Q1 a>

$E[L(y=keep,t)] = 0.9\cdot 0 + 0.1\cdot 1 = 0.1$

$E[L(y=remove,t)] = 0.9\cdot 100 + 0.1\cdot 0 = 90;$

Q1 b> suppose $p(t=spam|x)$ is known; then $p(t=non\text{-}spam|x) = 1 - p(t=spam|x)$

then $E[L(y=keep,t)] = 1\cdot p(t=spam|x) + 0\cdot p(t=nonspam|x)$

$E[L(y=remove,t)] = 0\cdot p(t=spam|x) + 100\cdot p(t=nonspam|x)$

thus $y^* = \begin{cases} keep, & \text{if } E[L(y=keep,t)] < E[L(y=remove,t)] \\ remove, & otherwise \end{cases}$

Q1 C> there are 4 pairs of feature vector: $(X_1, X_2) \Rightarrow (0,0), (0,1), (1,0), (1,1)$

since y^* considers $E[L(y,t)]$ and take the action $y \in \{keep, remove\}$ with minimal loss,

then:

$(0,0)$: first infer $p(t|x)$ from $p(x|t)$

since $p(t|x) = \frac{p(x|t)p(t)}{p(x)}$, will find $p(x)$

$p(x) = \sum_t p(x|t)p(t) = p((0,0)|spam)\cdot p(spam) + p((0,0)|nonspam)\cdot p(nonspam)$

$= 0.4\cdot 0.1 + 0.998\cdot 0.9 = 0.9382$

thus $p(spam|(0,0)) = \frac{0.4\cdot 0.1}{0.9382} = 0.042\,6$

$p(nonspam|(0,0)) = \frac{0.998\cdot 0.9}{0.9382} = 0.957;$

then: $E[L(keep,t)] = 1 \cdot 0.0426$

$E[L(remove,t)] = 100 \cdot 0.957 = 95.7$

thus by y^* from part b, $y = keep$;

(0,1): $p(X) = 0.3 \cdot 0.1 + 0.001 \cdot 0.9 = 0.0309$

$p(spam|(0,1)) = \frac{0.3 \cdot 0.1}{0.0309} = 0.971$ $\quad p(nonspam|(0,1)) = \frac{0.001 \cdot 0.9}{0.0309} = 0.029$

$E[L(keep,t)] = 1 \cdot 0.971$ $\quad E[L(remove,t)] = 100 \cdot 0.029 = 2.9$

thus by y^*, $y = keep$

(1,0) $p(X) = 0.2 \cdot 0.1 + 0.001 \cdot 0.9 = 0.0209$

$p(spam|(1,0)) = \frac{0.02}{0.0209} = 0.957$ $\quad p(nonspam|(1,0)) = \frac{0.0009}{0.0209} = 0.043$

$E[L(keep,t)] = 0.957$ $\quad E[L(remove,t)] = 4.3$

thus by y^*, $y = keep$;

(1,1) $p(X) = 0.1 \cdot 0.1 + 0 \cdot 0.9 = 0.01$

$p(spam|(1,1)) = 1$ $\quad p(nonspam|(1,1)) = 0$

$E[L(keep,t)] = 1$ $\quad E[L(remove,t)] = 0$

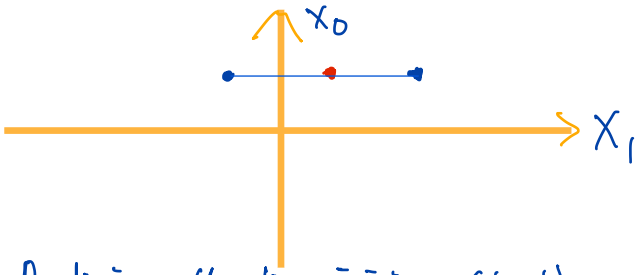
thus by y^*, $y = remove$

Q1 d> $E[L(y^*,t)] = 1 \cdot p((0,0)|spam) \cdot p(spam) + 1 \cdot p((0,1)|spam) \cdot p(spam) + 1 \cdot p((1,0)|spam) \cdot p(spam) + 100 \cdot p((1,1)|nonspam) \cdot p(non\text{-}spam)$

$= 1 \cdot 0.04 + 1 \cdot 0.03 + 1 \cdot 0.02 + 100 \cdot 0 = 0.09$

Q2 a) $(x_0, x_1) \Rightarrow (1,-1)\ (1,1),\ (1,3)$

$t \Rightarrow 1 \quad 0 \quad 1$



Realizing the line joining $(1,-1), (1,3)$ also contains $(1,1)$;
thus the point $(1,1)$ is supposed to lie in a different region but result in the same area as $(1,-1)$ and $(1,3)$.
Thus not linearly separable.

Q2 b) after applying feature vector, will acquire below table:

x	x^2	t
-1	1	1
1	1	0
3	9	1

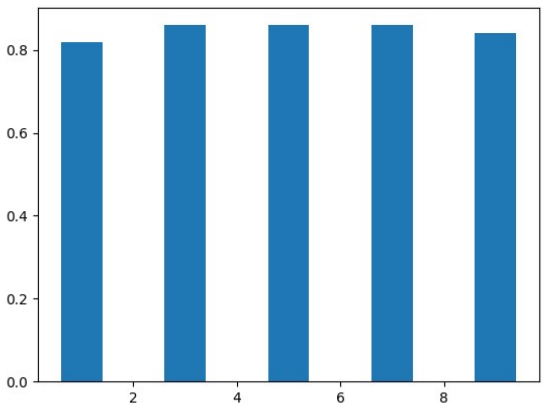
$\Rightarrow$

$$-w_1 + w_2 \geq 0$$
$$w_1 + w_2 < 0$$
$$3w_1 + 9w_2 \geq 0$$

thus one choice of (w_1, w_2) is: $(-1, 0.5)$

Q3 KNN:

classification accuracy



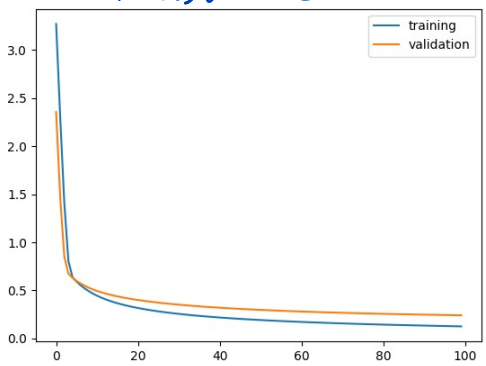
b) KNN has highest accuracy when k=5;
thus k=5 is preferred. The rate of accuracy is around 0.86;
when k=3 and k=7, the accuracy is also 0.86;

after running knn with k=5 on test set, the accuracy is 0.94, which is higher than validation set's performance.

Q3 logistic regression:

C>

mnist 'ce



$\alpha = 0.1$

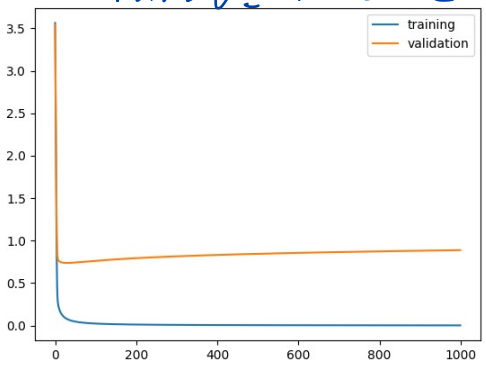
num_iter = 100

final training ce: 0.1260

final validation ce: 0.2411

final test ce: 0.2312

mnist_small ce



$\alpha = 0.1$

num_iter = 1000

final training ce: 0.002

final validation ce: 0.8879

final test ce: 0.8672

several rounds of re-running doesn't result in significant changes to 'ce' tendency and final 'ce'.

Q4 a> $X: N \times D$ $\quad w: D \times 1$

Take derivative with respect to w_p; (since $(a-b)^2 = (b-a)^2$)

$$\frac{\partial J}{\partial w_p} = \frac{\partial}{\partial w_p}\left[\left(\frac{1}{2}\sum_{i=1}^{N} a^i\left(-y^i + \sum_{d=1}^{D} w_d x_d^i\right)^2 + \frac{\lambda}{2}\sum_{d=1}^{D}(w_d)^2\right]\right.$$

$$= \left[\sum_{i=1}^{N} a^i\left(y^i + w_1 x_1^i + w_2 x_2^i \cdots + w_p x_p^i \cdots + w_d x_d^i\right)(x_p^i)\right] + \lambda w_p$$

$$= \underbrace{w_1\left(\sum_{i=1}^{N} a^i x_1^i x_p^i\right) + \cdots + w_p\left(\sum_{i=1}^{N} a^i x_p^i x_p^i\right) \cdots + w_D\left(\sum_{i=1}^{N} a^i x_D^i x_p^i\right)}_{①} + \lambda w_p \underbrace{- \sum_{i=1}^{N} a^i y^i x_p^i}_{③}$$

② —

one observation: $a^i x_p^i = 0 \cdot x_p^1 + 0 \cdot x_p^2 \cdots + a^i \cdot x_p^i + \cdots 0 \cdot x_p^n$

$$= [0 \cdots a^i \cdots 0] \cdot [x_p^1 \cdots x_p^i \cdots x_p^n]^T$$

each $\frac{\partial J}{\partial w_p}$ is a sum of:

① product of: pth row of X^T matrix, ith row of diagonal matrix A, matrix X, and w^T vector.

② : pth row of λI multiply with w^T

③ ith row of diagonal matrix A with vector y and pth column of matrix X;

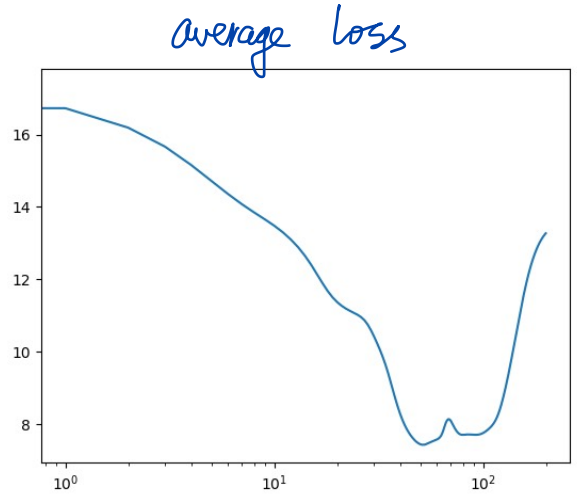
Thus $\frac{\partial J}{\partial w} = \begin{bmatrix} \vdots \\ \frac{\partial J}{w_p} \\ \vdots \end{bmatrix} = -((A y)^T X)^T + X^T(AX)\cdot w + \lambda I w = 0$

$\Rightarrow$ since A is diagonal, then $(Ay)^T = y^T A^T = y^T A$

$$(X^T A X + \lambda I) w = (y^T A X)^T = X^T A y$$

thus $w = (X^T A X + \lambda I)^{-1} X^T A y$

Q4 c>



d> as $T \to \infty$, the loss of this algorithm would be higher than 7, the minimal loss ; same for the case when $T \to 0$.

by running simulation with $T = 2, 3, 1000000, 10^8$,

the result is: when $T > 10^5$, the loss stablizes at 13.5 above

when T is 2, 3, the loss is 25 above, which are both higher than 7, the minimal loss could achieve.