Name: John Stephen Gutam
Email : jgutam@gmu.edu

CS 688 : Machine Learning

3/26/2024

Homework 3

(Q1) To construct a support vector machine that computes the XOR function with values of +1 and −1 for both inputs and outputs.

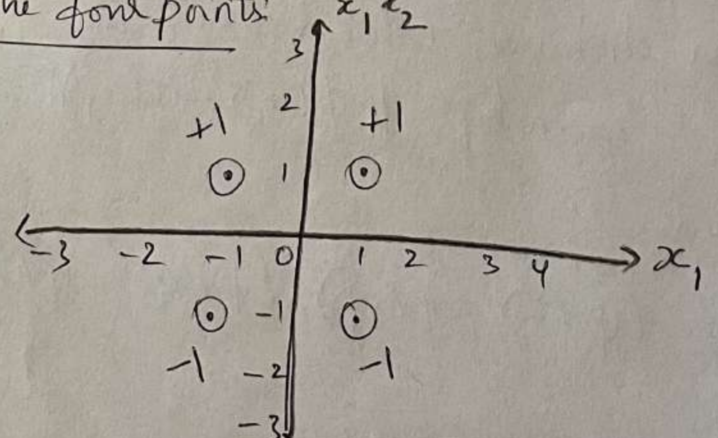
By ~~mapp~~

x_1	x_2	y
1	1	−1
1	−1	1
−1	1	1
−1	−1	−1

Lets map the input $[x_1, x_2]$ into a space consisting of x_1 and $x_1 x_2$.

x_1	$x_1 x_2$	y	Points
1	1	−1	(1, −1)
1	−1	1	(1, 1)
−1	−1	1	(−1, 1)
−1	1	−1	(−1, −1)

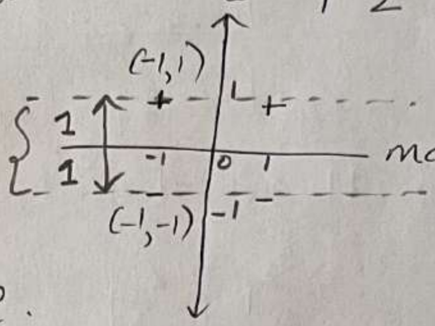
Draw the four points:



Now lets plot those points in the new space, where x-axis represents x1 and y-axis represents x1x2.

So the maximum margin seperator is the line $x_1 x_2 = 0$ (y-axis $y=0$)

The distance between the points $(-1,1)$ and $(-1,-1)$ is



margin ($x_1 x_2 = 0$) is the margin.

$\sqrt{(-1+1)^2 + (1-(-1))^2} = \sqrt{(1+1)^2} = \underline{2}$.

Therefore the margin is $\underline{2}$.

Source: own & some from google + youtube
https://nlp.stanford.edu/IR-book/

(Q2) From the dual function, we desired

$$\max_{\alpha \in R^l} \sum_{i=1}^{l} \alpha_i - \sum_{i,j} \alpha_i \alpha_j y_i y_j (X_i X_j)$$

dot product which is considered for kernel $k(x_i, x_j)$

In the dual formulation of the SVM, features only appear as dot products which can be represented compactly by kernels.

This enables to compute the dot product $\phi(x_i) \cdot \phi(x_j)$ in the higher dimensional space without explicitly calculating the transformed ~~function~~ feature vectors. This is achieved through kernel function $k(x_i, x_j)$, which computes dot product directly.

Given $k(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$ kernelfunction.

The squared Euclidean distance between two points x_i and x_j in the projected space is given by $\|\phi(x_i) - \phi(x_j)\|^2$.

Expanding the expression.

$$\|\phi(x_i) - \phi(x_j)\|^2 = (\phi(x_i) - \phi(x_j)) \cdot (\phi(x_i) - \phi(x_j))$$

$= \phi(x_i)\cdot\phi(x_i) - 2\phi(x_i)\cdot\phi(x_j) + \phi(x_j)\cdot\phi(x_j)$

$= k(x_i, x_i) - 2k(x_i, x_j) + k(x_j, x_j)$

This property is useful in SVMs as it allows us to work with training data in a higher dimensional space without having to explicitly compute the representation $\phi(x)$ of every point x in the original input space. **Source**: Youtube, gogle, chatgpt.

(Q3) From the (Q2) we got the equation

$$\|\phi(x_i) - \phi(x_j)\|^2 = (\phi(x_i) - \phi(x_j))\cdot(\phi(x_i) - \phi(x_j))$$

$$= k(x_i, x_i) - 2k(x_i, x_j) + k(x_j, x_j) \longrightarrow (1)$$

given the kernel function $k(x_i, x_j) = \exp\left(-\frac{1}{2}\|x_i - x_j\|^2\right) \longrightarrow (2)$

From (1) and (2)

$$= \exp\left(-\frac{1}{2}\|x_i - x_i\|^2\right) - 2\left(\exp\left(-\frac{1}{2}\|x_i - x_j\|^2\right)\right) + \exp\left(-\frac{1}{2}\|x_j - x_j\|^2\right)$$

$$= \exp\left(-\frac{1}{2}(0)\right) - 2\left(\exp\left(-\frac{1}{2}\|x_i - x_j\|^2\right)\right) + \exp\left(-\frac{1}{2}(0)\right)$$

$$= 1 - 2\left(\exp\left(-\frac{1}{2}\|x_i - x_j\|^2\right)\right) + 1$$

$$= 2 - 2\left(\exp\left(-\frac{1}{2}\|x_i - x_j\|^2\right)\right)$$

since $0 \leq \exp \leq 1$, we have

$$2 - 2\exp\left(-\frac{1}{2}\|x_i - x_j\|^2\right) \leq 2$$

source: own, google, chatgpt

Therefore, $\|\phi(x_i) - \phi(x_j)\|^2 \leq 2$.

(Q4) Given transform $\phi^n_{poly}(x)$ where $x \in \mathbb{R}^d$

where n (the degree of the polynomial)

d (the dimensionality of original input data).

eg $\begin{bmatrix} 1 \\ x1 \\ x2 \end{bmatrix} \rightarrow \underline{d=3}$

The exact dimensionality of the implied feature space is $\binom{n+d}{d}$

Lets prove this by induction

For $n=1$, $\Rightarrow \binom{n+d}{d} \Rightarrow {}^{1+d}C_d \Rightarrow {}^{1+d}C_1 \Rightarrow \underline{1+d}$.

Lets consider the general case $n>1$. we divide the monomials into two types:

- Those that contain at least one factor $x1$.
- Those that have $i_1 = 0$.

- Type 1 monomials:

there are $\binom{n+d-1}{d-1}$ monomials. This is a one-to-one correspondence between monomials of degree at most d with one factor $x1$ and monomials of degree at most $d-1$ involving all base features.

Type 2 Monomials:

The number of monomials of degree at most d satisfying $i_1 = 0$ is $\binom{n-1+d}{d}$.

Therefore total number of all monomials ~~of~~ is $\binom{n+d-1}{d-1} + \binom{n-1+d}{d}$

$$\Rightarrow \binom{n+d}{d}$$

ex 1

Lets verify this formula with $n = d = 10$:

$$\binom{10+10}{10} = {}^{20}C_{10} = \cancel{20x}\ 1{,}84{,}756 .$$

ex 2

$n = 2,\ d = 2$

$${}^{2+2}C_2 = {}^4C_2 = \frac{4 \times 3}{2} = 6$$

ex 3

For $d = 3,\ n = 2$. A monomial in 3 variables x_1, x_2, x_3 with degree atmost 2 can be written as $x_1^{i_1} \cdot x_2^{i_2} \cdot x_3^{i_3}$

$i_1 + i_2 + i_3 \leq 2$ (degree at most 2)

1. $x_1^0 \cdot x_2^0 \cdot x_3^0 = 1$
2. $x_1^1 \cdot x_2^0 \cdot x_3^0 = x_1$
3. $x_1^0 \cdot x_2^1 \cdot x_3^0 = x_2$
4. $x_1^0 \cdot x_2^0 \cdot x_3^1 = x_3$
5. $x_1^1 \cdot x_2^1 \cdot x_3^0 = x_1 x_2$
6. $x_1^1 \cdot x_2^0 \cdot x_3^1 = x_1 x_3$
7. $x_1^0 \cdot x_2^1 \cdot x_3^1 = x_2 x_3$
8. $x_1^2 \cdot x_2^0 \cdot x_3^0 = x_1^2$
9. $x_1^0 \cdot x_2^2 \cdot x_3^0 = x_2^2$
10. $x_1^0 \cdot x_2^0 \cdot x_3^2 = x_3^2$

$$\binom{n+d}{d}$$

$$= {}^{3+2}C_3 = {}^5C_2$$

$$= \frac{5 \times 4}{2} = \underline{\underline{10}}.$$

So we have 10 values which matches.

Source: Google, own, chatgpt, People.eecs.berkeley.edu

Q5) The loss function of the hypothesis h, is given as

$$L(h) = \mathbb{E}\left[(h(x)-y)^2\right]$$

we can write the loss function over the population as

$$L(h) = \iint P(x,y)\,(h(x)-y)^2\,dx\,dy.$$

Lets do the partial derivative w.r.t 'h' as we need to find and also minimize the derivative by equating to 0.

$$\frac{\partial L(h)}{\partial h} = \iint 2\,P(x,y)\,(h(x)-y)\,dx\,dy = 0$$

$$\iint P(x,y)\,h(x)\,dx\,dy - \iint P(x,y)\,y\,dx.dy = 0$$

$$\iint P(x,y)\,h(x)\,dx\,dy = \iint P(x,y).\,y.\,dx.\,dy$$

$\Rightarrow$ (The above eq. represents the expected value of y given x)

$$h^*(x) = \iint P(x,y)\,h(x)\,dx\,dy \Rightarrow \mathbb{E}[y|x]$$

This shows that the hypothesis $h^*(x) = \mathbb{E}[y|x]$ minimizes the loss function. This means that for any given input x, the optimal prediction is the expected value of the output y given that input.

__Source__: Google, own, chatgpt.

(Q6) The differentiable loss function $L(\omega)$ with L_2-regularized version is given as $L = L(\omega) + \lambda\omega^T\omega$ which involves weights with respect to ω. This can be expressed as in terms of gradient descent as

$$\omega^{(t+1)} = \omega^{(t)} - \eta \nabla_\omega(L)$$

where $L = L(\omega) + \lambda\omega^T\omega$

$$\omega^{(t+1)} = \omega^{(t)} - \eta \nabla_\omega\left(L(\omega) + \lambda\omega^T\omega\right)$$

η = learning rate.

λ = regularization parameter.

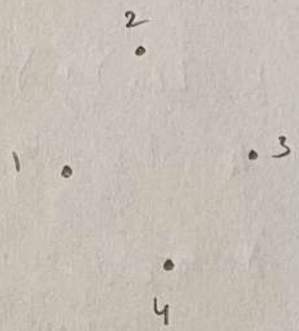
$\nabla_\omega(L)$ = gradient of the Loss function $L(\omega)$ w.r.t ω.

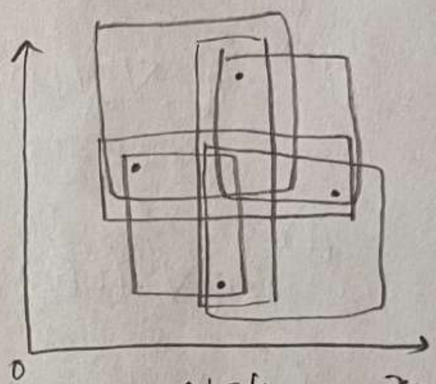
The GD contains extra term $\lambda\omega^T\omega$ in the update rule, which applies a penalty to the weights based on their magnitude. This helps the weights to stay small during training, effectively penalizing large weights to prevent overfitting by discouraging the model from learning overly complex patterns in the training data.

This L2 regularization is called as "weight ~~reduc~~ decay" because it continuously updated towards the direction that minimizes the loss function when the model is trained using L2 regularization the weights are pulled towards zero due to loss function and additional term. The weights are gradually reduced or decayed during training.

In summary, the L2 regularization penalizes large weights, encouraging the model to have smaller weights and preventing overfitting.

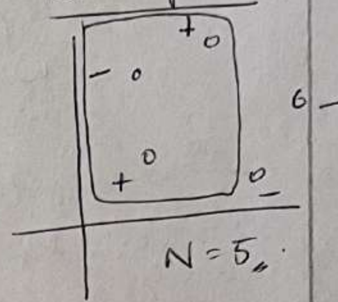
(Q7) (a) The VC-dimension of axis-parallel rectangle in the plane is 4. It is not possible with 5 points. Below plots illustrate the possibility of shattering for 4 points.





N=4

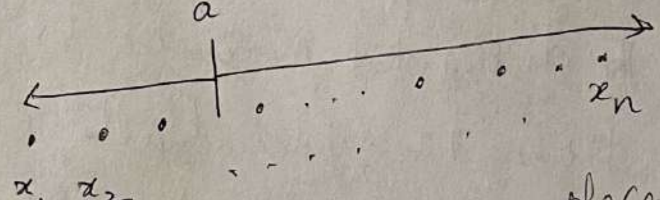
Break point



N=5.

(a) As we see above, we can achieve all possible dichotomies, that can separate them into any combination of +ve and -ve classifications using axis-parallel rectangles. It is not possible with 5 points. Therefore, the VC-dimension of axis-parallel rectangle is 4.

(b) (i) H is the positive ray such that $\{a \mid x \geq a \Rightarrow +1,\ x < a \Rightarrow -1\}$
a is the threshold.



we can place the separator in $n+1$ regions
For n points we,

$$\boxed{m_H(n) = n+1}$$

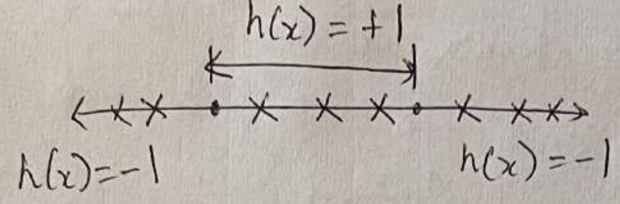
~~Exam~~. If $N=1$, then $m_H(1) = 2$ $= n+1$ | $2^n = 2^1 = 2$

- If $N=2$, then $m_H(2) = 2+1 = 3 \neq$ | $2^2 = 4$

From the definition of VC-dimension $d_{VC}(\mathcal{H})$: largest n for which $m_{\mathcal{H}}(n) = 2^n$.

so $\boxed{d_{VC} = 1}$ for the positive ray.

(ii) For the intervals that can be either positive or negative



So the $m_H(N) = \binom{n+1}{2} + 1 = \frac{n^2}{2} + \frac{n}{2} + 1$

- If $N = 2$, then $m_H(2) = \frac{4}{2} + 1 + 1 = 4 \iff 2^n = 2^2 = 4$.
- If $N = 3$, then $m_H(3) = \frac{9}{2} + \frac{3}{2} + 1 \not\Leftrightarrow 2^3 = 8$
- If $N = 4$, then $M_H(4) = 8 + 2 + 1 = 11 \neq 2^4 = 16$

So the break point is 3. As per the VC dimension definition, $\boxed{d_{VC} = 2}$ for the positive/negative intervals.

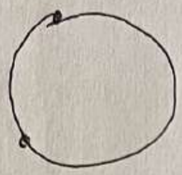
(iii) VC-dimension of the hypothesis space of concentric spheres in R^d

H contains the functions that are +1 for

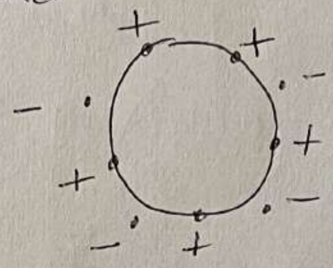
$$a \le \sqrt{x_1^2 + x_2^2 + \cdots x_d^2} \le b.$$

$$a^2 \le x_1^2 + x_2^2 + \cdots x_n^2 \le b^2$$

- If $n = 2$ $a_1^2 \le x_1^2 + x_2^2 \le b^2$ $\longrightarrow$ It is in the form of a circle.



If n points in d dimensions, we can form a circle/ sphere with $+1$ points and -1 points outside the sphere in $m_{\mathcal{H}}(n) = 2^n$.



From the VC dimension definition, if $m_{\mathcal{H}}(n) = 2^n \; \forall n$,

$d_{vc}(\mathcal{H}) = \infty$.