HOMEWORK-3

PROBLEM 5.5

a) According to linear regression model

$$d_i = w^T \phi_i + \varepsilon_i \qquad i = 1, 2, \ldots, N \qquad -①$$

Let $\phi = \begin{bmatrix} \phi_1^T \\ \phi_2^T \\ \vdots \\ \phi_N^T \end{bmatrix}$, $\varepsilon = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_N \end{bmatrix}$, $d = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_N \end{bmatrix}$.

We can write ① as

$$d = \phi w + \varepsilon \qquad -②$$

We obtain the least square estimate by minimizing the loss function as below

$$L(w) = \sum_{i=1}^{N} \varepsilon_i^2 = \sum_{i=1}^{N} (d_i - w^T \phi_i)^2$$

$$= \|d - \phi w\|^2$$

$$= (d - \phi w)^T (d - \phi w)$$

$$= dd^T + w^T \phi^T \phi w - 2 w^T \phi^T d \qquad -③$$

Taking gradient of $L(w)$ w.r.t w and equating to zero, we get

$$2 \phi^T \phi w - 2 \phi^T d = 0$$

$$\hat{w} = (\phi^T \phi)^{-1} \phi^T d$$

$$= (\phi^T \phi)^{-1} \phi^T (\phi w + \varepsilon)$$

$$= w + (\phi^T \phi)^{-1} \phi^T \varepsilon. \qquad -④$$

Eq ④ is the required result.

b) Using ④, we know

$$\hat{\omega} = \omega + (\phi^T\phi)^{-1}\phi^T\varepsilon$$

$$E\left[(\omega-\hat{\omega})(\omega-\hat{\omega})^T\right] = E\left[\left((\phi^T\phi)^{-1}\phi^T\varepsilon\right)\left((\phi^T\phi)^{-1}\phi^T\varepsilon\right)^T\right]$$

$$= E\left[(\phi^T\phi)^{-1}\phi^T\varepsilon\,\varepsilon^T\phi\,(\phi^T\phi)^{-T}\right] \quad -⑤$$

$E[\varepsilon\varepsilon^T]$ is the covariance of the white noise process with mean zero and component variance σ^2.

$$E[\varepsilon\varepsilon^T] = \sigma^2 I \quad -⑥$$

Using ⑥ in ⑤, after simplifying

$$E\left[(\omega-\hat{\omega})(\omega-\hat{\omega})^T\right] = \sigma^2(\phi^T\phi)^{-1} \quad -⑦$$

$$\phi^T\phi = \sum_{j=1}^{N}\phi_j\phi_j^T = R \quad -⑧$$

Using ⑧ in ⑦,

$$E\left[(\omega-\hat{\omega})(\omega-\hat{\omega})^T\right] = \sigma^2(R)^{-1}$$

$$= \sigma^2 R^{-1}$$

$$= \sigma^2 P.$$

PROBLEM 5.6:

a) $$\mathcal{E}_{av}(w) = \frac{1}{2}\sum_{i=1}^{N}\left(d(i) - \underline{w}^T\underline{\phi}(i)\right)^2 + \frac{1}{2}\lambda\|\underline{w}\|^2 \quad - (9)$$

Let $\underline{d} = \begin{bmatrix} d(1) \\ d(2) \\ \vdots \\ d(N) \end{bmatrix}$, $\underline{e} = \underline{d} - \phi\underline{w}$ $\quad \phi = \begin{bmatrix} \phi_1^T \\ \phi_2^T \\ \vdots \\ \phi_{(N)}^T \end{bmatrix}$

$$E(w) = \frac{1}{2}\sum_{i=1}^{N} e_i^2 + \frac{1}{2}\lambda\|\underline{w}\|^2$$

$$= \frac{1}{2}\underline{e}^T\underline{e} + \frac{1}{2}\lambda\|\underline{w}\|^2$$

$$= \frac{1}{2}\left[(\underline{d} - \phi\underline{w})^T(\underline{d} - \phi\underline{w}) + \frac{1}{2}\lambda\|\underline{w}\|^2\right.$$

Equating gradient to zero, we get

$$\left(\phi^T\phi + \lambda I\right)\hat{w} = \phi^T d \quad - (10)$$

Let $R = \phi^T\phi + \lambda\underline{I}$

$$\underline{r} = \sum_{i=1}^{N} d_i\phi_i = \phi^T d$$

At time n, (10) is of the form

$$R(n)\,\hat{\underline{w}}(n) = \underline{r}(n) \quad - (13)$$

$$R(n) = \sum_{i=1}^{n}\phi_i\phi_i^T + \lambda\underline{I} \quad - (11)$$

$$r(n) = \sum_{i=1}^{n} d_i\varphi_i \quad - (12)$$

From (12), $\underline{r}(n) = \underline{r}(n-1) + d_n\underline{\phi}_n$

From (13), $r(n) = R(n-1)\,\hat{\underline{w}}(n-1) + d_n\underline{\phi}_n$

Adding and subtracting $\underline{\phi}_n \underline{\phi}_n^T \hat{\underline{w}}(n-1)$,

$$r(n) = R(n-1)\hat{w}(n-1) + \underline{\phi}_n \underline{\phi}_n^T \hat{\underline{w}}(n-1) + d_n \underline{\phi}_n - \phi_n \phi_n^T \hat{w}(n-1)$$

$$= \left[R(n-1) + \underline{\phi}_n \underline{\phi}_n^T\right] \hat{\underline{w}}(n-1) + \underline{\phi}_n \left[d_n - \underline{\phi}_n^T \hat{\underline{w}}(n-1)\right] \quad -(14)$$

From (11),

$$R(n) = \sum_{i=1}^{n-1} \phi_i \phi_i^T + \lambda I + \phi_n \phi_n^T$$

$$= R(n-1) + \phi_n \phi_n^T \quad -(15)$$

Defining innovation as $\alpha_n = d_n - \hat{\underline{w}}^T(n-1)\underline{\phi}_n$ $-(16)$

$$r(n) = R(n)\hat{w}(n-1) + \phi_n \alpha_n \quad -(17)$$

$$R(n)\hat{\underline{w}}(n) = R(n)\hat{\underline{w}}(n-1) + \underline{\phi}_n \alpha_n$$

$$\hat{w}(n) = \hat{w}(n-1) + R^{-1}(n)\,\phi_n \alpha_n$$

Computing R^{-1} by matrix inversion formula

$$R^{-1}(n) = R^{-1}(n-1) - \frac{R^{-1}(n-1)\,\underline{\phi}_n \underline{\phi}_n^T R^{-1}(n-1)}{1 + \phi_n^T R^{-1}(n-1)\,\phi_n}$$

Let $P(n) = R^{-1}(n)$

$$P(n) = P(n-1) - \frac{P(n-1)\,\underline{\phi}_n \underline{\phi}_n^T P(n-1)}{1 + \underline{\phi}_n^T P(n-1)\,\underline{\phi}_n}$$

$$\therefore \hat{\underline{w}}(n) = \hat{\underline{w}}(n-1) + P(n)\,\underline{\phi}_n \alpha_n .$$

Therefore, the regularization term $\frac{1}{2}\lambda\|w\|^2$ has no effect on the composition of the RLS algorithm.

b) The only effect of the regularization term is the modification of the correlation matrix as

$$R(n) = \sum_{i=1}^{n} \phi_i \phi_i^T + \lambda \underline{I}.$$

The correlation matrix with regulizer is symmetric and positive definite, therefore it is invertible.

The practical benefit of introducing regularization term is avoiding singularity of $R(n)$ which improves stability of $R^{-1}(n)$ computation.

PROBLEM 6.17:

a) Solving the problem using radial basis functions (RBF):

. The RBF algorithm corresponds to choosing a function F of the form

$$F(x) = \sum_{i=1}^{N} w_i \phi(\|x - x_i\|) \qquad [N=8 \text{ in this problem}]$$

· Choose a gaussian function with $\sigma = 1$ as

$$\phi(\|x - x_i\|) = \exp\left(-\frac{1}{2}\|x^T x_i\|^2\right)$$

· Form the matrix $\underline{\Phi}$ such that $\underline{\Phi}(i,j) = \phi(\|x_j - x_i\|)$.

· We want $\underline{\Phi}\,\underline{w} = \underline{d}$ where $\underline{d}$ is the vector with desired output.

· Solving the system of linear equations $\underline{\Phi}\,\underline{w} = \underline{d}$, one can obtain the optimal weight values.

b) Solving the problem using support vector machines:

PROCEDURE:

The given problem is not linearly separable.

- Choosing a Mercer kernel of the form $k(\underline{x}, \underline{x}_i) = (1 + \underline{x}^T \underline{x}_i)^p$.
- Let $p = 3$ for this problem.

$$k(\underline{x}, x_i) = (1 + \underline{x}^T \underline{x}_i)^3$$

- Based on the values of the truth table, compute the values of the matrix $K(i,j) = k(\underline{x}_i, \underline{x}_j)$.
- We obtain an 8×8 matrix.

The objective function for the dual problem is given by

$$Q(a) = \sum_{i=1}^{8} \alpha_i - \frac{1}{2} \sum_{i=1}^{8} \sum_{j=1}^{8} \alpha_i \alpha_j d_i d_j K(\underline{x}_i, \underline{x}_j)$$

- Let $B(i,j) = d_i d_j K(\underline{x}_i, \underline{x}_j)$.
- Then $Q(a) = \alpha^T \mathbf{1} - \frac{1}{2} \alpha^T B \alpha$
- To find optimal α, differentiate w.r.t α and equate to 0:

$$B\alpha = 1.$$

- Solving this based on computation, we get $\alpha_i = \frac{1}{48} \; \forall i$.
- Further, extract $\underline{\phi}(x)$ from $K(\underline{x}_i, \underline{x}_j)$.
- The optimum weight vector is obtained as

$$\underline{w}_0 = \sum_{i=1}^{8} \alpha_i d_i \underline{\phi}(\underline{x}_i)$$

- Plugging d_i values and $\alpha_i = \frac{1}{48}$, the optimal weight can be obtained.

- The optimal hyperplane is given by $\underline{w}_o^T \underline{\phi}(x) = 0$.
- Based on the above computations, we get $x_1 x_2 x_3 = 0$.
- This satisfies the given problem of assigning $+1$ if $x_1 x_2 x_3 > 0$ and -1 if $x_1 x_2 x_3 < 0$.

$\Rightarrow$ You can verify the above by both hand calculations as well as by writing a code.

In this problem, all the N samples are support vectors.

The computational complexity of both the procedures for this problem is the same of complexity $O(N^3)$ as both of them require matrix inversion.

PROBLEM 6.25:

- Generate the data as shown in the figure.
- Choose an appropriate kernel like the RBF for training SVM.
- Perform experiments with various combinations of hyperparameter and value of C.
- Compute the classification error rate and check for overfitting.