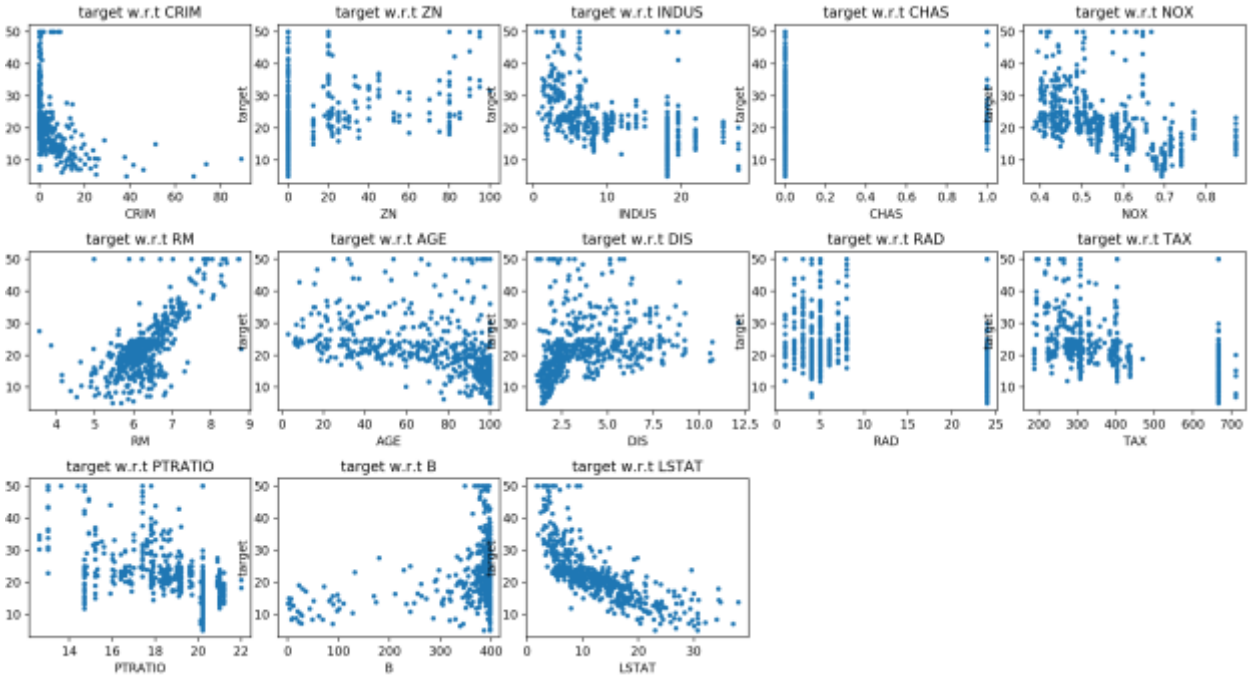
Q1

There are 13 features in the dataset, corresponding to the 13 columns in matrix X. X has 506 rows, which means there are 506 target data points, represented by array y. The target points stand for the house prices in Boston.



The weight of each features (including bias) is shown below:

Bias	23.5727509446
CRIM	-0.122575206824
ZN	0.0303988629728
INDUS	0.0217561213734
CHAS	2.79208018512
NOX	-15.2358481187
RM	5.26010409425
AGE	-0.0106641921922
DIS	-1.27070375914
RAD	0.264409416784
TAX	-0.0115069817816
PTRATIO	-0.918125669879
B	0.0102685908138
LSTAT	-0.391421685711

‘INDUS’ has a positive weight but close to zero, which means that it is positively correlated to the house price. However, in some cases with different random seed, it has a negative weight but still close to zero. Intuitively, the reason of this sign may be that although larger proportion of non-retail business acre may cause some inconvenience of shopping, it may be better choice for people who need to live near to their work place.

Mean squared error: 34.3223041482
Mean absolute error: 3.80364904969
Huber loss: 336.511983064

MSE is not robust to outliers, so MAE and the combination, Huber loss, are necessary to value.

The most significant feature should be ‘RM’, which has the most observable trend in the grid.

Q2

Chengyi Zhang 1001169610 CSC411 A1

Q2

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

$= \frac{1}{2}(y - Xw)^T A (y - Xw) + \frac{\lambda}{2}\|w\|^2$ # A is diagonal where $A_{ii} = a^{(i)}$

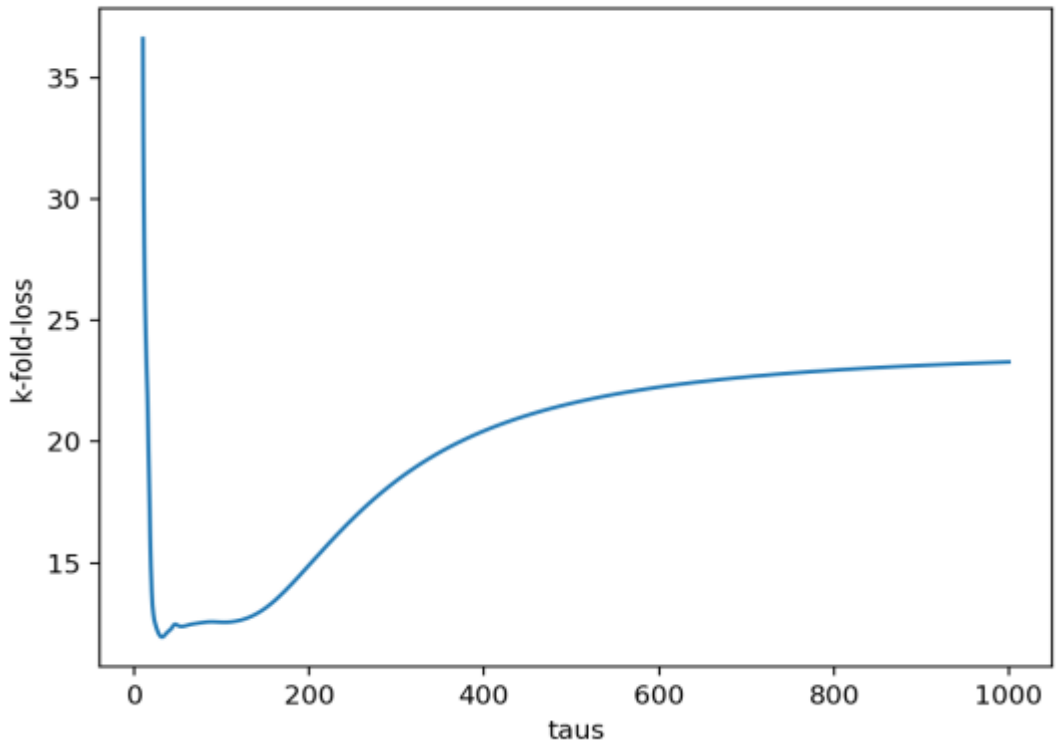
$= \frac{1}{2} y^T A y + \frac{1}{2} w^T X^T A X w - w^T X^T A y + \frac{\lambda}{2}\|w\|_2^2$

derivative $\Rightarrow X^T A X w^* - X^T A y + \lambda I w^* = 0$

$(X^T A X + \lambda I) w^* = X^T A y$

Thus, $w^* = (X^T A X + \lambda I)^{-1} X^T A y$

3.



4. When tau approaches infinity, the loss tends to converge. When tau approaches 0, the loss approaches infinity. The min loss for taus in [10, 1000] is 11.9415251864.

Q3

Chengyi zhang 1001169610 CSC411 A1

Q3

1. $E_I\left[\frac{1}{m}\sum_{i\in I} a_i\right] = \frac{1}{m} E_I\left[\sum_{i\in I} a_i\right]$

$= \frac{1}{m}\sum_{i\in I} E_I[a_i]$

$= E_I[a_i]$

Since I is drawn uniformly without replacement from $\{1,\dots,n\}$, then

$E_I[a_i] = E[a_i] = \frac{1}{n}\sum_{i=1}^{n} a_i$

2. $E_I[\nabla L_I(x,y,\theta)] = E_I\left[\nabla \frac{1}{m}\sum_{i\in I} l\left(x^{(i)}, y^{(i)}, \theta\right)\right]$

$= E_I\left[\frac{1}{m}\sum_{i\in I} \nabla l\left(x^{(i)}, y^{(i)}, \theta\right)\right]$

$= \frac{1}{n}\sum_{i=1}^{n} \nabla l\left(x^{(i)}, y^{(i)}, \theta\right)$ # By $E_I\left[\frac{1}{m}\sum_{i\in I} a_i\right] = \frac{1}{n}\sum_{i=1}^{n} a_i$

$= \nabla \frac{1}{n}\sum_{i=1}^{n} l\left(x^{(i)}, y^{(i)}, \theta\right)$

$= \nabla L(x,y,\theta)$

3. We can use this method to generate an unbiased estimator of the true gradient, which is the basis of optimizing ML algorithms with huge datasets

4. a) $L(x,y,\theta) = \frac{1}{n}\sum_{i=1}^{n} l\left(x^{(i)}, y^{(i)}, \theta\right) = \frac{1}{n}\sum_{i=1}^{n} \left(y^{(i)} - w^T x^{(i)}\right)^2 = \frac{1}{n}\|y - Xw\|^2$

$= \frac{1}{n}\left(y^T y + w^T X^T X w - 2 w^T X^T y\right)$

$\nabla L(x,y,\theta) = \frac{1}{n}\left(2X^T X w - 2X^T y\right)$ # $\theta = w$

5.
cosine similarity: 0.9999991318930902
squared distance: 770.14454296346992

In this case, cosine similarity is more meaningful, because this metric shows that two vectors are ‘in the same direction’. On the other hand, though two vectors are located far from each other, this metric cannot express the difference of weights (ratios of elements) in two vectors.

6.