CS480/680, Spring 2025

Review Notes

Student: Hongxu Xu (h445xu@uwaterloo.ca)

August 5, 2025

Part I
For Mid-Term

1 Perceptron

Question 1 (Linear Function)**.**

$$\begin{aligned}&\forall\alpha\beta\in\mathbb{R},\forall\boldsymbol{x},\boldsymbol{z}\in\mathbb{R}^d, f(\alpha\boldsymbol{x}+\beta\boldsymbol{z})=\alpha\cdot f(\boldsymbol{X})+\beta\cdot f(\boldsymbol{z})\\ \iff&\exists\boldsymbol{w}\in\mathbb{R}^d, f(\boldsymbol{x})=\langle\boldsymbol{x},\boldsymbol{w}\rangle\end{aligned}\tag{1}$$

Proof $\Rightarrow$: Let $\boldsymbol{w}\coloneqq[f(\boldsymbol{e_1}),\dots,f(\boldsymbol{e_d})]^T$, where $\boldsymbol{e_i}$ is the i-th coordinate vector.

$$\begin{aligned}f(\boldsymbol{x})&=f(x_1\boldsymbol{e_1}+\cdots+x_d\boldsymbol{e_d})\\&=x_1f(\boldsymbol{e_1})+\cdots+x_df(\boldsymbol{e_d})\\&=\langle\boldsymbol{x},\boldsymbol{w}\rangle\end{aligned}\tag{2}$$

Proof $\Leftarrow$:

$$\begin{aligned}f(\alpha\boldsymbol{x}+\beta\boldsymbol{z})&=\langle\alpha\boldsymbol{x}+\beta\boldsymbol{z},\boldsymbol{w}\rangle\\&=\langle\alpha\boldsymbol{x},\boldsymbol{w}\rangle+\langle\beta\boldsymbol{z},\boldsymbol{w}\rangle\\&=\alpha\langle\boldsymbol{x},\boldsymbol{w}\rangle+\beta\langle\boldsymbol{z},\boldsymbol{w}\rangle\\&=\alpha f(\boldsymbol{x})+\beta f(\boldsymbol{z})\end{aligned}\tag{3}$$

Question 2 ($\boldsymbol{w}$ is Orthogonal to Decision Boundary H)**.** Any vector on H can be written as $\overrightarrow{\boldsymbol{xx'}}=\boldsymbol{x'}-\boldsymbol{x}$,

$$\langle\boldsymbol{x'}-\boldsymbol{x},\boldsymbol{w}\rangle=\langle\boldsymbol{x'},\boldsymbol{w}\rangle-\langle\boldsymbol{x},\boldsymbol{w}\rangle=-b-(-b)=0\tag{4}$$

Question 3 (Update Rule for Perceptron)**.**

$$\begin{aligned}\boldsymbol{w}&\leftarrow\boldsymbol{w}+y_i\boldsymbol{x_i}\\b&\leftarrow=b+y_i\end{aligned}\tag{5}$$

Question 4 (Feasibility of Perceptron)**.** The goal is to find $\boldsymbol{w}\in\mathbb{R}^d, b\in\mathbb{R}$, such that $\forall i, y_i(\langle\boldsymbol{x_i},\boldsymbol{w}\rangle+b)>0$.
According to the update rule 5,

$$\begin{aligned}y\left[\langle\boldsymbol{x},\boldsymbol{w_{new}}\rangle+b_{new}\right]&=y\left[\langle\boldsymbol{x},\boldsymbol{w_{old}+yx}\rangle+b_{old}+y\right]\\&=y\left[\langle\boldsymbol{x},\boldsymbol{w_{old}}\rangle+b_{old}\right]+y\left[\langle\boldsymbol{x},y\boldsymbol{x}\rangle+y\right]\\&=y\left[\langle\boldsymbol{x},\boldsymbol{w_{old}}\rangle+b_{old}\right]+y\left[y\left\|\boldsymbol{x}\right\|_2^2+y\right]\\&=y\left[\langle\boldsymbol{x},\boldsymbol{w_{old}}\rangle+b_{old}\right]+y^2\left\|\boldsymbol{x}\right\|_2^2+y^2\\&=y\left[\langle\boldsymbol{x},\boldsymbol{w_{old}}\rangle+b_{old}\right]+\underbrace{\left\|\boldsymbol{x}\right\|_2^2+1}_{\text{always positive}}\end{aligned}\tag{6}$$

Notice that $y\in\{\pm1\}\Rightarrow y^2=1$.
$\left\|\boldsymbol{x}\right\|_2^2+1$ is always positive, which means we always increase the confidence $y\hat{y}$ after the update.

Question 5 (Trick for Hiding the Bias Term – Padding)**.**

$$\begin{aligned}\langle\boldsymbol{x},\boldsymbol{w}\rangle+b&=\left\langle\begin{pmatrix}\boldsymbol{x}\\1\end{pmatrix},\begin{pmatrix}\boldsymbol{w}\\b\end{pmatrix}\right\rangle\\&=\langle\boldsymbol{x_{pad}},\boldsymbol{w_{pad}}\rangle\end{aligned}\tag{7}$$

Correspondingly, the update rule can be written as:

$$\boldsymbol{w_{pad}}\leftarrow\boldsymbol{w_{pad}}+y\boldsymbol{x_{pad}}\tag{8}$$

Question 6 (Margin)**.** Suppose $\exists\boldsymbol{w^*}$ such that $\forall i, y_i\langle\boldsymbol{x_i},\boldsymbol{w^*}\rangle>0$.
We normalize $\boldsymbol{w^*}$ such that $\left\|\boldsymbol{w^*}\right\|_2=1$.
In other words, $\boldsymbol{w^*}$ is the normalized weight for the deicision boundary.

$$\text{Margin }\gamma\coloneqq\min_i\left|\langle\boldsymbol{x_i},\boldsymbol{w^*}\rangle\right|\tag{9}$$

Question 7 (Convergence Theorem – Linearly Separable Case)**.** Assume that $\forall i, \|\boldsymbol{x_i}\|_2 \leq C$ (i.e. within a circle with radius C). Then the Perceptron algorithm converges after $\frac{C^2}{\gamma^2}$ mistakes.

Proof:
Suppose $\boldsymbol{w}$ is the updating weight, and θ is the angle between $\boldsymbol{w}$ and $\boldsymbol{w}^*$.
We have $\langle \boldsymbol{w}, \boldsymbol{w}^* \rangle = \|\boldsymbol{w}\|_2 \|\boldsymbol{w}^*\|_2 \cos\theta = \|\boldsymbol{w}\|_2 \cos\theta$.
After an update, $\|\boldsymbol{w_{new}}\|_2 \cos\theta_{new}$ will be

$$\begin{aligned}\langle \boldsymbol{w} + y\boldsymbol{x}, \boldsymbol{w}^* \rangle &= \langle \boldsymbol{w}, \boldsymbol{w}^* \rangle + y \langle \boldsymbol{x}, \boldsymbol{w}^* \rangle \\ &= \langle \boldsymbol{w}, \boldsymbol{w}^* \rangle + |\langle \boldsymbol{x}, \boldsymbol{w}^* \rangle| \\ &\geq \langle \boldsymbol{w}, \boldsymbol{w}^* \rangle + \gamma\end{aligned} \tag{10}$$

Let's see the change of $\langle \boldsymbol{w_{new}}, \boldsymbol{w_{new}} \rangle = \|\boldsymbol{w_{new}}\|_2^2$,

$$\begin{aligned}\langle \boldsymbol{w} + y\boldsymbol{x}, \boldsymbol{w} + y\boldsymbol{x} \rangle &= \langle \boldsymbol{w}, \boldsymbol{w} \rangle + 2y \langle \boldsymbol{w}, \boldsymbol{x} \rangle + y^2 \langle \boldsymbol{x}, \boldsymbol{x} \rangle \\ &= \|\boldsymbol{w}\|_2^2 + 2y \langle \boldsymbol{w}, \boldsymbol{x} \rangle + \|\boldsymbol{x}\|_2^2\end{aligned} \tag{11}$$

Because $y \langle \boldsymbol{w}, \boldsymbol{x} \rangle < 0$ and $\|\boldsymbol{x}\|_2 \leq C$,

$$\begin{aligned}\langle \boldsymbol{w} + y\boldsymbol{x}, \boldsymbol{w} + y\boldsymbol{x} \rangle &= \|\boldsymbol{w}\|_2^2 + 2y \langle \boldsymbol{w}, \boldsymbol{x} \rangle + \|\boldsymbol{x}\|_2^2 \\ &\leq \|\boldsymbol{w}\|_2^2 + C^2\end{aligned} \tag{12}$$

Finally, suppose it converges after M updates, we have $\langle \boldsymbol{w}, \boldsymbol{w}^* \rangle \geq M\gamma$ and $\|\boldsymbol{w}\|_2^2 \leq MC^2$

$$\begin{aligned}1 = \cos\theta &= \frac{\langle \boldsymbol{w}, \boldsymbol{w}^* \rangle}{\|\boldsymbol{w}\|_2 \|\boldsymbol{w}^*\|_2} \\ &\geq \frac{M\gamma}{\sqrt{MC^2} \times 1} \\ &= \sqrt{M}\frac{\gamma}{C}\end{aligned} \tag{13}$$

which means $M \leq \frac{C^2}{\gamma^2}$.

Question 8 (Perceptron Loss)**.**

$$\begin{aligned}l(\boldsymbol{w}, \boldsymbol{x_t}, y_t) &= -y_t \langle \boldsymbol{w}, \boldsymbol{x_t} \rangle \mathbb{I}[\text{mistake on } \boldsymbol{x_t}] \\ &= -\min\{y_t \langle \boldsymbol{w}, \boldsymbol{x_i} \rangle, 0\}\end{aligned} \tag{14}$$

$$L(\boldsymbol{w}) = -\frac{1}{n}\sum_{t=1}^{n} y_t \langle \boldsymbol{w}, \boldsymbol{x_t} \rangle \mathbb{I}[\text{mistake on } \boldsymbol{x_t}] \tag{15}$$

2 Linear Regression

Question 9 (Least Square Regression)**.**

$$\min_{f:\mathcal{X}\to\mathcal{Y}} \mathbb{E}\|f(\boldsymbol{X}) - Y\|_2^2 \tag{16}$$

The optimal regression function is

$$f^*(\boldsymbol{x}) = m(x) = \mathbb{E}[Y|\boldsymbol{X} = \boldsymbol{x}] \tag{17}$$

Calculating it needs to know the distribution, i.e., all pairs $(\boldsymbol{X}, Y)$.

Question 10 (Bias-Variance Decomposition)**.**

$$\begin{aligned}\mathbb{E}\|f(\boldsymbol{X}) - Y\|_2^2 &= \mathbb{E}\|f(\boldsymbol{X}) - m(x) + m(x) - Y\|_2^2 \\ &= \mathbb{E}\|f(\boldsymbol{X}) - m(x)\|_2^2 + \mathbb{E}\|m(x) - Y\|_2^2 + 2\mathbb{E}\langle f(\boldsymbol{X}) - m(x), m(x) - Y\rangle \\ &= \mathbb{E}\|f(\boldsymbol{X}) - m(x)\|_2^2 + \mathbb{E}\|m(x) - Y\|_2^2 + \mathbb{E}\,\mathbb{E}_{Y|\boldsymbol{X}}[\langle f(\boldsymbol{X}) - m(x), m(x) - Y\rangle] \\ &= \mathbb{E}\|f(\boldsymbol{X}) - m(x)\|_2^2 + \mathbb{E}\|m(x) - Y\|_2^2 + \mathbb{E}\langle f(\boldsymbol{X}) - m(x), m(x) - \mathbb{E}_{Y|\boldsymbol{X}}[Y]\rangle \\ &= \mathbb{E}\|f(\boldsymbol{X}) - m(x)\|_2^2 + \mathbb{E}\|m(x) - Y\|_2^2 + \mathbb{E}\langle f(\boldsymbol{X}) - m(x), m(x) - m(x)\rangle \\ &= \mathbb{E}\|f(\boldsymbol{X}) - m(x)\|_2^2 + \underbrace{\mathbb{E}\|m(x) - Y\|_2^2}_{\text{noise (variance)}}\end{aligned} \tag{18}$$

The last term is the noise (variance), irrelevant to f. So, to minimize the squared error, we need $f \approx m$.
However, $m(\boldsymbol{x})$ is incalculable, because $\mathbb{E}[Y|\boldsymbol{X}=\boldsymbol{x}]$ is unknown.
Let's learn f_D from the training data D. Define $\bar{f}(\boldsymbol{X}) = \mathbb{E}_D[f_D(\boldsymbol{X})]$.

$$
\begin{aligned}
\underbrace{\mathbb{E}_{\boldsymbol{X},Y,D} \|f_D(\boldsymbol{X}) - Y\|_2^2}_{\text{test error}} &= \mathbb{E}_{\boldsymbol{X}} \|f_D(\boldsymbol{X}) - m(x)\|_2^2 + \underline{\mathbb{E}_{\boldsymbol{X},Y} \|m(x) - Y\|_2^2} \\
&= \mathbb{E}_{\boldsymbol{X},D} \left\|f_D(\boldsymbol{X}) - \bar{f}(\boldsymbol{X}) + \bar{f}(\boldsymbol{X}) - m(x)\right\|_2^2 + \underline{\mathbb{E}_{\boldsymbol{X},Y} \|m(x) - Y\|_2^2} \\
&= \mathbb{E}_{\boldsymbol{X},D} \left\|f_D(\boldsymbol{X}) - \bar{f}(\boldsymbol{X})\right\|_2^2 + \mathbb{E}_{\boldsymbol{X}} \left\|\bar{f}(\boldsymbol{X}) - m(x)\right\|_2^2 \\
&\quad + 2\mathbb{E}_{\boldsymbol{X},D} \left\langle f_D(\boldsymbol{X}) - \bar{f}(\boldsymbol{X}), \bar{f}(\boldsymbol{X}) - m(x)\right\rangle \\
&\quad + \underline{\mathbb{E}_{\boldsymbol{X},Y} \|m(x) - Y\|_2^2} \\
&= \dots + 2\mathbb{E}_{\boldsymbol{X}}\mathbb{E}_D \left\langle f_D(\boldsymbol{X}) - \bar{f}(\boldsymbol{X}), \bar{f}(\boldsymbol{X}) - m(x)\right\rangle + \underline{\dots} \\
&= \dots + 2\mathbb{E}_{\boldsymbol{X}} \left\langle \mathbb{E}_D[f_D(\boldsymbol{X})] - \bar{f}(\boldsymbol{X}), \bar{f}(\boldsymbol{X}) - m(x)\right\rangle + \underline{\dots} \\
&= \dots + 2\mathbb{E}_{\boldsymbol{X}} \left\langle \bar{f}(\boldsymbol{X}) - \bar{f}(\boldsymbol{X}), \bar{f}(\boldsymbol{X}) - m(x)\right\rangle + \underline{\dots} \\
&= \dots + 0 + \underline{\dots} \\
&= \mathbb{E}_{\boldsymbol{X},D} \left\|f_D(\boldsymbol{X}) - \bar{f}(\boldsymbol{X})\right\|_2^2 + \mathbb{E}_{\boldsymbol{X}} \left\|\bar{f}(\boldsymbol{X}) - m(x)\right\|_2^2 + \underline{\mathbb{E}_{\boldsymbol{X},Y} \|m(x) - Y\|_2^2} \\
&= \underbrace{\mathbb{E}_{\boldsymbol{X},D} \|f_D(\boldsymbol{X}) - \mathbb{E}_D[f_D(\boldsymbol{X})]\|_2^2}_{\text{variance}} + \underbrace{\mathbb{E}_{\boldsymbol{X}} \|\mathbb{E}_D[f_D(\boldsymbol{X})] - m(x)\|_2^2}_{\text{bias}^2} + \underbrace{\mathbb{E}_{\boldsymbol{X},Y} \|m(x) - Y\|_2^2}_{\text{noise (variance)}}
\end{aligned}
\tag{19}
$$

Question 11 (Sampling $\to$ Training)**.** Replace expectation with sample average: $(\boldsymbol{X_i}, Y_i)\,\tilde{P}$.

$$
\min_{f:\mathcal{X}\to\mathcal{Y}} \hat{\mathbb{E}} \|f(\boldsymbol{X}) - Y\|_2^2 \coloneqq \frac{1}{n}\sum_{i=1}^{n} \|f(\boldsymbol{X_i}) - Y_i\|_2^2 \tag{20}
$$

Uniform law of large numbers: as training data size $n \to$ argmin $\mathbb{E}$,
$\hat{\mathbb{E}} \to \mathbb{E}$ and (hopefully) argmin $\hat{\mathbb{E}} \to \mathbb{E}$.

Question 12 (Linear Regression)**.** Padding: $\boldsymbol{x} \leftarrow \begin{pmatrix}\boldsymbol{x}\\ 1\end{pmatrix}$, $W \leftarrow \begin{pmatrix}W\\ \boldsymbol{b}\end{pmatrix}$

$$
\begin{aligned}
X &= [\boldsymbol{x_1}, \dots, \boldsymbol{x_n}] \in \mathbb{R}^{(d+1)\times n}, \\
Y &= [\boldsymbol{y_1}, \dots, \boldsymbol{y_n}] \in \mathbb{R}^{t\times n}, \\
W &\in \mathbb{R}^{t\times(d+1)}, \\
\|A\|_F &= \sqrt{\sum_{ij} a_{ij}^2}
\end{aligned}
$$

Linear regression is:

$$
\min_{W\in\mathbb{R}^{t\times(d+1)}} \frac{1}{n}\|WX - Y\|_F^2 \tag{21}
$$

Question 13 (Optimality Condition)**.** If $\boldsymbol{w}$ is a minimizer (or maximizer) of a differentiable function f **over an open set**, then $f'(\boldsymbol{w}) = 0$.

Question 14 (Solving Linear Regression)**.**

$$
L(W) = \frac{1}{n}\|WX - Y\|_F^2 \tag{22}
$$

$$
\begin{aligned}
\nabla_W L(W) &= \frac{2}{n}(WX - Y)X^T = 0 \\
&\Rightarrow WXX^T = YX^T \\
&\Rightarrow W = YX^T(XX^T)^{-1}
\end{aligned}
\tag{23}
$$

Question 15 (Ill-Conditioning)**.** Slight pertubation leads to chaotic behavior, which happens whenever X is ill-conditioned, i.e., (close to) rank-deficient.

Rank-deficient X means:

1. two columns in X are linearly dependent (or simply the same)
2. but the corresponding y might be different

Question 16 (Ridge Regression)**.**

$$\min_{W} \frac{1}{n} \|WX - Y\|_F^2 + \lambda \|W\|_F^2 \tag{24}$$

$$\begin{aligned}\nabla_W L(W) &= \frac{2}{n}(WX - Y)X^T + 2\lambda W = 0 \\ &\Rightarrow WXX^T - YX^T + \lambda W = 0 \\ &\Rightarrow W(XX^T + n\lambda I) = YX^T\end{aligned} \tag{25}$$

$$\begin{aligned}&X = U\Sigma V^T \\ \Rightarrow &XX^T = U\Sigma(V^T V)\Sigma U^T = U\Sigma^2 U^T \\ \Rightarrow &XX^T + n\lambda I = U \underbrace{(\Sigma^2 + n\lambda I)}_{\text{strictly positive}} U^T \\ \Rightarrow &XX^T + n\lambda I \text{ is of full-rank}\end{aligned} \tag{26}$$

λ is regularization parameter. $\lambda = \infty \Rightarrow W \equiv \mathbf{0}$.

Question 17 (Regularization $\equiv$ Data Augmentation)**.**

$$\frac{1}{n} \|WX - Y\|_F^2 + \lambda \|W\|_F^2 = \frac{1}{n} \left\| W \begin{bmatrix} X & \sqrt{n\lambda} I \end{bmatrix} - \begin{bmatrix} Y & \mathbf{0} \end{bmatrix} \right\|_F^2 \tag{27}$$

3 Logistic Regression

Question 18 (Max Likelihood Estimation)**.** Let $\mathcal{Y} = \{0, 1\}$. Learn confidence $p(\boldsymbol{x}; \boldsymbol{w}) \coloneqq \Pr(Y = 1 | X = \boldsymbol{x})$.

$$\begin{aligned}\max_{\boldsymbol{w}} \Pr(Y_1 = y_1, \ldots, Y_n = y_n) &= \max_{\boldsymbol{w}} \prod_{i=1}^{n} \Pr(Y_i = y_i | X_i = x_i) \\ &\overset{\mathcal{Y}=\{0,1\}}{=} \max_{\boldsymbol{w}} \prod_{i=1}^{n} [p(\boldsymbol{x_i}; \boldsymbol{w})]^{y_i} [1 - p(\boldsymbol{x_i}; \boldsymbol{w})]^{1-y_i}\end{aligned} \tag{28}$$

Use negative log-likelihood:

$$\min_{\boldsymbol{w}} \sum_{i=1}^{n} [-y_i \log p(\boldsymbol{x_i}; \boldsymbol{w}) - (1 - y_i) \log(1 - p(\boldsymbol{x_i}; \boldsymbol{w}))] \tag{29}$$

Question 19 (Odds Ratio and Sigmoid)**.**

$$\text{Odds Ratio} = \frac{\Pr}{1 - \Pr} \tag{30}$$

Assume $\log \frac{p(\boldsymbol{x}; \boldsymbol{w})}{1 - p(\boldsymbol{x}; \boldsymbol{w})} = \langle \boldsymbol{x}, \boldsymbol{w} \rangle$.

The Sigmoid transformation is:

$$p(\boldsymbol{x}; \boldsymbol{w}) = \frac{1}{1 + \exp(-\langle \boldsymbol{x}, \boldsymbol{w} \rangle)} \tag{31}$$

Question 20 (Logistic Regression)**.** Plug the sigmoid in the negative log-likelihood:

$$\begin{aligned}&\min_{\boldsymbol{w}} \sum_{i=1}^{n} [-y_i \log p(\boldsymbol{x_i}; \boldsymbol{w}) - (1 - y_i) \log(1 - p(\boldsymbol{x_i}; \boldsymbol{w}))] \\ =&\min_{\boldsymbol{w}} \sum_{i=1}^{n} \left[y_i \log[1 + \exp(-\langle \boldsymbol{x_i}, \boldsymbol{w} \rangle)] - (1 - y_i) \log\left(1 - \frac{1}{1 + \exp(-\langle \boldsymbol{x_i}, \boldsymbol{w} \rangle)}\right) \right] \\ =&\min_{\boldsymbol{w}} \sum_{i=1}^{n} \left[y_i \log[1 + \exp(-\langle \boldsymbol{x_i}, \boldsymbol{w} \rangle)] - (1 - y_i) \log\left(\frac{\exp(-\langle \boldsymbol{x_i}, \boldsymbol{w} \rangle)}{1 + \exp(-\langle \boldsymbol{x}, \boldsymbol{w} \rangle)}\right) \right] \\ =&\min_{\boldsymbol{w}} \sum_{i=1}^{n} [y_i \log[1 + \exp(-\langle \boldsymbol{x_i}, \boldsymbol{w} \rangle)] + (1 - y_i) [\langle \boldsymbol{x_i}, \boldsymbol{w} \rangle + \log(1 + \exp(-\langle \boldsymbol{x}, \boldsymbol{w} \rangle))]] \\ =&\min_{\boldsymbol{w}} \sum_{i=1}^{n} [\log[1 + \exp(-\langle \boldsymbol{x_i}, \boldsymbol{w} \rangle)] + (1 - y_i) \langle \boldsymbol{x_i}, \boldsymbol{w} \rangle]\end{aligned} \tag{32}$$

Because $y_i \in \{0, 1\}$, let's map it to $\{\pm 1\}$.

$$\begin{aligned} L(\boldsymbol{w}) &\overset{y_i \in \{0,1\}}{=} \log[1 + \exp(-\langle \boldsymbol{x_i}, \boldsymbol{w} \rangle)] + (1 - y_i) \langle \boldsymbol{x_i}, \boldsymbol{w} \rangle \\ &\overset{y_i \in \{0,1\}}{=} \log[1 + \exp(-\langle \boldsymbol{x_i}, \boldsymbol{w} \rangle)] + \log\left[\exp((1 - y_i) \langle \boldsymbol{x_i}, \boldsymbol{w} \rangle)\right] \\ &\overset{y_i \in \{0,1\}}{=} \log[1 + \exp(-\langle \boldsymbol{x_i}, \boldsymbol{w} \rangle) \cdot \exp((1 - y_i) \langle \boldsymbol{x_i}, \boldsymbol{w} \rangle)] \\ &\overset{y_i \in \{0,1\}}{=} \log[\exp((1 - y_i) \langle \boldsymbol{x_i}, \boldsymbol{w} \rangle) + \exp(-y_i \langle \boldsymbol{x_i}, \boldsymbol{w} \rangle)] \\ &\overset{y_i \in \{0,1\}}{=} \begin{cases} \log[\exp(\langle \boldsymbol{x_i}, \boldsymbol{w} \rangle) + 1] & y_i = 0 \\ \log[1 + \exp(-\langle \boldsymbol{x_i}, \boldsymbol{w} \rangle)] & y_i = 1 \end{cases} \\ &\overset{y_i \in \{\pm 1\}}{=} \log[1 + \exp(-y_i \langle \boldsymbol{x_i}, \boldsymbol{w} \rangle)] \end{aligned} \tag{33}$$

Question 21 (Multi-Class: Sigmoid $\to$ Softmax)**.**

$$\Pr(Y = k | X = \boldsymbol{x}; \boldsymbol{W} = [\boldsymbol{w_1}, \ldots, \boldsymbol{w_c}]) = \frac{\exp(\langle \boldsymbol{x}, \boldsymbol{w_k} \rangle)}{\sum_{l=1}^{c} \exp(\langle \boldsymbol{x}, \boldsymbol{w_l} \rangle)} \tag{34}$$

Maximum likelihood estimation (log loss, cross-entropy loss):

$$\min_{\boldsymbol{W}} \sum_{i=1}^{n} \left[-\log \frac{\exp(\langle \boldsymbol{x}, \boldsymbol{w_k} \rangle)}{\sum_{l=1}^{c} \exp(\langle \boldsymbol{x}, \boldsymbol{w_l} \rangle)} \right] \tag{35}$$

4 Hard-Margin Support Vector Machines

Question 22 (Distance from a Point to a Hyperplane)**.** Let $H \coloneqq \{\boldsymbol{x} : \langle \boldsymbol{x}, \boldsymbol{w} \rangle + b = 0\}$, $\boldsymbol{x}$ be any vector in H.

$$\text{Distance}(\boldsymbol{x_i}, \boldsymbol{w}) = \frac{|\langle \boldsymbol{x_i} - \boldsymbol{x}, \boldsymbol{w} \rangle|}{\|\boldsymbol{w}\|_2} = \frac{|\langle \boldsymbol{x_i}, \boldsymbol{w} \rangle - \langle \boldsymbol{x}, \boldsymbol{w} \rangle|}{\|\boldsymbol{w}\|_2} \overset{\boldsymbol{x} \in H}{=} \frac{|\langle \boldsymbol{x_i}, \boldsymbol{w} \rangle + b|}{\|\boldsymbol{w}\|_2} \overset{y_i \hat{y}_i > 0}{=} \frac{y_i \hat{y}_i}{\|\boldsymbol{w}\|_2} \tag{36}$$

Question 23 (Margin Maximization)**.** Margin is the smallest distance to H among all separable data.

$$\max_{\boldsymbol{w}, b} \min_i \frac{y_i \hat{y}_i}{\|\boldsymbol{w}\|_2}, \text{ such that } \forall i, y_i \hat{y}_i > 0 \tag{37}$$

Let $c > 0$, then $\boldsymbol{w} = c\boldsymbol{w}, b = cb$ keeps the loss same:

$$\begin{aligned} \max_{\boldsymbol{w}, b} \min_i \frac{c y_i \hat{y}_i}{\|c\boldsymbol{w}\|_2} &= \max_{\boldsymbol{w}, b} \min_i \frac{y_i(\langle \boldsymbol{x}, c\boldsymbol{w} \rangle + cb)}{\|c\boldsymbol{w}\|_2} \\ &= \max_{\boldsymbol{w}, b} \min_i \frac{c y_i(\langle \boldsymbol{x}, \boldsymbol{w} \rangle + b)}{c \|\boldsymbol{w}\|_2} \\ &= \max_{\boldsymbol{w}, b} \min_i \frac{y_i(\langle \boldsymbol{x}, \boldsymbol{w} \rangle + b)}{\|\boldsymbol{w}\|_2} \\ &= \max_{\boldsymbol{w}, b} \min_i \frac{y_i \hat{y}_i}{\|\boldsymbol{w}\|_2} \end{aligned} \tag{38}$$

Let $c = \frac{1}{\min_i y_i \hat{y}_i}$,

$$\begin{aligned} \max_{\boldsymbol{w}, b} \min_i \frac{c y_i \hat{y}_i}{c \|\boldsymbol{w}\|_2} &= \max_{\boldsymbol{w}, b} \frac{1}{c \|\boldsymbol{w}\|_2} \\ &= \max_{\boldsymbol{w}, b} \frac{1}{\|\boldsymbol{w}\|_2} \text{ s.t. } \min_i y_i \hat{y}_i = 1 \end{aligned} \tag{39}$$

Max $\to$ Min:

$$\min_{\boldsymbol{w}, b} \frac{1}{2} \|\boldsymbol{w}\|_2^2 \text{ s.t. } \forall i, y_i \hat{y}_i \geq 1 \tag{40}$$

Question 24 (Hard-Margin SVM v.s. Perceptron)**.**

$$\text{Hard-Margin SVM:} \quad \min_{\boldsymbol{w}, b} \frac{1}{2} \|\boldsymbol{w}\|_2^2 \quad \text{s.t. } \forall i, y_i \hat{y}_i \geq 1 \tag{41}$$

$$\text{Perceptron:} \quad \min_{\boldsymbol{w}, b} 0 \quad \text{s.t. } \forall i, y_i \hat{y}_i \geq 1 \tag{42}$$

Question 25 (Lagrangian Dual)**.** Dual variables $\boldsymbol{\alpha} \in \mathbb{R}^n$.

$$\begin{aligned}
\min_{\boldsymbol{w},b} \max_{\boldsymbol{\alpha}\geq 0} \frac{1}{2}\|\boldsymbol{w}\|_2^2 - \sum_i \alpha_i \left[y_i(\langle \boldsymbol{x_i}, \boldsymbol{w}\rangle + b) - 1\right] &= \min_{\boldsymbol{w},b} \begin{cases} +\infty, & \text{if } \exists i, y_i(\langle \boldsymbol{x_i}, \boldsymbol{w}\rangle + b) < 1 \quad (\alpha_i = +\infty) \\ \frac{1}{2}\|\boldsymbol{w}\|_2^2, & \text{if } \forall i, y_i(\langle \boldsymbol{x_i}, \boldsymbol{w}\rangle + b) \geq 1 \quad (\forall i, \alpha_i = 0) \end{cases} \\
&= \min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|_2^2, \quad \text{s.t. } \forall i, y_i(\langle \boldsymbol{x_i}, \boldsymbol{w}\rangle + b) \geq 1 \\
&= \min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|_2^2 \text{ s.t. } \forall i, y_i\hat{y}_i \geq 1
\end{aligned} \tag{43}$$

Swap min and max:

$$\max_{\boldsymbol{\alpha}\geq 0} \min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|_2^2 - \sum_i \alpha_i \left[y_i(\langle \boldsymbol{x_i}, \boldsymbol{w}\rangle + b) - 1\right] \tag{44}$$

Solve inner problem by setting derivative to 0:

$$\frac{\delta}{\delta \boldsymbol{w}} = \boldsymbol{w} - \sum_i \alpha_i y_i \boldsymbol{x_i} = 0, \qquad \frac{\delta}{\delta b} = -\sum_i \alpha_i y_i = 0, \tag{45}$$

Plug them into the loss:

$$\begin{aligned}
L(\boldsymbol{\alpha}) &= \min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|_2^2 - \sum_i \alpha_i \left[y_i(\langle \boldsymbol{x_i}, \boldsymbol{w}\rangle + b) - 1\right] \\
&= \frac{1}{2}\left\|\sum_i \alpha_i y_i \boldsymbol{x_i}\right\|_2^2 - \sum_i \alpha_i y_i \langle \boldsymbol{x_i}, \boldsymbol{w}\rangle - \sum_i \alpha_i y_i b + \sum_i \alpha_i \\
&= \frac{1}{2}\left\|\sum_i \alpha_i y_i \boldsymbol{x_i}\right\|_2^2 - \left\langle \sum_i \alpha_i y_i \boldsymbol{x_i}, \sum_i \alpha_i y_i \boldsymbol{x_i} \right\rangle - b\sum_i \alpha_i y_i + \sum_i \alpha_i \\
&= \frac{1}{2}\left\|\sum_i \alpha_i y_i \boldsymbol{x_i}\right\|_2^2 - \left\|\sum_i \alpha_i y_i \boldsymbol{x_i}\right\|_2^2 + \sum_i \alpha_i \\
&= \sum_i \alpha_i - \frac{1}{2}\left\|\sum_i \alpha_i y_i \boldsymbol{x_i}\right\|_2^2, \qquad \text{s.t. } \sum_i \alpha_i y_i = 0
\end{aligned} \tag{46}$$

So, 44 is solved as:

$$\max_{\boldsymbol{\alpha}\geq 0} \sum_i \alpha_i - \frac{1}{2}\left\|\sum_i \alpha_i y_i \boldsymbol{x_i}\right\|_2^2, \qquad \text{s.t. } \sum_i \alpha_i y_i = 0 \tag{47}$$

Max → min and expand the norm:

$$\min_{\boldsymbol{\alpha}\geq 0} - \sum_i \alpha_i + \frac{1}{2}\sum_i\sum_j \alpha_i\alpha_j y_i y_j \underbrace{\langle \boldsymbol{x_i}, \boldsymbol{x_j}\rangle}_{\text{Kernel, closed form w.r.t. } \boldsymbol{x_i}, \boldsymbol{x_j}}, \qquad \text{s.t. } \sum_i \alpha_i y_i = 0 \tag{48}$$

Question 26 (Support Vectors)**.** From 45, we know $\boldsymbol{w} = \sum_i \alpha_i y_i \boldsymbol{x_i}$.
Vectors with $\alpha_i \neq 0$ are support vectors, which lie on the margin.

5 Soft-Margin Support Vector Machines

Question 27 (Goal)**.** minimize over $\boldsymbol{w}, b$,

$$\Pr(Y \neq \operatorname{sign}(\hat{Y})) = \Pr(Y\hat{Y} \leq 0) = \mathbb{E} \underbrace{\mathbb{I}[Y\hat{Y} \leq 0]}_{indicator function} := \mathbb{E}\, l_{0-1}(Y\hat{Y}) \tag{49}$$

where $\hat{Y} = \langle X, \boldsymbol{w}\rangle + b, Y = \pm 1$.

$$\begin{aligned}
\min_{\hat{Y}:\mathcal{X}\to\mathbb{R}} \mathbb{E}\, l_{0-1}(Y\hat{Y}) &= \min_{\hat{Y}:\mathcal{X}\to\mathbb{R}} \mathbb{E}_X \mathbb{E}_{Y|X}\, l_{0-1}(Y\hat{Y}) \\
&= \mathbb{E}_X \min_{\hat{Y}:\mathcal{X}\to\mathbb{R}} \mathbb{E}_{Y|X}\, l_{0-1}(Y\hat{Y})
\end{aligned} \tag{50}$$

Minimizing the 0-1 error is **NP-hard**.

Question 28 (Bayes Rule).

$$\eta(\boldsymbol{x}) := \underset{\hat{y}\in\mathbb{R}}{\operatorname{argmax}} \Pr(Y = \hat{y}|X = \boldsymbol{x}) \tag{51}$$

$$\eta(\boldsymbol{x}) := \underset{\hat{y}\in\mathbb{R}}{\operatorname{argmin}} \mathbb{E}_{Y|X=\boldsymbol{x}}\, l_{0-1}(Y\hat{y}) \tag{52}$$

Question 29 (Classification Calibrated). A loss $l(y\hat{y})$ is classification calibrated, iff $\forall \boldsymbol{x}$,

$$\hat{y}(\boldsymbol{x}) := \underset{\hat{y}\in\mathbb{R}}{\operatorname{argmin}} \mathbb{E}_{Y|X=\boldsymbol{x}}\, l(Y\hat{y})$$

has the same sign as the Bayes rule $\eta(\boldsymbol{x}) := \operatorname{argmin}_{\hat{y}\in\mathbb{R}} \mathbb{E}_{Y|X=\boldsymbol{x}}\, l_{0-1}(Y\hat{y})$
Notice: $\eta(\boldsymbol{x}), \hat{y}(\boldsymbol{x})$ provide the **score**, their sign provides the prediction.

Question 30 (Characterization under Convexity). Any **convex** loss l is classification calibrated iff l is differentiable at 0 and $l'(0) < 0$.

Question 31 (Hinge Loss).

$$l_{hinge}(y\hat{y}) = (1 - y\hat{y})^+ := \max\{0, 1 - y\hat{y}\} = \begin{cases} 1 - y\hat{y}, & \text{if } y\hat{y} < 1 \\ 0, & \text{otherwise} \end{cases} \tag{53}$$

The classifier that minimizes the expected hinge loss minimizes the expected 0-1 loss.

Question 32 (Soft-Margin SVM).

$$\min_{\boldsymbol{w},b} \frac{1}{2} \|\boldsymbol{w}\|_2^2 + C \cdot \sum_i l_{hinge}(y_i\hat{y}_i), \quad \text{s.t. } \hat{y}_i = \langle \boldsymbol{x_i}, \boldsymbol{w}\rangle + b \tag{54}$$

Question 33 (Lagrangian Dual). Apply $C \cdot l_{hinge}(t) := \max\{0, C(1-t)\} = \max_{0\le\alpha\le C} \alpha(1-t)$

$$\min_{\boldsymbol{w},b} \max_{0\le\boldsymbol{\alpha}\le C} \frac{1}{2} \|\boldsymbol{w}\|_2^2 + \sum_i \alpha_i(1 - y_i\hat{y}_i) \tag{55}$$

Swap min with max:

$$\max_{0\le\boldsymbol{\alpha}\le C} \min_{\boldsymbol{w},b} \frac{1}{2} \|\boldsymbol{w}\|_2^2 + \sum_i \alpha_i(1 - y_i\hat{y}_i) \tag{56}$$

Solve it by setting derivative to 0:

$$\frac{\delta}{\delta \boldsymbol{w}} = \boldsymbol{w} - \sum_i \alpha_i y_i \boldsymbol{x_i} = 0, \qquad \frac{\delta}{\delta b} = -\sum_i \alpha_i y_i = 0, \tag{57}$$

Plug them into the loss:

$$\begin{aligned} \max_{0\le\boldsymbol{\alpha}\le C} \min_{\boldsymbol{w},b} \frac{1}{2} \|\boldsymbol{w}\|_2^2 + \sum_i \alpha_i(1 - y_i\hat{y}_i) &= \max_{0\le\boldsymbol{\alpha}\le C} \min_{\boldsymbol{w},b} \frac{1}{2} \|\boldsymbol{w}\|_2^2 + \sum_i \alpha_i \left[1 - y_i(\langle \boldsymbol{x_i}, \boldsymbol{w}\rangle + b)\right] \\ &= \max_{0\le\boldsymbol{\alpha}\le C} \sum_i \alpha_i - \frac{1}{2} \left\| \sum_i \alpha_i y_i \boldsymbol{x_i} \right\|_2^2, \qquad \text{s.t. } \sum_i \alpha_i y_i = 0 \end{aligned} \tag{58}$$

Max $\rightarrow$ min and expand the norm:

$$\min_{0\le\boldsymbol{\alpha}\le C} -\sum_i \alpha_i + \frac{1}{2} \sum_i \sum_j \alpha_i\alpha_j y_i y_j \underbrace{\langle \boldsymbol{x_i}, \boldsymbol{x_j}\rangle}_{\text{Kernel, closed form w.r.t. } \boldsymbol{x_i}, \boldsymbol{x_j}}, \qquad \text{s.t. } \sum_i \alpha_i y_i = 0 \tag{59}$$

$C \rightarrow \infty \Rightarrow$ Hard-margin SVM, $C \rightarrow 0 \Rightarrow$ a constant classifier

6 Reproducing Kernels

Question 34 ((Reproducing) Kernels). $k : (X) \times \mathcal{X} \rightarrow \mathbb{R}$ is a (reproducing) kernel iff there exists some $\Phi : \mathcal{X} \rightarrow \mathcal{H}$ so that $\langle \Phi(\boldsymbol{x}), \Phi(\boldsymbol{z})\rangle = k(\boldsymbol{x}, \boldsymbol{z})$.

- A feature transform Φ determines the corresponding kernel k.
- A kernel k determines some feature transforms Φ, but may not be unique.
  E.g. $\langle \phi(\boldsymbol{x}), \phi(\boldsymbol{z})\rangle = \langle \phi'(\boldsymbol{x}), \phi'(\boldsymbol{z})\rangle$
  1. $\phi(x) := [x_1^2, \sqrt{2}x_1x_2] \in \mathbb{R}^2$
  2. $\phi'(x) := [x_1^2, x_1x_2, x_1x_2] \in \mathbb{R}^3$

Question 35 (Mercer's Theorem)**.** $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a kernel,
iff $\forall n \in \mathbb{N}, \forall \boldsymbol{x_1}, \dots, \boldsymbol{x_n} \in \mathcal{X}$, the kernel matrix K such that $K_{ij} \coloneqq k(\boldsymbol{x_i}, \boldsymbol{x_j})$ is symmetric and positive semi-definite (PSD).

$$k \text{ is a kernel} \Leftrightarrow \begin{cases} K_{ij} = K_{ji} & \text{(symmetric)} \\ \langle \boldsymbol{\alpha}, K\boldsymbol{\alpha} \rangle = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j K_{ij} \geq 0 \qquad \forall \boldsymbol{\alpha} \in \mathbb{R}^n & \text{(PSD)} \end{cases} \tag{60}$$

Question 36 (Symmetric PSD)**.** For a symmetric matrix A, the following conditions are equivalent.

1. $A \succeq 0$
2. $A = U^T U$ for some matrix U
3. $x^T A x \geq 0$ for every $x \in \mathbb{R}^n$
4. All principal minors of A are nonnegative

7 Gradient Descent

Question 37 (Gradient Descent Template)**.** Choose initial point $x^{(0)} \in \mathbb{R}^d$ and repeat:

$$x^{(k)} = x^{(k-1)} - \underbrace{\eta}_{\text{step size}} \nabla f(x^{(k-1)}), \quad k = 1, 2, \dots \tag{61}$$

Question 38 (Interpretation from Taylor Expansion)**.** Expand f locally at x:

$$\begin{aligned} f(y) &\approx f(x) + \nabla f(x)^T (y - x) + \frac{1}{2t} \|y - x\|_2^2 \\ \Rightarrow \min_y f(y) &\approx \min_y \left[f(x) + \nabla f(x)^T (y - x) + \frac{1}{2t} \|y - x\|_2^2 \right] \end{aligned} \tag{62}$$

When $y - x = \frac{\nabla f(x)}{-2\frac{1}{2t}} = -t\nabla f(x) \Longrightarrow y = x - t\nabla f(x)$, it reaches the minimum.

Question 39 (L-smooth or L-Lipschitz Continuous)**.** f is convex and differentiable. ∇f is L-Lipschitz continuous (L-smooth):

$$L\boldsymbol{I} \succeq \nabla^2 f(x), \forall x \tag{63}$$

Question 40 (Convergence Rate for Convex Case)**.** Assume f is L-smooth. Gradient descent with fixed step size $t \leq \frac{1}{L}$ satisfies:

$$f(x^{(k)}) - f(x^*) \leq \frac{\left\|x^{(0)} - x^*\right\|_2^2}{2tk} \tag{64}$$

We say gradient descent has convergence rate $O(\frac{1}{k})$, i.e. $f(x^{(k)}) - f(x^*) \leq \epsilon$ can be achieved using only $O(\frac{1}{\epsilon})$ iterations.

Proof

$$\begin{aligned} f(y) &= f(x) + \nabla f(x)^T (y - x) + \frac{1}{2}(y - x)^T \nabla^2 f(\xi)(y - x) \\ &\leq f(x) + \nabla f(x)^T (y - x) + \frac{1}{2} L \|y - x\|_2^2 \qquad (L\text{-smooth}, L\boldsymbol{I} \succeq \nabla^2 f(\xi)) \end{aligned} \tag{65}$$

Plug in gradient descent:

$$\begin{aligned} f(x^+) &= f(y) \\ &\leq f(x) + \nabla f(x)^T (x - t\nabla f(x) - x) + \frac{1}{2} L \|x - t\nabla f(x) - x\|_2^2 \\ &= f(x) - (1 - \frac{1}{2} Lt) t \|\nabla f(x)\|_2^2 \\ &\leq f(x) - \frac{1}{2} t \|\nabla f(x)\|_2^2 \quad (t \leq \frac{1}{L}) \end{aligned} \tag{66}$$

f is convex $\Rightarrow f(x^*) \geq f(x) + \nabla f(X)^T (x^* - x) \Rightarrow f(x) \leq f(x^*) + \nabla f(x)^T (x - x^*)$

Plug this into 66:

$$
\begin{aligned}
& f(x^+) \le f(x^*) + \nabla f(x)^T(x - x^*) - \frac{1}{2}t \left\|\nabla f(x)\right\|_2^2 \\
\Rightarrow & f(x^+) - f(x^*) \le \frac{1}{2t}\left(2t\nabla f(x)^T(x - x^*) - t^2 \left\|\nabla f(x)\right\|_2^2\right) \\
\Rightarrow & f(x^+) - f(x^*) \le \frac{1}{2t}\left(2t\nabla f(x)^T(x - x^*) - t^2 \left\|\nabla f(x)\right\|_2^2 - \left\|x - x^*\right\|_2^2 + \left\|x - x^*\right\|_2^2\right) \\
\Rightarrow & f(x^+) - f(x^*) \le \frac{1}{2t}\left(\left\|x - x^*\right\|_2^2 - (\left\|x - x^*\right\|_2^2 + t^2 \left\|\nabla f(x)\right\|_2^2 - 2t\nabla f(x)^T(x - x^*))\right) \\
\Rightarrow & f(x^+) - f(x^*) \le \frac{1}{2t}\left(\left\|x - x^*\right\|_2^2 - \left\|x - x^* - t\nabla f(x)\right\|_2^2\right) \\
\Rightarrow & f(x^+) - f(x^*) \le \frac{1}{2t}\left(\left\|x - x^*\right\|_2^2 - \left\|x^+ - x^*\right\|_2^2\right)
\end{aligned} \tag{67}
$$

Viewing x^+ as $x^{(i)}$ and x as $x^{(x-1)}$:

$$
\begin{aligned}
\sum_{i=1}^{k}\left(f(x^{(i)}) - f(x^*)\right) &\le \sum_{i=1}^{k}\frac{1}{2t}\left(\left\|x^{(i-1)} - x^*\right\|_2^2 - \left\|x^{(i)} - x^*\right\|_2^2\right) \\
&= \frac{1}{2t}\left(\left\|x^{(0)} - x^*\right\|_2^2 - \left\|x^{(k)} - x^*\right\|_2^2\right) \\
&\le \frac{1}{2t}\left\|x^{(0)} - x^*\right\|_2^2
\end{aligned} \tag{68}
$$

which implies

$$
f(x^{(k)}) \le \frac{1}{k}\sum_{i=1}^{k} f(x^{(i)}) \le f(x^*) + \frac{\left\|x^{(0)} - x^*\right\|_2^2}{2tk} \tag{69}
$$

Question 41 (Convergence Rate for Strong Convexity)**.** f is differentiable, L-smooth, and m-strongly convex.
m-strong convexity of f means $f(x) - \frac{m}{2}\left\|x\right\|_2^2$ is convex, i.e. $\nabla^2 f(x) \succeq m\boldsymbol{I}$
Then, there is a constant $0 < \gamma < 1$ such that gradient descent with fixed step size $t \le \frac{2}{m+L}$ satisfies:

$$
f(x^{(k)}) - f(x^*) \le \gamma^k \frac{L}{2}\left\|x^{(0)} - x^*\right\|_2^2 \tag{70}
$$

Rate is $O(\gamma^k)$. Only $O(\log_{\frac{1}{\gamma}}(\frac{1}{\epsilon}))$ iterations needed.

Question 42 (Convergence Rate for Non-Convex Case)**.** f is differentiable and L-smooth, but non-convex.
Gradient descent with fixed step size $t \le \frac{1}{L}$ satisifes:

$$
\min_{i=0,\dots,k}\left\|\nabla f(x^{(i)})\right\|_2 \le \sqrt{(}\frac{2(f(x^{(0)} - f^*))}{t(k+1)}) \tag{71}
$$

Rate is $O(\frac{1}{\sqrt{k}})$ for finding stationary point. $O(\frac{1}{\epsilon^2})$ iterations are needed.

Question 43 (Convergence Rate for Stochastic Gradient Descent)**.** For convex and L-smooth f,

- Gradient Descent

$$
\boldsymbol{w}^+ = \boldsymbol{w} - t \cdot \frac{1}{n}\sum_{i=1}^{n}\nabla f_i(\boldsymbol{w}) \tag{72}
$$

  - Step size $t \le \frac{1}{L}$
  - Time complexity $O(\frac{n}{\epsilon})$

- Stochastic Gradient Descent

$$
\boldsymbol{w}^+ = \boldsymbol{w} - t \cdot \nabla f_{I_{random}}(\boldsymbol{w}) \tag{73}
$$

  - Step size $t = \frac{1}{k}, k = 1, 2, 3, \dots$ (adaptive step size)
  - Time complexity $O(\frac{1}{\epsilon^2})$

8 Fully-Connected Neural Networks

Question 44 (Forward and Backward Pass of a 2-Layer MLP)**.** A 2-layer MLP (k is the NN width, c is the output dim):

$$\boldsymbol{x} = \text{input} \qquad (\boldsymbol{x} \in \mathbb{R}^d) \tag{74}$$

$$\boldsymbol{z} = \boldsymbol{W}\boldsymbol{x} + \boldsymbol{b_1} \qquad (\boldsymbol{W} \in \mathbb{R}^{k\times d}, \boldsymbol{z}, \boldsymbol{b} \in \mathbb{R}^k) \tag{75}$$

$$\boldsymbol{h} = \text{ReLU}(\boldsymbol{z}) \qquad (\boldsymbol{h} \in \mathbb{R}^k) \tag{76}$$

$$\boldsymbol{\theta} = \boldsymbol{U}\boldsymbol{h} + \boldsymbol{b_2} \qquad (\boldsymbol{U} \in \mathbb{R}^{c\times k}, \boldsymbol{\theta}, \boldsymbol{b_2} \in \mathbb{R}^c) \tag{77}$$

$$J = \frac{1}{2}\|\boldsymbol{\theta} - \boldsymbol{y}\|_2^2 \qquad (\boldsymbol{y} \in \mathbb{R}^c, J \in \mathbb{R}) \tag{78}$$

$$\text{ReLU} = \begin{cases} x & x > 0 \\ 0 & x \le 0 \end{cases} \tag{79}$$

$$\text{ReLU}' = \begin{cases} 1 & x > 0 \\ 0 & x \le 0 \end{cases} \tag{80}$$

Backward pass ($\odot$ is the Hadamard product, i.e. element-wise product):

$$\frac{\delta J}{\delta \boldsymbol{\theta}} = \boldsymbol{\theta} - \boldsymbol{y} \tag{81}$$

$$\frac{\delta J}{\delta \boldsymbol{U}} = \frac{\delta J}{\delta \boldsymbol{\theta}} \circ \frac{\delta \boldsymbol{\theta}}{\delta \boldsymbol{U}} = (\boldsymbol{\theta} - \boldsymbol{y})\boldsymbol{h}^T \tag{82}$$

$$\frac{\delta J}{\delta \boldsymbol{b_2}} = \frac{\delta J}{\delta \boldsymbol{\theta}} \circ \frac{\delta \boldsymbol{\theta}}{\delta \boldsymbol{b_2}} = \boldsymbol{\theta} - \boldsymbol{y} \tag{83}$$

$$\frac{\delta J}{\delta \boldsymbol{h}} = \frac{\delta J}{\delta \boldsymbol{\theta}} \circ \frac{\delta \boldsymbol{\theta}}{\delta \boldsymbol{h}} = \boldsymbol{U^T}(\boldsymbol{\theta} - \boldsymbol{y}) \tag{84}$$

$$\frac{\delta J}{\delta \boldsymbol{z}} = \frac{\delta J}{\delta \boldsymbol{h}} \circ \frac{\delta \boldsymbol{h}}{\delta \boldsymbol{z}} = \boldsymbol{U^T}(\boldsymbol{\theta} - \boldsymbol{y}) \odot \text{ReLU}'(\boldsymbol{z}) \tag{85}$$

$$\frac{\delta J}{\delta \boldsymbol{W}} = \frac{\delta J}{\delta \boldsymbol{z}} \circ \frac{\delta \boldsymbol{z}}{\delta \boldsymbol{W}} = \boldsymbol{U^T}(\boldsymbol{\theta} - \boldsymbol{y}) \odot \text{ReLU}'(\boldsymbol{z})\boldsymbol{x}^T \tag{86}$$

$$\frac{\delta J}{\delta \boldsymbol{b_1}} = \frac{\delta J}{\delta \boldsymbol{z}} \circ \frac{\delta \boldsymbol{z}}{\delta \boldsymbol{b_1}} = \boldsymbol{U^T}(\boldsymbol{\theta} - \boldsymbol{y}) \odot \text{ReLU}'(\boldsymbol{z}) \tag{87}$$

$$\tag{88}$$

Question 45 (Universal Approximation Theorem by 2-Layer NNs)**.** For any continuous function $f : \mathbb{R}^d \to \mathbb{R}^c$ and any $\epsilon > 0$, there exists $k \in \mathbb{N}, \boldsymbol{W} \in \mathbb{R}^{k\times d}, \boldsymbol{b} \in \mathbb{R}^k, \boldsymbol{U} \in \mathbb{R}^{c\times k}$ such that

$$\sup_{\boldsymbol{x}} \|f(\boldsymbol{x}) - g(\boldsymbol{x})\|_2 < \epsilon \tag{89}$$

where $g(\boldsymbol{x}) = \boldsymbol{U}(\sigma(\boldsymbol{W}\boldsymbol{x} + \boldsymbol{b}))$ and σ is the element-wise ReLU operation.
As long as the 2-layer MLP is wide enough, it can approximate any continuous function arbitrarily closely.

9 Convolutional Neural Networks

Question 46 (Controlling the Convolution)**.** Hyperparameters.

- **Filter (kernel) size**: width *times* height.
- **Number of filters (kernels)**.
  Weights are not shared between different filters (kernels)
- **Stride**: how many pixels the filter moves each time.
- **Padding**: add zeros around the boundary of the input.

Question 47 (Size Calculation)**.**

$$\begin{aligned} \text{Input size: } & m \times n \times c_{in} \\ \text{Filter size: } & a \times b \times c_{in} \\ \text{Stride: } & s \times t \\ \text{Padding: } & p \times q \end{aligned}$$

Output size:

$$\left\lfloor 1 + \frac{m + 2p - a}{s} \right\rfloor \times \left\lfloor 1 + \frac{n + 2q - b}{t} \right\rfloor \tag{90}$$

Part II
For Final

10 Transformer

Question 48 (Attention Layer Inputs and Outputs)**.** Inputs: $V \in \mathcal{R}^{n\times d}$, $K \in \mathcal{R}^{n\times d}$, $Q \in \mathcal{R}^{m\times d}$, Outputs: an $m \times d$ matrix.

- **Self Attention**: $m = n$,
- **Cross Attention**: $m \neq n$ where m is the sequence length of decoder, n is the sequence length of encoder.

Question 49 (Attention Layer Calculation)**.**

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V \tag{91}$$

Softmax is row-wise, i.e. for each row of QK^T, it is normalized to sum to 1.

Question 50 (Learnable Attention Layer)**.**

$$\text{Attention}(XW^v, XW^k, XW^q) = \text{softmax}\left(\frac{XW^q(XW^k)^T}{\sqrt{d}}\right)XW^v \tag{92}$$

Question 51 (RMSNorm (LLaMA's Choice))**.**

$$\bar{a_i} = \frac{a_i}{\text{RMS}(a)}\gamma = \frac{a_i}{\sqrt{\frac{1}{d}\sum_{j=1}^{d} a_j^2}}\gamma \tag{93}$$

Question 52 (Transformer Loss)**.**

$$\min_W \hat{\mathbb{E}}\left[-\left\langle Y, \log \hat{Y}\right\rangle\right] \tag{94}$$

Y is output sequence, one-hot;
$\hat{Y}$ is the predicted probabilities

Question 53 (Transformer Implementation)**.** As following.

```
import torch
import torch.nn as nn
import torch.F as F
import math

class RMSNorm(nn.Module):
  def __init__(self, hidden_dim, eps = 1e-6):
    super().__init__()
    self.eps = eps
    self.weight = nn.Parameter(torch.ones(hidden_dim))

  def forward(self, hidden_state):
    norm = hidden_state.pow(2).mean(-1, keepdim = True)
    output = hidden_state * self.weight * torch.rsqrt(norm + self.eps)
    return output

class MultiHeadAttention(nn.Module):
  def __init__(self, hidden_dim, num_heads):
    super().__init__()
    self.hidden_dim = hidden_dim
    self.num_heads = num_heads
    self.head_dim = hidden_dim // num_heads
    self.q_linear = nn.linear(hidden_dim, hidden_dim)
    self.k_linear = nn.linear(hidden_dim, hidden_dim)
    self.v_linear = nn.linear(hidden_dim, hidden_dim)
    self.o_linear = nn.linear(hidden_dim, hidden_dim)
    self.norm = RMSNorm(hidden_dim)

  def forward(self, hidden_state, mask, past_kv = None, use_cache = True):
    bs = hidden_state.shape[0]

    residual = hidden_state

    hidden_state = self.norm(hidden_state) # LLAMA style normalization

    q = self.q_linear(hidden_state) # (bs, seqlen, hidden_dim)
```

```
k = self.k_linear(hidden_state) # (bs, seqlen, hidden_dim)
v = self.v_linear(hidden_state) # (bs, seqlen, hidden_dim)

q = q.view(bs, -1, self.num_heads, self.head_dim).tranpose(1, 2)
k = k.view(bs, -1, self.num_heads, self.head_dim).tranpose(1, 2)
v = v.view(bs, -1, self.num_heads, self.head_dim).tranpose(1, 2)
# (bs, nums_head, seqlen, head_dim)

q, k = apply_rope(q, k)

# kv cache
if past_kv is not None:
  past_k, past_v = past_kv
  k = torch.cat([past_k, k], dim = 2)
  v = torch.cat([past_v, v], dim = 2)
new_past_kv = (k, v) if use_cache else None

# compute attention
attention_scores = torch.matmul(q, k.tranpose(-1, -2)) / math.sqrt(self.head_dim)
attention_scores += mask * -1e9
attention_scores = F.softmax(attention_scores, dim = -1)
output = torch.matmul(attention_scores, v)

# concat
output = output.tranpose(1, 2).contiguous().view(bs, -1, self.hidden_dim)

# o_linear
output = self.o_linear(output)

output += residual

return output, new_past_kv if use_cache else output
```

11 Large Language Models

Question 54 (BERT v.s. GPT)**.** BERT is encoder; GPT is decoder.

- BERT predicts middle words; GPT predicts the next word.
- BERT is **NOT** auto-regressive; GPT is auto-regressive.

Question 55 (GPT – Generative Pre-Training)**.**

$$\min_{\Theta} \hat{\mathbb{E}} - \log \prod_{j=1}^{m} \Pr(x_j | x_1, \ldots, x_{j-1}; \Theta) \tag{95}$$

Question 56 (Fine-Tuning Tasks)**.** Supervised fine-tuning tasks:

$$\min_{\Theta} \underbrace{-\hat{\mathbb{E}} \log \Pr(y | X_{1:m}; \Theta)}_{\text{task-aware supervised loss}} \underbrace{-\lambda \hat{\mathbb{E}} \log \prod_{j=1}^{m} \Pr(x_j | X_{1:j-1}; \Theta)}_{\text{pretraining loss}} \tag{96}$$

Question 57 (BERT → RoBERTa)**.** Training longer, with bigger batches, over more data and longer sequence. Removing the next sentence prediction objective.

Question 58 (Sentence-BERT)**.** a twin network architecture that uses BERT to derive sentence embeddings.

Question 59 (GPT-2)**.** 1.5B parameters.

- 10x larger than GPT-1
- Training method is same as GPT-1.
- Performs on par with BERT on fine-tuning tasks.
- Good at zero-shot learning.
- Open-source.

Question 60 (GPT-3)**.** 175B parameters.

- 100x larger than GPT-2.

- Training method is same as GPT-2.
- New phenomenon: **in-context learning** (ICL, or few-shot learning) and **chain-of-thoughts** (CoT).

Question 61 (GPT-3.5 – RLHF)**.** Reinforcement Learning from Human Feedback (RLHF).

- state = prompt
- action = model output
- policy function = LLM
- reward = levels of matching human feedback

Pari-wise comparison loss to train reward model r_θ:

$$\mathcal{L}_{\text{pair}}(\theta) = -\frac{1}{\binom{K}{2}} \mathbb{E}_{(x,y_w,y_l)} \left[\log(\sigma(r_\theta(x,y_w) - r_\theta(x,y_l)))\right] \tag{97}$$

Proximal Policy Optimization (PPO) to maximize objective:

$$\max_{\Phi} \mathbb{E}_{(x,y)} \left[\underbrace{r_\theta(x,y)}_{\text{maximize reward}} - \beta \underbrace{\log\left(\frac{\pi_\Phi^{\text{RL}}(y|x)}{\pi^{\text{SFT}}(y|x)}\right)}_{\text{model is close to SFT model}} + \gamma \underbrace{\mathbb{E}[\log(\pi_\Phi^{\text{RL}}(x))]}_{\text{pretraining loss}} \right] \tag{98}$$

12 Speculative Sampling

Question 62 (Reject Sampling for Check)**.** Check in parallel.

- $r \sim U(0,1), \quad \text{if } r < \underbrace{\min\left(1, \frac{p(t)}{q(t)}\right)}_{\text{accept rate}}$, next token $= t$.
- else: next token $= t' \sim \underbrace{\text{norm}(\max(0, p-q))}_{\text{residual distribution}}$.

Question 63 (Proof: Reject Sampling $\equiv t \sim p$)**.**

$$\begin{aligned} \min(p(t),q(t)) + \max(0,p(t)-q(t)) &= \begin{cases} p(t) + 0 & \text{if } p(t) < q(t) \\ q(t) + p(t) - q(t) & \text{if } p(t) \geq q(t) \end{cases} \\ &= p(t) \end{aligned} \tag{99}$$

$$\Rightarrow \sum_t (\min(p(t),q(t)) + \max(0,p(t)-q(t))) = \sum_t p(t) = 1 \tag{100}$$

$$\Rightarrow 1 - \sum_t \min(p(t),q(t)) = \sum_t \max(0,p(t)-q(t)) \tag{101}$$

$$\begin{aligned} \Pr(X=t) &= \Pr(\tilde{X}=t)\Pr(\tilde{X} \text{ accept}|\tilde{X}=t) + \Pr(\tilde{X} \text{ reject})\Pr(\tilde{\tilde{X}}=t|\tilde{X} \text{ reject}) \\ &= q(t)\cdot\min\left(1,\frac{p(t)}{q(t)}\right) + (1-\Pr(\tilde{X} \text{ accept}))\cdot\text{norm}(\max(0,p(t)-q(t))) \\ &= \min\left(q(t),p(t)\right) + (1 - \sum_t \min\left(p(t),q(t)\right))\cdot\frac{\max(0,p(t)-q(t))}{\sum_t \max(0,p(t)-q(t))} \\ &= \min\left(q(t),p(t)\right) + \max(0,p(t)-q(t)) \\ &= p(t) \end{aligned} \tag{102}$$

13 Generative Adversarial Networks

Question 64 (Representation through Push-Forward)**.** Let r be any continuous distribution on $\mathbb{R}^h$. For any distribution p on $\mathbb{R}^d$, there exists push-forward maps $G : \mathbb{R}^h \to \mathbb{R}^d$ such that

$$\boldsymbol{z} \sim r \Longrightarrow G(\boldsymbol{z}) \sim p \tag{103}$$

Question 65 (Discriminator's Goal)**.** For a fixed generator G, minimize a log loss over D:

- If x is real, minimize $-\log D(x)$;
- If x is fake, minimize $-\log(1-D(x))$.

$$\min_D -\frac{1}{2}\mathbb{E}_{x\sim p_{\text{data}}}[\log D(x)] - \frac{1}{2}\mathbb{E}_{z\sim r}[\log(1-D(G(z)))] \tag{104}$$

Question 66 (Generator's Goal)**.** For a fixed discriminator D, maximize a log loss over G:

$$\color{red}{\max_G} - \frac{1}{2}\mathbb{E}_{x\sim p_{\text{data}}}[\log D(x)] - \frac{1}{2}\mathbb{E}_{z\sim r}[\log(1-D(G(z)))] \tag{105}$$

Question 67 (Solver)**.**

$$\min_G \max_D V(G,D) = \mathbb{E}_{x\sim p_{\text{data}}}[\log D(x)] + \mathbb{E}_{z\sim r}[\log(1-D(G(z)))] \tag{106}$$

Solved by alternative minimization-maximization:

- G step: Fix D and update G by one-step gradient descent
- D step: Fix G and update D by one-step gradient descent
- Repeat until the algorithm reaches an approximate equilibrium

Question 68 (Solution of D^*)**.** Let $p_g(x)$ be the density of x estimated by the generator G. For G fixed, the optimal discriminator D is $D^*_G(x) = \frac{p_{\text{data}}(x)}{p_{\text{data}}(x)+p_g(x)}$
Proof:

$$\begin{aligned} V(G,D) &:= \mathbb{E}_{x\sim p_{\text{data}}}[\log D(x)] + \mathbb{E}_{z\sim r}[\log(1-D(G(z)))] \\ &= \int \log D(x) p_{\text{data}}(x)dx + \int_z \log(1-D(G(z)))p_z(z)dz \\ &= \int \underbrace{\log D(x)p_{\text{data}}(x) + p_g(x)\log(1-D(x))}_{f(D(x))} dx \end{aligned} \tag{107}$$

For any fixed x, taking derivative $=0$:

$$\begin{aligned} f'(D(x)) &= \frac{p_{\text{data}}(x)}{D(x)} - \frac{p_g(x)}{1-D(x)} = 0 \\ D^*_G(x) &= \frac{p_{\text{data}}(x)}{p_{\text{data}}(x)+p_g(x)} \end{aligned} \tag{108}$$

Question 69 (Solution of G^*)**.** The global minimum of $\min_G \max_D V(G,D)$ is achieved if and only if $p_g = p_{\text{data}}$. The optimal objective value is $-\log 4$.
Proof:

$$\begin{aligned} V(G,D^*_G) &= \mathbb{E}_{x\sim p_{\text{data}}}[\log D^*_G(x)] + \mathbb{E}_{z\sim r}[\log(1-D^*_G(G(z)))] \\ &= \mathbb{E}_{x\sim p_{\text{data}}}[\log D^*_G(x)] + \mathbb{E}_{x\sim p_g}[\log(1-D^*_G(x))] \\ &= \mathbb{E}_{x\sim p_{\text{data}}}\left[\log \frac{p_{\text{data}}(x)}{p_{\text{data}}(x)+p_g(x)}\right] + \mathbb{E}_{x\sim p_g}\left[\frac{p_g(x)}{p_{\text{data}}(x)+p_g(x)}\right] \end{aligned} \tag{109}$$

By definition of KL divergence $\text{KL}(P\|Q) = \mathbb{E}_{x\sim P}\left[\log \frac{p(x)}{q(x)}\right]$, we have:

$$\begin{aligned} V(G,D^*_G) &= \mathbb{E}_{x\sim p_{\text{data}}}\left[\log \frac{p_{\text{data}}(x)}{p_{\text{data}}(x)+p_g(x)}\right] + \mathbb{E}_{x\sim p_g}\left[\log \frac{p_g(x)}{p_{\text{data}}(x)+p_g(x)}\right] \\ &= -\log 4 + \text{KL}\left(p_{\text{data}}\|\frac{p_{\text{data}}+p_g}{2}\right) + \text{KL}\left(p_g\|\frac{p_{\text{data}}+p_g}{2}\right) \\ &= -\log 4 + 2\cdot \text{JSD}(p_{\text{data}}\|p_g) \\ &\geq -\log 4 \end{aligned} \tag{110}$$

The equality holds if and only if $p_{\text{data}} = p_g$.

14 Adversarial Attacks

Question 70 (Principle of Generating Adversarial Attacks).

$$\max_{\|x_{\text{adv}}-x\|_{\infty}\leq\epsilon} \mathcal{L}(C(x_{\text{adv}}), y) \tag{111}$$

where C is the composition of h and f.

Question 71 (Different Solvers). to optimize the adversarial attack.

- **Zero-Order Solvers** (only access to the output of NN)
  - Black-box attack
- **First-Order Solvers** (access to the gradient of NN)
  - White-box attack
  - Fast Gradient Sign Method (FGSM), BIM, PGD, CW attack, ...
- **Second-Order Solvers** (access to the Hessian matrix)
  - White-box attack
  - L-BFGS attack

Question 72 (Holder Inequality). For any $p, q \geq 1$ such that $\frac{1}{p} + \frac{1}{q} = 1$,

$$\|x\|_p \cdot \|y\|_q \geq \langle x, y\rangle \tag{112}$$

where $\langle x, y\rangle$ is the inner product.
$\|\cdot\|_p$ and $\|\cdot\|_q$ are also known as dual norms.

- $\|\cdot\|_2$ is self-dual.
- $\|\cdot\|_\infty$ and $\|\cdot\|_1$ are dual norms.

Question 73 (FGSM – Fast Gradient Sign Method). White-box and non-targeted (maximize the loss w.r.t. the true label).
Do linear expansion at x:

$$\mathcal{L}(C(x+\delta), y) \approx \underbrace{\mathcal{L}(C(x), y)}_{\text{constant}} + \nabla_x \mathcal{L}(C(x), y) \cdot \delta \tag{113}$$

The problem reduces to:

$$\max_{\|\delta\|_\infty \leq \epsilon} \nabla_x \mathcal{L}(C(x), y) \cdot \delta \tag{114}$$

Because of holder inequality (112), we have:

$$\nabla_x \mathcal{L}(C(x), y) \cdot \delta \leq \|\delta\|_\infty \cdot \|\nabla_x \mathcal{L}(C(x), y)\|_1 \leq \epsilon \cdot \|\nabla_x \mathcal{L}(C(x), y)\|_1 \tag{115}$$

Thus, the adversarial example is generated by:

$$x_{\text{adv}} = x + \epsilon \cdot \text{sign}(\nabla_x \mathcal{L}(C(x), y)) \tag{116}$$

where ϵ is the perturbation size.

Question 74 (BIM – Basic Iterative Method). BIM is an iterative version of FGSM.

- Initialize $x^{(0)} = x$.
- For $k = 1, 2, \ldots, K$:

$$x^{(k)} = x^{(k-1)} + \gamma \cdot \text{sign}(\nabla_x \mathcal{L}(C(x^{(k-1)}), y)) \tag{117}$$

Issues:

- By repeating, the pertubation size ϵ will become larger.
- For a pre-defined ϵ, $x^{(k)}$ may not satisfy $\|x^{(k)} - x\|_\infty \leq \epsilon$.

Question 75 (PGD – Projected Gradient Descent). To resolve the issue of BIM, PGD involves a truncation operation:

- Initialize $x^{(0)} = x + \delta$, where $\delta \in (-\epsilon, \epsilon)$.
- For $k = 1, 2, \ldots, K$:

$$x^{(k)} = \text{clip}_{(-\epsilon,\epsilon)}(x^{(k-1)} + \gamma \cdot \text{sign}(\nabla_x \mathcal{L}(C(x^{(k-1)}), y))) \tag{118}$$

where $\text{clip}_{(-\epsilon,\epsilon)}(x)$ projects x back to the ℓ_∞ ball of radius ϵ around x.

Question 76 (Targeted PGD Attack). Objective:

- **Untargeted**:

$$\max_{\|\delta\|_\infty \leq \epsilon} \mathcal{L}(C(x+\delta), y_{\text{true}}) \tag{119}$$

- **Targeted**:

$$\min_{\|\delta\|_\infty \leq \epsilon} \mathcal{L}(C(x+\delta), y_{\text{target}}) \tag{120}$$

15 Adversarial Robustness

Question 77 (Defense Mechanisms). Categorized into two types:

- **Gradient Masking**: hide the gradients and make first-order attacks fail
  - **Shasttered Gradients** By applying a non-smooth or non-differentiable preprocessor g to the inputs, and then training a DNN model f on the preprocessed inputs $g(x)$.
  - **Stochastic/Randomized Gradients** Apply some form of randomization of the DNN model. E.g. train a set of classifiers and during the testing phase randomly select one classifier to predict the labels.
- **Adversarial Training**:

$$\min_C \mathbb{E}_{(x,y)\sim\mathcal{D}^n}\left[\max_{\|\delta\|_\infty \leq \epsilon} \mathcal{L}(C(x+\delta), y)\right] \tag{121}$$

Question 78 (Trade-Off between Natural and Robust Error).

$$\min_f R_{\text{nat}}(f) + R_{\text{rob}}(f)/\lambda \tag{122}$$

$$\begin{aligned} R_{\text{nat}}(f) &:= \Pr_{x,y\sim\mathcal{D}}\{f(x)y \leq 0\} \\ &= \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\mathbb{I}(f(x)y \leq 0)\right] \end{aligned} \tag{123}$$

$$\begin{aligned} R_{\text{rob}}(f) &:= \Pr_{x,y\sim\mathcal{D}}\{\exists \delta \in B_\epsilon(x) \text{ s.t. } f(x+\delta)y \leq 0\} \\ &= \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\max_{\|\delta\|_\infty \leq \epsilon} \mathbb{I}(f(x+\delta)y \leq 0)\right] \end{aligned} \tag{124}$$

Approximate by a differentiable surrogate loss Φ:

$$R_{\text{nat}}(f) \approx \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\Phi(f(x)y)\right] \tag{125}$$

Question 79 (TRADES).

$$\min_f \underbrace{\left[\underbrace{\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\Phi(f(x)y)\right]}_{\text{minimize diff btw } f(x) \text{ and } y \text{ for accuracy}} + \underbrace{\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\max_{\|\delta\|_\infty \leq \epsilon} \Phi(f(x+\delta)f(x))\right]}_{\text{minimize diff btw } f(x) \text{ and } f(x+\delta) for robustness} /\lambda\right]}_{\text{TRADES Loss}} \tag{126}$$

16 Self-Supervised Learning

Question 80 (Contrastive Learning). Loss:

$$\max_\Theta \Pr_1 = \frac{\exp(z_1)}{\sum_j \exp(z_j)} \tag{127}$$