1. a)

To optimize $argmin_{m \subseteq \boldsymbol{R}}\{ \frac{1}{n} \sum_{i=1}^{n} |Y_i - m|^2\}$, we need to take its derivative first:

$$\frac{d}{dm} \frac{1}{n} \sum_{i=1}^{n} |Y_i - m|^2 = \frac{1}{n} \sum_{i=1}^{n} \frac{d}{dm} |Y_i - m|^2$$

$$= \frac{1}{n} \sum_{i=1}^{n} -2(Y_i - m)$$

Setting the derivative to 0 and solve for *m* will give its extrema:

$$\frac{1}{n} \sum_{i=1}^{n} -2(Y_i - m) = 0$$

$$\Rightarrow \frac{1}{n} \sum_{i=1}^{n} (Y_i - m) = 0$$

$$\Rightarrow \frac{1}{n} \sum_{i=1}^{n} (Y_i) - \frac{1}{n} \sum_{i=1}^{n} m = 0$$

$$\Rightarrow \frac{1}{n} \sum_{i=1}^{n} Y_i = \frac{1}{n} \sum_{i=1}^{n} m$$

$$\Rightarrow m = \frac{1}{n} \sum_{i=1}^{n} Y_i$$

To show the solution we find is the global minimum, we need to take the second derivative of the original function and show it is greater than 0:

$$\frac{d}{dm}\frac{1}{n} \sum_{i=1}^{n} -2(Y_i - m) = \frac{d}{dm}\frac{1}{n} \sum_{i=1}^{n} 2m - 2Y_i$$

$$= \frac{1}{n} \sum_{i=1}^{n} 2$$

$$= 2$$

$$> 0$$

Therefore, $h_{avg} = \frac{1}{n} \sum_{i=1}^{n} Y_i$ is the solution of the optimization problem.

b)

Bias: $|E[h(D)] - \mu|^2 = |E[\frac{1}{n} \sum_{i=1}^{n} Y_i] - \mu|^2$

$$= |(\frac{1}{n} \sum_{i=1}^{n} E[Y_i]) - \mu|^2$$

$$= |(\frac{1}{n} \sum_{i=1}^{n} \mu) - \mu|^2$$

$$= |\mu - \mu|^2$$

$= 0$

Variance: $E[|h(D) - E[h(D)]|^2] = E[|(\frac{1}{n}\sum_{i=1}^{n} Y_i) - E[\frac{1}{n}\sum_{i=1}^{n} Y_i]|^2]$

$= E[|(\frac{1}{n}\sum_{i=1}^{n} Y_i) - (\frac{1}{n}\sum_{i=1}^{n} E[Y_i])|^2]$

$= E[|\frac{1}{n}\sum_{i=1}^{n} (Y_i - E[Y_i])|^2]$

$= \frac{1}{n^2}\sum_{i=1}^{n} E[|Y_i - \mu|^2]$

$= \frac{1}{n^2}\sum_{i=1}^{n} \sigma^2$

$= \sigma^2 /n$

c)

The derivative of $h_\lambda(D)$ is $\frac{1}{n}\sum_{i=1}^{n} (-2(Y_i - m) + 2\lambda m)$.

Let the derivative of $h_\lambda(D)$ be 0 and solve for *m*:

$\frac{1}{n}\sum_{i=1}^{n} (-2(Y_i - m) + 2\lambda m) = 0$

$\Rightarrow \frac{1}{n}\sum_{i=1}^{n} ((Y_i - m) - \lambda m) = 0$

$\Rightarrow \frac{1}{n}\sum_{i=1}^{n} (Y_i - (\lambda + 1)m) = 0$

$\Rightarrow \frac{1}{n}\sum_{i=1}^{n} Y_i - \frac{1}{n}\sum_{i=1}^{n} (\lambda + 1)m = 0$

$\Rightarrow \frac{1}{n}\sum_{i=1}^{n} (\lambda + 1)m = \frac{1}{n}\sum_{i=1}^{n} Y_i$

$\Rightarrow (\lambda + 1)m = \frac{1}{n}\sum_{i=1}^{n} Y_i$

$\Rightarrow m = \frac{1}{n(\lambda+1)}\sum_{i=1}^{n} Y_i$

d)

Bias: $|E[h_\lambda(D)] - \mu|^2 = |E[\frac{1}{n(\lambda+1)}\sum_{i=1}^{n} Y] - \mu|^2$

$= |(\frac{1}{n(\lambda+1)}\sum_{i=1}^{n} E[Y_i]) - \mu|^2$

$$= |(\frac{1}{n(\lambda+1)} \sum_{i=1}^{n} \mu) - \mu |^2$$

$$= | \frac{\mu}{\lambda+1} - \mu |^2$$

$$= (\frac{\lambda\mu}{\lambda+1})^2$$

Variance: $E[|h_\lambda(D) - E[h_\lambda(D)]|^2] = E[|(\frac{1}{n(\lambda+1)} \sum_{i=1}^{n} Y_i) - E[\frac{1}{n(\lambda+1)} \sum_{i=1}^{n} Y_i]|^2]$

$$= E[|(\frac{1}{n(\lambda+1)} \sum_{i=1}^{n} Y_i) - (\frac{1}{n(\lambda+1)} \sum_{i=1}^{n} E[Y_i])|^2]$$

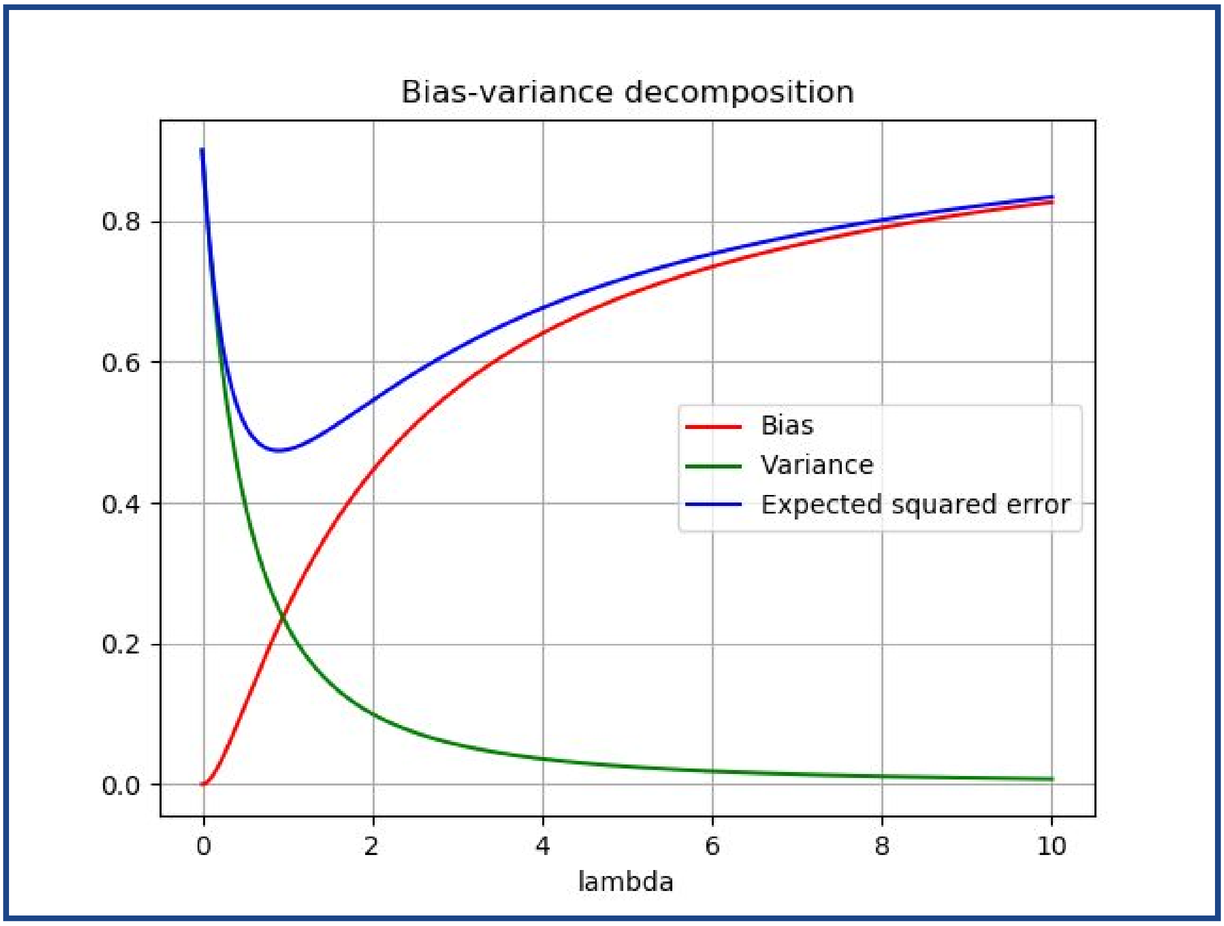
$$= E[| \frac{1}{n(\lambda+1)} \sum_{i=1}^{n} (Y_i - E[Y_i])|^2]$$

$$= \frac{1}{n^2(\lambda+1)^2} \sum_{i=1}^{n} E[|Y_i - \mu |^2]$$

$$= \frac{1}{n^2(\lambda+1)^2} \sum_{i=1}^{n} \sigma^2$$

$$= \frac{\sigma^2}{(\lambda+1)^2 n}$$

e)



f)

When $\lambda = 0$, bias has no effect on the expected squared error and the expected squared error is exactly the same as the variance.

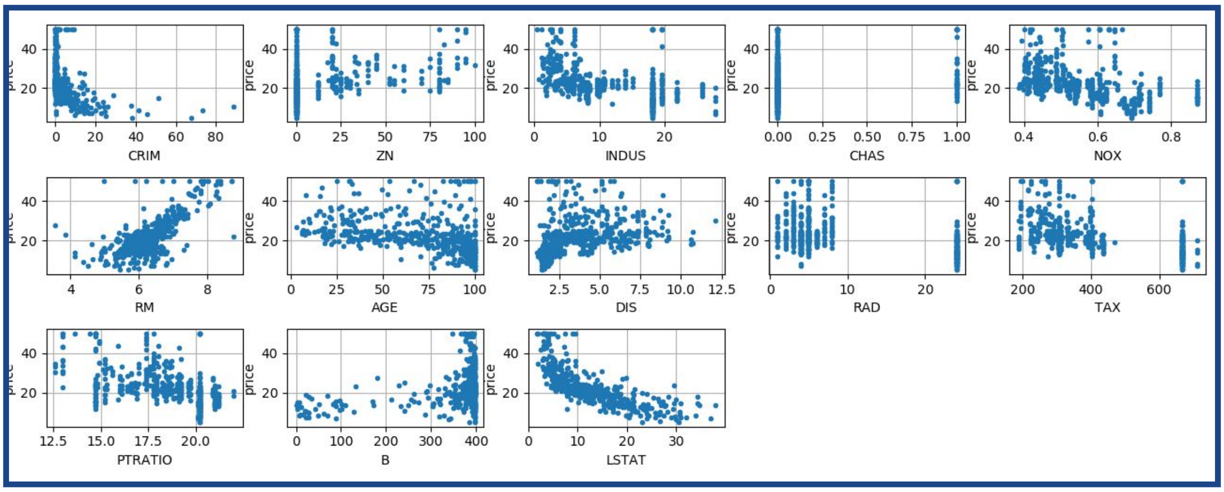
As λ increases, variance tends to decrease and bias increases. After one certain value of λ, bias becomes the dominant part of the expected squared error and variance has little effect on the error.

When $\lambda = \infty$, variance becomes 0 and the expected squared error will be exactly the same the bias, and bias approaches to μ as $\lambda \rightarrow \infty$.

2.

The Boston data set has 506 data points, each data point has 13 dimensions. Each data point represents one house in Boston, and each house has different features. The targets are the value of each house corresponding to its own features.

The following is a graph describing the relationship between each feature and the price of the house, where the x-axis is the quantity of feature and y-axis is the price of houses:

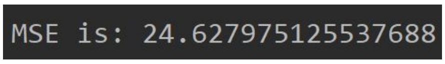


After normalizing the data points using Min-max normalization, the weight (rounded to 3 decimals) of each feature through linear regression is as the following table:

Bias	**CRIM**	**ZN**	**INDUS**	**CHAS**	**NOX**	**RM**
27.013	-7.721	6.093	-0.297	3.903	-8.067	18.907
AGE	**DIS**	**RAD**	**TAX**	**PTRATIO**	**B**	**LSTAT**
-0.965	-17.583	6.501	-5.981	-7.127	3.513	-19.267

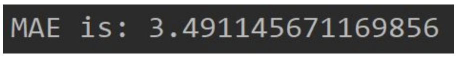
The sign of the feature INDUS is negative, which means it has a negative impact on the price of houses. This is reasonable because as on the graph, the price of houses tends to decrease as the number of INDUS increases.

The Mean Squared Error of the linear regression model is:



```
MSE is: 24.627975125537688
```

The error can also be calculated using the Mean Absolute Error, and the error is:

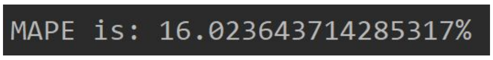


```
MAE is: 3.491145671169856
```

The magnitude of this error measurement is much smaller than MSE because it measures the average of the Manhatten distance between the fitted values and the correct result rather than the squared Euclidean distance which is calculated by MSE. This error measurement metric is good in the Boston data set because the Boston data set has a lot of outliers and MAE is not sensitive to outliers. It produces a much more reasonable result than MSE.

However, MAE only gives a number representing the quantitive error between the fitted values and the real answer. The quantitive error is not as expressive as the percentage error, because percentages give a general sense of how much the fitted values are expected to differ from the real values.

This suggests the use of Mean Absolute Percentage Error as the error measurement metric and the error is:



```
MAPE is: 16.023643714285317%
```

Based on the weights of the linear regression model, it is clear to see that the weights for the feature RM, LSTAT and DIS are very big in magnitude. This implies that RM, LSTAT and DIS are the most significant features.

By the sample graphs, we can see that the feature ‘RM’ has a clear tendency of leading price to a higher number as the number of ‘RM’ increases. The shape of the graph is almost the same as an increasing linear function.

Also, the graph of LSTAT shows that as the number of LSTAT increases, the price has a clear tendency of decreasing. This gives a good sense that LSTAT has a big effect on the price of houses.

DIS also has a great weight because when DIS is low, the prices are more tend to be low and gathered in a group. As DIS increases, the prices are more and more sparse, which says DIS has a huge impact on the prices.

3. a)

Transform the optimization problem into the matrix form:

$$\mathbf{w}^* = argmin_{\mathbf{w}} \frac{1}{2} (\mathbf{y} - \mathbf{Xw})^T\mathbf{A}(\mathbf{y} - \mathbf{Xw})$$

Take the derivative of the inner part of the *argmin* function:

$$\frac{d}{dw}\frac{1}{2} (\mathbf{y} - \mathbf{Xw})^T\mathbf{A}(\mathbf{y} - \mathbf{Xw}) = \frac{d}{dw}\frac{1}{2} (\mathbf{y} - \mathbf{Xw})^T(\mathbf{Ay} - \mathbf{AXw})$$

$$= \frac{d}{dw}\frac{1}{2} (\mathbf{y}^T - (\mathbf{Xw})^T)(\mathbf{Ay} - \mathbf{AXw})$$

$$= \frac{d}{dw}\frac{1}{2} (\mathbf{y}^T - \mathbf{w}^T\mathbf{X}^T)(\mathbf{Ay} - \mathbf{AXw})$$

$$= \frac{d}{dw}\frac{1}{2} (\mathbf{y}^T\mathbf{Ay} - \mathbf{y}^T\mathbf{AXw} - \mathbf{w}^T\mathbf{X}^T\mathbf{Ay} + \mathbf{w}^T\mathbf{X}^T\mathbf{AXw})$$

Note that, $\mathbf{y}^T\mathbf{AXw} = \mathbf{w}^T\mathbf{X}^T\mathbf{Ay}$ because they are the multiplication of same matrices in different sequential order. Therefore, the function above can be written as:

$$\frac{d}{dw}\frac{1}{2} (\mathbf{y}^T\mathbf{Ay} - \mathbf{w}^T\mathbf{X}^T\mathbf{Ay} - \mathbf{w}^T\mathbf{X}^T\mathbf{Ay} + \mathbf{w}^T\mathbf{X}^T\mathbf{AXw})$$

$$= \frac{d}{dw}\frac{1}{2} (\mathbf{y}^T\mathbf{Ay} - 2\mathbf{w}^T\mathbf{X}^T\mathbf{Ay} + \mathbf{w}^T\mathbf{X}^T\mathbf{AXw})$$

$$= ½ (\mathbf{0} - 2\mathbf{X}^T\mathbf{Ay} + \mathbf{2X}^T\mathbf{AXw})$$

$$= \mathbf{X}^T\mathbf{AXw} - \mathbf{X}^T\mathbf{Ay}$$

Set the derivative to 0 to solve for **w**:

$$\mathbf{X}^T\mathbf{AXw} - \mathbf{X}^T\mathbf{Ay} = \mathbf{0}$$

$$\Rightarrow \mathbf{X}^T\mathbf{AXw} = \mathbf{X}^T\mathbf{Ay}$$

$$\Rightarrow (\mathbf{X}^T\mathbf{AX})^{-1}\mathbf{X}^T\mathbf{AXw} = (\mathbf{X}^T\mathbf{AX})^{-1}\mathbf{X}^T\mathbf{Ay}$$

$$\Rightarrow \mathbf{w} = (\mathbf{X}^T\mathbf{AX})^{-1}\mathbf{X}^T\mathbf{Ay}$$

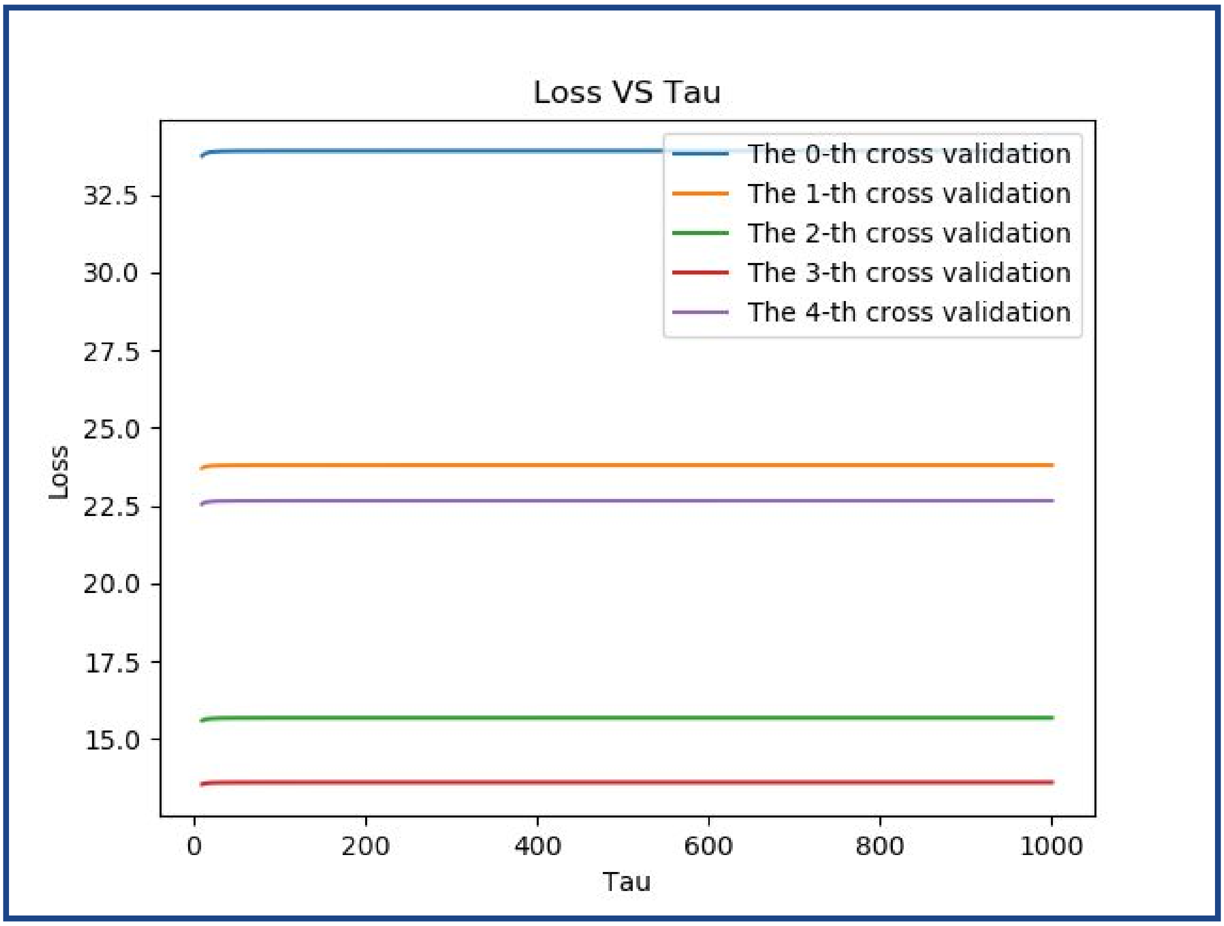
Therefore, $\mathbf{w} = (\mathbf{X}^T\mathbf{AX})^{-1}\mathbf{X}^T\mathbf{Ay}$ is the solution of this optimization problem.

b)

See q3.py.

c)

The graph is generated using normalized data points. The data points are normalized using Min-max normalization.



d)

When $\tau \rightarrow \infty$, the local weight of each training example approaches to $\frac{1}{j}$ where j = number of training samples. This makes the weight of each sample approximately the same, which makes the algorithm more like an unweighted linear regression.

When $\tau \rightarrow 0$, the algorithm tends to memorize each training sample because of the weight of each training sample becomes much more significant. This leads to a decrease in the loss in the test phase because the algorithm is generalizing the characteristics of training samples.

e)

Advantages:

1. Locally weighted linear regression does not need much of feature selections because it generalizes the data points by memorizing their features.
2. This algorithm can do a better job in modelling non-linear data, so it is more flexible than ordinary linear regression.

Disadvantages:

1. Locally weighted linear regression is much more computationally expensive than ordinary linear regression by computing the local weight for each test data point.
2. The model is much harder to interpret than an ordinary linear regression model where one linear function can sufficiently describe the entire model.