STA 243 - Computational Statistics

Homework II

Authors:
Han Chen 917870226
Cassie Xu 917800791

Instructor:
Dr. Krishnakumar
Balasubramanian

May 8th, 2020

Contents

1 Question 1

(**5 Points**) Prove that a differentiable function $f(\theta) : \mathbb{R}^d \to \mathbb{R}$ is convex if and only if

$$f(\theta_2) \geq f(\theta_1) + \nabla f(\theta_1)^\top (\theta_2 - \theta_1)$$

Hint: Think of 1-dimensional case and extend the intuition to d-dimensional case.

Solution

- if differentiable function $f(\theta)$ is convex function, then $\forall t \in [0,1]; \theta_1, \theta_2 \in \mathbb{R}^d$
  Denote $g(t) = f(\theta_1 + t(\theta_2 - \theta_1))$, from the convexity of function f

  $$g(t) \leq (1-t)g(0) + tg(1)$$

  which is equivalent to

  $$\frac{g(t) - g(0)}{t} \leq \frac{g(1) - g(t)}{1-t}$$

  On LHS, let $t \to 0$, then it follows $g'(t) \leq g(1) - g(0)$, i.e.

  $$\nabla f(\theta_1)^T (\theta_2 - \theta_1) \leq f(\theta_2) - f(\theta_1)$$

- if $\theta_1, \theta_2 \in \mathbb{R}^d$ and

  $$f(\theta_2) \geq f(\theta_1) + \nabla f(\theta_1)^\top (\theta_2 - \theta_1)$$

  Denote $\theta_3 = \alpha\theta_1 + (1-\alpha)\theta_2$, $\forall \alpha \in [0,1]$

  $$f(\theta_1) \geq f(\theta_3) + \nabla f(\theta_3)(\theta_1 - \theta_3)$$

  $$f(\theta_2) \geq f(\theta_3) + \nabla f(\theta_3)(\theta_2 - \theta_3)$$

  then

  $$(1-\alpha)f(\theta_2) + \alpha f(\theta_1) \geq f(\theta_3)$$

  Here we use the fact that $(1-\alpha)(\theta_2 - \theta_3) + \alpha(\theta_1 - \theta_3) = 0$

2 Question 2

(**20 Points**) The origin of the dataset `housingprice.csv` we will use in this question is from the Coursera open course Machine Learning Foundations: A Case Study Approach by Prof. Carlos Guestrin and Prof. Emily Fox. Load the training data `train.data.csv` and testing data `test.data.csv`. We'll build our regression model on the training data and evaluate the model on the testing data.

1. Build a linear model on the training data using lm() by regessing the housing price on these variables: `bedrooms`, `bathrooms`, `sqft_living`, and `sqft_lot`. What's the R^2 of the model on training data? What's the R^2 on testing data?

	Training Data	Test Data
R-square Value	0.510114	0.504933

2. The image below is Bill Gates' house. Load the file `fancyhouse.csv` to obtain the features of the house. Guess the price of his house using your linear model. Do you think the predicted price is reasonable?

Figure 1: Image from Wikipedia Commons

	Predicted Value	Real Value
Bill Gate's House	15,390,168	127,000,000

From the table above, we can see that the predicted price is only about one-tenth of the real price of Bill Gates's house. The reason may be that the linear model cannot handle an outlier such as the condition for a fancy house well.

3. Let's continue to improve the linear model we have. Instead of throwing only the raw data into the statistical model, we might want to use our intuition and domain expertise to extract more meaningful features from the raw data. This step is called feature engineering. Using meaningful features in the model is often crucial for successful data analysis. Add another variable by multiplying the number of bedrooms by the number of bathrooms, which describes the combined benefit of having more bedrooms and bathrooms. Add this variable to the linear model we have in Part (a). What's the R^2 of the new model on the training data and testing data respectively?

   (**Hint:** You don't have to create a new column in the data frame. Try this trick in `lm()`: `lm( y ~ x1 + x2 + x1 * x2, data = your.data)`)

	Training Data	Test Data
R-square Value	0.517353	0.510528

4. Perform all the things above without using the in-built function `lm()` in R, but by using `gradient descent algorithm` on the sample-based least-squares objective function, to estimate the OLS regression parameter vector. How does your result compare to the result from previous part ? Note that you have to set the tuning parameter appropriately for this method.

	lm fit	Gradient Descent
Training data	0.510114	0.510113
test data	0.504933	0.505274

Figure 2: R square result for simple linear regression model.

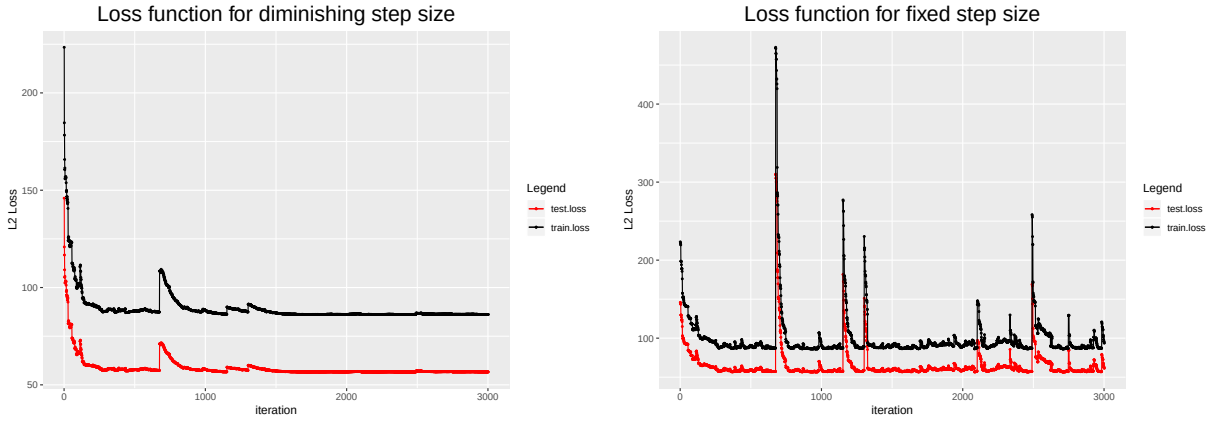
	lm fit	Gradient Descent
Training data	0.510114	0.517346
test data	0.504933	0.510861

Figure 3: R square result for linear regression model involving interaction term.

- For the simple linear regression model. We summarise the result in figure 2
- For the regression involving the interaction term. We summarise the result in figure 3

Base on the table above, it seems gradient descent algorithm does a pretty good/similar job as the linear model since their R-square values are very close.

5. Perform all the things above now using `stochastic gradient descent` (with one sample in each iteration). How does your result compare to the result from previous parts ? Note: while running `stochastic gradient descent`, you can sample without replacement and when you run out of samples, just start over. Note that you have to set the tuning parameter appropriately for this method.



(a) **(b)**

Figure 4: We use the L2 loss $||Y - X\hat{\beta}||_2$ as the criterion of loss function, and implement the stochastic gradient descent under two different step-size choosing method. The first one is the diminishing step size, $\eta = \frac{c_1}{t+1}$, and the second one is the fixed step-size $\eta = c_2$. Here we choose $c_1 = 2$ and $c_2 = 0.02$. As for the numbers of iterations, we focus on the loss function until it is stable enough.

- For the simple linear regression model. We summarise the result in the figure 4. As for the choice of the tuning parameters, we just take the best c such that the loss function on both the training and testing data are stable and small enough through objective judgement. Thus, we take $c = 2$ to calculate our tuning parameter, and the final R-square values comparison is presented in the table in figure 5.

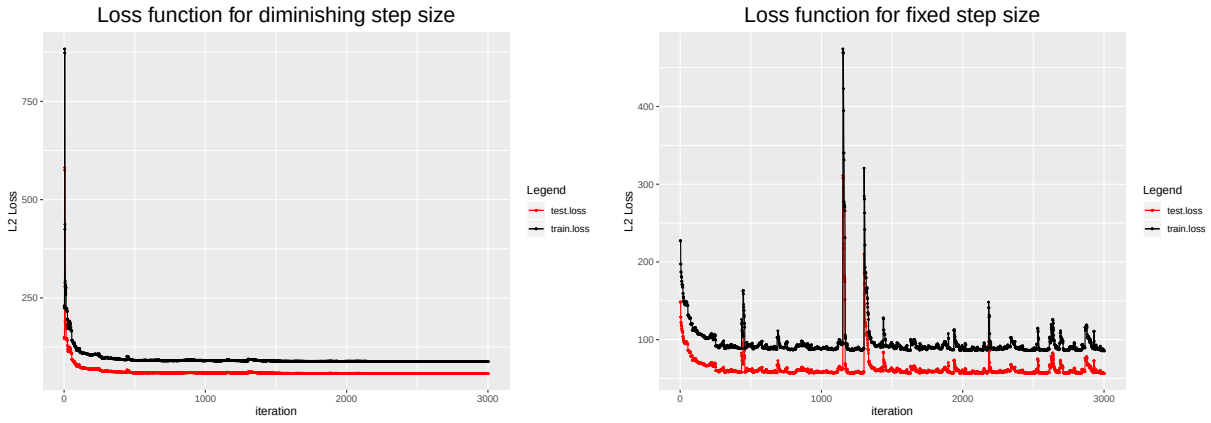
	linear Model	Gradient Descent	SGD with diminishing stepsize	SGD with fixed stepsize
R2 Train	0.51011	0.51011	0.5053	0.50326
R2 Test	0.50493	0.50527	0.50093	0.49805

Figure 5: Summarised result for simple linear regression.

Stochastic gradient descent algorithm also does a pretty good job but not as good as gradient descent because we used randomly selected samples instead of the whole data. When our data is not too large, such as the current data we are using, it is better to perform the gradient descent algorithm.

- For the regression involving the interaction term. We summarise the result in figure 6. We can see the loss function under fixed step size is a little be messy, the fixed step size will oscillate around the optimum after some steps. It is one of the drawbacks of the

fixed step size algorithm. We also summarise the final R-square values comparison in figure 7.



(a) (b)

Figure 6: We use the L2 loss $||Y - X\hat{\beta}||_2$ as the criterion of loss function, and implement the stochastic gradient descent under two different step-size choosing method. The first one is the diminishing step size, $\eta = \frac{c_1}{t+1}$, and the second one is the fixed step-size $\eta = c_2$. Here we choose $c_1 = 2$ and $c_2 = 0.02$. As for the numbers of iterations, we focus on the loss function until it is stable enough.

	linear Model	Gradient Descent	SGD with diminishing stepsize	SGD with fixed stepsize
R2 Train	0.51735	0.51735	0.51367	0.49857
R2 Test	0.51053	0.51086	0.50823	0.49455

Figure 7: Summarised result for simple linear regression model involving interaction term.

3 Question 3

(**15 Points**) Prove Fact 6.1.1 in `OPT.pdf` and solve the recursion to obtain the final result of Theorem 6.1. (**Hint:** You can use induction)

Solution

- **Proof of Fact 6.1.1**
  From the fact of μ-strongly convexity, it follows that

$$f(\theta_2) - f(\theta_1) \geq \nabla f(\theta_1)^T(\theta_2 - \theta_1) + \frac{\mu}{2}||\theta_2 - \theta_1||_2^2$$

$$f(\theta_1) - f(\theta_2) \geq \nabla f(\theta_2)^T(\theta_1 - \theta_2) + \frac{\mu}{2}||\theta_2 - \theta_1||_2^2$$

  Combine these two formulars, we can conclude that

$$\nabla f(\theta_1)^T(\theta_2 - \theta_1) + \frac{\mu}{2}||\theta_2 - \theta_1||_2^2 \leq -\nabla f(\theta_2)^T(\theta_1 - \theta_2) - \frac{\mu}{2}||\theta_2 - \theta_1||_2^2$$

i.e.

$$\mu||\theta_1 - \theta_2||_2^2 \leq (\nabla f(\theta_1) - \nabla f(\theta_2))^T(\theta_1 - \theta_2)$$

- **Recursion in Theorem 6.1**
  Assume: c_0, c satisfy:

$$\frac{1}{\mu} < c < \frac{(t+1)\mu}{2M_g L^2};$$

the potential set of c is not empty when t is large enough. Here we choose T_0 as the smallest t which make it nonempty. And we also require

$$c_0 \geq (t+1)\mathbb{E}[||\theta^t - \theta^*||_2^2] \quad \forall t = 1, ..., T_0$$

$$c_0 > \frac{2c\sigma_g^2}{\mu}$$

Note that the choice of c_0, c does not depend on the step t. It only depends on μ, M_g, L, σ_g. We are now in the process of recursion.

  - Obviously for $t = 1, ..., T_0$, we have

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

  - If the t-th step holds true, $\forall t \geq T_0$ i.e. $\mathbb{E}[||\theta^{(t)} - \theta^*||_2^2] \leq \frac{c_0}{t+1}$ for the t+1 th step, we have

$$\begin{aligned}
\mathbb{E}[||\theta^{(t+1)} - \theta^*||_2^2] &\leq (1 - 2\mu\eta_t + \eta_t^2 M_g L^2)\mathbb{E}[||\theta^{(t)} - \theta^*||_2^2] + \eta_t^2\sigma_g^2 \\
&\leq \frac{c_0}{t+1} - \frac{2\mu c c_0}{(t+1)^2} + \frac{c^2 M_g L^2 c_0}{(t+1)^3} + \frac{c^2\sigma_g^2}{(t+1)^2} \\
&\leq (\frac{c_0}{t+1} - \frac{\mu c c_0}{(t+2)(t+1)}) - (\frac{\mu c c_0}{2(t+1)^2} - \frac{c^2 M_g L^2 c_0}{(t+1)^3}) - (\frac{\mu c c_0}{2(t+1)^2} - \frac{c^2\sigma_g^2}{(t+1)^2}) \\
&\leq (\frac{c_0}{t+2}) - 0 - 0 \\
&\leq \frac{c_0}{t+2}
\end{aligned}$$

Which complete the recursion process.

4 Pledge

✓ 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 - Han Chen

Team Member 2 - Cassie Xu