Name :- Prateek Mahajan

EE511 : Homework 2

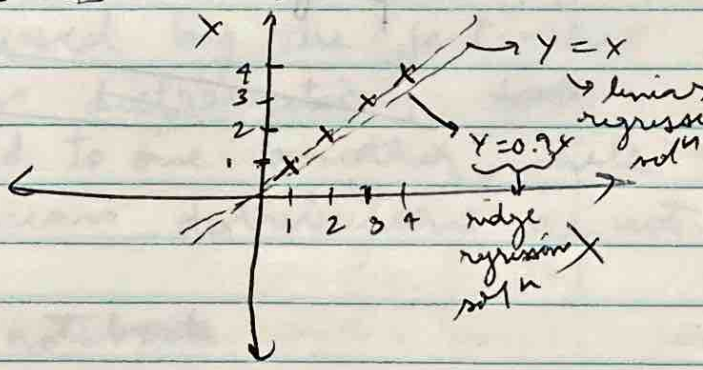
Problem 1 :-

1a) I have read and understood the general instructions ~~for HW2~~ at the top of HW2 and I formally declare that all the work I turn in for everything in this course will not contain or involve any cheating at all.

Problem 2 :-

2a) Consider a simple dataset as below :-

X	Y
1	1
2	2
3	3
4	4
⋮	⋮



For a dataset like the above, linear regression would give a sol^n of $Y = X$ with zero error

For ridge regression :- (setting $\lambda = 0.4$ / $\eta = 0.6$ & and assuming 4 datapoints

Let's assume that we get $Y = 0.9X$

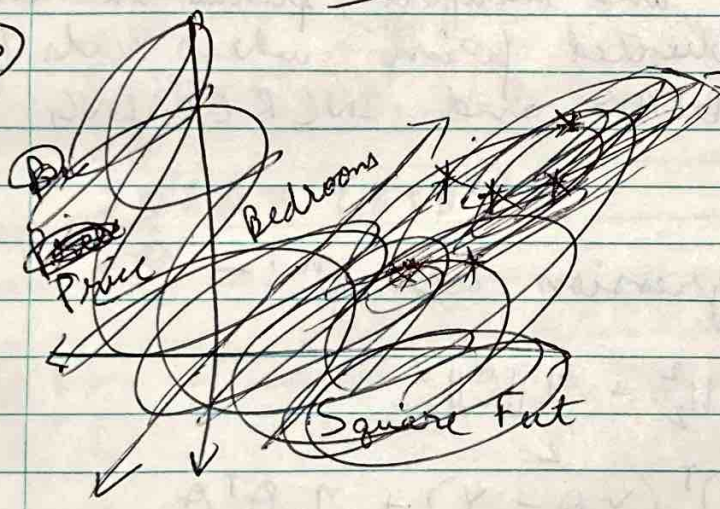
Using the ridge regression sol^n, Error $(Y = X) = 4$,
Error $(Y = 0.9X) = 3.9$

so clearly, it would be preferred despite $Y = X$

fitting the data better

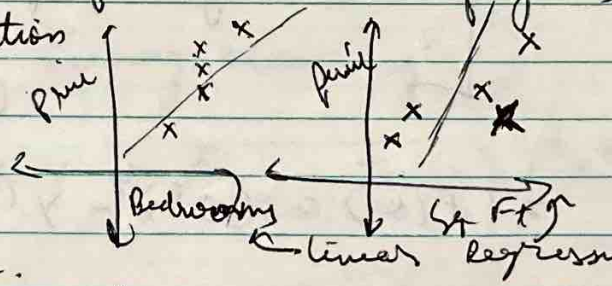
- in cases like the above, where there is only 1 feature or there is NO correlation amongst features, the linear regression is better than ridge. The reason for this is that ridge regression adds a bias that makes it good at dealing with collinearity amongst various features but makes it perform worse when the focus is on interpreting the individual effects of features & when there isn't collinearity amongst them.

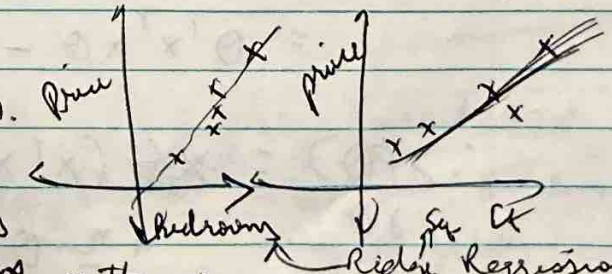
b)



Price	Bedrooms	Sq. ft.
400	3	2104
330	3	1600
369	3	2400
232	2	1416
540	4	3000
⋮	⋮	⋮

In the given example, there are 2 correlated features (no. of bedrooms & sq. feet). The addition of a regularization term adds a bias to the model that helps it better represent the correlation b/w sq. ft & no. of bedrooms than the standard linear regression. Here, we are not interested in the combined effect of the features than just their individual contributions, as they are correlated ∴ we prefer ridge regression.





b) ~~Increasing η modifies the optimization sol^n by increasing regularization and therefore, increases bias & reduces variance.~~

c) Increasing η tries to push the parameters of the optimisation problem to zero thereby simplifying the model and INCREASING BIAS and DECREASING VARIANCE by pushing the model to a more general form.

Decreasing η tries to give more importance to square error, and therefore, pushes the model to a more complicated form, which leads to DECREASING BIAS and INCREASING VARIANCE //

d) Ridge regression cost f^n :-

$$F(\theta) = \|X\theta - y\|_2^2 + \frac{\eta}{2}\|\theta\|_2^2$$

$$= (X\theta - Y)^T \cdot (X\theta - Y) + \frac{\eta}{2}\theta^T\theta$$

[in matrix form, if $X \to n \times m$ design matrix
$Y \to n \times 1$ output matrix
$\theta \to m \times 1$ parameter/feature matrix]

$$\Rightarrow F(\theta) = (\theta^T X^T - Y^T)(X\theta - Y) + \frac{\eta}{2}\theta^T\theta$$

$$= \theta^T X^T X\theta - Y^T X\theta - \theta^T X^T Y + Y^T Y + \eta/2\,\theta^T\theta$$

$$\therefore \frac{\partial F(\theta)}{\partial \theta} = (X^T X + X^T X)\theta - X^T Y - X^T Y + 0 + \frac{\eta}{2}\cdot\theta[I + I]$$

$$\Rightarrow \frac{\partial F(\theta)}{\partial \theta} = 2X^TX\theta - 2X^TY + \eta\theta$$

Setting $\frac{\partial F(\theta)}{\partial \theta} = 0 \Rightarrow 2X^TX\theta + \eta\theta = 2X^TY$

$$\Rightarrow \left(X^TX + \frac{\eta I}{2}\right).\theta = X^TY$$

$$\Rightarrow \tilde{\theta}_{optimal} = \left(X^TX + \frac{\eta I}{2}\right)^{-1}.X^TY$$

closed form soln of ridge regression

Formulas used:-

→ $\frac{\partial b^T\theta^T\theta c}{\partial \theta} = \theta(bc^T + cb^T)$

→ $\frac{\partial \theta^T a}{\partial \theta} = \frac{\partial a^T\theta}{\partial \theta} = a$

→ $\frac{\partial a^T\theta b}{\partial \theta} = ab^T$

→ $\frac{\partial \theta^T B\theta}{\partial \theta} = (B + B^T)\theta$

e) For a vanilla linear regression

1) $F(\theta) = ||X\theta - y||_2^2 = (X\theta - Y)^T.(X\theta - Y)$

$= (\theta^TX^T - Y^T).(X\theta - Y)$

$= \theta^TX^TX\theta - Y^TX\theta - \theta^TX^TY + Y^TY$

, where $X \rightarrow n \times m$ design matrix, $\theta \rightarrow m \times 1$ matrix, $Y \rightarrow n \times 1$ matrix

$\therefore \frac{\partial F}{\partial \theta} = 2X^TX\theta - 2X^TY$ (using above eqns)

$\Rightarrow \frac{\partial F}{\partial \theta} = 0 \Rightarrow X^TX\theta = X^TY \Rightarrow \tilde{\theta}_{optimal} = (X^TX)^{-1}X^TY$

$X^TX \rightarrow m \times m$ matrix

If $m > n$ and features of X are highly correlated, X^TX may NOT be invertible. (since the determinant may be zero)

X^TX is (if not invertible)

In that case, the $\tilde{\theta}$ optimal / closed form sol^n of the vanilla linear regression may not be computable.

2) Comparing the vanilla linear regression closed form sol^n to ridge regression's :-
the matrix to be inverted in ridge regression is $(X^TX + \lambda I)$, not X^TX.
∴ as long as we make sure $\lambda > 0$, $(X^TX + \lambda I)$ is always invertible ⇒ the closed form sol^n for ridge regression always exists (which is the benefit of ridge regression)

f)

(i) $F(\theta, \theta_0) = \|X\theta + \theta_0 1 - y\|_2^2 + \frac{\eta}{2}\|\theta\|_2^2$

$= (X\theta + \theta_0 1 - y)^T (X\theta + \theta_0 1 - y) + \frac{\eta}{2}\theta^T\theta$

$= (\theta^T X^T + 1^T\theta_0^T - y^T)(X\theta + \theta_0 1 - y) + \frac{\eta}{2}\theta^T\theta$

$= \theta^T X^T X\theta + \theta^T X^T \theta_0 1 - \theta^T X^T y + 1^T\theta_0^T X\theta + 1^T\theta_0^T\theta_0 1 - 1^T\theta_0^T y - y^T X\theta - y^T\theta_0 1 + y^Ty + \frac{\eta}{2}\theta^T\theta$

$\therefore \frac{\partial F(\theta, \theta_0)}{\partial \theta_0} = 0 + 1^T X\theta - 0 + 1^T\theta_0(1 + 1) - 1^T y - 0 - 1^T y + 0 + 0 + 1^T X\theta$

$\Rightarrow \frac{\partial F(\theta, \theta_0)}{\partial \theta_0} = 2X\theta 1^T + 2\theta_0 1 1^T - 2y1^T$

$= 2\,1^T X\theta + 2\,1^T\theta_0 1 - 2\,1^T y$

Setting $\frac{\partial F(\theta, \theta_0)}{\partial \theta_0} = 0$

$\Rightarrow 1^T y = 1^T X\theta + 1\theta_0 1^T$ — (1)

$\Rightarrow y = X\theta + \theta_0 1$ — (1)

$\Rightarrow y = X\theta + \theta_0 \cdot 1$ — (1)

Now consider $\frac{\partial F(\theta, \theta_0)}{\partial \theta}$

$\therefore \frac{\partial F(\theta, \theta_0)}{\partial \theta} = 2X^T X\theta + X^T \theta_0 1 - X^T y + X^T \theta_0 1 + 0 - 0$
$- X^T y - 0 + 0 + \eta\theta$

$\Rightarrow \frac{\partial F(\theta, \theta_0)}{\partial \theta} = 2X^T X\theta + 2X^T \theta_0 1 - 2X^T y + \eta\theta$

Setting $\frac{\partial F(\theta, \theta_0)}{\partial \theta} = 0$

$\Rightarrow 2X^T y = 2X^T X\theta + 2X^T \theta_0 1 + \eta\theta$

On substituting θ_0 :—

$2X^T y = 2X^T X\theta + 2X^T y - 2X^T X\theta + \eta\theta$

$2X^T y = 2X^T X\theta + 2X^T X\theta - 2X^T y + \eta\theta$

$\Rightarrow 4X^T y = 4X^T X\theta + \eta\theta$

$\Rightarrow 2X^T y 1^T = 2X^T X\theta 1^T + 2X^T (y 1^T - X\theta 1^T) + \eta\theta 1^T$

$\Rightarrow (2X^T X + \eta I) \cdot \theta = 2X^T y - 2X^T \theta_0 \cdot 1$

$\Rightarrow \theta = (X^T X + \frac{\eta}{2} I)^{-1} (X^T)(y - \theta_0 \cdot 1)$ — (2)

Taking θ_0 :—

$y \cdot 1^T = X\theta 1^T + \theta_0 1 1^T$

However, θ_0 is a scalar bias.

$\therefore \theta_0 1 1^T = \{\theta_0\}_{n \times n}$

$\therefore$ This eqn ~~is only satisfied if~~ can be interpreted as θ_0 optimal

Taking ① :—

$$1^T y = 1^T X\theta + 1^T \theta_0 1$$

$$\Rightarrow 1^T \theta_0 1 = 1^T y - 1^T X\theta$$

$$\Rightarrow \theta_0 1^T 1 = 1^T y - 1^T X\theta \quad — ③$$

$$\Rightarrow n\theta_0 = \sum_{i=1}^{n} y_i - \sum_{i=1}^{n}\sum_{j=1}^{m} x_{i,j}\,\theta_{j,1}$$

$$\Rightarrow \theta_0 = \frac{1}{n}\left(\sum_{i=1}^{n} y_i - \sum_{j=1}^{m} x_{i,j}\,\theta_{j,1}\right)$$

$$= \underline{\text{Mean of prediction error (say } \mu_e)}$$

Taking ② :—

$$\theta = (X^T X + \frac{n}{2} I)^{-1} X^T (y - \theta_0 1)$$

$$\Rightarrow \theta = (X^T X + \frac{n}{2} I)^{-1} X^T (y - \mu_e 1)$$

If we manipulate the eqn differently (i.e. substitute θ_0 before inverse) :

And $\theta_0 = \mu_y - \frac{1^T X \theta}{n}$ (from ③)

$$(X^T X + \frac{n}{2} I)\theta = X^T y + \frac{X^T 1^T X\theta 1}{n} - X^T \mu_y 1$$

$$\left[\mu_y = \frac{1}{n}\sum_{i=1}^{n} y_i\right]$$

Note that $(1^T X\theta)$ is a 1×1 result & so can be treated as a scalar

$$\Rightarrow (X^T X + \frac{n}{2} I)\theta = X^T y + \frac{X^T 1 (1^T X\theta)}{n} - X^T \mu_y 1$$

$$\Rightarrow (X^T X + \frac{n}{2} I - \frac{X^T 1 1^T X}{n})\theta = X^T (y - \mu_y 1)$$

$$\Rightarrow \theta = (X^T X + \frac{n}{2} I - \frac{X^T 1 1^T X}{n})^{-1} \cdot X^T (y - \mu_y 1)$$

NOTE :- 2f(ii) is after Problem 3b)

sorry about the confusion

~~model, and so regularising it would~~

Problem 3:-

a) Consider a gaussian noise linear least squares regression model, where $\vec{y} = X\theta + \vec{\epsilon}$, where X is a $n \times m$ design matrix & $\vec{\epsilon}$ is a length-n vector of gaussians, $\epsilon_i \in N(0, \sigma^2)$.
MLE parameter estimate $= \tilde{\theta} = (X^T X)^{-1} X^T \vec{y}$

Let θ^* be the "true" parameter estimate for the gaussian LLS model. $(\theta^* \in \mathbb{R}^{m \times 1})$

~~$\therefore \mathbb{E}(Y|x) = x\theta^*$ (x is 1 instance of X from a~~

$\therefore$ $\mathbb{E}(Y|x)$ = true prediction of model
(as $\mathbb{E}(Y|x)$ is the best fit soln) $= x^T\theta^*$, where x ~~is~~ is the instances of X from a dataset D. — (1)
($x \in \mathbb{R}^{m \times n}$)

~~$\mathbb{E}_D[h_D(x)]$~~

Consider $h_D(x) = x^T\tilde{\theta}$
$= x^T((X^TX)^{-1}X^T\vec{y})$
$= x^T(X^TX)^{-1}X^T(X\theta^* + \epsilon)$ (since $\vec{y} = X\theta^* + \epsilon$)
$= x^T\theta^* + x^T(X^TX)^{-1}X^T\epsilon$

$\therefore \mathbb{E}_D(h_D(x)) = \mathbb{E}_D(x^T\theta^* + x^T(X^TX)^{-1}X^T\epsilon)$
$= \mathbb{E}_D(x^T\theta^*) + \mathbb{E}_D(x^T(X^TX)^{-1}X^T\epsilon)$

x, X are fixed matrices which implies this term is $x^T(X^TX)^{-1}X^T \cdot \mathbb{E}_D(\epsilon) = 0$

$\therefore E_D(h_D(x)) = \mathbb{E}_D(x^T\theta^*)$
$= x^T\theta^*$ (as x, θ^* are not variables) — (2)

$\therefore$ (1) = (2) $\Rightarrow \mathbb{E}_D(h_D(x)) = \mathbb{E}(Y|x) \Rightarrow$ this model is unbiased

b) Consider the LHS $= \mathbb{E}_D[(h_D(x) - \mathbb{E}_D(h_D(x)))^2]$

$\therefore$ LHS $= \mathbb{E}_D[(h_D(x) - x^T\theta^*)^2]$

(Since $\mathbb{E}_D(h_D(x)) = x^T\theta^*$ as per 3 a))

$h_D(x) = x^T\hat{\theta}$

$= x^T(X^TX)^{-1}X^Ty$

$= x^T(X^TX)^{-1}X^T(X\theta^* + \epsilon)$

$= x^T(X^TX)^{-1}X^TX\theta^* + x^T(X^TX)^{-1}X^T\epsilon$

$= x^T\theta^* + x^T(X^TX)^{-1}X^T\epsilon$

$\therefore$ LHS $= \mathbb{E}_D((x^T\theta^* + x^T(X^TX)^{-1}X^T\epsilon - x^T\theta^*)^2)$

~~$= \mathbb{E}_D[(x(X^TX)^{-1}X^T\epsilon)^T(x(X^TX)^{-1}X^T\epsilon)]$~~

~~$= \mathbb{E}_D(x(X^TX)^{-1}X^T\epsilon\epsilon^TX(X^TX)^{-1}x^T)$~~

~~$= \mathbb{E}\, x(X^TX)^{-1}X^T\epsilon^TX(X^TX)^T$~~

~~this is~~

~~$= \mathbb{E}\, x(X^TX)^{-1}X^T\mathbb{E}(\epsilon\epsilon^T).X(X^TX)^{-1}x^T$~~

$\Rightarrow$ LHS $= \mathbb{E}_D[(x^T(X^TX)^{-1}X^T\epsilon)^2]$

$= \mathbb{E}_D[(x^T(X^TX)^{-1}X^T\epsilon)^T(x^T(X^TX)^{-1}X^T\epsilon)^T]$

$= \mathbb{E}_D[(x^T(X^TX)^{-1}X^T\epsilon)^T(x^T(X^TX)^{-1}X^T\epsilon)^T]$

$= \mathbb{E}_D[x^T(X^TX)^{-1}X^T\epsilon\epsilon^TX^T(X^TX)^{-1}]^Tx]$

x, X are not random variables

$\Rightarrow$ LHS $= \mathbb{E}[x^T(X^TX)^{-1}X^T.\mathbb{E}(\epsilon\epsilon^T).X^T(X^TX)^{-1}x]$

[Note:- (X^TX) is symmetric as it is a covariance matrix $\Rightarrow (X^TX)^T = X^TX$

$\Rightarrow ((X^TX)^T)^T = ((X^TX)^T)^{-1} = (X^TX)^{-1}$

Similarly, $\mathbb{E}(\epsilon\epsilon^T) = \sigma^2I$ (as ϵ is a GRV & $\epsilon\epsilon^T$ is a covariance matrix)]

~~LHS~~

$$\Rightarrow LHS = x^T (X^T X)^{-1} X^T (\sigma^2 I) X (X^T X)^{-1} x$$
$$= \sigma^2 x^T (X^T X)^{-1} X^T X (X^T X)^{-1} x$$
$$= \sigma^2 x^T (X^T X)^{-1} x = RHS$$

$\therefore$ LHS = RHS $\Rightarrow$ 3b) is correct

Problem 2:- (continued)

f)

(ii) $\theta_{\text{unregularised bias ridge}} = (X^T X + \frac{\eta}{2} I - \frac{X^T 1 1^T X}{n})^{-1} X^T (y - \mu_y 1)$

$\theta_{\text{normal ridge}} = (X^T X + \frac{\eta}{2} I)^{-1} X^T y$

On comparing these clearly, it looks like when we do not penalize the bias / offset / intercept parameter by regularising it, the resultant answer seems to center the "X" & "y" terms ~~more, and may~~ by subtracting their means from them. [μ_y = mean of y, $\frac{X^T 1 1^T X}{n}$ is kind of like the mean of $X^T X$ as we are essentially ~~not~~ adding a multiplicant of $\frac{1 1^T}{n}$ between $X^T X$, which is somewhat like that the average of $X^T X$]

$\therefore$ not penalizing the offset parameter "normalizes" ~~the effect of~~ X & Y ~~not~~ by subtracting their means

~~Further, bias is an inherent prope~~

Further bias is an inherent property of the model. The offset parameter is not responsible for changing the shape of the

model and so, regularising it would not prevent it from overfitting (or have any function whatsoever).

Problem 4:—

a) According to the scatter plot, LSTAT & RM are the most correlated to MEDV

b) According to the correlation matrix RM, LSTAT & RATIO are the most correlated to MEDV. The answer is different from 4a, as ~~we are trying to correlate house prices from the same dataset~~ the 3rd most correlated feature is not clear in the graphs.

c) Linear Regression:—
Closed form soln:— $\tilde{\theta} = (X^T X)^{-1} X^T y$

Ridge Regression:—
Closed form soln:— $\tilde{\theta} = (X^T X + \frac{\eta I}{2})^{-1} X^T y$

Please refer to the end of this pdf of the ipynb files for coefficient values (note that I used $\eta = 20.0$ for this problem)

d) Please refer to the .ipynb file attached at the end of this pdf for the final RMSE outputs I got for the linear & ridge regression.

Training RMSE:—
Linear < Ridge (linear is better)
This is because the ridge regression performs regularisation to reduce the tendency of the model to overfit to the training data and generalise it more (i.e. increase bias & reduce variance)

∴ since the linear regression model is trained to fit the training data better, it performs better on that data

Testing RMSE :-

Linear > Ridge (ridge is better)

The linear model does not regularise & so has higher variance than the ridge model, which regularises the cost fn to reduce the tendency of the model to overfit the training data. For this reason, the ridge model is more generalised & performs better on another dataset (i.e. the testing one)

e) Please refer to the .ipynb file for my RMSE outputs (NOTE $\eta = 20$)

On comparing the RMSE outputs of Top 3 features vs that of all 13, we can see that the RMSE is higher when we use just 3 features (albeit not significantly)

This indicates that while a large chunk of correlation with MEDV originates in these features, there are some other features in the model that have significance & should be ignored.

Problem 5:-

a) $F(\theta) = \frac{1}{n}\sum_{i=1}^{n} -\log\left[P(y=y_i \mid x=x_i, \theta)\right] + \frac{\eta}{2}\|\theta\|_2^2$

~~First let us consider~~

, where $x \in \mathbb{R}^{n \times m}$, $\theta \in \mathbb{R}^{m \times c}$

First, let us consider

$$P(y=y_i \mid x_i, \theta) = \frac{\exp(x \cdot \theta^{(k)})}{\sum_{j=1}^{c} \exp(x \cdot \theta^{(j)})}$$

(where $\theta^{(k)}$ is the k^{th} column of θ)

$$\Rightarrow \log\left(P(y=y_i \mid x_i, \theta)\right) = \underbrace{x\theta^{(k)}}_{\text{take this as } z^{(k)}} - \log\Big(\underbrace{\sum_{j=1}^{c} \exp(x \cdot \theta^{(j)})}_{\text{which implies that this is } z^{(j)}}\Big)$$

$$\therefore \frac{\partial}{\partial z^{(l)}}\left(\log(P(y=y_i \mid x_i, \theta))\right) = \frac{\partial(x\theta^{(k)})}{\partial z^{(l)}} - \frac{\partial}{\partial z^{(l)}}\left(\sum_{j=1}^{c} \exp(x \cdot \theta^{(j)})\right)$$

$$= \frac{\partial(z^{(k)})}{\partial z^{(l)}} - \frac{\partial}{\partial z^{(l)}}\left(\sum_{j=1}^{c} \exp(z^{(j)})\right)$$

(NOTE:- l is any value b/w 1 & c]

$$\frac{\partial z^{(k)}}{\partial z^{(l)}} = \begin{cases} 1, & \text{if } k = l \\ 0, & \text{otherwise} \end{cases}$$

$$\therefore \frac{\partial}{\partial z^{(l)}}\left(\log(P(y=y_i \mid x_i, \theta))\right) = 1\{k==l\} - \frac{1}{\sum_{j=1}^{c} \exp(z^{(j)})} \cdot \frac{\partial \sum_{j=1}^{c} e^{z^{(j)}}}{\partial z^{(l)}}$$

But $\frac{\partial}{\partial z^{(l)}} \sum_{j=1}^{c} e^{z^{(j)}} \simeq \frac{\partial}{\partial z^{(l)}} e^{z^{(l)}} = e^{z^{(l)}}$

(as all other terms will give zero)

(5 cont.)

$$\therefore \frac{\partial\left(\log\left(P(y=y_i|x_i,\theta)\right)\right)}{\partial z^{(l)}} = 1\{k==l\} - \frac{e^{z^{(l)}}}{\sum_{j=1}^{c} \exp(z^{(j)})}$$

$$\Rightarrow \frac{\partial\left(\log\left(P(y=y_i|x_i,\theta)\right)\right)}{\partial z^{(l)}} = 1\{k==l\} - P(y=l\,|\,x_i,\theta) \quad (1)$$

~~$\therefore \frac{\partial\left(\log\left(P(y=y_i|x_i,\theta)\right)\right)}{\partial \theta} = \frac{\partial z^{(l)}}{\partial \theta} \cdot \frac{\partial\left(\log\left(P(y=y_i|x_i,\theta)\right)\right)}{\partial z^{(l)}}$~~

~~Consider $\frac{\partial z^{(l)}}{\partial \theta} = \frac{\partial(x\cdot\theta^{(l)})}{\partial \theta} = \frac{\partial \sum_{j=1}^{m} x_j \theta_j^{(l)}}{\partial \theta}$~~

~~$= \begin{bmatrix} & 0 & \\ & \vdots & \\ & 0 & \\ & 0 & \\ x_1 & x_2 & \cdots & x_m \\ & 0 & \\ & 0 & \end{bmatrix}$ ← lth row~~

$$\therefore \frac{\partial\left(\log\left(P(y=y_i|x_i,\theta)\right)\right)}{\partial \theta} = \frac{\partial z^{(l)}}{\partial \theta} \cdot \frac{\partial}{\partial z^{(l)}}\left(\log\left(P(y=y_i|x_i,\theta)\right)\right)$$

Consider $\frac{\partial z^{(l)}}{\partial \theta} = \frac{\partial(x\cdot\theta^{(l)})}{\partial \theta} = x$ (for a single column)

$$\therefore \frac{\partial}{\partial \theta}\left(\log\left(P(y=y_i|x_i,\theta)\right)\right) = x_i\left(1\{k==l\} - P(y=l\,|\,x_i,\theta)\right)$$

For $F(\theta)$ we ~~want to~~ sum this log probability for all n & classes.

$$\Rightarrow \frac{\partial F(\theta)}{\partial \theta} = \frac{\partial}{\partial \theta}\left[\frac{1}{n}\sum_{i=1}^{n} -\log\left(\Pr(y=y_i|x_i,\theta)\right) + \frac{\eta}{2}\|\theta\|_2^2\right]$$

$$= -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{c} x_i\left(1_{\{k==l\}} - \Pr(y_i=j|x_i,\theta)\right) + \frac{\eta}{2}\frac{\partial\, \theta^T\theta}{\partial \theta}$$

$$\Rightarrow \frac{\partial F(\theta)}{\partial \theta} = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{c} x_i\left(1_{\{k==l\}} - \Pr(y_i=j|x_i,\theta)\right) + \eta\theta$$

Gradient Descent Rule for θ :-

$$\theta \Leftarrow \theta - \alpha \nabla . \frac{\partial F(\theta)}{\partial \theta}$$

$$\Rightarrow \theta \Leftarrow \theta + \frac{\alpha}{n} \sum_{i=1}^{n} \sum_{j=1}^{c} x_i \left(1_{\{k=j\}} - P_\theta(y_i = j \mid x_i, \theta) \right)$$

$$- \alpha \eta \theta \quad \text{---} \quad (2)$$

For a matrix formulation :-

revert back to

$$\frac{\partial}{\partial \theta} \Rightarrow \frac{\partial (\log(P(y=y_i \mid x_i, \theta)))}{\partial \theta} = \frac{\partial z^{(i)}}{\partial \theta} \cdot \frac{\partial (\log(P(y=y_i \mid x_i \theta)))}{\partial z^{(i)}}$$

(consider $\frac{\partial z^{(i)}}{\partial \theta}$)

Branching it out further (to ∂z instead of $\partial z^{(i)}$)

$$\frac{\partial (\log(P(y=y_i \mid x_i, \theta)))}{\partial \theta} = \frac{\partial z}{\partial \theta} \cdot \frac{\partial}{\partial z} (\log(P(y=y_i \mid x_i, \theta)))$$

$$\frac{\partial z}{\partial \theta} \approx X^T \theta \quad (\text{where } z = x\theta)$$

Using θ (extrapolating θ) :-

$$\frac{\partial}{\partial z} (\log(P(y=y_i \mid x_i, \theta))) = Y - P,$$

where P is a $n \times c$ matrix, ~~containing the~~ where

~~$P_{i,j} = \exp(x_{i,j} \cdot \theta_{j,c})$~~

$$P_{i,k} = \frac{\exp(x_{i,j} \cdot \theta_{j,c})}{\sum_{l=1}^{c} \exp(x_{i,j} \cdot \theta_{j,c})}$$ and assuming Y is a 1 hot encoding matrix of size $n \times c$

~~the matrix for~~

$$\therefore \frac{\partial}{\partial \theta} (\log(P(y=y_i \mid x_i, \theta))) = X^T . (Y - P)$$

$$\therefore \frac{\partial F}{\partial \theta} \text{ (in matrix form)} = -\frac{1}{N} X^T (Y - P) + \eta \theta$$

The least mean squares gradient descent rule is :-

$$\theta \leftarrow \theta + \alpha \left(y^{(i)} - h_\theta(x^{(i)})\right)(x^{(i)})$$

On comparing ② to the above :-

clearly both are of the form
$\theta \leftarrow \theta' + \alpha \times$ error, where error is the deviation of the prediction from the original output in the data.

b) Please refer to the .ipynb file at the end of this pdf for this qn.

c) Based on the convergence curve, clearly as we increase the learning rate, the model gets trained faster / requires fewer training epochs.
However we also see that the final cost fn is lower & training / testing precision is higher to a certain point as we reduce the learning rate.
The above are the tradeoffs to consider.
However, clearly, 0.01 is the sweet spot of this model as it is the point where cost fn is lowest, precision is highest and training time is acceptable / not too long

d) Please refer to the .ipynb file attached at the end of the pdf for this qn.

e) No, different batch sizes show different convergence rates with the same learning

rate. From the fig. convergence speed reduces as batch size increases.

Pleased find the curves with the tuned learning rates at the end of the .ipynb notebook.

Since the new curves are tuned better they give a better precision o/p than the earlier ones.

A higher learning rate worked better for smaller batches sizes because increasing λ reduces the impact of regularisation, which can become overpowering when you reduce your batch size.

A high learning rate of 0.3 yielded the overall fastest convergence in terms of wall clock time