Department of Electronic Systems

Examination paper for TTT4275 Estimation, Detection and Classification

Academic contact during examination:: Magne Hallstein Johnsen
Phone: 93025534

Examination date: Date: Wednesday May 16th, 2018

Examination time (from - to):

Permitted examination support material: –

Language: English

Number of pages (front page excluded): 4

Number of pages enclosed:

Informasjon om trykking av eksamensoppgave			
Originalen er :			
1-sidig	☐	**2-sidig**	☐
sort/hvit	☐	**farger**	☐
skal ha flervalgsskjema			☐

Checked by:

Date Signature

Problem 1 Estimation (4+4+4+4+3=19)

Consider a sensor network consisiting of N sensors measuring an environmental parameter A. The sensors all send their data to a fusion center, where the estimate of A is computed. We assume that the measurement from sensor n is

$$x[n] = A + w[n]$$

where $w[n] \sim \mathcal{N}(0, \sigma_n^2)$, that is, the noise has zero mean, but different variance for each sensor. We assume σ_n is known for all n.

1a) Write the estimation problem as a linear model.

1b) Write down the Cramer-Rao bound for the estimation problem. (Hint: For a general linear problem with colored noise, $\mathbf{x} = H\Theta + \mathbf{w}$ we have)

$$\nabla_{\hat{\Theta}} \log p(\mathbf{x}; \Theta) = H^T \Sigma^{-1} H \left((H^T \Sigma^{-1} H)^{-1} H^T \Sigma^{-1} \mathbf{x} - \Theta \right)$$

1c) Find a closed form expression for the estimator.

1d) Write down the likelihood function for the problem, and use this to obtain the MLE.

1e) Assume now that A is a random variable,

$$A \sim \mathcal{N}(0, \sigma_A^2).$$

Explain why a you should use a Bayesian estimator in this case, and explain the differense between the Bayesian Mean square error estimator (B_{mse}) and the Maximum a Posteriori (MAP) estimator.

1a) We let $\mathbf{x}$ be the vector of observations, $\mathbf{w}$ the vector of noise samples and $\mathbf{1}$ a vector of just ones. Then the linear model is

$$\mathbf{x} = \mathbf{1} A + \mathbf{w}$$

1b) We see that we have

$$\nabla_\theta \log p(\mathbf{x};\theta)$$
$$= H^T \Sigma^{-1} H \left((H^T \Sigma^{-1} H)^{-1} H^T \Sigma^{-1} \mathbf{x} - \theta \right),$$

where $(I(\theta))^{-1}$ is the CRLB. This means that in our case, with $H = \mathbf{1}$ and Σ a diagonal matrix with σ_n^2 at the n'th position, we have

$$\mathrm{var}(\hat{A}) \geq (H^T \Sigma^{-1} H)^{-1}$$
$$= \left(\sum_{n=0}^{N-1} \sigma_n^{-2} \right)^{-1}$$

1c) From the hint in the previous section we see that

$$g(\mathbf{x}) = (H^T \Sigma^{-1} H)^{-1} H^T \Sigma^{-1} \mathbf{x}$$

Plugging in $H = \mathbf{1}$ we get

$$g(\mathbf{x}) = \left(\sum_{n=0}^{N-1} \sigma_n^{-2} \right)^{-1} \left(\sum_{n=0}^{N-1} \frac{x_n}{\sigma_n^2} \right)$$

1d) The likelihood of the observations $\mathbf{x}$ is

$$L(A \mid \mathbf{x})$$

$$= P(\mathbf{x}; A)$$

$$= \prod_{n=0}^{N-1} \mathcal{N}(A - x_n ; 0, \sigma_n^2)$$

$$= \prod_{n=0}^{N-1} (2\pi\sigma_n^2)^{-\frac{1}{2}} e^{-\frac{1}{2} \frac{(x_n - A)^2}{\sigma_n^2}}$$

Likewise, we have the log-likelihood

$$\ell(A \mid \mathbf{x}) = \log P(\mathbf{x}; A)$$

$$= \sum_{n=0}^{N-1} \log \mathcal{N}(A - x_n ; 0, \sigma_n^2)$$

$$= \sum_{n=0}^{N-1} -\frac{1}{2}\log 2\pi\sigma^2 - \frac{1}{2}\frac{(x_n - A)^2}{\sigma_n^2}$$

To solve for A we differentiate the log-likelihood and set it equal to zero:

$$\frac{d}{dA}\, l(A \mid \mathbf{X})$$

$$= -\sum_{n=0}^{N-1} \frac{d}{dA}\, \frac{1}{2}\, \frac{(x_n - A)^2}{\sigma_n^2}$$

$$= \sum_{n=0}^{N-1} \frac{(x_n - A)}{\sigma_n^2} = 0$$

$$\Rightarrow \sum_{n=0}^{N-1} \frac{x_n}{\sigma_n^2} = A \sum_{n=0}^{N-1} \frac{1}{\sigma_n^2}$$

$$\Rightarrow A = \left(\sum_{n=0}^{N-1} \sigma_n^{-2}\right)\left(\sum_{n=0}^{N-1} \sigma_n^{-2} x_n\right)$$

1e) Only the Bayesian framework can utilize the extra information by taking

$$P(A) = \mathcal{N}(0, \sigma_A^2)$$

as the prior. Not using the Bayesian framework would be to ignore this information

The B_{mse} and MAP estimators are based on different loss functions:

For B_{mse} we measure the squared error between the estimate and the true value: $(\theta - \hat{\theta})^2$

Minimizing the expected loss, $E_{\theta,x}\{(\theta-\theta)^2\}$, yields the conditional mean as the estimator

$$\hat{\theta} = \int \theta \, p(\theta|\mathbf{x}) \, d\theta$$

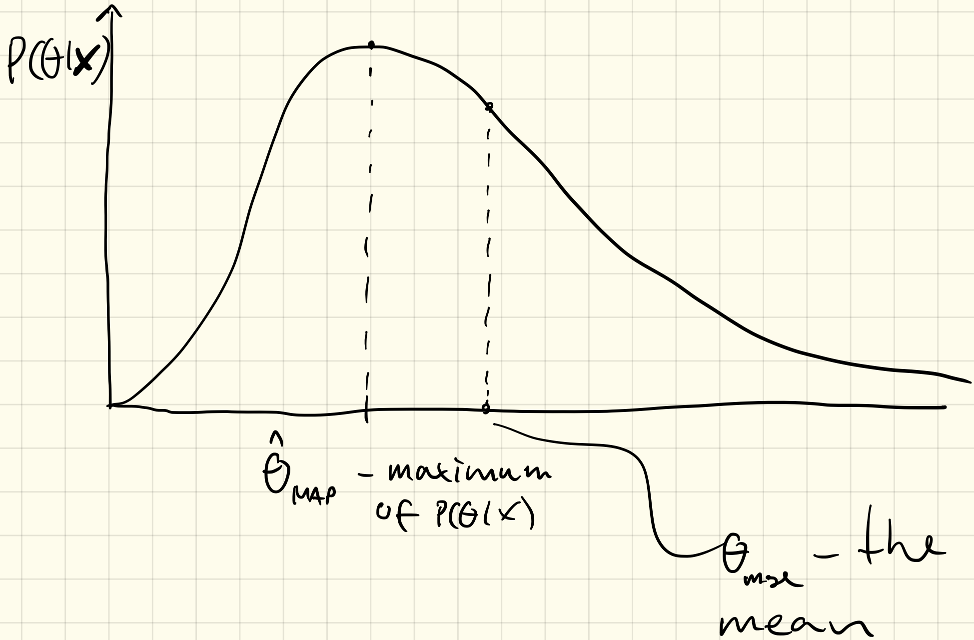
MAP is based on the loss function

$$d(\theta,\hat{\theta}) = \begin{cases} 0, & |\theta-\hat{\theta}| \leq \varepsilon \\ 1, & \text{otherwise} \end{cases}$$

It can be shown that as $\varepsilon \to 0$ the estimator becomes

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} \; P(\theta|\mathbf{X})$$

The two estimators can be visualized using a skewed distribution



If $P(\theta|\mathbf{x})$ is symmetric,

$$\hat{\theta}_{MAP} = \hat{\theta}_{mse}$$

Problem 2 Detection (4+4+3+5+3 = 19)

Consider the following binary hypothesis testing problem

$$H_0 : x[n] \sim N(0, 0), n = 0, \ldots, N-1$$
$$H_1 : x[n] \sim N(0, 1)$$

2a) For the case of $N = 1$, design an NP detector (decision rule and threshold) that ensures that the probability of false alarm does not exceed $P_{FA} = 0.1$.

2b) Find the probability of detection P_D of the detector developed in a).

2c) Assuming that someone tells you that the occurrence probability of hypothesis H_0 is $\pi_0 = 0.2$. Find the test that will yield the minimum probability of error P_e.

2d) What is the probability of error in Problem 2c)?

2e) Assume that you get access to two samples instead of one, i.e., $N = 2$. How would you modify your NP detector? Will the new detector result in an increased or decreased value of P_D?

2a) The NP detector decides H_1 if

$$L(x) = \frac{p_1(x)}{p_0(x)} = \frac{\frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(x-1)^2}}{\frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}} > \lambda$$

$$\Leftrightarrow \quad e^{-\frac{1}{2}\{(x-1)^2 - x^2\}} = e^{x-\frac{1}{2}} > \lambda$$

$$\Leftrightarrow \quad \boxed{x > \ln\lambda + \frac{1}{2} = \lambda'}$$

P_{FA} = Prob{ decide H_1 when H_0 is true } = Prob{ $x > \lambda'$; H_0 }

$$= \int_{\lambda'}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2} dx = Q(\lambda')$$

$$\Rightarrow \quad \lambda' = Q^{-1}(P_{FA})$$

$\therefore$ Decision rule: Decide H_1 if sample

$x(0) > \lambda'$ where $\lambda' = Q^{-1}(0{,}1)$

This rule guarantees $P_{FA} = 0{,}1$

$Q(x)$ and $Q^{-1}(x)$ need tables or software

2b)

P_D = Prob{ decide H_1 when H_1 is true } = Prob{ $x > \lambda'$; H_1 }

$$= \int_{\lambda'}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(x-1)^2} dx = Q(\lambda' - 1) = Q(Q^{-1}(P_{FA}) - 1)$$

2c) $\pi_0 = 0,2 \Rightarrow \pi_1 = 1 - \pi_0 = 0,8$

MPE decides H_1 if

$$L(x) = \frac{p_1(x)}{p_0(x)} > \frac{\pi_0}{\pi_1}$$

$$\Leftrightarrow \quad e^{x - \frac{1}{2}} > \frac{2}{8} = \frac{1}{4}$$

$$\Leftrightarrow \quad x > \ln\frac{1}{4} + \frac{1}{2} \approx -0,886 = \lambda'$$

$\therefore$ MPE decides H_1 if

$$x > -0,886$$

2d) $P_e = \pi_0 \, Prob\{\delta(x) = 1 \mid H_0\} + \pi_1 \, Prob\{\delta(x) = 0 \mid H_1\}$

$$= \pi_0 \int_{\lambda'}^{\infty} p_0(x)\,dx + \pi_1 \underbrace{\int_{-\infty}^{\lambda'} p_1(x)\,dx}_{1 - \int_{\lambda}^{\infty} p_1(x)\,dx}$$

$$= \pi_0 Q(\lambda') + \pi_1 (1 - Q(\lambda' - 1))$$

2e) Intuitively P_D should increase when we obtain more samples (information) from the process.

Decide H_1:

$$L(x) = \frac{p_1(x_0, x_1)}{p_0(x_0, x_1)} = \frac{p_1(x_0)\,p(x_1)}{p_0(x_0)\,p_0(x_1)}$$

$$= \frac{\frac{1}{2\pi} e^{-\frac{1}{2}\{(x_0-1)^2 + (x_1-1)^2\}}}{\frac{1}{2\pi} e^{-\frac{1}{2}\{x_0^2 + x_1^2\}}} > \lambda$$

$$\Leftrightarrow \quad e^{-\frac{1}{2}\{\cancel{x_0^2} + \cancel{x_1^2} - 2x_0 - 2x_1 + 2 - \cancel{x_0^2} - \cancel{x_1^2}\}} > \lambda$$

$$\Longleftrightarrow \quad x_0 + x_1 - 1 > \ln \lambda$$

$$\Longleftrightarrow \quad \underbrace{\tfrac{1}{2}(x_0 + x_1)}_{T(\underset{\sim}{x})} > \frac{\ln\lambda + 1}{2} = \lambda'$$

$\therefore$ Decide H_1 if

$$T(\underset{\sim}{x}) = \frac{1}{2}(x_0 + x_1) > \lambda' , \quad T(\underset{\sim}{x}) \overset{H_0}{\sim} N(0, \tfrac{1}{2}) \quad T(\underset{\sim}{x}) \overset{H_1}{\sim} N(1, \tfrac{1}{2})$$

$$P_{FA} = \text{Prob}\{\text{decide } H_1 \text{ when } H_0\} = \text{Prob}\{T(\underset{\sim}{x}) > \lambda' ; H_0\}$$

$$= \text{Prob}\Big\{ \underbrace{\frac{T(\underset{\sim}{x})}{1/2}}_{N(0,1)} > \frac{\lambda'}{1/2} \Big\} = Q(2\lambda')$$

$$\Rightarrow \lambda' = \frac{1}{2} Q^{-1}(P_{FA})$$

$$P_D = \text{Prob}\{T(\underset{\sim}{x}) > \lambda', H_1\} = \text{Prob}\Big\{ \frac{T(\underset{\sim}{x}) - 1}{1/2} > \frac{\lambda' - 1}{1/2} \Big\}$$

$$= Q(2\lambda' - 2) = Q(Q^{-1}(P_{FA}) - 2) > \underbrace{Q(Q^{-1}(P_{FA}) - 1)}_{\text{single-sample performance}}$$

Problem 3 Classification (2+3+3+5+4 = 17)

3a) Give the Bayes Decision Rule (BDR) for a C-class problem.

Use Bayes rule to reformulate BDR using class priors $P(\omega_i)$ and class densities $p(x/\omega_i)$.

Answer :
BDR : $x \in \omega_i \;\Leftrightarrow\; P(\omega_i/x) = max_k P(\omega_k/x)$

BDR + BR : $x \in \omega_i \;\Leftrightarrow\; p(x/\omega_i)P(\omega_i) = max_k p(x/\omega_)P(\omega_k)$

3b) Assume a parametric form of the densities; i.e. $p(x/\theta_i) = p(x/\omega_i)$.

Explain the principle for Maximum Likelihood (ML) based estimation of θ_i from a training set $X = [x_1, \ldots, x_N]$

Answer :
$LL(\theta_i/X) = log[p(X/\theta_i)] = \sum_n log[p(x_n/\theta_i)] \;\Leftrightarrow\; \theta_{iML} = argmax\{LL(\theta_i/X)\}$

The max values are found by setting the gradient to zeros :

$\nabla_{\theta_i} LL(\theta_i/X) = \sum_n \nabla_{\theta_i} log[p(x_n/\theta_i)] = 0$

3c) Given a scalar observation x (1-dimensional) and assume Gaussian densities $p(x/\mu_i) = N(\mu_i, \sigma_i^2)$.

Derive the expression for the ML-estimate of the mean μ_i.

Answer :
$p(x/\mu_i) = N(\mu_i, \sigma_i^2) = \frac{1}{\sqrt{(2\pi)}\sigma_i} e^{-(x-\mu_i)^2/2\sigma_i^2} \;\Rightarrow\; log[p(x\mu_i)] = K - (x-\mu_i)^2/2\sigma_i^2$
$\nabla_{\mu_I} p(x/\mu_i) = (x - mu_i)/\sigma_i^2 \;\Rightarrow\; \sum_n (x_n - \mu_i)/\sigma_i^2 = 0 \;\Rightarrow$

$\mu_{iML} = \frac{1}{N} \sum_n x_n$ (sample mean)

3d) Give the decision rule and the discriminant formula for a linear discriminant classifier for C classes.

Sketch a linear discriminant classifier using sigmoids and binary targets.

Derive the gradient upgrade expression for MSE-based training.

Answer :
Decision rule : $x \in \omega_i \;\Leftrightarrow\; g_i(x) = max_k g_k(x)$

Discriminant formula : $g = Wx$ where g is a C-dimensional vector, x has dimension $D_x + 1$ (including offset) and W is a $Cx(D_x + 1)$ matrix See separate sheet for sketch.

Using sigmoids : $y = Wx$ and $g = sigmoide(y) = \frac{1}{1+e^{-y}}$

$\nabla_W (t-g)^T(t-g) = (t-g).\nabla_W g = (t-g).\nabla_y g.\nabla_W y = (t-g).g.(1-g).x^T$

For the whole training set we get :
$\nabla_W \sum_n (t_n - g_n)^T(t_n - g_n) = \sum_n (t_n - g_n).g_n.(1-g_n).x_n^T$ where we use elementwise multiplications

Finally : $W_{new} = W_{old} - \alpha \sum_n (t_n - g_n).g_n.(1-g_n).x_n^T$

3e) Explain shortly the principle of clustering.

What is meant by hierarchical clustering?

Answer :
Clustering means to organize a set of N observations into a set L of clusters where $L << N$. In order to to this we have to decide upon a measure for the similarity/distance between an observation and a cluster (center). We start by choosing a number of clusters and intitial values for the cluster centers. We then "classify" all the observations and update the cluster (centers) based on the new labels. We do this classifying/updating procedure iteratively until no (or only small) improvements are made.

In hierarchical clustering we start by a single cluster (L=1) and increase the cluster numbers (L=L+1) when clustering into L clusters is finished. We stop when no improvement is made by increasing the cluster numbers.