CSC2515 HW2

Bingzhang Zhu
(Code Collaborators: Zhimao Lin)

October 25, 2021

1 Bias and Variance Decomposition for the l_2-regularized Mean Estimator

1.1 (a)

$$\begin{aligned}\frac{1}{n}\sum_{i=1}^{n}|Y_i - m|^2 &= E_D[|Y - m|^2] \\ &= E_D[|Y - E_D[Y] + E_D[Y] - m|^2] \\ &= E_D[|Y - E_D[Y]|^2] + E_D[|E_D[Y] - m|^2] + 2E_D[(Y - E_D[Y])(E_D[Y] - m)]^2 \\ &= E_D[|Y - E_D[Y]|^2] + E_D[|E_D[Y] - m|^2] + 2E_D[(Y - E_D[Y])(E_D[Y] - m)]\end{aligned} \tag{1}$$

As $E_D[Y]$ is not a random variable but a scalar, we have:

$$E_D[|E_D[Y] - m|^2] = |E_D[Y] - m|^2$$

$$\begin{aligned}2E_D[(Y - E_D[Y])(E_D[Y] - m)] &= 2(E_D[Y] - m)E_D[(Y - E_D[Y])] \\ &= 2(E_D[Y] - m)(E_D[Y] - E_D[Y]) \\ &= 0\end{aligned} \tag{2}$$

Therefore, we could know that:

$$\begin{aligned}\arg\min_{m\in R}\frac{1}{n}\sum_{i=1}^{n}|Y_i - m|^2 &= \arg\min_{m\in R} E_D[|Y - E_D[Y]|^2] + |E_D[Y] - m|^2 \\ &= \arg\min_{m\in R}|E_D[Y] - m|^2\end{aligned} \tag{3}$$

As a result, we have $m = E_D[Y] = \frac{1}{n}\sum_{i=1}^{n} Y_i = h_{avg}(D)$.

1.2 (b)

Bias:

$$\begin{aligned}|E_D[h_{avg}(D)] - \mu|^2 &= |E_D[\frac{1}{n}\sum_{i=1}^{n} Y_i] - \mu|^2 \\ &= |\frac{1}{n}\sum_{i=1}^{n} E_D[Y_i] - \mu|^2 \\ &= |\frac{1}{n}\sum_{i=1}^{n}\mu - \mu|^2 \\ &= |\mu - \mu|^2 \\ &= 0\end{aligned} \tag{4}$$

Variance:

$$\begin{aligned}
E_D[|h_{avg}(D) - E_D[h_{avg}(D)]|^2] &= E_D[|\frac{1}{n}\sum_{i=1}^{n} Y_i - E_D[\frac{1}{n}\sum_{i=1}^{n} Y_i]|^2] \\
&= E_D[|\frac{1}{n}\sum_{i=1}^{n} Y_i - \frac{1}{n}\sum_{i=1}^{n} E_D[Y_i]|^2] \\
&= E_D[|\frac{1}{n}\sum_{i=1}^{n} Y_i - \frac{1}{n}\sum_{i=1}^{n} \mu|^2] \\
&= E_D[|\frac{1}{n}\sum_{i=1}^{n} (Y_i - \mu)|^2] \\
&= \frac{1}{n^2}\sum_{i=1}^{n} E_D[|(Y_i - \mu)|^2] \\
&= \frac{1}{n^2}\sum_{i=1}^{n} Var_D[Y_i] \\
&= \frac{1}{n^2}\sum_{i=1}^{n} \sigma^2 \\
&= \frac{\sigma^2}{n}
\end{aligned} \tag{5}$$

1.3 (c)

In this question, I assume m is positive. According to quetsion 1(a), we have:

$$\arg\min_{m\in R} \frac{1}{n}\sum_{i=1}^{n} |Y_i - m|^2 = \arg\min_{m\in R} |E_D[Y] - m|^2 \tag{6}$$

And thus, we have:

$$\begin{aligned}
\arg\min_{m\in R} \frac{1}{n}\sum_{i=1}^{n} |Y_i - m|^2 + \lambda m^2 &= \arg\min_{m\in R} |E_D[Y] - m|^2 + \lambda m^2 \\
&= \arg\min_{m\in R} E_D[Y]^2 - 2mE_D[Y] + (1+\lambda)m^2 \\
&= \arg\min_{m\in R} (1+\lambda)m^2 - 2mE_D[Y]
\end{aligned} \tag{7}$$

Let $f(m) = (1+\lambda)m^2 - 2mE_D[Y]$, when $\frac{df}{dm} = 2(1+\lambda)m - 2E_D[Y] = 0$, we get the minimum of $f(m)$. Therefore, we get $m = \frac{E_D[Y]}{1+\lambda} = \frac{1}{n(1+\lambda)}\sum_{i=1}^{n} Y_i = h_\lambda(D)$

1.4 (d)

Bias:

$$\begin{aligned}
|E_D[h_\lambda(D)] - \mu|^2 &= |E_D[\frac{1}{n(1+\lambda)}\sum_{i=1}^{n} Y_i] - \mu|^2 \\
&= |\frac{1}{n(1+\lambda)}\sum_{i=1}^{n} E_D[Y_i] - \mu|^2 \\
&= |\frac{1}{n(1+\lambda)}\sum_{i=1}^{n} \mu - \mu|^2 \\
&= |\frac{\mu}{1+\lambda} - \mu|^2 \\
&= |\frac{\lambda\mu}{1+\lambda}|^2
\end{aligned} \tag{8}$$

Variance:

$$
\begin{aligned}
E_D[|h_\lambda(D) - E_D[h_{avg}(D)]|^2] &= E_D[|\frac{1}{n(1+\lambda)}\sum_{i=1}^{n} Y_i - E_D[\frac{1}{n(1+\lambda)}\sum_{i=1}^{n} Y_i]|^2] \\
&= E_D[|\frac{1}{n(1+\lambda)}\sum_{i=1}^{n} Y_i - \frac{1}{n(1+\lambda)}\sum_{i=1}^{n} E_D[Y_i]|^2] \\
&= E_D[|\frac{1}{n(1+\lambda)}\sum_{i=1}^{n} Y_i - \frac{1}{n(1+\lambda)}\sum_{i=1}^{n} \mu|^2] \\
&= E_D[|\frac{1}{n(1+\lambda)}\sum_{i=1}^{n} (Y_i - \mu)|^2] \\
&= \frac{1}{n^2(1+\lambda)^2}\sum_{i=1}^{n} E_D[|(Y_i - \mu)|^2] \\
&= \frac{1}{n^2(1+\lambda)^2}\sum_{i=1}^{n} Var_D[Y_i] \\
&= \frac{1}{n^2(1+\lambda)^2}\sum_{i=1}^{n} \sigma^2 \\
&= \frac{\sigma^2}{n(1+\lambda)^2}
\end{aligned}
\tag{9}
$$

1.5 (e)

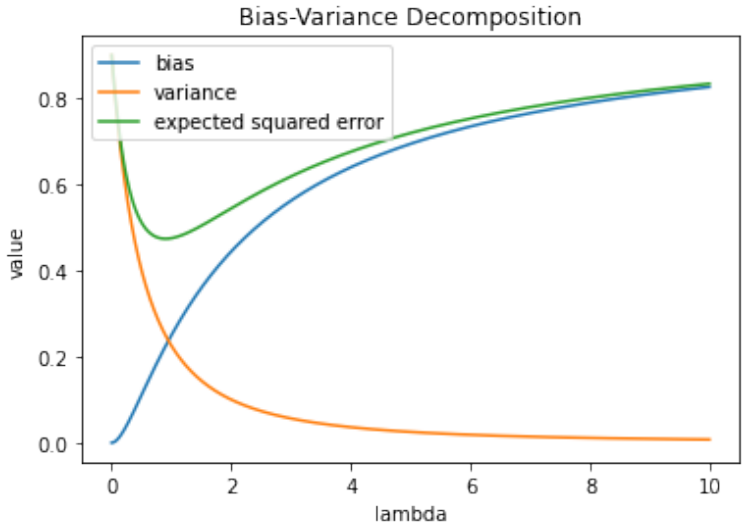

Figure 1: Bias-Variance Decomposition

1.6 (f)

(1) Both Bias and Variance contribute to the expected squared error and there is a trade-off between Bias and Variance. As lambda increases, the value of bias increase while the value of variance decrease. They cannot increase or decrease at the same time, which also means we cannot increase the accuracy of the model and decrease the chance of over-fitting at the same time.
(2) As lambda increase, the speed of Bias increasing and Variance decreasing become slower and slower.
(3) As a result, as lambda increase to the positive infinity, the variance is coming to μ^2, while the variance is coming close to 0.

(4) As lambda increase, the expected squared error decreases at first and then increase.
(5) The expected squared error achieve its minimum value when the line of bias and variance come across (the value of bias is equal to the value of variance).

2 Learning the basics of regression in Python

2.1 (a)

See q2.py

2.2 (b)

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
count	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000
mean	3.613524	11.363636	11.136779	0.069170	0.554695	6.284634	68.574901	3.795043	9.549407	408.237154	18.455534	356.674032	12.653063
std	8.601545	23.322453	6.860353	0.253994	0.115878	0.702617	28.148861	2.105710	8.707259	168.537116	2.164946	91.294864	7.141062
min	0.006320	0.000000	0.460000	0.000000	0.385000	3.561000	2.900000	1.129600	1.000000	187.000000	12.600000	0.320000	1.730000
25%	0.082045	0.000000	5.190000	0.000000	0.449000	5.885500	45.025000	2.100175	4.000000	279.000000	17.400000	375.377500	6.950000
50%	0.256510	0.000000	9.690000	0.000000	0.538000	6.208500	77.500000	3.207450	5.000000	330.000000	19.050000	391.440000	11.360000
75%	3.677083	12.500000	18.100000	0.000000	0.624000	6.623500	94.075000	5.188425	24.000000	666.000000	20.200000	396.225000	16.955000
max	88.976200	100.000000	27.740000	1.000000	0.871000	8.780000	100.000000	12.126500	24.000000	711.000000	22.000000	396.900000	37.970000

Figure 2: Descriptive statistics of the features of Boston data set

This data set includes 506 data points and 14 dimensions (13 features and a target). The descriptive statistics of the features of the Boston data set is shown as Figure 2. Besides, we know the mean, max, min and standard deviation of the target (MEDV) is 22.533, 50.0, 5.0, 9.188 accordingly.

CRIM - per capita crime rate by town
ZN - proportion of residential land zoned for lots over 25,000 sq.ft.
INDUS - proportion of non-retail business acres per town.
CHAS - Charles River dummy variable (1 if tract bounds river; 0 otherwise)
NOX - nitric oxides concentration (parts per 10 million)
RM - average number of rooms per dwelling
AGE - proportion of owner-occupied units built prior to 1940
DIS - weighted distances to five Boston employment centres
RAD - index of accessibility to radial highways
TAX - full-value property-tax rate per $10,000
PTRATIO - pupil-teacher ratio by town
B - $1000(Bk - 0.63)^2$ where Bk is the proportion of blacks by town
LSTAT - % lower status of the population
MEDV - Median value of owner-occupied homes in $1000's

2.3 (c)

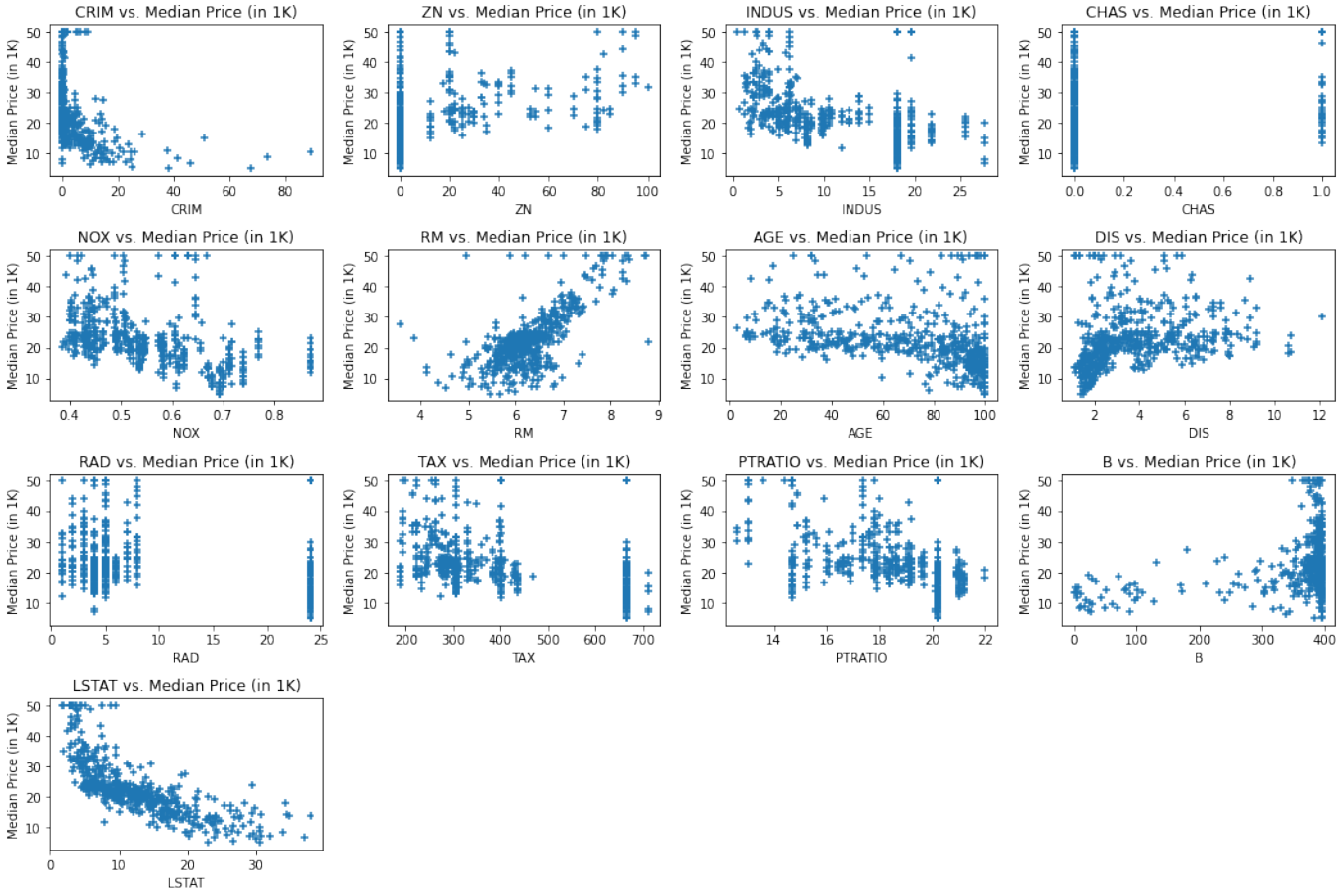

Figure 3: Visualization

2.4 (d)

See q2.py

2.5 (e)

CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
-0.099	0.061	0.059	2.440	-21.470	2.796	0.004	-1.516	0.308	-0.011	-1.005	0.006	-0.569

Figure 4: Each feature along with its associated weight

Besides, the intercept is 46.396493871823864.
The sign of the weight means in the third column ('INDUS') of this table is positive, which means that as INDUS increase, the price will increase either.
It does not fit my expectation. From the visualization in previous question, we can see that there is a tendency that as the INDUS increase the price will decrease, so I thought the sign should be negative.
The reason for this could be our random train set and test set splitting. Besides, this feature might be not important for prediction of price, and thus the sign of the weight of INDUS affect less on the prediction.

2.6 (f)

See q2.py and the MSE is 19.831323672063235

2.7 (g)

The error measurements that I suggests are MAE and R^2. The reason I chose MAE is that MAE is conceptually simpler and also easier to interpret. The reason I chose R^2 is that it is a great metrics for evaluating the regression model and it indicates how much variation of a dependent variable is explained by the independent variable(s) in a regression model.
Mean absolute error is: 3.3446655035987476
R^2 score is: 0.7836295385076281

2.8 (h)

I believe the most significant feature is NOX and then CHAS and RM, because the weight of NOX is the furthest away from 0, and then CHAS and RM. As the weights represent how much the target will change when the feature change one unit, we know that the bigger its absolute value is the larger it could cause to the change of target. As a result, the top three big absolute value of weight are NOX, CHAS and RM, which means they are the most significant feature.
However, if we consider about the dimensionless, things could be different.

3 Locally weighted regression

3.1 (a)

From the question, we know that:

$$\arg\min \frac{1}{2}\sum_{i=1}^{N} a^{(i)}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)})^2 = \arg\min \frac{1}{2}(\mathbf{y} - \mathbf{w}^T\mathbf{X})\mathbf{A}(\mathbf{y} - \mathbf{w}^T\mathbf{X})^T \tag{10}$$

Since $\frac{1}{2}$ is a constant, when we minimize a function, multiply or divide the cost function by a non-zero constant doesn't affect the minimization result, thus in this case, we omit this constant term. For convenience, our cost function becomes:

$$\begin{aligned} J(\mathbf{w}) &= (\mathbf{y} - \mathbf{w}^T\mathbf{X})\mathbf{A}(\mathbf{y} - \mathbf{w}^T\mathbf{X})^T \\ &= \mathbf{y}\mathbf{A}\mathbf{y}^T - \mathbf{y}\mathbf{A}\mathbf{X}^T\mathbf{w} - \mathbf{w}^T\mathbf{X}\mathbf{A}\mathbf{y}^T + \mathbf{w}^T\mathbf{X}\mathbf{A}\mathbf{X}^T\mathbf{w} \end{aligned} \tag{11}$$

$J(\mathbf{w})$ could get the minimum value, when:

$$\begin{aligned} \frac{dJ}{d\mathbf{w}} &= -\mathbf{X}\mathbf{A}\mathbf{y}^T - \mathbf{X}\mathbf{A}\mathbf{y}^T + 2\mathbf{X}\mathbf{A}\mathbf{X}^T\mathbf{w} \\ &= 2\mathbf{X}\mathbf{A}\mathbf{X}^T\mathbf{w} - 2\mathbf{y}^T\mathbf{A}\mathbf{X} \\ &= 0 \end{aligned} \tag{12}$$

As a result, we can get $\mathbf{w}^* = (\mathbf{X}\mathbf{A}\mathbf{X}^T)^{-1}\mathbf{y}^T\mathbf{A}\mathbf{X}$.
This is different from the given formula because I use the different design matrix $\mathbf{X}$, $\mathbf{w}$ and $\mathbf{y}$. Therefore, if I use the transposition formula of the $\mathbf{X}$, $\mathbf{w}$ and $\mathbf{y}$ above. The result would be $\mathbf{w}^* = (\mathbf{X}^T\mathbf{A}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{A}\mathbf{y}$.

3.2 (b)

See q3.py

3.3 (c)

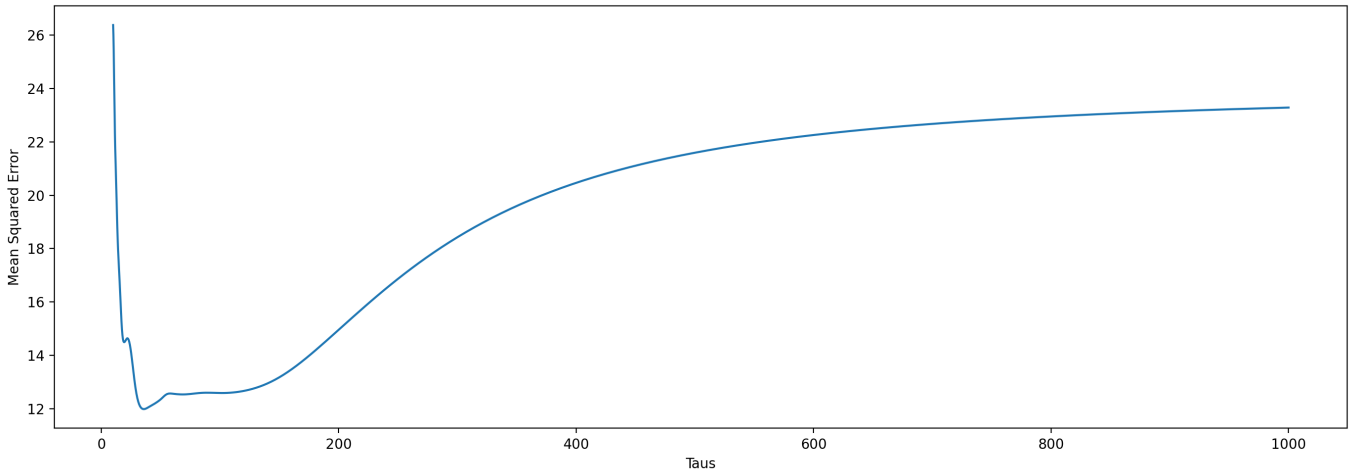

Figure 5: Loss for different τ

3.4 (d)

When $\tau \to \infty$, the weight could be $a^{(i)} \to \frac{1}{n}$, which is equal for each data point. In this way, the locally weighted regression would performs actually the same with the normal linear regression.

When $\tau \to 0$, the weight will depends on the distance between x and $x^{(i)}$. If the x and $x^{(i)}$ are pretty close to each other or they are actually the same, then the weight could be $a^{(i)} \to \frac{1}{n}$; However, under most circumstances, it could be $a^{(i)} \to 0$. In this way, the locally weighted regression would performs bad.

3.5 (e)

Advantages:
(1) It enjoys more flexibility than Linear Regression, so it could fit nonlinear data better.
(2) With the penalty term $\frac{\lambda}{2}||\mathbf{w}||^2$ inside the loss function, it has less chance to over-fit the data than Linear Regression.
Disadvantages:
(1) It is more complex and has more computational cost than Linear Regression.
(2) It could has larger variance than Linear Regression.