02471 Machine Learning for Signal Processing solution

Problem 1 Correlation functions

Problem 1.1

The biased cross-correlation function is defined in exercise 3.1.3, and is:

$$\hat{r}_{xy} = \frac{1}{N}\sum_{n=k}^{N-1} x(n)y(n-m), \quad \text{for } k = 0, 1, \cdots, N-1$$

Inserting the numbers give:

$$\begin{aligned}
\hat{r}_{xy}(k=0) &= \frac{1}{N}\sum_{n=m}^{N-1} x(n)x(n) \\
&= \frac{1}{4}(5^2 + 3^2 + 6^2 + 2^2) = 18.5 \\
\hat{r}_{xy}(k=3) &= \frac{1}{N}\sum_{n=m}^{N-1} x(n)x(n-3) \\
&= \frac{1}{4}(x(3)x(0) = \frac{1}{4}(5 \cdot 2) = 2.5
\end{aligned}$$

Problem 1.2

The auto-correlation function, according to ML p 42–43, eq (2.101)+(2.105), is $r_x(n, n-k) = r_x(k) = \mathbb{E}[\mathrm{x}_n \mathrm{x}_{n-k}]$ when the process is wide-sense stationary.

The analytical expression is (using that WSS processes can be freely time-shifted). From exercise 3.1.2, we have almost the same case.

$$\begin{aligned}
\mathbb{E}[\mathrm{x}_n \mathrm{x}_{n-k}] &= \mathbb{E}[(\alpha \mathrm{y}_{n-d_1} + \beta \mathrm{z}_{n-d_2})\,(\alpha \mathrm{y}_{n-d_1-k} + \beta \mathrm{z}_{n-d_2-k})] \\
&= \mathbb{E}\left[\left(\alpha \mathrm{y}_n + \beta \mathrm{z}_{n-(d_2-d_1)}\right)\left(\alpha \mathrm{y}_{n-k} + \beta \mathrm{z}_{n-(d_2-d_1)-k}\right)\right]
\end{aligned}$$

The equation is now on the same form as exercise 3.1.2, if $d = d_2 - d_1$. Thus, reusing the result from that exercise, we get directly

$$\mathbb{E}[\mathrm{x}_n \mathrm{x}_{n-k}] = = \alpha^2 r_y(k) + \beta^2 r_z(k) + \alpha\beta r_{yz}(d_2 - d_1 + k) + \alpha\beta r_{yz}(d_2 - d_1 - k)$$

If the solution from this exercise is not realized (but the idea is that exercise 3.1.2 should be used), one can also take the long route

$$\begin{aligned}
\mathbb{E}[\mathrm{x}_n\mathrm{x}_{n-k}] &= \mathbb{E}[(\alpha \mathrm{y}_{n-d_1} + \beta \mathrm{z}_{n-d_2})\,(\alpha \mathrm{y}_{n-d_1-k} + \beta \mathrm{z}_{n-d_2-k})] \\
&= \mathbb{E}\big[\alpha^2 \mathrm{y}_{n-d_1}\mathrm{y}_{n-d_1-k} + \beta^2 \mathrm{z}_{n-d_2}\mathrm{z}_{n-d_2-k} + \alpha\beta\,(\mathrm{y}_{n-d_1}\mathrm{z}_{n-d_2-k} + \mathrm{z}_{n-d_2}\mathrm{y}_{n-d_1-k})\big] \\
&= \mathbb{E}\big[\alpha^2 \mathrm{y}_n\mathrm{y}_{n-k} + \beta^2 \mathrm{z}_n\mathrm{z}_{n-k} + \alpha\beta\,(\mathrm{y}_n\mathrm{z}_{n-d_2-k+d_1} + \mathrm{z}_{n-d_2+d_1+k}\mathrm{y}_n)\big] \\
&= \mathbb{E}\big[\alpha^2 \mathrm{y}_n\mathrm{y}_{n-k} + \beta^2 \mathrm{z}_n\mathrm{z}_{n-k} + \alpha\beta\,\big(\mathrm{y}_n\mathrm{z}_{n-(d_2+k-d_1)} + \mathrm{z}_{n-(d_2-d_1-k)}\mathrm{y}_{n)}\big)\big] \\
&= \alpha^2 r_y(k) + \beta^2 r_z(k) + \alpha\beta r_{yz}(d_2+k-d_1) + \alpha\beta r_{yz}(d_2-d_1-k)
\end{aligned}$$

Problem 1.3

According to ML def 2.3, page 44, and example 2.2, page 45, this is not an ergodic procees. According to ML p 44, eg. (2.108), we have

$$\mathbb{E}[\mathrm{x}_n] = \mu = \lim \hat{\mu}_N, N \to \infty$$

However, as pointed out in example 2.2, page 45, the time realizations will have different mean values since z_n is drawn randomly.

Problem 2 Parameter estimation

Problem 2.1

This is solved using the LS estimate, according to equation (3.17) in ML, page 73.

$$\hat{\boldsymbol{\theta}} = (X^T X)^{-1} X^T \boldsymbol{y}$$

Defining X and $\boldsymbol{y}$ as

$$X = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \\ 1 & 3 \end{bmatrix} \quad \boldsymbol{y} = \begin{bmatrix} 0.5 \\ 1.75 \\ 4.3 \\ 6.1 \end{bmatrix}$$

This gives

$$\hat{\boldsymbol{\theta}} = \begin{bmatrix} 0.26 \\ 1.94 \end{bmatrix}$$

Problem 2.2

This problem is described in the setup in ML section 3.5.1, and in particular eq (3.21) and eq. (3.26):

$$-\frac{2\mathrm{MSE}(\hat{\theta}_{\mathrm{MVU}})}{\mathrm{MSE}(\hat{\theta}_{\mathrm{MVU}}) + \theta_o^2} < \alpha < 0$$

Since $\hat{\theta}_{\text{MVU}}$ is unbiased, we know that $\mathbb{E}\left[\hat{\theta}_{\text{MVU}}\right] = \theta^{opt}$ (bias is defined in eq (3.22)).

The variance of a random variable is defined (eq 2.25) as $\mathbb{E}\left[(\mathrm{x}-\mu_{\mathrm{x}})^2\right]$, hence we see that $\text{MSE}(\hat{\theta}_{\text{MVU}}) = \text{VAR}\left[\hat{\theta}_{\text{MVU}}\right] = 1$.

Plugin into the above formula we get

$$-\frac{2\text{MSE}(\hat{\theta}_{\text{MVU}})}{\text{MSE}(\hat{\theta}_{\text{MVU}}) + \theta_o^2} = -\frac{2\cdot 1}{1+2^2} = -0.4$$

Hence we get $-0.4 < \alpha < 0$.

Problem 2.3

The shrinkage / thresholding function is defined in ML sec 10.2.2, page 481, as

$$S_{\lambda\mu}(\theta) = \text{sgn}(\theta)\left(|\theta| - \lambda\mu\right)_+$$

We complete the iteration as

$$\begin{aligned}
\theta^{(i)} &= S_{\lambda\mu}\left(\theta^{(i-1)} + \mu X^T \boldsymbol{e}^{(i-1)}\right) \\
&= S_{\lambda\mu}\left(4 + 0.5\cdot\begin{bmatrix}0 & 1 & 2 & 3\end{bmatrix}\begin{bmatrix}1 & -1 & -3 & 2\end{bmatrix}^T\right) \\
&= S_{\lambda\mu}\left(3.5\right) \\
&= \text{sgn}(3.5)\left(|3.5| - 0.5\lambda\right)_+
\end{aligned}$$

For $\lambda = 1$ we then get

$$\theta^{(i)} = \text{sgn}(3.5)\left(3.5 - 0.5\cdot 1\right)_+ = 1\cdot 3 = 3$$

Problem 2.4

As described in ML sec 3.10, and exercise 9.1.2, this situation corresponds to the measurement noise being Gaussian white noise (normal distributed noise with residuals uncorrelated).

From ML sec 12.2.2, page 598, we in particular have

$$\lambda = \frac{\sigma_\eta^2}{\sigma_\theta^2}$$

For $\lambda = 2$ and $\sigma_\theta^2 = 0.5$, we get $\sigma_\eta^2 = \lambda\cdot\sigma_\theta^2 = 1$.

Problem 2.5

According to ML p 614. we have

$$\begin{aligned}\ln p(\boldsymbol{y}, \boldsymbol{\theta}; \alpha, \beta) =& \frac{N}{2}\ln\beta + \frac{K}{2}\ln\alpha - \frac{\beta}{2}\|\boldsymbol{y} - \Phi\boldsymbol{\theta}\|^2 - \frac{\alpha}{2}\boldsymbol{\theta}^T\boldsymbol{\theta} \\ &- \left(\frac{N}{2} + \frac{K}{2}\right)\ln(2\pi)\end{aligned}$$

where $\alpha = \frac{1}{\sigma_\theta^2}$ and $\beta = \frac{1}{\sigma_\eta^2}$

Relating the expression to the setup in the problem, we get

$$\begin{aligned}\ln p(\boldsymbol{y}, \theta; \sigma_\theta^2, \sigma_\eta^2) =& \frac{N}{2}\ln\frac{1}{\sigma_\eta^2} + \frac{K}{2}\ln\frac{1}{\sigma_\theta^2} - \frac{1}{2\sigma_\eta^2}\|\boldsymbol{y} - \boldsymbol{x}\theta\|^2 - \frac{1}{2\sigma_\theta^2}\theta^2 \\ &- \left(\frac{N}{2} + \frac{K}{2}\right)\ln(2\pi)\end{aligned}$$

In our case, $K = 1$ (number of weights) and $N = 4$ (number of points), so we get

$$\begin{aligned}\ln p(\boldsymbol{y}, \theta; \sigma_\theta^2, \sigma_\eta^2) =& 2\ln\frac{1}{\sigma_\eta^2} + \frac{1}{2}\ln\frac{1}{\sigma_\theta^2} - \frac{1}{2\sigma_\eta^2}\|\boldsymbol{y} - \boldsymbol{x}\theta\|^2 - \frac{1}{2\sigma_\theta^2}\theta^2 \\ &- \frac{5}{2}\ln(2\pi)\end{aligned}$$

Problem 3 Linear adaptive filtering

Problem 3.1

According to ML sec 4.5 we have

$$\begin{bmatrix} r_u(0) & r_u(1) & r_u(2) \\ r_u(1) & r_u(0) & r_u(1) \\ r_u(2) & r_u(1) & r_u(0) \end{bmatrix} \cdot \begin{bmatrix} w_0 \\ w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} r_{du}(0) \\ r_{du}(1) \\ r_{du}(2) \end{bmatrix}$$

For the filter coefficients we then get

$$\begin{aligned}\begin{bmatrix} w_0 \\ w_1 \\ w_2 \end{bmatrix} &= \begin{bmatrix} r_u(0) & r_u(1) & r_u(2) \\ r_u(1) & r_u(0) & r_u(1) \\ r_u(2) & r_u(1) & r_u(0) \end{bmatrix}^{-1} \begin{bmatrix} r_{du}(0) \\ r_{du}(1) \\ r_{du}(2) \end{bmatrix} \\ &= \begin{bmatrix} 1.7 & 0.7 & 0.4 \\ 0.7 & 1.7 & 0.7 \\ 0.4 & 0.7 & 1.7 \end{bmatrix}^{-1} \begin{bmatrix} 1.2 \\ 0.6 \\ 0.35 \end{bmatrix} \\ &= \begin{bmatrix} .67 \\ .07 \\ .02 \end{bmatrix}\end{aligned}$$

According to ML eq 4.9, we get, for the minimum error

$$\begin{aligned} J(\boldsymbol{\theta}) &= \sigma_y^2 - \boldsymbol{p}^T \boldsymbol{w}_* \\ &= r_d(0) - \begin{bmatrix} 1.2 \\ 0.6 \\ 0.35 \end{bmatrix}^T \begin{bmatrix} 0.67 \\ 0.07 \\ 0.02 \end{bmatrix} \\ &= 0.54 \end{aligned}$$

where $r_d(0) = 1.4$.

Problem 3.2

For this problem, the model in ML, section 6.8 is used. According to table 6.1, page 276, we have

$$J_{\text{exc}} \simeq \frac{1}{2}\left(\mu\sigma_\eta^2 \operatorname{trace}\{\Sigma_x\} + \frac{1}{\mu}\operatorname{trace}\{\Sigma_\omega\}\right)$$

Inserting the values from the problem, we get

$$J_{\text{exc}} \simeq \frac{1}{2}\left(0.5 \cdot 0.5 \cdot 2 \cdot 1.7 + \frac{1}{0.5} \cdot (0.1 + 0.2)\right) = 0.725$$

Problem 4 Dictionary learning

Problem 4.1

Using the ICA model ML eq (19.44), $\mathbf{x} = A\mathbf{s}$, we see by inspection that

$$A = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix}$$

If this is not directly seen, it can also be solved by using the ICA model and the pseudo–inverse, or even normal linear algebra as is done using least squares

$$X = AS \quad \Leftrightarrow \quad XS^T = ASS^T \quad \Leftrightarrow \quad XS^T(SS^T)^{-1} = A$$

Problem 4.2

ICA minimizes the mutual information of the sources (z), not the mixing matrix, thus ”E” is the false statement (correct answer).

Problem 5 State-space models and time-frequency analysis

Problem 5.1

The description of the short-time Fourier transform is found in week 7, slide 33. The sampling frequency is $F_s = 5Hz$, with a windows size of 30 seconds, we get $30 \times Fs = 150$ samples per window.

With a window overlap of 50%, we process a total of $\frac{600 seconds}{30 seconds \cdot 50\%} - 1 = 39$.

Problem 5.2

According to ML, sec 16.5.1, the HMM has the parameters

1. Number of states K
2. Initial probabilities P_k
3. Transition probabilities P_{ij}
4. Emission distributions, which is discrete in this case, $p(y|k)$.

From the description, we recognize the bar plots will describe the emission probabilities $p(y|k)$, and the number of states are $K = 2$ (walking, running).

We read off the following probabilities (with columns being low, medium and high frequency respectively)

$$P(Y|k = \text{walking}) = \begin{bmatrix} 0.45 & 0.35 & 0.2 \end{bmatrix}$$
$$P(Y|k = \text{running}) = \begin{bmatrix} 0.05 & 0.15 & 0.8 \end{bmatrix}$$

Problem 5.3

Since we are given the information that the person was running for sure, we can use the Markov property and discard the observations from the first 60 seconds. The initial probability is then $P_{\text{running}} = 1$.

Thus we need to calculate $P(\mathbf{x}_n = \text{running}|Y_1 = \text{high freq})$. Using the equations from ML, page 850, we get

$$\begin{aligned} P(\mathbf{x}_1 = \text{running}|Y_1 = \text{high freq}) &= \frac{\alpha(\mathbf{x}_1 = \text{running})}{P(Y_1)} \\ &= \frac{\alpha(\mathbf{x}_1 = \text{running})}{\alpha(\mathbf{x}_1 = \text{running}) + \alpha(\mathbf{x}_1 = \text{walking})} \end{aligned}$$

According to ML sec 16.5, eq (16.40)–(16.41), we have

$$\alpha(\mathbf{x}_1) = P(\mathbf{y}_1, \mathbf{x}_1) = P(\mathbf{y}_1|\mathbf{x}_1)P(\mathbf{x}_1)$$

Denoting s_1 = running, s_2 = walking, and f_3 = high-freq, we then have for the sequence $y_{1:1} = \{f_3\}$:

$$\alpha(\mathbf{x}_1) = P(y_1 = f_3|\mathbf{x}_1)P(\mathbf{x}_1)$$
$$\begin{bmatrix} \alpha(\mathrm{x}_1 = s_1) \\ \alpha(\mathrm{x}_1 = s_2) \end{bmatrix} = \begin{bmatrix} P(\mathrm{y}_1 = f_3|\mathrm{x}_1 = s_1)P(\mathrm{x}_1 = s_1) \\ P(\mathrm{y}_1 = f_3|\mathrm{x}_1 = s_2)P(\mathrm{x}_1 = s_2) \end{bmatrix}$$

Since we have confirmation that the person is running, we then have $P(\mathrm{x}_1 = s_1)$ and $P(\mathrm{x}_1 = s_2)$ as the transition probabilities, hence

$$\begin{aligned} \begin{bmatrix} \alpha(\mathrm{x}_1 = s_1) \\ \alpha(\mathrm{x}_1 = s_2) \end{bmatrix} &= \begin{bmatrix} .8 \cdot .9 \\ .2 \cdot .1 \end{bmatrix} \\ &= \begin{bmatrix} .72 \\ .02 \end{bmatrix} \end{aligned}$$

Thus, we get

$$\begin{aligned} P(\mathbf{x}_1 = \text{running}|Y_1 = \text{high freq}) &= \frac{\alpha(\mathbf{x}_1 = \text{running})}{\alpha(\mathbf{x}_1 = \text{running}) + \alpha(\mathbf{x}_1 = \text{walking})} \\ &= \frac{0.72}{0.72 + 0.02} \\ &= 0.973 \end{aligned}$$

Alternatively, this can also be solved using Bayes and the structure of the Hidden Markov model.

Problem 6 Kernels

Problem 6.1

The two closest points to the test point are $x_{blue} = (-1.5, 0)$ and $x_{red} = (-3, 0)$ respectively. Computing the kernel for these values give:

$$\begin{aligned}
\kappa(\boldsymbol{x}_{blue}, \boldsymbol{x}_{test}) &= \exp\left(-\frac{\|\boldsymbol{x}_{blue} - \boldsymbol{x}_{test}\|^2}{2\sigma^2}\right) \\
&= \exp\left(-\frac{(-1.5 - (-2))^2}{2}\right) \\
&= 0.88 \\
\kappa(\boldsymbol{x}_{red}, \boldsymbol{x}_{test}) &= \exp\left(-\frac{\|\boldsymbol{x}_{red} - \boldsymbol{x}_{test}|^2}{2\sigma^2}\right) \\
&= \exp\left(-\frac{(-3 - (-2))^2}{2}\right) \\
&= 0.61
\end{aligned}$$

Since the kernel assigns a higher value for the points of the correct class, the kernel can be used.

E.g applying the representer theorem with $\theta_k = 1$ for all red points and $\theta_k = -1$ for all blue will readily show a separation, see Figure 1, page 9 (plot is not expected as part of the solution).

Problem 6.2

The weight vector is computed, according to ML eq. 11.26

$$\hat{\boldsymbol{\theta}} = (\mathcal{K} + CI)^{-1}\boldsymbol{y}$$

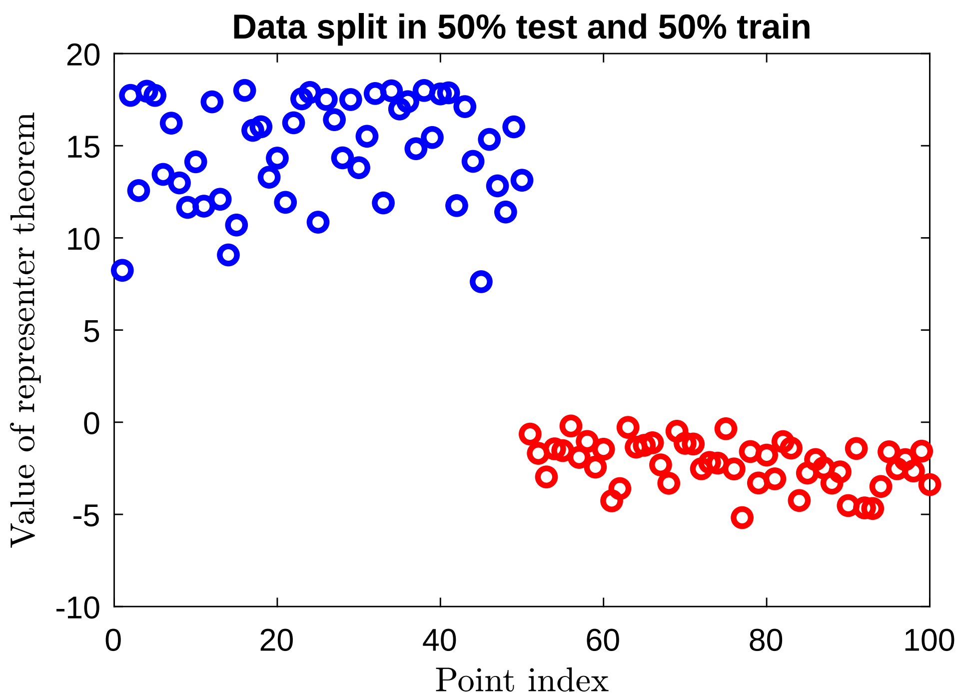

Figure 1: Representer theorem applied

With $\sigma = 1.5$ and $C = 0.02$, and using the definition of $\mathcal{K}$ (11.11) we get

$$\begin{aligned}
\hat{\boldsymbol{\theta}} &= \left(\begin{bmatrix} \kappa(x_1, x_1) & \kappa(x_1, x_2) & \kappa(x_1, x_3) \\ \kappa(x_2, x_1) & \kappa(x_2, x_2) & \kappa(x_2, x_3) \\ \kappa(x_3, x_1) & \kappa(x_3, x_2) & \kappa(x_3, x_3) \end{bmatrix} + CI \right)^{-1} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} \\
&= \begin{bmatrix} \exp(0) + 0.02 & \exp(-0.222) & \exp(-0.889) \\ \exp(-0.222) & \exp(0) + 0.02 & \exp(-0.222) \\ \exp(-0.889) & \exp(-0.222) & \exp(0) + 0.02 \end{bmatrix}^{-1} \begin{bmatrix} 0.5 \\ 0.8 \\ 0.7 \end{bmatrix} \\
&= \begin{bmatrix} 1.02 & 0.801 & 0.411 \\ 0.801 & 1.02 & 0.801 \\ 0.411 & 0.801 & 1.02 \end{bmatrix}^{-1} \begin{bmatrix} 0.5 \\ 0.8 \\ 0.7 \end{bmatrix} \\
&= \begin{bmatrix} -0.335 \\ 1.037 \\ 0.003 \end{bmatrix}
\end{aligned}$$

Problem 6.3

To calculate the prediction, we get, according to eq (11.27):

$$
\begin{aligned}
\hat{y}(x) &= \boldsymbol{\theta}^T \kappa(x) \\
\hat{y}(x) &= \theta_1 \kappa(x, x_1) + \theta_2 \kappa(x, x_2) + \theta_3 \kappa(x, x_3) \Leftrightarrow \\
\theta_3 &= \frac{\hat{y}(x) - \theta_1 \kappa(x, x_1) + \theta_2 \kappa(x, x_2)}{\kappa(x, x_3)} \\
&= \frac{3 - 1 \cdot 0.3 + 3 \cdot 0}{0.6} \\
&= 4.5
\end{aligned}
$$