Support Vector Machines (SVM)

True non-determinism: you make a prediction (of the future), yet the true outcome is yet to be determined.

Why might predictions be wrong:

① true non-determinism: you can't observe ALL outcomes at your training time, so it's always a "prophecy"

② Noise in the observation

③ representational bias

Rephrase the parameters and representations:

f is changed to h_θ, w is changed to θ

decision boundary: $w^T x = 0 \Rightarrow \theta^T x = 0$

$$\langle \theta, x \rangle = \theta^T x = \sum_i \theta_i x_i = \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_D x_D$$

Decision function $h(x) = \text{sign}(\theta^T x)$

Maximizing the margin allows for more (measurement & otherwise) errors in the representations of samples

Alternative view on logistic regression:

$$g(z) = \frac{1}{1+e^{-z}} = \frac{1}{1+e^{-w^T x}} = \frac{1}{1+e^{-\theta^T x}} = h_\theta(x)$$

Loss: $L_\theta = -y_i \log(g(z)) - (1-y_i)\log(1-g(z))$

$= -y_i \log h_\theta(x_i) - (1-y_i)\log(1-h_\theta(x_i))$

Cost: $\frac{1}{N}\sum_i^N h_{\theta i} = \frac{1}{N}\sum_i^N -y_i \log(h_\theta(x_i)) - (1-y_i)\log(1-h_\theta(x_i))$

$= -\frac{1}{N}\sum_i^N y_i \log(h_\theta(x_i)) + (1-y_i)\log(1-h_\theta(x_i))$

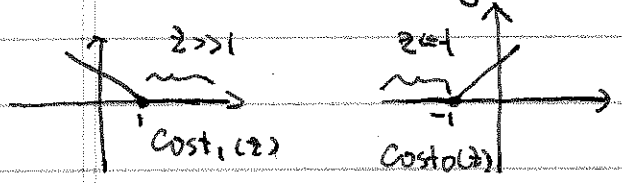
Interpret $\frac{1}{N}$ as a coefficient that is also a constant. the cost can be rephrased by getting rid of $\frac{1}{N}$

$$\text{Cost} = -\sum_i^N \underbrace{y_i \log(h_\theta(x_i))}_{\text{Cost 1}} + \underbrace{(1-y_i)\log(1-h_\theta(x_i))}_{\text{cost 0}}$$

Logistic regression: $\min_\theta -\sum_i^N y_i \log(h_\theta(x_i)) + (1-y_i)\log(1-h_\theta(x_i)) + \frac{\lambda}{2}\sum_j^P \theta_j^2$

Support Vector machine: $\min_\theta C\sum_i^N y_i \text{cost}_1(\theta^T x_i) + (1-y_i)\text{cost}_0(\theta^T x_i) + \frac{1}{2}\sum_j^P \theta_j^2$. where C is the tunable hyper-parameter like λ.

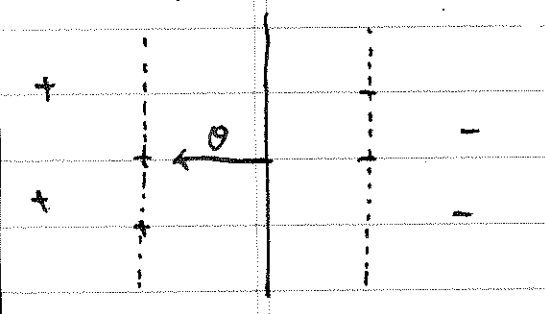
Hinge loss = $\max(0, 1-h(x)\cdot y)$



Note: here, 0 = negative class, not the number 0. (not necessarily)

1 = positive class not necessarily the number 1.

① y	$h_\theta(x)$	$h(x)\cdot y$	$1-h(x)\cdot y$	Hinge
−	−	+	−	0
−	+	−	+	+/pos, >0
+	−	−	+	+/pos, >0
+	+	+	−	0



$\hat{n} = \frac{\theta}{\|\theta\|_2}$, where $\hat{n}$ is the unit vector that represents the direction of θ

$[\theta^T x_+ = 1] - [\theta^T x_- = -1]$

$\Rightarrow \theta^T [x_+ - x_-] = 2$

$\Rightarrow \theta^T \cdot \Delta x = 2, \quad \Delta x = \frac{2}{\theta^T}$

$\hat{n} \cdot \Delta x = \frac{\theta}{\|\theta\|_2} \cdot \frac{2}{\theta^{(T)}} = \frac{2}{\|\theta\|_2}$ = the magnitude of the margin.

For 2 vectors u and v:

$u^T v = v^T u$

$u^T v = u_1 v_1 + u_2 v_2$

$u^T v = \|u\|_2 \|v\|_2 \cos\theta$ $\Rightarrow \theta$ (angle, not bold)

$u^T v = p \|u\|_2$, where $p = \|v\|_2 \cos\theta$

Now substitute u^T with θ^T and v with x (parameter, bold)

$\theta^T x = \|\theta\|_2 \|x\|_2 \cos\theta = p\|\theta\|_2$

SVM objective: $\min_\theta C \sum_{i=1}^{N} [y_i \text{cost}_1(\theta^T x_i) + (1-y_i)\text{cost}_0(\theta^T x_i)] + \frac{1}{2}\sum_{j=1}^{d} \theta_j^2$

Our goal is the maximization of the margin between the positive support vectors and the negative support vectors.

$\Rightarrow$ we want the projections, or p's, to be as large as possible.

$\Rightarrow$ Since we are maximizing the margin, ignore the first term in the objective by setting C to be an arbitrarily small value

$\Rightarrow$ objective becomes $\frac{1}{2}\sum_{j}^{d} \theta_j^2$

Recall the scenario $\hat{n} \cdot \Delta x = \frac{2}{\|\theta\|_2}$, to maximize the margin, we want $\|\theta\|_2$ to be as small as possible.

$\theta^T x = \|\theta\|_2 \|x\|_2 \cos\theta$ when p is small, $\|\theta\|_2$ must be large.

$= p \cdot \|\theta\|_2$ when p is large, $\|\theta\|_2$ must be small

Primal SVM: $\frac{1}{2}\sum_{j=1}^{d} \theta_j^2$, s.t. $y_i(\theta^T x_i + b) \geq 1$, $\forall i$

$y_i(\theta^T x_i + b) - 1 \geq 0$

$\Rightarrow \frac{1}{2}\sum_{j=1}^{d} \theta_j^2 - \sum_{i=1}^{n} \alpha_i \cdot [y_i(\theta^T x_i + b) - 1] \Rightarrow$ to minimize the objective.

minimize θ & maximize α_i

$= \frac{1}{2}\sum_{j=1}^{d} \theta_j^2 - \sum_{i=1}^{n} \alpha_i \cdot y_i(\theta^T x_i + b) + \sum_{i=1}^{N} \alpha_i$

$= \frac{1}{2}\sum_{j=1}^{d} \theta_j^2 - \sum_{i=1}^{n} \alpha_i \cdot y_i \cdot \theta^T x_i - b \cdot \sum_{i=1}^{n} \alpha_i y_i + \sum_{i=1}^{N} \alpha_i$

$\frac{\partial J}{\partial \theta} = \frac{\partial}{\partial \theta} \frac{1}{2}\sum_{j=1}^{d} [\theta_1^2, \theta_2^2, \dots, \theta_j^2, \dots, \theta_d^2] = [\theta_1, \theta_2, \dots, \theta_d] = \theta$

$- \frac{\partial}{\partial \theta}\sum_{i=1}^{n} \alpha_i y_i \theta^T x_i - \frac{\partial}{\partial \theta}\sum_{i=1}^{n} b \cdot \alpha_i y_i + \frac{\partial}{\partial \theta}\sum_{i=1}^{n} \alpha_i$

$\Rightarrow \theta - \sum_{i=1}^{n} \alpha_i y_i x_i = 0 \Rightarrow \theta = \sum_{i=1}^{n} \alpha_i y_i x_i$

$\frac{\partial J}{\partial b} = \frac{\partial}{\partial b}\frac{1}{2}\sum_{j=1}^{d} \theta_j^2 - \frac{\partial}{\partial b}\sum_{i=1}^{n} \alpha_i y_i \cdot \theta^T x_i - \frac{\partial}{\partial b} b \cdot \sum_{i=1}^{n} \alpha_i y_i + \frac{\partial}{\partial b}\sum_{i=1}^{n} \alpha_i$

$= 0 - 0 - \sum_{i=1}^{n} \alpha_i y_i + 0 \Rightarrow -\sum_{i=1}^{n} \alpha_i y_i = 0 = \sum_{i=1}^{n} \alpha_i y_i.$

$$\theta = \sum_i^n a_i y_i x_i \; ; \; \sum_{i=1}^n a_i y_i = 0$$

$$\text{objective} = \frac{1}{2}\sum_{j=1}^{d} \theta_j^2 - \sum_{i=1}^{n} a_i (y_i (\theta^T x_i + b) - 1)$$

$$= \frac{1}{2}\sum_{j=1}^{d} \theta_j^2 - \left[\sum_{i=1}^{n} a_i y_i (\theta^T x_i + b) - \sum_{i=1}^{n} a_i \right]$$

$$= \frac{1}{2}\sum_{j=1}^{d} \theta_j^2 - \left[\sum_{i=1}^{n} a_i y_i \theta^T x_i + \sum_{i=1}^{n} a_i y_i \cdot b - \sum_{i=1}^{n} a_i \right]$$

$$= \frac{1}{2}\sum_{j=1}^{d} \theta_j^2 - \sum_{i=1}^{n} a_i y_i \theta^T x_i - \sum_{i=1}^{n} a_i y_i \cdot b + \sum_{i=1}^{n} a_i$$

$$= \frac{1}{2}\sum_{i}^{n}\sum_{j}^{d} a_i a_j y_i y_j \cdot x_i x_j - \sum_{i=1}^{n} a_i y_i \sum_{j=1}^{d} a_j y_j x_j \cdot x_i - 0 + \sum_{i=1}^{n} a_i$$

$$= -\frac{1}{2}\sum_i \sum_j a_i a_j y_i y_j x_i x_j + \sum a_i$$

⇒ reformulate the problem into $\tilde{J}(a)$ from $J(\theta)$

$$\tilde{J}(\theta) = \sum_{i=1}^{n} a_i - \frac{1}{2}\sum_i \sum_j a_i a_j y_i y_j x_i x_j$$

the decision function is given by:

$$h(x) = \text{sign} \sum_{i \in SV} a_i y_i \langle x_i \cdot x \rangle + b, \text{ whereas } x \text{ is the sample we tried to classify.}$$

$$b = \frac{1}{N}\sum_i^N \left(y_i - \sum_j^N a_j y_j \langle x_i, x_j \rangle\right)$$

$\langle x_i, x_j \rangle$ = Compare the signss classes of 2 support vectors.

$$\Rightarrow b = \frac{1}{|SV|} \sum_{i \in SV} \left(y_i - \sum_{j \in SV} a_j y_j \langle x_i, x_j \rangle \right)$$

α_i is the "coefficient" that indicates if a vector/sample is a support vector.

$\alpha_i \geq 0, \forall i \Rightarrow \alpha_i$ can't be negative but could be 0.

$\sum_i \alpha_i y_i = 0 \Rightarrow$ the positive and negative support vectors balance each other out

$$J(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_i^n \sum_j^n \alpha_i \alpha_j y_i y_j \underbrace{\langle x_i, x_j \rangle}_{\text{Similarity between points}}$$

x_i and x_j have different labels $\Rightarrow \langle x_i, x_j \rangle$ is large $\Rightarrow J(\alpha) \uparrow$
x_i and x_j have same labels $\Rightarrow \langle x_i, x_j \rangle$ is small $\Rightarrow J(\alpha) \downarrow$

$\alpha_i > 0 \Rightarrow x_i$ is a support vector. $\alpha_i = 0 \Rightarrow x_i$ is not a support vector.

When points are not linearly separable $\Rightarrow$ no θ that satisfies $y_i(\theta^T x_i) \geq 1 \; \forall i$
$\Rightarrow$ Introduce the slack variable ξ_i:

$$y_i(\theta^T x_i) \geq 1 - \xi_i, \quad \forall i$$

$$\Rightarrow \min_\theta \frac{1}{2} \sum_j^d \theta_j^2 + C \cdot \sum_i^n \xi_i$$

Kernel tricks:

$$\Phi([x_{i1}, x_{i2}]) = [x_{i1}, x_{i2}, x_{i1} \cdot x_{i2}, x_{i1}^2, x_{i2}^2]$$

and instead of running a SVM on x_i, run it on $\Phi([x_{i1}, x_{i2}]) = \Phi(x_i)$

$K(x_i, x_j) = \langle \Phi(x_i), \Phi(x_j) \rangle$; compute $K(x_i, x_j)$ should be more computationally effective/efficient compared to computing $\Phi(x_i)$ & $\Phi(x_j)$

The polynomial kernel

$$K(x_i, x_j) = \langle x_i, x_j \rangle^2$$
$$= (x_{i1} \cdot x_{j1} + x_{i2} \cdot x_{j2})^2$$
$$= x_{i1} \cdot x_{j1}^2 + 2 x_{i1} \cdot x_{j1} \cdot x_{i2} \cdot x_{j2} + x_{i2} x_{j2}^2$$
$$= \langle \Phi(x_i), \Phi(x_j) \rangle$$

where: $\Phi(x_i) = [x_{i1}^2, x_{i2}^2, \sqrt{2} x_{i1} \cdot x_{i2}]$

$\Phi(x_j) = [x_{j1}^2, x_{j2}^2, \sqrt{2} x_{j1} \cdot x_{j2}]$

Originally, $J(\alpha) = \sum_i^n \alpha_i - \frac{1}{2} \sum_i^n \sum_j^n \alpha_i \cdot \alpha_j \cdot y_i \cdot y_j \cdot \langle x_i \cdot x_j \rangle$

with kernels, $J(\alpha) = \sum_i^n \alpha_i - \frac{1}{2} \sum_i^n \sum_j^n \alpha_i \cdot \alpha_j \cdot y_i \cdot y_j \cdot K(x_i, x_j)$

Gaussian kernel: $K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|_2^2}{2\sigma^2}\right)$

just like $\langle x_i, x_j \rangle$, $K(x_i, x_j)$ measures similarity. the more similar / smaller distance between x_i & x_j, the bigger the value for K is.