HW3

Ibrahim Gabr

Problem 1

We need to find the expressions for $\mathbf{H, f, A, a, B}$ and $\mathbf{b}$ such that we can setup the dual optimization problem for the kernel SVM:

$$\underset{\mathbf{w}}{\mathbf{argmin}}\left\{\frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{N}\underset{0\leq\alpha\leq C}{\mathbf{max}}[0, 1 - y_i(\mathbf{w}\cdot\phi(\mathbf{x_i}) - w_0)]\right\}$$

From the problem above, we see that we are dealing with the case of not linearly separable data, which means that we are interested in imposing a maximum penalty (C) on constraint violation, where $\xi_i = \mathbf{max}[0, 1 - y_i(\mathbf{w}\cdot\phi(\mathbf{x_i}) - w_0)]$. We can rewrite the dual problem using a Lagrangian with the following constraints and KKT conditions:

$$\begin{aligned}
\alpha_i &\geq 0,\\
y_i(w_0 + \mathbf{w}\cdot\phi(\mathbf{x_i})) - 1 + \xi_i &\geq 0,\\
\alpha_i(y_i(w_0 + \mathbf{w}\cdot\phi(\mathbf{x_i})) - 1 + \xi_i) &= 0,\\
\mu_i &\geq 0,\\
\xi_i &\geq 0,\\
\mu_i\xi_i &= 0
\end{aligned}$$

Now, we have to solve this problem:

$$L = \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}\alpha_i[y_i(w_0 + \mathbf{w}\cdot\phi(\mathbf{x_i}) - 1 + \xi_i] - \sum_{i=1}^{N}\mu_i\xi_i$$

Taking the derivative with respect to w, w_0, and ξ_i we get

$$\frac{\delta L}{\delta \mathbf{w}} = \mathbf{w} - \sum_{i=1}^{N} \alpha_i y_i \phi(x_i) = 0$$

$$\Rightarrow \mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i \phi(x_i) \qquad (1)$$

$$\frac{\delta L}{\delta \mathbf{w_0}} = -\sum_{i=1}^{N} \alpha_i y_i = 0$$

$$\Rightarrow \sum_{i=1}^{N} \alpha_i y_i = 0 \qquad (2)$$

$$\frac{\delta L}{\delta \mathbf{\xi_i}} = C - \alpha_i - \mu_i = 0$$

$$\Rightarrow \alpha_i = C - \mu_i \qquad (3)$$

If we replace the values for w, w_0, and $\{\xi_i\}$ in the Lagrangian, we can reformulate the original problem as follows:

$$\begin{aligned} L = & \frac{1}{2}\left(\sum_{i=1}^{N} \alpha_i y_i \phi(x_i)\right) \cdot \left(\sum_{i=j}^{N} \alpha_j y_j \phi(x_j)\right) + C\sum_{i=1}^{N} \xi_i - \sum_{i=1}^{N} (C - \alpha_i)\xi_i \\ & - \left[w_0 \sum_{i=1}^{N} \alpha_i y_i + \sum_{i=1}^{N} \alpha_i y_i \left(\sum_{j=1}^{N} \alpha_j y_j \phi(x_j)\right) \cdot \phi(x_i) - \sum_{i=1}^{N} \alpha_i + \sum_{i=1}^{N} \alpha_i \xi_i\right] \\ & = \frac{1}{2}\left(\sum_{i=1}^{N} \alpha_i y_i \phi(x_i)\right) \cdot \left(\sum_{i=j}^{N} \alpha_j y_j \phi(x_j)\right) + \sum_{i=1}^{N} \alpha_i - \left(\sum_{i=1}^{N} \alpha_i y_i \phi(x_i)\right) \cdot \left(\sum_{j=1}^{N} \alpha_j y_j \phi(x_j)\right) \\ & = \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\left(\sum_{i=1}^{N} \alpha_i y_i \phi(x_i)\right) \cdot \left(\sum_{i=j}^{N} \alpha_j y_j \phi(x_j)\right) \end{aligned}$$

Before we continue, lets rewrite the last expression in terms of matrices

$$\sum_{i=1}^{N} \alpha_i y_i \phi(x_i) = \mathbf{Z}^{\mathrm{T}} \boldsymbol{\alpha} \quad \text{where} \quad \mathbf{Z} = \begin{pmatrix} y_1\phi(x_{11}) & y_1\phi(x_{21}) & \cdots & y_1\phi(x_{N1}) \\ y_2\phi(x_{12}) & y_2\phi(x_{22}) & \cdots & y_2\phi(x_{N2}) \\ y_3\phi(x_{13}) & y_3\phi(x_{23}) & \cdots & y_3\phi(x_{N3}) \\ \vdots & \ddots & \ddots & \vdots \\ y_N\phi(x_{1N}) & y_N\phi(x_{2N}) & \cdots & y_N\phi(x_{NN}) \end{pmatrix} \quad \text{and} \quad \boldsymbol{\alpha} = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_N \end{pmatrix}$$

$$\sum_{i=1}^{N} \alpha_i = [1, 1, \cdots, 1]^T \boldsymbol{\alpha}$$

Therefore, the Lagrangian for the original problem can be expressed as:

$$L = [1, 1, \cdots, 1]^T \boldsymbol{\alpha} - \frac{1}{2} (\mathbf{Z}^{\mathrm{T}} \boldsymbol{\alpha})^T (\mathbf{Z}^{\mathrm{T}} \boldsymbol{\alpha})$$

$$= [1, 1, \cdots, 1]^T \boldsymbol{\alpha} - \frac{1}{2} \boldsymbol{\alpha}^T \mathbf{Z}\mathbf{Z}^{\mathrm{T}} \boldsymbol{\alpha}$$

$$= [1, 1, \cdots, 1]^T \boldsymbol{\alpha} - \frac{1}{2} \boldsymbol{\alpha}^T \mathbf{H} \boldsymbol{\alpha} \quad \text{where} \quad \mathbf{H} = \mathbf{Z}\mathbf{Z}^{\mathrm{T}} = diag(\mathrm{y}) \boldsymbol{\Phi}(\mathbf{X}) \boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}} diag(\mathrm{y})$$

$$\boldsymbol{\Phi}(\mathbf{X}) = \begin{pmatrix} \phi(x_{11}) & \phi(x_{21}) & \cdots & \phi(x_{N1}) \\ \phi(x_{12}) & \phi(x_{22}) & \cdots & \phi(x_{N2}) \\ \phi(x_{13}) & \phi(x_{23}) & \cdots & \phi(x_{N3}) \\ \vdots & \ddots & \ddots & \vdots \\ \phi(x_{1N}) & \phi(x_{2N}) & \cdots & \phi(x_{NN}) \end{pmatrix} \quad \text{and} \quad \mathrm{K} = \boldsymbol{\Phi}(\mathbf{X}) \boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}}$$

We can rewrite the above proble as a minimization problem with respect to $\boldsymbol{\alpha}$.

$$\min_{\boldsymbol{\alpha}} L = \frac{1}{2} \boldsymbol{\alpha}^T \mathbf{H} \boldsymbol{\alpha} + [-1, -1, \cdots, -1]^T \boldsymbol{\alpha} \quad \text{where} \quad \mathbf{f}^{\mathrm{T}} = [-1, -1, \cdots, -1]^T$$

$$= \frac{1}{2} \boldsymbol{\alpha}^T \mathbf{H} \boldsymbol{\alpha} + \mathbf{f}^{\mathrm{T}} \boldsymbol{\alpha}$$

This is identical to the separable case, except that the constraints are somewhat different. To see what these constraints are, we note that $\boldsymbol{\alpha_i} \geq \mathbf{0}$ is required because these are Lagrange multipliers. We also know that $\boldsymbol{\mu_i} \geq \mathbf{0}$ is required because these also are Lagrange multipliers.

When we combine this second condition with the result from (3), we get $C - \boldsymbol{\alpha_i} \geq \mathbf{0}$, which implies that $\boldsymbol{\alpha_i} \leq C$.

Therefore, we have to miminze the new Lagrangian with respect to the variables $\{\boldsymbol{\alpha_i}\}$ subject to

$$0 \leq \alpha_i \leq C$$

$$\sum_{i=1}^{N} \alpha_i y_i = 0$$

We can express the constraint $0 \leq \alpha_i \leq C$ as follows:

$$\mathbf{A}\boldsymbol{\alpha} \leq \mathbf{a} \quad \textbf{where}$$

$$A, \mathbf{a} = \begin{cases} \mathrm{I},\ \mathbf{a} = [C, C, \cdots, C]^T & \text{if} \quad \alpha_i \leq C \\ \gamma\mathrm{I},\ \mathbf{a} = [0, 0, \cdots, 0]^T & \text{if} \quad -\alpha_i \leq 0 \quad \textbf{where} \quad \gamma = -1 \end{cases}$$

$$A = \begin{pmatrix} \mathrm{I} \\ \gamma\mathrm{I} \end{pmatrix} \quad \textbf{and} \quad \mathbf{a} = \begin{pmatrix} \vec{\mathrm{C}} \\ \vec{0} \end{pmatrix}$$

Finally, we need to write an expression for the second constraint, that is, $\sum_{i=1}^{N} \alpha_i y_i = 0$

$$\mathrm{B}\boldsymbol{\alpha} = 0 \quad \textbf{where} \quad \mathrm{B} = \boldsymbol{y}^T$$

Problem 3

Yes - we can use a decision tree in this instance!

In a very simple case, we would require only one horizontal or vertical split to perfectly classify all the data. In this case, the slope of the decision boundary is $\mathbf{0}$.

Let us now consider the case of a positively sloped decision boundary. After our first horizontal and vertical split, we are left with 4 quadrants. The points in the upper left (2nd quadrant) and lower right (4th quadrant) will be correctly classified points.

As such, we are left with two quadrants (lower left and upper right) that contain points that still need to be classified.

If we repeat the above process, that is, creating additional horizontal and vertical splits in these quadrant, we will reach a stage where all points are correctly classified.

Given N data points and the space of $\mathbb{R}^2$ we know that the complexity is $O(log_2\ n^2)$ complexity. This simplifies to $O(log\ n)$. This is the depth of the tree.

Note: The use of a positive slope will not result in any loss of generality, for if the slope were negative, we would simply flip the points and follow the same logic.

Problem 4

There are two forms of non-linear separability.

Form 1: We have two points belonging to two different classes which ***lie*** on the same point $\mathbf{x}$ in the space. Where $\mathbf{x} = (x_1, x_2)$. In this instance, there is no tree which can be constructed that can separate these points. As such, we will always have 1 point that is misclassified.

Form 2: All points, with distinct class labels, reside on distinct points in the space. As such, we use the same logic employed in problem 3, however, we are now required to explore all resulting quadrants from the horizontal and vertical splits. It is *no longer* valid to assume that upper left and lower right quadrants correctly classify points. In the worst case, we perform these steps for *every* resulting quadrant. As such, this gives us a depth of:
$O(log_2\ n^{2^2}) = O(log_2\ n^4) = O(4 \cdot log_2\ n) = O(log\ n)$

Note: For a dimension d, the complexity for a linearly separable instance is $O(d \times log\ n)$. For the non-linearly separable instance it is $O(2^d \times log\ n)$

Problem 5

To answer this problem, suppose that we can express $W_i^{(T+1)}$ as follows (See Bishop p. 661, (14.24)):

$$W_i^{(T+1)} = \frac{W_i^{(T)}}{Z} e^{-y_i \alpha_T h_T(\mathbf{x_i})} \quad (1)$$

where *Z* represents some normalizing constant.

We know that the training error for the period $T+1$ is defined as $\epsilon_{T+1} = \sum_{i: yi \neq h_{T+1}(\mathbf{x_i})} W_i^{(T+1)}$, but before we calculate the sum over the misclassified weights in period $T+1$, it will be useful to compute $\sum_{i=1}^N W_i^{(T+1)}$ and we will assume that this sum is equal to 1.

$$\sum_{i=1}^N W_i^{(T+1)} = \sum_{i=1}^N \frac{W_i^{(T)}}{Z} e^{-y_i \alpha_T h_T(\mathbf{x_i})}$$

$$= \sum_{i:\, y_i = h_T(\mathbf{x_i})} \frac{W_i^{(T)}}{Z} e^{-\alpha_T} + \sum_{i:\, y_i \neq h_T(\mathbf{x_i})} \frac{W_i^{(T)}}{Z} e^{\alpha_T}$$

$$= e^{-\alpha_T} \sum_{i:\, y_i = h_T(\mathbf{x_i})} \frac{W_i^{(T)}}{Z} + e^{\alpha_T} \sum_{i:\, y_i \neq h_T(\mathbf{x_i})} \frac{W_i^{(T)}}{Z} \quad \text{substitute} \quad \alpha_T = \frac{1}{2} log \frac{1-\epsilon_T}{\epsilon_T}$$

$$= \frac{1}{e(log(\frac{1-\epsilon_T}{\epsilon_T})^{1/2})} \sum_{i:\, y_i = h_T(\mathbf{x_i})} \frac{W_i^{(T)}}{Z} + e\left(log\left(\frac{1-\epsilon_T}{\epsilon_T}\right)^{1/2}\right) \sum_{i:\, y_i \neq h_T(\mathbf{x_i})} \frac{W_i^{(T)}}{Z}$$

$$= \left(\frac{\epsilon_T}{1-\epsilon_T}\right)^{1/2} \sum_{i:\, y_i = h_T(\mathbf{x_i})} \frac{W_i^{(T)}}{Z} + \left(\frac{1-\epsilon_T}{\epsilon_T}\right)^{1/2} \sum_{i:\, y_i \neq h_T(\mathbf{x_i})} \frac{W_i^{(T)}}{Z}$$

Since, $\epsilon_T = \sum_{i:\, y_i \neq h_T(\mathbf{x_i})} W_i^{(T)}$ and we assumed that $\sum_{i=1}^{N} W_i^{(T)} = 1$, we have $1 - \epsilon_T = \sum_{i:\, y_i = h_T(\mathbf{x_i})} W_i^{(T)}$

.

With this last two pieces of information, we can express the ensemble loss as:

$$\sum_{i=1}^{N} W_i^{(T+1)} = \frac{1}{Z}\left[\left(\frac{\epsilon_T}{1-\epsilon_T}\right)^{1/2} \sum_{i:\, y_i = h_T(\mathbf{x_i})} W_i^{(T)} + \left(\frac{1-\epsilon_T}{\epsilon_T}\right)^{1/2} \sum_{i:\, y_i \neq h_T(\mathbf{x_i})} W_i^{(T)}\right]$$

$$= \frac{1}{Z}\left[\left(\frac{\epsilon_T}{1-\epsilon_T}\right)^{1/2} (1-\epsilon_T) + \left(\frac{1-\epsilon_T}{\epsilon_T}\right)^{1/2} \epsilon_T\right]$$

$$= \frac{1}{Z}\left[(\epsilon_T(1-\epsilon_T))^{1/2} + (\epsilon_T(1-\epsilon_T))^{1/2}\right]$$

$$= \frac{1}{Z}\left[2(\epsilon_T(1-\epsilon_T))^{1/2}\right]$$

Since we assumed that $\sum_{i=1}^{N} W_i^{(T+1)} = 1$, this implies that $Z = 2(\epsilon_T(1-\epsilon_T))^{1/2}$.

We can rewrite (1) to see the value that $W_i^{(T+1)}$ takes when we classify samples correctly and when we misclassify samples using again $\alpha_T = \frac{1}{2}\, log \frac{1-\epsilon_T}{\epsilon_T}$ and substituting our value for *Z*.

$$W_i^{(T+1)} = \begin{cases} \frac{W_i^{(T)}}{Z} e^{-\alpha_T} & \text{if} \quad y_i = h_T(\mathbf{x_i}) \\ \frac{W_i^{(T)}}{Z} e^{\alpha_T} & \text{if} \quad y_i \neq h_T(\mathbf{x_i}) \end{cases}$$

$$= \begin{cases} \frac{W_i^{(T)}}{Z} \left(\frac{\epsilon_T}{1-\epsilon_T}\right)^{1/2} & \text{if} \quad y_i = h_T(\mathbf{x_i}) \\ \frac{W_i^{(T)}}{Z} \left(\frac{1-\epsilon_T}{\epsilon_T}\right)^{1/2} & \text{if} \quad y_i \neq h_T(\mathbf{x_i}) \end{cases}$$

$$= \begin{cases} \frac{W_i^{(T)}}{2(\epsilon_T(1-\epsilon_T))^{1/2}} \left(\frac{\epsilon_T}{1-\epsilon_T}\right)^{1/2} & \text{if} \quad y_i = h_T(\mathbf{x_i}) \\ \frac{W_i^{(T)}}{2(\epsilon_T(1-\epsilon_T))^{1/2}} \left(\frac{1-\epsilon_T}{\epsilon_T}\right)^{1/2} & \text{if} \quad y_i \neq h_T(\mathbf{x_i}) \end{cases}$$

$$= \begin{cases} \frac{W_i^{(T)}}{2} \frac{1}{1-\epsilon_T} & \text{if} \quad y_i = h_T(\mathbf{x_i}) \\ \frac{W_i^{(T)}}{2} \frac{1}{\epsilon_T} & \text{if} \quad y_i \neq h_T(\mathbf{x_i}) \end{cases}$$

Finally, if we sum over the misclassified samples we get:

$$\begin{aligned}
\epsilon_{T+1} &= \sum_{i:y_i \neq h_{T+1}(\mathbf{x_i})} W_i^{(T+1)} \\
&= \frac{1}{2}\frac{1}{\epsilon_T} \sum_{i:y_i \neq h_T(\mathbf{x_i})} W_i^{(T)} \\
&= \frac{1}{2}\frac{1}{\epsilon_T}\epsilon_T \\
&= \frac{1}{2}
\end{aligned}$$

This result also implies that $1 - \epsilon_{T+1} = \sum_{i:y_i = h_{T+1}(\mathbf{x_i})} W_i^{(T+1)} = \frac{1}{2}$.

The fact that $\epsilon_{T+1} = \epsilon_T = \frac{1}{2}$ implies that in order to get the weights for the next generation, we just need to take the sum of the correctly classified weights and rescale these weights so that they add up to one half. However, the rule for updating the weights is not related to our choice for the classifier in the next period. In period $T+2$, we want to pick a classifier h_{T+2} that can better classify the previously misclassified points. Therefore, $h_{T+2} \neq h_{T+1}$.

Problem 6

Suppose that the base classifiers $h_1(x), \ldots, h_{m-1}(x)$ are fixed, as are their coefficients $\alpha_1, \ldots, \alpha_{m-1}$, and so we are minimizing only with respect to α_m and $h_m(x)$.

Separating off the contribution from base classifier $h_m(x)$, we can then write the error function in the form:

$$
\begin{aligned}
L(\mathrm{H}_m, X) &= \sum_{i=1}^{N} e^{-y_i \cdot [\mathrm{H}_{m-1}(\mathbf{x}_i) + \alpha_m h_m(\mathbf{x}_i)]} \\
&= \sum_{i=1}^{N} e^{-y_i \mathrm{H}_{m-1}(\mathbf{x}_i) - \alpha_m h_m(\mathbf{x}_i)} \\
&= \sum_{i=1}^{N} e^{-y_i \mathrm{H}_{m-1}(\mathbf{x}_i)} \cdot e^{-\alpha_m h_m(\mathbf{x}_i)} \\
&= \sum_{i=1}^{N} W_i^{(m-1)} e^{-y_i \alpha_m h_m(\mathbf{x}_i)} \quad \text{where} \quad W_i^{(m-1)} = e^{-y_i H_{m-1}(\mathbf{x}_i)} \\
&= \sum_{i:\, y_i = h_m(\mathbf{x}_i)} W_i^{(m-1)} e^{-\alpha_m} + \sum_{i:\, y_i \neq h_m(\mathbf{x}_i)} W_i^{(m-1)} e^{\alpha_m} \\
&= e^{-\alpha_m} \sum_{i:\, y_i = h_m(\mathbf{x}_i)} W_i^{(m-1)} + e^{\alpha_m} \sum_{i:\, y_i \neq h_m(\mathbf{x}_i)} W_i^{(m-1)} \\
&= e^{-\alpha_m} \left(\sum_{i=1}^{N} W_i^{(m-1)} - \sum_{i:\, y_i \neq h_m(\mathbf{x}_i)} W_i^{(m-1)} \right) + e^{\alpha_m} \sum_{i:\, y_i \neq h_m(\mathbf{x}_i)} W_i^{(m-1)} \\
&= (e^{\alpha_m} - e^{-\alpha_m}) \sum_{i:\, y_i \neq h_m(\mathbf{x}_i)} W_i^{(m-1)} + e^{-\alpha_m} \sum_{i=1}^{N} W_i^{(m-1)}
\end{aligned}
$$

Taking the derivative with respect to α_m, we get the following:

$$\frac{\delta L}{\delta \alpha_m} = e^{\alpha_m} \sum_{i:\, y_i \neq h_m(\mathbf{x_i})} W_i^{(m-1)} + \quad e^{-\alpha_m} \sum_{i:\, y_i \neq h_m(\mathbf{x_i})} W_i^{(m-1)} - \quad e^{-\alpha_m} \sum_{i=1}^{N} W_i^{(m-1)} = 0$$

$$= e^{\alpha_m} \sum_{i:\, y_i \neq h_m(\mathbf{x_i})} W_i^{(m-1)} \quad + \quad e^{-\alpha_m} \left(\sum_{i:\, y_i \neq h_m(\mathbf{x_i})} W_i^{(m-1)} - \quad \sum_{i=1}^{N} W_i^{(m-1)} \right) = 0$$

$$= e^{\alpha_m} \sum_{i:\, y_i \neq h_m(\mathbf{x_i})} W_i^{(m-1)} \quad - \quad e^{-\alpha_m} \sum_{i:\, y_i = h_m(\mathbf{x_i})} W_i^{(m-1)} = 0$$

$$\Rightarrow e^{\alpha_m} \sum_{i:\, y_i \neq h_m(\mathbf{x_i})} W_i^{(m-1)} \quad = \quad e^{-\alpha_m} \sum_{i:\, y_i = h_m(\mathbf{x_i})} W_i^{(m-1)}$$

$$\Rightarrow \alpha_m + log\left(\sum_{i:\, y_i \neq h_m(\mathbf{x_i})} W_i^{(m-1)} \right) = -\alpha_m + log\left(\sum_{i:\, y_i = h_m(\mathbf{x_i})} W_i^{(m-1)} \right)$$

$$\Rightarrow 2\alpha_m = log\left(\sum_{i:\, y_i = h_m(\mathbf{x_i})} W_i^{(m-1)} \right) - log\left(\sum_{i:\, y_i \neq h_m(\mathbf{x_i})} W_i^{(m-1)} \right)$$

$$\Rightarrow \alpha_m = \frac{1}{2}\, log\left(\frac{\sum_{i:\, y_i = h_m(\mathbf{x_i})} W_i^{(m-1)}}{\sum_{i:\, y_i \neq h_m(\mathbf{x_i})} W_i^{(m-1)}} \right)$$

Since, $\epsilon_m = \sum_{i:\, y_i \neq h_m(\mathbf{x_i})} W_i^{(m-1)}$ and we assumed that $\sum_{i=1}^{N} W_i^{(m-1)} = 1$, we have $1 - \epsilon_m = \sum_{i:\, y_i = h_m(\mathbf{x_i})} W_i^{(m-1)}$.

We get that the α_m that minimizes the empirical exponential loss is:

$$\alpha_m = \frac{1}{2}\, log\left(\frac{1 - \epsilon_m}{\epsilon_m} \right)$$