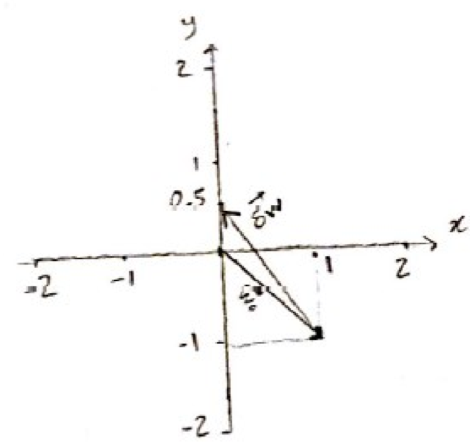
Q1: a) $\vec{w}_{t+1} = \vec{w}_t + y_n \vec{x}_n \quad (\text{if } \operatorname{sign}(\vec{w}^T \vec{x}_n) \neq y_n)$

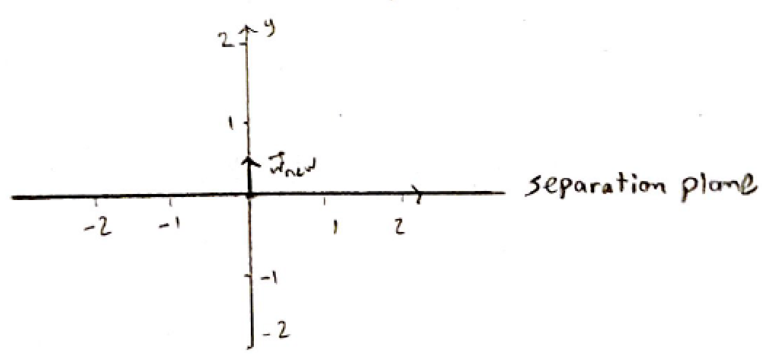
The influence of this update depends on ability of the separable plane to classify $(\vec{x}_n)$ correctly before the update. If it already did a good job, the weights won't update. But if it misclassified that point, the update considers the new point and it will be better at classifying $(\vec{x}_n, y_n)$

$$
\left.\begin{aligned}
@t: &\quad \operatorname{sgn}(\vec{w}^T \vec{x}_n) - y_n \\
@t+1: &\quad \operatorname{sgn}(\vec{w}^T x_n + y_n |\vec{x}_n|^2) - y_n
\end{aligned}\right\} \rightarrow
\left\{\begin{aligned}
&\text{if } y_n = -1,\ w^T x_n \geq 0 \Rightarrow w^T \vec{x}_n + y_n |\vec{x}_n|^2 < w^T \vec{x}_n \\
&\text{if } y_n = +1,\ w^T x_n < 0 \Rightarrow w^T \vec{x}_n + y_n |\vec{x}_n|^2 > w^T \vec{x}_n
\end{aligned}\right\} \text{argument gets better probably}
$$

b) i- $w_n^T \begin{bmatrix} -1 \\ 1.5 \end{bmatrix} = -1 - 1.5 = -2.5 \rightarrow \operatorname{sgn}(-2.5) \neq y = 1 \rightarrow \delta\vec{w} = \eta \times 1 \times \begin{bmatrix} -1 \\ 1.5 \end{bmatrix} \xrightarrow[\eta=1]{\text{assume}} = \begin{bmatrix} -1 \\ 1.5 \end{bmatrix}$



ii- $\vec{w}_{new} = \vec{w} + \delta\vec{w} = \begin{bmatrix} 1 \\ -1 \end{bmatrix} + \begin{bmatrix} -1 \\ 1.5 \end{bmatrix} = \begin{bmatrix} 0 \\ 0.5 \end{bmatrix}$

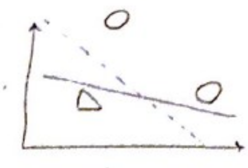


iii - Use $[1, 0.5] \rightarrow w^T x = 0.25 \longrightarrow sgn(w^T x) = 1 = y$ → doesn't update

Use $[-1, -0.5] \rightarrow w^T x = -0.25 \rightarrow sgn(w^T x) = -1 = y$ → doesn't update

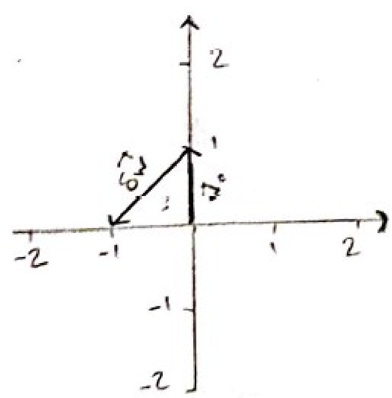
Since the points are linearly separable, the updates will converge to a state that all points would be correctly classified (as in this case).

c) NO. There are an infinite number of lines that can separate these dots. Amongst them, the line that maintains a large threshold is better than others, and can generalize better.



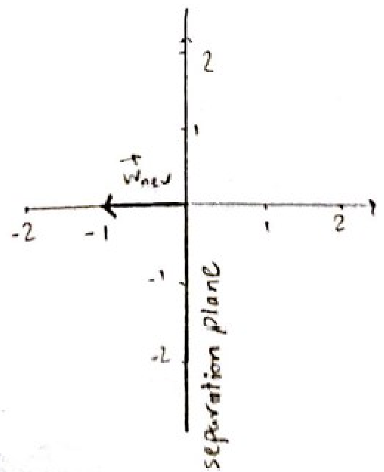
→ The dotted line is better than the solid one.

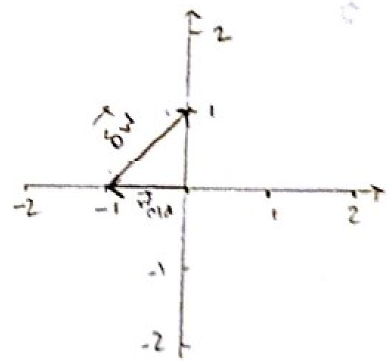
d) i- $sgn(w^T x) = sgn(-1) = -1 \neq y = 1 \Rightarrow \delta \vec{w} = y_n x_n = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$



ii - $\vec{w}_{new} = \begin{bmatrix}0\\1\end{bmatrix} + \begin{bmatrix}-1\\-1\end{bmatrix} = \begin{bmatrix}-1\\0\end{bmatrix}$

Now use $\begin{bmatrix}1\\1\end{bmatrix} \rightarrow sgn(w^T x) = sgn(-1) = -1 \neq y = 1 \rightarrow \delta\vec{w} = yx = \begin{bmatrix}1\\1\end{bmatrix} \rightarrow \vec{w}_{new} = \begin{bmatrix}-1\\0\end{bmatrix} + \begin{bmatrix}1\\1\end{bmatrix} = \begin{bmatrix}0\\1\end{bmatrix}$





iii - NO, for example $sgn(\vec{w}_{new}\begin{bmatrix}-1\\-1\end{bmatrix}) = -1 \neq y = 1$

After two updates, it is same as the initial weights (w_0).

As the data points are not linearly separable, the algorithm won't converge.

Q2 : a) Assume that $\vec{w}^1 = 0$ $\Rightarrow$ $n = 0 \rightarrow \vec{w}^{n+1} \cdot \vec{w}^* = \vec{w}^1 \cdot \vec{w}^* = 0$

$n = K \rightarrow \vec{w}^{K+1} \cdot \vec{w}^* = (\vec{w}^K + y_n \vec{x}_n^T) \cdot \vec{w}^*$

$= \vec{w}^K \cdot \vec{w}^* + y_n \vec{x}_n^T \vec{w}^*$

From linear separability, we know that $y_n \vec{x}_n^T \vec{w}^* \geqslant \gamma$; so $\Rightarrow \vec{w}^{K+1} \cdot \vec{w}^* \geqslant \vec{w}^K \cdot \vec{w}^* + \gamma$

$\Rightarrow \vec{w}^2 \cdot \vec{w}^* \geqslant \gamma$, $\vec{w}^3 \cdot \vec{w}^* \geqslant 2\gamma$, ---, $\boxed{\vec{w}^{K+1} \cdot \vec{w}^* \geqslant K\gamma}$

b) $\vec{w}^{K+1} \cdot \vec{w}^* = \|\vec{w}^{K+1}\| \, \cancel{\|\vec{w}^*\|} \cos\theta \leqslant \|\vec{w}^{K+1}\|$ $\Rightarrow$ $\boxed{\|\vec{w}^{K+1}\| \geqslant K\gamma}$ (I)

c) $\|\vec{w}^{K+1}\|^2 = \|\vec{w}^K + y_n \vec{x}_n\|^2 \leqslant \|\vec{w}^K\|^2 + \|y_n \vec{x}_n\|^2 = \|\vec{w}^K\|^2 + \|\vec{x}_n\|^2$ R^2

$\Rightarrow \|\vec{w}^2\|^2 \leqslant R^2$, $\|\vec{w}^3\|^2 \leqslant R^2 + R^2 = 2R^2$, ---, $\boxed{\|w^{K+1}\|^2 \leqslant KR^2}$ (II)

d) $\xrightarrow{\text{(I), (II)}}$ $(K\gamma)^2 \leqslant \|\vec{w}^{K+1}\|^2 \leqslant KR^2$ $\Rightarrow KR^2 \geqslant K^2\gamma^2 \Rightarrow K \leqslant \frac{R^2}{\gamma^2}$

Q3 : Flipping a coin follows binomial distribution. $\rightarrow \begin{cases} P(heads) = \theta \\ P(tails) = 1-\theta \end{cases} \Rightarrow P(\theta) = \theta^{i} (1-\theta)^{10-i}$

$i \rightarrow$ # heads

MLE : m = #observations $\hat{\theta} = \underset{\theta}{\arg\max}\ P(D) = (1-\theta)^{m-i}\ \theta^{i}$

"Log" is a monotonic function $\Rightarrow \underset{\theta}{\arg\max}\ P(D) = \underset{\theta}{\arg\max}\ \log P(D)$

$\Rightarrow \log P(D) = i \log(\theta) + (m-i)\log(1-\theta)$

$\Rightarrow \frac{d \log P(D)}{d\theta} = \frac{i}{\hat{\theta}} + \frac{-m+i}{1-\hat{\theta}} = 0 \rightarrow i - i\hat{\theta} = m\hat{\theta} - i\hat{\theta}$

$\Rightarrow \boxed{\hat{\theta} = i/m}$

* if observation is the set of ten flips, we can use $m_{new} = 10m$

Q5: $\nabla_w l(w) = \begin{cases} (x^T w - f(x))\,\vec{x} + 2\lambda \vec{w} & |\hat{f}(x) - f(x)| \leq \delta \\ \delta\, sgn(x^T w - f(x))\,\vec{x} + 2\lambda\vec{w} & \text{otherwise} \end{cases}$

$\frac{d \sum w_i^2}{dw} = 2\begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_p \end{bmatrix} = 2\vec{w}$

$w^{k+1} = w^k - \eta \nabla_w l(w) = w^k - \eta \times \begin{cases} (\vec{x}_n^T \vec{w}^k - f(\vec{x}_n))\,\vec{x}_n + 2\lambda \vec{w}^k & |\hat{f}(\vec{x}_n) - f(\vec{x}_n)| \leq \delta \\ \delta\, sgn(\vec{x}_n^T w^k - f(\vec{x}_n))\,\vec{x}_n + 2\lambda \vec{w}^k & \text{otherwise} \end{cases}$

Q6:

a) Done. It is uploaded.

b) 0.8478 = 84.78 %

c) No, it remains as same as before.

d) Although we change the fractions and possibilities, but what is important for us is the probability that is larger. As the number in the denominator is by far larger than the added value, this won't change the predicted label.

Q7: a) $-\frac{1}{3}\log_2(\frac{1}{3}) - \frac{2}{3}\log_2(\frac{2}{3}) - \frac{0}{3}\log_2(\frac{0}{3}) = 0.918$

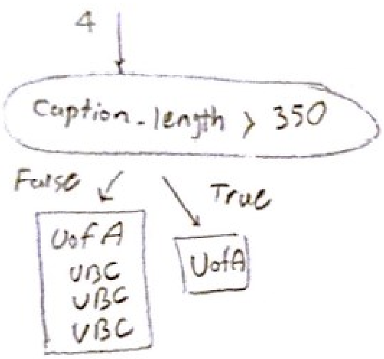
b) comment_count is better.
Because there exist a threshold that can fully separate universities.

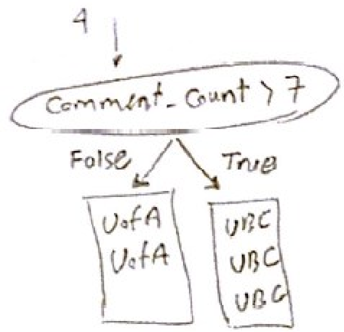
caption_length:

UofA	UBC
163	320
397	256
	222

Comment_count:

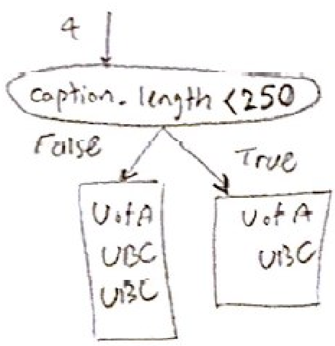
UofA	UBC
2	8
6	18
	16





→ The nodes are pure.

c)



E(false)

$-\frac{1}{3}\log_2(\frac{1}{3}) - \frac{2}{3}\log_2(\frac{2}{3}) = 0.918$

E(True)

$2\times\left(-\frac{1}{2}\log_2(\frac{1}{2})\right) = 1$

$E(\text{node }4) = -\frac{2}{5}\log_2(\frac{2}{5}) - \frac{3}{5}\log_2(\frac{3}{5}) \simeq 0.971$

$\Rightarrow 0.971 - 0.918 = 0.053 \rightarrow$ less than 0.1

$\Rightarrow$ It will be pruned.