P(x) \) for \( n \to \infty \)

\( P(x) \) is local property: \( V_n \to 0 \) for \( n \to \infty \)

Reliability of estimate: \( k_n \to \infty \) for \( n \to \infty \)

\( \frac{k_n}{n} \to 0 \) for \( n \to \infty \): otherwise volume cannot shrink to zero
Possible choice for $k_n$

\[ k_n = \sqrt{n} \]

and pick $V_n$ such that is include exactly $k_n$ samples
Expanding Number of Samples/Shrinking Volume in $k$-Nearest-Neighbour Estimation

$k_n = \sqrt{n}$

From: Duda+Hart: Pattern Classification
6.2. Parzen Windows
Introduction

• Each piece of training contributes its own bit of probability distribution

• Possible choice:
  • Cubes
  • Sphere
  • Normal distribution

For the beginning start with cubes
Volume of d-dimensional cubes

- Length of edge $h_n$

- Volume of cube

\[ V_n = h_n^d \]

$d$: dimension of features space

“Home work”: volume of a d-dimensional sphere?
Introduce Window Function

• **Goal:** generalize and formalize method

\[
\varphi(x) = \begin{cases} 
1 & \text{if } |x^j| \leq \frac{1}{2} \text{ for all } j = 1 \ldots d \\
0 & \text{else}
\end{cases}
\]

\[x^j : \text{j-th component of } x\]

Unit cube centered at origin

---

**Draw a unit cube**

for d=1 and d=2
Shift and scale the unit cube

What is the window function for a cube centered at $x_i$
with length of edge $h_n$???

$$\varphi\left(\frac{x - x_i}{h_n}\right)$$
Number of Samples at a point inside the volume $V_n$

Express number of data points $x_i$ that contribute in terms of window functions

$$\varphi\left(\frac{x - x_i}{h_n}\right)$$

If $\varphi\left(\frac{x - x_i}{h_n}\right) = 1$ then $x_i$ is in the volume $V_n$ and hence contributes to $k_n$

$$k_n(x) = \sum_{i=1}^{n} \varphi\left(\frac{x - x_i}{h_n}\right)$$
Estimate of probability using section 6.1

\[ P_n(x) = \frac{k_n(x)}{nV_n} \]

\[ \sum_{i=1}^{n} \varphi\left(\frac{x - x_i}{h_n}\right) = \frac{nV_n}{nV_n} \]

\[ P_n(x) = \frac{1}{nV_n} \sum_{i=1}^{n} \varphi\left(\frac{x - x_i}{h_n}\right) \]

This formula also works for other window functions
Other Window Functions

• Normal (Gaussian) distribution (covariance matrix is unit matrix)

\[ \varphi(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} \vec{x}^T \vec{x}} \]

• Sphere

\[ \varphi(x) = \begin{cases} 1 & \text{if } |x| \leq 1 \\ 0 & \text{else} \end{cases} \]
Gaussian Parzen Window 1 d

\[ h_1 = 1 \quad h_1 = 0.5 \quad h_1 = 0.1 \]

\[ n = 100 \]

\[ n = \infty \]

\[ h_n = h_1 / \sqrt{n} \]
Gaussian Parzen Window 2 d

$h_1=2$

$h_1=1$

$h_1=0.5$

$n=1$  

$n=10$
Gaussian Parzen Window 2 d

Converges for all $h_n$
Gaussian Parzen Window 1 d
In classifiers based on Parzen-window estimation:

- Estimate probability density using a given window
- Pick suitable $h_n$
- Classify using Bayes decision rule
Classification example

Small $h_n$

Large $h_n$
6.3. $k_n$-nearest Neighbor Estimation
Basic idea

Find k most similar cases to test sample $x$ and claim that $x$ is like majority of these cases.
Other names for similar/related methods

• Instance-Based Methods (IBM), or Instance Based Learning (IBL)
• Memory-Based Methods (MBM),
• Case-Based Methods (CBM),
• Case-Based Reasoning (CBR),
• Memory-Based Reasoning (MBR),
• Similarity-Based Reasoning (SBR),
• Similarity-Based Methods (SBM)
Estimate probability in nearest neighbor case

$x_1, x_2, x_3$: training data
$x$: point where we want probability $P(x)$

$V = 2 |x - x_2|$

$$P(x) = \frac{1}{2 |x - x_2|}$$
kNN-Estimation in 1 Dimension

From Duda+Hart: Pattern Classification
Estimating the Posterior $P(\omega_i | x)$

$V$ : volume under consideration

$k$ : total number of samples in $V$

$k_i$ : number of samples of class $\omega_i$ in $V$

$$k = \sum_{i=1}^{c} k_i$$

$$P_n(x, \omega_i) = \frac{k_i}{nV}$$

$$P_n(x) = \sum_{i=1}^{c} P_n(x, \omega_i) = \sum_{i=1}^{c} \frac{k_i}{nV} = \frac{k}{nV}$$

$$P_n(\omega_i | x) = \frac{P_n(x, \omega_i)}{P_n(x)} = \frac{\frac{k_i}{nV}}{\frac{k}{nV}} = \frac{k_i}{k}$$

$$P_n(\omega_i | x) = \frac{k_i}{k}$$
6.4. Nearest-Neighbor Rule
Voroni-Tessellation

• See white board
Voronoi Cells in 2 Dimensions

From Duda+Hart: Pattern Classification
Decision Boundary for a nearest-neighbour classifier in a Simulation
(Probability Distribution given)

From Hastie et al.: Statistical Learning
Voronoi Cells in 3 Dimensions

From Duda+Hart: Pattern Classification
6.5. Error of Nearest Neighbor Rule
Error of Nearest-Neighbour-Classifier (NN)

NN-Classifier:
• Can be as good as Bayes
• In worst case twice as bad

From: Duda+Hart: Pattern Classification
k-Nearest-Neighbour-Classifier

From Duda+Hart: Pattern Classification
Error of k-Nearest-Neighbour-Classifier

From Duda+Hart: Pattern Classification
Missclassification vs. Number of Neighbours

Classification Error Rate

$k$

From: Hastie et al.: Statistical Learning
Decision Boundary for a nearest-neighbour classifier in a Simulation
(Probability Distribution given)

From Hastie et al.: Statistical Learning
Decision Boundary for a $k$-nearest-neighbour classifier in a Simulation (Probability Distribution given)

From: Hastie et al.: Statistical Learning
Decision Boundaries of Bayes Classifier for the known Probabilities

From: Hastie et al.: Statistical Learning
Other Popular distance functions

$L_\alpha$ distance from 0:

$$D(\mathbf{X}, 0)^\alpha = \left( \sum_{i=1}^{d} |X_i|^\alpha \right)^{1/\alpha}$$

Manhattan distance or $L_1$ norm:

$$D(\mathbf{X}, \mathbf{Y}) = \sum_{i=1}^{d} |X_i - Y_i|$$

Euclidean distance or $L_2$ norm:

$$D(\mathbf{X}, \mathbf{Y})^2 = \sum_{i=1}^{d} (X_i - Y_i)^2$$

$\alpha = 1/2, 1, 2, \text{ and } 10$
Summary

• Parzen method
• k-Nearest Neighbour classifier