

Decision and Estimation in Data Processing

Chapter II. Elements of Signal Detection Theory

II.1 Introduction

Introduction

- ▶ Signal detection = the problem of deciding which signal is present from 2 or more possibilities
 - ▶ one possibility may be that there is no signal
- ▶ Based on **noisy** observations
 - ▶ signals are affected by noise
 - ▶ noise is additive (added to the original signal)

The context for signal detection

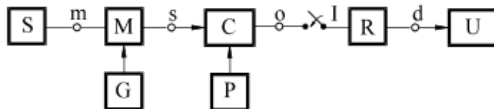


Figure 1: Block scheme of a communication system

- ▶ Block scheme of a communication system:
 - ▶ Information source: generates messages a_n with probabilities $p(a_n)$
 - ▶ Generator: generates different signals $s_1(t), \dots, s_n(t)$
 - ▶ Modulator: transmits a signal $s_n(t)$ for message a_n
 - ▶ Channel: adds random noise
 - ▶ Sampler: takes samples from the signal $s_n(t)$
 - ▶ Receiver: **decides** what message a_n has been transmitted
 - ▶ User receives the recovered messages

Problem formulation

- ▶ There are two messages a_0 and a_1 (e.g. logical 0 and 1)
- ▶ Messages are encoded as signals $s_0(t)$ and $s_1(t)$
 - ▶ for a_0 : send $s(t) = s_0(t)$
 - ▶ for a_1 : send $s(t) = s_1(t)$
- ▶ The signal is affected by additive white noise $n(t)$
- ▶ Receiver receives noisy signal $r(t) = s(t) + n(t)$
- ▶ **Decision problem:** based on $r(t)$, decide which signal was received, $s_0(t)$ or $s_1(t)$?

- ▶ Data transmission with various binary modulations:
 - ▶ Constant voltage levels (e.g. $s_n(t) = \text{constant} = 0$ or 5V)
 - ▶ PSK modulation (Phase Shift Keying): $s_n(t) = \text{cosine}$ with same frequency but various initial phases
 - ▶ FSK modulation (Frequency Shift Keying): $s_n(t) = \text{cosines}$ with different frequencies
 - ▶ OFDM modulation (Orthogonal Frequency Division Multiplexing): particular case of FSK
 - ▶ The receiver gets some noisy signal, has to decide when it is 0 and when it is 1

- ▶ Radar detections:

- ▶ a signal is emitted; if there is an obstacle, the signal gets reflected back
- ▶ the receiver waits for possible reflections of the signal and must **decide**:
 - ▶ no reflection is present -> no object
 - ▶ reflected signal is present -> object detected

Generalizations

- ▶ Decide between more than two signals
- ▶ Number of observations:
 - ▶ use only one sample
 - ▶ use multiple samples
 - ▶ observe the whole continuous signal for some time T

II.2 Detection of signals based on 1 sample

Problem formulation

- ▶ There are two messages a_0 and a_1 (e.g. logical 0 and 1)
- ▶ Messages are encoded as signals $s_0(t)$ and $s_1(t)$
 - ▶ for a_0 : send $s(t) = s_0(t)$
 - ▶ for a_1 : send $s(t) = s_1(t)$
- ▶ The signal is affected by additive white noise $n(t)$
- ▶ Receiver receives noisy signal $r(t) = s(t) + n(t)$
- ▶ **Decision problem:** based on $r(t)$, decide which signal was received, $s_0(t)$ or $s_1(t)$?
- ▶ Simplest case: receiver **takes just 1 sample** at time t_0 , value is $r = r(t_0)$

Hypotheses and decisions

- ▶ There are **two hypotheses**:
 - ▶ H_0 : true signal is $s(t) = s_0(t)$ (a_0 has been transmitted)
 - ▶ H_1 : true signal is $s(t) = s_1(t)$ (a_1 has been transmitted)
- ▶ Receiver can take **two decisions**:
 - ▶ D_0 : receiver decides that signal was $s(t) = s_0(t)$
 - ▶ D_1 : receiver decides that signal was $s(t) = s_1(t)$

Possible outcomes

► There are 4 possible outcomes:

1. **Correct rejection**: true hypothesis is H_0 , decision is D_0

► Probability is $P_r = P(D_0 \cap H_0)$

► Also known as **True Negative**

2. **False alarm**: true hypothesis is H_0 , decision is D_1

► Probability is $P_{fa} = P(D_1 \cap H_0)$

► Also known as **False Positive**

3. **Miss**: true hypothesis is H_1 , decision is D_0

► Probability is $P_m = P(D_0 \cap H_1)$

► Also known as **False Negative**

4. **Correct detection** (“hit”): true hypothesis is H_1 , decision is D_1

► Probability is $P_d = P(D_1 \cap H_1)$

► Also known as **True Positive**

Origin of terms

- ▶ The terms originate from radar applications:
 - ▶ a signal is emitted from source
 - ▶ received signal = possible reflection from a target, with lots of noise
 - ▶ H_0 = no target is present, no reflected signal (only noise)
 - ▶ H_1 = target is present, there is a reflected signal
 - ▶ hence the names “miss”, “hit” etc.

The noise

- ▶ In general we consider **additive, white, stationary** noise
 - ▶ additive = the noise is added to the signal
 - ▶ white = two samples from the noise are uncorrelated
 - ▶ stationary = has same statistical properties at all times
- ▶ The noise signal $n(t)$ is unknown
 - ▶ it's random
 - ▶ we just know it's distribution, but not the actual values

The sample

- ▶ The receiver receives:

$$r(t) = s(t) + n(t)$$

- ▶ $s(t)$ = original signal, either $s_0(t)$ or $s_1(t)$
 - ▶ $n(t)$ = unknown noise
- ▶ The value of the sample taken at t_0 is:

$$r(t_0) = s(t_0) + n(t_0)$$

- ▶ $s(t_0)$ = the true signal = either $s_0(t_0)$ or $s_1(t_0)$
 - ▶ $n(t_0)$ = a sample from the noise

The sample

- ▶ The sample $n(t_0)$ is a **random variable**
 - ▶ since it is a sample of noise (a sample from a random process)
 - ▶ assume is a continuous r.v., i.e. range of possible values is continuous
- ▶ $r(t_0) = s(t_0) + n(t_0) = \text{a constant} + \text{a random variable}$
 - ▶ it is also a random variable
 - ▶ $s(t_0)$ is a constant, either $s_0(t_0)$ or $s_1(t_0)$
- ▶ What distribution does $r(t_0)$ have?
 - ▶ a constant + a r.v. = has same distribution as r.v., but shifted with the constant

The conditional distributions

- ▶ Assume the noise has known distribution $w(x)$
- ▶ The distribution of $r = w(x)$ shifted by $s(t_0)$
- ▶ In hypothesis H_0 , the distribution is $w(r|H_0) = w(x)$ shifted by $s_0(t_0)$
- ▶ In hypothesis H_1 , the distribution is $w(r|H_1) = w(x)$ shifted by $s_1(t_0)$
- ▶ $w(r|H_0)$ and $w(r|H_1)$ are known as **conditional distributions** or **likelihood functions**
 - ▶ “|” means “conditioned by”, “given that”
 - ▶ i.e. considering one hypothesis or the other one
 - ▶ r is the unknown of the function

The conditional distributions

Example:

- ▶ A constant signal $s(t)$ can have two values, 0 or 4. The signal is affected by noise $\mathcal{N}(\mu = 0, \sigma^2 = 2)$. What is the distribution of a sample r , in both hypotheses?

Decision problem

The problem of decision:

- ▶ We have two possible distributions (one in each hypothesis)
- ▶ We have a sample $r = r(t_0)$, which could have come from either one
- ▶ Which hypothesis do we **decide** is the correct one?

The likelihood of a parameter

- ▶ In general, the **likelihood** of a some parameter P based on some **observation** O = the probability density of O , if the parameter has value P :

$$L(P|O) = w(O|P)$$

- ▶ In our case:
 - ▶ the unknown parameter = which hypothesis H is the true one
 - ▶ the observation = the sample r that we got
- ▶ The **likelihood of a hypothesis H** based on the **observation r** is:

$$L(H_0|r) = w(r|H_0)$$

$$L(H_1|r) = w(r|H_1)$$

Maximum Likelihood decision criterion

- ▶ **Maximum Likelihood (ML) criterion:** choose the hypothesis that has the **highest likelihood** of having generated the observed sample value $r = r(t_0)$
 - ▶ “pick the most likely hypothesis”
 - ▶ “pick the hypothesis with a higher likelihood”

$$\frac{L(H_1|r)}{L(H_0|r)} = \frac{w(r|H_1)}{w(r|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} 1$$

- ▶ We choose the higher value between $w(r(t_0)|H_0)$ and $w(r(t_0)|H_1)$
- ▶ This is known as a **likelihood ratio** test

Example: gaussian noise

Example (follow-up):

- ▶ A constant signal $s(t)$ can have two values, 0 or 4. The signal is affected by noise $\mathcal{N}(\mu = 0, \sigma^2 = 2)$.
- ▶ What is the decision taken with the ML criterion, if $r = 1.6$?
- ▶ At blackboard:
 - ▶ plot the two conditional distributions for $w(r|H_0)$, $w(r|H_1)$
 - ▶ discuss the decision taken for different values of r
 - ▶ discuss the choice of the threshold value T for taking decisions

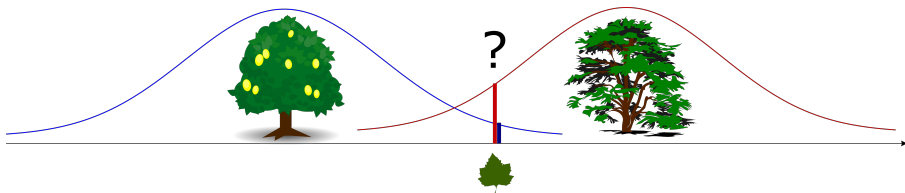
Example: Trees

From what tree did the leaf fall?



Example: Trees

Pick the tree with the **highest likelihood**:



Gaussian noise (AWGN)

- ▶ Particular case: the noise has normal distribution $\mathcal{N}(0, \sigma^2)$

- ▶ i.e. it is AWGN

- ▶ Likelihood ratio is $\frac{w(r|H_1)}{w(r|H_0)} = \frac{e^{-\frac{(r-s_1(t_0))^2}{2\sigma^2}}}{e^{-\frac{(r-s_0(t_0))^2}{2\sigma^2}}} \underset{H_0}{\overset{H_1}{\gtrless}} 1$

- ▶ For normal distribution, it is easier to apply **natural logarithm** to the terms

- ▶ logarithm is a monotonic increasing function, so it won't change the comparison
 - ▶ if $A < B$, then $\log(A) < \log(B)$

Gaussian noise (AWGN)

- ▶ Applying natural logarithm to both sides leads to:

$$-(r - s_1(t_0))^2 + (r - s_0(t_0))^2 \underset{H_0}{\overset{H_1}{\geq}} 0$$

- ▶ Which means

$$|r - s_0(t_0)| \underset{H_0}{\overset{H_1}{\geq}} |r - s_1(t_0)|$$

- ▶ Note that $|r - A| = \mathbf{distance}$ from r to A
 - ▶ $|r| = \mathbf{distance}$ from r to 0
- ▶ So we choose the **smallest distance** between $r(t_0)$ and $s_1(t_0)$ vs $s_0(t_0)$

Maximum Likelihood for gaussian noise

- ▶ ML criterion **for gaussian noise**: choose the hypothesis based on whichever of $s_0(t_0)$ or $s_1(t_0)$ is **nearest** to our observed sample $r = r(t_0)$
 - ▶ also known as **nearest neighbor** principle / decision
 - ▶ very general principle, encountered in many other scenarios
 - ▶ because of this, a receiver using ML is also known as **minimum distance receiver**

Steps for ML decision

1. Sketch the two conditional distributions $w(r|H_0)$ and $w(r|H_1)$
2. Find out which function is higher at the observed value $r = r(t_0)$ given.

Steps for ML decision in case of gaussian noise

- ▶ Only if the noise is Gaussian, identical for all hypotheses:
 1. Find $s_0(t_0)$ = the value of the original signal, in absence of noise, in case of hypothesis H_0
 2. Find $s_1(t_0)$ = the value of the original signal, in absence of noise, in case of hypothesis H_1
 3. Compare with observed sample $r(t_0)$ and choose **the nearest**

Thresholding based decision

- ▶ Choosing the nearest value = same thing as **comparing r with a threshold $T = \frac{s_0(t_0) + s_1(t_0)}{2}$**
 - ▶ i.e. if the two values are 0 and 5, decide by comparing with 2.5 (like in laboratory)
- ▶ For the **ML criterion**, the threshold = the **cross-over point** between the conditioned distributions

Exercise

- ▶ A signal can have two possible values, 0 or 5. The signal is affected by white gaussian noise $\mathcal{N}(\mu = 0, \sigma^2 = 2)$. The receiver takes one sample with value $r = 2.25$.
 - a. Write the expressions of the conditional probabilities and sketch them
 - b. What is the decision based on the Maximum Likelihood criterion?
 - c. What if the signal 0 is affected by gaussian noise $\mathcal{N}(0, 0.5)$, while the signal 5 is affected by uniform noise $\mathcal{U}[-4, 4]$?
 - d. Repeat b. and c. assuming the value 0 is replaced by -1

Decision regions

- ▶ The **decision regions** = the range of values of r for which a certain decision is taken
- ▶ Decision regions R_0 = all the values of r which lead to decision D_0
- ▶ Decision regions R_1 = all the values of r which lead to decision D_1
- ▶ The decision regions cover the whole \mathbb{R} axis
- ▶ Example: indicate the decision regions for the previous exercise:
 - ▶ $R_0 = [-\infty, 2.5]$
 - ▶ $R_1 = [2.5, \infty]$

The likelihood function

- ▶ The subtle distinction in terms: “probability” vs “likelihood”
- ▶ Consider the conditional distribution $w(r|H_i)$ in the previous example:

$$\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(r-s_i(t_0))^2}{2\sigma^2}}$$

- ▶ Which is the unknown in this function?
 - ▶ in general, the unknown is r
 - ▶ but for our decision problem it is i , and r is known

Terminology: probability vs likelihood

- ▶ In the same mathematical expression of a distribution function:
 - ▶ if we know the parameters (e.g. μ, σ, H_i), and the unknown is the value (e.g. r, x) we call it **probability density function** (distribution)
 - ▶ if we know value (e.g. r, x), and the unknown is some statistical parameter (e.g. μ, σ, i), we call it a **likelihood function**

Generalizations

- ▶ What if the noise has another distribution?
 - ▶ Sketch the conditional distributions
 - ▶ Locate the given $r = r(t_0)$
 - ▶ ML criterion = choose the highest function $w(r|H_i)$ in that point
- ▶ The decision regions are defined by the **cross-over points**
 - ▶ There can be more cross-overs, so multiple thresholds

Generalizations

- ▶ What if the noise has a different distribution in hypothesis H_0 than in hypothesis H_1 ?
- ▶ Same thing:
 - ▶ Sketch the conditional distributions
 - ▶ Locate the given $r = r(t_0)$
 - ▶ ML decision = choose **the highest function** $w(r|H_i)$ in that point

Generalizations

- ▶ What if the two signals $s_0(t)$ and $s_1(t)$ are constant / not constant?
- ▶ We **don't care about the shape** of the signals
- ▶ All we care about are **the two values at the sample time t_0** :
 - ▶ $s_0(t_0)$
 - ▶ $s_1(t_0)$

Generalizations

- ▶ What if we have more than two hypotheses?
- ▶ Extend to n hypotheses
 - ▶ We have n possible signals $s_0(t), \dots, s_{n-1}(t)$
 - ▶ We have n different values $s_0(t_0), \dots, s_{n-1}(t_0)$
 - ▶ We have n conditional distributions $w(r|H_i)$
 - ▶ We **choose the highest function** $w(r|H_i)$ in the point $r = r(t_0)$

Generalizations

- ▶ What if we take more than 1 sample?
- ▶ Patience, we'll treat this later as a separate sub-chapter

Multiple separate detection

- ▶ In a binary communications setup, each detection/decision reads 1 bit
- ▶ We have a different detection for the next bit, and so on

Exercise

- ▶ A signal can have four possible values: -6 , -2 , 2 , 6 . Each value lasts for 1 second. The signal is affected by white noise with normal distribution. The receiver takes 1 sample per second. Using ML criterion, decide what signal has been transmitted, if the received samples are:

4, 6.6, -5.2 , 1.1, 0.3, -1.5 , 7, -7 , 4.4

Conditional probabilities

- ▶ We compute the **conditional probabilities** of the 4 possible outcomes
- ▶ Consider the decision regions:
 - ▶ R_0 : when $r \in R_0$, decision is D_0
 - ▶ R_1 : when $r \in R_1$, decision is D_1
- ▶ Conditional probability of correct rejection
 - ▶ = probability to take decision D_0 in the case that hypothesis is H_0
 - ▶ = probability that r is in R_0 computed from the distribution $w(r|H_0)$

$$P(D_0|H_0) = \int_{R_0} w(r|H_0) dx$$

- ▶ Conditional probability of false alarm
 - ▶ = probability to take decision D_1 in the case that hypothesis is H_0
 - ▶ = probability that r is in R_1 computed from the distribution $w(r|H_0)$

$$P(D_1|H_0) = \int_{R_1} w(r|H_0) dx$$

Conditional probabilities

- ▶ Conditional probability of miss
 - ▶ = probability to take decision D_0 in the case that hypothesis is H_1
 - ▶ = probability that r is in R_0 computed from the distribution $w(r|H_1)$

$$P(D_0|H_1) = \int_{R_0} w(r|H_1) dx$$

- ▶ Conditional probability of correct rejection
 - ▶ = probability to take decision D_1 in the case that hypothesis is H_1
 - ▶ = probability that r is in R_1 computed from the distribution $w(r|H_1)$

$$P(D_1|H_1) = \int_{R_1} w(r|H_1) dx$$

Conditional probabilities

- ▶ Relation between them:
 - ▶ $P(D_0|H_0) + P(D_1|H_0) = 1$ (correct rejection + false alarm)
 - ▶ $P(D_0|H_1) + P(D_1|H_1) = 1$ (miss + correct detection)
 - ▶ Why? Prove this.

Computing conditional probabilities

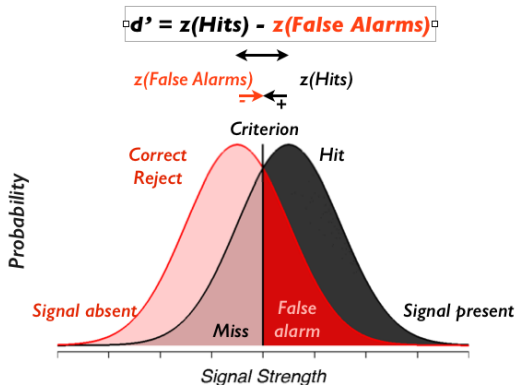


Figure 2: Conditional probabilities

- ▶ Ignore the text, just look at the nice colors
- ▶ [image from <http://gru.stanford.edu/doku.php/tutorials/sdt>]*

ML criterion optimality

Theorem:

The ML criterion **minimizes the total conditioned probability of error**
 $P(D_1|H_0) + P(D_0|H_1)$

Proof:

Informal: on the previous pixture, if T is moved either to the right or to the left, the sum of the two areas of false alarm + misses increases.

TODO: rigorous proof

Probabilities of the 4 outcomes

- ▶ Conditional probabilities are computed **given that** one or the other hypothesis is true
- ▶ They do not account for the probabilities **of the hypotheses themselves**
 - ▶ i.e. $P(H_0)$ = how many times does H_0 happen?
 - ▶ $P(H_1)$ = how many times does H_1 happen?
- ▶ To account for these, multiply with $P(H_0)$ or $P(H_1)$
 - ▶ $P(H_0)$ and $P(H_1)$ are known as the **prior** (or **a priori**) probabilities of the hypotheses

Reminder: the Bayes rule

- ▶ Reminder: **the Bayes rule**

$$P(A \cap B) = P(B|A) \cdot P(A)$$

- ▶ Interpretation:

- ▶ The probability $P(A)$ is taken out from $P(B|A)$
- ▶ $P(B|A)$ gives no information on $P(A)$, the chances of A actually happening
- ▶ Example: $P(\text{score} \mid \text{shoot}) = \frac{1}{2}$. How many goals are scored?

- ▶ In our case:

$$P(D_i \cap H_j) = P(D_i|H_j) \cdot P(H_j)$$

- ▶ for all i and j , i.e. all 4 cases

Exercise

- ▶ A constant signal can have two possible values, -2 or 5 . The signal is affected by gaussian noise $\mathcal{N}(\mu = 0, \sigma^2 = 2)$. The receiver performs ML decision based on a single sample.
 - a. Compute the conditional probability of a false alarm
 - b. Compute the conditional probability of a miss
 - c. If $P(H_0) = \frac{1}{3}$ and $P(H_1) = \frac{2}{3}$, compute the actual probabilities of correct rejection and correct detection (not conditional)

Pitfalls of ML decision criterion

- ▶ The ML criterion is based on comparing **conditional** distributions
 - ▶ conditioned by H_0 or by H_1
- ▶ Conditioning by H_0 and H_1 **ignores the prior probabilities of H_0 or H_1**
 - ▶ Our decision doesn't change if we know that $P(H_0) = 99.99\%$ and $P(H_1) = 0.01\%$, or vice-versa
- ▶ But if $P(H_0) > P(H_1)$, we may want to move the threshold towards H_1 , and vice-versa
 - ▶ because it is more likely that the true signal is $s_0(t)$
 - ▶ and thus we want to “encourage” decision D_0
- ▶ Looks like we want a more general criterion . . .

Example: Football fields

TODO

Minimum error probability criterion

- ▶ Takes into account the probabilities $P(H_0)$ and $P(H_1)$
- ▶ **The minimum probability of error** criterion (MPE):

$$\frac{w(r|H_1)}{w(r|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{P(H_0)}{P(H_1)}$$

$$\frac{P(H_1) \cdot w(r|H_1)}{P(H_0) \cdot w(r|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} 1$$

The minimum error probability criterion

Theorem:

The MPE decision criterion **minimizes the total probability of errors**:

$$P_e = P_{fa} + P_m = P(D_1 \cap H_0) + P(D_0 \cap H_1)$$

- errors = false alarms and misses

Minimum error probability criterion

Proof:

- ▶ The probability of false alarm is:

$$\begin{aligned}P(D_1 \cap H_0) &= P(D_1|H_0) \cdot P(H_0) \\&= \int_{R_1} w(r|H_0) dx \cdot P(H_0) \\&= (1 - \int_{R_0} w(r|H_0) dx) \cdot P(H_0)\end{aligned}$$

- ▶ The probability of miss is:

$$\begin{aligned}P(D_0 \cap H_1) &= P(D_0|H_1) \cdot P(H_1) \\&= \int_{R_0} w(r|H_1) dx \cdot P(H_1)\end{aligned}$$

- ▶ The total error probability (their sum) is:

$$P_e = P(H_0) + \int_{R_0} [w(r|H_1) \cdot P(H_1) - w(r|H_0) \cdot P(H_0)] dx$$

Minimum probability of error

- ▶ An integral is always minimal when you integrate the function on all the domain where it is negative
- ▶ The term $w(r|H_1) \cdot P(H_1) - w(r|H_0) \cdot P(H_0)$ is negative whenever

$$w(r|H_1) \cdot P(H_1) < w(r|H_0) \cdot P(H_0)$$

i.e. when we take decision D_0 , i.e. on region R_0

- ▶ Therefore the integral on R_0 is minimal,
- ▶ $P(H_0)$ is a constant $\Rightarrow P_e$ is minimal

Interpretation

- ▶ MPE criterion is more general than ML, depends on probabilities of the two hypotheses
 - ▶ Also expressed as a likelihood ratio test
- ▶ When one hypothesis has higher probability than the other, the threshold is **pushed in its favor**, towards the other one
- ▶ The ML criterion is a particular case of the MPE criterion, for $P(H_0) = P(H_1) = \frac{1}{2}$

Minimum probability of error - Gaussian noise

- Assuming the noise has normal distribution $\mathcal{N}(0, \sigma^2)$

$$w(r|H_1) = e^{-\frac{(r-s_1(t_0))^2}{2\sigma^2}}$$

$$w(r|H_0) = e^{-\frac{(r-s_0(t_0))^2}{2\sigma^2}}$$

- Apply natural logarithm

$$-\frac{(r-s_1(t_0))^2}{2\sigma^2} + \frac{(r-s_0(t_0))^2}{2\sigma^2} \underset{H_0}{\overset{H_1}{\gtrless}} \ln \left(\frac{P(H_0)}{P(H_1)} \right)$$

- Equivalently

$$(r-s_0(t_0))^2 \underset{H_0}{\overset{H_1}{\gtrless}} (r-s_1(t_0))^2 + 2\sigma^2 \cdot \ln \left(\frac{P(H_0)}{P(H_1)} \right)$$

- or, after further processing:

$$r \underset{H_0}{\overset{H_1}{\gtrless}} \frac{s_0(t_0) + s_1(t_0)}{2} + \frac{\sigma^2}{s_1(t_0) - s_0(t_0)} \cdot \ln \left(\frac{P(H_0)}{P(H_1)} \right)$$

Interpretation 1: Comparing distance

- ▶ For ML criterion, we compare the (squared) distances:

$$|r - s_0(t_0)| \underset{H_0}{\overset{H_1}{\geq}} |r - s_1(t_0)|$$

$$(r - s_0(t_0))^2 \underset{H_0}{\overset{H_1}{\geq}} (r - s_1(t_0))^2$$

- ▶ For MPE criterion, we compare the squared distances, but a supplementary term appears in favour of the most probable hypothesis:

$$(r - s_0(t_0))^2 \underset{H_0}{\overset{H_1}{\geq}} (r - s_1(t_0))^2 + 2\sigma^2 \cdot \ln \left(\frac{P(H_0)}{P(H_1)} \right)$$

- ▶ term depends on the ratio $\frac{P(H_0)}{P(H_1)}$

Interpretation 2: The threshold value

- For ML criterion, we compare r with a threshold T

$$r \underset{H_0}{\overset{H_1}{\gtrless}} \frac{s_0(t_0) + s_1(t_0)}{2}$$

- For MPE criterion, the threshold is moved towards the less probable hypothesis:

$$r \underset{H_0}{\overset{H_1}{\gtrless}} \frac{s_0(t_0) + s_1(t_0)}{2} + \frac{\sigma^2}{s_1(t_0) - s_0(t_0)} \cdot \ln \left(\frac{P(H_0)}{P(H_1)} \right)$$

- depending on the ratio $\frac{P(H_0)}{P(H_1)}$

Exercises

- ▶ Consider the decision between two constant signals: $s_0(t) = -5$ and $s_1(t) = 5$. The signals are affected by gaussian noise $\mathcal{N}(0, \sigma^2 = 3)$. The receiver takes one sample r .
 - Find the decision regions R_0 and R_1 according to the MPE criterion
 - What are the probabilities of false alarm and of miss?
 - Repeat a) and b) considering that $s_1(t)$ is affected by uniform noise $\mathcal{U}[-4, 4]$

Minimum risk criterion

- ▶ What if we care more about one type of errors (e.g. false alarms) than other kind (e.g. miss)?
 - ▶ MPE criterion treats all errors the same
 - ▶ Need a more general criterion
- ▶ Idea: assign a **cost** to each scenario, minimize average cost
- ▶ C_{ij} = cost of decision D_i when true hypothesis was H_j
 - ▶ C_{00} = cost for good detection D_0 in case of H_0
 - ▶ C_{10} = cost for false alarm (detection D_1 in case of H_0)
 - ▶ C_{01} = cost for miss (detection D_0 in case of H_1)
 - ▶ C_{11} = cost for good detection D_1 in case of H_1
- ▶ The idea of assigning “costs” and minimizing average cost is very general
 - ▶ e.g. IT: Shannon coding: “cost” of each message is the length of its codeword, we want to minimize average cost, i.e. minimize average length

Minimum risk criterion

- ▶ Define the **risk** = **the average cost** value

$$R = C_{00}P(D_0 \cap H_0) + C_{10}P(D_1 \cap H_0) + C_{01}P(D_0 \cap H_1) + C_{11}P(D_1 \cap H_1)$$

- ▶ Minimum risk criterion: **minimize the risk R**
 - ▶ i.e. minimize the average cost
 - ▶ also known as “minimum cost criterion”

- ▶ Proof on blackboard: (sorry, no time to put in on slides)
 - ▶ Use Bayes rule
 - ▶ Notations: $w(r|H_j)$ (*likelihood*)
 - ▶ Probabilities: $\int_{R_i} w(r|H_j) dV$
- ▶ Conclusion, **decision rule is**

$$\frac{w(r|H_1)}{w(r|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{(C_{10} - C_{00})p(H_0)}{(C_{01} - C_{11})p(H_1)}$$

Minimum risk criterion

Minimum risk criterion (MR):

$$\frac{w(r|H_1)}{w(r|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{(C_{10} - C_{00})p(H_0)}{(C_{01} - C_{11})p(H_1)}$$

- ▶ MR is a generalization of MPE criterion (which was itself a generalization of ML)
 - ▶ also expressed as a likelihood ratio test
- ▶ Both **probabilities** and the assigned **costs** can influence the decision towards one hypothesis or the other
- ▶ If $C_{10} - C_{00} = C_{01} - C_{11}$, MR reduces to MPE:
 - ▶ e.g. if $C_{00} = C_{11} = 0$, and $C_{10} = C_{01}$

Minimum Risk - gaussian noise

- ▶ If the noise is gaussian (normal), do like for the other criteria, apply logarithm
- ▶ Obtain:

$$(r - s_0(t_0))^2 \underset{H_0}{\overset{H_1}{\gtrless}} (r - s_1(t_0))^2 + 2\sigma^2 \cdot \ln \left(\frac{(C_{10} - C_{00})p(H_0)}{(C_{01} - C_{11})p(H_1)} \right)$$

▶ or

$$r \underset{H_0}{\overset{H_1}{\gtrless}} \frac{s_0(t_0) + s_1(t_0)}{2} + \frac{\sigma^2}{s_1(t_0) - s_0(t_0)} \cdot \ln \left(\frac{(C_{10} - C_{00})p(H_0)}{(C_{01} - C_{11})p(H_1)} \right)$$

Interpretation 1: Comparing distance

- ▶ For MPE criterion, we compare the squared distances, but a supplementary term appears in favour of the most probable hypothesis:

$$(r - s_0(t_0))^2 \underset{H_0}{\overset{H_1}{\gtrless}} (r - s_1(t_0))^2 + 2\sigma^2 \cdot \ln \left(\frac{P(H_0)}{P(H_1)} \right)$$

- ▶ term depends on the ratio $\frac{P(H_0)}{P(H_1)}$
- ▶ For MR criterion, besides the probabilities we also are influenced by the costs

$$(r - s_0(t_0))^2 \underset{H_0}{\overset{H_1}{\gtrless}} (r - s_1(t_0))^2 + 2\sigma^2 \cdot \ln \left(\frac{(C_{10} - C_{00})p(H_0)}{(C_{01} - C_{11})p(H_1)} \right)$$

Interpretation 2: The threshold value

- ▶ For MPE criterion, the threshold is moved towards the less probable hypothesis:

$$r \underset{H_0}{\overset{H_1}{\gtrless}} \frac{s_0(t_0) + s_1(t_0)}{2} + \frac{\sigma^2}{s_1(t_0) - s_0(t_0)} \cdot \ln \left(\frac{P(H_0)}{P(H_1)} \right)$$

- ▶ depending on the ratio $\frac{P(H_0)}{P(H_1)}$
- ▶ For MR criterion, besides the probabilities we also are influenced by the costs

$$r \underset{H_0}{\overset{H_1}{\gtrless}} \frac{s_0(t_0) + s_1(t_0)}{2} + \frac{\sigma^2}{s_1(t_0) - s_0(t_0)} \cdot \ln \left(\frac{(C_{10} - C_{00})p(H_0)}{(C_{01} - C_{11})p(H_1)} \right)$$

Influence of costs

- ▶ The MR criterion pushes the decision towards **minimizing the high-cost scenarios**
- ▶ Example: from the equations:
 - ▶ what happens if cost C_{01} increases, while the others are unchanged?
 - ▶ what happens if cost C_{10} increases, while the others are unchanged?
 - ▶ what happens if both costs C_{01} and C_{10} increase, while the others are unchanged?

Pascal's wager

Reasoning of the French philosopher and mathematician Blaise Pascal (1623–1662):

God is, or God is not. Reason cannot decide between the two alternatives

You must wager (it is not optional)

If you gain, you gain all; if you lose, you lose nothing

Wager, then, without hesitation that He is. There is here an infinity of an infinitely happy life to gain, against a finite number of chances of loss.¹

- ▶ A philosophical example of using the Minimum Risk criterion

¹text source: Wikipedia

General form of ML, MPE and MR criteria

- ▶ ML, MPE and MR criteria all have the following form

$$\frac{w(r|H_1)}{w(r|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} K$$

- ▶ for ML: $K = 1$
- ▶ for MPE: $K = \frac{P(H_0)}{P(H_1)}$
- ▶ for MR: $K = \frac{(C_{10} - C_{00})p(H_0)}{(C_{01} - C_{11})p(H_1)}$

General form of ML, MPE and MR criteria

In gaussian noise, all criteria reduce to:

- ▶ Comparing squared distances:

$$(r - s_0(t_0))^2 \underset{H_0}{\overset{H_1}{\gtrless}} (r - s_1(t_0))^2 + 2\sigma^2 \cdot \ln(K)$$

- ▶ Comparing the sample r with a threshold T :

$$r \underset{H_0}{\overset{H_1}{\gtrless}} \underbrace{\frac{s_0(t_0) + s_1(t_0)}{2} + \frac{\sigma^2}{s_1(t_0) - s_0(t_0)} \cdot \ln(K)}_T$$

Exercise

- ▶ A vehicle airbag system detects a crash by evaluating a sensor which provides two values: $s_0(t) = 0$ (no crash) or $s_1(t) = 5$ (crashing)
- ▶ The signal is affected by gaussian noise $\mathcal{N}(\mu = 0, \sigma^2 = 1)$.
- ▶ The costs of the scenarios are: $C_{00} = 0$, $C_{01} = 100$, $C_{10} = 10$, $C_{11} = -100$
 - a. Find the decision regions R_0 and R_1 .

Neyman-Pearson criterion

- ▶ An even more general criteria than all the others until now
- ▶ **Neyman-Pearson criterion:** maximize probability of correct detection ($P(D_1 \cap H_1)$) while keeping probability of false alarms smaller then a limit ($P(D_1 \cap H_0) \leq \lambda$)
 - ▶ Deduce the threshold T from the limit condition $P(D_1 \cap H_0) = \lambda$
- ▶ ML, MPE and MR criteria are particular cases of Neyman-Pearson, for particular values of λ

Exercise

- ▶ An information source provides two messages with probabilities $p(a_0) = \frac{2}{3}$ and $p(a_1) = \frac{1}{3}$.
- ▶ The messages are encoded as constant signals with values -5 (a_0) and 5 (a_1).
- ▶ The signals are affected by noise with uniform distribution $U[-5, 5]$.
- ▶ The receiver takes one sample r .
 - a. Find the decision regions according to the Neyman-Pearson criterion, considering $P_{fa} \leq 10^{-2}$
 - b. What is the probability of correct detection, in this case?

Summary of criteria

- ▶ We have seen decision based on 1 sample r , between 2 signals (mostly)
- ▶ All decisions are based on a likelihood-ratio test

$$\frac{w(r|H_1)}{w(r|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} K$$

- ▶ Different criteria differ in the chosen value of K (likelihood threshold)
- ▶ Depending on the noise distributions, the real axis is partitioned into regions
 - ▶ region R_0 : if r is in here, decide D_0
 - ▶ region R_1 : if r is in here, decide D_1
- ▶ For gaussian noise, the boundary of the regions (threshold) is

$$T = \frac{s_0(t_0) + s_1(t_0)}{2} + \frac{\sigma^2}{s_1(t_0) - s_0(t_0)} \cdot \ln(K)$$

Comparing two decision problems

- ▶ Suppose we have a decision problem with $s_0(t) = 0$, $s_1(t) = 10$, and noise $\mathcal{N}(\mu = 0, \sigma^2 = 4)$
- ▶ Suppose we have another totally different decision problem, with $s_0(t) = 10$, $s_1(t) = 16$, and noise $\mathcal{U}[-8, 8]$
- ▶ Which one is easier? How can we compare them?
- ▶ How to evaluate the overall performance in a decision problem?
 - ▶ We need to compare the “good” probabilities (P_{cd} , P_{cr}) and the “bad” probabilities (P_{fa} , P_m)

Receiver Operating Characteristic

- ▶ A decision performance is usually represented with “**Receiver Operating Characteristic**” (ROC) graph
- ▶ It is a graph of $P_d = P(D_1|H_1)$ as a function of $P_{fa} = P(D_1|H_0)$,
 - ▶ obtained for different values of the threshold value T
 - ▶ i.e. for every T you get a certain value of P_{fa} and a certain value of P_d

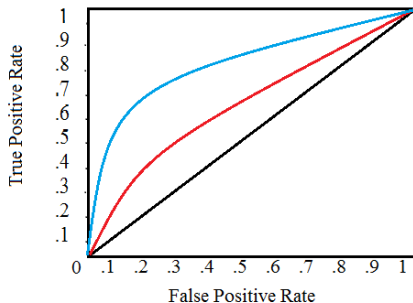


Figure 3: Sample ROC curves

Receiver Operating Characteristic

- ▶ ROC graph shows there is always a **tradeoff** between good P_d and bad P_{fa}
 - ▶ to increase P_d one must also increase P_{fa}
 - ▶ if we want to make sure we don't miss any real detections (increase P_d), we pay by increasing the chances of false alarms
- ▶ Different criteria = different likelihood thresholds K = different points on the graph = different tradeoffs
- ▶ An overall performance measure is the total **Area Under the Curve** (AUC)
 - ▶ this doesn't depend on the choice of a particular threshold value
- ▶ We can compare two different decision situations (e.g. different signals, or different algorithms etc) by plotting their ROC and comparing their AUC

The Precision-Recall curve

- ▶ A similar curve is the **Precision vs. Recall** curve

- ▶ **Precision** = $\frac{P(D_1 \cap H_1)}{P(D_1 \cap H_1) + P(D_1 \cap H_0)}$

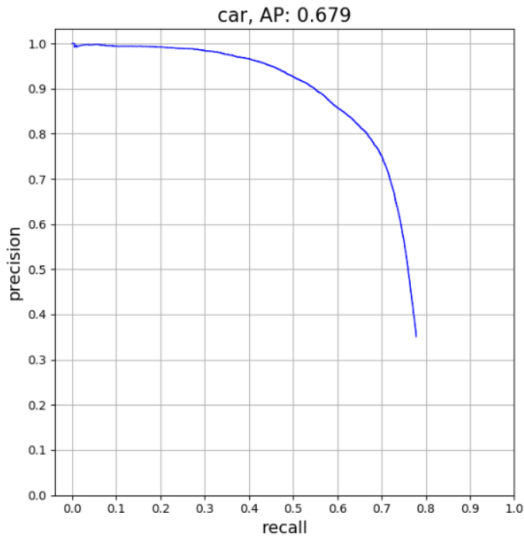
- ▶ = True Positives / (True Positives + False Positives)

- ▶ **Recall** = $\frac{P(D_1 \cap H_1)}{P(D_1 \cap H_1) + P(D_0 \cap H_1)} = P(D_1 | H_1)$

- ▶ = True Positives / (True Positives + False Negatives)

Precision-Recall curve

Example of a Precision vs Recall Curve



Precision-Recall curve

Real-life app from which the preceding curve was taken:



Signal-to-Noise Ratio

- ▶ How to improve the detection performance?
 - ▶ i.e. increase P_D while keeping P_{fa} the same
 - ▶ irrespective of what threshold is chosen
- ▶ Two solutions:
 - ▶ Increase the separation between $s_0(t)$ and $s_1(t)$ (increase **signal power**)
 - ▶ Reduce the noise (decrease **noise power**)
 - ▶ i.e. increase **Signal-to-Noise ratio**

- ▶ 2020-2021 Exam: Skip next 3 slides (until Signal-to-noise ratio)

Performance of likelihood-ratio decoding in AWGN

- ▶ WGN = “White Gaussian Noise”
- ▶ Assume equal probabilities $P(H_0) = P(H_1) = \frac{1}{2}$
 - ▶ Equivalently, consider only the conditional probabilities
- ▶ All decisions are based on a likelihood-ratio test

$$\frac{w(r|H_1)}{w(r|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} K$$

- ▶ Conditional probability of correct detection is:

$$\begin{aligned} P_d &= P(D_1|H_1) \\ &= \int_T^\infty w(r|H_1) \\ &= (F(\infty) - F(T)) \\ &= \frac{1}{2} \left(1 - \operatorname{erf} \left(\frac{T - s_1(t_0)}{\sqrt{2}\sigma} \right) \right) \\ &= Q \left(\frac{T - s_1(t_0)}{\sqrt{2}\sigma} \right) \end{aligned}$$

Performance of likelihood-ratio decoding in AWGN

- Conditional probability of false alarm is:

$$\begin{aligned}P_{fa} &= P(D_1|H_0) \\&= \int_T^\infty w(r|H_0) \\&= (F(\infty) - F(T)) \\&= \frac{1}{2} \left(1 - \operatorname{erf} \left(\frac{T - s_0(t_0)}{\sqrt{2}\sigma} \right) \right) \\&= Q \left(\frac{T - s_0(t_0)}{\sqrt{2}\sigma} \right)\end{aligned}$$

- Therefore $\frac{T - s_0(t_0)}{\sqrt{2}\sigma} = Q^{-1}(P_{fa})$,
- And: $\frac{T - s_1(t_0)}{\sqrt{2}\sigma} = Q^{-1}(P_{fa}) + \frac{s_0(t_0) - s_1(t_0)}{\sqrt{2}\sigma}$

Performance of likelihood-ratio decoding in AWGN

- ▶ Replacing in P_d yields:

$$P_d = Q \left(\underbrace{Q^{-1}(P_{fa})}_{\text{constant}} + \frac{s_0(t_0) - s_1(t_0)}{\sqrt{2}\sigma} \right)$$

- ▶ Consider a simple case:

- ▶ $s_0(t_0) = 0$
- ▶ $s_1(t_0) = A = \text{constant}$

- ▶ We get:

$$P_d = Q \left(\underbrace{Q^{-1}(P_{fa})}_{\text{constant}} - \frac{A}{\sqrt{2}\sigma} \right)$$

Signal-to-noise ratio

- ▶ **Signal-to-noise ratio (SNR)** = $\frac{\text{power of original signal}}{\text{power of noise}}$
- ▶ Average power of a signal = average squared value = $\overline{X^2}$
 - ▶ Original signal power of $s(t)$ is $\frac{A^2}{2}$
 - ▶ Noise power is $\overline{X^2} = \sigma^2$ (when noise mean value $\mu = 0$)
- ▶ In our case, $\text{SNR} = \frac{A^2}{2\sigma^2}$

$$P_d = Q \left(\underbrace{Q^{-1}(P_{fa})}_{\text{constant}} - \sqrt{\text{SNR}} \right)$$

- ▶ For a fixed P_{fa} , P_d **increases with SNR**
 - ▶ Q is a monotonic decreasing function

Performance depends on SNR

- ▶ Receiver performance increases with SNR increase
 - ▶ high SNR: good performance
 - ▶ poor SNR: bad performance

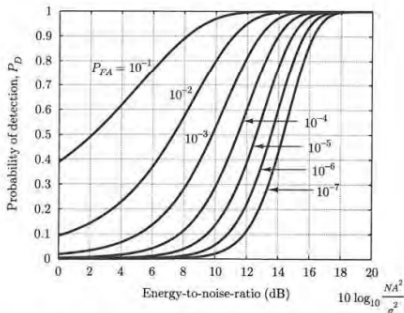


Figure 6: Detection performance depends on SNR

[source: *Fundamentals of Statistical Signal Processing*, Steven Kay]

Applications of decision theory

- ▶ Can we apply these decision criteria in other engineering problems?
 - ▶ e.g. not for deciding between two signals, but for something else
- ▶ The core mathematical problem we solve is:
 - ▶ we have 2 (or more) possible distributions
 - ▶ we observe 1 value
 - ▶ we determine the most likely distribution, according to the value
- ▶ In our particular problem, we decide between two signals
- ▶ But this can be applied to many other statistical problems:
 - ▶ medicine: does this ECG signal look healthy or not?
 - ▶ business: will this client buy something or not?
 - ▶ Typically we use more than 1 value for these, but the mathematical principle is the same

Applications of decision theory

Example (purely imaginary):

- ▶ A healthy person of weight = X kg has the concentration of thrombocytes per ml of blood distributed approximately as $\mathcal{N}(\mu = 10 \cdot X, \sigma^2 = 20)$.
- ▶ A person suffering from disease D has a much lower value of thrombocytes, distributed approximately as $\mathcal{N}(100, \sigma^2 = 10)$.
- ▶ The lab measures your blood and finds your value equal to $r = 255$. Your weight is 70 kg.
- ▶ Decide: are you most likely healthy, or ill?

II.3 Signal detection with multiple samples

Multiple samples from a signal

- ▶ The overall context stays the same:
 - ▶ A signal $s(t)$ is transmitted
 - ▶ There are **two hypotheses**:
 - ▶ H_0 : true signal is $s(t) = s_0(t)$
 - ▶ H_1 : true signal is $s(t) = s_1(t)$
 - ▶ Receiver can take **two decisions**:
 - ▶ D_0 : receiver decides that signal was $s(t) = s_0(t)$
 - ▶ D_1 : receiver decides that signal was $s(t) = s_1(t)$
- ▶ There 4 possible outcomes

Multiple samples from a signal

- ▶ The overall context stays the same:
 - ▶ There is noise on the channel (unknown)
 - ▶ The receiver receives $r(t) = s(t) + n(t)$
- ▶ Suppose we take N samples from $r(t)$, not just 1
 - ▶ Each sample is $r_i = r(t_i)$, taken at moment t_i
- ▶ The samples are arranged in a **sample vector**

$$\mathbf{r} = [r_1, r_2, \dots, r_N]$$

Multiple samples from a signal

- ▶ Each sample r_i is a **random variable**
 - ▶ since $r(t_i) = s(t_i) + n(t_i) = \text{a constant} + \text{a random variable}$
- ▶ The sample vector \mathbf{r} is a set of N random variables from a random process
- ▶ Considering the whole sample vector \mathbf{r} as a whole, the values of \mathbf{r} are described by the **distributions of order N**
- ▶ In hypothesis H_0 :

$$w_N(\mathbf{r}|H_0) = w_N(r_1, r_2, \dots, r_N|H_0)$$

- ▶ In hypothesis H_1 :

$$w_N(\mathbf{r}|H_1) = w_N(r_1, r_2, \dots, r_N|H_1)$$

Likelihood of vector samples

- ▶ We can apply **the same criteria** based on likelihood ratio as for 1 sample

$$\frac{w_N(\mathbf{r}|H_1)}{w_N(\mathbf{r}|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} K$$

- ▶ Notes:
 - ▶ \mathbf{r} is a vector; we consider the likelihood of all the sample vector as a whole
 - ▶ $w_N(\mathbf{r}|H_0)$ = likelihood of the whole vector \mathbf{r} being obtained in hypothesis H_0
 - ▶ $w_N(\mathbf{r}|H_1)$ = likelihood of the whole vector \mathbf{r} being obtained in hypothesis H_1
 - ▶ the value of K is given by the actual decision criterion used
- ▶ Interpretation: we choose the hypothesis that is most likely to have produced the observed data
 - ▶ now the data = a set of samples, not just 1

Separation

- ▶ Assuming the noise is white noise, the noise samples are independent, and therefore the samples r_i are independent
- ▶ In that case the joint distribution $w_N(\mathbf{r}|H_i)$ can be decomposed as a **product of individual distributions**:

$$w_N(\mathbf{r}|H_i) = w(r_1|H_i) \cdot w(r_2|H_i) \cdot \dots \cdot w(r_N|H_i)$$

- ▶ e.g. the likelihood of obtaining $[5.1, 4.7, 4.9] =$ likelihood of obtaining $5.1 \times$ likelihood of getting $4.7 \times$ likelihood of getting 4.9
- ▶ The $w(r_i|H_i)$ are just conditional distributions for each sample
 - ▶ we've seen them already

- ▶ Then all likelihood ratio criteria can be written as:

$$\frac{w_N(\mathbf{r}|H_1)}{w_N(\mathbf{r}|H_0)} = \frac{w(r_1|H_1)}{w(r_1|H_0)} \cdot \frac{w(r_2|H_1)}{w(r_2|H_0)} \cdots \frac{w(r_N|H_1)}{w(r_N|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} K$$

- ▶ The likelihood ratio of a vector of samples = product of likelihood ratio for each sample
- ▶ We **multiply** the likelihood ratio **of each sample**, and then use the same criteria for the end result

Criteria for decisions

- ▶ All likelihood ratio criteria can be written as:

$$\frac{w_N(\mathbf{r}|H_1)}{w_N(\mathbf{r}|H_0)} = \frac{w(r_1|H_1)}{w(r_1|H_0)} \cdot \frac{w(r_2|H_1)}{w(r_2|H_0)} \cdots \frac{w(r_N|H_1)}{w(r_N|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} K$$

- ▶ The value of K is the same as for 1 sample:
 - ▶ for ML: $K = 1$
 - ▶ for MPE: $K = \frac{P(H_0)}{P(H_1)}$
 - ▶ for MR: $K = \frac{(C_{10} - C_{00})p(H_0)}{(C_{01} - C_{11})p(H_1)}$

Particular case: AWGN

► AWGN = “Additive White Gaussian Noise”

► In hypothesis H_1 : $w(r_i|H_1) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(r_i-s_1(t_i))^2}{2\sigma^2}}$

► In hypothesis H_0 : $w(r_i|H_0) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(r_i-s_0(t_i))^2}{2\sigma^2}}$

► Likelihood ratio for vector \mathbf{r}

$$\frac{w_N(\mathbf{r}|H_1)}{w_N(\mathbf{r}|H_0)} = \frac{e^{-\frac{\sum (r_i-s_1(t_i))^2}{2\sigma^2}}}{e^{-\frac{\sum (r_i-s_0(t_i))^2}{2\sigma^2}}} = e^{\frac{\sum (r_i-s_0(t_i))^2 - \sum (r_i-s_1(t_i))^2}{2\sigma^2}}$$

Decision criteria for AWGN

- ▶ The global likelihood ratio is compared with K :

$$\frac{w_N(\mathbf{r}|H_1)}{w_N(\mathbf{r}|H_0)} = e^{\frac{\sum (r_i - s_0(t_i))^2 - \sum (r_i - s_1(t_i))^2}{2\sigma^2}} \underset{H_0}{\overset{H_1}{\gtrless}} K$$

- ▶ Applying the natural logarithm, this becomes:

$$\sum (r_i - s_0(t_i))^2 \underset{H_0}{\overset{H_1}{\gtrless}} \sum (r_i - s_1(t_i))^2 + 2\sigma^2 \ln(K)$$

Interpretation 1: geometrical distance

- ▶ The sums are squared **geometrical distances**:

$$\sum (r_i - s_1(t_i))^2 = \|\mathbf{r} - \mathbf{s}_1(\mathbf{t})\|^2 = d(\mathbf{r}, s_1(t))^2$$

$$\sum (r_i - s_0(t_i))^2 = \|\mathbf{r} - \mathbf{s}_0(\mathbf{t})\|^2 = d(\mathbf{r}, s_0(t))^2$$

- ▶ the distance between the observed samples \mathbf{r} and the true possible underlying signals $s_1(t)$ and $s_0(t)$
- ▶ with N samples \Rightarrow distance between vectors of size N
- ▶ It comes down to a decision between distances

Interpretation 1: geometrical distance

- ▶ Maximum Likelihood criterion:
 - ▶ $K = 1$, $\ln(K) = 0$
 - ▶ we choose the **minimum distance** between what is (\mathbf{r}) and what should have been in absence of noise ($s_1(t)$ and $s_0(t)$)
 - ▶ hence the name “minimum distance receiver”
- ▶ Minimum Probability of Error criterion:
 - ▶ $K = \frac{P(H_0)}{P(H_1)}$
 - ▶ An additional term appears in favor of the most probable hypothesis
- ▶ Minimum Risk criterion:
 - ▶ $K = \frac{(C_{10} - C_{00})p(H_0)}{(C_{01} - C_{11})p(H_1)}$
 - ▶ Additional term depends on both probabilities and costs

Exercise

Exercise:

- ▶ A signal can have two values, 0 (hypothesis H_0) or 6 (hypothesis H_1). The signal is affected by AWGN $\mathcal{N}(0, \sigma^2 = 1)$. The receiver takes 5 samples with values $\{1.1, 4.4, 3.7, 4.1, 3.8\}$.
 - a. What is decision according to Maximum Likelihood criterion?
 - b. What is decision according to Minimum Probability of Error criterion, assuming $P(H_0) = 2/3$ and $P(H_1) = 1/3$?
 - c. What is the decision according to Minimum Risk Criterion, assuming $P(H_0) = 2/3$ and $P(H_1) = 1/3$, and $C_{00} = 0$, $C_{10} = 10$, $C_{01} = 20$, $C_{11} = 5$?

Another exercise

Another Exercise:

- ▶ Consider detecting a signal $s_1(t) = 3 \sin(2\pi f_1 t)$ that can be present (hypothesis H_1) or not ($s_0(t) = 0$, hypothesis H_0). The signal is affected by AWGN $\mathcal{N}(0, \sigma^2 = 1)$. The receiver takes 2 samples.
 - a. What are the best sample times t_1 and t_2 to maximize detection performance?
 - b. The receiver takes 2 samples with values $\{1.1, 4.4\}$, at sample times $t_1 = \frac{0.125}{f_1}$ and $t_2 = \frac{0.625}{f_1}$. What is decision according to Maximum Likelihood criterion?
 - c. What if we take the decision with Minimum Probability of Error criterion, assuming $P(H_0) = 2/3$ and $P(H_1) = 1/3$?
 - d. What is the decision according to Minimum Risk Criterion, assuming $P(H_0) = 2/3$ and $P(H_1) = 1/3$, and $C_{00} = 0$, $C_{10} = 10$, $C_{01} = 20$, $C_{11} = 5$?
 - e. What if the receiver takes an extra third sample at time $t_3 = \frac{0.5}{f_1}$. Will the detection be improved?

Interpretation 2: inner-product

- ▶ Let's decompose the parentheses in the distances:

$$\sum (r_i - s_0(t_i))^2 \underset{H_0}{\overset{H_1}{\gtrless}} \sum (r_i - s_1(t_i))^2 + 2\sigma^2 \ln(K)$$

- ▶ Equivalent to:

$$\begin{aligned} \sum (r_i)^2 + \sum s_0(t_i)^2 - 2 \sum r_i s_0(t_i) &\underset{H_0}{\overset{H_1}{\gtrless}} \sum (r_i)^2 + \\ &+ \sum s_1(t_i)^2 - 2 \sum r_i s_1(t_i) + 2\sigma^2 \ln(K) \end{aligned}$$

- ▶ Equivalent to:

$$\sum r_i s_1(t_i) - \frac{\sum (s_1(t_i))^2}{2} \underset{H_0}{\overset{H_1}{\gtrless}} \sum r_i s_0(t_i) - \frac{\sum (s_0(t_i))^2}{2} + \sigma^2 \ln(K)$$

Interpretation 2: inner-product

- ▶ Linear algebra: **inner product** of vectors **a** and **b**:

$$\langle a, b \rangle = \sum_i a_i b_i$$

- ▶ $\sum r_i s_1(t_i) = \langle \mathbf{r}, \mathbf{s}_1(\mathbf{t}) \rangle$ is the inner product of vector $\mathbf{r} = [r_1, r_2, \dots, r_N]$ with $\mathbf{s}_1(\mathbf{t}) = [s_1(t_1), s_1(t_2), \dots, s_1(t_N)]$
- ▶ $\sum r_i s_0(t_i) = \langle \mathbf{r}, \mathbf{s}_0(\mathbf{t}) \rangle$ is the inner product of vector $\mathbf{r} = [r_1, r_2, \dots, r_N]$ with $\mathbf{s}_0(\mathbf{t}) = [s_0(t_1), s_0(t_2), \dots, s_0(t_N)]$
- ▶ $\sum (s_1(t_i))^2 = \sum s_1(t_i) \cdot s_1(t_i) = \langle \mathbf{s}_1(\mathbf{t}), \mathbf{s}_1(\mathbf{t}) \rangle = E_1$ is the **energy** of vector $s_1(t)$
- ▶ $\sum (s_0(t_i))^2 = \sum s_0(t_i) \cdot s_0(t_i) = \langle \mathbf{s}_0(\mathbf{t}), \mathbf{s}_0(\mathbf{t}) \rangle = E_0$ is the **energy** of vector $s_0(t)$

Interpretation 2: inner-product

- ▶ The decision can be rewritten as:

$$\langle \mathbf{r}, \mathbf{s}_1 \rangle - \frac{E_1}{2} \underset{H_0}{\overset{H_1}{\gtrless}} \langle \mathbf{r}, \mathbf{s}_0 \rangle - \frac{E_0}{2} + \sigma^2 \ln(K)$$

- ▶ Interpretation: we **compare the inner-products**
 - ▶ also subtract the energies of the signals, for a fair comparison
 - ▶ also with a term depending on the criterion

Interpretation 2: inner-product

- ▶ Particular case:

- ▶ If the two signals have the same energy:

$$E_1 = \sum s_1(t_i)^2 = E_0 = \sum s_0(t_i)^2$$

- ▶ Examples:

- ▶ BPSK modulation: $s_1 = A \cos(2\pi ft)$, $s_0 = -A \cos(2\pi ft)$

- ▶ 4-PSK modulation: $s_{n=0,1,2,3} = A \cos(2\pi ft + n\frac{\pi}{4})$

- ▶ Then it is simplified as:

$$\langle \mathbf{r}, \mathbf{s}_1 \rangle \underset{H_0}{\overset{H_1}{\geq}} \langle \mathbf{r}, \mathbf{s}_0 \rangle + \sigma^2 \ln(K)$$

Interpretation 2: inner-product

- ▶ Inner-product in signal processing measures **similarity** of two signals
- ▶ Interpretation: we check if the received samples **r** look **more similar** to $s_1(t)$ or to $s_0(t)$
 - ▶ Choose the one which shows more similarity to **r**
 - ▶ There is also the subtraction of the energies, for a fair comparison (due to mathematical reasons)
- ▶ **Inner product** of vectors **a** and **b**:

$$\langle a, b \rangle = \sum_i a_i b_i$$

Decision with correlator circuits

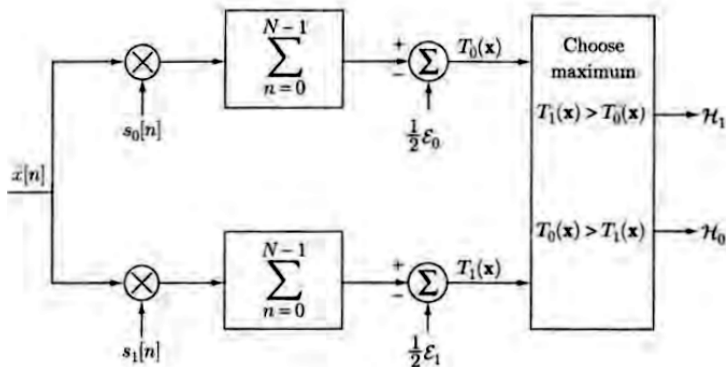


Figure 7: Decision between two signals

[source: *Fundamentals of Statistical Signal Processing*, Steven Kay]

Matched filters

- ▶ How to compute the inner product of two signals $r[n]$ and $s[n]$ of length N ?

$$\langle \mathbf{r}, \mathbf{s} \rangle = \sum r_i s(t_i)$$

- ▶ Let $h[n]$ be the signal $s[n]$ **flipped** / **mirrored** (“oglidit”) and delayed with N
 - ▶ starts from time 0, goes up to time $N - 1$, but backwards

$$h[n] = s[N - 1 - n]$$

- ▶ Example:

- ▶ if $s[n] = [1, 2, 3, 4, 5, 6]$
 - ▶ then $h[n] = s[N - 1 - n] = [6, 5, 4, 3, 2, 1]$

Matched filters

- ▶ The convolution of $r[n]$ with $h[n]$ is

$$y[n] = \sum_k r[k]h[n - k] = \sum_k r[k]s[N - 1 - n + k]$$

- ▶ The convolution sampled at the end of the signal, $y[N - 1]$ (for $n = N - 1$), is the inner product:

$$y[N - 1] = \sum_k r[k]s[k]$$

Matched filters

- ▶ To detect a signal $s[n]$ we can use a **filter with impulse response = mirrored version of $s[n]$** , and take the final sample of the output

$$h[n] = s[N - 1 - n]$$

- ▶ it is identical to computing the inner product
- ▶ **Matched filter** = a filter designed to have the impulse response the flipped version of a signal we search for
 - ▶ the filter is *matched* to the signal we want to detect
 - ▶ rom. “filtru adaptat”

Signal detection with matched filters

- ▶ Use one filter matched to signal $s_1(t_i)$
- ▶ Use another filter matched to signal $s_0(t_i)$
- ▶ Sample both filters at the end of the signal $n = N - 1$
 - ▶ obtain the values of the inner products
- ▶ Use the decision rule (with the inner products) to decide

Signal detection with matched filters

- ▶ In case $s_0(t) = 0$, we need only one matched filter for $s_1(t)$, and compare the result to a threshold

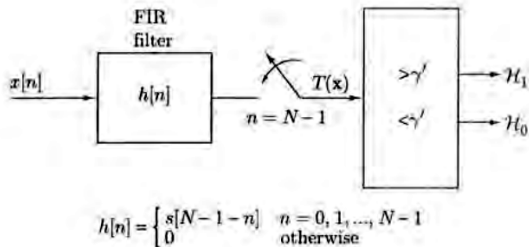


Figure 8: Signal detection with matched filter

[source: *Fundamentals of Statistical Signal Processing*, Steven Kay]

II.4 Detection of general signals with continuous observations

Continuous observation of a general signal

- ▶ Continuous observation = we don't take samples anymore, we use **all the continuous signal**
 - ▶ like taking N samples but with $N \rightarrow \infty$
- ▶ Original signals are $s_0(t)$ and $s_1(t)$
- ▶ Signals are affected by noise
 - ▶ Assume **only Gaussian noise**, for simplicity
- ▶ Received signal is $r(t)$

Euclidian space

- ▶ Extend from N samples to the case a full continuous signal
- ▶ Each signal $r(t)$, $s_1(t)$ or $s_0(t)$ is a data point in an **infinite-dimensional Euclidean space**
- ▶ **Distance** between two signals is:

$$d(\mathbf{r}, \mathbf{s}) = \sqrt{\int (r(t) - s(t))^2 dt}$$

- ▶ **Inner product** between two signals is:

$$\langle \mathbf{r}, \mathbf{s} \rangle = \int r(t)s(t)dt$$

- ▶ Similar with the N dimensional case, but with integral instead of sum

Decision rule for AWGN: distances

- ▶ For AWGN, same decision rule as always:

$$d(\mathbf{r}, \mathbf{s}_0)^2 \underset{H_0}{\overset{H_1}{\geq}} d(\mathbf{r}, \mathbf{s}_1)^2 + 2\sigma^2 \ln(K)$$

- ▶ Distance = previous formula, with integral
- ▶ Same criteria:
 - ▶ Maximum Likelihood criterion: $K = 1$, $\ln(K) = 0$
 - ▶ we choose the **minimum distance**
 - ▶ Minimum Probability of Error criterion: $K = \frac{P(H_0)}{P(H_1)}$
 - ▶ Minimum Risk criterion: $K = \frac{(C_{10} - C_{00})p(H_0)}{(C_{01} - C_{11})p(H_1)}$

Decision rule for AWGN: inner products

- ▶ For AWGN, same decision rule as always:

$$\langle \mathbf{r}, \mathbf{s}_1 \rangle - \frac{E_1}{2} \underset{H_0}{\overset{H_1}{\geq}} \langle \mathbf{r}, \mathbf{s}_0 \rangle - \frac{E_0}{2} + \sigma^2 \ln(K)$$

- ▶ Inner product = previous formula, with integral
- ▶ All interpretations remain the same
 - ▶ we only change the **type of signal** we work with

Matched filters

- ▶ Inner product of signals can be computed with **matched filters**
- ▶ **Matched filter** = a filter designed to have the impulse response the flipped version of a signal we search for
 - ▶ if original signal $s(t)$ has length T
 - ▶ then $h(t) = s(T - t)$
 - ▶ filter is analogical, impulse response is continuous
- ▶ Output of a matched filter at time $t = T$ is equal to the inner product of the input $r(t)$ with $s(t)$

Signal detection with matched filters

- ▶ Use one filter matched to signal $s_1(t)$
- ▶ Use another filter matched to signal $s_0(t)$
- ▶ Sample both filters at the end of the signal $t = T$
 - ▶ obtain the values of the inner products
- ▶ Use the decision rule (with the inner products) to decide

Review of Euclidean vector spaces

- ▶ Review of Euclidean vector spaces
- ▶ Vector space
 - ▶ one thing + another thing = still in same space
 - ▶ constant \times a vector = still in same space
 - ▶ has basic arithmetic: sum, multiplication by a constant
 - ▶ Examples:
 - ▶ 1D = a line
 - ▶ 2D = a plane
 - ▶ 3D = a 3-D space
 - ▶ N-D = ...
 - ▶ ∞ -D = ..

Review of Euclidean vector spaces

- ▶ The fundamental function: **inner product**

- ▶ for discrete signals

$$\langle \mathbf{x}, \mathbf{y} \rangle = \sum_i x_i y_i$$

- ▶ for continuous signals

$$\langle \mathbf{x}, \mathbf{y} \rangle = \int x(t)y(t)$$

- ▶ Norm (length) of a vector = sqrt(inner product with itself)

$$\|\mathbf{x}\| = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle}$$

- ▶ Distance between two vectors = norm of their difference

$$d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|$$

Review of Euclidean vector spaces

- ▶ Energy of a signal = squared norm

$$E_x = \|x\|^2 = \langle x, x \rangle$$

- ▶ Angle between two vectors

$$\cos(\alpha) = \frac{\langle x, y \rangle}{\|x\| \cdot \|y\|}$$

- ▶ value between -1 and 1
- ▶ if $\langle x, y \rangle = 0$, the two vectors are **orthogonal** (perpendicular)

Review of Euclidean vector spaces

- ▶ Bonus: the Fourier transform = inner product with $e^{j\omega t}$

$$\mathcal{F}\{x(t)\} = \langle x(t), e^{j\omega t} \rangle = \int x(t) e^{-j\omega t}$$

- ▶ for complex signals, the second function is conjugated, hence $-j$ instead of j

$$\langle \mathbf{x}, \mathbf{y} \rangle = \sum_i x_i y_i^*$$

$$\langle \mathbf{x}, \mathbf{y} \rangle = \int x(t) y(t)^*$$

- ▶ Also same for discrete signals

Review of Euclidean vector spaces

- ▶ Conclusion: expressing algorithms in a generic way, with inner products / distances / norms, is very powerful
 - ▶ they automatically apply to all vector spaces
 - ▶ work once, reuse in many places

II.5 Decision with unknown distributions

Knowing vs not knowing the distribution

- ▶ Until now, we always knew what samples we expect
 - ▶ We knew the signals:
 - ▶ $s_0(t) = \dots$
 - ▶ $s_1(t) = \dots$
 - ▶ We knew the noise type
 - ▶ gaussian, uniform, etc.
 - ▶ So we knew the sample distributions:
 - ▶ $w(r|H_0) = \dots$
 - ▶ $w(r|H_1) = \dots$
- ▶ In real life, things are more complicated

Typical example

- ▶ What if the signals $s_0(t)$ and $s_1(t)$ do not exist / we do not know them?
- ▶ Example: face recognition
 - ▶ Task: identify person A vs B based on a face image
 - ▶ We have:
 - ▶ 100 images of person A, in various conditions
 - ▶ 100 images of person B, in various conditions

Samples vs distributions

- ▶ Compare face recognition with our previous signal detection
- ▶ We still have:
 - ▶ two hypotheses H_0 (person A) and H_1 (person B)
 - ▶ a sample vector \mathbf{r} = the test image we need to decide upon
 - ▶ we can take two decisions
 - ▶ 4 scenarios: correct rejection, false alarm, miss, correct detection
- ▶ What's different? We don't have formulas
 - ▶ there is no "true" data described by formulas $s_0(t) = \dots$ and $s_1(t) \dots$
 - ▶ (faces of persons A and B are not signals)
 - ▶ instead, we have lots of examples of each distribution
 - ▶ 100 images of A = examples of \mathbf{r} might look in hypothesis H_0
 - ▶ 100 images of B = examples of \mathbf{r} might look in hypothesis H_1

Machine learning terminology

- ▶ Terminology used in **machine learning**:
 - ▶ This kind of problem = signal **classification** problem
 - ▶ given one data vector, specify which class it belongs to
 - ▶ The **classes** = the two categories, hypotheses H_i , persons A/B etc
 - ▶ A **training set** = a set of known data
 - ▶ e.g. our 100 images of each person
 - ▶ it will be used in the decision process
 - ▶ Signal **label** = the class of a signal

Samples vs distributions

- ▶ The training set gives us the same information as the conditional distributions $w(r|H_0)$ and $w(r|H_1)$
 - ▶ $w(r|H_0)$ tells us how r looks like in hypothesis H_0
 - ▶ $w(r|H_1)$ tells us how r looks like in hypothesis H_1
 - ▶ the training set shows the same thing, without formulas, but via many examples
- ▶ OK, so how to classify the data in these conditions?

The k-NN algorithm

The k-Nearest Neighbours algorithm (k-NN)

► Input:

- a labelled training set of vectors $\mathbf{x}_1 \dots \mathbf{x}_N$, from L possible classes $C_1 \dots C_L$
- a test vector \mathbf{r} we need to classify
- a parameter k

1. Compute distance from \mathbf{r} to each training vector \mathbf{x}_i

- can use same Euclidean distance we used for signal detection with multiple samples

2. Choose the closest k vectors to \mathbf{r} (the k nearest neighbours)

3. Determine class of \mathbf{r} = the majority class among the k nearest neighbours

► Output: the class of \mathbf{r}

The k-NN algorithm

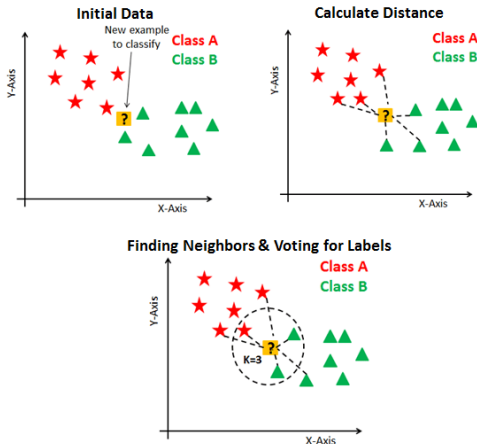


Figure 9: The k-NN algorithm illustrated [1]

[1] image from "KNN Classification using Scikit-learn", Avinash Navlani,

<https://www.datacamp.com/community/tutorials/k-nearest-neighbor-classification-scikit-learn>

k-NN and ML decision

- ▶ If the training set is very large, the k-NN algorithm is a kind of ML decision
- ▶ The number of samples of a class in the vicinity of our point is proportional to $w(r|H_i)$
- ▶ More neighbors of class A than B $\Leftrightarrow w(r|H_A) > w(r|H_B)$

k-NN and ML decision

- ▶ Example: leaves and trees

Exercise

Exercise

1. Consider the k-NN algorithm with the following training set, composed of 5 vectors of class A and another 5 vectors from class B:

► Class A:

$$\mathbf{v}_1 = \begin{bmatrix} 1 \\ -2 \end{bmatrix} \quad \mathbf{v}_2 = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \quad \mathbf{v}_3 = \begin{bmatrix} -4 \\ 2 \end{bmatrix} \quad \mathbf{v}_4 = \begin{bmatrix} 2 \\ 1 \end{bmatrix} \quad \mathbf{v}_5 = \begin{bmatrix} -2 \\ -2 \end{bmatrix}$$

► Class B:

$$\mathbf{v}_6 = \begin{bmatrix} 7 \\ 0 \end{bmatrix} \quad \mathbf{v}_7 = \begin{bmatrix} 2 \\ 3 \end{bmatrix} \quad \mathbf{v}_8 = \begin{bmatrix} 3 \\ 2 \end{bmatrix} \quad \mathbf{v}_9 = \begin{bmatrix} -3 \\ 8 \end{bmatrix} \quad \mathbf{v}_{10} = \begin{bmatrix} -2 \\ 5 \end{bmatrix}$$

Compute the class of the vector $\mathbf{x} = \begin{bmatrix} -3 \\ 6 \end{bmatrix}$ using the k-NN algorithm, with $k = 1$, $k = 3$, $k = 5$, $k = 7$ and $k = 9$

Discussion

- ▶ k-NN is a supervised learning algorithm
 - ▶ training data needs to be labelled
- ▶ Effect of k is to smooth the decision boundary:
 - ▶ small k : lots of edges
 - ▶ large k : smooth boundary
- ▶ How to find k ?

Cross-validation

- ▶ How to find a good value for k ?
 - ▶ by trial and error (“băbește”)
- ▶ **Cross-validation** = use a small testing set for checking what parameter value is best
 - ▶ this data set is known as **cross-validation set**
 - ▶ use $k = 1$, test with cross-validation set and see how many vectors are classified correctly
 - ▶ repeat for $k = 2, 3, \dots, \max$
 - ▶ choose value of k with best results on the cross-validation set

Evaluating algorithms

- ▶ How to evaluate the performance of k-NN?
 - ▶ Use a testing set to test the algorithm, check the percentage of correct classification
- ▶ Final testing set should be different from the cross-validation set
 - ▶ For final testing, use data that the algorithm has never seen, for fairness
- ▶ How to split the data into datasets?

Datasets

- ▶ Suppose you have 200 face images, 100 images of person A and 100 of person B
- ▶ Split the data into:
 - ▶ Training set
 - ▶ data that shall be used by the algorithm
 - ▶ largest part (about 60% of the whole data)
 - ▶ i.e. 60 images of person A and 60 images of B
 - ▶ Cross-validation set
 - ▶ used to test the algorithm and choose best value of parameters (k)
 - ▶ smaller, about 20%, e.g. 20 images of A and 20 images of B
 - ▶ Testing set
 - ▶ used to evaluate the final algorithm, with all parameters set to a final value
 - ▶ smaller, about 20%, e.g. 20 images of A and 20 images of B

The k-Means algorithm

- ▶ k-Means: an algorithm for data **clustering**
 - ▶ identifying groups of close vectors in data
- ▶ Is an example of unsupervised learning algorithm
 - ▶ “unsupervised learning” = we don’t know the data classes of the signals beforehand

The k-Means algorithm

The k-Means algorithm

- ▶ Input:

- ▶ unlabelled training set of vectors $\mathbf{x}_1 \dots \mathbf{x}_N$
- ▶ number of classes C

- ▶ Initialization: randomly initialize the C centroids

$$\mathbf{c}_i \leftarrow \text{random values}$$

- ▶ Repeat

1. Classification: assign each data \mathbf{x} to the nearest centroid \mathbf{c}_i :

$$l_n = \arg \min_i d(\mathbf{x}, \mathbf{c}_i), \forall \mathbf{x}$$

2. Update: update each centroids $\mathbf{c}_i = \text{average of the } \mathbf{x} \text{ assigned to } \mathbf{c}_i$

$$\mathbf{c}_i \leftarrow \text{average of } \mathbf{x}, \forall \mathbf{x} \text{ in class } i$$

- ▶ Output: return the centroids \mathbf{c}_i , the labels l_i of the input data \mathbf{x}_i

The k-Means algorithm

Video explanations of the k-Means algorithm:

- ▶ Watch this, starting from time 6:28 to 7:08

<https://www.youtube.com/watch?v=4b5d3muPQmA>

- ▶ Watch this, starting from time 3:05 to end

<https://www.youtube.com/watch?v=luRb3y8qKX4>

The k-Means algorithm

- ▶ Not guaranteed that k-Means identifies good clusters
 - ▶ results depend on the random initialization of centroids
 - ▶ repeat many times, choose best result
 - ▶ smart initializations are possible (*k-Means++*)

Exercise

Exercise

1. Consider the following data

$$\{\mathbf{v}_n\} = [1.3, -0.1, 0.5, 4.7, 5.1, 5.8, 0.4, 4.8, -0.7, 4.9]$$

Use the k-Means algorithm to find the two centroids \mathbf{c}_1 and \mathbf{c}_2 , starting from two random values $\mathbf{c}_1 = -0.5$ and $\mathbf{c}_2 = 0.9$. Perform 5 iterations of the algorithm.