SOGANG UNIVERSITY

Probability & Bayesian Decision Theory

Jihoon Yang

Machine Learning Research Laboratory
Department of Computer Science & Engineering
Sogang University
Email: yangjh@sogang.ac.kr
URL: mllab.sogang.ac.kr

Learning as Bayesian inference

- Bayesian (subjective) probability provides a basis for updating beliefs based on evidence
- By updating beliefs about hypotheses based on data, we can learn about the world 데이터의 의미 원리를 캐내간다
- Formulate the learning task as a process of probabilistic inference
- *Inference step*: determine *P(x,t)* from data
- *Decision step*: for given *x*, determine optimal *t*
- Bayes Decision Theory: a fundamental statistical approach to the problem of pattern recognition and machine learning
- Bayesian framework provides a sound probabilistic basis for understanding many learning algorithms and designing new algorithms

A Classification Problem

- The sea bass/salmon example: a fish-packing plant wants to automate the process of sorting incoming fish on a conveyor belt according to species

- State of nature, *C* (the category of the fish)
  - *C* = {*c1, c2*} = {sea bass, salmon}
  - State of nature is a random variable

- We assume that there is some *a priori* probability P*(C), Prior*
  - The catch of salmon and sea bass is equally probable?
  - Prior probabilities reflect our prior knowledge

- Suppose we have to make a decision about the type of fish that will appear next without seeing it
- A seemingly logical decision rule with only the prior information:
  - Decide c1 if *P(c1) > P(c2);* otherwise decide *c2*
- The probability of error is the smaller of *P(c1)* and *P(c2)*

A Classification Problem

- Measure a feature value *x*, say length: continuous random variable
- Different fish will yield different length readings
  - *Class-conditional probability* density function *P(x|C)*
- *P(x|c1)* and *P(x|c2)* describes the difference in length between populations of sea bass and salmon
- Suppose we measure the length of a fish and discover that its value is *x*
- *A posteriori probability* (or *posterior*) *P(cj|x)*: the probability of the state of nature given that feature value *x* has been measured
- *P(cj|x) = P(x|cj)*P(cj)/P(x)* [Bayes' Rule]

Representing and Reasoning under Uncertainty

- Example of reasoning under uncertainty
  - Beliefs:
    - If Kim studies, there is 60% chance that he will pass the test; and 40% chance that he will not
    - If he does not study, there is 20% chance that he will pass the test; and 80% chance that he will not
  - Observation: Kim did not study
  - Inference task: What is the chance that he will pass the test? What is the chance that he will fail?

- Probability theory generalizes propositional logic
  - Probability theory associates probabilities that lie in the interval [0,1] as opposed to 0 or 1 (exclusively)

Foundations of probability theory

- An *atomic event /world state/sample point* is a *complete specification* of the state of the agent's world

- *Event set* (or *sample space*) is a set of mutually exclusive and exhaustive possible world states (relative to an agent's representational commitments and sensing abilities)

- *From the point of view of an agent Park* who can sense only 3 colors and 2 shapes, *the world can be in only one of 6 states*

- Properties of atomic events
  - They are mutually exclusive
  - The set of possible atomic events is exhaustive

Atomic events and Event sets

Examples

- **Event set**

  E1 = {(red, square), (green, square), (red, circle), (green, circle)}

  describes the event set (possible worlds) of an agent that can sense two kinds of shape to which it has given the names – *square, circle* – and two colors to which it has given the names – *red, green*

- **Event set *E2 = {(H, H), (H, T), (T, H), (T, T)}***

  is the set of possible outcomes of a sequence of two coin tosses

- **Event set *E3 = {H, T}* where *H* and *T* are atomic events corresponding to mutually exclusive outcomes of a coin toss**

Semantics:
Probability as a subjective measure of belief

- Suppose there are 3 agents – Kim, Lee, Park, in a world where a fair dice has been tossed. Kim observes that the outcome is a "6" and whispers Lee that the outcome is "even" but Park knows nothing about the outcome:

  – Set of possible mutually exclusive and exhaustive world states
  = {1, 2, 3, 4, 5, 6}

  – Set of possible states of the world based on what Lee knows
  = {2, 4, 6}

Probability as a subjective measure of belief

- **Probability is a measure over all of the world states that are possible, or simply, possible worlds, *given what the agent knows***

$$Possibleworlds_{Kim} = \{6\}$$
$$Possibleworlds_{Lee} = \{2,4,6\}$$
$$Possibleworlds_{Park} = \{1,2,3,4,5,6\}$$

$$P_{Kim}(worldstate = 6) = 1$$
$$P_{Lee}(6) = \frac{1}{3}$$
$$P_{Park}(6) = \frac{1}{6}$$

→ **Kim, Lee, and Park assign different beliefs to the same world state because of differences in their knowledge**

Random variables

- A *random variable* is a function from sample points to some range (e.g. the reals or Booleans)

- The "*domain*" (or *space*) of a random variable is the set of values it can take; The values are mutually exclusive and exhaustive
  (i.e. $0 \le P(x) \le 1, \quad \sum_{x \in domain(X)} P(x) = 1$)

- The domain of a Boolean random variable *X* is {true, false} or {1, 0}

- *Discrete random variables* take values from a *countable* domain
  - The domain of the random variable Color may be {Red, Green}
  - If *E* = {(Red, Square), (Green, Circle), (Red, Circle), (Green, Square)}, the proposition (Color = Red) is True in the world states {(Red, Square), (Red, Circle)}
  - Each state of a discrete random variable corresponds to a proposition (e.g. (Color = Red))

Syntax

- **Basic element: random variable**
  - **Similar to propositional logic: possible worlds defined by assignment of values to random variables**
  - ***Cavity* (do I have a cavity?)**
  - ***Weather* is one of <*sunny*, *rainy*, *cloudy*, *snow*>**
  - **Values must be *exhaustive* and *mutually exclusive***

- **Elementary proposition constructed by assignment of a value to a random variable**
  - ***Weather* = *sunny*, *Cavity* = *false* (abbreviated as ¬*cavity*)**

- **Complex propositions formed from elementary propositions and standard logical connectives**
  - ***Weather* = *sunny V Cavity* = *false***

Syntax and Semantics

- **Atomic event: A *complete* specification of the state of the world about which the agent is uncertain**

- **Atomic events correspond to possible worlds (much like in the case of propositional logic)**
  - **The sample space is the (Cartesian product of the) domain of the variables**
  - **Atomic events are mutually exclusive and exhaustive**

(e.g.) If the world consists of only two Boolean variables *Cavity* and *Toothache*, then there are 4 distinct atomic events:

Cavity* = *false* Λ *Toothache* = *false

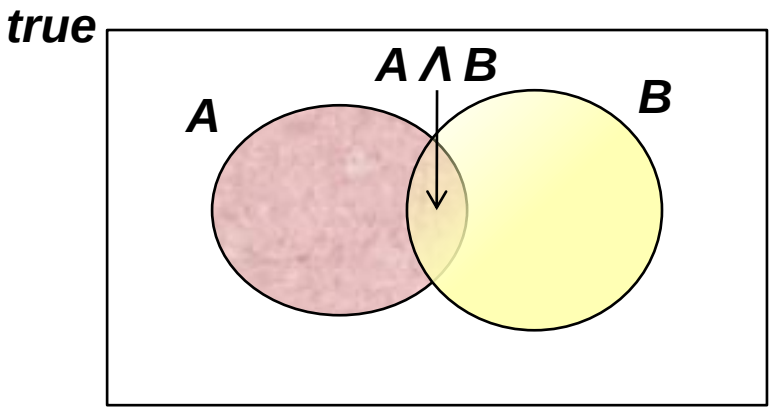
Cavity* = *false* Λ *Toothache* = *true

Cavity* = *true* Λ *Toothache* = *false

Cavity* = *true* Λ *Toothache* = *true

Axioms of probability

- **For any propositions *A*, *B***
  - ***0 ≤ P(A) ≤ 1***
  - ***P(true) = 1 and P(false) = 0***
  - ***P(A V B) = P(A) + P(B) – P(A Λ B)***



Prior probability

- **Prior** or **unconditional probabilities** of propositions
  - *P*(*Cavity* = true) = 0.2 and *P*(*Weather* = sunny) = 0.72 correspond to belief prior to arrival of any (new) evidence

- **Probability distribution** gives values for all possible assignments:
  - P(*Weather*) = <0.72, 0.1, 0.08, 0.1>
  - Note that the probabilities sum to 1

- **Joint probability distribution** for a set of random variables gives the probability of every atomic event on those random variables
  - P(*Weather*, *Cavity*) = a 4 x 2 matrix of values

Prior probability

- **Joint probability distribution for a set of random variables gives the probability of every atomic event on those random variables**

P(*Weather*, *Cavity*) = a 4 x 2 matrix of values:

Weather =	sunny	rainy	cloudy	snow
Cavity = true	0.144	0.02	0.016	0.02
Cavity = false	0.576	0.08	0.064	0.08

- ***Every question about a domain can be answered by the joint distribution***

Inference using the joint distribution

	Toothache = true	*Toothache* = false
***Cavity* = true**	0. 4	0.1
***Cavity* = false**	0.1	0.4

P(cavity) = P(cavity, ache) + P(cavity, ¬ache)

Conditional probability

- **Conditional or posterior probabilities**

  ***P*(*cavity* | *toothache*) = 0.8**

  → probability of *cavity* given that *toothache*

  (note *cavity* is shorthand for *Cavity* = true)

- **Notation for conditional distributions:**

  P(*Cavity* | *Toothache*) = 2-element vector of 2-element vectors

  ***P*(*Cavity* | *Toothache, Cavity*) = 1**

- **New evidence may be irrelevant (probability of *cavity* given *toothache* is independent of *Weather*)**

  ***P*(*cavity* | *toothache, sunny*) = *P*(*cavity* | *toothache*) = 0.8**

Conditional probability

- **Definition of conditional probability:**

  $P(a \mid b) = P(a \wedge b) / P(b)$ **if** $P(b) > 0$

- **Product rule gives an alternative formulation:**

  $P(a \wedge b) = P(a \mid b)\, P(b) = P(b \mid a)\, P(a)$

- **Example: Suppose I have two coins – one a normal fair coin, and the other with 2 heads. I pick a coin at *random* and tell you that the side I am looking at is a head. What is the probability that I am looking at a normal coin?**

Conditional probability (Example cont'd)

- **Label the sides *h*, *h'* so that the side labeled *h* corresponds to a head on both coins, and the side labeled *h'* corresponds to a tail on the normal coin and a head on the 2-head coin**

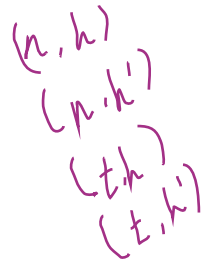
- ***n*, *t* – normal versus 2-sided**
- ***E* = *{n, t} X {h, h'}***

- **Compound events *N*, *H*:**

  ***N* = *{(n, h), (n, h')}* (selecting the normal coin)**

  ***H* = *{(n, h), (t, h), (t, h')}* (selecting a head)**

$$P(N \mid H) = \frac{P(N \wedge H)}{P(H)} = \frac{1/4}{3/4} = \frac{1}{3}$$



Conditional probability

- **A general version holds for whole distributions**
  ***e.g.* P(*Weather*, *Cavity*) = P(*Weather* | *Cavity*) P(*Cavity*)**
- **View as a compact notation for a set of 4 x 2 equations, *not* matrix multiplication**

- **Chain rule is derived by successive application of product rule:**

$$\begin{aligned}
\mathrm{P}(X_1,\ldots,X_n) &= \mathrm{P}(X_1,\ldots,X_{n-1})\,\mathrm{P}(X_n \mid X_1,\ldots,X_{n-1}) \\
&= \mathrm{P}(X_1,\ldots,X_{n-2})\,\mathrm{P}(X_{n-1} \mid X_1,\ldots,X_{n-2})\,\mathrm{P}(X_n \mid X_1,\ldots,X_{n-1}) \\
&= \ldots \\
&= \Pi_i \mathrm{P}(X_i \mid X_1,\ldots,X_{i-1}) \ (i \text{ ranges from } 1 \text{ to } n)
\end{aligned}$$

Inference by enumeration

- **Start with the joint probability distribution:**

	toothache		*¬toothache*	
	catch	***¬catch***	***catch***	***¬catch***
cavity	.108	.012	.072	.008
¬cavity	.016	.064	.144	.576

- **For any proposition φ, sum the atomic events where it is true:**

$$P(\varphi) = \sum_{\omega:\omega|=\varphi} P(\omega) \quad (\omega \text{ is a possible world})$$

- ***P*(*toothache*) = 0.108 + 0.012 + 0.016 + 0.064 = 0.2**

Probabilistic Inference

- One common task is to extract the distribution over a single variable or some subset of variables, called marginal distribution

  P(*toothache*) = 0.108 + 0.012 + 0.016 + 0.064 = 0.2

  P(¬*toothache*) = … = 0.8

- This process is called marginalization or summing out: for any sets of variables *Y* and *Z*

$$P(Y) = \sum_z P(Y, z) = \sum_z P(Y \mid z) P(z)$$

- A distribution over *Y* can be obtained by summing out all other variables from any joint distribution containing *Y*

Inference by enumeration

- **Start with the joint probability distribution:**

	toothache		*¬toothache*	
	catch	*¬catch*	*catch*	*¬catch*
cavity	.108	.012	.072	.008
¬cavity	.016	.064	.144	.576

- **Can also compute conditional probabilities:**

$$P(\neg cavity \mid toothache) = P(\neg cavity \wedge toothache) / P(toothache)$$
$$= (0.016 + 0.064) / (0.108 + 0.012 + 0.016 + 0.064)$$
$$= 0.4$$

Normalization

	toothache		*¬toothache*	
	catch	***¬catch***	***catch***	***¬catch***
cavity	.108	.012	.072	.008
¬cavity	.016	.064	.144	.576

- **Denominator can be viewed as a normalization constant α**

  P(*Cavity* | *toothache*) = αP(*Cavity*, *toothache*)

  = α[P(*Cavity*, *toothache*, *catch*) + P(*Cavity*, *toothache*, *¬catch*)]

  = α[<0.108, 0.016> + <0.012, 0.064>]

  = α<0.12, 0.08> = <0.6, 0.4>

- **General idea: compute distribution on query variable by fixing *evidence variables* and summing over *unobserved variables***

Probabilistic Inference

- Let *X* be all variables. Typically we want the posterior distribution of the query variables *Y* given specific values e for the evidence variables *E*

- Let other variables be *H = X – Y – E*

- Then the required summation of joint entries is done by summing out the other variables:

$$\mathbf{P}(\mathbf{Y} \mid \mathbf{E} = \mathbf{e}) = \sum_{\mathbf{h}} \mathbf{P}(\mathbf{Y}, \mathbf{H} = \mathbf{h} \mid \mathbf{E} = \mathbf{e}) = \alpha \sum_{\mathbf{h}} \mathbf{P}(\mathbf{Y}, \mathbf{H} = \mathbf{h}, \mathbf{E} = \mathbf{e})$$

Probabilistic Inference

- **In principle, joint distributions can be used to answer any probabilistic queries**

- **Obvious problems:**
  - **Worst-case time complexity $O(d^n)$ where *d* is the largest arity**
  - **Space complexity $O(d^n)$ to store the joint distribution**
  - **How to find the numbers for $O(d^n)$ entries??**

Expected Values

- **Expectation of *r.v. X* (expected value, mean, average)**

$$E[X] = \sum_x xP(x)$$

- **Expectation of *function f(x)***

$$E[f(x)] = \sum_x f(x)P(x)$$

$$E[f(x)] = \int_x f(x)P(x)dx$$

- **Variance (and standard deviation) provides a measure of variability around mean value**

$$\text{var}[X] = E[(X - E[X])^2] = E[X^2] - E[X]^2$$

- **Covariance: a measure of statistical dependence between *X* and *Y***

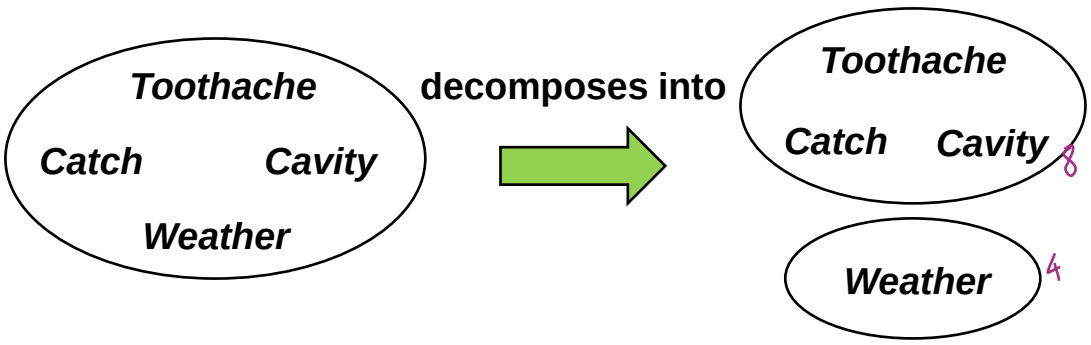
$$\text{cov}[X, Y] = E[(X - E[X])(Y - E[Y])]$$

$$= \sum_{x,y} (x - E[X])(y - E[Y])P(x, y)$$

Independence

- ***A*** **and** ***B*** **are independent iff**

  P(*A*|*B*) = P(*A*) or P(*B*|*A*) = P(*B*) or P(*A*, *B*) = P(*A*)P(*B*)



- **P(*Toothache, Catch, Cavity, Weather*)**

  = P(*Toothache, Catch, Cavity*) P(*Weather*)

- **32 entries reduced to 12; for *n* independent biased coins,** $O(2^n) \rightarrow O(n)$

- **Absolute independence powerful but rare**
- **How can we manage a large numbers of variables?**

Conditional independence

- P(*Toothache, Catch, Cavity*) has 2^3 - 1 = 7 independent entries

- If I have a cavity, the probability that the probe catches in it doesn't depend on whether I have a toothache:

  P(catch | toothache, cavity) = P(catch | cavity)

- The same independence holds if I haven't got a cavity:

  P(catch | toothache, ¬ cavity) = P(catch | ¬ cavity)

- *Catch* is conditionally independent of *Toothache* given *Cavity*:

  P*(Catch | Toothache, Cavity)* = P*(Catch | Cavity)*

Conditional independence

- ***Catch*** **is conditionally independent of** ***Toothache*** **given** ***Cavity*****:**

  P*(Catch | Toothache, Cavity)* = P*(Catch | Cavity)*

- **Equivalent statements:**

  P*(Toothache | Catch, Cavity)* = P*(Toothache | Cavity)*

  P*(Toothache, Catch | Cavity)*

  = P*(Toothache | Cavity)* P*(Catch | Cavity)*

Conditional independence

- **Write out full joint distribution using chain rule:**

  P(*Toothache, Catch, Cavity*)

  = P*(Toothache | Catch, Cavity)* P*(Catch, Cavity)*

  = P*(Toothache | Catch, Cavity)* P*(Catch | Cavity)* P*(Cavity)*

  = P*(Toothache | Cavity)* P*(Catch | Cavity)* P*(Cavity)*

  i.e. 2 + 2 + 1 = 5 independent numbers

- **Conditional independence**
  - **Often reduces the size of the representation of the joint distribution from exponential in *n* to linear in *n***
  - **Is one of the most basic and robust form of knowledge about uncertain environments**

Conditional Independence

- ***X* is *conditionally independent* of *Y* given *Z* if the probability distribution governing *X* is independent of the value of *Y* given the value of *Z*:**

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

that is, if

$$(\forall x_i, y_j, z_k) P(X = x_i \mid Y = y_j, Z = z_k) = P(X = x_i \mid Z = z_k)$$

Another example

- ***Thunder*** **is conditionally independent of** ***Rain*** **given** ***Lightening***

$$P(Thunder=1|Rain=1, Lightening=1) = P(Thunder=1|Lightening=1) = P(Thunder=1|Rain=0, Lightening=1)$$

$$P(Thunder=1|Rain=1, Lightening=0) = P(Thunder=1|Lightening=0) = P(Thunder=1|Rain=0, Lightening=0)$$

$$P(Thunder=0|Rain=1, Lightening=1) = P(Thunder=0|Lightening=1) = P(Thunder=0|Rain=0, Lightening=1)$$

$$P(Thunder=0|Rain=1, Lightening=0) = P(Thunder=0|Lightening=0) = P(Thunder=0|Rain=0, Lightening=0)$$

Independence and Conditional Independence

- **Let $Z_1 ... Z_n$ and *W* be pairwise disjoint sets of random variables on a given event space**

  $Z_1 ... Z_n$ are mutually independent given *W* if

$$\mathrm{P}(Z_1 \cup ... \cup Z_n \mid W) = \prod_{i=1}^{n} \mathrm{P}(Z_i \mid W)$$

  $\mathrm{P}(Z_1 \mid Z_2 \cup W) = \mathrm{P}(Z_1 \mid W)$ if *Z1* and *Z2* are independent

- **Note that these represent sets of equations, for all possible value assignments to random variables**

Independence properties of random variables

- **Let *W, X, Y, Z* be pairwise disjoint sets of random variables on a given event space**

  Let *I(X, Y, Z)* denote that *X* and *Z* are *independent* given *Y*

  That is, $\mathrm{P}(X \cup Z \mid Y) = \mathrm{P}(X \mid Y)\mathrm{P}(Z \mid Y), \text{ or } \mathrm{P}(X \mid Y \cup Z) = \mathrm{P}(X \mid Y)$

  then

  a. $I(X,Z,Y) \Rightarrow I(Y,Z,X)$

  b. $I(X,Z,Y \cup W) \Rightarrow I(X,Z,Y)$

  c. $I(X,Z,Y \cup W) \Rightarrow I(X,Z \cup W,Y)$

  d. $I(X,Z,Y) \wedge I(X,Z \cup Y,W) \Rightarrow I(X,Z,Y \cup W)$

  (Proof: Follows from definition of independence)

Bayes Rule

- **Does patient have cancer or not?**

  A patient takes a lab test and the result comes back positive. The test returns a correct positive result in only 98% of the cases in which the disease is actually present, and a correct negative result in only 97% of the cases in which the disease is not present. Furthermore, .008 of the entire population have this cancer.

$$P(cancer) = \qquad P(\neg cancer) =$$

$$P(+ \mid cancer) = \qquad P(- \mid cancer) =$$

$$P(+ \mid \neg cancer) = \qquad P(- \mid \neg cancer) =$$

Bayes Rule

- **Does patient have cancer or not?**

$$P(cancer) = 0.008 \qquad P(\neg cancer) = 0.992$$
$$P(+ \mid cancer) = 0.98 \qquad P(- \mid cancer) = 0.02$$
$$P(+ \mid \neg cancer) = 0.03 \qquad P(- \mid \neg cancer) = 0.97$$

$$P(cancer|+) = \frac{P(+|cancer)P(cancer)}{P(+)}; \qquad P(\neg cancer|+) = \frac{P(+|\neg cancer)P(\neg cancer)}{P(+)}$$

$$P(cancer|+)P(+) = 0.98 \times 0.008 = 0.0078; \quad P(\neg cancer|+)P(+) = 0.03 \times 0.992 = 0.0298$$

$$P(+) = 0.0078 + 0.0298$$

$$P(cancer \mid +) = 0.21; \qquad P(\neg cancer \mid +) = 0.79$$

The patient, more likely than not, does not have cancer

3/16.

Bayes Rule

- **Product rule**

  $P(a \wedge b) = P(a \mid b)\, P(b) = P(b \mid a)\, P(a)$

  Bayes' rule: $P(a \mid b) = P(b \mid a)\, P(a) / P(b)$

- **In distribution form:**

  $\mathbf{P}(Y \mid X) = \mathbf{P}(X \mid Y)\, \mathbf{P}(Y) / \mathbf{P}(X) = \alpha\, \mathbf{P}(X \mid Y)\, \mathbf{P}(Y)$

Bayes' rule and conditional independence

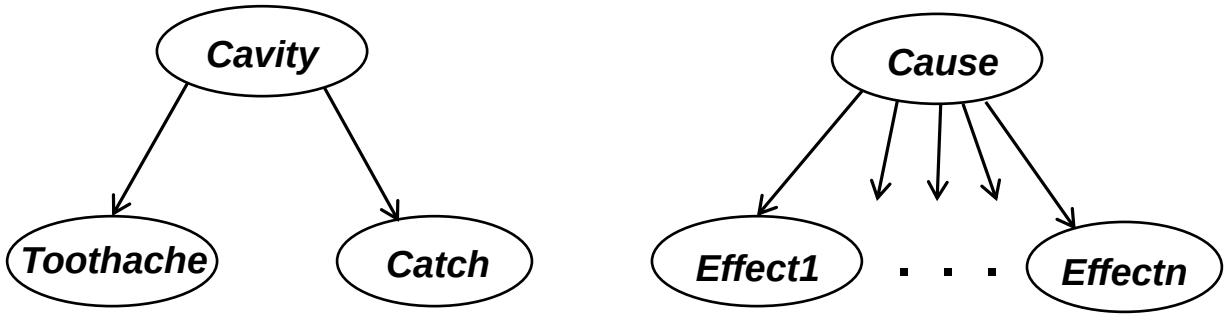
P(*Cavity* | *toothache* Λ *catch*)

= α P*(toothache* Λ *catch* |*Cavity)* P*(Cavity)*

= α P*(toothache* | *Cavity)* P*(catch* | *Cavity)* P*(Cavity)*

- **This is an example of a naïve Bayes (idiot Bayes) model:**

$$\mathrm{P}(Cause, Effect_1, \ldots, Effect_n) = \mathrm{P}(Cause)\Pi_i \mathrm{P}(Effect_i \mid Cause)$$



- **Total number of parameter is linear in *n***

Summary of basic Probability Theory

- Probability is a rigorous formalism for uncertain knowledge
- Joint probability distribution specifies probability of every atomic event
- Queries can be answered by summing over atomic events
- For nontrivial domains, we must find a way to reduce the joint size
- Independence and conditional independence provide the tools

Classification using Bayesian decision theory

- **Consider the problem of classifying an instance *X* into one of two mutually exclusive classes c_1 or c_2**

  $P(c_1|X)$ = probability of class c_1 given the evidence *X*

  $P(c_2|X)$ = probability of class c_2 given the evidence *X*

What is the probability of error?

$P(error|X) = P(c_1|X)$ if we choose c_2 as the classification for *X*

$= P(c_2|X)$ if we choose c_1 as the classification for *X*

Minimum error classification

- **To minimize classification error**

Choose c_1 if $P(c_1|X) > P(c_2|X)$
Choose c_2 if $P(c_2|X) > P(c_1|X)$

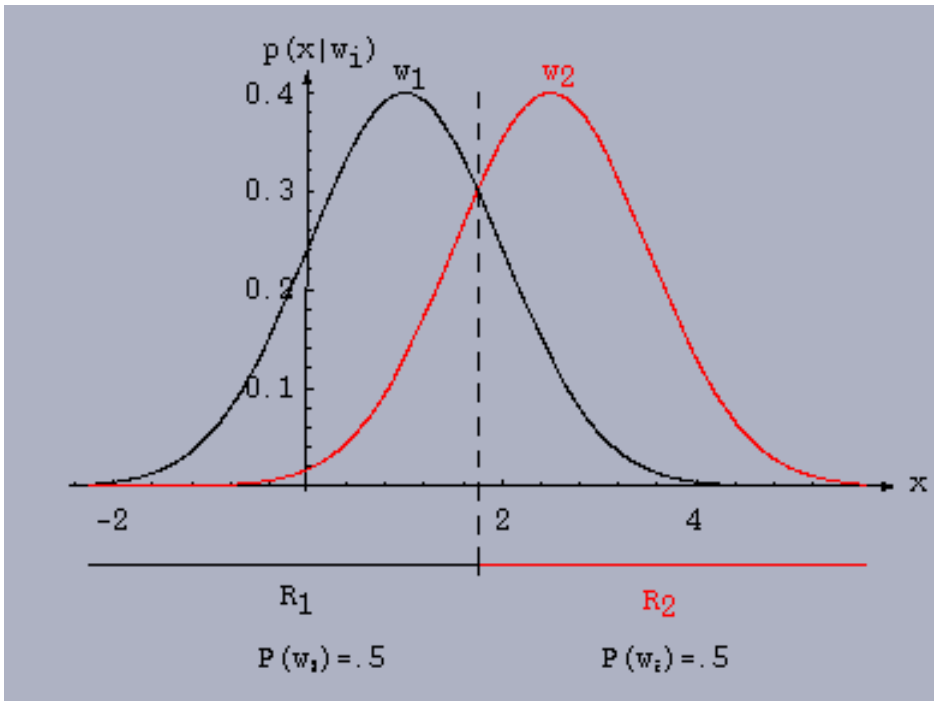
which yields

$$P(error|X) = \min[P(c_1|X), P(c_2|X)]$$

We have

$$P(c_1|X) = P(X|c_1)P(c_1)$$
$$P(c_2|X) = P(X|c_2)P(c_2)$$

Classification using Bayesian decision theory



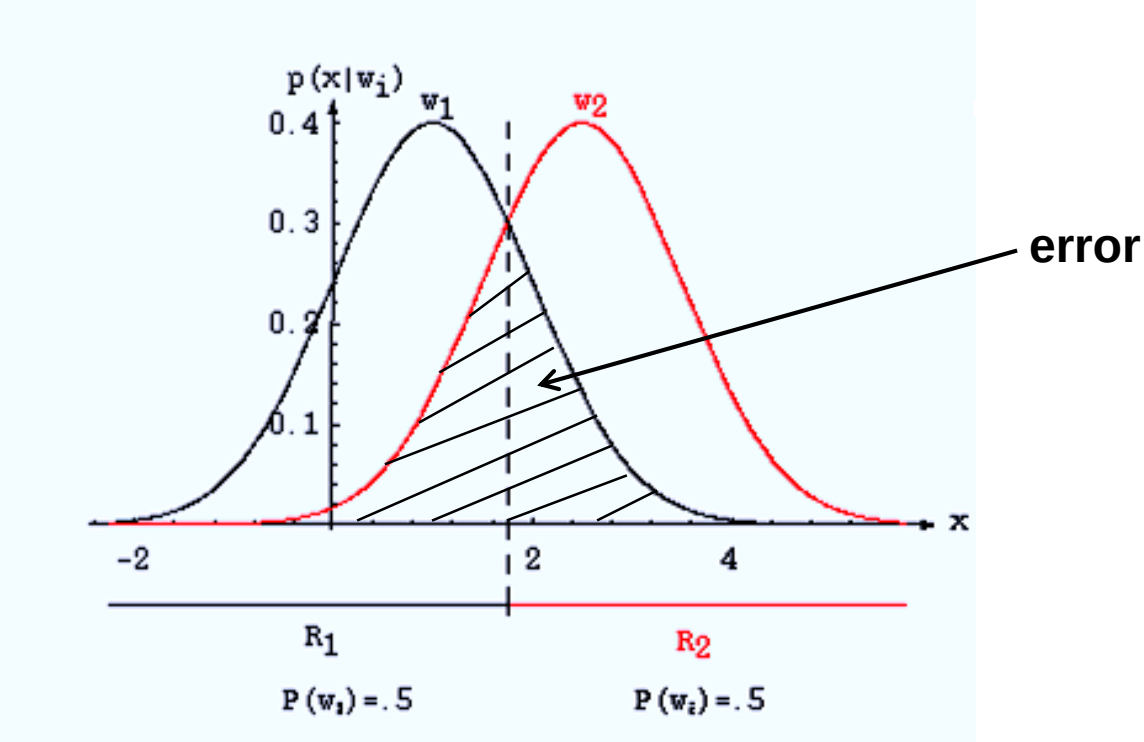
Note that:
- ***ci*** **is represented by** ***wi***
- ***Ri*** **is the region where we decide** ***ci***

Choose ***c1*** **if** ***P(c1|X) > P(c2|X)*** **i.e.** $X \in R_1$

Choose ***c2*** **if** ***P(c2|X) > P(c1|X)*** **i.e.** $X \in R_2$

Optimality of Bayesian decision rule

- **We can show that the Bayesian classifier is optimal in that it is guaranteed to minimize the probability of misclassification**



Optimality of Bayesian decision rule

$$
\begin{aligned}
P_e &= P(x \in R_2, x \in c_1) + P(x \in R_1, x \in c_2) \\
&= P(x \in R_2 \mid c_1)P(c_1) + P(x \in R_1 \mid c_2)P(c_2) \\
&= P(c_1)\int_{R_2} P(x \mid c_1)dx + P(c_2)\int_{R_1} P(x \mid c_2)dx
\end{aligned}
$$

Applying Bayes rule:

$$P(x \mid c_i)P(c_i) = P(c_i \mid x)P(x) = P(x, c_i)$$

$$P_e = \int_{R_2} P(c_1 \mid x)P(x)dx + \int_{R_1} P(c_2 \mid x)P(x)dx$$

Optimality of Bayesian decision rule

$$P_e = \int_{R_2} P(c_1 \mid x)P(x)dx + \int_{R_1} P(c_2 \mid x)P(x)dx$$

Because $R_1 \cup R_2$ covers the entire input space,

$$\int_{R_1} P(c_1 \mid x)P(x)dx + \int_{R_2} P(c_1 \mid x)P(x)dx = P(c_1)$$

$$P_e = P(c_1) - \int_{R_1} \big(P(c_1 \mid x) - P(c_2 \mid x)\big)P(x)dx$$

P_e is minimized by choosing

R_1 such that $P(c_1 \mid x) > P(c_2 \mid x)$ and

R_2 such that $P(c_2 \mid x) > P(c_1 \mid x)$

Bayes decision rule yields minimum error classification

- **The proof generalizes to *multivariate* input spaces**

- **Similar results can be proved in the case of discrete (as opposed to continuous) input spaces – replace integral over the input space by sum**

- **To minimize classification error**

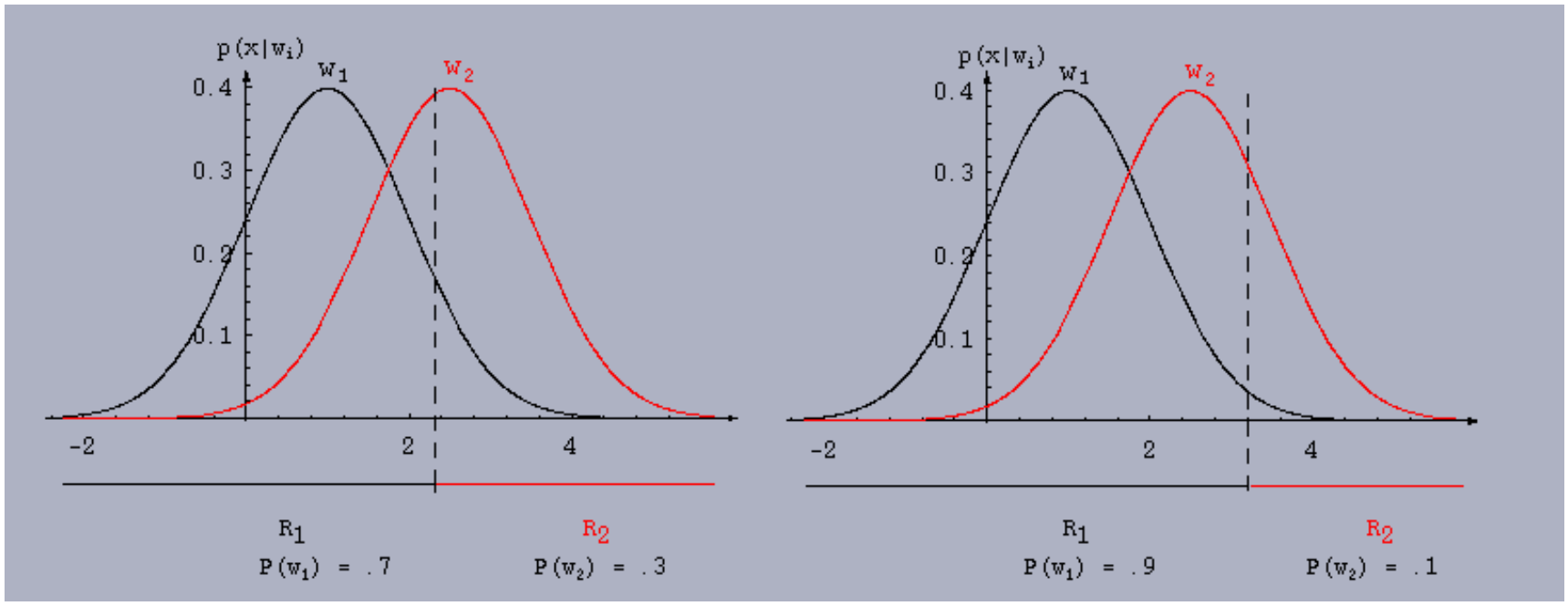
  Choose c_1 if $P(c_1|X) > P(c_2|X)$
  Choose c_2 if $P(c_2|X) > P(c_1|X)$

  which yields

  $$P(error|X) = \min[P(c_1|X), P(c_2|X)]$$

Bayesian decision rule

- **Behavior of Bayes decision rule as a function of prior probability of classes**



Bayes optimal classifier

- **Classification rule that guarantees minimum error:**
  Choose c_1 if $P(X|c_1)P(c_1) > P(X|c_2)P(c_2)$
  Choose c_2 if $P(X|c_2)P(c_2) > P(X|c_1)P(c_1)$

if $P(X|c_1) = P(X|c_2)$
classification depends entirely on $P(c_1)$ and $P(c_2)$
if $P(c_1) = P(c_2)$
classification depends entirely on $P(X|c_1)$ and $P(X|c_2)$

Bayes classification rule combines the effect of the two terms optimally – so as to yield minimum error classification

Generalization to *multiple classes*

$$c(X) = \arg\max_{c_j} P(c_j \mid X)$$

Minimum risk classification

- Let λ_{ij} = risk or cost associated with assigning an instance to class c_j when the correct classification is c_i

$R(c_i|X)$ = expected loss incurred in assigning X to class c_i

$$R(c_1 \mid X) = \lambda_{11} P(c_1 \mid X) + \lambda_{21} P(c_2 \mid X)$$
$$R(c_2 \mid X) = \lambda_{12} P(c_1 \mid X) + \lambda_{22} P(c_2 \mid X)$$

Classification rule that guarantees minimum risk:

Choose c_1 if $R(c_1|X) < R(c_2|X)$

Choose c_2 if $R(c_2|X) < R(c_1|X)$

Flip a coin otherwise

Minimum risk classification

- **Introduce a *loss function* which is more general than the probability of error**

- **λ_{ij} = risk or cost associated with assigning an instance to class c_j when the correct classification is c_i**

  Ordinarily $\lambda_{21} - \lambda_{22}$ and $\lambda_{12} - \lambda_{11}$ are positive (cost of being correct is less then the cost of error)

  So we choose c_1 if $$\frac{P(X \mid c_1)}{P(X \mid c_2)} > \frac{(\lambda_{21} - \lambda_{22})P(c_2)}{(\lambda_{12} - \lambda_{11})P(c_1)}$$

  Otherwise choose c_2

- **Minimum error classification rule is a special case:**
$$\lambda_{ij} = 0 \text{ if } i = j \text{ and } \lambda_{ij} = 1 \text{ if } i \neq j$$

- ***This classification rule can be shown to be optimal in that it is guaranteed to minimize the risk of misclassification***

Minimum risk classification

- λ_{ij} = risk or cost associated with assigning an instance to class c_j when the correct classification is c_i

Choose c_1 if

$$\frac{P(X \mid c_1)}{P(X \mid c_2)} > \frac{(\lambda_{21} - \lambda_{22})P(c_2)}{(\lambda_{12} - \lambda_{11})P(c_1)}$$

Choose c_2 if

$$\frac{P(X \mid c_1)}{P(X \mid c_2)} < \frac{(\lambda_{21} - \lambda_{22})P(c_2)}{(\lambda_{12} - \lambda_{11})P(c_1)}$$

- Minimum error classification rule is a special case:

Choose c_1 if $\frac{P(X \mid c_1)}{P(X \mid c_2)} > \frac{P(c_2)}{P(c_1)}$ Otherwise choose c_2

Discriminant Functions

- One way to represent pattern classifiers is in terms of discriminant functions

- The multi-category case

  - Set of discriminant functions $g_i(x)$, $i = 1, \ldots, c$

  - The classifier assigns a feature vector x to class ω_i if:

$$g_i(x) > g_j(x) \quad \forall j \neq i$$

- For the minimum error rate, we take

$$g_i(x) = P(\omega_i \mid x)$$

(max. discrimination corresponds to max. posterior!)

$$g_i(x) = p(x \mid \omega_i)\, P(\omega_i)$$

$$g_i(x) = \ln p(x \mid \omega_i) + \ln P(\omega_i)$$

Discriminant Functions

- The two-category case
  - Instead of using two discriminant functions g_1 and g_2

    Let $g(x) = g_1(x) - g_2(x)$

    Decide ω_1 *if* $g(x) > 0$; Otherwise decide ω_2

  - Minimum-error-rate discriminant

$$g(x) = P(\omega_1 \mid x) - P(\omega_2 \mid x)$$

$$g(x) = \ln \frac{p(x \mid \omega_1)}{p(x \mid \omega_2)} + \ln \frac{P(\omega_1)}{P(\omega_2)}$$

Discriminant Functions

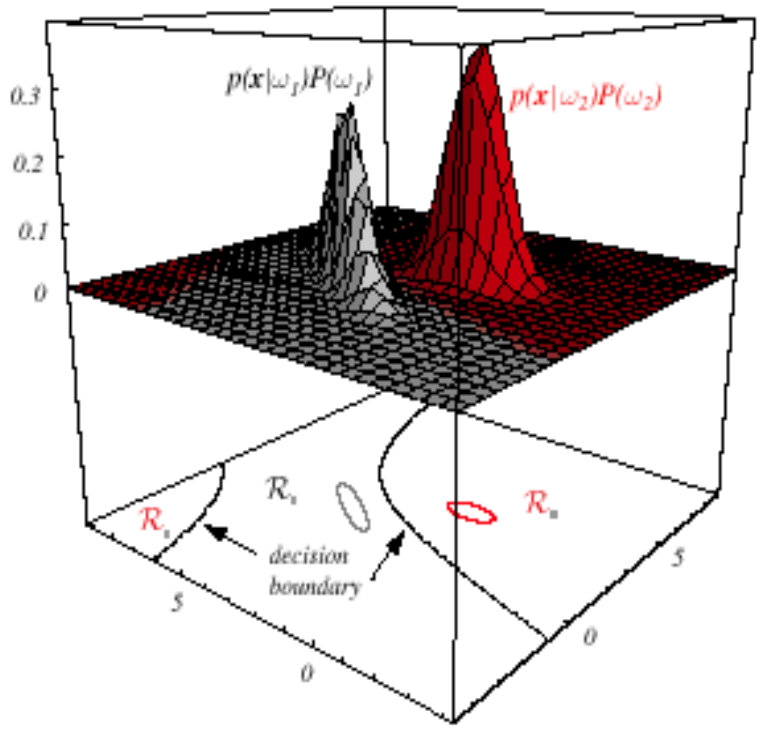


FIGURE 2.6. In this two-dimensional two-category classifier, the probability densities are Gaussian, the decision boundary consists of two hyperbolas, and thus the decision region $\mathcal{R}_2$ is not simply connected. The ellipses mark where the density is $1/e$ times that at the peak of the distribution. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Discriminant Functions for Normal Density

- Univariate normal (Gaussian) density

$$p(x) = N(\mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right],$$

Where:

μ = mean (or expected value) of x

σ^2 = variance, σ the standard deviation

- The expected value (mean, average) of x

$$\mu = E[x] = \int_{-\infty}^{\infty} x p(x) dx$$

- The variance

$$\text{Var}[x] = \sigma^2 = E[(x-\mu)^2] = \int_{-\infty}^{\infty} (x-\mu)^2 p(x) dx$$

Discriminant Functions for Normal Density

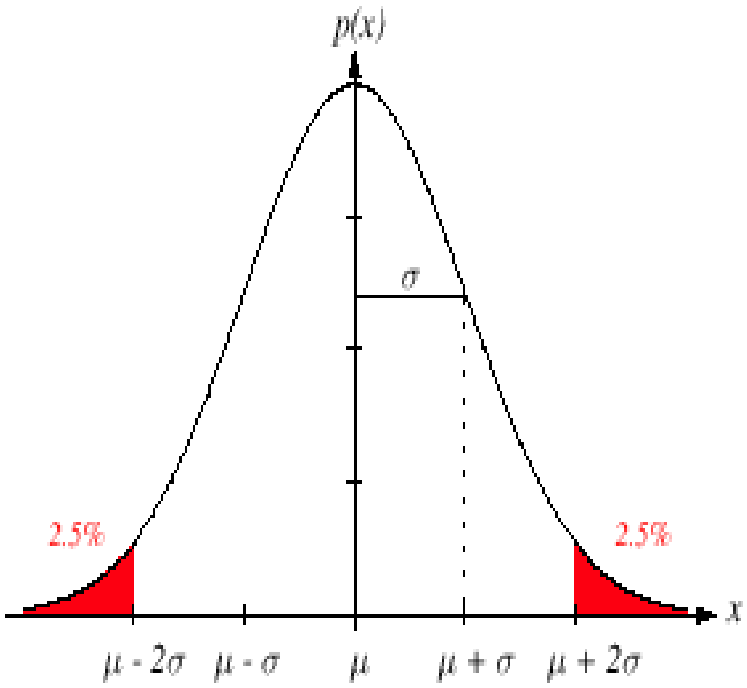


FIGURE 2.7. A univariate normal distribution has roughly 95% of its area in the range $|x - \mu| \leq 2\sigma$, as shown. The peak of the distribution has value $p(\mu) = 1/\sqrt{2\pi}\sigma$. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Discriminant Functions for Normal Density

- Multivariate normal density $p(x) \sim N(\mu, \Sigma)$

$$p(x) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left[-\frac{1}{2}(x-\mu)^t \Sigma^{-1}(x-\mu)\right]$$

where:

$x = (x_1, x_2, \ldots, x_d)^t$ (t stands for the transpose vector form)

$\mu = (\mu_1, \mu_2, \ldots, \mu_d)^t$ mean vector

Σ = d*d covariance matrix

$|\Sigma|$ and Σ^{-1} are determinant and inverse respectively

- The covariance of x_i and x_j

$$\sigma_{ij} = E[(x_i - \mu_i)(x_j - \mu_j)] = \int_{-\infty}^{\infty}(x_i - \mu_i)(x_j - \mu_j)p(x)dx$$

- If x_i and x_j are independent, then $\sigma_{ij} = 0$

Discriminant Functions for Normal Density

- Generative model: multivariate normal $p(x \mid \omega_i) \sim N(\mu_i, \Sigma_i)$
- The minimum error-rate classification can be achieved by the discriminant function

$g_i(x) = \ln p(x \mid \omega_i) + \ln P(\omega_i)$

$$g_i(x) = -\frac{1}{2}(x-\mu_i)^t \Sigma_i^{-1}(x-\mu_i) - \frac{d}{2}\ln 2\pi - \frac{1}{2}\ln|\Sigma_i| + \ln P(\omega_i)$$

Discriminant Functions for Normal Density

[1] Case $\Sigma_i = \sigma^2 I$ (I stands for the identity matrix)

$$g_i(x) = -\frac{\|x-\mu_i\|^2}{2\sigma^2} + \ln P(\omega_i) + \text{indept of } i$$

$$= -\frac{(x-\mu_i)^t(x-\mu_i)}{2\sigma^2} + \ln P(\omega_i)$$

$$= -\frac{1}{2\sigma^2}[x^t x - 2\mu_i^t x + \mu_i^t \mu_i] + \ln P(\omega_i)$$

- **If *x* is equally near two different mean vectors, the optimal decision will favor the prior more likely category (because of $\ln P(\omega_i)$ term)**

$g_i(x) = w_i^t x + w_{i0}$ (linear discriminant function)

where :

$$w_i = \frac{\mu_i}{\sigma^2};\ w_{i0} = -\frac{1}{2\sigma^2}\mu_i^t \mu_i + \ln P(\omega_i)$$

(ω_{i0} is called the threshold or bias for the ***i***th category)

Discriminant Functions for Normal Density

- **A classifier that uses linear discriminant functions is called "*a linear machine*"**

- **The decision surfaces for a linear machine are pieces of *hyperplanes* defined by:**

$$g_i(x) = g_j(x)$$

$$\Leftrightarrow w^t (x - x_0) = 0$$

$$\text{where } w = \mu_i - \mu_j$$

$$\text{and } x_0$$

Discriminant Functions for Normal Density

- **The hyperplane separating $\mathcal{R}_i$ and $\mathcal{R}_j$**

$$x_0 = \frac{1}{2}(\mu_i + \mu_j) - \frac{\sigma^2}{\|\mu_i - \mu_j\|^2} \ln \frac{P(\omega_j)}{P(\omega_j)} (\mu_i - \mu_j)$$

always orthogonal to the line linking the means!

$$if \quad P(\omega_i) = P(\omega_j) \quad \text{then} \quad x_0 = \frac{1}{2}(\mu_i + \mu_j)$$

Discriminant Functions for Normal Density

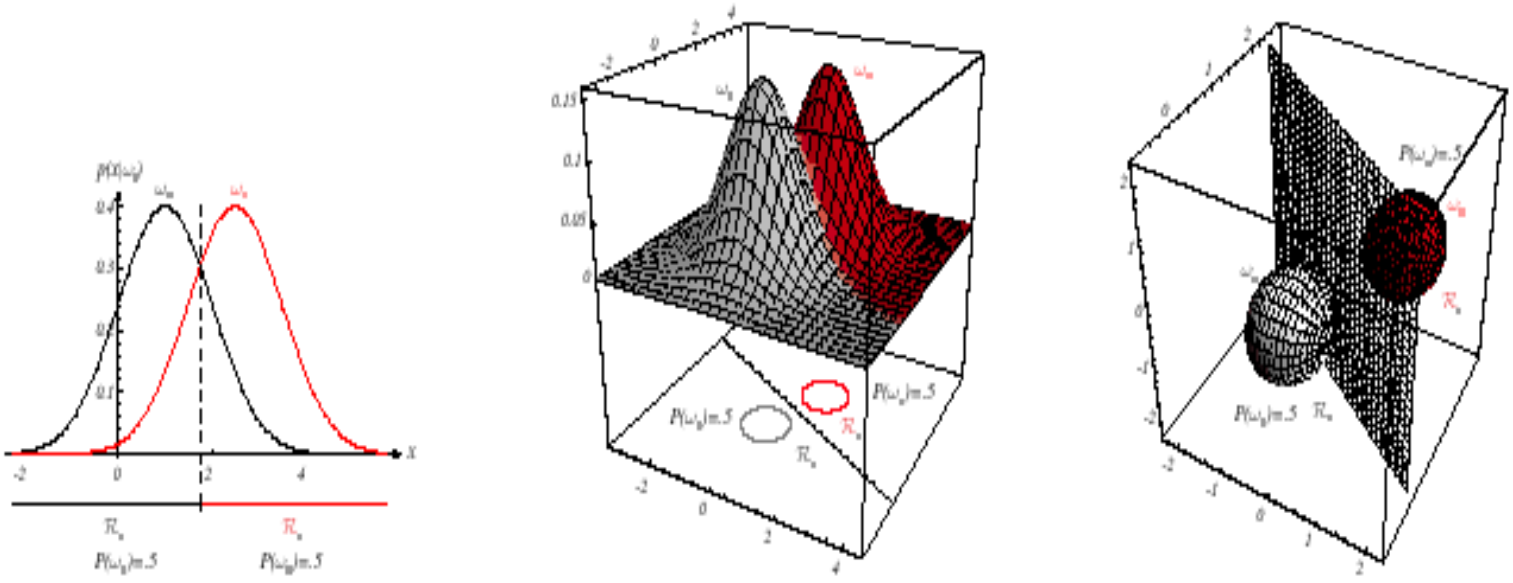


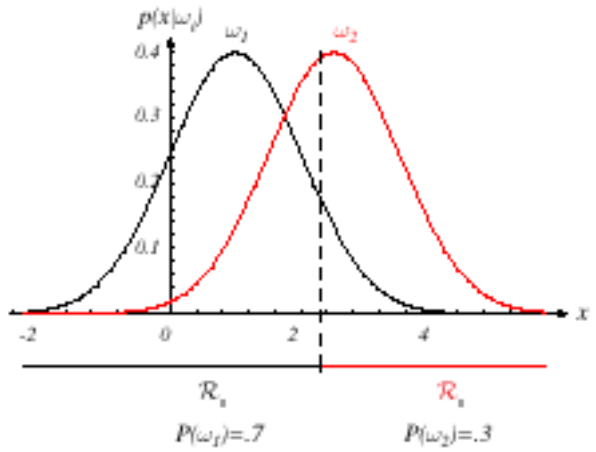
FIGURE 2.10. If the covariance matrices for two distributions are equal and proportional to the identity matrix, then the distributions are spherical in d dimensions, and the boundary is a generalized hyperplane of $d-1$ dimensions, perpendicular to the line separating the means. In these one-, two-, and three-dimensional examples, we indicate $p(\mathbf{x}|\omega_i)$ and the boundaries for the case $P(\omega_1) = P(\omega_2)$. In the three-dimensional case, the grid plane separates $\mathcal{R}_1$ from $\mathcal{R}_2$. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

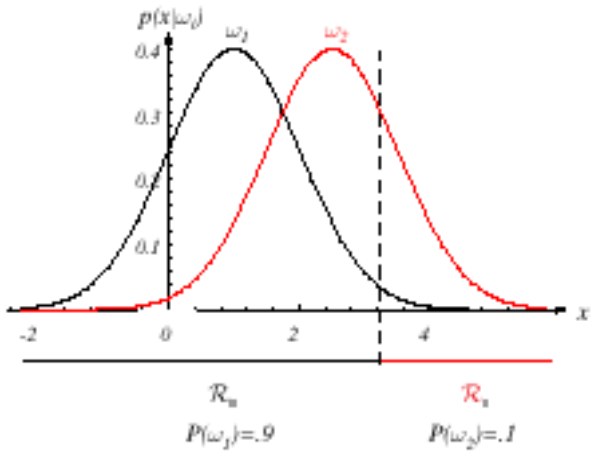
Discriminant Functions for Normal Density

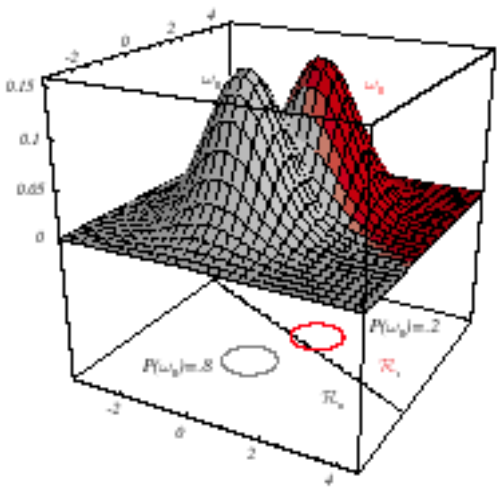
- If $P(\omega_i) \neq P(\omega_j)$ the point x_0 shifts away from the more likely mean

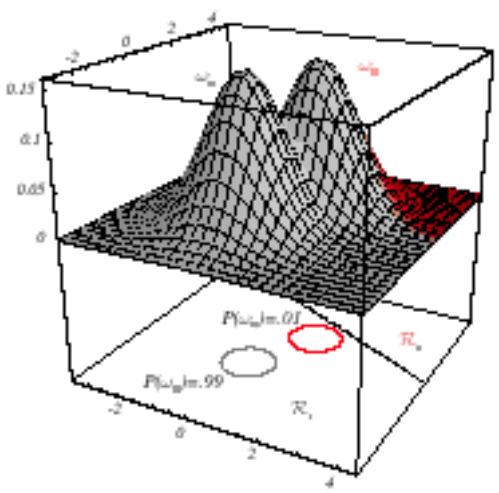
- If the prior prob $P(w_i)$ are the same for all c classes, then ln $P(w_i)$ is unimportant
- In this case, to classify a feature vector x, measure the Euclidean distance $||x-\mu_i||$ from each x to each of the c mean vectors, and assign x to the category of the nearest mean. : *minimum-distance classifier → nearest neighbor algorithm*

Discriminant Functions for Normal Density









Discriminant Functions for Normal Density

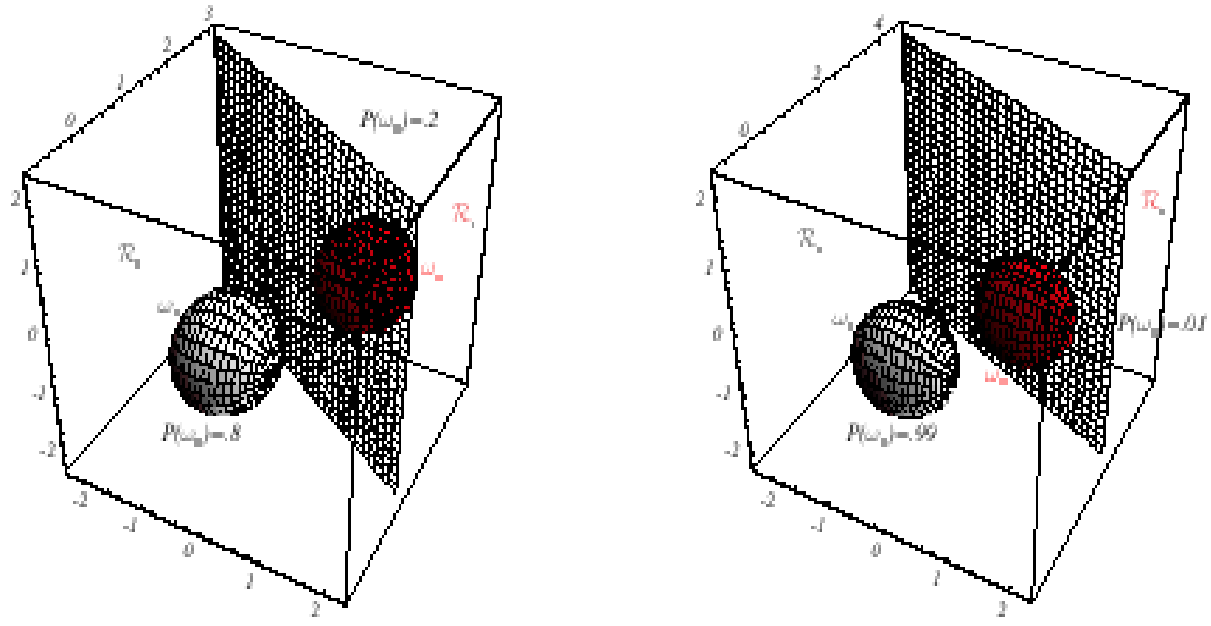


FIGURE 2.11. As the priors are changed, the decision boundary shifts; for sufficiently disparate priors the boundary will not lie between the means of these one-, two- and three-dimensional spherical Gaussian distributions. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Discriminant Functions for Normal Density

[2] Case $\Sigma_i = \Sigma$ (covariance of all classes are identical but arbitrary!)

- **Discriminant function**

$$g_i(x) = -\frac{1}{2}(x-\mu_i)^t \sum_i^{-1}(x-\mu_i) - \frac{d}{2}\ln 2\pi - \frac{1}{2}\ln|\Sigma_i| + \ln P(\omega_i)$$

$$= -\frac{1}{2}(x-\mu_i)^t \sum_i^{-1}(x-\mu_i) + \ln P(\omega_i) + \text{indept. of i}$$

- **If the prior prob *P(w$_i$)* are the same for all *c* classes, then ln *P(w$_i$)* can be ignored**

- **Thus, to classify a feature *x*, measure the squared Mahalanobis distance $(x-\mu_i)^t \sum_i^{-1}(x-\mu_i)$ from *x* to each of the *c* mean vectors, and assign *x* to the category of the nearest mean**

Discriminant Functions for Normal Density

- **Discriminant fn.**

$$g_i(x) = \mathbf{w_i^t x} + w_{i0} \text{ (linear discriminant function)}$$

$$\text{where:}$$

$$\mathbf{w_i} = \sum^{-1} \boldsymbol{\mu}_\mathbf{i};\ w_{i0} = -\frac{1}{2}\boldsymbol{\mu}_\mathbf{i}^\mathbf{t} \sum^{-1} \boldsymbol{\mu}_\mathbf{i} + \ln P(\omega_i)$$

- **The boundary surfaces for a linear machine are pieces of *hyperplanes* defined by:**

$$g_i(x) = g_j(x)$$

$$\Leftrightarrow w^t (x - x_0) = 0$$

$$\text{where } w = \Sigma^{-1}(\mu_i - \mu_j)$$

$$\text{and } x_0$$

Discriminant Functions for Normal Density

- **Hyperplane separating $\mathcal{R}_i$ and $\mathcal{R}_j$**

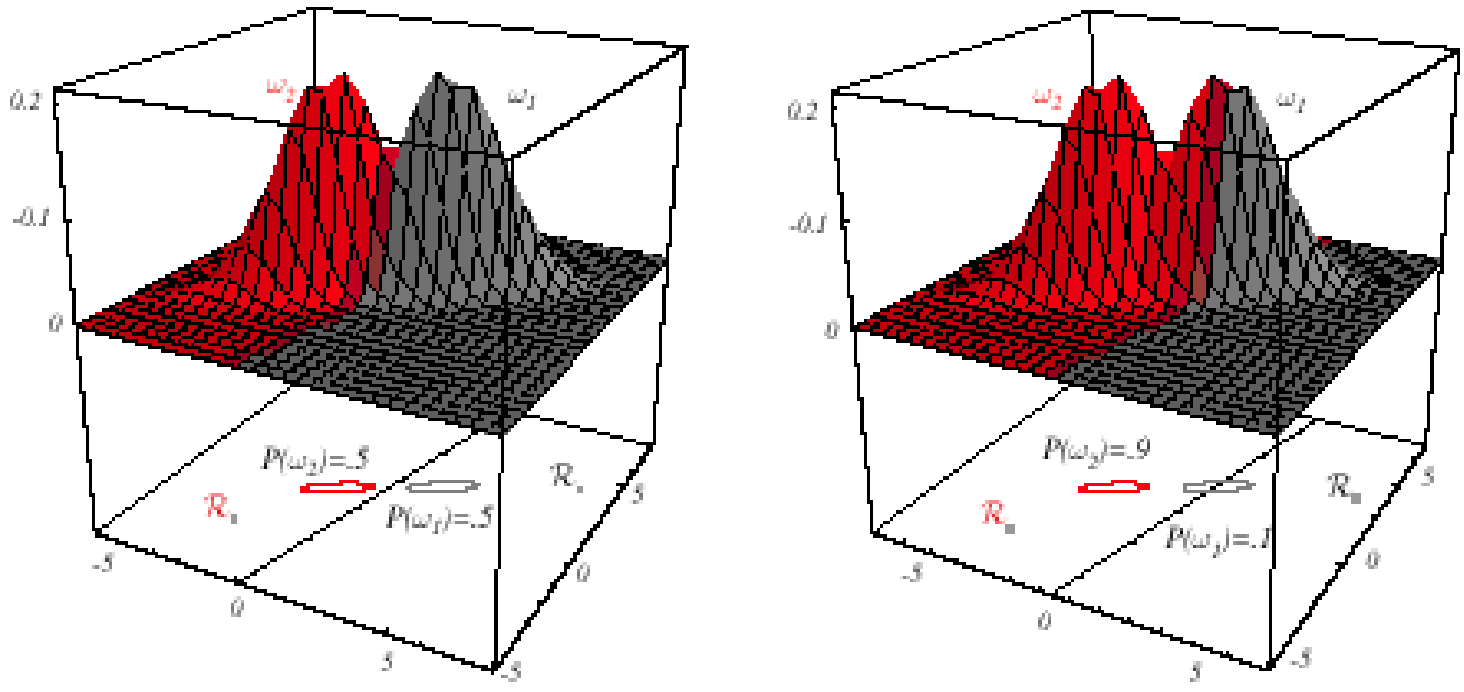
$$x_0 = \frac{1}{2}(\mu_i + \mu_j) - \frac{\ln\left[P(\omega_i)/P(\omega_j)\right]}{(\mu_i - \mu_j)^t \Sigma^{-1}(\mu_i - \mu_j)}.(\mu_i - \mu_j)$$

(Because $w = \Sigma^{-1}(\mu_i - \mu_j)$ is generally not in the direction of $(\mu_i - \mu_j)$, the hyperplane separating $\mathcal{R}_i$ and $\mathcal{R}_j$ is generally not orthogonal to the line between the means!)

- **But, it does intersect that line at the point x_0; if the prior prob are equal then x_0 is halfway between the means**

- **If the prior prob are not equal, the optimal boundary hyperplane is shifted away from the more likely mean**

Discriminant Functions for Normal Density



Discriminant Functions for Normal Density

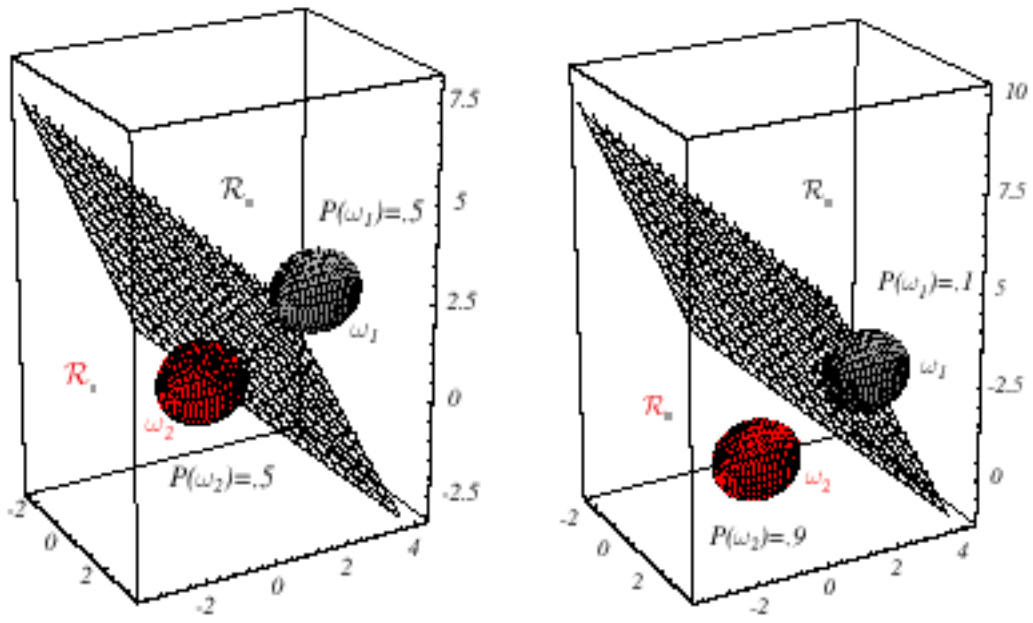


FIGURE 2.12. Probability densities (indicated by the surfaces in two dimensions and ellipsoidal surfaces in three dimensions) and decision regions for equal but asymmetric Gaussian distributions. The decision hyperplanes need not be perpendicular to the line connecting the means. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Discriminant Functions for Normal Density

[3] Case Σ_i = arbitrary

- **The covariance matrices are different for each category**

$$g_i(x) = x^t W_i x + w_i^t x + w_{i0}$$

where :

$$W_i = -\frac{1}{2}\Sigma_i^{-1}$$

$$w_i = \Sigma_i^{-1}\mu_i$$

$$w_{i0} = -\frac{1}{2}\mu_i^t\Sigma_i^{-1}\mu_i - \frac{1}{2}\ln|\Sigma_i| + \ln P(\omega_i)$$

(*Hyperquadrics* which are: hyperplanes, pairs of hyperplanes, hyperspheres, hyperellipsoids, hyperparaboloids, hyperhyperboloids of various types)

Discriminant Functions for Normal Density

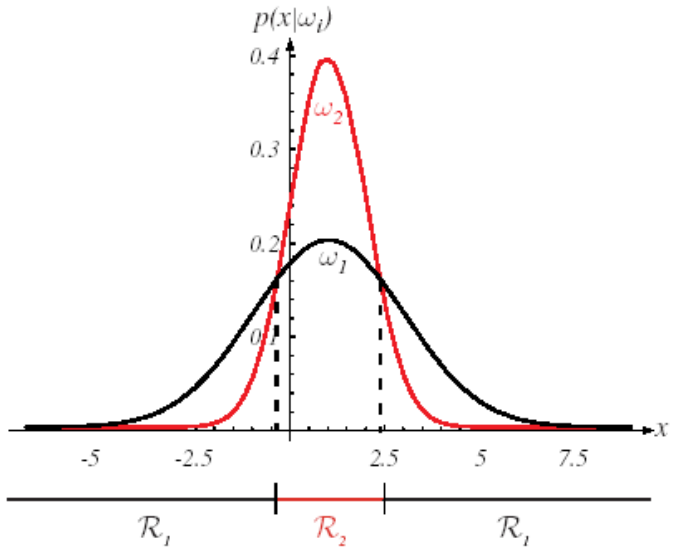
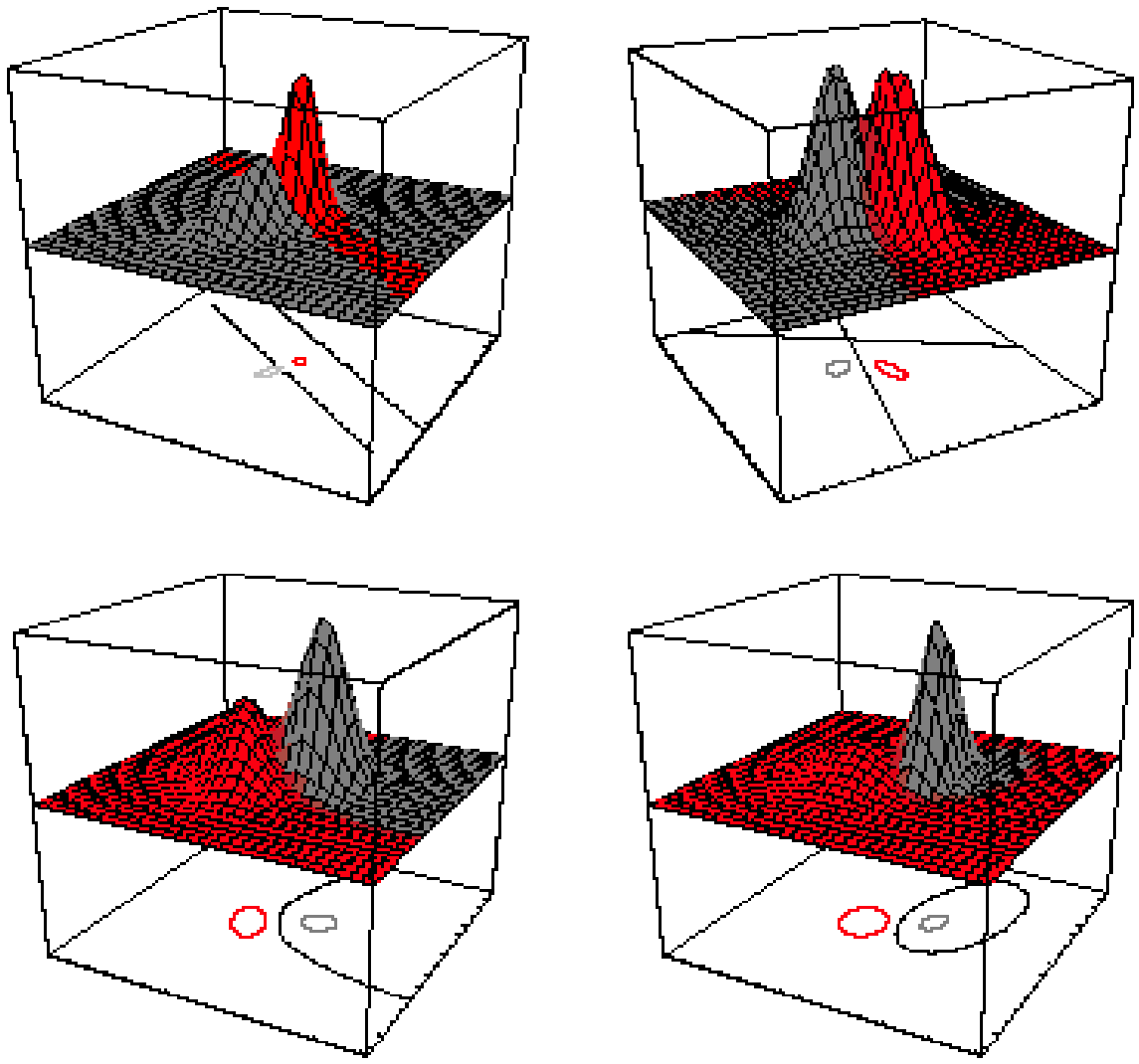


FIGURE 2.13. Non-simply connected decision regions can arise in one dimensions for Gaussians having unequal variance. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Discriminant Functions for Normal Density



Discriminant Functions for Normal Density

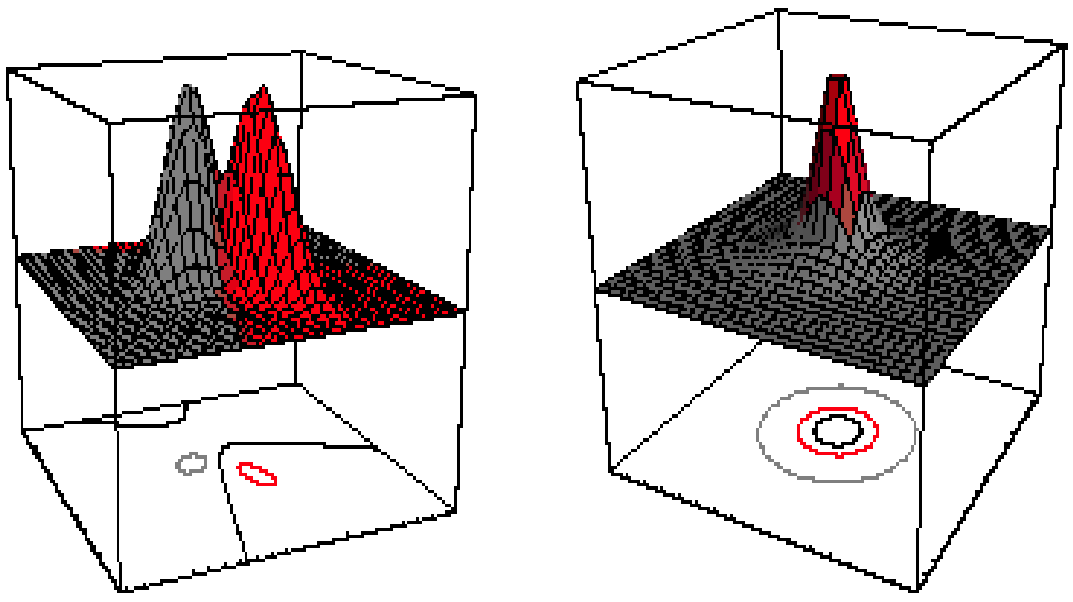


FIGURE 2.14. Arbitrary Gaussian distributions lead to Bayes decision boundaries that are general hyperquadrics. Conversely, given any hyperquadric, one can find two Gaussian distributions whose Bayes decision boundary is that hyperquadric. These variances are indicated by the contours of constant probability density. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Discriminant Functions for Normal Density

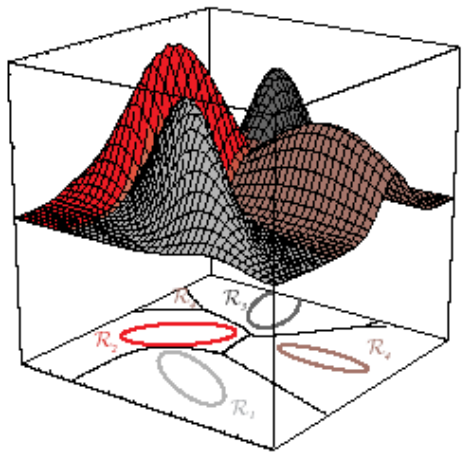


FIGURE 2.16. The decision regions for four normal distributions. Even with such a low number of categories, the shapes of the boundary regions can be rather complex. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Bayesian recipe for classification

- **Chef 1 – *Generative (Informative)* model**

  Note that $P(c_1|X) = P(X|c_1)P(c_1) / P(X)$

  Model $P(X|c_1)$, $P(X|c_2)$, $P(c_1)$, and $P(c_2)$

  Using Bayes rule, choose c_1 if $P(X|c_1)P(c_1) > P(X|c_2)P(c_2)$

  otherwise choose c_2

- **Chef 2 – *Discriminative* model**

  Model $P(c_1|X)$, $P(c_2|X)$, or the ratio $P(c_1|X) / P(c_2|X)$ directly

  Choose c_1 if $P(c_1|X) / P(c_2|X) > 1$

  Otherwise choose c_2

Summary of Bayesian recipe for classification

- The Bayesian recipe is simple, optimal, and in principle, straightforward to apply
- To use this recipe in practice, we need to know
  - $P(X|c_i)$: the *generative model for data* for each class, and
  - $P(c_i)$: the *prior probabilities of classes*
- Because these probabilities are unknown, we need to estimate them from data – or learn them!
- X is typically high-dimensional
- Need to estimate $P(X|c_i)$ from *limited* data

Naïve Bayes classifier

- We can classify *X* if we know $P(X|c_i)$
- How to learn $P(X|c_i)$?

- One solution: Assume that the random variables in *X* are conditionally independent given the class
- *Result*: Naïve Bayes classifier which performs optimally under certain assumptions
- A simple, practical learning algorithm grounded in Probability Theory

- *When to use*:
  - Attributes that describe instances are likely to be conditionally independent given classification
  - The data is insufficient to estimate all the probabilities reliably if we do not assume independence

Conditional Independence

- **Let $Z_1 ... Z_n$ and *W* be random variables on a given event space**

  $Z_1 ... Z_n$ are mutually independent given *W* if

$$P(Z_1, ..., Z_n \mid W) = \prod_{i=1}^{n} P(Z_i \mid W)$$

- **Note that these represent sets of equations, for all possible value assignments to random variables**

Implication of Independence

- **Suppose we have 5 binary attributes and a binary class label**
- **Without independence, in order to specify the joint distribution, we need to specify a probability for each possible assignment of values to each variable resulting in a table of size** $2^6 = 64$
- **Suppose the features are independent given the class label – we only need 5(2*2) = 20 entries**
- ***The reduction in the number of probabilities to be estimated is even more striking when N, the number of attributes is large – from*** $O(2^N)$ ***to*** $O(N)$

Naïve Bayes classifier

- **Consider a discrete valued target function $f : \chi \rightarrow \Omega$ where an instance $X = (X_1,..., X_n) \in \chi$ is described in terms of attribute values $X_1 = x_1,..., X_n = x_n$ where $x_i \in Domain(X_i)$**

$$c_{MAP} = \arg\max_{c_j \in \Omega} P(c_j \mid X_1 = x_1,..., X_n = x_n)$$

$$= \arg\max_{c_j \in \Omega} \frac{P(X_1 = x_1,..., X_n = x_n \mid c_j) P(c_j)}{P(X_1 = x_1,..., X_n = x_n)}$$

$$= \arg\max_{c_j \in \Omega} P(X_1 = x_1,..., X_n = x_n \mid c_j) P(c_j)$$

- c_{MAP} **is called the *maximum a posteriori* classification**

Naïve Bayes classifier

$$c_{MAP} = \arg\max_{c_j \in \Omega} P(c_j \mid X_1 = x_1, ..., X_n = x_n)$$
$$= \arg\max_{c_j \in \Omega} P(X_1 = x_1, ..., X_n = x_n \mid c_j) P(c_j)$$

- **If the attributes are *independent* given the class, we have**

$$c_{MAP} = \arg\max_{c_j \in \Omega} \prod_{i=1}^{n} P(X_i = x_i \mid c_j) P(c_j)$$
$$= c_{NB}$$
$$= \arg\max_{c_j \in \Omega} P(c_j) \prod_{i=1}^{n} P(X_i = x_i \mid c_j)$$

Naïve Bayes classifier

- **For each possible value c_j of Ω**

$$\hat{P}(\Omega = c_j) \leftarrow Estimate(P(\Omega = c_j), D)$$

- **For each possible value a_{i_k} of X_i**

$$\hat{P}(X_i = a_{i_k} \mid c_j) \leftarrow Estimate(P(X_i = a_{i_k} \mid \Omega = c_j), D)$$

- **Classify a new instance $X = (X_1, ..., X_n)$**

$$c(X) = \arg\max_{c_j \in \Omega} P(c_j) \prod_{i=1}^{n} P(X_i = x_i \mid c_j)$$

- ***Estimate* is a procedure for estimating the relevant probabilities from set of *training examples***

Estimation of probabilities from small samples

$$\hat{P}(X_i = a_{i_k} \mid c_j) \leftarrow \frac{n_{ji_k} + mp}{n_j + m}$$

n_j **is the number of training examples of class** c_j

n_{ji_k} **is the number of training examples of class** c_j **which have attribute value** a_{i_k} **for attribute** X_i

p **is the prior estimate for** $\hat{P}(X_i = a_{i_k} \mid c_j)$

m **is the weight given to the prior (*equivalent sample size*)**

- ***As*** $n_j \to \infty$, $\hat{P}(X_i = a_{i_k} \mid c_j) \to \frac{n_{ji_k}}{n_j}$

Sample applications of Naïve Bayes classifier

- Learn dating preferences
- Learn which news articles are of interest
- Learn to classify web pages by topic
- Learn to classify SPAM
- Learn to assign proteins to functional families based on amino acid composition
- …

- Naïve Bayes is among the most useful algorithms

- What attributes shall we use to represent text?

Learning dating preferences

- **Instances:**
  ordered 3-tuples of attribute values corresponding to

  Height (tall, short)
  Hair (dark, blonde, red)
  Eye (blue, brown)

- **Classes: +, -**

- **Training data:**

Instance	*Class label*
I1(t, d, l)	+
I2(s, d, l)	+
I3(t, b, l)	-
I4(t, r, l)	-
I5(s, b, l)	-
I6(t, b, w)	+
I7(t, d, w)	+
I8(s, b, w)	+

Probabilities to estimate

- *P(+) = 5/8, P(-) = 3/8*

P(Height|c)	*t*	*s*
+	3/5	2/5
-	2/3	1/3

P(Hair|c)	*d*	*b*	*r*
+	3/5	2/5	0
-	0	2/3	1/3

P(Eye|c)	*l*	*w*
+	2/5	3/5
-	1	0

- **Classify *(Height = t, Hair = b, Eye = l)*:**

  P(X|+) = (3/5)(2/5)(2/5) = 12/125

  P(X|-) = (2/3)(2/3)(1) = 4/9

  classification? → - class

- **Classify *(Height = t, Hair = r, Eye = w)***
- **Note the problem with zero probabilities**

  Solution: use the Laplace estimator (m-estimate of probabilities)

Learning to classify text

- Target concept *interesting?* : Documents → {+, -}
- Learning: use training examples to estimate

  $P(+), P(-), P(d|+), P(d|-)$
- Alternative generative models for documents:
  - Represent each document by sequence of words
    - In the most general case, we need a probability for each word occurrence in each position in the document, for each possible document length
    - Too many probabilities to estimate!
  - Represent each document by tuples of word counts

Learning to classify text

$$P(d \mid c_j) = P(length(d) \mid c_i) \prod_{i=1}^{length(d)} P(X_i = w_k \mid c_j, length(d))$$

- **This would require estimating for each document,**

$$|Vocabulary|^{length(d)} \times |\Omega|$$

  probabilities for each possible document length!

- **To simplify matters, assume that probability of encountering a specific word in a document is independent of the position, and of document length**

- **Treat each document as a bag of words!**

Bag of words representation

- So we estimate one position-independent class-conditional probability P($w_k|c_j$) for each word instead of the set of probabilities

$$P(X_1 = w_k \mid c_j) \ldots P(X_{length(d)} = w_k \mid c_j)$$

- The number of probabilities to be estimated drops to

$$|Vocabulary| \times |\Omega|$$

- The result is a generative model for documents that treats each document as an ordered tuple of word frequencies

- More sophisticated models can consider dependencies between adjacent word positions (Markov models)

Learning to classify text (Binary Independence model)

- **Given a vocabulary *V: (w1,…,w|V|)***
- **A document is a vector of binary features *(X1,…,X|V|)***
- ***Xi* is 1 if *wi* appears in the document, 0 otherwise**

$$P(d \mid c_j) = \prod_{i=1}^{|V|} P(x_i|c_j) = \prod_{i=1}^{|V|} \theta_{ji}^{x_i} (1-\theta_{ji})^{1-x_i}$$

$$\hat{\theta}_{ji} = \frac{N_{ji} + c_i}{N_j + c}$$

$$N_{ji} : \#\text{ of documents in class } j \text{ with word } w_i$$

- ***Multi-variate Bernoulli Model***
- **The number of times a word occurs in a document is not captured**

Learning to classify text (Multinomial model)

- **With the bag of words representation, we have**

$$P(d \mid c_j) \text{ is proportional to } \left\{ \frac{\left(\sum_k n_{kd} \right)!}{\prod_k n_{kd}!} \right\} \prod_k \left(P(w_k \mid c_j) \right)^{n_{kd}}$$

where n_{kd} is the number of occurrences of w_k in document *d* (ignoring dependence on length of the document)

- **We can estimate $P(w_k \mid c_j)$ from the labeled bags of words we have**

$$P(w_k \mid c_j) = \frac{n_{kj} + 1}{\sum_k n_{kj} + |Vocabulary|}$$

where n_{kj} is the number of occurrences of w_k in documents in class *j*

- **The number of times a word occurs in a document is captured**

Naïve Bayes text classifier

- **Given 1000 training documents from each group, learn to classify new documents according to the newsgroup where it belongs**
- **Naive Bayes achieves 89% classification accuracy**
  - **2/3 training; 1/3 testing**
  - **100 most frequent & less than 3 occurrence words were removed**
- **20 news groups**

comp.graphics
comp.os.ms-windows.misc
comp.sys.ibm.pc.hardwadre
comp.sys.mac.hardware
comp.windows.x
alt.atheism
soc.religion.christian
talk.religion.misc
talk.politics.mideast
talk.politics.misc
talk.politics.guns

misc.forsale
rec.autos
rec.motorcycles
rec.sport.baseball
rec.sport.hockey
sci.space
sci.crypt
sci.electronics
sci.med

Naïve Bayes text classifier

- **Representative article from** rec.sport.hockey

Path: cantaloupe.srv.cs.cmu.edu!das-news.harvard.edu!ogicse!uwm.edu
From: xxx@yyy.zzz.edu (John Doe)
Subject: Re: This year's biggest and worst (opinion)...
Date: 5 Apr 93 09:53:39 GMT

I can only comment on the Kings, but the most obvious candidate for pleasant surprise is Alex Zhitnik. He came highly touted as a defensive defenseman, but he's clearly much more than that. Great skater and hard shot (though wish he were more accurate). In fact, he pretty much allowed the Kings to trade away that huge defensive liability Paul Coffey. Kelly Hrudey is only the biggest disappointment if you thought he was any good to begin with. But, at best, he's only a mediocre goaltender. A better choice would be Tomas Sandstrom, though not through any fault of his own, but because some thugs in Toronto decided ….

Naïve Bayes Learner – Summary

- **Produces minimum error classifier if attributes are conditionally independent given the class**

- ***When to use***
  - **Attributes that describe instances are likely to be conditionally independent given classification**
  - **There is not enough data to estimate all the probabilities reliably if we do not assume independence**
  - **Often works well even if when independence assumption is violated (Domingos and Pazzani, 1996)**
  - **Can be used iteratively (Kang *et al*., 2006) – generates a tree of NB classifiers**