CSC2515: Assignment 1

Due on Thursday, October 05, 2017

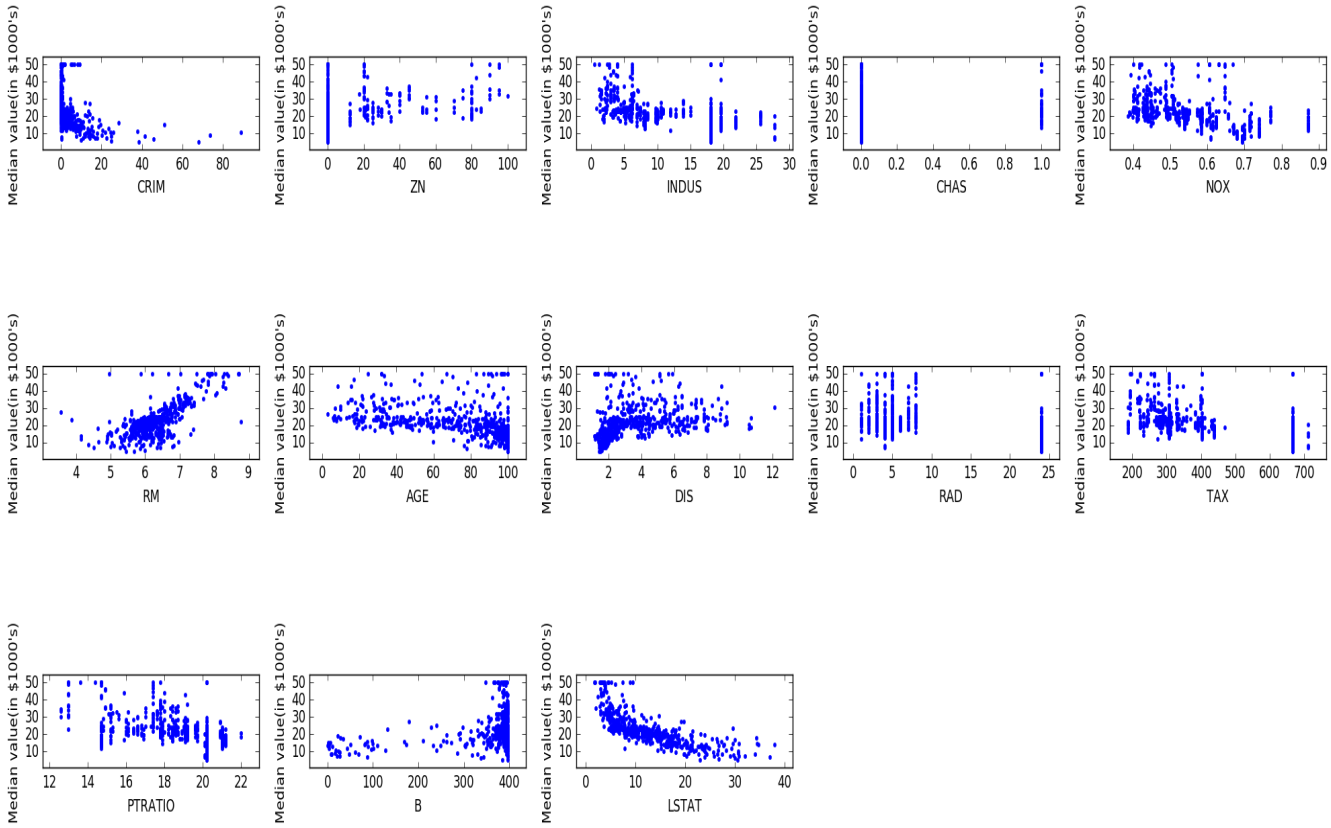
Professor Ethan Fetaya

Zhaoyu Guo (999008069)

Question One - Learning basics of regression in Python

The Boston Housing Dataset

Data Visualization



Dataset Descriptions

The Boston Housing Dataset contains a total of 506 data points. There are 14 attributes in each data point, with 13 features and one target. The target is the median value of owner-occupied homes in $1000's. All 13 features are:

1. **CRIM**: per capita crime rate by town
2. **ZN**: proportion of residential land zoned for lots over 25,000 sq.ft.
3. **INDUS**: proportion of non-retail business acres per town.
4. **CHAS**: Charles River dummy variable (1 if tract bounds river; 0 otherwise)
5. **NOX**: nitric oxides concentration (parts per 10 million)
6. **RM**: average number of rooms per dwelling
7. **AGE**: proportion of owner-occupied units built prior to 1940

8. **DIS**: weighted distances to five Boston employment centres
9. **RAD**: index of accessibility to radial highways
10. **TAX**: full-value property-tax rate per $10,000
11. **PTRATIO**: pupil-teacher ratio by town
12. **B**: square of 1000(Bk - 0.63) where Bk is the proportion of blacks by town
13. **LSTAT**: % lower status of the population

Table 1: Tabulate each feature along with its associated weight

Features	weight
Bias	33.2435
CRIM	-0.1071
ZN	0.0413
INDUS	0.0348
CHAS	1.7371
NOX	-16.4415
RM	4.3269
AGE	-0.0054
DIS	-1.4730
RAD	0.2768
TAX	-0.0132
PTRATIO	-0.9741
B	0.0091
LSTAT	-0.4683

Table 2: Error Measurement Metrics

Error Measurement Metrics	
Mean Square Error	20.68923
Mean Absolute Error	3.07199
R Squared	0.75169

From table 1, we can see the sign of the weight for "INDUS" is positive, with coefficient 0.0348. This implies feature "INDUS"(proportion of non-retail business acres per town) is positively associated with target(median value). The sign of "INDUS" does not match what we expected.The scatter plot looks like the target(median value) should decrease as the value of "INDUS" increases, which should be a negative weight. However, the coefficient of weight for "INDUS" is really small, this suggests the feature "INDUS" does not have a significant effect on the target. From table 2, we can see the value of MSE(mean squared error), MAE(mean absolute error), and R square for test data. MSE measures the average of the squares of the errors. Then we selected two mores error metrics. MAE is the average absolute difference between features and target, which is easier to calculate. In addition, the outlier will have less effect on the error measurement in MAE. R squared measures how close the data are to the fitted regression line. The value of R squared is always between 0% and 100%, and the higher the R squared, the better the model fits the data. Based on the result, we got R squared 0.75169, which means the model does not fits the data very well.

Feature Selection:
According to table 1, we can see that "CHAS", "NOX", "RM" and "DIS"(features with top 4 weights in absolute value) have more significant impact on the target. However, by looking the scatterplot in the data visualization part, we can observe feature "CHAS" is a categorical variable, which can not fit by linear regression. In addition, we can see the scale of each features are different. For example, the value of "NOX" ranging from 0.3 to 0.9, but the value of "TAX" is ranging from 100 to 800. Therefore, we should exclude some features and normalizing data before running the linear regression. Based on the scatterplot and table 1 results, I will conclude "NOX" and "RM" are the most significant features that best predict the price, and feature "NOX" has a negatively association with the house price and "RM" has a positively association with the house price.

Question Two - Locally reweighted regression

Part 1

We want to show the solution to the weighted least square problem

$$w^* = \arg\min \frac{1}{2}\sum_{i=1}^{N} a^{(i)}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)})^2 + \frac{\lambda}{2}\|\mathbf{w}\|^2$$

is given by the formula $w^* = (\mathbf{X}^T\mathbf{A}\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{A}\mathbf{y}$, where $\mathbf{X}$ is the design matrix and $\mathbf{A}$ is a diagonal matrix where $\mathbf{A}_{ii} = a^{(i)}$
Solution:
Let $Ł(w) = \frac{1}{2}\sum_{i=1}^{N} a^{(i)}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)})^2 + \frac{\lambda}{2}\|\mathbf{w}\|^2$. Then

$$\begin{aligned}
Ł(w) &= \frac{1}{2}\sum_{i=1}^{N} a^{(i)}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)})^2 + \frac{\lambda}{2}\|\mathbf{w}\|^2 \\
&= \frac{1}{2}[a^{(1)}[y^{(1)} - \mathbf{w}^T\mathbf{x}^{(1)}]^2 + a^{(2)}[y^{(2)} - \mathbf{w}^T\mathbf{x}^{(2)}]^2 + ... + a^{(N)}[y^{(N)} - \mathbf{w}^T\mathbf{x}^{(N)}]^2] + \frac{\lambda}{2}\|\mathbf{w}\|^2 \\
&= \frac{1}{2}[(\mathbf{y} - \mathbf{X}\mathbf{w})^T\mathbf{A}(\mathbf{y} - \mathbf{X}\mathbf{w})] + \frac{\lambda}{2}\|\mathbf{w}\|^2 \\
&= \frac{1}{2}[(\mathbf{y}^T\mathbf{A} - \mathbf{w}^T\mathbf{X}^T\mathbf{A})(\mathbf{y} - \mathbf{X}\mathbf{w})] + \frac{\lambda}{2}\|\mathbf{w}\|^2 \\
&= \frac{1}{2}[\mathbf{y}^T\mathbf{A}\mathbf{y} - \mathbf{y}^T\mathbf{A}\mathbf{X}\mathbf{w} - \mathbf{w}^T\mathbf{X}^T\mathbf{A}\mathbf{y} + \mathbf{w}^T\mathbf{X}^T\mathbf{A}\mathbf{X}\mathbf{w}] + \frac{\lambda}{2}\|\mathbf{w}\|^2 \\
&= \frac{1}{2}[\mathbf{y}^T\mathbf{A}\mathbf{y} + \mathbf{w}^T\mathbf{X}^T\mathbf{A}\mathbf{X}\mathbf{w} - \mathbf{w}^T\mathbf{X}^T\mathbf{A}\mathbf{y} - (\mathbf{w}^T\mathbf{X}^T\mathbf{A}\mathbf{y})^T] + \frac{\lambda}{2}\|\mathbf{w}\|^2 \\
&= \frac{1}{2}[\mathbf{y}^T\mathbf{A}\mathbf{y} + \mathbf{w}^T\mathbf{X}^T\mathbf{A}\mathbf{X}\mathbf{w} - 2\mathbf{w}^T\mathbf{X}^T\mathbf{A}\mathbf{y}] + \frac{\lambda}{2}\mathbf{w}^T\mathbf{w}
\end{aligned}$$

Taking the derivative with respect to $\mathbf{w}$, we have $\frac{\partial L}{\partial \mathbf{w}} = \mathbf{X}^T\mathbf{A}\mathbf{X}\mathbf{w}$ - $\mathbf{X}^T\mathbf{A}\mathbf{y} + \lambda\mathbf{w}$
Then we set the $\frac{\partial L}{\partial \mathbf{w}} = 0$, we will have

$$\begin{aligned}
\mathbf{X}^T\mathbf{A}\mathbf{X}\mathbf{w} - \mathbf{X}^T\mathbf{A}\mathbf{y} + \lambda\mathbf{w} &= 0 \\
\mathbf{X}^T\mathbf{A}\mathbf{X}\mathbf{w} + \lambda\mathbf{w} &= \mathbf{X}^T\mathbf{A}\mathbf{y} \\
(\mathbf{X}^T\mathbf{A}\mathbf{X} + \lambda\mathbf{I})\mathbf{w} &= \mathbf{X}^T\mathbf{A}\mathbf{y} \\
(\mathbf{X}^T\mathbf{A}\mathbf{X} + \lambda\mathbf{I})^{-1}(\mathbf{X}^T\mathbf{A}\mathbf{X} + \lambda\mathbf{I})\mathbf{w} &= (\mathbf{X}^T\mathbf{A}\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{A}\mathbf{y} \\
\mathbf{w}^* &= (\mathbf{X}^T\mathbf{A}\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{A}\mathbf{y}
\end{aligned}$$

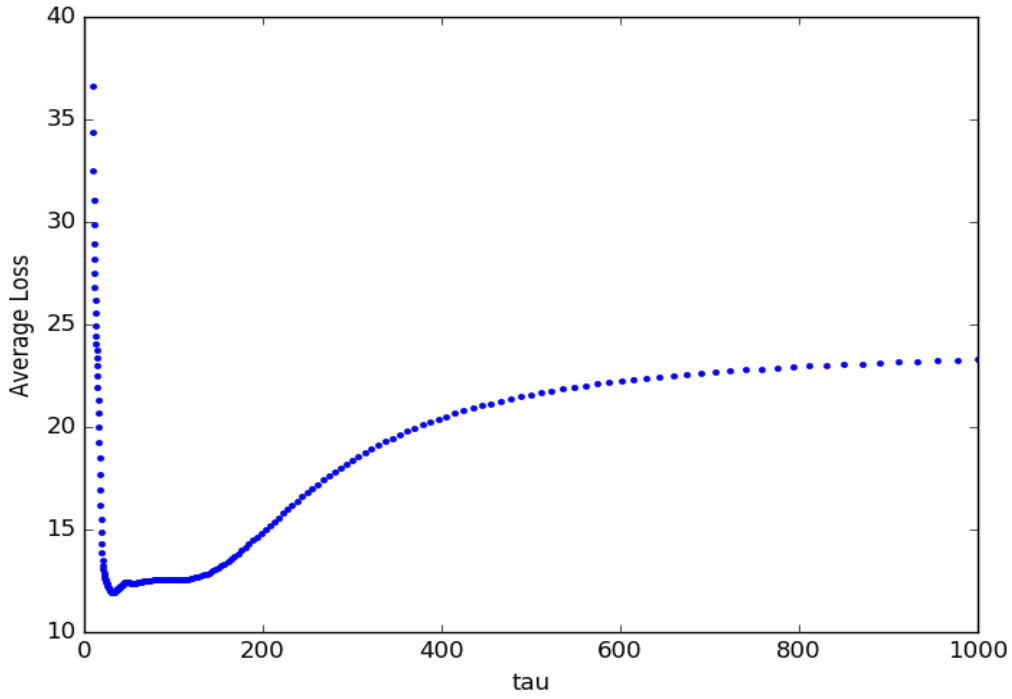
Part 2

Solution:
Please see **q2.py** for the implementation.
In the implementation, I applied the following equivalence to calculate matrix A:

$$\frac{exp(A_i)}{\sum_j exp(A_j)} = exp(log(\frac{exp(A_i)}{\sum_j exp(A_j)})) = exp(A_i - log\sum_j exp(A_j))$$

Part 3

Solution:
Please see **q2.py** for the implementation.



Part 4

According to the plot,we can observe that average loss could increase to infinity when τ approaches to zero. However, when τ increases, the average loss firstly go down(with minimum average loss achieves 11.9415) and then go up, and when τ greater than 800, the average loss seems converge to a certain value between 20 and 25. Thus, we can conclude the average loss will converge when τ approaches to infinity.

Question Three - Mini-batch SGD Gradient Estimator

Part 1

We want to show

$$\mathrm{E}_I[\frac{1}{m}\sum_{i\in I} a_i] = \frac{1}{n}\sum_{i=1}^{n} a_i$$

Solution:
Given a set $\{a_1, a_2,...,a_n\}$, let's denote $\{Z_1,Z_2,...,Z_n\}$ be identically distributed Bernoulli random variables such that $Z_i = 1$ if a_i is in the random mini-batches I of size m and 0 otherwise. Then, we have that $E(Z_i) = P(Z_i = 1) = \frac{m}{n}$

$$\begin{aligned}
\mathrm{E}_I[\frac{1}{m}\sum_{i\in I} a_i] &= \frac{1}{m}\mathrm{E}_I[\sum_{i\in I} a_i] \\
&= \frac{1}{m}[\mathrm{E}_I[\sum_{i=1}^{n} a_i Z_i]] \\
&= \frac{1}{m}\sum_{i=1}^{n} a_i[\mathrm{E}_I(Z_i)] \\
&= \frac{1}{m}\sum_{i=1}^{n} a_i[\frac{m}{n}] \\
&= \frac{1}{n}\sum_{i=1}^{n} a_i
\end{aligned}$$

Part 2

We want to show

$$\mathrm{E}_I[\nabla Ł_I(\mathbf{x}, y, \theta)] = \nabla Ł(\mathbf{x}, y, \theta)$$

Solution:

$$\begin{aligned}
\mathrm{E}_I[\nabla Ł_I(\mathbf{x}, y, \theta)] &= \mathrm{E}_I[\frac{1}{m}\sum_{i\in I} \nabla l(\mathbf{x}^{(i)}, y^{(i)}, \theta)] \\
&= \frac{1}{n}\sum_{i\in I} \nabla l(\mathbf{x}^{(i)}, y^{(i)}, \theta) && \text{based on the result in part 1} \\
&= \nabla\frac{1}{n}\sum_{i=1}^{n} l(\mathbf{x}^{(i)}, y^{(i)}, \theta) \\
&= \nabla Ł(\mathbf{x}, y, \theta)
\end{aligned}$$

Part 3

Solution:
This result implies the drawn uniformly from a set without replacement is an unbiased estimator so the expected value of a Mini-batch SGD is equal to the true empirical gradient.

Part 4

(a) We want to derive the gradient, $\nabla Ł$ above, for a linear regression model with cost function $l(\mathbf{x}, y, \theta) = (\mathbf{y} - \mathbf{w}^T\mathbf{x})^2$

Solution:

We know $l(\mathbf{x}^{(i)}, y^{(i)}, \theta) = (y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)})^2$ and $\nabla l(\mathbf{x}, \mathbf{y}, \theta) = \nabla \sum_{i=1}^{n} l(\mathbf{x}^{(i)}, y^{(i)}, \theta)$

$$\begin{aligned} l(\mathbf{x}, \mathbf{y}, \theta) &= (\mathbf{y} - \mathbf{w}^T\mathbf{x})^2 \\ &= (\mathbf{y} - \mathbf{x}\mathbf{w})^T(\mathbf{y} - \mathbf{x}\mathbf{w}) \\ &= (\mathbf{y}^T - \mathbf{w}^T\mathbf{x}^T)(\mathbf{y} - \mathbf{x}\mathbf{w}) \\ &= \mathbf{y}^T\mathbf{y} - \mathbf{y}^T\mathbf{x}\mathbf{w} - \mathbf{w}^T\mathbf{x}^T\mathbf{y} + \mathbf{w}^T\mathbf{x}^T\mathbf{x}\mathbf{w} \\ &= \mathbf{y}^T\mathbf{y} + \mathbf{w}^T\mathbf{x}^T\mathbf{x}\mathbf{w} - \mathbf{w}^T\mathbf{x}^T\mathbf{y} - (\mathbf{w}^T\mathbf{x}^T\mathbf{y})^T \\ &= \mathbf{y}^T\mathbf{y} + \mathbf{w}^T\mathbf{x}^T\mathbf{x}\mathbf{w} - 2\mathbf{w}^T\mathbf{x}^T\mathbf{y} \end{aligned}$$

Then, taking partial derivative with respect to $\mathbf{w}$

$$\begin{aligned} \nabla l(\mathbf{x}, \mathbf{y}, \theta) &= 2\mathbf{x}^T\mathbf{x}\mathbf{w} - 2\mathbf{x}^T\mathbf{y} \\ &= 2[\mathbf{x}^T\mathbf{x}\mathbf{w} - \mathbf{x}^T\mathbf{y}] \end{aligned}$$

In addition, we know $\nabla Ł(\mathbf{x}, y, \theta) = \frac{1}{n}\nabla l(\mathbf{x}, y, \theta)$, so

$$\begin{aligned} \nabla Ł(\mathbf{x}, \mathbf{y}, \theta) &= \frac{1}{n}\nabla l(\mathbf{x}, \mathbf{y}, \theta) \\ &= \frac{1}{n}[2[\mathbf{x}^T\mathbf{x}\mathbf{w} - \mathbf{x}^T\mathbf{y}]] \end{aligned}$$

(b) Please see **q3.py** for the implementation.

Part 5

Solution:

Based on our experiment, we can conclude that cosine similarity is a more meaningful measure in this case.

Table 3: Comparing squared distance metric and cosine similarity

Measure	
Square Distance Metric	1290653.8021
Cosine Similarity	0.999997516156

When we run **q3.py** several times, we got almost same numbers for cosine similarity. However, the squared distance metric changed a lot. I think the reason for this because the cosine measures the similarity between vector's direction instead of magnitude, and this will be more important when we want to optimize weights $\mathbf{w}$.

Part 6

Solution:
Please see **q3.py** for the implementation.

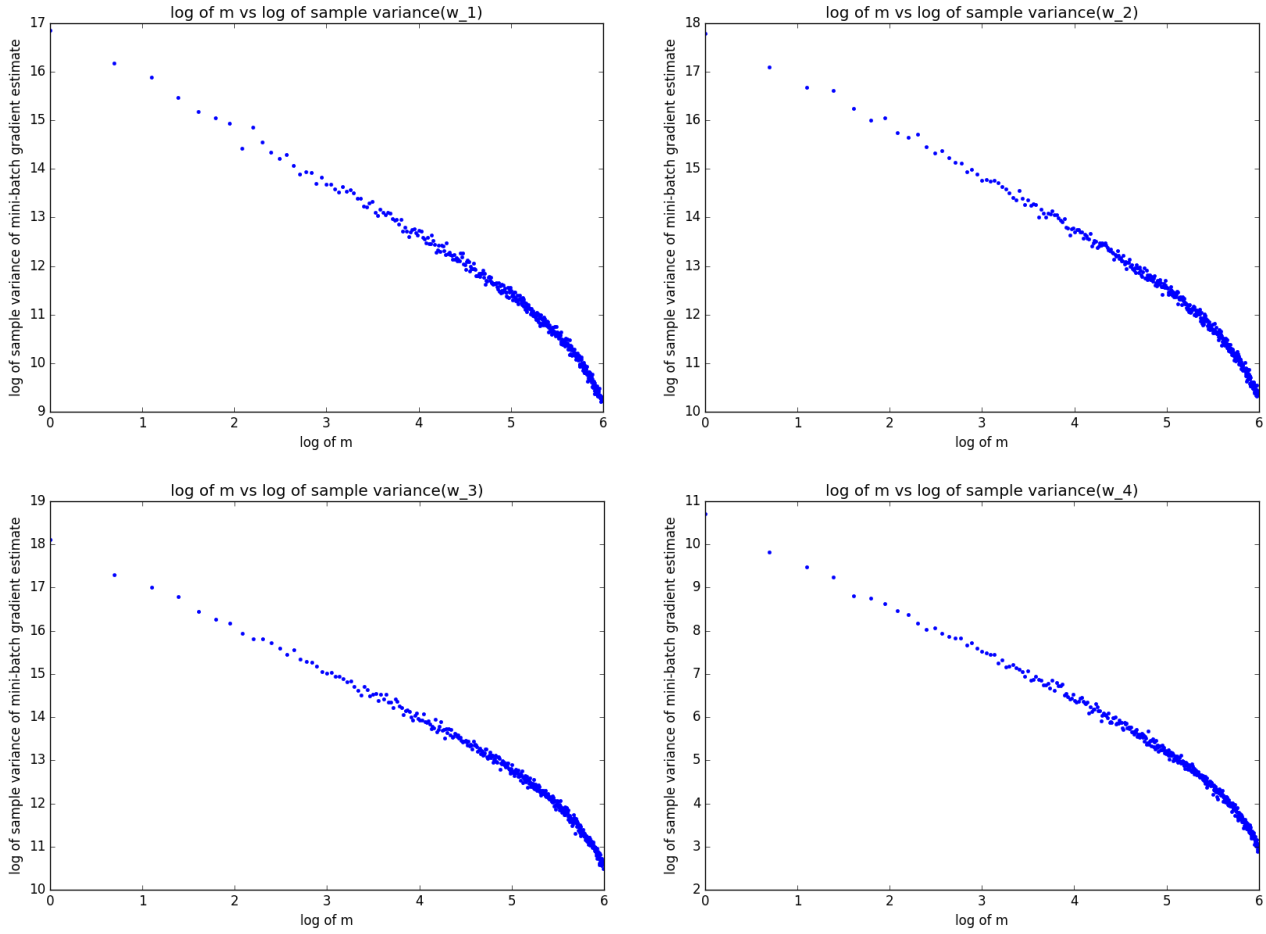


Figure 1: Compare the log of sample variance against log of m for first 4 features

We can see that all of those 4 plots shared same pattern. When log of m increases, the sample variance of the mini-batch gradient estimate will decrease.