A. Optimal Values of a, b, α in AUC Square Loss

In Section 3.2, we use the optimal values of a, b, α. In this section, we show how to derive these values. We first re-present the min-max problem in (3) as follows

$$\min_{\substack{\mathbf{w}\in\mathbb{R}^d \\ (a,b)\in\mathbb{R}^2}} \max_{\alpha\in\mathbb{R}} f(\mathbf{w}, a, b, \alpha) := \mathbb{E}_{\mathbf{z}}\left[F(\mathbf{w}, a, b, \alpha; \mathbf{z})\right],$$

where

$$\begin{aligned} F(\mathbf{w}, a, b, \alpha; \mathbf{z}) &= (1-p)\left(h_{\mathbf{w}}(\mathbf{x}) - a\right)^2 \mathbb{I}_{[y=1]} \\ &+ p(h_{\mathbf{w}}(\mathbf{x}) - b)^2 \mathbb{I}_{[y=-1]} - p(1-p)\alpha^2 \\ &+ 2\alpha\left(p(1-p) + p h_{\mathbf{w}}(\mathbf{x})\mathbb{I}_{[y=-1]} - (1-p)h_{\mathbf{w}}(\mathbf{x})\mathbb{I}_{[y=1]}\right). \end{aligned}$$

Given a fixed $\mathbf{w}$,the variable a is only involved in the first term in F, so we have the a-subproblem as

$$\begin{aligned} &\min_a \mathrm{E}_{\mathbf{z}}[(1-p)(h_{\mathbf{w}}(\mathbf{x}) - a)^2 \mathbb{I}_{[y=1]}] \\ =&(1-p)\mathrm{E}_{\mathbf{z}}[(h_{\mathbf{w}}(\mathbf{x}) - a)^2] \cdot \mathrm{E}_z[\mathbb{I}_{[y=1]}] \\ =&(1-p)\mathrm{E}_{\mathbf{z}}[(h_{\mathbf{w}}(\mathbf{x}) - a)^2 | y=1] \cdot p. \end{aligned}$$

As can be seen, $\mathrm{E}_{\mathbf{z}}[(h_{\mathbf{w}}(\mathbf{x}) - a)^2 | y=1]$ achieves minimum value when $a = \mathrm{E}[h_{\mathbf{w}}(\mathbf{x})|y=1]$, which becomes the variance of $h_{\mathbf{w}}(\mathbf{x})$.

The optimal value of $b = \mathrm{E}[h_{\mathbf{w}}(\mathbf{x})|y=-1]$ can be achieved in the same way as a.

The subproblem of α is

$$\begin{aligned} &\max_\alpha \mathrm{E}_{\mathbf{z}}[2\alpha(p(1-p) + p h_{\mathbf{w}}(\mathbf{x})\mathbb{I}_{[y=-1]} - (1-p)h_{\mathbf{w}}(\mathbf{x})\mathbb{I}_{[y=1]})] - p(1-p)\alpha^2 \\ =&2\alpha(p(1-p) + p\mathrm{E}_z[h_{\mathbf{w}}(\mathbf{x})\mathbb{I}_{[y=-1]}] - (1-p)\mathrm{E}_{\mathbf{z}}[h_{\mathbf{w}}(\mathbf{x})]\mathbb{I}_{[y=1]}) - p(1-p)\alpha^2 \\ =&2\alpha(p(1-p) + p(1-p)\mathrm{E}_z[h_{\mathbf{w}}(\mathbf{x})|y=-1] - p(1-p)\mathrm{E}_{\mathbf{z}}[h_{\mathbf{w}}(\mathbf{x})|y=-1]) - p(1-p)\alpha^2 \\ =&p(1-p)\cdot(1 + 2\alpha(\mathrm{E}_z[h_{\mathbf{w}}(\mathbf{x})|y=-1] - \mathrm{E}_{\mathbf{z}}[h_{\mathbf{w}}(\mathbf{x})|y=-1]) - \alpha^2) \end{aligned}$$

where we can derive its optimal value simply setting its gradient as zero. This leads to

$$\begin{aligned} \alpha^* =&1 + \mathrm{E}_z[h_{\mathbf{w}}(\mathbf{x})|y=-1] - \mathrm{E}_{\mathbf{z}}[h_{\mathbf{w}}(\mathbf{x})|y=-1] \\ =&1 + b(\mathbf{w}) - a(\mathbf{w}). \end{aligned}$$

B. Reformulation of AUC Square Loss

In this section, we reformulate AUC square loss as follows

$$\begin{aligned} A_{\mathrm{S}}(\mathbf{w}) =&\mathrm{E}[(1 - h_{\mathbf{w}}(\mathbf{x}) + h_{\mathbf{w}}(\mathbf{x}'))^2 | y=1, y'=-1] \\ =&\mathrm{E}[(1 - a(\mathbf{w}) + a(\mathbf{w}) - h(\mathbf{w};\mathbf{x}) + h(\mathbf{w};\mathbf{x}') - b(\mathbf{w}) + b(\mathbf{w}))^2 | y=1, y'=-1] \\ =&\mathrm{E}[\big((a(\mathbf{w}) - h(\mathbf{w};\mathbf{x}) + h(\mathbf{w};\mathbf{x}') - b(\mathbf{w})) + (1 + b(\mathbf{w}) - a(\mathbf{w}))^2\big) | x=1, y'=-1] \\ =&\mathrm{E}[(a(\mathbf{w}) - h(\mathbf{w};\mathbf{x}) + h(\mathbf{w};\mathbf{x}') - b(\mathbf{w}))^2 + (1 + b(\mathbf{w}) - a(\mathbf{w}))^2 \\ &+ 2\big(a(\mathbf{w}) - h(\mathbf{w};\mathbf{x}) + h(\mathbf{w};\mathbf{x}') - b(\mathbf{w})\big)\cdot(1 + b(\mathbf{w}) - a(\mathbf{w})) | y=1, y'=-1] \\ \overset{(e1)}{=}&\mathrm{E}[(h(\mathbf{w};\mathbf{x}) - a(\mathbf{w}))^2 + (h(\mathbf{w};\mathbf{x}') - b(\mathbf{w}))^2 - 2(h(\mathbf{w};x) - a(\mathbf{w}))\cdot(h(\mathbf{w};\mathbf{x}') - b(\mathbf{w})) \\ &+ (1 + b(\mathbf{w}) - a(\mathbf{w}))^2 | y=1, y'=-1] \\ \overset{(e2)}{=}&\mathrm{E}[(h(\mathbf{w};\mathbf{x}) - a(\mathbf{w}))^2 | y=1] + \mathrm{E}[(h(\mathbf{w};\mathbf{x}') - b(\mathbf{w}))^2 | y'=-1] \\ &+ (1 + b(\mathbf{w}) - a(\mathbf{w}))^2 \\ \overset{(e3)}{=}&\mathrm{E}[(h(\mathbf{w};\mathbf{x}) - a(\mathbf{w}))^2 | y=1] + \mathrm{E}[(h(\mathbf{w};\mathbf{x}') - b(\mathbf{w}))^2 | y'=-1] \\ &+ \max_\alpha 2\alpha(1 + b(\mathbf{w}) - a(\mathbf{w})) - \alpha^2, \end{aligned}$$

where equality $(e1)$ is due to the definitions $a(\mathbf{w}) = \mathrm{E}[h(\mathbf{w};\mathbf{x})|y=1]$ and $b(\mathbf{w}) = \mathrm{E}[h(\mathbf{w};\mathbf{x}')|y'=-1]$, $\mathrm{E}[a(\mathbf{w})] = a(\mathbf{w})$ and $\mathrm{E}[b(\mathbf{w})] = b(\mathbf{w})$ ($a(\mathbf{w})$ and $b(\mathbf{w})$ are expectations, so they are constants). Equality $(e2)$ is due to the independence of the positive and negative samples. Equality $(e3)$ is due to the convex conjugate of the square function:

$$x^2 = \max_y 2y\cdot x - y^2.$$

C. Proof of Theorem 1

Below, we start from the min-max problem and prove it is equivalent to the AUC margin loss in (6).

$$
\begin{aligned}
&\min_{a,b}\max_{\alpha\geq 0}\mathrm{E}_{\mathbf{z}}[F_{\mathrm{M}}(\mathbf{w},a,b,\alpha;\mathbf{z})]\\
=&\min_{a,b}\max_{\alpha\geq 0}\mathrm{E}_{\mathbf{z}}\Big[(1-p)\left(h_{\mathbf{w}}(\mathbf{x})-a\right)^2\mathbb{I}_{[y=1]}+p(h_{\mathbf{w}}(\mathbf{x})-b)^2\mathbb{I}_{[y=-1]}-p(1-p)\alpha^2\\
&\qquad+2\alpha\left(p(1-p)m+ph_{\mathbf{w}}(\mathbf{x})\mathbb{I}_{[y=-1]}-(1-p)h_{\mathbf{w}}(\mathbf{x})\mathbb{I}_{[y=1]}\right)\Big]\\
=&\min_{a,b}\max_{\alpha\geq 0}\Big[(1-p)\mathrm{E}_{\mathbf{z}}[\left(h_{\mathbf{w}}(\mathbf{x})-a\right)^2\mathbb{I}_{[y=1]}]+p\mathrm{E}_{\mathbf{z}}[(h_{\mathbf{w}}(\mathbf{x})-b)^2\mathbb{I}_{[y=-1]}]-p(1-p)\alpha^2\\
&\qquad+2\alpha\left(p(1-p)m+p\mathrm{E}_{\mathbf{z}}[h_{\mathbf{w}}(\mathbf{x})\mathbb{I}_{[y=-1]}]-(1-p)\mathrm{E}_{\mathbf{z}}[h_{\mathbf{w}}(\mathbf{x})\mathbb{I}_{[y=1]}]\right)\Big]\\
=&\max_{\alpha\geq 0}\quad p(1-p)\Big[\mathrm{E}_{\mathbf{z}}[\left(h_{\mathbf{w}}(\mathbf{x})-a(\mathbf{w})\right)^2|y=1]+\mathrm{E}_{\mathbf{z}}[(h_{\mathbf{w}}(\mathbf{x})-b(\mathbf{w}))^2|y=-1]-\alpha^2\\
&\qquad+2\alpha\left(m+b(\mathbf{w})-a(\mathbf{w})\right)\Big]=p(1-p)A_{\mathrm{M}}(\mathbf{w}) \qquad (9)\\
=&\qquad p(1-p)\Big[\mathrm{E}_{\mathbf{z}}[\left(h_{\mathbf{w}}(\mathbf{x})-a(\mathbf{w})\right)^2|y=1]+\mathrm{E}_{\mathbf{z}}[(h_{\mathbf{w}}(\mathbf{x})-b(\mathbf{w}))^2|y=-1]+(m+b(\mathbf{w})-a(\mathbf{w}))_+^2\Big]
\end{aligned}
$$

where in (9), we show the equivalence between minimizing $A_{\mathrm{M}}(\mathbf{w})$ in (6) and $\min_{\mathbf{w},a,b}\max_{\alpha\geq 0}\mathrm{E}_{\mathbf{z}}[F_{\mathrm{M}}(\mathbf{w},a,b,\alpha;\mathbf{z})]$, i.e.,

$$\min_{a,b}\max_{\alpha\geq 0}\mathrm{E}_{\mathbf{z}}[F_{\mathrm{M}}(\mathbf{w},a,b,\alpha;\mathbf{z})]=p(1-p)A_{\mathrm{M}}(\mathbf{w}).$$

The last equality is to explicitly show the squared hinge loss.

D. Analysis of Adverse Effect on Easy Data of Square loss based on the min-max formulation

In particular, the gradient of $F(\mathbf{w},a,b,\alpha;\mathbf{z})$ is given by $\nabla_{\mathbf{w}}F(\mathbf{w},a,b,\alpha;\mathbf{z})=2(1-p)\mathbf{x}\mathbb{I}_{[y=1]}\cdot(h_{\mathbf{w}}(\mathbf{x})-a-\alpha)+2p\mathbf{x}\mathbb{I}_{[y=-1]}\cdot(h_{\mathbf{w}}(\mathbf{x})-b+\alpha)$. When $\mathbf{z}$ is positive, the first term above is active, by plugging the optimal value of a,b,α given $\mathbf{w}$, the stochastic gradient descent update will yields an updated model as

$$\mathbf{w}_+=\mathbf{w}-\eta 2(1-p)\mathbf{x}\mathbb{I}_{[y=1]}\cdot(h_{\mathbf{w}}(\mathbf{x})-1-b),$$

where b is the mean prediction score on negative data. When $\mathbf{x}$ is an easy positive data such that $h_{\mathbf{w}}(\mathbf{x})-1-b>0$, then $\mathbf{w}_+$ will move towards the negative direction of the positive data $\mathbf{x}$, as a result it will push the score $h_{\mathbf{w}_+}(\mathbf{x})$ on the positive data smaller than $h_{\mathbf{w}}(\mathbf{x})$, which is harmful for AUC maximization. Similarly, we have the same phenomenon when the sampled data $\mathbf{z}$ is negative.

E. A 1-Dim Example of Easy/Noisy Data for AUC Square and Margin Loss

Suppose we have a 1-dimensional AUC maximization problem with a linear model parameterized by a 1-dimensional model $\mathbf{w}$, i.e., $h_{\mathbf{w}}(\mathbf{x})=\mathbf{w}\cdot\mathbf{x}$, so that $\nabla_{\mathbf{w}}h_{\mathbf{w}}(\mathbf{x})=\mathbf{x}$. Recall the definition of F in (3), we have its gradient w.r.t. $\mathbf{w}$ as follows

$$
\begin{aligned}
\nabla_{\mathbf{w}}F(\mathbf{w},a,b,\alpha;\mathbf{z})=&2(1-p)(h_{\mathbf{w}}(\mathbf{x})-b)\cdot\nabla_{\mathbf{w}}h_{\mathbf{w}}(\mathbf{x})\mathbb{I}_{[y=1]}+2p(h_{\mathbf{w}}(\mathbf{x})-b)\cdot\nabla_{\mathbf{w}}h_{\mathbf{w}}(\mathbf{x})\mathbb{I}_{[y=-1]}\\
&+2\alpha(p\nabla_{\mathbf{w}}h_{\mathbf{w}}(\mathbf{x})\mathbb{I}_{[y=-1]}-(1-p)\nabla_{\mathbf{w}}h_{\mathbf{w}}(\mathbf{x})\mathbb{I}_{[y=1]})\\
=&2(1-p)\nabla_{\mathbf{w}}h_{\mathbf{w}}(\mathbf{x})\mathbb{I}_{[y=1]}\cdot\underbrace{(h_{\mathbf{w}}(\mathbf{x})-a-\alpha)}_{=B}+2p\nabla_{\mathbf{w}}h_{\mathbf{w}}(\mathbf{x})\mathbb{I}_{[y=-1]}\cdot\underbrace{(h_{\mathbf{w}}(\mathbf{x})-b+\alpha)}_{=C},
\end{aligned}
$$

where our study focuses on the two terms B and C, which determines the direction of $\nabla_{\mathbf{w}}F$ for $y=1$ and $y=-1$, respectively.

α is the key difference between AUC square loss in (3) and AUC margin loss in (6). To simplify the explanation, we let $a=a(\mathbf{w})$ and $b=b(\mathbf{w})$ achieve their optimal values. In AUC square loss (3), α is not constrained, and the optimal value is

$\alpha = 1 + b - a$. In AUC margin loss (6), it has a non-negative constraint on α, so the optimal value is $\alpha = \max\{0, 1 + b - a\}$.

E.1. Easy Data for AUC Square Loss

At the t-th iteration, let $\mathbf{w}_t = 1$ and we have two easy data $(\mathbf{x}_1 = 1, y_1 = 1)$ and $(\mathbf{x}_2 = -1, y_2 = -1)$. We assume that $a = 0.5$ and $b = -0.5$.

For $(\mathbf{x}_1, y_1 = 1)$

$$B = h_{\mathbf{w}}(\mathbf{x}_1) - a - \alpha = h_{\mathbf{w}}(\mathbf{x}_1) - 1 - b = 1 \times 1 - 1 - (-0.5) = 0.5,$$

which indicates that $\nabla_{\mathbf{w}} F \propto \nabla_{\mathbf{w}} h_{\mathbf{w}}(\mathbf{x}_1)$ (they are in the same direction). By assuming all the constants and the step size can be merged into a constant value 0.1, the stochastic gradient descent can be

$$\mathbf{w}_{t+1} = \mathbf{w}_t - 0.1 \times \nabla_{\mathbf{w}} h_{\mathbf{w}_t}(\mathbf{x}_1) = 1 - 0.1 \times \mathbf{x}_1 = 1 - 0.1 \times 1 = 0.9.$$

Then we re-evaluate the prediction score by $\mathbf{w}_{t+1}$:

$$h_{\mathbf{w}_{t+1}}(\mathbf{x}_1) = 0.9 \times 1 = 0.9 < h_{\mathbf{w}_t}(\mathbf{x}_1) = 1.$$

In this case, the prediction score for a positive sample decreases, which is an undesirable update.

For $(\mathbf{x}_2, y_2 = -1)$

$$C = h_{\mathbf{w}}(\mathbf{x}_1) - b + \alpha = h_{\mathbf{w}}(\mathbf{x}_1) + 1 - a = -1 + 1 - 0.5 = -0.5,$$

which indicates that $\nabla_{\mathbf{w}} F \propto -\nabla_{\mathbf{w}} h_{\mathbf{w}}(\mathbf{x}_1)$ (they are in the negative direction of each other). By assuming all the constants and the step size can be merged into a constant value 0.1, the stochastic gradient descent can be

$$\mathbf{w}_{t+1} = \mathbf{w}_t - 0.1 \times (-1) \times \nabla_{\mathbf{w}} h_{\mathbf{w}_t}(\mathbf{x}_1) = 1 + 0.1 \times \mathbf{x}_2 = 1 + 0.1 \times (-1) = 0.9.$$

Then we re-evaluate the prediction score by $\mathbf{w}_{t+1}$:

$$h_{\mathbf{w}_{t+1}}(\mathbf{x}_2) = 0.9 \times (-1) = -0.9 > h_{\mathbf{w}_t}(\mathbf{x}_2) = -1.$$

In this case, the prediction score for a negative sample increases, which is an undesirable update.

E.2. Easy Data for AUC Margin Loss

Since the optimal $\alpha = \max\{0, m + b - a\}$, we consider the two cases, respectively.

Case 1: $\alpha = 0$. This case indicates that $m + b - a \leq 0$ or $m + b \leq a$, which is a good situation, because a (the mean prediction of positive data) and b (the mean prediction of negative data) are sufficiently far away from each other by a margin of m. Here for simplicity, we assume that at the t-th iteration, $\mathbf{w} = 1$, $m = 1$, $a = 1$ and $b = -0.5$.

For $(\mathbf{x}_1 = 0.75, y = 1)$:

$$B = h_{\mathbf{w}}(\mathbf{x}_1) - a - \alpha = h_{\mathbf{w}}(\mathbf{x}_1) - a = 0.75 - 1 = -0.25 \qquad \text{(negative direction)},$$

where $h_{\mathbf{w}}(\mathbf{x}_1) > m + b = 0.5$ means that $\mathbf{x}_1$ is well classified, but F_{M} still suffers a penalty on it and push it to be closer to $a = 1$.

For $(\mathbf{x}_1 = 1.25, y = 1)$:

$$B = h_{\mathbf{w}}(\mathbf{x}_1) - a - \alpha = h_{\mathbf{w}}(\mathbf{x}_1) - a = 1.25 - 1 = 0.25 \qquad \text{(negative direction)},$$

where $h_{\mathbf{w}}(\mathbf{x}_1) > m + b = 0.5$ means that $\mathbf{x}_1$ is well classified, but F_{M} still suffers a penalty on it and push it to be closer to $a = 1$. To sum up, when the model is good enough, i.g., $m + b < a$, F_{M} only push positive data towards a and negative data towards b.

Case 2: $\alpha = m + b - a$. This case indicates that $m + b - a > 0$ or $m + b > a$, which is a undesirable situation, because a (the mean prediction of positive data) and b (the mean prediction of negative data) are within a margin of m. Here for simplicity, we assume that at the t-th iteration, $\mathbf{w} = 1, m = 1, a = 0, b = -0.5$.

For $(\mathbf{x}_1 = 0.25, y_1 = 1)$:

$$B = h_{\mathbf{w}}(\mathbf{x}_1) - a - \alpha = h_{\mathbf{w}}(\mathbf{x}_1) - m - b = 0.25 - 1 + 0.5 = -0.25 \qquad \text{(negative direction)},$$

where $h_{\mathbf{w}}(\mathbf{x}_1) < m + b = 0.5$ means that $\mathbf{x}_1$ is not well classified. Thus, the stochastic gradient descent for updating $\mathbf{w}_t$ can be

$$\mathbf{w}_{t+1} = \mathbf{w}_t - 0.1 \times (-1) \times \nabla_w h_{\mathbf{w}_t}(\mathbf{x}_1) = 1 + 0.1 \times \mathbf{x}_1 = 1 + 0.1 \times 0.25 = 1.025,$$

which makes the prediction of $\mathbf{x}_1$ larger: $h_{\mathbf{w}_{t+1}}(\mathbf{x}_1) = 1.025 \times 0.25 = 0.2562 > h_{\mathbf{w}_t}(\mathbf{x}_1) = 0.25$.

Examples for negative data can be derived in a similar way, so we omit those presentation.

E.3. Noisy Data for AUC Square Loss

Assuming $\mathbf{w} = 1$, consider the case where $m + b > a$, i.e., the model is not good, e.g., $a = 0.25, b = -0.5$. For $(\mathbf{x}_1 = 0.25, y_1 = -1, y_1^{\text{true}} = 1)$, since only y_1 is revealed, we will use term C to determine $\nabla_{\mathbf{w}} F$. On the other hand, since $y_1^{\text{true}} = 1$, we know that $h_{\mathbf{w}}(\mathbf{x})$ can be large. Then we can compute its term C

$$C = h_{\mathbf{w}}(\mathbf{x}_1) - b + \alpha = h_{\mathbf{w}}(\mathbf{x}_1) + 1 - a = 0.25 \times 1 + 1 - 0.25 = 1 \qquad \text{(positive direction)},$$

which means that $\nabla_{\mathbf{w}} F$ is in the same direction of $\nabla_w h_{\mathbf{w}}(\mathbf{x}_1)$. It is exactly the same case in Section E.1 when $B > 0$, so it will give an undesirable update.

Negative sample $(\mathbf{x}_2 = -1, y_1 = 1, y_1^{\text{true}} = -1)$ can be developed in the same way, which also gives an undesirable update.

E.4. Noisy Data for AUC Margin Loss

Assuming $\mathbf{w} = 1$, consider the case where $m + b > a$, i.e., the model is not good, and $\alpha = m + b - a$. We assume $a = 0.25, b = -0.5$. For $(\mathbf{x}_1 = 0.25, y_1 = -1, y_1^{\text{true}} = 1)$:

$$C = h_{\mathbf{w}}(\mathbf{x}_1) - b + \alpha = h_{\mathbf{w}}(\mathbf{x}_1) - b + (m + b - a) = h_{\mathbf{w}}(\mathbf{x}_1) + m - a = 0.25 \times 1 + m - 0.25 = m.$$

m is positive by dentition. However, unlike the previous AUC square loss where $m = 1$, in AUC margin loss m is a hyper-parameter. Even though we cannot completely resolve the noisy data issue by using AUC margin loss, we can still reduce the magnitude of update along with the wrong direction by changing m to a smaller value from constant 1.

The same situation happens for noisy negative data on the not-so-good model.

F. An Example of Sensitivity of AUC

Table 6. Illustrations of sensitivity of Accuracy and AUC on an imbalanced dataset of 25 samples with a positive ratio of 3/25. The accuracy threshold is 0.5. **Example 1** shows that all positive instances rank higher than negative instances and two negative instances are misclassified to positive class. **Example 2** shows that 1 positive instance ranks lower than 7 negative instances and 1 positive and 1 negative instances are missclassifed. **Example 3** shows that 2 positive instances rank lower than 7 negative instances, and 2 positive instances are also missclassifed as negative class. Overall, we can observe that AUC drops dramatically as the ranks of positive instances drop but meanwhile Accuracy remains unchanged.

Example 1		**Example 2**		**Example 3**	
Prediction	Ground Truth	Prediction	Ground Truth	Prediction	Ground Truth
0.9	1	0.9	1	0.9	1
0.8	1	**0.41**(↓)	1	**0.41**(↓)	1
0.7	1	0.7	1	**0.40**(↓)	1
0.6	0	0.6	0	**0.49**(↓)	0
0.6	0	**0.49**(↓)	0	**0.48**(↓)	0
0.47	0	0.47	0	0.47	0
0.47	0	0.47	0	0.47	0
0.45	0	0.45	0	0.45	0
0.43	0	0.43	0	0.43	0
0.42	0	0.42	0	0.42	0
⋮	⋮	⋮	⋮	⋮	⋮
0.1	0	0.1	0	0.1	0
Acc=0.92		Acc=0.92 (—)		Acc=0.92 (—)	
AUC=1.00		AUC=**0.89** (↓)		AUC=**0.78** (↓)	

G. Descriptions of Imbalanced Datasets

Table 7. Description of of Datasets. Note that "size of training set" refers to the number of samples for the original training set. Datasets with suffix "-IB" denote that we manually construct the imbalanced datasets by randomly removing some positive samples.

Datasets	**Size of image**	**Size of training set**
Cat&Dog-IB	low resolution	25,000
CIFAR10-IB	low resolution	50,000
CIFAR100-IB	low resolution	50,000
STL10-IB	medium resolution	5,000
PatchCamelyon-IB	medium resolution	294,912
Melanoma	high resolution	46,131
CheXpert	high resolution	223,416
DDSM+	high resolution	55,890

H. More Experiments on Benchmark Datasets

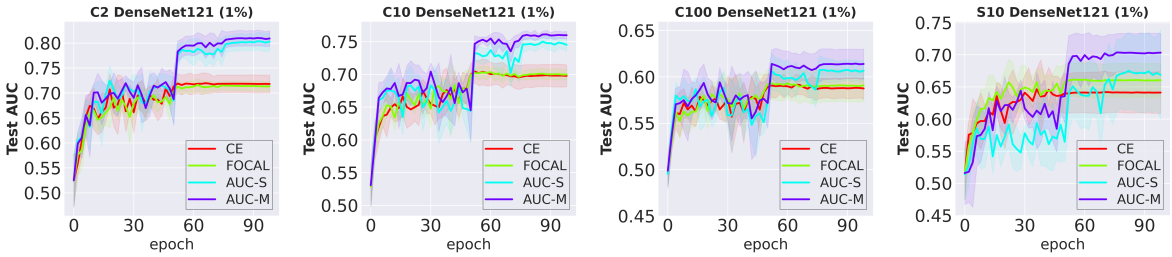

Figure 2. Testing AUC vs epochs on Benchmark Datasets for DenseNet121.

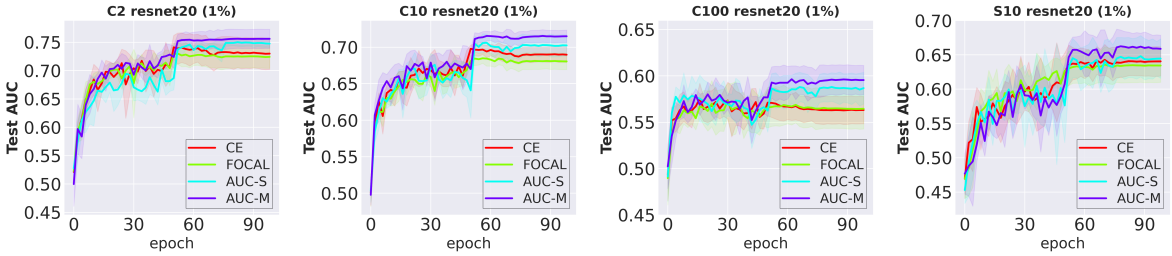

Figure 3. Testing AUC vs epochs on Benchmark Datasets for ResNet20.

Table 8. Testing AUC of benchmark datasets with DenseNet121(D) and ResNet20(R) for imratio=10%. Note that when the imbalance ratio increases e.g., from 1% to 10%, data becomes less imbalanced and the classification becomes easier.

Dataset	imratio	CE	Focal	AUC-S	AUC-M	AUC-M>AUC-S
C2 (D)	10%	0.893±0.004	0.879±0.005	0.901±0.002	**0.902±0.001**	✓
C10 (D)	10%	**0.898±0.005**	0.879±0.005	0.889±0.002	0.887±0.005	✗
S10 (D)	10%	0.820±0.015	0.819±0.010	0.825±0.013	**0.846±0.015**	✓
C100 (D)	10%	0.710±0.007	0.705±0.007	0.720±0.003	**0.723±0.006**	✓
C2 (R)	10%	0.920±0.004	0.881±0.008	0.897±0.007	**0.920±0.006**	✓
C10 (R)	10%	**0.898**±0.004	0.851±0.018	0.872±0.007	0.898±0.005	✓
S10 (R)	10%	**0.825**±0.013	0.813±0.009	0.819±0.013	0.821±0.011	✓
C100(R)	10%	0.669±0.006	0.666±0.012	0.686±0.005	**0.695±0.003**	✓

I. The Choice of Margin m for AUC-M Loss

Margin m is an important parameter for AUC-M loss. As illustrated in Section 3.3, when the model is not good enough, noisy data may produce a stochastic gradient that indicates a wrong direction. In this case, a smaller m can alleviate such sensitivity to noisy data. Tuning m parameter can trade off the margin benefit and the robustness to noisy data. That is the reason why tuning m is important in AUC-M. On benchmark datasets, the average values of m over different random trials are 0.7,0.8,0.7,0.5 on C2, C10, S10, C100, respectively. On Melanoma, the best m is 0.8. On CheXpert, the best m is 0.8 in average over 5 classes. On DDSM, the best m is 0.5. On PatchCamelyon, the best m is 0.7. For the results of ablation studies, we use $m = 0.3$ for AUC-M loss.

J. Illustrations of Prediction Distribution for Cross-Entropy and AUC-M Losses

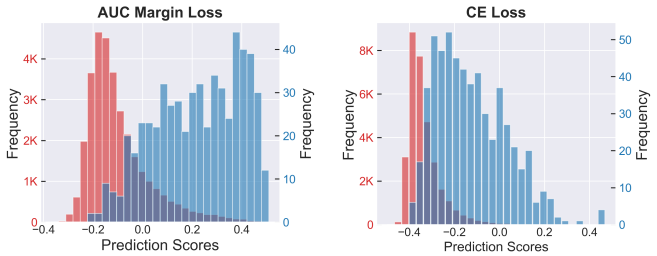

Figure 4. Prediction histogram of positive (blue) and negative (red) samples for the models trained by AUC-M loss and CE loss on Melanoma training dataset. We can see that the predictions made by the DAM method have two well-separated patterns corresponding to positive and negative data. In contrast, the predictions made by optimizing the CE loss is more mixed together.

K. A Two-stage Training Framework for DAM

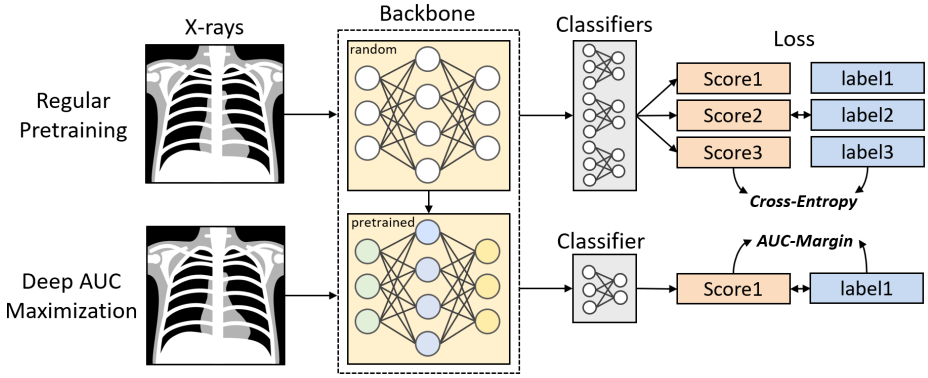

Figure 5. A Two-stage Deep AUC Maximization Framework. For the pretraining stage, we focus on learning representation by optimizing a standard CrossEntropy loss. For the AUC maximization stage, we focus on finetuning the decision boundary of classifier by optimizing AUC margin loss.

L. Network Architecture for Melanoma Classification

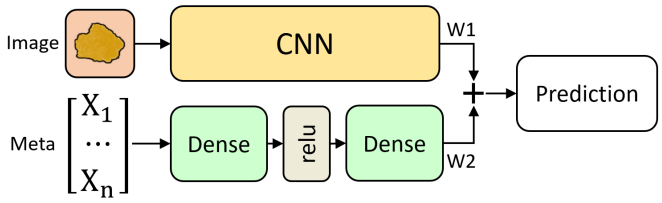

Figure 6. A mixed network architecture of a CNN (EfficientNet) and a 2-layer Neural Network for predicting Melanoma using image and patient contextual data. For training, we first train the CNN model and then train DNN model (using same configurations) but freeze the parameter updates for CNN model. The training configurations are described in main section.

M. Ablation Study on Batch Score Normalization (BSN)

We run experiments with DenseNet121 on four benchmark datasets with two imbalance ratio, e.g., 1%, 10% with and without applying batch score normalization. The results are shown in Figure 7. We can see that applying the BSN can improve the performance.

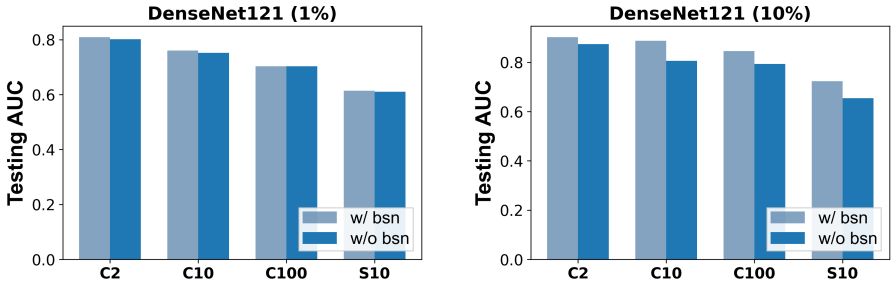

Figure 7. Ablation Study on Batch Score Normalization.

N. Ablation Studies on AUC-M Loss

Robustness to Noisy Data and Easy Data. We conduct ablation studies on the C2-IB data. To verify the robustness of our AUC-M loss to noisy data, we manually create some data with noisy labels. We construct the noisy dataset by modifying the C2 (imratio=1%). To this end, we sample 1% and 5% from negative class to flip their labels to positive, and also randomly sample 1% and 5% positive data from the deleted positive examples and flip their labels and add them to the training data. This gives us two datasets with 1% and 5% noisy ratio. To verify the robustness of our AUC loss to easy data, we first pre-train a model by minimizing CE loss on C2 (imratio=1%) and then we make predictions on the removed positive samples and sort all prediction scores in descending order. Finally, we choose top 10%, 20% of sorted samples and add them to training data. We train DenseNet121 using batch size of 128 and initial learning rate of 0.1. Other parameter settings are the same as in Section 4.1. We run experiments 5 times and plot the average testing AUC curve in Figure 8 for the setting with 1% noisy data and 10% easy data. In Figure 9, we report results on other settings. All results clearly show that AUC-M outperforms AUC-S by a large margin.

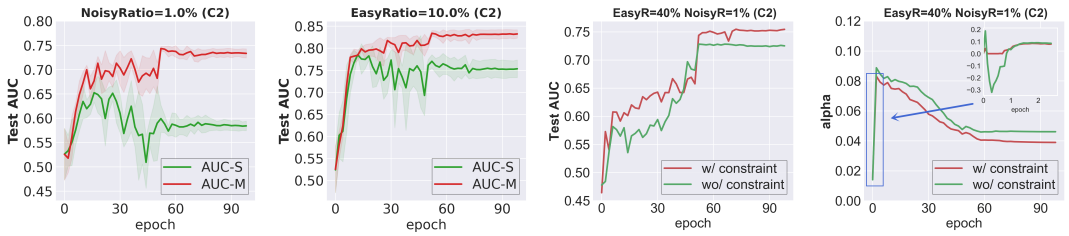

Figure 8. First two plots: comparison when adding noisy and easy samples. Last two plots: comparison between with/without $\alpha \geq 0$.

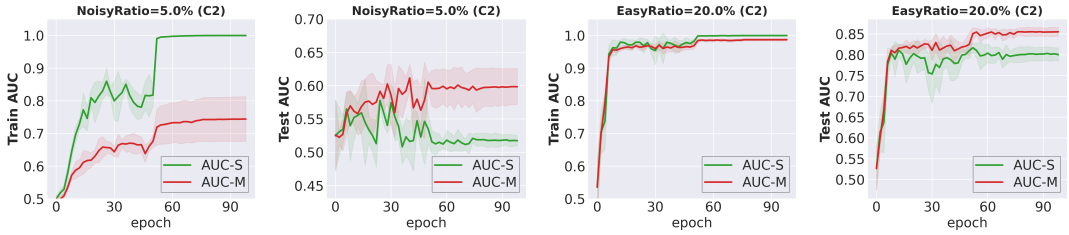

Figure 9. Comparison when adding extra 20% easy samples and 5% noisy samples.

Effect of Alpha Constraint. To verify the effectiveness of non-negative constraint on α, we design an experiment to compare the performance of AUC-M with and without $\alpha \geq 0$ constraint. We start with C2-IB with imbalance ratio of 1%

and add 40% easy (positive) samples and 1% noisy samples to the training set similar to that is done above. We fix margin $m = 0.1$. The curve of testing AUC and the curve of α v.s. # of epochs are plotted in Figure 8 (bottom panel). We observe that the performance with enforcing $\alpha \geq 0$ is better than that without enforcing it. The bottom right plot in Figure 8 gives us a better illustration about the change of α during training. The plot inside it reveals the change of α in the first 2 epochs. It shows that the constraint prevents the value of α from dropping to a bad region and hence yields a faster convergence and better result.