Assignment Sheet - III

Stylianos Sygletos
s.sygletos@aston.ac.uk

May 17, 2021

Problem 1 [2 points]

Consider a case where we have learned a conditional probability distribution $P(y|\boldsymbol{x})$. Suppose there are only two classes, and let $p_0 = P(Y = 0|\boldsymbol{x})$ and $p_1 = P(Y = 1|\boldsymbol{x})$. Consider the loss matrix below:

predicted label $\hat{y}$	true label y: 0	true label y: 1
0	0	λ_{01}
1	λ_{10}	0

i. Show that the decision $\hat{y}$ that minimizes the expected loss is equivalent to setting a probability threshold θ and predicting $\hat{y} = 0$ if $p_1 < \theta$ and $\hat{y} = 1$ if $p_1 \geq \theta$. Express θ as a function of λ_{01} and λ_{10}.[1pt]

ii. Derive the loss matrix where the threshold is 0.1. [1pt]

Problem 2 [5 points]

Consider the generation of $M = 1000$ sets of training points that are generated according to the following regression model:

$$y = g(x) + \eta = \begin{bmatrix} 0.1 & 0.6 & 0.5 & -0.8 & 0.2 & 0.3 \end{bmatrix} \begin{bmatrix} 1 \\ x \\ x^2 \\ x^3 \\ x^4 \\ x^5 \end{bmatrix} + \eta$$

i.e. the function $g(x)$ is a fifth-order polynomial and η is a Gaussian random variable of zero mean and variance $\sigma^2 = 0.3$. The training sets are generated as follows. First, The x-axis is sampled at $N = 10$ equidistant points $x_n, n = 1, ..., N$ in the interval [-1, 1]. Then each training set D_i is created as

$$D_i = \{(g(x_n) + \eta_{n,i}, x_n) : n = 1, 2, ...N\}, \quad i = 1, 2, ..., M$$

where $n_{n,i}$ denotes different noise samples, drawn i.d.d. from a white noise process. In other words, all training points have the same x-coordinate but different y-coordinates due to the different values of the noise. We will make two attempts to fit the noisy data. The first setup is with a high-order polynomial of degree equal to 10 and a second setup adopts a second-order polynomial.

i. Take one of the generated training set D_i and perform the fitting with each one of the two setups. Plot the results together in the same graph along with the "unknown" (original g) function. Try to make a clear and conclusive plot by selecting the appropriate plotting ranges for the $x-$ and $y-$ axis. [1.0pt]

ii. Perform the above experiment $M = 1000$ times, each with a different data set D_i. Select and plot 10 only out of 1000 fitting curves (for visibility reason), for the high- and low-order polynomials, respectively. [1.0pt]

iii. Show the corresponding curves which result from averaging over the 1000 performed experiments, together with the graph of the "unknown" (original g) function. Comment on the results. [1.0pt]

iv. Play with the number of training points, the degree of the involved polynomials, and the noise variance σ^2 in the regression model. Comment on the results. [2.0pt]

Problem 3 [3 points]

This problem deals with Bayesian regression, based on the model developed in the sections (1.2.6) and (3.3.1-3.3.2) of the PRML textbook, as well as, at the relevant lab session.

Let us start. Data are generated based on the following nonlinear model:

$$y_n = \theta_0 + \theta_1 x_n + \theta_2 x_n^2 + \theta_3 x_n^3 + \theta_5 x_n^5 + \eta_n \quad \text{with} \quad n = 1, 2, ..., N$$

where η_n are i.i.d noise samples drawn from a zero mean Gaussian with variance σ_n^2. The parameter values used to generate the data were equal to : $\theta_0 = 0.2, \theta_1 = -1, \theta_2 = 0.9, \theta_3 = 0.7, \theta_5 = -0.2$. Samples x_n are equidistant points in the interval $[0, 2]$. Create the output samples y_n, according to the above non-linear model where the noise variance is set equal to $\sigma_\eta^2 = 0.05$

i. For the regression consider the aforementioned 5^{th}-order polynomial model structure to construct the design matrix $\mathbf{\Phi}$ and let the parameters of the Gaussian prior be $\mathbf{\Theta}_0 = [0.2, -1, 0.9, 0.7, -0.2]$ and $\Sigma_\theta = 0.1I$. Compute the covariance matrix and the mean of the posterior Gaussian distribution, assuming a training set size of $N = 20$. Then, select randomly $K = 20$ test-set points x_k in the interval $[0, 2]$. Make sure that you use different seeds for the test and training sets in the random generator. Compute the predictions for the mean values, μ_y, and the associated variances, σ_y^2. Plot the graph of the true function together with the predicted mean values, μ_y, and show the confidence intervals on these predictions by using the error-bar function from Matplotlib [1] package. Repeat the experiment again, using a training set of $N = 500$ points, and try different values of σ_η^2, to notice the change on the estimated confidence intervals. [1.5pt]

ii. Repeat the previous experiment using a randomly chosen value for the prior's mean $\mathbf{\Theta}_0$ and different values on the parameters, for example $\sigma_\eta^2 = 0.05$ or $\sigma_\eta^2 = 0.15$, $\sigma_\theta^2 = 0.1$ or $\sigma_\theta^2 = 2$ and $N = 500$ or $N = 20$. Comment on the results [1.5pt]

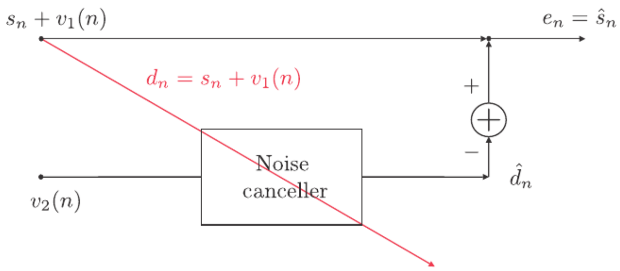

Figure 1: Block diagram for the noise canceller. Using as desired response the contaminated signal, the output of the optimal filter is an estimate of the noise component

Problem 4 [5 points]

Consider the noise cancellation task shown in Figure 1 . Write the necessary code to solve the problem using PYTHON following the below steps:

1. Create 5000 data samples of the signal $s_n = \cos(w_0 n)$, for $w_0 = 2 \times 10^{-3}\pi$ [0.5pt]

2. Create 5000 data samples of the AR process $v_1(n) = a_1 \cdot v_1(n-1) + \eta_n$ (initializing at zero), where η_n represents zero mean Gaussian noise with variance $\sigma_n^2 = 0.0025$ and $a_1 = 0.8$ [0.5pt]

3. Add the two sequences (i.e. $d_n = s_n + v_1(n)$) and plot the result. This represents the contaminated signal.[0.5pt]

4. Create 5000 data samples of the AR process $v_2(n) = a_2 v_2(n-1) + \eta_n$ (initializing at zero), where η_n represents the same noise sequence and $a_2 = 0.75$.[0.5pt]

5. Solve for the optimum (in the MSE sense) $\boldsymbol{w} = [w_0, w_1]^T$. Create the sequence of the restored signal $\hat{s}_n = d_n - w_0 v_2(n) - w_1 v_2(n-1)$ and plot the result.[1.0pt]

6. Repeat steps (ii)-(v) using $a_2 = 0.9, 0.8, 0.7, 0.6, 0.5, 0.3$. Comment on the results.[1.0pt]

7. Repeat steps (ii)-(v) using $\sigma_\eta^2 = 0.01, 0.05, 0.1, 0.2, 0.5$ for $a_2 = 0.9, 0.8, 0.7, 0.6, 0.5, 0.3$. Comment on the results.[1.0pt]

[1] https://matplotlib.org/