$$\max_{\gamma,\boldsymbol{\lambda},\lambda_0} \gamma \quad \text{s.t.} \quad y_i \frac{\boldsymbol{\lambda}^T \mathbf{x}_i + \lambda_0}{||\boldsymbol{\lambda}||} \geq \gamma \quad i = 1 \dots n$$

$$\min_{\boldsymbol{\lambda},\lambda_0} \frac{1}{2} ||\boldsymbol{\lambda}||_2^2 \quad \text{s.t.} \quad y_i(\boldsymbol{\lambda}^T \mathbf{x}_i + \lambda_0) - 1 \geq 0 \quad i = 1 \dots n$$

$$\mathcal{L}([\boldsymbol{\lambda}, \lambda_0], \boldsymbol{\alpha}) = \frac{1}{2} \sum_{j=1}^{n} \lambda_j^2 + \sum_{i=1}^{n} \alpha_i \left[-y_i(\boldsymbol{\lambda}^T \mathbf{x}_i + \lambda_0) + 1 \right]$$

$$\nabla_{\boldsymbol{\lambda}} \mathcal{L}([\boldsymbol{\lambda}, \lambda_0], \boldsymbol{\alpha}) = \boldsymbol{\lambda} - \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i = \mathbf{0} \Longrightarrow \boldsymbol{\lambda} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i.$$

$$\frac{\partial}{\partial \lambda_0} \mathcal{L}([\boldsymbol{\lambda}, \lambda_0], \boldsymbol{\alpha}) = -\sum_{i=1}^{n} \alpha_i y_i = 0 \Longrightarrow \sum_{i=1}^{n} \alpha_i y_i = 0.$$

$$\begin{aligned} \alpha_i &\geq 0 \quad \forall i && \text{(dual feasibility)} \\ \alpha_i \left[-y_i(\boldsymbol{\lambda}^T \mathbf{x}_i + \lambda_0) + 1 \right] &= 0 \quad \forall i && \text{(complementary slackness)} \\ -y_i(\boldsymbol{\lambda}^T \mathbf{x}_i + \lambda_0) + 1 &\leq 0. && \text{(primal feasibility)} \end{aligned}$$

$$\mathcal{L}(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,k} \alpha_i \alpha_k y_i y_k \mathbf{x}_i^T \mathbf{x}_k \quad \text{s.t.} \quad \begin{cases} \alpha_i \geq 0 \quad i = 1 \dots n \\ \sum_{i=1}^{n} \alpha_i y_i = 0 \end{cases}$$

$$\min_{\boldsymbol{\lambda},\lambda_0,\xi} \frac{1}{2} ||\boldsymbol{\lambda}||_2^2 + C \sum_{i=1}^{n} \xi_i \quad \text{s.t.} \quad \begin{cases} \xi_i \geq 0 \\ y_i(\boldsymbol{\lambda}^T \mathbf{x}_i + \lambda_0) \geq 1 - \xi_i \end{cases}$$

$$\mathcal{L}(\boldsymbol{\lambda}, \lambda_0, \xi, \alpha, r) = \frac{1}{2} ||\boldsymbol{\lambda}||_2^2 + C \sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i \left[y_i(\boldsymbol{\lambda}^T \mathbf{x}_i + \lambda_0) - 1 + \xi_i \right] - \sum_{i=1}^{n} r_i \xi_i$$

where α_i's and r_i's are Lagrange multipliers (constrained to be ≥ 0). The dual turns out to be (after some work)

$$\max_{\alpha} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,k=1}^{n} \alpha_i \alpha_k y_i y_k \mathbf{x}_i^T \mathbf{x}_k \quad \text{s.t.} \quad \begin{cases} 0 \leq \alpha_i \leq C \quad i = 1 \dots n \\ \sum_{i=1}^{n} \alpha_i y_i = 0 \end{cases} \tag{6}$$

$$b^* = -\frac{\max_{i:y_i=-1} w^{*T} x_i + \min_{i:y_i=1} w^{*T} x_i}{2}$$

多项式核 $K(x_1, x_2) = (\langle x_1, x_2 \rangle + R)^d$，

高斯核 $K(x_1, x_2) = exp(-\|x_1 - x_2\|^2 / 2\sigma^2)$，

$$\begin{aligned} \alpha_i = 0 &\Leftrightarrow y_i u_i \geq 1, \\ 0 < \alpha_i < C &\Leftrightarrow y_i u_i = 1, \\ \alpha_i = C &\Leftrightarrow y_i u_i \leq 1. \end{aligned}$$

$$\begin{cases} L = \max(0, \alpha_2^{old} - \alpha_1^{old}), H = \min(C, C + \alpha_2^{old} - \alpha_1^{old}) & \text{if } y_1 \neq y_2 \\ L = \max(0, \alpha_2^{old} + \alpha_1^{old} - C), H = \min(C, \alpha_2^{old} + \alpha_1^{old}) & \text{if } y_1 = y_2 \end{cases}$$

令 $E_i = u_i - y_i$（表示预测值与真实值之差），$\eta = K(\vec{x}_1, \vec{x}_1) + K(\vec{x}_2, \vec{x}_2) - 2K(\vec{x}_1, \vec{x}_2)$，然后上式两边同时除以 η，得到一个关于单变量 α_2 的解：

$$\alpha_2^{new,unc} = \alpha_2^{old} + \frac{y_2(E_1 - E_2)}{\eta} \tag{3.44}$$

这个解没有考虑其约束条件 $0 \leq \alpha_2 \leq C$，即是未经剪辑时的解。

然后考虑约束 $0 \leq \alpha_2 \leq C$ 可得到经过剪辑后的 α_2^{new} 的解析解为：

$$\alpha_2^{new,unc} = \begin{cases} H, & \alpha_2^{new,unc} > H \\ \alpha_2^{new,unc} & L \leq \alpha_2^{new,unc} \leq H \\ L & \alpha_2^{new,unc} < L \end{cases} \tag{3.45}$$

求出了后 α_2^{new}，便可以求出 α_1^{new}，得 $\alpha_1^{new} = \alpha_1^{old} + y_1 y_2(\alpha_2^{old} - \alpha_2^{new})$。

- 对于 α_1，即第一个乘子，可以通过刚刚说的那 3 种不满足 KKT 的条件来找；
- 而对于第二个乘子 α_2 可以寻找满足条件：$\max|E_i - E_j|$ 的乘子。

而 b 在满足下述条件：

$$b = \begin{cases} b_1, & 0 < \alpha_1^{new} < C \\ b_2 & 0 < \alpha_2^{new} < C \\ (b_1 + b_2)/2 & \text{otherwise} \end{cases} \tag{3.46}$$

下更新 b：

$$b_1^{new} = b^{old} - E_1 - y_1(\alpha_1^{new} - \alpha_1^{old})K(x_1, x_1) - y_2(\alpha_2^{new} - \alpha_2^{old})K(x_1, x_2) \tag{3.47}$$

$$b_2^{new} = b^{old} - E_2 - y_1(\alpha_1^{new} - \alpha_1^{old})K(x_1, x_2) - y_2(\alpha_2^{new} - \alpha_2^{old})K(x_2, x_2) \tag{3.48}$$

且每次更新完两个乘子的优化后，都需要再重新计算 b，及对应的 E_i 值。

$$\begin{aligned} j_t &\in \operatorname{argmax}_j \left[-\frac{\partial R^{\text{train}}(\boldsymbol{\lambda}_t + \alpha \mathbf{e}_j)}{\partial \alpha} \bigg|_{\alpha=0} \right] \\ &= \operatorname{argmax}_j \left[-\frac{\partial}{\partial \alpha} \left[\frac{1}{n} \sum_{i=1}^{n} e^{-(\mathbf{M}(\boldsymbol{\lambda}_t + \alpha \mathbf{e}_j))_i} \right] \bigg|_{\alpha=0} \right] \\ &= \operatorname{argmax}_j \left[-\frac{\partial}{\partial \alpha} \left[\frac{1}{n} \sum_{i=1}^{n} e^{-(\mathbf{M}\boldsymbol{\lambda}_t)_i - \alpha(\mathbf{M}\mathbf{e}_j)_i} \right] \bigg|_{\alpha=0} \right] \\ &= \operatorname{argmax}_j \left[-\frac{\partial}{\partial \alpha} \left[\frac{1}{n} \sum_{i=1}^{n} e^{-(\mathbf{M}\boldsymbol{\lambda}_t)_i - \alpha M_{ij}} \right] \bigg|_{\alpha=0} \right] \\ &= \operatorname{argmax}_j \left[\frac{1}{n} \sum_{i=1}^{n} M_{ij} e^{-(\mathbf{M}\boldsymbol{\lambda}_t)_i} \right]. \end{aligned}$$

$$d_{t,i} = e^{-(\mathbf{M}\boldsymbol{\lambda}_t)_i} / Z_t \text{ where } Z_t = \sum_{i=1}^{n} e^{-(\mathbf{M}\boldsymbol{\lambda}_t)_i}$$

$$\begin{aligned} 0 &= \frac{\partial R^{\text{train}}(\boldsymbol{\lambda}_t + \alpha \mathbf{e}_{j_t})}{\partial \alpha} \bigg|_{\alpha_t} \\ &= -\frac{1}{n} \sum_{i=1}^{n} M_{ij_t} e^{-(\mathbf{M}\boldsymbol{\lambda}_t)_i - \alpha_t M_{ij_t}} \\ &= -\frac{1}{n} \sum_{i:M_{ij_t}=1} e^{-(\mathbf{M}\boldsymbol{\lambda}_t)_i} e^{-\alpha_t} - \frac{1}{n} \sum_{i:M_{ij_t}=-1} -e^{-(\mathbf{M}\boldsymbol{\lambda}_t)_i} e^{\alpha_t}. \end{aligned}$$

- *(Primal feasibility)* $g_i(\mathbf{x}^*) \leq 0, i = 1, ..., m$ and $h_i(\mathbf{x}^*) = 0, i = 1, ..., p$.
- *(Dual feasibility)* $\alpha_i^* \geq 0, i = 1, \dots, m$.
- *(Complementary Slackness)* $\alpha_i^* g_i(\mathbf{x}^*) = 0, i = 1, \dots, m$.
- *(Lagrangian stationary)* $\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) = \mathbf{0}$.

$$P(Y = y_i | \boldsymbol{\lambda}, \mathbf{x}_i) = \frac{1}{1 + e^{-y_i \boldsymbol{\lambda}^T \mathbf{x}_i}}.$$

$$\begin{aligned} \boldsymbol{\lambda}^* &\in \operatorname*{argmax}_{\boldsymbol{\lambda}} \log L(\boldsymbol{\lambda}) \\ &= \operatorname*{argmax}_{\boldsymbol{\lambda}} \sum_{i=1}^{n} \log \frac{1}{1 + e^{-y_i \boldsymbol{\lambda}^T \mathbf{x}_i}} \\ &= \operatorname*{argmin}_{\boldsymbol{\lambda}} \sum_{i=1}^{n} \log(1 + e^{-y_i \boldsymbol{\lambda}^T \mathbf{x}_i}). \end{aligned}$$

$$\begin{aligned} d_{1,i} &= \frac{1}{n} \text{ for all } i \\ d_{t+1,i} &= \frac{d_{t,i}}{Z_t} \times \begin{cases} e^{-\alpha_t} & \text{if } y_i = h_{(t)}(x_i) \ \text{(smaller weights for easy examples)} \\ e^{\alpha_t} & \text{if } y_i \neq h_{(t)}(x_i) \ \text{(larger weights for hard examples)} \end{cases} \end{aligned}$$

where Z_t is a normalization constant for the discrete distribution that ensures $\sum_i d_{t+1,i} = 1$

$$= \frac{d_{t,i}}{Z_t} e^{-y_i \alpha_t h_{(t)}(x_i)}$$

$$H(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_{(t)}(x) \right) \qquad \alpha_t = \frac{1}{2} \ln\left(\frac{1 - \epsilon_t}{\epsilon_t} \right).$$

$$\epsilon_t = P_{i \sim \mathbf{d}_t}[h_{(t)}(x_i) \neq y_i] = \sum_i d_{t,i} \mathbf{1}_{[h_{(t)}(x_i) \neq y_i]}$$

Consider the misclassification error:

$$\text{Miscl. error} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{[y_i f(x_i) \leq 0]}$$

which is upper bounded by the exponential loss:

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

Choose f to be some linear combination of weak classifiers,

$$f(x) = \sum_{j=1}^{p} \lambda_j h_j(x).$$

Define an $m \times n$ matrix $\mathbf{M}$ so that $M_{ij} = y_i h_j(x_i)$.

$$\mathbf{M} = \begin{array}{cc} & \begin{array}{c}\text{weak classifiers} \\ j\end{array} \\ \begin{array}{c}\text{examples} \\ i\end{array} & \begin{bmatrix} & & \\ & \pm 1 & \\ & & \end{bmatrix} \end{array}$$

So matrix $\mathbf{M}$ encodes all of the training examples and the whole weak learning algorithm. In other words, $\mathbf{M}$ contains all the inputs to AdaBoost. The ij^{th} entry in the matrix is 1 whenever weak classifier j correctly classifies example i. (Note: we might never write out the whole matrix $\mathbf{M}$ in practice!)

Then

$$y_i f(x_i) = \sum_j \lambda_j y_i h_j(x_i) = \sum_j \lambda_j M_{ij} = (\mathbf{M}\boldsymbol{\lambda})_i.$$

Then the exponential loss is:

$$R^{\text{train}}(\boldsymbol{\lambda}) = \frac{1}{n} \sum_i e^{-y_i f(x_i)} = \frac{1}{n} \sum_i e^{-(\mathbf{M}\boldsymbol{\lambda})_i}. \tag{3}$$

$d_{1,i} = 1/n$ `for` $i = 1 \dots n$
$\boldsymbol{\lambda}_1 = \mathbf{0}$
`loop` $t = 1 \dots T$
  $j_t \in \operatorname{argmax}_j (\mathbf{d}_t^T \mathbf{M})_j$
  $d_- = \sum_{M_{ij_t}=-1} d_{t,i}$
  $\alpha_t = \frac{1}{2} \ln\left(\frac{1-d_-}{d_-} \right)$
  $\boldsymbol{\lambda}_{t+1} = \boldsymbol{\lambda}_t + \alpha_t \mathbf{e}_{j_t}$
  $d_{t+1,i} = e^{-(\mathbf{M}\boldsymbol{\lambda}_{t+1})_i} / Z_{t+1}$ `for each` i`, where` $Z_{t+1} = \sum_{i=1}^{n} e^{-(\mathbf{M}\boldsymbol{\lambda}_{t+1})_i}$
`end`

$$\begin{aligned}
\mathbf{E}e^{-Yf(x)} &= P(Y=1|x)e^{-f(x)} + P(Y=-1|x)e^{f(x)} \\
0 = \frac{d\mathbf{E}(e^{-Yf(x)}|x)}{df(x)} &= -P(Y=1|x)e^{-f(x)} + P(Y=-1|x)e^{f(x)} \\
P(Y=1|x)e^{-f(x)} &= P(Y=-1|x)e^{f(x)} \\
\frac{P(Y=1|x)}{P(Y=-1|x)} &= e^{2f(x)} \Rightarrow f(x) = \frac{1}{2}\ln\frac{P(Y=1|x)}{P(Y=-1|x)}.
\end{aligned}$$

$$\frac{1}{n}\sum_{i=1}^{n}\mathbf{1}_{[y_i \neq H(x_i)]} \leq R^{\text{train}}(\boldsymbol{\lambda}_T) \leq e^{-2\sum_{t=1}^{T}\gamma_t^2} \leq e^{-2\gamma_{WLA}^2 T}.$$