STA 243: Homework 2

Chenghan Sun (SID: 915030521)
e-mail: chesun@ucdavis.edu

Nanhao Chen (SID: 914432243)
e-mail: nhchen@ucdavis.edu

Submitted on 05/08/2020

Pledge:

Please sign below (print full name) after checking (✓) the following. If you can not honestly check each of these responses, please email me at kbala@ucdavis.edu to explain your situation.

- We pledge that we are honest students with academic integrity and we have not cheated on this homework. (✓)
- These answers are our own work. (✓)
- We did not give any other students assistance on this homework. (✓)
- We understand that to submit work that is not our own and pretend that it is our is a violation of the UC Davis code of conduct and will be reported to Student Judicial Affairs. (✓)
- We understand that suspected misconduct on this homework will be reported to the Office of Student Support and Judicial Affairs and, if established, will result in disciplinary sanctions up through Dismissal from the University and a grade penalty up to a grade of "F" for the course. (✓)

Team Member 1: *Nanhao Chen* Team Member 2: *Chenghan Sun*

1. Starting from the 1-dimensional case, assume that f is differentiable and convex. Let $\theta_2 > \theta_1$ and θ_0 is in between θ_1 and θ_2. Since f is convex, then

$$f(\theta_0) = f(\frac{\theta_2 - \theta_0}{\theta_2 - \theta_1}\theta_1 + \frac{\theta_0 - \theta_1}{\theta_2 - \theta_1}\theta_2) \leq \frac{\theta_2 - \theta_0}{\theta_2 - \theta_1}f(\theta_1) + \frac{\theta_0 - \theta_1}{\theta_2 - \theta_1}f(\theta_2).$$

Therefore,

$$\frac{f(\theta_0) - f(\theta_1)}{\theta_0 - \theta_1} \leq \frac{\frac{\theta_2 - \theta_0}{\theta_2 - \theta_1}f(\theta_1) + \frac{\theta_0 - \theta_1}{\theta_2 - \theta_1}f(\theta_2) - f(\theta_1)}{\theta_0 - \theta_1} = \frac{f(\theta_2) - f(\theta_1)}{\theta_2 - \theta_1}$$

According to the definition of the derivative, $\bigtriangledown f(\theta_1) = lim\frac{f(\theta_0) - f(\theta_1)}{\theta_0 - \theta_1}$, therefore,

$$\bigtriangledown f(\theta_1) \leq \frac{f(\theta_2) - f(\theta_1)}{\theta_2 - \theta_1}.$$

And thus,

$$f(\theta_2) \geq f(\theta_1) + \bigtriangledown f(\theta_1)(\theta_2 - \theta_1)$$

By simply flip the dummy variables θ_1, θ_2, this assertion is still true if $\theta_2 < \theta_1$.

On the other hand, if $f(\theta_2) \geq f(\theta_1) + \bigtriangledown f(\theta_1)(\theta_2 - \theta_1)$ with a differential f and $\theta_2 > \theta_0 > \theta_1$,

$$f(\theta_2) \geq f(\theta_0) + \bigtriangledown f(\theta_0)(\theta_2 - \theta_0)$$
$$\Rightarrow \bigtriangledown f(\theta_0) \leq \frac{f(\theta_2) - f(\theta_0)}{\theta_2 - \theta_0}$$
$$f(\theta_1) \geq f(\theta_0) + \bigtriangledown f(\theta_0)(\theta_1 - \theta_0)$$
$$\Rightarrow \bigtriangledown f(\theta_0) \geq \frac{f(\theta_1) - f(\theta_0)}{\theta_1 - \theta_0}, since(\theta_1 - \theta_0) < 0$$

Suppose $f(\theta_0) > \frac{\theta_2 - \theta_0}{\theta_2 - \theta_1}f(\theta_1) + \frac{\theta_0 - \theta_1}{\theta_2 - \theta_1}f(\theta_2)$, we will have

$$\bigtriangledown f(\theta_0) \leq \frac{f(\theta_2) - f(\theta_0)}{\theta_2 - \theta_0} < \frac{f(\theta_2) - \frac{\theta_2 - \theta_0}{\theta_2 - \theta_1}f(\theta_1) - \frac{\theta_0 - \theta_1}{\theta_2 - \theta_1}f(\theta_2)}{\theta_2 - \theta_0} = \frac{f(\theta_2) - f(\theta_1)}{\theta_2 - \theta_1}$$

$$\bigtriangledown f(\theta_0) \geq \frac{f(\theta_1) - \frac{\theta_2 - \theta_0}{\theta_2 - \theta_1}f(\theta_1) - \frac{\theta_0 - \theta_1}{\theta_2 - \theta_1}f(\theta_2)}{\theta_1 - \theta_0} = \frac{f(\theta_2) - f(\theta_1)}{\theta_2 - \theta_1}$$

These two inequities contradict with each other, and therefore, it has to be

$$f(\theta_0) \leq \frac{\theta_2 - \theta_0}{\theta_2 - \theta_1}f(\theta_1) + \frac{\theta_0 - \theta_1}{\theta_2 - \theta_1}f(\theta_2).$$

$$f(\theta_0) = f(\frac{\theta_2 - \theta_0}{\theta_2 - \theta_1}\theta_1 + \frac{\theta_0 - \theta_1}{\theta_2 - \theta_1}\theta_2) \leq \frac{\theta_2 - \theta_0}{\theta_2 - \theta_1}f(\theta_1) + \frac{\theta_0 - \theta_1}{\theta_2 - \theta_1}f(\theta_2).$$

Thus, function f is convex.

Given to the multi-dimensional cases, the derivation still works. Starting from $f(x)$, and unit direction vector k, we have $f(x + ak)$. The gradient now becomes

$$\frac{\partial}{\partial a}f(x + ak) = \bigtriangledown f(x + ak)^T k.$$

Suppose the function with another vector $f(x + bk)$, for any x and k, we use the 1-D result. The function is convex if and only if

$$f(x + ak) > f(x + bk) + (a - b) \bigtriangledown f(x + bk)^T k, \forall a, b \in R.$$

2. **a**

For part (a), a linear model (denoted "linear model a") was built using the lm() build-in function in R based on the training data (data provided as "train.data.csv"). To perform the linear regression, we used four variables: bedrooms, bathrooms, sqft living, and sqft lot, which represents features of an arbitrary house. The R^2 of this model is provided in Table 1 as below; we also used the same four variables versus "price" as the response variable to perform a linear regression on the test data (data provided as "test.data.csv"), the corresponding R^2 on test data is also provided as below:

Table 1: R^2 of Linear Model A

	Training data	Testing data
R^2	0.510113853079458	0.50499446140371

Clearly, the poor R^2 indicates relatively bad explained variances for both training and test data.

b

For part (b), we applied the "linear model A" from Part (a) to Bill Gates' house data (data provided as "fancyhouse.csv") for prediction of BC's house price. The resulting predicted house price is \$15436769.5382226. To demonstrate if the estimation is reasonable, from Wikipwdia, we found the actual BC's house price is \$147.5 million, which is roughly 10 times the value of our predicted price. This result is as expected, since the data of features in fancyhouse.csv was majorly based on "normal" house features, thus most features of BC's fancy house would be identified as outliers for the linear model we obtained using data from "normal" houses. In conclusion, this estimation of BC's house price is a poor extrapolation.

c

For Part (c), we raised a improved model (denoted as "linear model C") by adding another variable by multiplying the number of bedrooms by the number of bathrooms to the "linear model A". After this step of feature engineering, we reported the R^2 on training data of this new model as provided in Table 2 below; same as before, we also calculated R^2 of test data, provided in the same table.

Table 2: R^2 of Improved Model C

	Training data	Testing data
R^2	0.517353292773831	0.510535545859055

Clearly, the poor R^2 still indicates relatively bad explained variances for both training and test data, but slightly improved from Part (a).

d

For Part (d), we are asked to implement gradient descent algorithm on the sample-based least-squares objective function, to estimate the OLS regression parameter vector. The code could be found from the submitted .R file named "CS_NC_code_hw2.R". The implementation guide is provided as below:

Firstly, in order to apply gradient descent algorithm, make a function as step size finder by concept of bounded eigenvalues called "select_eta". This numbers calculated based on the argument standardized design matrix X from the two models in Part (a) and (c) would be used as fixed step size for gradient descent algorithm. Secondly, the gradient descent algorithm was implemented in the next function called "gradient_descent", which took multiple arguments (detailed arguments list was provided as doc-string in the .R file under the function). Noted that the deign matrix was standardized (controlled by verbose "standardize" argument) in the function as it would improve the algorithm performance without negative effects on the fitting model.

Here we summarized both the parameters and algorithm performances as Table 3 below, for both the two models ("linear model A", and "Improved model C").

Table 3: Gradient Descent Algorithm Model Facts

Models	Step Size (η)	Tolerance	Max Iterations	Actual Iterations
Linear model A	$5.31 * 10^{-05}$	10^{-8}	10^4	150
Improved model C	$4.1 * 10^{-05}$	10^{-8}	10^4	1448

We concluded that the gradient descent algorithm took relatively less number of iterations to converge even for such strict tolerance, which identified the improved performance of the algorithm. To look into the predictive power, we summarized the R^2s on both training and test data sets and corresponding BC's house prices as Table 4 below.

Table 4: Gradient Descent Algorithm Model Results

Models	Training data R^2	Testing data R^2	BC's House Price
Linear model A	0.510113853079458	0.504933222347898	15436769.5382226
Improved model C	0.51735329277383	0.51052055091153	18607312.8904335

Based on the summarizing above, we conclude that the gradient descent algorithm showed excellent performance for the estimation of the OLS regression parameter vectors, as the corresponding estimated R^2's were very close to the R^2s listed in Part (a) and (c) as standards. However, for the prediction of BC's house prices, we saw difference between the two models. The linear model A predicted more similar price compared with the predicted price in Part (b) than improved model C, as mentioned before, this is because of the poor extrapolation nature of the models.

e

For Part (e), we are asked to perform all the things above now using stochastic gradient descent algorithm (with one sample in each iteration). The code could be found from the submitted .R file named "CS_NC_code_hw2.R". The implementation guide is provided as below:

The stochastic gradient descent algorithm was implemented in the function called "stochastic_gd", which took multiple arguments (detailed arguments list was provided as doc-string in the .R file under the function). Noted that the deign matrix was standardized (controlled by verbose "standardize" argument) in the function as it would improve the algorithm performance without negative effects on the fitting model. Based on the requirements of the question, we employed the feature for sampling without replacement and when running out of samples, randomly pick one sample from the sample pool. The step size for the stochastic gradient descent algorithm is no longer a constant, we used the decreasing step sizes $\eta_t = C/(t+1)$ with iteration steps based on conclusion of course material page.17 of OPT.pdf.

Here we summarized both the parameters and algorithm performances as Table 5 below, for both the two models ("linear model A", and "Improved model C").

Table 5: Stochastic Gradient Descent Algorithm Model Facts

Models	Tuning Parameter C	Tolerance	Max Iterations	Actual Iterations
Linear model A	1	10^{-8}	10^5	3685
Improved model C	1	10^{-8}	10^5	18423

We concluded that the stochastic gradient descent algorithm took significantly more number of iterations to converge comparing with the gradient descent algorithm, with the same tolerance. To look into the predictive power, we summarized the R^2s on both training and test data sets and corresponding BC's house prices as Table 6 below.

Table 6: Stochastic Gradient Descent Algorithm Model Results

Models	Training data R^2	Testing data R^2	BC's House Price
Linear model A	0.499375814372952	0.496599946287728	14643347.8745706
Improved model C	0.510706532541672	0.506375594450772	15269772.9836313

Based on the summarizing above, we conclude that the stochastic gradient descent algorithm showed pretty good performance for the estimation of the OLS regression parameter vectors (especially for the "Improved model C"), even the predicted R^2s were not as accurate as the gradient descent algorithm. We expect this behavior since we performed a mini-batch sample selection rule. As we don't iterate through all the data rows, our selected samples from the overall sample pool were limited (a.k.a not using all data information). Thus, the stochastic gradient descent algorithm R^2s performance were slightly poor than using the gradient descent algorithm.

For the prediction of BC's house prices, we saw difference between the two models. The improved model C predicted more similar price compared with the predicted price

in Part (b) than linear model A, as mentioned before, this is because of the poor extrapolation nature of the models.

As the last component of Part (e), to demonstrate the randomness of stochastic gradient descent algorithm, and prove the quality of results we obtained in the Table above, we provided two plots of the loss functions versus iteration numbers for both the linear model A and Improved model C as Figure 1 and Fugure 2, respectively.

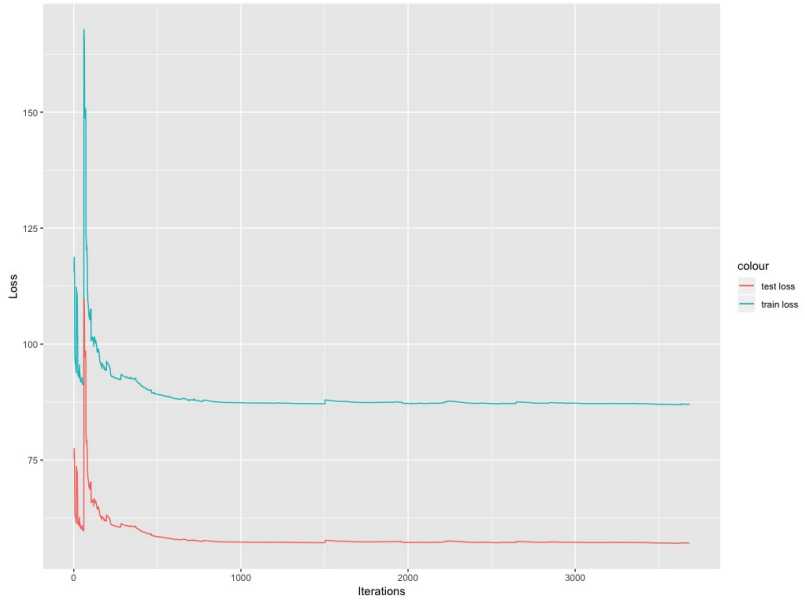

Figure 1: The loss value of Linear model A

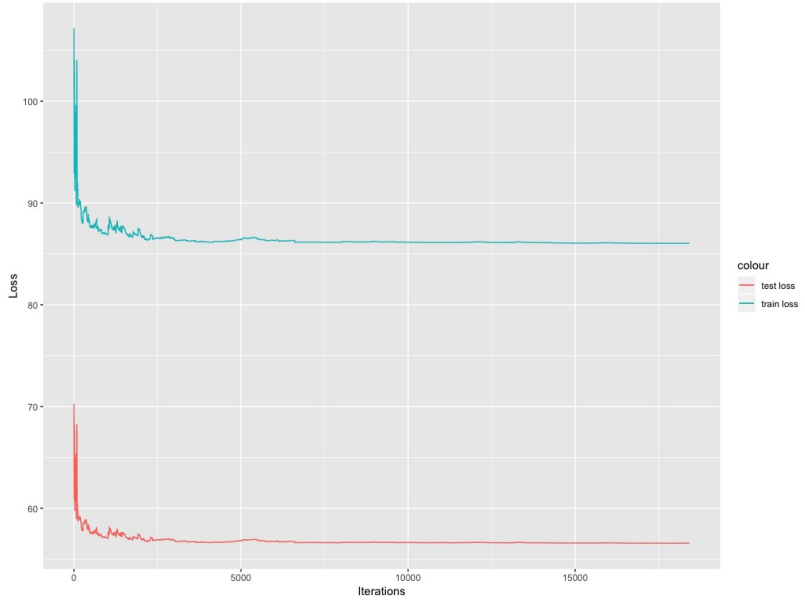

Figure 2: The loss value of Improved model C

Based on the figures above, we saw the step-wise converge curve of the loss function

as training the models. We could conclude that our model were converged and not got circumstanced by local optimal points. Thus, the models performance results we showed previously were reliable enough for our experiments.

3. Proof:

Since $f(x)$ is μ-strongly convex and differentiable, then $g(x) = f(x) - \frac{\mu}{2}||x||^2$ is convex and differentiable. Thus

$$g(\theta_2) \geq g(\theta_1) + \bigtriangledown g(\theta_1)^T(\theta_2 - \theta_1)$$

$$\Rightarrow f(\theta_2) - \frac{\mu}{2}||\theta_2||^2 \geq f(\theta_1) - \frac{\mu}{2}||\theta_1||^2 + \bigtriangledown f(\theta_1)^T(\theta_2 - \theta_1) - \mu\theta_1(\theta_2 - \theta_1)$$

$$\Rightarrow f(\theta_2) \geq f(\theta_1) + \frac{\mu}{2}||\theta_2||^2 - \frac{\mu}{2}||\theta_1||^2 - \mu\theta_1^T\theta_2 + \mu||\theta_1||^2 + \bigtriangledown f(\theta_1)^T(\theta_2 - \theta_1)$$

$$\Rightarrow f(\theta_2) \geq f(\theta_1) + \frac{\mu}{2}||\theta_2 - \theta_1||^2 + \bigtriangledown f(\theta_1)^T(\theta_2 - \theta_1)$$

Since function $g(x)$ is convex and differentiable, $\bigtriangledown g(x)$ is monotonically non-decreasing in x. Therefore,

$$(\bigtriangledown g(\theta_2) - \bigtriangledown g(\theta_1))^T(\theta_2 - \theta_1) \geq 0, \forall\theta_1, \theta_2.$$

$$\Rightarrow (\bigtriangledown f(\theta_2) - \bigtriangledown f(\theta_1) + \mu(\theta_2 - \theta_1))^T(\theta_2 - \theta_1) \geq 0$$

$$\Rightarrow (\bigtriangledown f(\theta_2) - \bigtriangledown f(\theta_1))^T(\theta_2 - \theta_1) - \mu||\theta_2 - \theta_1||^2 \geq 0$$

$$\Rightarrow (\bigtriangledown f(\theta_2) - \bigtriangledown f(\theta_1))^T(\theta_2 - \theta_1) \geq \mu||\theta_2 - \theta_1||^2, \forall\theta_1, \theta_2.$$

In order to prove the inequity $E[||\theta^{(t)} - \theta^*||_2^2] \leq \frac{c_0}{t+1}$, several properties have been stated following:

$$E_\xi[||g(\theta^{(t)}, \xi^{(t)}||_2^2] \leq \sigma_g^2 + M_g E[|| \bigtriangledown f(\theta^{(t)})||_2^2] \tag{1}$$

$$E[(\theta^{(t)} - \theta^*)^T g(\theta^{(t)}; \xi^{(t)})] = E[(\theta^{(t)} - \theta^*)E[g(\theta^{(t)}; \xi^{(t)})|\xi^{(1),\ldots,\xi^{(t-1)}}]] = E[(\theta^{(t)} - \theta^*)^T \bigtriangledown f(\theta^{(t)})] \tag{2}$$

Plus the inequity proved above

$$(\bigtriangledown f(\theta_2) - \bigtriangledown f(\theta_1))^T(\theta_2 - \theta_1) \geq \mu||\theta_2 - \theta_1||^2$$

$$\Rightarrow E[(\theta^{(t)} - \theta^*)^T(\bigtriangledown f(\theta^{(t)}) - \bigtriangledown f(\theta^*))] = E[(\theta^{(t)} - \theta^*)^T(\bigtriangledown f(\theta^{(t)})] \geq \mu E[||\theta^{(t)} - \theta^*||_2^2] \tag{3}$$

Starting from $\theta^{(1)}$, we have

$$||\theta^{(1)} - \theta^*||_2^2 = ||\theta^{(0)} - \eta_0 g(\theta^{(0)}; \xi^{(t)}) - \theta^*||_2^2$$

$$= ||\theta^{(0)} - \theta^*||_2^2 + \eta_1^2||g(\theta^{(0)}; \xi^{(t)})||_2^2 - 2\eta_1(\theta^{(0)} - \theta^*)^T g(\theta^{(0)}; \xi^{(t)})$$

$$\leq ||\theta^{(0)} - \theta^*||_2^2 + \eta_0^2(\sigma_g^2 + M_g E[|| \bigtriangledown f(\theta^{(0)})||_2^2]) - 2\eta_0\mu||\theta^{(0)} - \theta^*||_2^2$$

Since function $f(\theta)$ is a L-smooth convex function, $E[||\bigtriangledown f(\theta^{(t)})||_2^2] \leq L^2 E[||\theta^{(t)} - \theta^*||_2^2]$. Due to the fact that it is an optimization process, the gradient of the function will get

closer and closer to a small number, indicating that the gradient is bound to constant ceiling G. And therefore, we assume $E_\xi[||g(\theta^{(t)}, \xi^{(t)}||_2^2] \leq G$ The inequity becomes

$$||\theta^{(1)} - \theta^*||_2^2 \leq (1 - 2\mu\eta_0)||\theta^{(0)} - \theta^*||_2^2 + \eta_0^2 G^2.$$

And therefore,

$$E[||\theta^{(1)} - \theta^*||_2^2] \leq (1 - 2\mu\eta_0)E[||\theta^{(0)} - \theta^*||_2^2] + \eta_0^2 G^2$$

Therefore, according to the induction, we have the following inequity:

$$E[||\theta^{(k+1)} - \theta^*||_2^2] \leq (1 - 2\mu\eta_k)E[||\theta^{(k)} - \theta^*||_2^2] + \eta_k^2 G^2 \tag{4}$$

Since $\theta^{(0)}$ is the initial guess of the SGD, let $c_0 = \max\{E[||\theta^{(0)} - \theta^*||_2^2], G^2/\mu^2\}$ and $c = \frac{1}{\mu}$. Assume that the convergence rate holds with t, and $E[||\theta^{(t)} - \theta^*||_2^2] \leq \frac{c_0}{t+1}$. We can easily prove that when $t = 0$, $E[||\theta^{(0)} - \theta^*||_2^2] \leq c_0 = \frac{c_0}{0+1}$. Then

$$E[||\theta^{(t+1)} - \theta^*||_2^2] \leq (1 - 2\mu\eta_t)E[||\theta^{(t)} - \theta^*||_2^2] + \eta_t^2 G^2 \leq (1 - 2\mu\frac{c}{t+1})\frac{c_0}{t+1} + \frac{c^2 G^2}{(t+1)^2}$$

$$\leq (1 - \frac{2}{t+1})\frac{c_0}{t+1} + \frac{\frac{1}{\mu^2}c_0\mu^2}{(t+1)^2} = (\frac{1}{t+1} - \frac{2}{(t+1)^2})c_0 + \frac{c_0}{(t+1)^2}$$
$$\leq (\frac{1}{t+1} - \frac{2}{(t+1)^2})c_0 \leq (\frac{1}{t+1} - \frac{1}{(t+1)^2})c_0$$

For a positive number $k > 2$, $\frac{1}{k} - \frac{1}{k^2} \leq \frac{1}{k+1}$ is always true because $\frac{1}{k} - \frac{1}{k^2} = \frac{k-1}{k^2} \leq \frac{1}{k+1}$

$$\iff (k-1)(k+1) \leq k^2 \iff k^2 - 1 \leq k^2$$

Therefore, when $k = t + 1$, then

$$\frac{1}{t+1} - \frac{1}{(t+1)^2} \leq \frac{1}{t+2}$$

The inequity becomes

$$E[||\theta^{(t+1)} - \theta^*||_2^2] \leq (\frac{1}{t+1} - \frac{1}{(t+1)^2})c_0 \leq \frac{c_0}{t+2}$$

Therefore, $E[||\theta^{(t)} - \theta^*||_2^2] \leq \frac{c_0}{t+1}$ is true for some $c > 0$ and c_0 is constant.