Optimization and Machine Learning, Spring 2020
Homework 4
(Due Tuesday, May 12 at 11:59pm (CST))

1. Given a training dataset $S = \{(x_i, y_i)\}_{i=1}^n$, in which $x_i \in \mathbb{R}^d$ and $y_i \in \{-1, 1\}$ denote the i-th sample and the i-th label, respectively. Suppose that we use S to train a machine learning model based on Adaboost. At the end of the t-th iteration ($t = 1, 2, ..., T$), the importance of the i-th ($i = 1, 2, ..., n$) sample x_i is reweighted as

$$D_i^{(t+1)} = D_i^{(t)} \exp\left(-\alpha_t y_i h_t(x_i)\right),$$

where α_t is the weight of the t-th weakly binary classifier h_t, i.e.,

$$\alpha_t = \frac{1}{2}\ln(\frac{1-\epsilon_t}{\epsilon_t}), \text{ with } \epsilon_t = \sum_{i=1}^{n} D_i^{(t)} \mathbb{1}(y_i \neq h_t(x_i)).$$

To classify an arbitrary test sample x, we calculate $f(x) = \sum_{t=1}^{T} \alpha_t h_t(x)$ and then return its sign. Now let's show that if every learner h_t ($\forall t$) achieves 51% classification accuracy (that is, only slightly better than random guessing), AdaBoost will converge to zero training error.

   (a) Let's change the update rule so that the weights of each iteration are normalized, that is, $\sum_{i=1}^{n} D_i^{(t)} = 1$ ($\forall t$). In this sense, we can treat the weights as a discrete probability distribution over the sample points. Hence we rewrite the update rule by

$$D_i^{(t+1)} = \frac{D_i^{(t)} \exp(-\alpha_t y_i h_t(x_i))}{Z_t},$$

   where Z_t is the normalization factor, $\forall t$. Please show that the following formula is satisfied,

$$Z_t = 2\sqrt{\epsilon_t(1-\epsilon_t)}.$$

   (5 points)
   Solution:

$$\begin{aligned} Z_t &= \sum_{i=1}^{n} D_i^{(t)} \exp(-\alpha_t y_i h_t(x_i)) \\ &= \sum_{y_i \neq h_t(x_i)} D_i^{(t)} \exp(\alpha) + \sum_{y_i = h_t(x_i)} D_i^{(t)} \exp(-\alpha) \\ &= \epsilon_t (\frac{1-\epsilon_t}{\epsilon_t})^{1/2} + (1-\epsilon_t)(\frac{1-\epsilon_t}{\epsilon_t})^{-1/2} \\ &= 2\sqrt{\epsilon_t(1-\epsilon_t)} \end{aligned}$$

   (b) Assume that the initial weights follow uniform distribution, i.e.,

$$D_1^{(1)} = D_2^{(1)} = \cdots = D_n^{(1)} = \frac{1}{n}.$$

   Please show that

$$D_i^{(t)} = \frac{1}{n \prod_{t=1}^{T} Z_t} e^{-y_i f(x_i)},$$

   where $i = 1, 2, ..., n$ and $t = 2, 3, ..., T$. (5 points)
   Solution:

$$
\begin{aligned}
D_i^{(T)} &= \frac{D_i^{(T-1)} \exp\left(-\alpha_{T-1} y_i h_{T-1}(x_i)\right)}{Z_{T-1}} \\
&= \frac{D_i^{(T-2)} \exp\left(-\alpha_{T-1} y_i h_{T-1}(x_i)\right) \exp\left(-\alpha_{T-2} y_i h_{T-2}(x_i)\right)}{Z_{T-1} Z_{T-2}} \\
&\cdots \\
&= \frac{D_i^{(1)} e^{-y_i \sum_{t=1}^{T-1} \alpha_t h_t(x_i)}}{\prod_{t=1}^{T-1} Z_t} \\
&= \frac{e^{-y_i f(x_i)}}{n \prod_{t=1}^{T-1} Z_t}
\end{aligned}
$$

(c) Let m be the number of sample points that Adaboost classifies incorrectly. Please show that

$$
\sum_{i=1}^{n} e^{-y_i f(x_i)} \geq m.
$$

(5 points)
Solution:

$$
\begin{aligned}
\sum_{i=1}^{n} e^{-y_i f(x_i)} &= \sum_{y=f(x_i)} e^{-1} + \sum_{y \neq f(x_i)} e^{1} \\
&= me + (n-m)e^{-1} \\
&= ne^{-1} + (e - e^{-1})m \\
&\geq m
\end{aligned}
$$

(d) Based on the results in (a), (b), and (c), please show that once $\epsilon_t \leq 0.49$ is satisfied for every learner h_t ($\forall t$), then we have $m \to 0$ as $T \to \infty$. (5 points)
Solution:

$$
\begin{aligned}
&Z_t = 2\sqrt{\epsilon_t(1-\epsilon_t)} < 1 \\
&T \to \infty, \sum_{i=1}^{n} e^{-y_i f(x_i)} = n \prod_{t=1}^{T-1} Z_t \to 0 \\
&m \leq \sum_{i=1}^{n} e^{-y_i f(x_i)} \to 0
\end{aligned}
$$

2. Suppose that we are interested in learning a classifier, such that at any turn of a game we can pose a question, like "should I attack this ant hill now?", and get an answer. That is, we want to build a classifier which we can feed some features on the current game state, and get the output "attack" or "don't attack". There are many possible ways to define what the action "attack" means, but for now let's define it as sending all friendly ants that can see the ant hill under consideration towards it.

   Let's recall the AdaBoost algorithm described in class. Its input is a dataset $\{(x_i, y_i)\}_{i=1}^{n}$, with x_i being the i-th sample, and $y_i \in \{-1, 1\}$ denoting the i-th label, $i = 1, 2, ..., n$. The features might be composed of a count of the number of friendly ants that can see the ant hill under consideration, and a count of the number of enemy ants these friendly ants can see. For example, if there were 10 friendly ants that could see a particular ant hill, and 5 enemy ants that the friendly ants could see, we would have:

$$
x_1 = \begin{bmatrix} 10 \\ 5 \end{bmatrix}.
$$

   The label of the example x_1 is $y_1 = 1$, once the friendly ants were successful in razing the enemy ant hill, and $y_1 = 0$ otherwise. We could generate such examples by running a greedy bot (or any other opponent bot) against a bot that we make periodically try to attack an enemy ant hill. Each time this bot tries the attack, we record (say, after 20 turns or some other significant amount of time) whether the attack was successful or not.

(a) Let ϵ_t denote the error of a weak classifier h_t:

$$\epsilon_t = \sum_{i=1}^{n} D_i^{(t)} \mathbb{1}(y_i \neq h_t(x_i)).$$

In the simple "attack" / "don't attack" scenario, suppose that we have implemented the following six weak classifiers:

$$h^{(1)}(x_i) = 2 * \mathbb{1}(x_{i1} \geq 2) - 1, \qquad h^{(4)}(x_i) = 2 * \mathbb{1}(x_{i2} \leq 2) - 1,$$
$$h^{(2)}(x_i) = 2 * \mathbb{1}(x_{i1} \geq 5) - 1, \qquad h^{(5)}(x_i) = 2 * \mathbb{1}(x_{i2} \leq 5) - 1,$$
$$h^{(3)}(x_i) = 2 * \mathbb{1}(x_{i1} \geq 10) - 1, \qquad h^{(6)}(x_i) = 2 * \mathbb{1}(x_{i2} \leq 10) - 1.$$

Given ten training data points ($n = 10$) as shown in Fig. 1, please show that what is the minimum value

i	x_{i1}	x_{i2}	y_i
1	3.5	0.5	1
2	2.5	1.5	1
3	5.5	5.5	1
4	7.5	7.5	1
5	6.5	9.5	1
6	1.5	2.5	-1
7	4.5	4.5	-1
8	3.5	7.5	-1
9	1.5	11.5	-1
10	6.5	11.5	-1

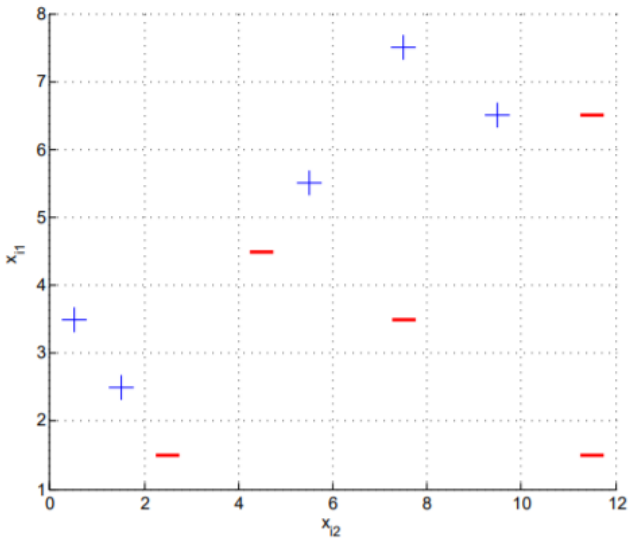

Figure 1: The training data in (a).

of ϵ_1 and which of $h^{(1)}, ..., h^{(6)}$ achieve this value? Note that there may be multiple classifiers that all have the same ϵ_1. You should list all classifiers that achieve the minimum ϵ_1 value. (5 points)
Solution:
The value of ϵ_1 for each of the classifiers is: $3/10, 3/10, 5/10, 3/10, 5/10$, and $3/10$. So, the minimum value is $3/10$ and classifiers 1, 2, 4, and 6 achieve this value.

(b) For all the questions in the remainder of this section, let h_1 denote $h^{(1)}$ chosen in the first round of boosting. (That is, $h^{(1)}$ was the classifier that achieved the minimum ϵ_1.)

(1) What is the value of α_1 (the weight of this first classifier h_1)? Keep in mind that the log in the formula for α_t is a natural log (base e). (5 points)
Solution:
Plugging into the formula for α we get: $\alpha_1 = \frac{1}{2} \log \frac{1 - \epsilon_1}{\epsilon_1} = \frac{1}{2} \log \frac{7}{3} = 0.4236$

(2) What should Z_t be in order to make sure the distribution $D^{(t+1)}$ is normalized correctly? That is, derive the formula of Z_t in terms of $D^{(t)}$, α_t, h_t, and $\{(x_i, y_i)\}_{i=1}^n$, that will ensure $\sum_{i=1}^n D_i^{(t+1)} = 1$. (5 points)
Solution:

$$Z_t = \sum_{k=1}^{n} D_T(k) \exp(-\alpha_t y_k h_t(x_k))$$

(3) Which points will increase in significance in the second round of boosting? That is, for which points will we have $D_i^{(1)} < D_i^{(2)}$? What are the values of $D^{(2)}$ for these points? (5 points)

Solution:
The points that $h(1)$ misclassifies will increase in weight. These are the points $i = 7, 8, 10$ from the data table. Their new weight under D_2 will be:

$$\begin{aligned} D_2(i) &= \frac{D_t(i)\exp(-\alpha_t y_i h_t(x_i)}{Z_t} \\ &= \frac{\exp\{0.4236\}}{3 * \exp\{0.4236\} + 7 * \exp\{-0.4236\}} \\ &= \frac{1}{6} \end{aligned}$$

(4) In the second round of boosting, the weights on the points will be different, and thus the error ϵ_2 will also be different. Which of $h^{(1)}, ..., h^{(6)}$ will minimize ϵ_2? (Which classifier will be selected as the second weak classifier h_2?) What is its value of ϵ_2? (5 points)
Solution:
$h^{(4)}$ will be chosen.

Classifier	ϵ_2
$h^{(1)}$	$1/2$
$h^{(2)}$	$1/6 + 2/14 = 13/42$
$h^{(3)}$	$5/14 = 0.3571$
$h^{(4)}$	$3/14$
$h^{(5)}$	$1/6 + 4/14 = 19/42$
$h^{(6)}$	$2/6 + 1/14 = 17/42$

(5) What will the average error of the final classifier H be, if we stop after these two rounds of boosting? That is, if $H(x) = \text{sign}(\alpha_1 h_1(x) + \alpha_2 h_2(x))$, what will the training error $\epsilon = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}(y_i \neq h(x_i))$ be? Is this more, less, or the same as the error we would get, if we just used one of the weak classifiers instead of this final classifier H? (5 points)
Solution:
The classifier after two rounds is:

$$h(x) = sign(0.5\log(7/3)h^{(1)}(x) + 0.5\log(11/3)h^{(4)}(x))$$

Since $log(11/3) > log(7/3)$ the classifier h will always go with the guess made by $h(4)$. So, it will not do any better than the error we could get using a single weak classifier, $\epsilon = 3/10$. More rounds of boosting are necessary before the interplay of specific settings of the α becomes relevant and allows us to do better than a single weak classifier.

3. Given valid kernels $k_1(\mathbf{x}, \mathbf{x}')$ and $k_2(\mathbf{x}, \mathbf{x}')$, please verify the following new kernels will also be valid:

(a) $k(\mathbf{x}, \mathbf{x}') = f(\mathbf{x})k_1(\mathbf{x}, \mathbf{x}')f(\mathbf{x}')$, where $f(\cdot)$ is any function. (2 points)
(b) $k(\mathbf{x}, \mathbf{x}') = q(k_1(\mathbf{x}, \mathbf{x}'))$, where $q(\cdot)$ is a polynomial with nonnegative coefficients. (3 points)
(c) $k(\mathbf{x}, \mathbf{x}') = \exp(k_1(\mathbf{x}, \mathbf{x}'))$. (5 points)
(d) $k(\mathbf{x}, \mathbf{x}') = \mathbf{x}^\top \mathbf{A}\mathbf{x}'$, where $\mathbf{A}$ is a symmetric positive semi-definite matrix. (5 points)

Solution:

(a) Since $k_1(\mathbf{x}, \mathbf{x}')$ is a valid kernel, there must exist a feature vector $\phi(\mathbf{x})$ such that

$$k_1(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^\top \phi(\mathbf{x}').$$

Then we can rewrite the given kernel as

$$\begin{aligned} k(\mathbf{x}, \mathbf{x}') &= f(\mathbf{x})\phi(\mathbf{x})^\top \phi(\mathbf{x}')f(\mathbf{x}') \\ &= \mathbf{v}(\mathbf{x})^\top \mathbf{v}(\mathbf{x}'), \end{aligned}$$

where $\mathbf{v}(\mathbf{x}) \triangleq f(\mathbf{x})\phi(\mathbf{x})$. We can see that the kernel can be rewritten as the scalar product of feature vectors, and hence is a valid kernel.

(b) Suppose $q(x) = \sum_{i=1}^{n} a_n x^n, \forall a_n \geq 0$, then the kernel can be expressed as

$$k(\mathbf{x}, \mathbf{x}') = \sum_{i=1}^{n} a_n \left(k_1(\mathbf{x}, \mathbf{x}')\right)^n .$$

We focus on the i-th term of the kernel, which is $a_n \left(k_1(\mathbf{x}, \mathbf{x}')\right)^n$. Since $k_1(\mathbf{x}, \mathbf{x}')$ is a valid kernel, the product of kernels is also a valid kernel. Hence, $a_n \left(k_1(\mathbf{x}, \mathbf{x}')\right)^n$ is a valid kernel. With the fact that the sum of kernels is a valid kernel, the original kernel is valid.

(c) Let $\mathbf{K}$ be the Gram matrix. The (i, j)-th entry of $\mathbf{K}$ is defined by $\mathbf{K}_{i,j} \triangleq k(\mathbf{x}_i, \mathbf{x}_j)$. Since $k_1(\mathbf{x}, \mathbf{x}')$ is a valid kernel, we have $k_1(\mathbf{x}, \mathbf{x}') = k_1(\mathbf{x}', \mathbf{x})$. Hence, the Gram matrix $\mathbf{K}$ is symmetric. In addition, $k(\mathbf{x}, \mathbf{x}')$ is an exponential function, which leads to $k(\mathbf{x}, \mathbf{x}')$ is always greater than zero. Therefore, the Gram matrix $\mathbf{K}$ is positive definite. Applying, Mercer's condition, $k(\mathbf{x}, \mathbf{x}')$ is a valid kernel.

(d) Since $\mathbf{A}$ is a symmetric positive semi-definite matrix, we can decompose $\mathbf{A}$ as $\mathbf{A} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^\top$ where $\mathbf{Q}$ is an orthogonal matrix and $\mathbf{\Lambda}$ is a diagonal matrix. When $\mathbf{A}$ is positive semi-definite, the entries of $\mathbf{\Lambda}$ are nonnegative. Hence, we can rewrite the kernel as

$$\begin{aligned} k(\mathbf{x}, \mathbf{x}') &= \mathbf{x}^\top \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^\top \mathbf{x}' \\ &= (\mathbf{\Lambda}^{1/2}\mathbf{Q}^\top \mathbf{x})^\top (\mathbf{\Lambda}^{1/2}\mathbf{Q}^\top \mathbf{x}') \\ &= \mathbf{\Phi}(\mathbf{x})^\top \mathbf{\Phi}(\mathbf{x}'), \end{aligned}$$

where $\mathbf{\Phi}(\mathbf{x}) \triangleq \mathbf{\Lambda}^{1/2}\mathbf{Q}^\top \mathbf{x}$. We can see that the kernel can be rewritten as the scalar product of feature vectors, and hence is a valid kernel.

4. Consider the space of all possible subsets A of a given fixed set D. Show that the kernel function $k(A_1, A_2) = 2^{|A_1 \cap A_2|}$ corresponds to an inner product in a feature space of dimensionality $2^{|D|}$ defined by the mapping $\phi(A)$ where A is a subset of D and the element $\phi_U(A)$, indexed by the subset U, is given by

$$\phi_U(A) = \begin{cases} 1, & \text{if } U \subseteq A; \\ 0, & \text{otherwise.} \end{cases}$$

Here $U \subseteq A$ denotes that U is either a subset of A or is equal to A. (10 points)

Solution:
First of all, we consider the case of $A = D$. In that case, $\phi(D)$ must be defined. This will map to a vector $2^{|D|}$ 1s, one for each possible subset of D, including D itself as well as the empty set. Now we consider another case of $A \subset D$, $\phi(A)$ will have 1s in all positions that correspond to subsets of A and 0s in all other positions. Therefore, $\phi(A_1)^\top \phi(A_2)$ will count the number of subsets shared by A_1 and A_2. However, this can just as well be obtained by counting the number of elements in the intersection of A_1 and A_2, and then raising 2 to this number, which is exactly $2^{|A_1 \cap A_2|}$ does.

5. Suppose we have a data set of input vectors $\{\mathbf{x}_n\}$ with corresponding target values $t_n \in \{-1, 1\}$, and suppose that we model the density of input vectors within each class separately using a Parzen kernel density estimator which is defined as follows

$$p(\mathbf{x}|t) = \frac{1}{N_t} \sum_{n=1}^{N} \frac{1}{Z_k} k\left(\mathbf{x}, \mathbf{x}_n\right) \delta(t, t_n)$$

where $k(\mathbf{x}, \mathbf{x}')$ is a valid kernel, Z_k is the normalization constant for the kernel and $\delta(t, t_n)$ equals 1 if $t = t_n$ and 0 otherwise.

(a) Write down the minimum misclassification-rate decision rule assuming the two classes have equal prior probability. (3 points)

(b) Show that, if the kernel is chosen to be $k(\mathbf{x}, \mathbf{x}') = \mathbf{x}^\top \mathbf{x}'$, then the classification rule reduces to simply assigning a new input vector to the class having the closest mean. (4 points)

(c) Show that, if the kernel takes the form $k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^\top \phi(\mathbf{x}')$, that the classification is based on the closest mean in the feature space $\phi(\mathbf{x})$. (4 points)

Solution:

(a) From Bayes' theorem we have

$$p(t|\mathbf{x}) \propto p(\mathbf{x}|t)p(t)$$

where, from the stem,

$$p(\mathbf{x}|t) = \frac{1}{N_t}\sum_{n=1}^{N}\frac{1}{Z_k}k\left(\mathbf{x},\mathbf{x}_n\right)\delta\left(t,t_n\right).$$

The minimum misclassification-rate is achieved if, for each new input vector, $\tilde{\mathbf{x}}$, we chose $\tilde{t}$ to maximize $p(\tilde{t}|\tilde{\mathbf{x}})$. With equal class priors, this is equivalent to maximizing $p(\tilde{\mathbf{x}}|\tilde{t})$ and thus

$$\tilde{t} = \begin{cases} +1 & \text{iff } \frac{1}{N_{+1}}\sum_{i:t_i=+1}k\left(\tilde{\mathbf{x}},\mathbf{x}_i\right) \geqslant \frac{1}{N_{-1}}\sum_{j:t_j=-1}k\left(\tilde{\mathbf{x}},\mathbf{x}_j\right) \\ -1 & \text{otherwise} \end{cases}$$

Here we have dropped the factor $1/Z_k$ since it only acts as a common scaling factor. Using the encoding scheme for the label, this classification rule can be written in the more compact form

$$\tilde{t} = \text{sign}\left(\sum_{n=1}^{N}\frac{t_n}{N_{t_n}}k\left(\tilde{\mathbf{x}},\mathbf{x}_n\right)\right).$$

(b) Now we take $k(\mathbf{x},\mathbf{x}') = \mathbf{x}^\top\mathbf{x}_n$, which results in the kernel density

$$p(\mathbf{x}|t=+1) = \frac{1}{N_{+1}}\sum_{n:t_n=+1}\mathbf{x}^{\mathrm{T}}\mathbf{x}_n = \mathbf{x}^{\mathrm{T}}\bar{\mathbf{x}}^{+}.$$

Here, the sum in the middle expression runs over all vectors $\mathbf{x}_n$ for which $t_n = +1$ and $\bar{x}^+$ denotes the mean of these vectors, with the corresponding definition for the negative class. Note that this density is improper, since it cannot be normalized. However, we can still compare likelihoods under this density, resulting in the classification rule

$$\tilde{t} = \begin{cases} +1 & \text{if } \tilde{\mathbf{x}}^{\mathrm{T}}\bar{\mathbf{x}}^{+} \geqslant \tilde{\mathbf{x}}^{\mathrm{T}}\bar{\mathbf{x}}^{-} \\ -1 & \text{otherwise} \end{cases}$$

(c) The same argument in (a) could also apply that the feature space is $\phi(\mathbf{x})$.

6. The problem of maximizing margin can be converted into an following equivalent problem

$$\begin{aligned} &\underset{\mathbf{w},b}{\text{argmin}} \ \frac{1}{2}\|\mathbf{w}\|^2 \\ &\text{subject to } t_n(\mathbf{w}^\top\phi(\mathbf{x}_n)+b) \geq 1, \quad n = 1,\cdots,N. \end{aligned}$$

where $\phi(\mathbf{x})$ is a fixed feature-space transformation.

(a) By introducing Lagrange multipliers $\{a_n\}$, please give the Lagrangian function and the dual representation of the maximum margin problem. (8 points)

(b) Please show that the value ρ of the margin for the maximum-margin hyperplane is given by

$$\frac{1}{\rho^2} = \sum_{n=1}^{N} a_n.$$

(Hint: $\{a_n\}$ can be obtained by solving the dual representation of the maximum margin problem.) (6 points)

Solution:

(a) The Lagrangian function is given by

$$L(\mathbf{w},b,\mathbf{a}) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{n=1}^{N}a_n\left\{t_n\left(\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}\left(\mathbf{x}_n\right)+b\right)-1\right\}. \tag{1}$$

The dual representation of the maximum margin problem is given by

$$\max_{\mathbf{a}} \widetilde{L}(\mathbf{a}) = \sum_{n=1}^{N} a_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} a_n a_m t_n t_m k(\mathbf{x}_n, \mathbf{x}_m) \tag{2}$$

$$\text{subject to } a_n \geq 0, \quad n = 1, \cdots, N, \tag{3}$$

$$\sum_{n=1}^{N} a_n t_n = 0, \tag{4}$$

where $k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^\top \phi(\mathbf{x}')$.

(b) Let the value of the margin ρ be $1/\|\mathbf{w}\|$ and so $1/\rho^2 = \|\mathbf{w}\|^2$. From the KKT conditions of the dual problem which is

$$\begin{aligned} a_n &\geqslant 0 \\ t_n y(\mathbf{x}_n) - 1 &\geqslant 0 \\ a_n \{t_n y(\mathbf{x}_n) - 1\} &= 0, \end{aligned}$$

we see that, for the maximum margin solution, the second term of (1) vanishes and so we have

$$L(\mathbf{w}, b, \mathbf{a}) = \frac{1}{2}\|\mathbf{w}\|^2. \tag{5}$$

By setting the derivatives of (1) with respect to $\mathbf{w}$ and b equal to zero, we obtain the following two conditions

$$\mathbf{w} = \sum_{n=1}^{N} a_n t_n \phi(\mathbf{x}_n) \tag{6}$$

$$0 = \sum_{n=1}^{N} a_n t_n. \tag{7}$$

Using (5) together with (6), the dual (2) can be written as

$$\frac{1}{2}\|\mathbf{w}\|^2 = \sum_{n}^{N} a_n - \frac{1}{2}\|\mathbf{w}\|^2,$$

from which the desired result follows.