Solutions Exercise 5

Question 1

a) See Figure 1

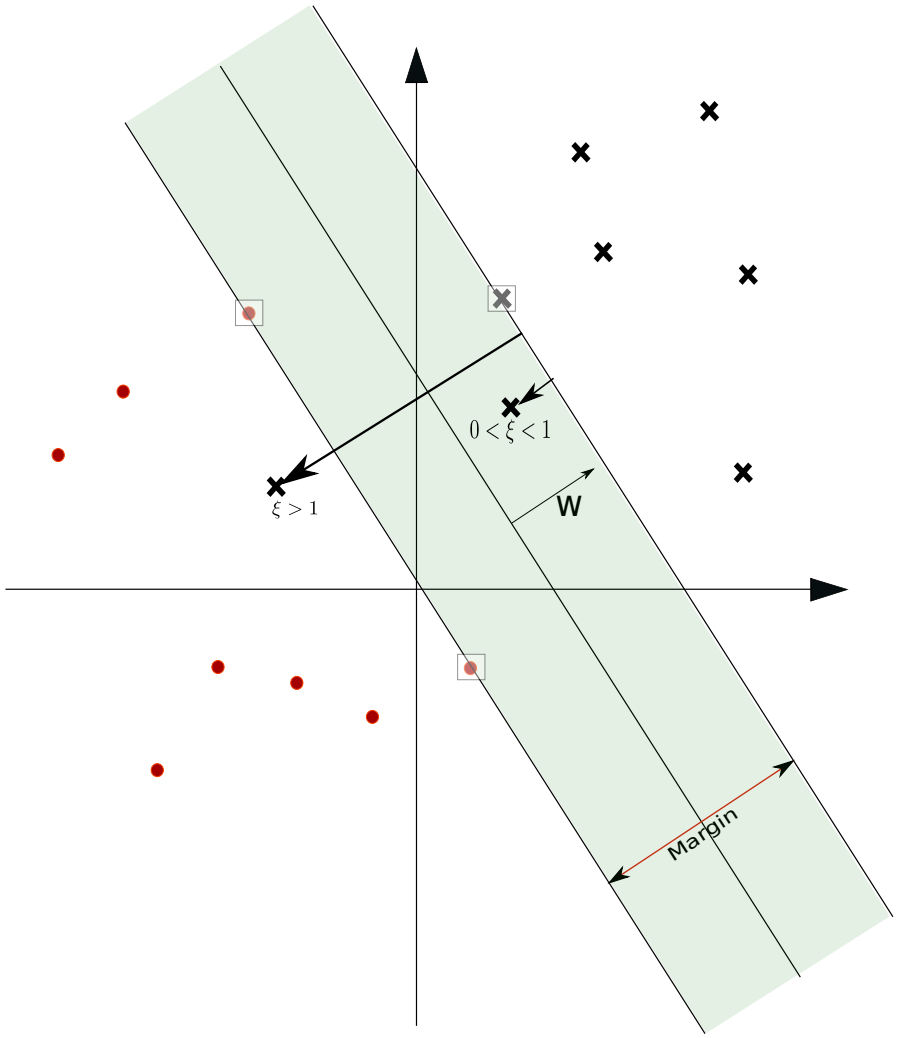

Figure 1: The plot illustrates the solution to part (a) of question 1. The support vectors are marked using square boxes and the data samples with $\xi_i > 1$ and $0 < \xi_i < 1$ are shown as well.

b) In order to optimize the augmented loss with Lagrange multipliers as shown in equation 1,

$$L(\mathbf{w}, \alpha, b) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_i \alpha_i[y_i(\mathbf{w}^T x_i + b) - 1] \tag{1}$$

we make use of the quantity $\min_{\mathbf{w},b} L(\mathbf{w}, \alpha, b)$, which is the dual form of the problem. When optimizing problem using dual form, we first minimize $L(\mathbf{w}, \alpha, b)$ with respect to $\mathbf{w}, b$ followed by optimizing (maximizing) with respect to α. Minimizing $L(\mathbf{w}, \alpha, b)$ can be done by solving $\nabla_{\mathbf{w}} L(\mathbf{w}, \alpha, b) = 0$ and $\nabla_b L(\mathbf{w}, \alpha, b) = 0$. First let us solve $\nabla_{\mathbf{w}} L(\mathbf{w}, \alpha, b) = 0$,

$$\begin{aligned} \nabla_{\mathbf{w}} L(\mathbf{w}, \alpha, b) = \mathbf{w} - \sum_i \alpha_i y_i x_i = 0 \\ \implies \mathbf{w} = \sum_i \alpha_i y_i x_i \end{aligned} \tag{2}$$

Followed by solution to $\nabla_b L(\mathbf{w}, \alpha, b) = 0$:

$$\nabla_b L(\mathbf{w}, \alpha, b) = \sum_i \alpha_i y_i = 0 \tag{3}$$

Using equations 2 and 3; we can rewrite equation 1 as

$$\begin{aligned} L(\mathbf{w}, \alpha, b) &= \frac{1}{2}(\sum_i \alpha_i y_i x_i)^T(\sum_j \alpha_j y_j x_j) - \sum_i \alpha_i[y_i(\sum_j \alpha_j y_j x_j)^T x_i + b)] + \sum_i \alpha_i \\ &= \frac{1}{2}\sum_{i,j} y_i y_j \alpha_i \alpha_j (x_i)^T x_j - \sum_{i,j} y_i y_j \alpha_i \alpha_j (x_i)^T x_j - b\sum_i \alpha_i y_i + \sum_i \alpha_i \\ &= \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} y_i y_j \alpha_i \alpha_j (x_i)^T x_j - b\sum_i \alpha_i y_i \end{aligned}$$

Since $\sum_i \alpha_i y_i = 0$, we get

$$L(\mathbf{w}, \alpha, b) = \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} y_i y_j \alpha_i \alpha_j (x_i)^T x_j \tag{4}$$

Now, the optimization problem becomes

$$\begin{aligned} &\max_{\alpha} \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} y_i y_j \alpha_i \alpha_j (x_i)^T x_j \\ &\sum_i \alpha_i y_i = 0 \end{aligned} \tag{5}$$

Finally, the condition $\alpha_i \geq 0$ comes from the fact that Lagrange multipliers are positive. For more details on optimization using Lagrange multipliers refer this link.

c) The optimization problem, given in equation 5, can also be written as follows

$$\max_{\alpha} \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} y_i y_j \alpha_i \alpha_j K(x_i, x_j)$$
$$\sum_i \alpha_i y_i = 0$$

where, we replaced the inner product $x_i^T x_j$ with Kernel $K(x_i, x_j)$. For classification we have

$$\begin{aligned} y &= \mathbf{w}^T x + b \\ &= \sum_i \alpha_i y_i \langle x_i, x \rangle + b \\ &= \sum_i \alpha_i y_i K(x_i, x) + b \end{aligned}$$

where we made use of equation 2 and for any x its corresponding class is given by $\text{sign}(\sum_i \alpha_i y_i K(x_i, x) + b)$

Question 2

In this exercise, we were asked to construct a random forest with one decision node on MNIST dataset. We split the data into training and validation sets, of size 5000 each. The Figure 2 shows the classification errors for these datasets. The Table 1 describes the confusion matrix of the validation set. The prediction accuracy of the random forest obtained can be improved if we used boosting instead of bagging; it should increase the accuracy.

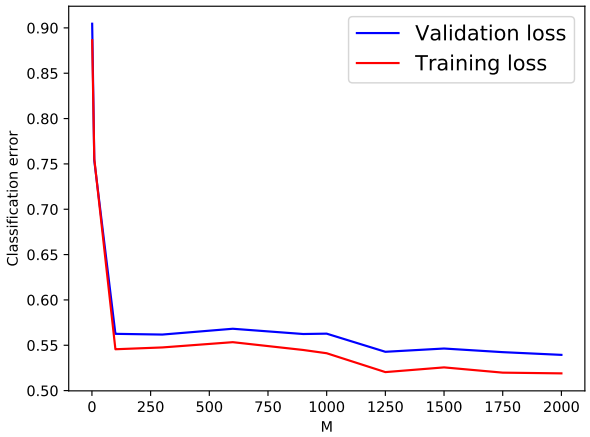

Figure 2: The plot illustrates training and validation errors given by red and blue curves respectively, where x-axis corresponds to the stumps and y-axis to the classification error. In the plot, we observe that as we increase the number of stumps the error decreases, which is a valid observation. The error stabilizes around 0.5-0.55 mark, even after using more stumps. Furthermore, the training error is computed over all 5000 samples even though only 100 were used for determining the threshold for each stump – what we computed here as training error is dominated by "out-of-bag" error, the error for "training samples" that were actually not used for training the stump and hence it is understandable that the loss is not much better than the real validation loss.

Confusion Matrix	0	1	2	3	4	5	6	7	8	9
0	479	0	23	21	2	37	20	4	28	14
1	12	559	124	86	75	192	40	54	264	57
2	0	0	289	4	0	0	8	0	2	1
3	1	3	10	369	0	84	0	0	55	3
4	1	0	10	6	343	60	6	7	57	129
5	0	0	0	0	0	0	0	0	0	0
6	23	1	35	9	4	20	438	0	17	0
7	6	1	12	44	18	34	1	455	17	239
8	0	0	0	0	0	2	0	0	40	0
9	0	0	0	0	3	0	0	0	2	40

Table 1: The table shows the confusion matrix of the random forest learnt using 2000 stumps on the validation dataset, where rows correspond to predicted labels and columns to true labels. We can see that it can predict the digits 0,2, and 6 with good accuracy. On the other hand it finds it very difficult to classify the digits 5, 8, and 9; furthermore, the digit 1 gets confused with all the remaining digits with significant probability.