PS 2.

1. (a) $P(y;\lambda) = \frac{e^{-\lambda}\lambda^y}{y!}$

$= \frac{1}{y!} \cdot \frac{\lambda^y}{e^{\lambda}}$

For exponential form: $P(y;\lambda) = b(y)e^{\eta^T \cdot T(y) - a(\eta)}$

$= b(y) \cdot \frac{e^{\eta^T T(y)}}{e^{a(\eta)}}$

$\therefore$ $b(y) = \frac{1}{y!}$, $a(\eta) = \lambda = e^{\eta}$

$\lambda^y = e^{\eta^T T(y)}$

$\ln \lambda^y = \eta^T T(y)$

$\eta^T T(y) = y \cdot \ln\lambda$

~~$T(y) = \ln\lambda \cdot (\eta^T)^{-1} y$~~.

$T(y) = y$

$\eta = (\ln\lambda)$

$a(\eta) = \lambda \Rightarrow a(\eta) = e^{\eta}$

(b) canonical response function

$g(\eta) = E[T(y);\eta]$

$E[y] = \lambda$ $\eta = \ln\lambda$

$g(\eta) = \ln$.

1. (C) ~~$\log p(y^{(i)}|x^{(i)};\theta)$~~

$$\ell(\theta) = \log p(\vec{y}|X;\theta)$$

$$= \log \prod_{i=1}^{m} p(y^{(i)}|x^{(i)};\theta)$$

$$= \log \prod_{i=1}^{m} \frac{1}{y^{i}!} \cdot e^{\ln\lambda \cdot y^{(i)} - e^{\eta}}$$

$$= \sum_{i=1}^{m} \log\left(\frac{1}{y^{i}!} \cdot e^{\ln\lambda \cdot y^{(i)} - e^{\eta}}\right)$$

$$= \sum_{i=1}^{m} \log \frac{1}{y^{(i)}!} + \eta \cdot y^{(i)} - e^{\eta}$$

$$= \sum_{i=1}^{m} \log \frac{1}{y^{(i)}!} + \theta^{T}(x)^{i} \cdot y^{(i)} - e^{\theta^{T}x^{i}}$$

$$\frac{\partial \ell(\theta)}{\partial \theta_j} = \sum_{i=1}^{m} (x_j^{(i)} \cdot y^{(i)} - e^{\theta^{T}x^{i}} \cdot x_j^{i})$$

$$= \sum_{i=1}^{m} (y^{(i)} - e^{\theta^{T}x^{i}}) \cdot x_j^{(i)}$$

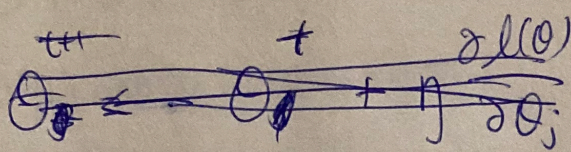
$\eta = \ln\lambda = \theta^{T}x \Rightarrow \lambda = e^{\theta^{T}x^{i}}$

$h(x) = E[T(y)|x] = \lambda = e^{\theta^{T}x^{i}} = g^{-1}$

$$\therefore \frac{\partial \ell(\theta)}{\partial \theta_j} = \sum_{i=1}^{m} (y^{(i)} - h_\theta(x^{i}))\, x_j^{i}$$

~~1.(d) plot~~

1.(c)



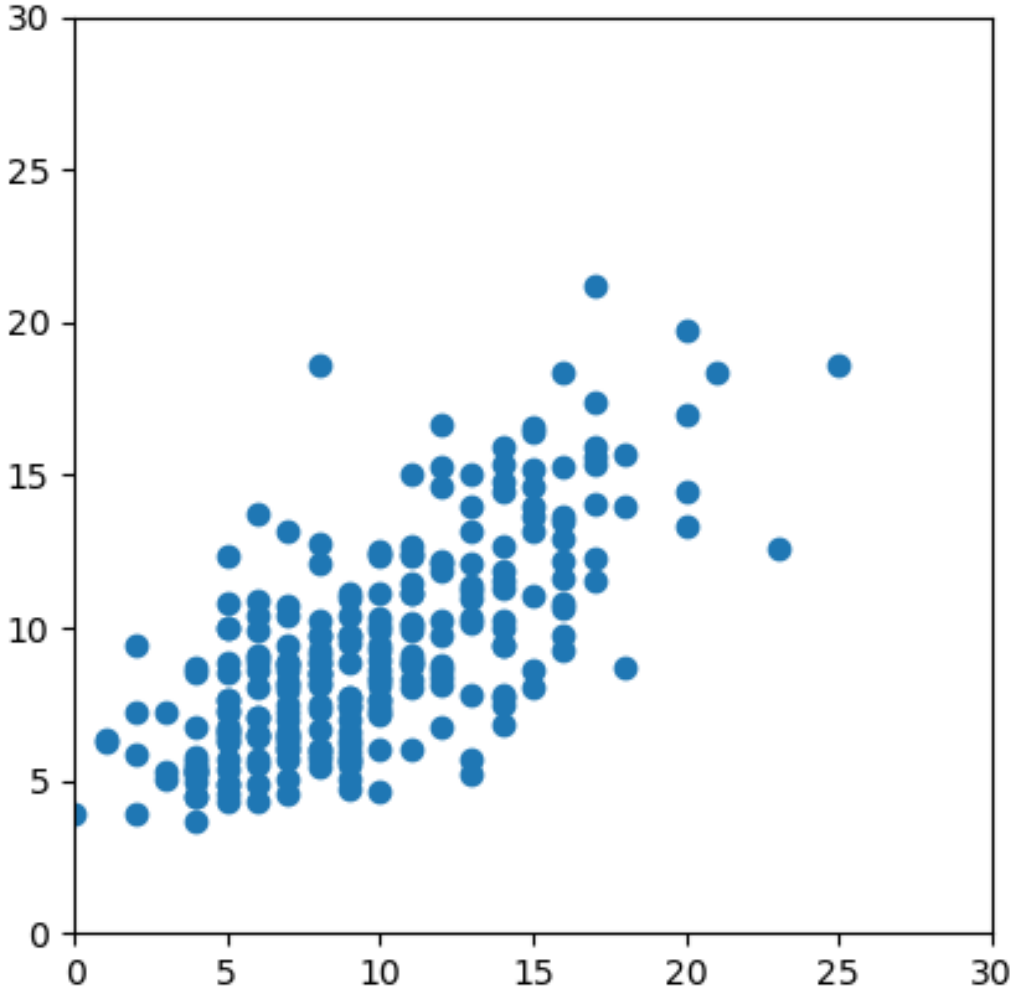
$$\theta^{t+1} \leftarrow \theta^t + \eta \frac{\partial l(\theta)}{\partial \theta_j}$$

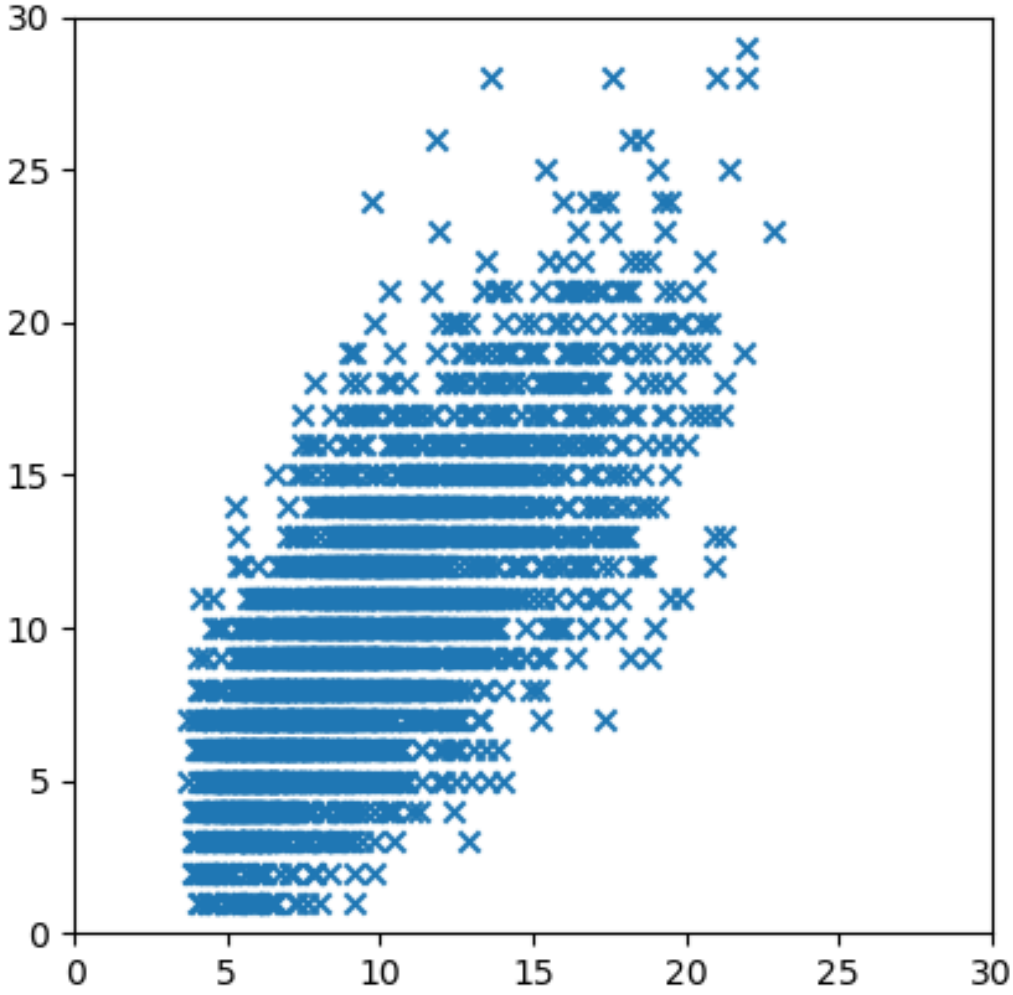
$$= \theta^t + \eta \cdot \sum_{i=1}^{m} (y^{(i)} - g^{-1}(x^{(i)})) \cdot x_j^{(i)}$$

1.(d) plot.

predict plot

validation plot.


30
25
20
15
10
5
0
0
5
10
15
20
25
30


30
25
20
15
10
5
0
0
5
10
15
20
25
30

2. (a)

$\int p(y;\eta)\,dy = 1$ because the sum of all probability is 1

$\int b(y)\exp(\eta y - a(\eta))\,dy = 1$

$\int \frac{b(y)\exp(\eta y)}{\exp(a(\eta))}\,dy = 1$ ∵ $a(\eta)$ doesn't have y.

$\exp(a(\eta)) = \int b(y)\exp(\eta y)\,dy$

$a(\eta) = \log \int b(y)\exp(\eta y)\,dy$

$\frac{\partial a(\eta)}{\partial \eta} = \frac{\partial}{\partial \eta}\log \int b(y)\exp(\eta y)\,dy$

$= \frac{1}{\int b(y)\exp(\eta y)\,dy} \cdot \frac{\partial}{\partial \eta}\int b(y)\exp(\eta y)\,dy$

$= \frac{1}{\exp(a(\eta))} \cdot \int \frac{\partial}{\partial \eta} b(y)\exp(\eta y)\,dy$

$= \frac{1}{\exp(a(\eta))} \cdot \int b(y)\cdot\exp(\eta y)\cdot y\,dy$

$= \int \frac{b(y)\exp(\eta y)}{\exp(a(\eta))} \cdot y\,dy$

$= \int p(y;\eta)\,y\,dy$

$= E(Y;\eta)$

2. (b)

$$\frac{\partial^2}{\partial \eta^2} a(\eta) = \frac{\partial}{\partial \eta}\left(\frac{\partial a(\eta)}{\partial \eta}\right)$$

$$= \frac{\partial}{\partial \eta} \int \frac{b(y)\exp(\eta y)}{\exp(a(\eta))} \cdot y \, \partial y$$

$$= \int b(y) y \cdot \frac{\partial \exp(\eta y - a(\eta))}{\partial \eta} \partial y$$

$$= \int b(y) y \cdot \exp(\eta y - a(\eta)) \cdot \left[y - \frac{\partial a(\eta)}{\partial \eta}\right] \partial y$$

$$= \int b(y) \exp(\eta y - a(\eta)) y^2 \, \partial y$$

$$- \int b(y) \exp(\eta y - a(\eta)) y \frac{\partial a(\eta)}{\partial \eta} \partial y$$

$$= E[y^2] - \frac{\partial a(\eta)}{\partial \eta} \cdot \int b(y) \exp(\eta y - a(\eta)) y \, dy$$

$$= E[y^2] - E[y] \cdot E(y)$$

$$= E[y^2] - E[y]^2$$

$$= \mathrm{Var}(Y; \eta)$$

2.(C) for a single example

$$NLL = \ell(\theta) = -\log(L(\theta)) = -\log(p(y_i|x_i;\theta))$$

$$= -\log(b(y)\exp(\eta y - a(\eta)))$$

$$= -[\log(b(y)) + \eta y - a(\eta)]$$

$$= a(\eta) - \eta y - \log(b(y))$$

$$\text{Hessian} = \begin{bmatrix} \cdots & \cdots & \cdots \\ \vdots & \frac{\partial^2 \ell(\theta)}{\partial\theta_i \partial\theta_j} & \vdots \\ \cdots & \cdots & \cdots \end{bmatrix}$$

for a given i, j.

$$\frac{\partial^2 \ell(\theta)}{\partial\theta_i \partial\theta_j} = \frac{\partial}{\partial\theta_i}\left(\frac{\partial \ell(\theta)}{\partial\theta_j}\right) = \frac{\partial}{\partial\theta_i}\left(\frac{\partial a(\eta)}{\partial\theta_j} - \frac{\partial \eta y}{\partial\theta_j} - \frac{\partial \log b(y)}{\partial\theta_j}\right)$$

$\frac{\partial \log b(y)}{\partial\theta_j} \to 0$

$$= \frac{\partial}{\partial\theta_i}\left(x_j \frac{\partial a(\eta)}{\partial\eta} - x_j y\right) \qquad \boxed{\eta = \theta^T x, \quad \frac{\partial\eta}{\partial\theta_j} = x_j}$$

$$= \frac{\partial}{\partial\theta_i}\left(x_j \frac{\partial a(\eta)}{\partial\eta}\right) - \underbrace{\frac{\partial}{\partial\theta_i} x_i y}_{\to 0}$$

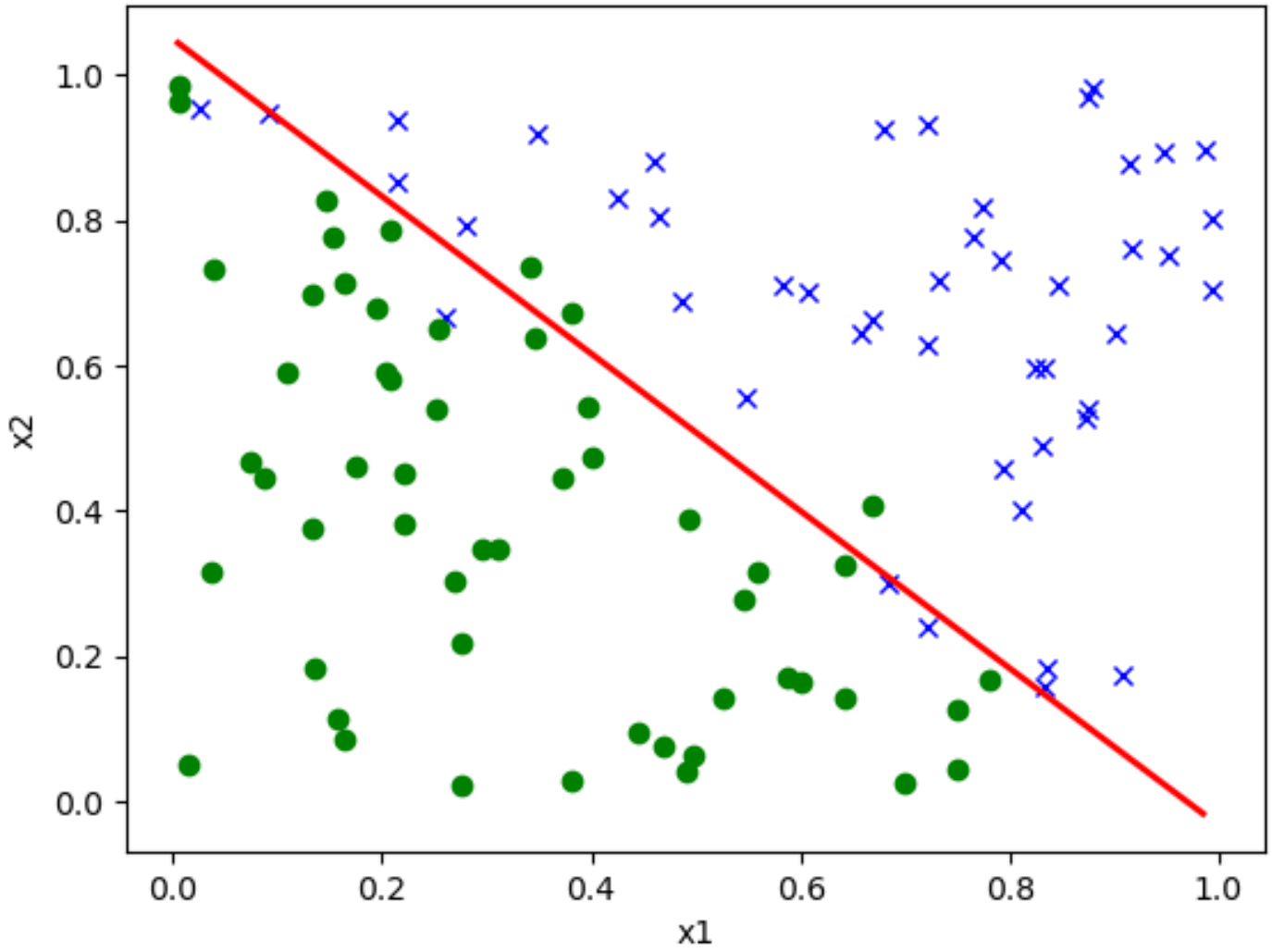
$$= x_i x_j \cdot \frac{\partial^2 a(\eta)}{\partial\eta^2}$$

$\frac{\partial^2 a(\eta)}{\partial\eta^2} \to$ constant

$$\text{Hessian} = \frac{\partial^2 a(\eta)}{\partial\eta^2} \cdot \begin{bmatrix} x_1^2 & x_1x_2 & \cdots & x_1x_d \\ x_2x_1 & x_2^2 & \cdots & x_2x_d \\ \vdots & & & \\ x_dx_1 & \cdots & \cdots & x_d^2 \end{bmatrix} = \frac{\partial^2 a(\eta)}{\partial\eta^2} \cdot \vec{X}^T \cdot \vec{X}$$

where $\vec{X} = [x_1, x_2 \cdots x_d]$

Hessian is by definition PSD $= A^T \cdot A$ where A has independant col


1.0
0.8
0.6
0.4
0.2
0.0
x2
0.0
0.2
0.4
0.6
0.8
1.0
x1

3. (a) plot

(b) training set B ~~didn~~ causes the traing iteration maxed out 10^5 threshold: it didn't converge.

it's coefficient keeps increasing its magnitude.

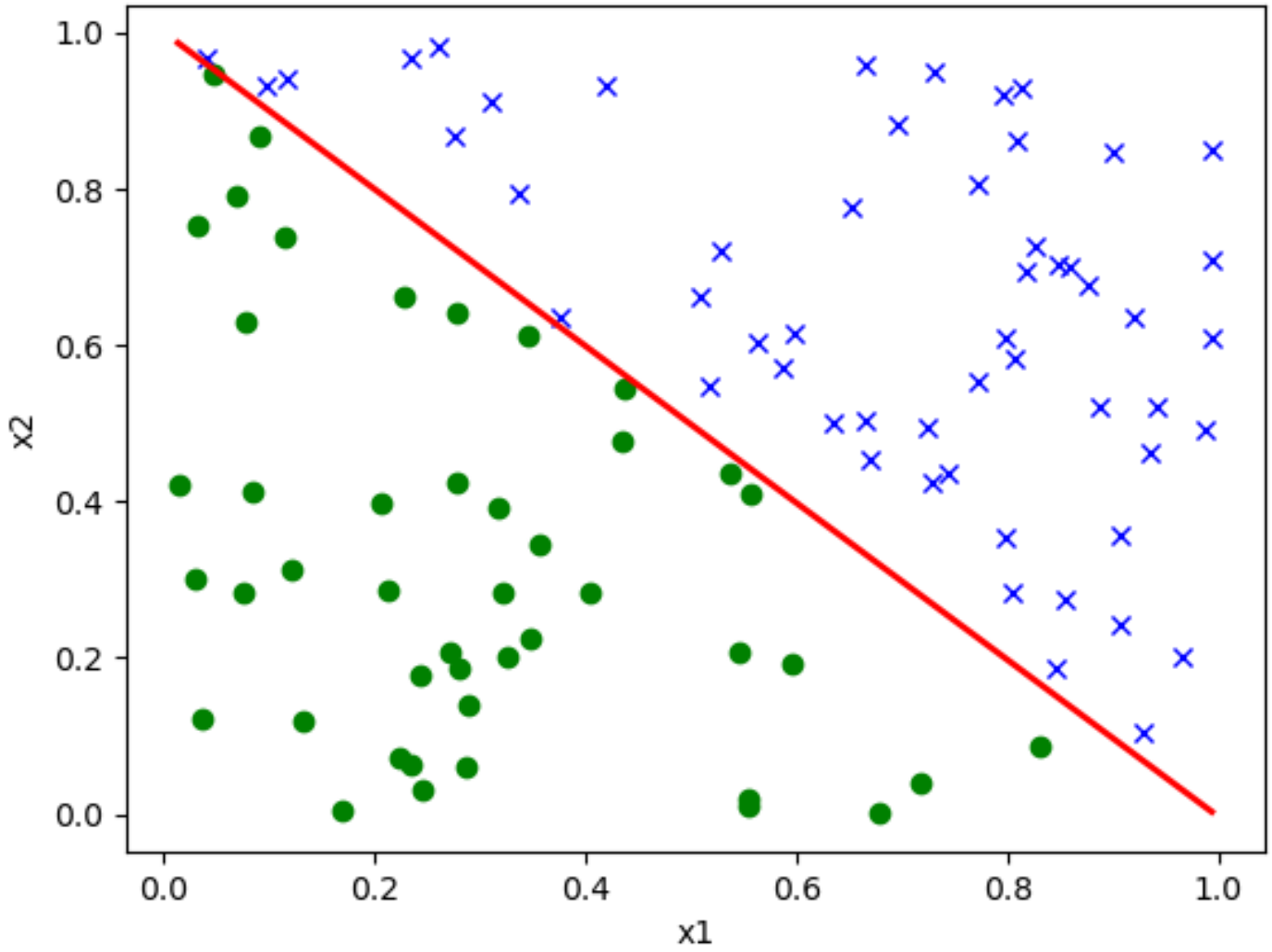
because B is a linearly separable ~~trai~~ data set.

it exists a $\hat{w}$ such that
for all positive training data
$$\hat{w}^T h(x) > 0$$
for all negative traing data, $\hat{w}^T h(x) < 0$.

so, $\alpha\hat{w}$, where $\alpha \in \mathbb{R}$ is also a decision boundry

because we maximize the likelihood estimation, we keeps inrease the $\hat{w}$ (overfitting), thus it keeps increasing.

A doesn't have this problem because it's not linearly separable in the ~~tt~~ current dimension (x_1, x_2)


1.0
0.8
0.6
0.4
0.2
0.0
x2
0.0
0.2
0.4
0.6
0.8
1.0
x1

3(c).

(i) no. it's linearly separable data. changing learning rate will not stop the coefficient from increasing.

(ii) no. same as above

(iii) no. linear scaling of linear separable data is still linear separable.

(iv) yes, L_2 regulation penalizes large coefficient so it will stop increasing

(v). yes. adding 0-mean Gaussian noise to data can change it to linear unseparable data.

3(d) without Regularization:

data set A: theta = [-20.7796 21.4171 2.3825]

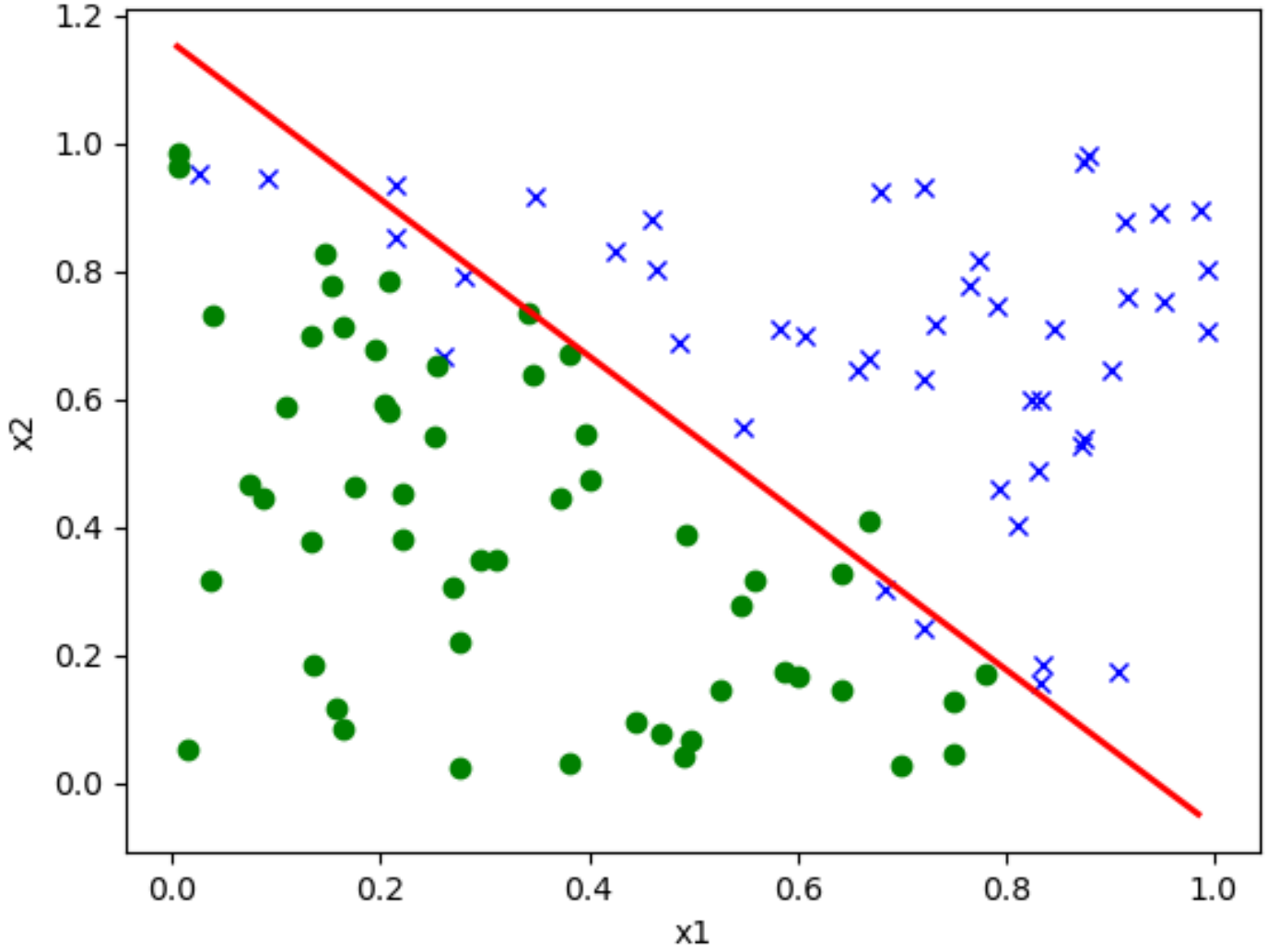
B: [-52.7200 52.9088 52.6758]

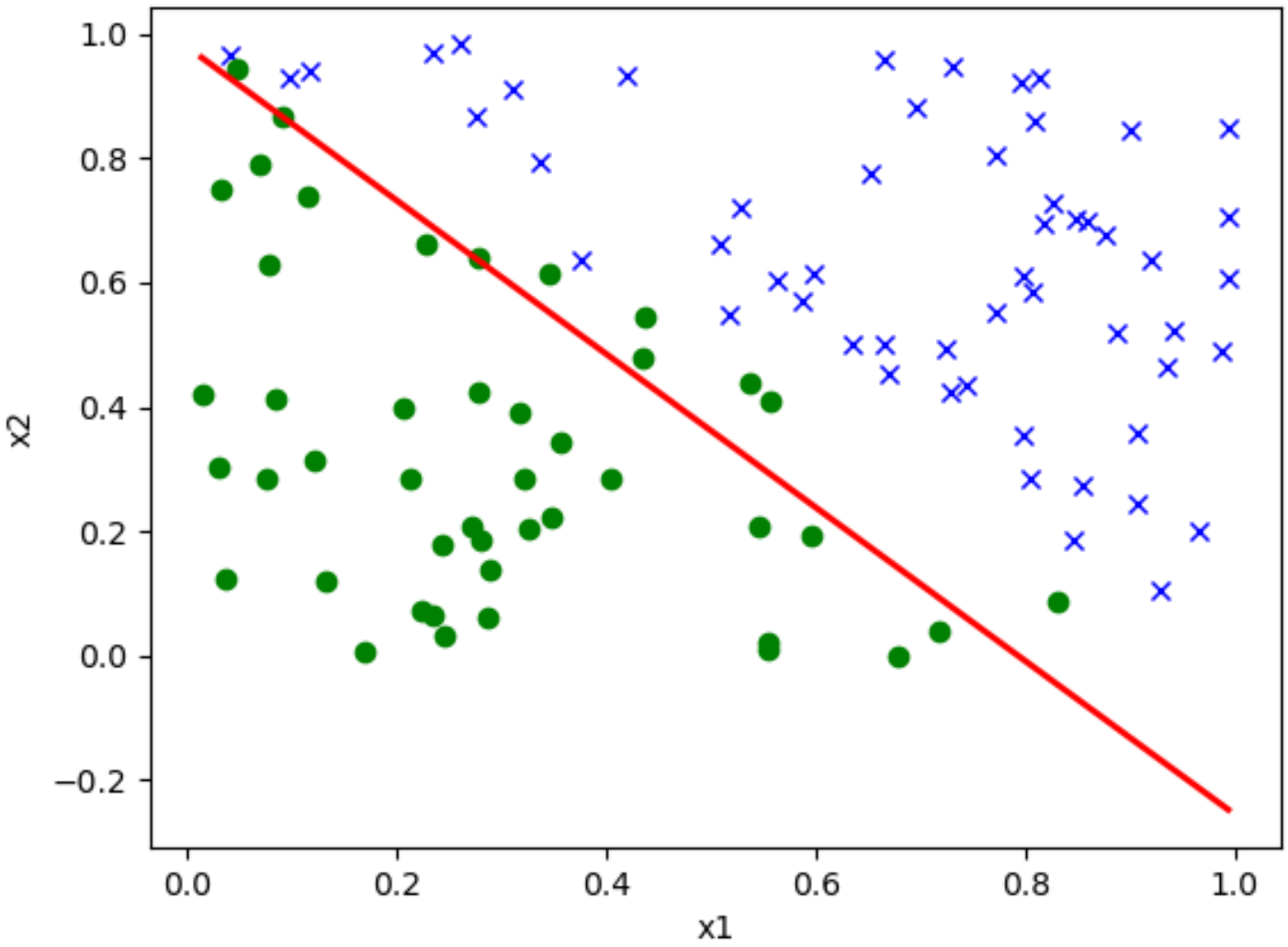
With Regularization.

A: [-2.5402 2.6892 2.1954]

B: [-2.3337 2.9438 2.3825]

plot


1.2
1.0
0.8
0.6
0.4
0.2
0.0
x2
0.0
0.2
0.4
0.6
0.8
1.0
x1


1.0
0.8
0.6
0.4
0.2
0.0
−0.2
x2
0.0
0.2
0.4
0.6
0.8
1.0
x1

3.(d) (continue)

for A, regularization didn't significantly impact the decision boundary. However, for B, regularization moved the decision boundary to ~~include~~ include some positive examples in negative prediction (i.e. false negative predictions).

Without regulation, the ~~decision~~ prediction will be overly confident. ~~and~~ for example, if such model is used to predict cancer, it might overfit to training data and gave false but confident prediction in real world. therefore, we'll prefer a regulated model to make sure it is not overfit.

4. (a) (i)

classifier = negative for all data.

then we will always be correct for ~~(1-ρ)~~ all the negative examples. ~~so~~ so $1-\rho$ is the accuracy.

(ii) $$\rho = \frac{P}{P+N} = \frac{TP+FN}{(TP+FN)+(TN+FP)}$$

$$= \frac{TP+FN}{TP+TN+FP+FN}$$

$$\rho A_1 + (1-\rho)A_0 = \frac{TP+FN}{TP+TN+FP+FN} \cdot \left(\frac{TP}{TP+FN} - \frac{TN}{TN+FP}\right) + \frac{TN}{TN+FP}$$

$$=$$

4.(a) (ii) continue.

$$\rho A_1 + (1-\rho) A_0$$

$$= \frac{TP+FN}{S} A_1 + \frac{TN+FP}{S} A_0 \quad (\text{where } S = TP+TN+FP+FN).$$

$$= \frac{TP+FN}{S} \cdot \frac{TP}{TP+FN} + \frac{TN+FP}{S} \cdot \frac{TN}{TN+FP}$$

$$= \frac{TP}{S} + \frac{TN}{S}$$

$$= \frac{TP+TN}{S} = \frac{TP+TN}{TP+TN+FP+FN}$$

$$= A.$$

4.(a) (iii)

$$\bar{A} = \frac{1}{2}(A_0 + A_1)$$

$$= \frac{1}{2}\left(\frac{TP}{TP+FN} + \frac{TN}{TN+FP}\right)$$

for a classifier always predict false.

$TP = 0$

$TN = (1-\rho) \cdot S$ where $S = TP+TN+FP+FN$

$FP = 0$

$FN = \rho \cdot S$

$$\bar{A} = \frac{1}{2}\left(\frac{0}{0+\rho S} + \frac{(1-\rho)S}{(1-\rho)S+0}\right)$$

$$= \frac{1}{2}$$

4.(b).

$A = 0.947273$

$\bar{A} = 0.7775$

$A_0 = 0.985$

$A_1 = 0.57$

Plot

4.(c) let # of all examples in D be S. then $N = (1-\rho)S$, where N is # of all negative examples and P is " " positive "

$P = \rho S$

in D' $\Rightarrow$ $TN' = TN$, $FP' = FP$

$N' = N = (1-\rho)S$

$P' = \frac{1}{k} \cdot P = \frac{(1-\rho)}{\rho} \cdot \rho S = (1-\rho)S \Rightarrow TP' = \frac{1}{k} TP$, $FN' = \frac{1}{k} FN$

$\therefore N' = P'$

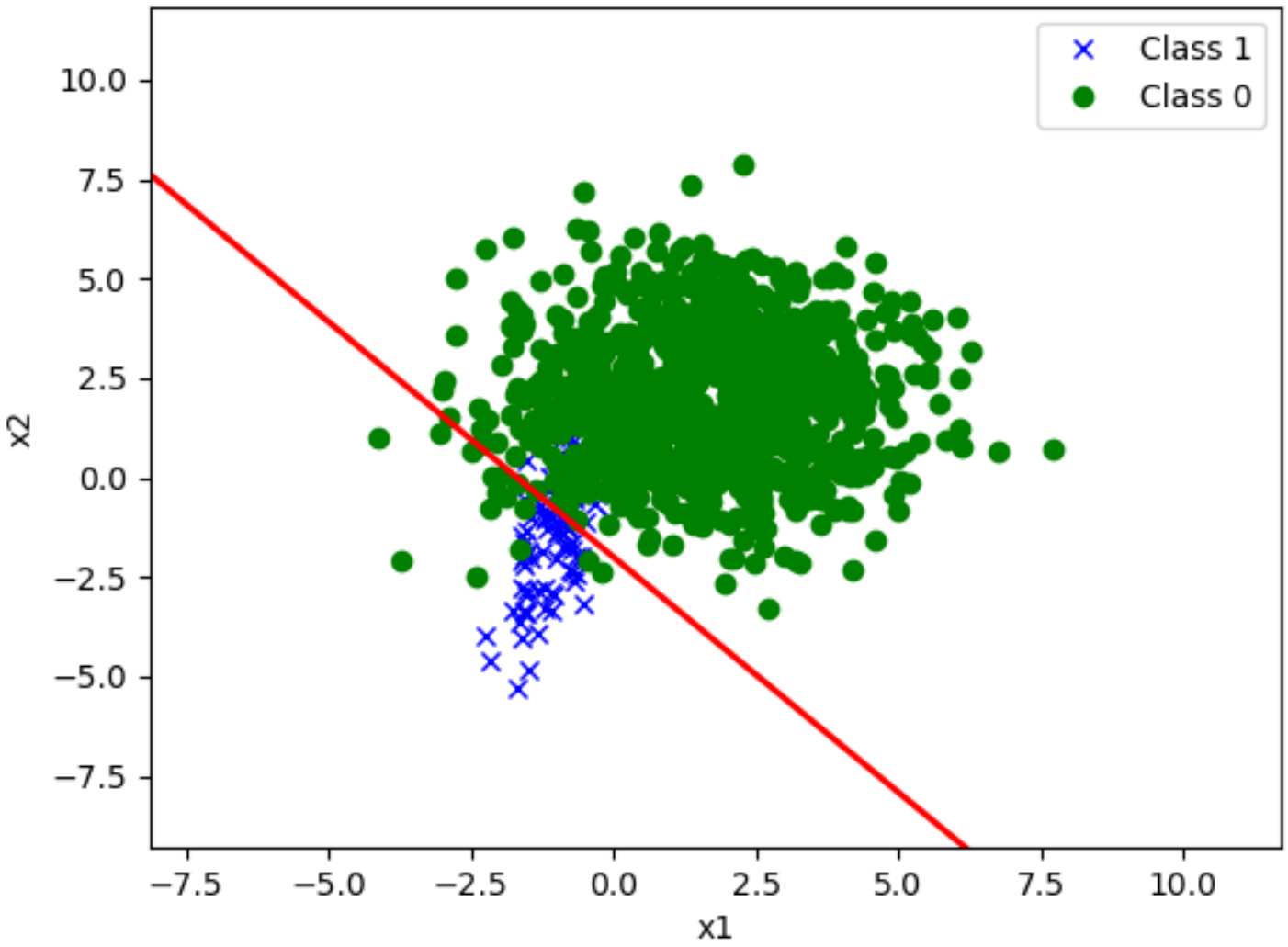
$\rho' = \frac{P'}{N' + P'} = \frac{1}{2}$

A' (accuracy of D') $= \rho' \cdot A_1' + (1-\rho') A_0'$

$= \frac{1}{2}(A_1' + A_0')$

$= \frac{1}{2}\left(\frac{TP'}{P'} + \frac{TN'}{N'}\right)$

$= \frac{1}{2}\left(\frac{\frac{1}{k}TP}{\frac{1}{k}P} + \frac{TN}{N}\right) = \frac{1}{2}\left(\frac{TP}{P} + \frac{TN}{N}\right) = \bar{A}$


Class 1
Class 0
10.0
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
x2
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0
x1

4.(c) (continue)

# of example in D' is n'

$n' = p' + N' = 2(1-p)n$ where n is # of examples in D.

$$-\frac{1}{n'} = -\frac{1}{2(1-p)n} = -\frac{1+k}{2n}$$

$J(\theta)'$ for D'

$$= -\frac{1}{n'} \sum_{i=1}^{n'} \left(y^{(i)} \log(h_\theta(x^{(i)})) + (1-y^{(i)}) \log(1-h_\theta(x^{(i)})) \right)$$

$$= -\frac{1}{n'} \left[\sum_{i=P_1', P_2', \cdots}^{P'} y^{(i)} \log(h_\theta(x^{(i)})) + \sum_{i=N_1', N_2', \cdots}^{N'} (1-y^{(i)}) \log(1-h_\theta(x^{(i)})) \right]$$

all positive examples in D' ; all negative examples in D'

$$= -\frac{1}{n'} \left[\frac{1}{k} \sum_{i=P_1, P_2, \cdots}^{P} y^{(i)} \log(h_\theta(x^{(i)})) + \sum_{i=N_1, N_2, \cdots}^{N} (1-y^{(i)}) \log(1-h_\theta(x^{(i)})) \right]$$

all positive examples in D ; all negative examples in D

$$= -\frac{1}{n'} \left[\sum_{i=P_1, P_2, \cdots}^{P} \frac{1}{k} y^{(i)} \log(h_\theta(x^{(i)})) + \sum_{i=N_1, N_2, \cdots}^{N} (1-y^{(i)}) \log(1-h_\theta(x^{(i)})) \right]$$

$$= -\frac{1}{n'} \left[\sum_{i=P_1, P_2, \cdots}^{P} w^{(i)} y^{(i)} \log(h_\theta(x^{(i)})) + \sum_{i=N_1, N_2, \cdots}^{N} w^{(i)} (1-y^{(i)}) \log(1-h_\theta(x^{(i)})) \right]$$

$w^{(i)} = 1$ if $y^{(i)} = 0$, i.e negative examples.

$= \frac{1}{k}$ if $y^{(i)} = 1$, i.e positive examples.

$$= -\frac{1}{n'} \sum_{i=1}^{n} w^{(i)} \left[y^{(i)} \log(h_\theta(x^{(i)})) + (1-y^{(i)}) \log(1-h_\theta(x^{(i)})) \right]$$

$$= -\frac{1+k}{2n} \sum_{i=1}^{n} w^{(i)} \left[y^{(i)} \log(h_\theta(x^{(i)})) + (1-y^{(i)}) \log(1-h_\theta(x^{(i)})) \right]$$

4. (d)

for D'

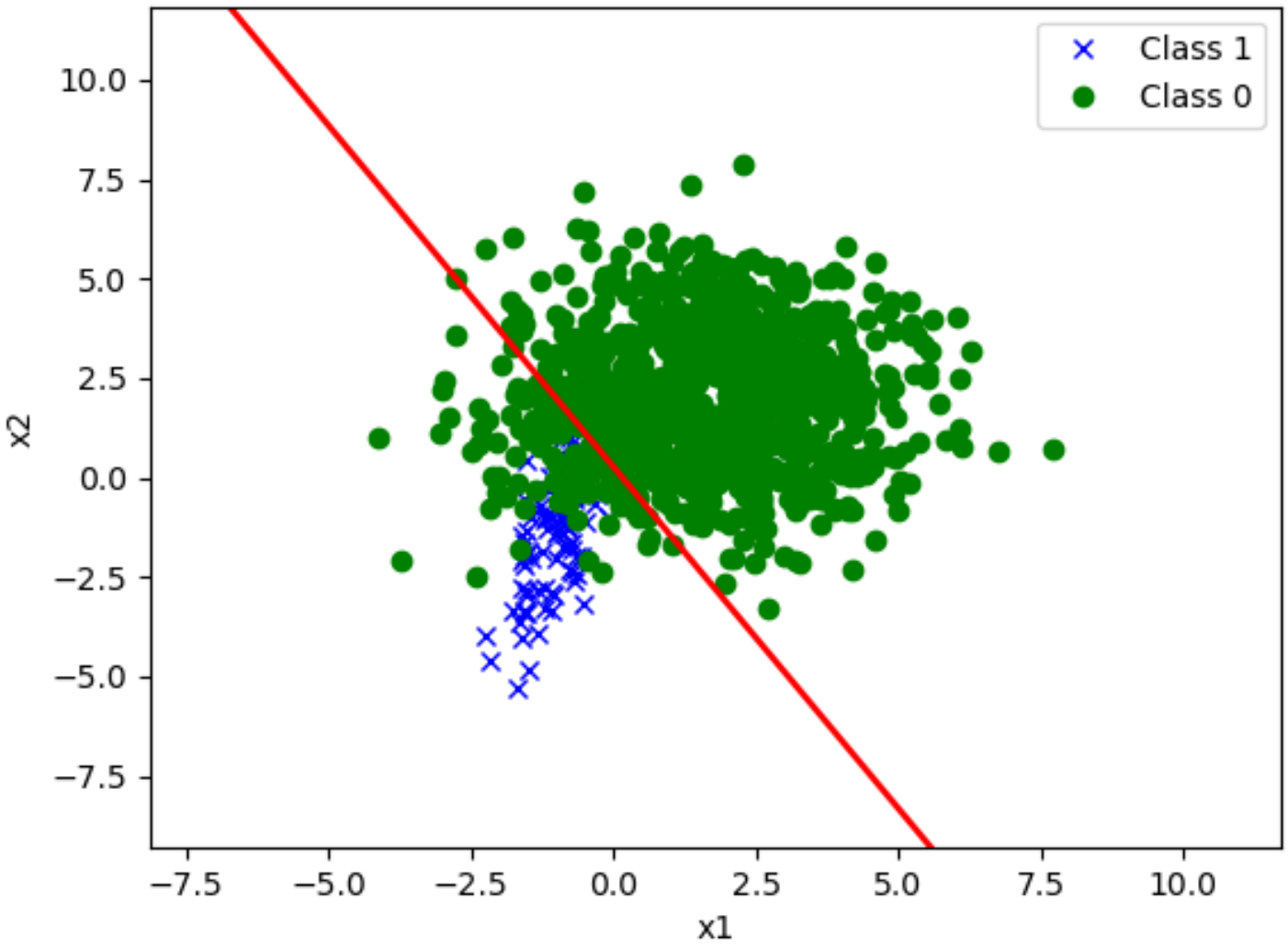
$A = 0.8991$

$\bar{A} = 0.904$

$A_0 = 0.898$

$A_1 = 0.91$

Plot


Class 1
Class 0
10.0
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
x2
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0
x1

(5) (a)

$$\frac{\partial l_{CE}(t,y)}{\partial t} = \frac{\partial}{\partial t}\left[-\log\left(\frac{\exp(t_y)}{\sum_{s=1}^{K}\exp(t_s)}\right)\right]$$

$$= \frac{\partial}{\partial t}\left[-\left(\log(\exp(t_y)) - \log\left(\sum_{s=1}^{K}\exp(t_s)\right)\right)\right]$$

$$= \frac{\partial}{\partial t}\left[\log\left(\sum_{s=1}^{K}\exp(t_s)\right) - t_y\right]$$

$$= \begin{bmatrix} \frac{\partial}{\partial t_1}\left\{\log\sum_{s=1}^{K}\exp(t_s) - t_y\right\} \\ \vdots \\ \frac{\partial}{\partial t_K}\left\{\log\sum_{s=1}^{K}\exp(t_s) - t_y\right\} \end{bmatrix}$$. by chain rule:

$$= \begin{bmatrix} \frac{\frac{\partial}{\partial t_1}\left(\sum_{s=1}^{K}\exp(t_s)\right)}{\sum_{s=1}^{K}\exp(t_s)} - \frac{\partial}{\partial t_1}t_y \\ \vdots \\ \frac{\frac{\partial}{\partial t_K}\left(\sum_{s=1}^{K}\exp(t_s)\right)}{\sum_{s=1}^{K}\exp(t_s)} - \frac{\partial}{\partial t_K}t_y \end{bmatrix}$$

$$= \begin{bmatrix} \frac{\exp(t_1)}{\sum_{s=1}^{K}\exp(t_s)} - 0 \\ \vdots \\ \frac{\exp(t_y)}{\sum_{s=1}^{K}\exp(t_s)} - 1 \end{bmatrix}$$

$$= \begin{bmatrix} \frac{\exp(t_1)}{\sum_{s=1}^{K} \exp(t_s)} - 0 \\ \vdots \\ \frac{\exp(t_y)}{\sum_{s=1}^{K} \exp(t_s)} - 1 \\ \vdots \\ \frac{\exp(t_K)}{\sum_{s=1}^{K} \exp(t_s)} - 0 \end{bmatrix} \leftarrow y^{th} \text{ item}$$

by definition.

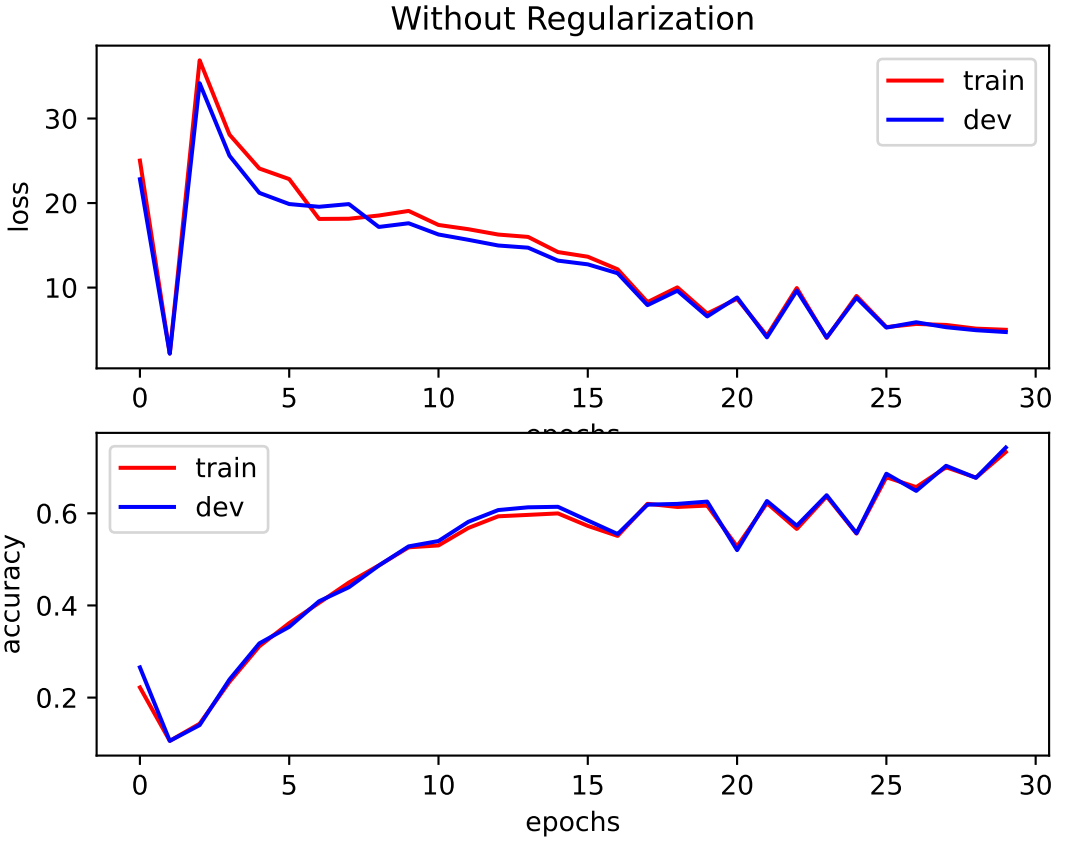
$$= \text{softmax}(t) - e_y$$

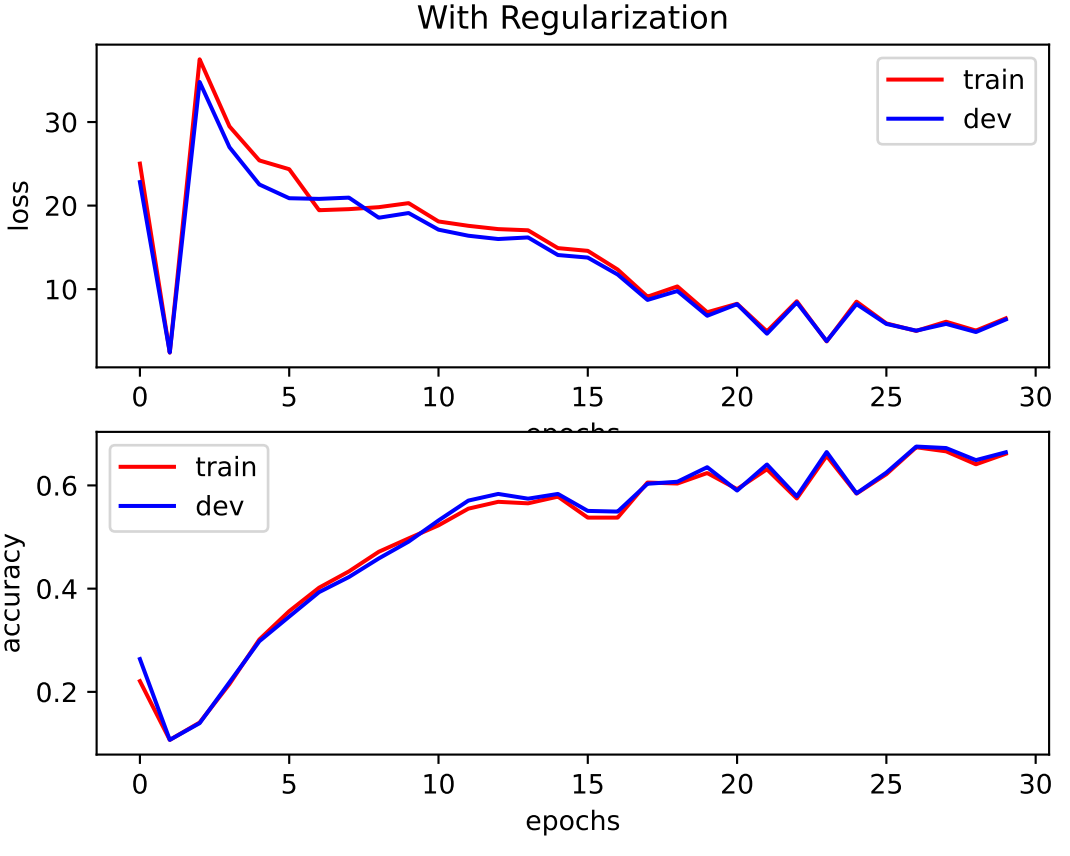
(5) (b) plot

(c) plot

For model baseline, got accuracy: 0.733700

For model regularized, got accuracy: 0.633900


Without Regularization
train
dev
loss
30
20
10
0
5
10
15
20
25
30
train
dev
accuracy
0.6
0.4
0.2
0
5
10
15
20
25
30
epochs


With Regularization
train
dev
loss
30
20
10
0
5
10
15
20
25
30
train
dev
accuracy
0.6
0.4
0.2
0
5
10
15
20
25
30
epochs