Machine learning

Assignment :- 3

1) (a) $D = \{(x_i, y_i)\}_{i=1}^{n}$ — 2 class classification problem.

$y_i = \{-1, +1\}$

BOOSTING ALGORITHM.

G_t's Error rate $= \varepsilon_t = P_{x \sim w_t}[G_t(x) \neq y] \leq (1/2 - \gamma/2) \; \forall t \; 0 < \gamma < 1.$

$g(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t G_t(x)\right)$ (To show Adaboost satisfies the Inequality)

(ii) $\frac{1}{N} \sum_{i=1}^{N} \mathbb{1}(g(x_i) \neq y_i) \leq \exp\left(-\frac{\gamma^2}{2} T\right).$

This proof as derived in class can be done in 3 steps,

Step 1 :-

$$\frac{1}{N} \sum_{i=1}^{N} \mathbb{1}(g(x_i) \neq y_i) \leq \frac{1}{N} \sum_{i=1}^{N} \exp(-y_i g(x_i)) \quad \text{——— (1)}$$

To prove this statement is true, let us consider $\mathbb{1}(g(x_i) \neq y_i)$, so this function gives a 1 for all wrong predictions and gives '0' otherwise. (like a zero-one loss function) we take an average of these values.

Now, consider the second term, ∵ our classifier wants us to classify into $-1, +1$ classes ∴ $(-y_i g(x_i))$ will return a negative value for all correctly predicted values and it will give a positive value otherwise.

∴ For correctly predicted value,

left hand side term will give 0

(and) Right hand side term will give exp(−ve) value which will exponentially fall to zero (ie) $\exp(-ve) \geq 0$.

(It will go close to zero but will never be zero).

⇒ On the right hand side plane $\exp(-y_i g(x_i)) \geq \mathbb{1}(g(x_i) \neq y_i)$.

For wrong Prediction,

Right hand side $= \exp(+ve)$ $\exp(0) = 1$.

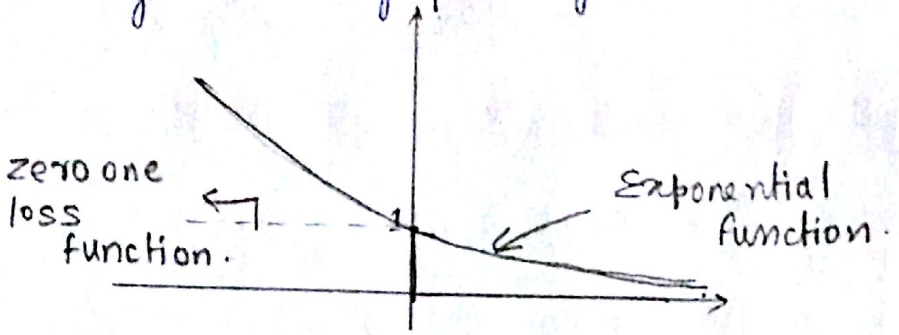
$\exp(+ve) \geq 1$.

left hand side = 1.

⇒ On the left hand side also the function will be above the zero-one loss function.

Thus, the statement is proved.

To verify this result graphically let's plot the functions,



Step (2) :-

$$\frac{1}{N}\sum_{i=1}^{N} \mathbb{1}(g(x_i) \neq y_i) \leq \frac{1}{N}\sum_{i=1}^{N} \exp(-y_i g(x_i)) = \prod_{t=1}^{T} Z_t$$

where Z_t is the Normalization factor,.

Let us take the weight equation,

$$w_{t+1}(i) = \frac{w_t(i)\ \exp(-\alpha_t y_i G_t(x_i))}{Z_t}.$$

Following the same equation we get,

$$w_{t+1}(i) = \frac{w_{t-1}(i) \exp\{\alpha_{t-1} y_i G_{t-1}(x_i)) \cdot \exp(-\alpha_t y_i G_t(x_i))}{Z_t \cdot Z_{t-1}}.$$

Product of exponents add up,

So, if we keep proceeding,

$$w_{t+1}(i) = \frac{w_1(i)\ (\exp)\cdot(-\sum_{t=1}^{T} \alpha_t y_i G_t(x_i))}{\prod_{t=1}^{T} Z_t}.$$

(where T is the number of classifiers).

* Since, we have no proper information at the first run we will all classifier equal weight $\therefore w_1(i) = 1/N$.

$$\Rightarrow w_{t+1}(i) = \frac{1/N\ \exp(-\sum_{t=1}^{T} \alpha_t y_i G_t(x_i))}{\prod Z_t}.$$

Summing for all values of i we get

$$\sum_{i=1}^{N} w_{t+1}(i) = \frac{\sum_{i=1}^{N} \frac{1}{N} \exp\left(-\sum_{t=1}^{T} \alpha_t y_i G_t(x_i)\right)}{\prod_{t=1} z_t}$$

$$\Rightarrow 1 = \frac{\frac{1}{N} \sum_{i=1}^{N} \exp\left(-\sum_{t=1}^{T} \alpha_t y_i G_t(x_i)\right)}{\prod_{t=1} z_t} \quad \text{(Sum of all probabilities (vi) = 1).}$$

$$= \frac{\frac{1}{N} \sum_{i=1}^{N} \exp\left(-y_i g(x_i)\right)}{\prod_{t=1} z_t} \quad \text{(definition).}$$

$$\Rightarrow \prod_{t=1}^{N} z_t = \frac{1}{N} \sum_{i=1}^{N} \exp(-y_i g(x_i))$$

where $z_t = \sum_{i=1}^{N} w_t(i) \exp\{\alpha_t y_i G_t(x_i)\}$

Step 3:-

Each $z_t = 2(\varepsilon_t)^{1/2}(1-\varepsilon_t) = 2\sqrt{\varepsilon_t(1-\varepsilon_t)}$

$$\varepsilon_t = \sum_{i=1}^{N} w_t(i)\, \mathbb{1}(y_i \neq g(x_i)) \quad \text{—— (2)}$$

From Step (2) we can infer that the values / bound of error will be lower for a lower value of z_t. We minimise wrt the only variable in the equation α_t.

$$\Rightarrow \sum_{i=1}^{N} w_t(i) \exp(-\alpha_i y_i G_t(x_i)) = e^{-\alpha_t} \sum_{y_i = G_t(x_i)} w_t(i) + e^{\alpha_t} \sum_{y_i \neq G_t(x_i)} w_t(i)$$

$$\Rightarrow f(\alpha) = (e^{\alpha_t} - e^{-\alpha_t}) \sum_{i=1}^{N} w_t(i)\, \mathbb{1}(y_i \neq G_t(x_i)) + e^{-\alpha_t} \sum_{i=1}^{N} w_t(i)$$

Minimising $f(\alpha)$ over α,

$$(e^{\alpha_t} + e^{-\alpha_t}) \sum_{i=1}^{N} w_t(i)\, \mathbb{1}(y_i \neq G_t(x_i)) - e^{-\alpha_t} \sum_{i=1}^{N} w_t(i) = 0. \quad \text{—— (3)}$$

$$\Rightarrow \frac{e^{-\alpha_t}}{e^{\alpha_t} + e^{-\alpha_t}} = \varepsilon_t \Rightarrow \alpha_t = \frac{1}{2} \log\left(\frac{1-\varepsilon_t}{\varepsilon_t}\right)$$

(from (2) and (3))

$$Z_t = \sum_{i=1}^{N} w_t(i) \exp(-\alpha_t y_i G_t(x_i))$$

$$= \sum_{y = G_t(x_i)} w_t(i) \exp(-\alpha_t) + \sum_{y \neq G_t(x_i)} w_t(i) \exp(\alpha_t)$$

$$= \exp(-\alpha_t) \sum_{i=1}^{N} w_t(i) + (\exp(\alpha_t) - \exp(-\alpha_t)) \sum_{i=1}^{N} w_t(i) \mathbb{1}(y_i \neq G_t(x_i))$$

$$= \exp(-\alpha_t) + (\exp(\alpha_t) - \exp(-\alpha_t)) \varepsilon_t .$$

$$= \varepsilon_t \exp(\alpha_t) + (1-\varepsilon_t) \exp(-\alpha_t)$$

From α_t Value.

$$Z_t = \varepsilon_t \left(\frac{1-\varepsilon_t}{\varepsilon_t}\right)^{1/2} + (1-\varepsilon_t)\left(\frac{\varepsilon_t}{1-\varepsilon_t}\right)^{1/2}.$$

$$\Rightarrow Z_t = 2\sqrt{\varepsilon_t(1-\varepsilon_t)} .$$

It is given that $\varepsilon_t \leq \frac{1}{2} - \frac{\gamma}{2}$.

Bound on training error $\leq \prod_{t=1}^{T} Z_t = 2^T \prod_{t=1}^{T} \sqrt{\varepsilon_t(1-\varepsilon_t)}$.

$$= 2^T \prod_{t=1}^{T} \sqrt{\left(\frac{1}{2}-\frac{\gamma}{2}\right)\left(1-\frac{1}{2}+\frac{\gamma}{2}\right)}$$

$$= 2^T \prod_{t=1}^{T} \sqrt{\frac{(1-\gamma)(1+\gamma)}{(2)^2}} = \frac{2^T}{2^T} \prod_{t=1}^{T} \sqrt{1-\gamma^2}$$

According to Binomial expansion,

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots .$$

From the above we can see that $e^x \leq 1+x$.

$$\Rightarrow \prod_{t=1}^{T} \sqrt{(1-\gamma^2)} \leq \prod_{t=1}^{T} \sqrt{e^{-\gamma^2}}$$

$$\prod_{t=1}^{T} \sqrt{e^{-\gamma^2}} = \prod_{t=1}^{T} e^{-\gamma^2/2}$$

Product of e can be found by raising to the sum of powers.

$$= e^{-\sum_{t=1}^{T} \gamma^2/2} = e^{-T\gamma^2/2} .$$

$$\Rightarrow \frac{1}{N} \sum_{i=1}^{N} \mathbb{1} g(x_i) \neq y_i \leq e^{-T\gamma^2/2} .$$

(b) The bound of the error of combined hypothesis is given as.

Training Error of combined hypothesis $H \leq \exp(-2t\gamma^2)$

where t is the no of iterations.

∴ After some bounded no. of iterations 't' on the k set boosting algorithm the Hypothesis H will be consistent with m training example. (result given in cs.cmu.edu).

(or) From the equations derived in part (a) we can see that the error is bound by a exponentially decaying function and thus as the value of 'T' increases the function will decrease and will take a value very close to zero.

This value can be taken as zero.

∴ As T increases the error will reach zero.

3 (a). $f_{sigmoid}(a_i) = \frac{1}{1+\exp(-a_i)}$ ——— (1)

$h_{sq}^{(sigmoid)}(a) = \sum_{i=1}^{n} (y_i - f_{sigmoid}(a_i))^2$

To prove that the function is convex we have to prove that the Hessian of h_{sq} is positive semidefinite which we can ~~do~~ by as follows,

$a = [a_1, a_2, \ldots a_n]$

Consider,

$f_{sigmoid}(a_i) = \frac{1}{1+\exp(-a_i)}$

$\frac{d}{da_i} f_{sigmoid}(a_i) = \frac{(1-1)}{(1+\exp(-a_i))^2} \times \exp(-a_i)(-1)$

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

$= (f_{sigmoid}(a_i))(1 - f_{sigmoid}(a_i))$ —— (1)

Now let us consider the second derivative.

$\frac{\partial}{\partial a_i}(f_{sigmoid}(a_i))(1 - f_{sigmoid}(a_i))$ this will give some finite value/ an equation since the first derivative got is still a function of a_i. (in terms of a_i)

Now let us consider some other value of a from the vector let's call it a_j,

$\frac{\partial}{\partial a_j}(f_{sigmoid}(a_i))(1 - f_{sigmoid}(a_i)) = 0$. ∵ the function does not depend on a_j.

Now let us consider,

$h_{sq}^{sigmoid}(a) = \sum_{i=1}^{N}(y_i - f_{sigmoid}(a_i))^2$

Let us take the first derivative of this function,

$\frac{\partial}{\partial a_i} h_{sq}(a_i) = 2(y_i - f_{sigmoid}(a_i)) \times (f'_{sigmoid}(a_i))(-1)$ —— (1)

As can be seen clearly the differentiation of (1) wrt a_i will give a value or expression but if we differentiate wrt a_j we get 0.

Let us now consider the definition of Hessian matrix.

$$H = \begin{bmatrix} \frac{\partial}{\partial a_1}\left(\frac{\partial}{\partial a_1} h_{sq}^{sig}(a)\right) & \cdots & \frac{\partial}{\partial a_n}\left(\frac{\partial}{\partial a_1} h_{sq}^{sig}(a)\right) \\ \vdots & & \\ \frac{\partial}{\partial a_n}\left(\frac{\partial}{\partial a_1} h_{sq}^{sig}(a)\right) & \cdots & \frac{\partial}{\partial a_n}\left(\frac{\partial}{\partial a_n} h_{sq}^{sig}(a)\right) \end{bmatrix}$$

From all the results above we can conclude that this matrix is diagonal.

So, now let us compute the diagonal elements,

$\frac{\partial}{\partial a_i} h_{sq}^{sig}(a_i) = \frac{\partial}{\partial a_i}(y_i - f_{sigmoid}(a_i))^2$

$$= 2(y_i - f_{sigmoid}(a_i)) \times \frac{\partial}{\partial a_i}\left(\overset{0}{y_i} + f_{sigmoid}(a_i)\right)(-1)$$

$$\frac{\partial^2}{\partial a_i} h_{sq} = -2(y_i - f_{sig}(a_i)) \cdot \frac{\partial^2}{\partial a_i}(+f_{sigmoid}(a_i)) + (-2)\left[\frac{\partial}{\partial a_i}(f_{sig}(a_i))\right]^2 (-1)$$

$$= -2(y_i - f_{sig}(a_i)) \cdot \frac{\partial^2}{\partial a_i}(f_{sig}(a_i)) + 2\left[\frac{\partial}{\partial a_i}(f_{sig}(a_i))\right]^2 \quad \text{——(3)}$$

Now let's compute $\frac{\partial^2}{\partial a_i}(f_{sig}(a_i))$

$$= \frac{\partial}{\partial a_i}\left[f_{sig}(a_i)(1 - f_{sig}(a_i))\right]$$

$$= f_{sig}(a_i)(-1)\frac{\partial}{\partial a_i} f_{sig}(a_i) + (1 - f_{sig}(a_i))\frac{\partial}{\partial a_i} f_{sig}(a_i)$$

Substituting result ①,

$$= -f_{sig}(a_i)\left[f_{sig}(a_i)(1 - f_{sig}(a_i))\right] + (1 - f_{sig}(a_i))^2 f_{sig}(a_i)$$

$$= (1 - f_{sig}(a_i))^2 f_{sig}(a_i) - (1 - f_{sig}(a_i))(f_{sig}(a_i))^2 \quad \text{——(2)}$$

Substituting (1) and (2) in (3),

$$\frac{\partial^2}{\partial a_i} h_{sq} = (-2)(y_i - f_{sig}(a_i))\left[(1 - f_{sig}(a_i))(f_{sig}(a_i))(1 - 2f_{sig}(a_i))\right] + 2\left[f_{sig}(a_i)(1 - f_{sig}(a_i))\right]^2$$

$$= 2 f_{sig}(a_i)(1 - f_{sig}(a_i))\left[(y_i - f_{sig}(a_i))(2 f_{sig}(a_i) - 1) + f_{sig}(a_i)(1 - f_{sig}(a_i))\right]$$

$$= 2 f_{sig}(a_i)(1 - f_{sig}(a_i))\left[2 y_i f_{sig}(a_i) - y_i + f_{sig}(a_i) - 2 f_{sig}^2(a_i) + f_{sig}(a_i) - f_{sig}(a_i)^2\right]$$

$$= 2 f_{sig}(a_i)(1 - f_{sig}(a_i))\left[2 f_{sig}(a_i)(y_i + 1) - (y_i + 3 f_{sig}^2(a_i))\right]$$

We know that the value of y_i is $\neq 0$ or 1.

So let us assume $y_i = 0$,

$$\frac{\partial^2}{\partial a_i} L_{sq} = 2 f_{sig}(a_i)(1 - f_{sig}(a_i)) \left[2 f_{sig}(a_i) + 3 f_{sig}^2(a_i)\right] \quad \text{---(4)}$$

Let us consider $f_{sig}(a_i)$

$$f_{sig}(a_i) = \frac{1}{1 + \exp(-a_i)}$$

Let us assume that a_i varies from $(-\infty, \infty)$

then $f_{sig}(a_i)$ varies $[1, 0]$. (substitution)

Now equation (4) can be written as,

$$\frac{\partial^2}{\partial a_i} L_{sq} = 2 f_{sig}^2(a_i)(1 - f_{sig}(a_i))(2 - 3 f_{sig}(a_i))$$

At a value close to zero the term is positive and at a value close to 1 it is negative.

Let us assume $y_i = 1$.

$$\frac{\partial^2}{\partial a_i} L_{sq} = 2 f_{sig}(a_i)(1 - f_{sig}(a_i))\left[4 f_{sig}(a_i) - 1 - 3 f_{sig}^2(a_i)\right]$$

$$= 2 f_{sig}(a_i)(1 - f_{sig}(a_i))\left[3 f_{sig}(a_i)(1 - f_{sig}(a_i)) - 1 + f_{sig}(a_i)\right]$$

$$= 2 f_{sig}(a_i)(1 - f_{sig}(a_i))^2 \left[3 f_{sig}(a_i) - 1\right]$$

At value close to zero the term is negative otherwise it is Positive.

∴ The diagonal can be Positive or negative. But all the elements of the diagonal of Hessian matrix must be strictly positive for the matrix to be positive semi definite.

∴ $L_{sq}^{sigmoid}$ is not convex.

(b) $L_{sq}^{(relu)}(a) = \sum_{i=1}^{n} (y_i - f_{relu}(a_i))^2$.

$$f_{relu}(a_i) = \max(0, a_i)$$

$$\Rightarrow f_{relu}(a_i) = \begin{cases} a_i & a_i > 0 \\ 0 & \text{otherwise.} \end{cases}$$

$$\frac{\partial}{\partial a_i} f_{relu}(a_i) = \begin{cases} 1 & a_i \geq 0 \\ 0 & \text{otherwise.} \end{cases} \quad \text{——①}$$

$$\frac{\partial}{\partial a_j}\left(\frac{\partial}{\partial a_i} f_{relu}(a_i)\right) = \{0\} \text{ (from the above expression)}$$

y_i is a constant (do not dependent on a_i ∴ Proceeding like 3(a) we take the Hessian to be a diagonal matrix.

There is a sudden shift when we ~~go from~~ consider the point $a_i = 0$.

At this point using derivative formula,

$$\lim_{h \to 0} \frac{f_{relu}(a+h) - f(h)}{h} = \frac{h-0}{h} = 1. \quad \text{(therefore ① is true).}$$

$$\frac{\partial^2}{\partial a_i} f_{relu}(a_i) = \begin{cases} 0 & a_i \neq 0 \\ 1 & \text{otherwise.} \end{cases}$$

Now lets us consider the hsq function,

$$\frac{\partial}{\partial a_i} h_{sq}^{(relu)}(a_i) = 2\left(y_i - f_{relu}(a_i)\right)(-1) \times \frac{\partial}{\partial a_i}\left(f_{relu}(a_i)\right)$$

$$\frac{\partial^2}{\partial a_i} h_{sq}^{(relu)}(a_i) = (-2)\left(y_i - f_{relu}(a_i)\right)\left[\left(1 - f_{relu}(a_i)\right)\left(f'_{relu}(a_i)\right)\left(1 - 2f_{relu}(a_i)\right) + 2\left[f_{relu}(a_i)\left(1 - f_{relu}(a_i)\right)\right]^2\right].$$

$$\frac{\partial^2}{\partial a_i} h_{sq}^{(relu)}(a_i) = (-2)\left[\left(y_i - f_{relu}(a_i)\right) f''_{relu}(a_i) + f'_{relu}(a_i)\left(-f'_{relu}(a_i)\right)\right]$$

$$= 2\left(f'^2_{relu}(a_i)\right) - 2\left(\left(y_i - f_{relu}(a_i)\right) f''_{relu}(a_i)\right].$$

Now $y_i = 0$,

$$\frac{\partial^2}{\partial a_i} h_{sq}^{(relu)}(a_i) = 2\left(f'_{relu}(a_i)\right)^2 - 2\left(y_i - f_{relu}(a_i)\right) f''_{relu}(a_i)$$

At $a_i = 0$ $\quad \frac{\partial^2}{\partial a_i} h_{sq}^{relu}(a_i) = 0.$

At $a_i > 0$. $\quad \frac{\partial^2}{\partial a_i} h_{sq}^{(relu)} a_i > 0$ (2nd term 0)

At $a_i < 0$ $\quad \frac{\partial^2}{\partial a_i} h_{sq}^{(relu)} a_i = 0.$

For $y_i = 0$,

$\frac{\partial^2}{\partial a_i} h_{sq} \geq 0$

At $y_i = 1$.

At $a > 0$, second term zero $\therefore$ Positive.

At $a < 0$, value $= 0$.

At $a = 0$, $\frac{\partial^2}{\partial a_i} L_{sq}^{(relu)}(a_i) = 2 > 0$.

$\therefore$ All the diagonal elements are always positive.

$\therefore$ The function is convex.

~~4(b)~~. Citation :- All definition are taken from wikipedia.org.

4(a). Pseudo Code :-

Let us assume all the variables as defined in the question.

(i) Convolution (X, K)

→ Get the number of rows and columns in X (set n, m)

→ Get the number of rows and columns in K (set P, q) [in the question it is given as (3, 3)]

→ for $i = 0$ to $(n - P + 1)$

for $j = 0$ to $(m - P + 1)$

result = 0

for $i\_1 = 0$ to P

for $i\_2 = 0$ to q

result = result + X $((i - i\_1), (j - i\_2)) * K(i\_1, i\_2)$

↑ Product.

end

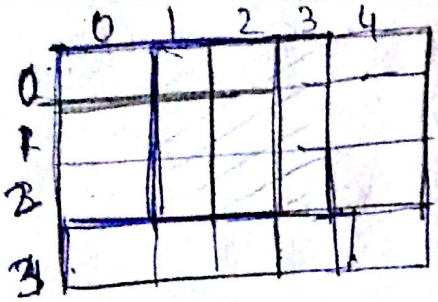
end $Z(i, j) =$ result.

end

end

→ Print Z.

Let us consider a small example to understand why it works, This works if there is no zero padding.



4x5

So using this diagram which is a x matrix we can see that the value j can take are 1 and 0

And 'i' can take 0, 1 and 2 this is the condition checked while entering the loop.

Then to get convolution product we just find element wise product and sum it up this is done by the inner loop.

(b) let us consider consider an example,

$$x = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 & x_5 \\ x_6 & x_7 & x_8 & x_9 & x_{10} \\ x_{11} & x_{12} & x_{13} & x_{14} & x_{15} \\ x_{16} & x_{17} & x_{18} & x_{19} & x_{20} \end{bmatrix}_{4\times5} \qquad K = \begin{bmatrix} k_1 & k_2 & k_3 \\ k_4 & k_5 & k_6 \\ k_7 & k_8 & k_9 \end{bmatrix}.$$

$$Z = X * K.$$

$$Z = \begin{bmatrix} x_1k_1+x_2k_2+x_3k_3+ & x_2k_1+x_3k_2+ & x_3k_1+x_4k_2+x_5k_3 \\ x_6k_4+x_7k_5+x_8k_6+ & \cdots & + \cdots \\ x_{11}k_7+x_{12}k_8+x_{13}k_9 & & \\ \vdots & \cdots & x_8k_1+x_9k_2+x_{10}k_3 \\ & & + \cdots \end{bmatrix}$$

we want express the matrix Z as A·vec(x).

~~Vec(x) is~~ A (n-2)(m-2) x (nm) matrix.

Z is a (n-2)x(m-2) matrix.

Vec(x) = X[1:n, 1] (ie) take all elements in the column of x first.

$$X = \begin{bmatrix} x_1 \\ x_6 \\ x_{11} \\ x_{16} \\ \vdots \\ x_{20} \end{bmatrix} \qquad A \times \begin{bmatrix} x_1 \\ x_6 \\ x_{11} \\ x_{16} \\ \vdots \\ x_{20} \end{bmatrix} = Z.$$

In our case according to definition A must be a 6×20 matrix.
In this case A will be.

$$A = \begin{bmatrix} k_1 & k_4 & k_7 & 0 & k_2 & k_5 & k_8 & 0 & k_3 & k_6 & k_9 & 0 & \cdots & 0 \\ 0 & k_1 & k_4 & k_7 & 0 & k_2 & k_5 & k_8 & 0 & k_3 & k_6 & k_9 & \cdots & \\ 0 & 0 & 0 & 0 & k_1 & k_4 & k_7 & 0 & k_2 & k_5 & k_8 & \cdots & & \\ \vdots & & & & & & & & & & & & & \end{bmatrix}$$

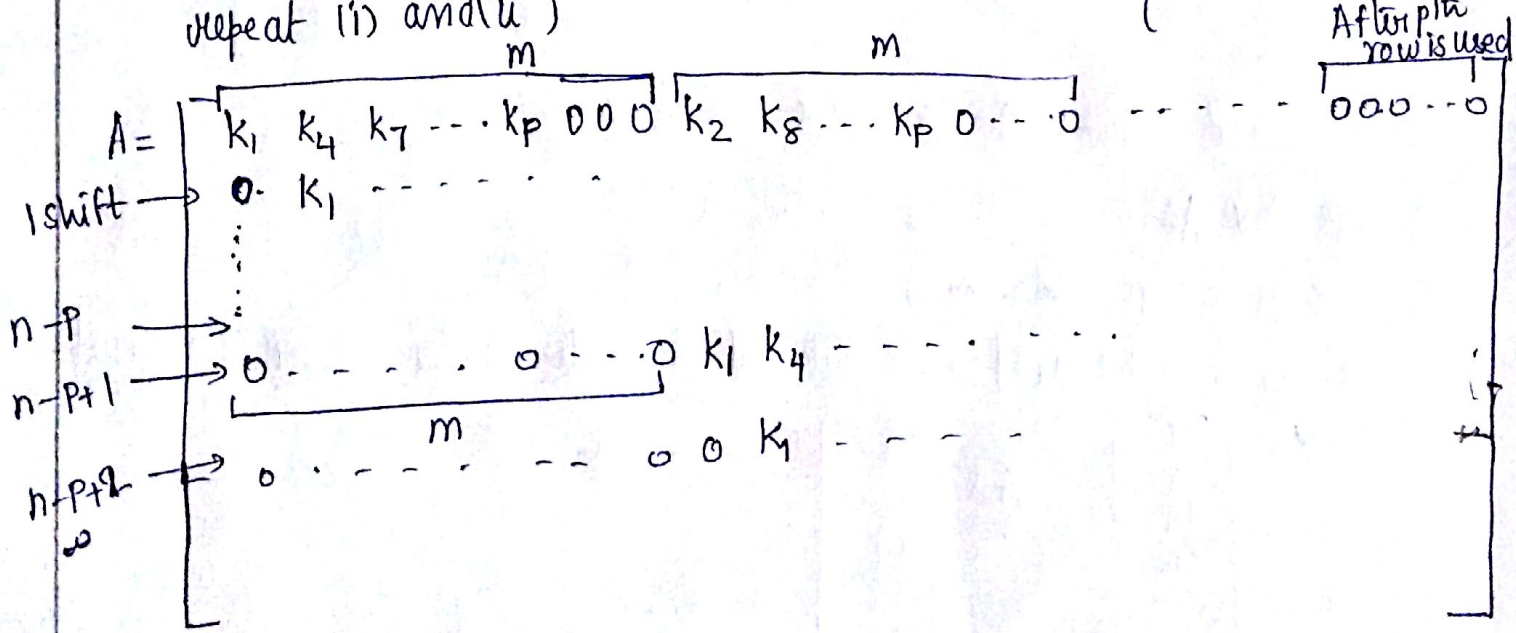
From this let us define the matrix A for a $n \times m$ matrix x and a $P \times P$ matrix K.

we know that $n, m \geq P$. (n: rows, m: column)

(i) At first the element will have P columns of kernel then embed with zero till 'm', then start with 2nd column and embed with zero and do till P and embed rest to 0.

(ii) Till $n-P+1$ keep shifting the column value by 1 this is done by adding one zero at front.

(iii) After that embed the first P columns in the $(n-P+1)^{th}$ row and repeat (i) and (ii)



2(a) Bagging.

Procedure Used:-

* Consider all the n features and based on the value of information gain calculated select the best feature. That is select the feature that gives maximum information gain.

* The split which is given as binary is also done based on the Information gain. (ie) consider all possible splits and take the one that maximises the Information gain ~~of the~~.

* After splitting the maximum value is considered as the decision in the leaf node (ie) if no. of +ve is more than -ve we give the decision output as +ve.

* Then the training and test set errors are calculated.

Information Gain = $H(X) - H(X|Y) = IG(X|Y)$

where $H(Y)$ is entropy of Y and $H(X|Y)$ is entropy of X given Y.

$$H(X) = \sum_{i=1}^{n} -P(x_i)\log_2 P(x_i)$$

$$H(X|Y) = \sum_{j=1}^{m} P(y_j)\, H(X|y_j)$$

$$H(X|Y) = \sum_{i} \frac{P_i + n_i}{P+n}\, H(\langle P_i | P_i+n_i), n_i | (P_i+n_i)\rangle)$$

(b) Random Forest:-

It uses the same procedure as bagging except for the fact that we don't consider all features we just take some number of random features.

When the number of features considered = Total number of features. Then Random forest is bagging.