Lecture 14
SVM:

- Linear:
  \[ s = f(x) = \langle x, w \rangle + b \]

- Kernel:
  \[ s = f(x) = \langle h(x), w \rangle + b \]
  \[ s = \sum_{i=1}^{n} \alpha_i y_i K(x_i, x) + b, \quad K(x, x') = \langle h(x), h(x') \rangle \]

Interpolative Memorization & Retrieval!
• Back to linear, non-separable

\[
\min \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{n} \xi_i; \\
(w, b)
\]

subject to \( \gamma_i (\langle x_i, w \rangle + b) \geq 1 - \xi_i; \)

\( \xi_i \geq 0, \ i = 1, \ldots, n \)

• for each \( \xi_i; \) \( \min \xi_i; \)

\[
\text{if } \gamma_i (\langle x_i, w \rangle + b) \geq 1 \Rightarrow \xi_i = 0 \\
\text{s.}
\]

\[
\text{if } \gamma_i (\langle x_i, w \rangle + b) < 1 \Rightarrow \xi_i = 1 - \gamma_i (\langle x_i, w \rangle + b) \\
\]

So \( \xi_i = \max (0, 1 - \gamma_i s_i; ) \)

\[
s_i = \langle x_i, w \rangle + b \\
-\gamma_i s_i + 1 \geq 0 \Rightarrow \xi_i = 1 - \gamma_i (\langle x_i, w \rangle + b) \\
\]

hinge-loss
So we only need to minimize:

\[
\min_{(w, b)} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{n} \max(0, 1 - y_i s_i)
\]

Recall logistic regression

\[
\begin{align*}
\hat{p}(y_i = +1 | s_i) &= \frac{e^{s_i}}{1 + e^{s_i}} = \frac{1}{1 + e^{-s_i}} \\
\hat{p}(y_i = -1 | s_i) &= \frac{1}{1 + e^{s_i}} \\
\]

\[
\hat{p}(y_i | s_i) = \frac{1}{1 + e^{-y_i s_i}}
\]

Log-likelihood:

\[
\sum_{i=1}^{n} \log(\hat{p}(y_i | s_i)) = - \sum_{i=1}^{n} \log(1 + e^{-y_i s_i})
\]

Logistic loss:

\[
\sum_{i=1}^{n} \log(1 + e^{-y_i s_i})
\]
\[
\log (1 + e^{-\gamma_i}) = \begin{cases} 
\gamma_i, & \gamma_i \rightarrow \infty \\
0, & -\gamma_i \rightarrow \infty
\end{cases}
\]

- **Unified**: Loss + Regularization
  
  - Ridge: \( \frac{1}{2} \| w \|_2^2 \)
  - Lasso: \( \| w \|_1 \)

- **Primal-Dual**:

  \[
  \min_{(w, b)} \sum_{i=1}^{n} \ell(y_i; \cdot) + \frac{\lambda}{2} \| w \|_2^2 \\
  \text{or} \\
  \ell(y_i; \cdot) \quad \text{classification}
  \]

  \[
  \min_{(w, b)} \max_{c} \quad \max_{c} \min_{(w, b)}
  \]

  \[
  \text{Representer} \quad \text{Kernel Trick}
  \]
\[ l(s_i) = \max_{c_i} \left[ c_i s_i - l^*(c_i) \right] \]

\[ l^*(c_i) = \max_{s_i} \left( c_i s_i - l(s_i) \right) \]
Part 4: Deep Learning (neural network)

Unified Framework for supervised learning:

Kernel: $K(x, x') = \langle h(x), h(x') \rangle \rightarrow \mathbb{R}$

XGB: $h(x)$ is a tree

tree: $h(x)$ is one-hot
- features are designed.

Neural Network:

$X^T = (x_1, x_2, \ldots, x_p)$

$S = \sum_{k=1}^{d} \beta_k h_{x_k}$

$h_{x_k} = \sigma(s_{x_k})$

$S_{x_k} = \sum_{j=0}^{p} \alpha_{x_k,j} x_j = x^T \alpha_{x_k}$

$\Theta = \left\{ \begin{array}{l} \beta_{x_k}, k = 1 \ldots d \\ \alpha_{x_k,j}, j = 0 \ldots p \end{array} \right\}$

connection weights
\[ s = f(x) = \sum_{k=1}^{d} \beta_k \sigma_k(x - b_k) + \beta_0 \]

If \( \sigma_k \to \infty \)

\[ \alpha_k \leq \infty \]

\[ s = f(x) = \sum_{k=1}^{d} \beta_k \text{ReLU}(x - b_k) + \beta_0 \]
\[ h_k = \text{ReLU} \left( x_k, x_t + \alpha_{k+2} x_{k+2} + \alpha_{k+3} \right) \]

\[ \text{sigmoid} \left( \ldots \right) \]
Random initialization:

\[ s = f(x) = \frac{1}{\sqrt{d}} \sum_{k=1}^{d} \beta_k \sigma \left( x^T \alpha_k \right) \]

\[ \alpha_k \sim N(0, I_d) \]

\[ \beta_k \sim N(0, 1) \quad \beta_k \perp \alpha_k \]

\[ f(x) \xrightarrow{d} \text{GP} \quad \text{(central limit theorem)} \]

Gradient Learning

\[ s = f_\theta(x), \quad \Theta = (\alpha, \beta) \]

\[ \text{Loss} = \frac{1}{2n} \sum_{i=1}^{n} \left( y_i - f_\theta(x_i) \right)^2 \]

\[ \Theta_{t+1} = \Theta_t + \eta_t \frac{1}{n} \sum_{i=1}^{n} \left( y_i - f_\theta(x_i) \right) \frac{\partial f_\theta(x_i)}{\partial \Theta} \]

\[ \Delta \theta \]
\[ f_{\theta + \Delta \theta}(x) = f_{\theta}(x) + \left( \frac{\partial f_{\theta}(x)}{\partial \theta}, \Delta \theta \right) \]

\[ = f_{\theta}(x) + \eta \frac{1}{n} \sum_{i=1}^{n} (y_i - f_{\theta}(x_i)) \left( \frac{\partial f_{\theta}(x_i)}{\partial \theta}, \frac{\partial f_{\theta}(x)}{\partial \theta} \right) \]

Neural Tangent \( K(x_i, x) \) is learned as opposed to designed in classical kernel machines.