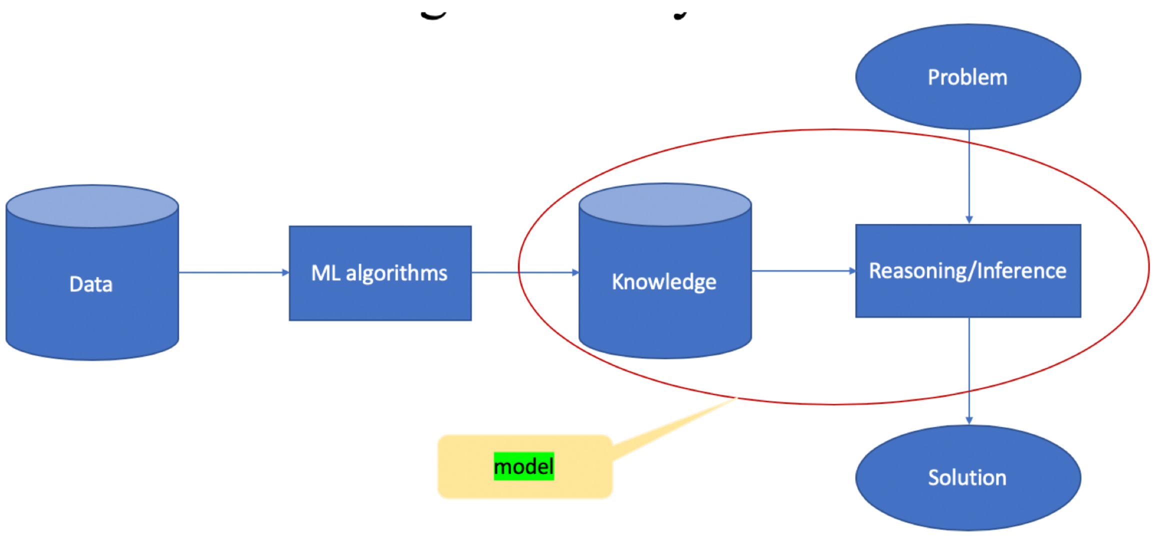
Linear Regression and Gradient Descent Algorithm

LÊ ANH CƯỜNG

TDTU

Recap from the previous lecture

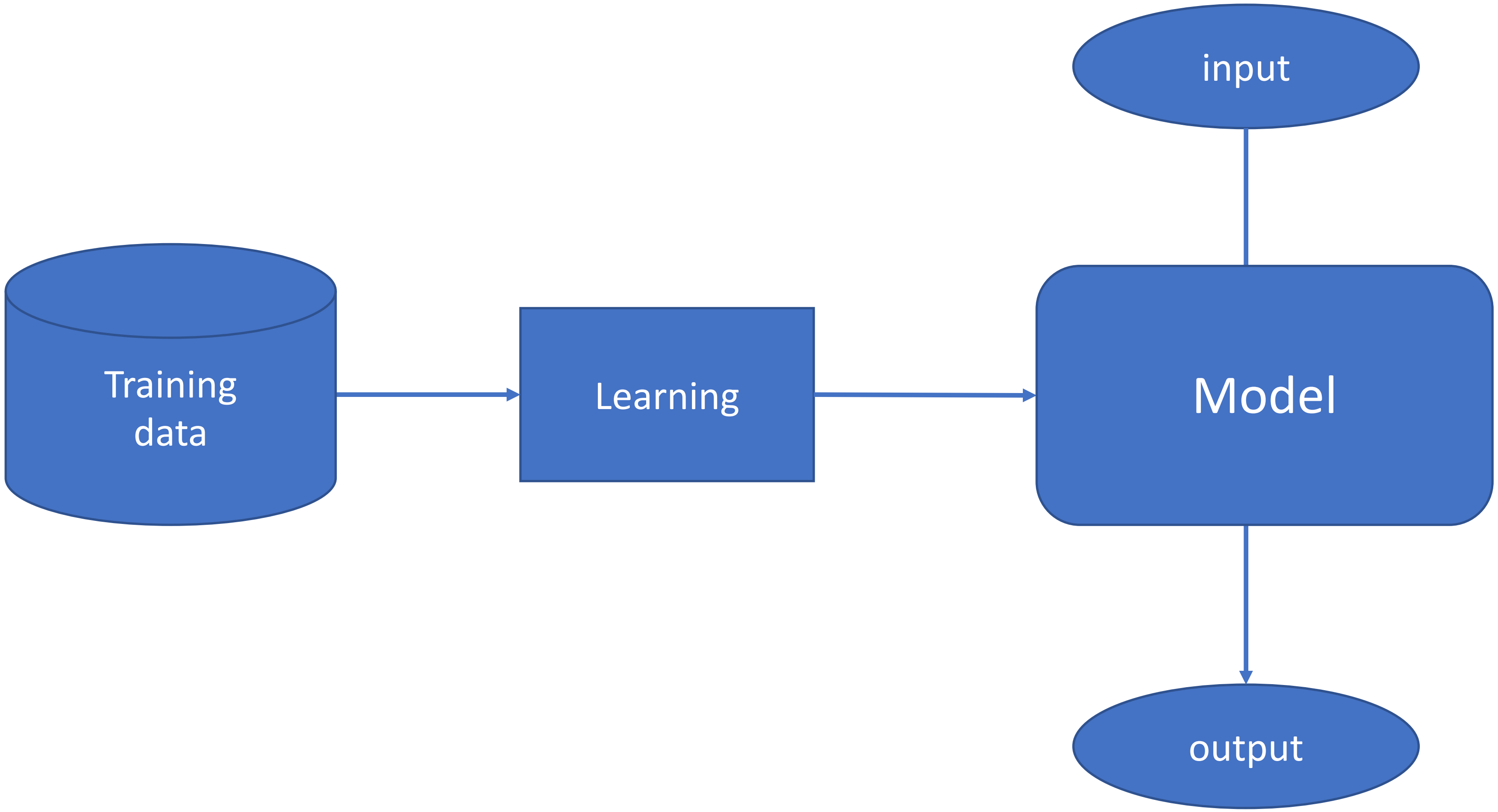
- Role of machine learning and artificial intelligence.
- A general diagram of AI systems.
- Concepts: knowledge, reasoning, inference, model, learning.
- Issues in machine learning.
- Understanding machine learning concepts through examples.



Outline

1. Supervised learning for classification and regression problems
2. Linear Regression (LR) model
3. Training LR by Gradient Descent Algorithm
4. Implementation LR

Supervised learning vs Unsupervised learning



Regression vs Classification

- Classification is about predicting a label and Regression is about predicting a quantity.
- Classification is the problem of predicting a discrete class label output for an example. The predicted label is from a set of predefined labels.
  - For example: predict an email is spam or ham
- Regression is the problem of predicting a continuous quantity output for an example.
  - For example: predict house price from its features (i.e. information).

Regression (Hồi qui)

- From Height, predict Weight?

Height(cm)	Weight(kg)
147	49
150	50
153	51
155	52
158	54
160	56
163	58
165	59

Training data

(x1, y1)

(x2, y2)

.

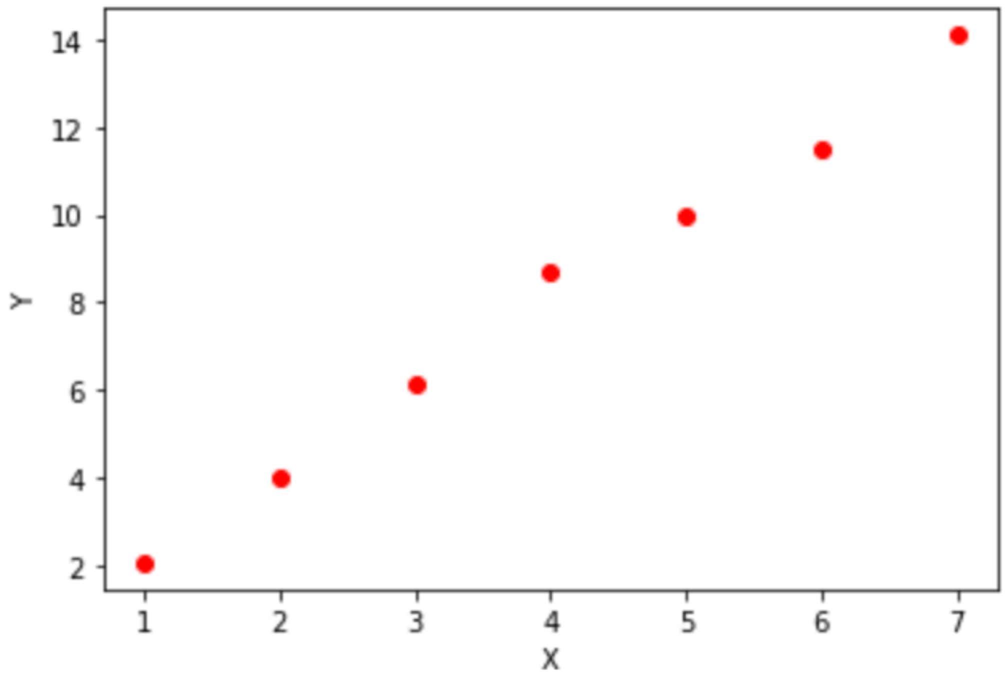
.

.

(xn, yn)

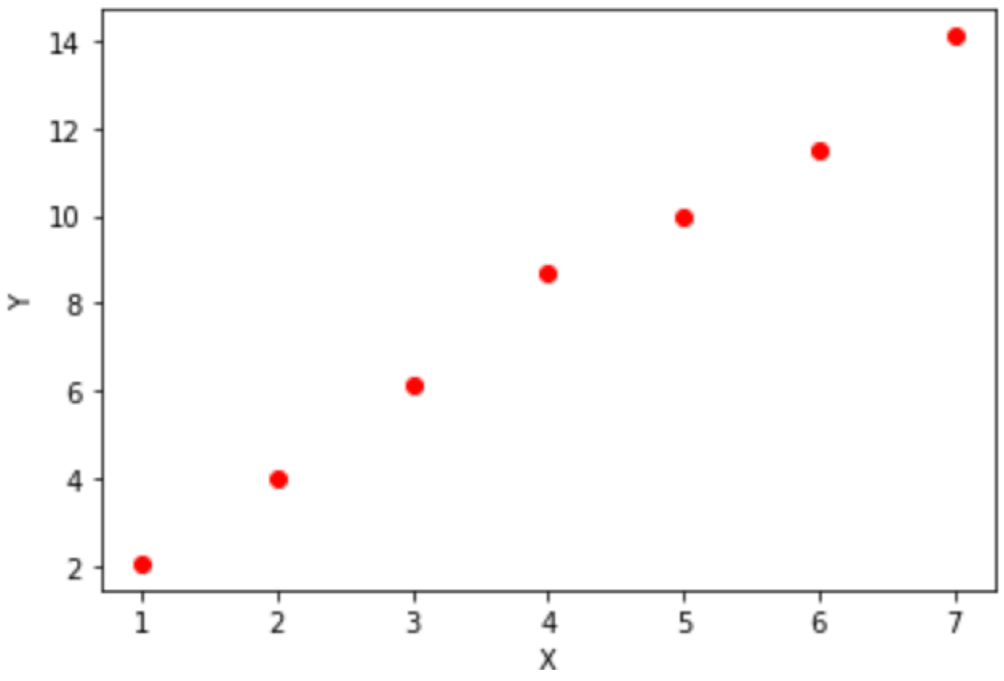
Linear Regression

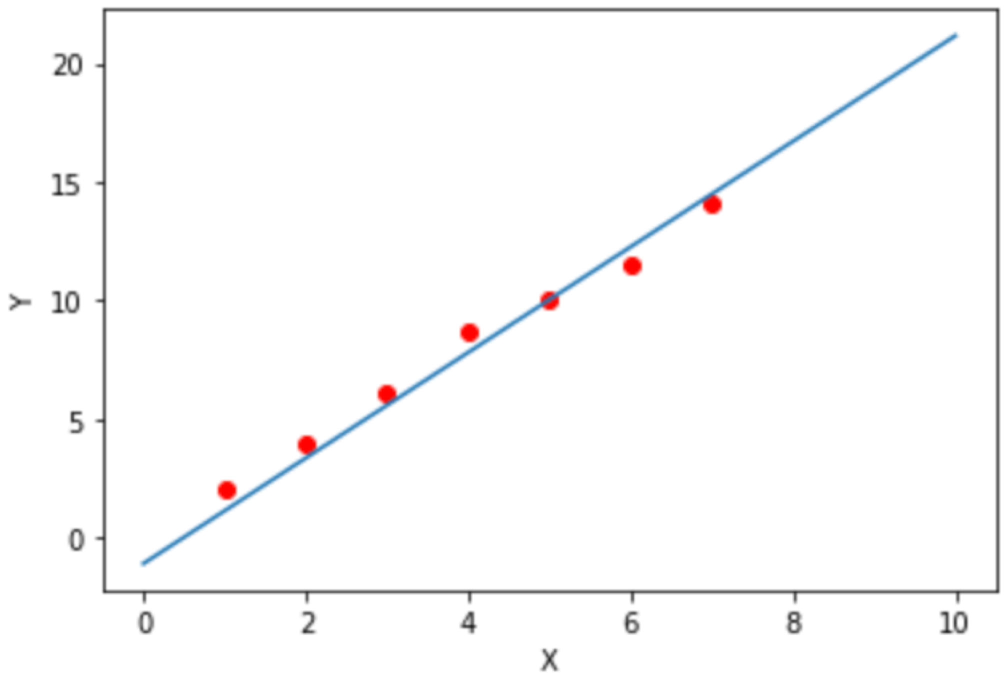
- How is the relationship between Height and Weight?



Linear Regression

Suppose that Weight linearly depends on Height





Simple Linear Regression: one variable

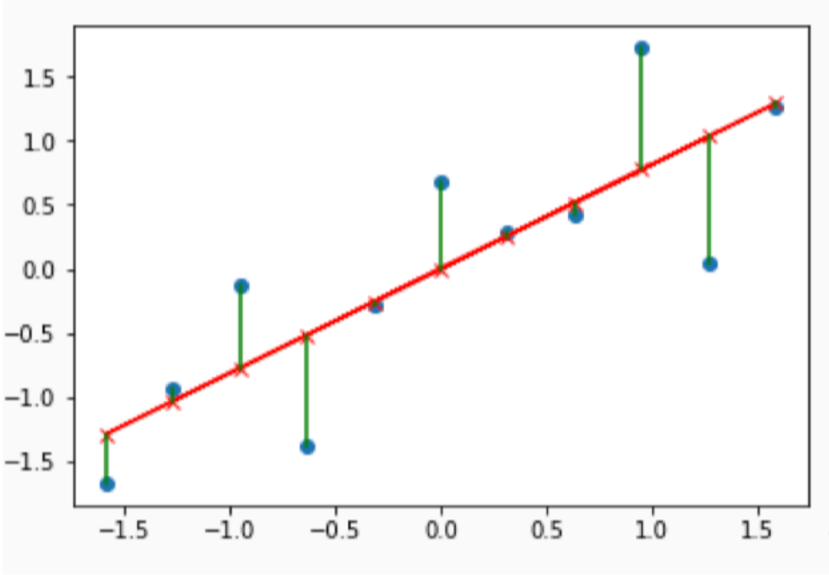
Training data: $(x_1, y_1), (x_2, y_2). . . (x_n, y_n)$

$y = f(\mathrm{x}) = w_0 + w_1 x$

Learn w_0 and w_1

Loss Function

Training data: $(x_1, y_1), (x_2, y_2)... (x_n, y_n)$



$$y = f(x) = w_0 + w_1 x$$

Prediction by the model: x -> f(x)

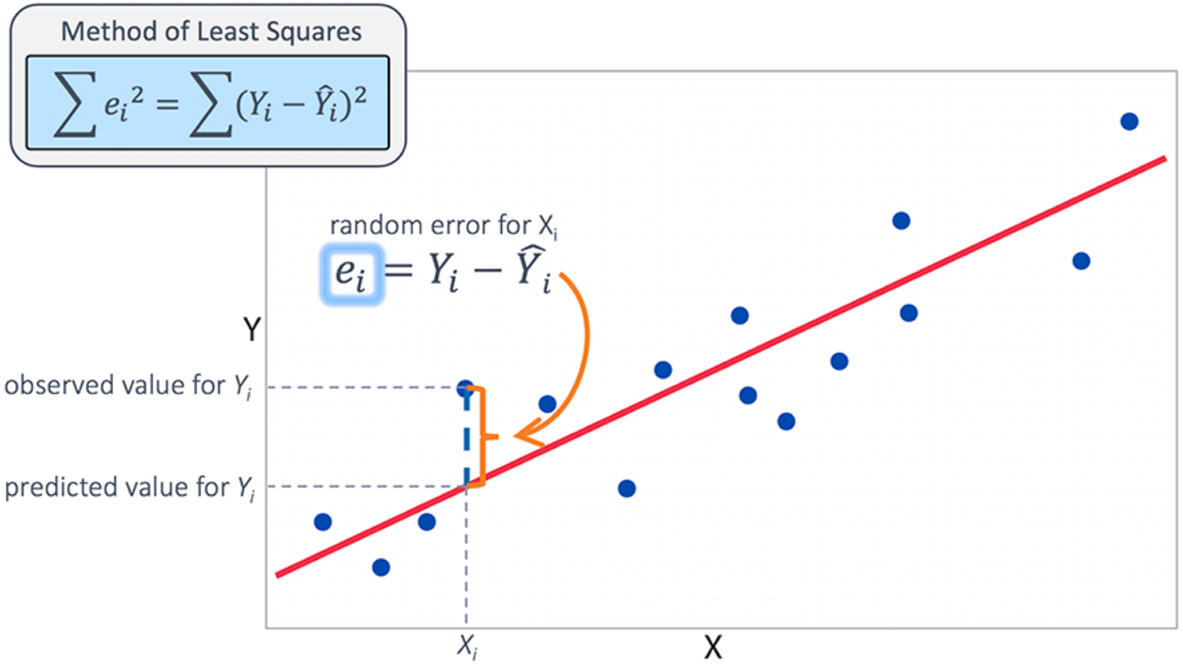
For every training example (x_i, y_i) we compare $f(x_i)$ with y_i

Loss Function

Training data:

$(x_1, y_1), (x_2, y_2)...(x_n, y_n)$

$$y = f(x) = w_0 + w_1 x$$



MSE: Mean Square Error

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

Loss Function

Training data:

$(x_1, y_1), (x_2, y_2)...(x_n, y_n)$

$$y = f(x) = w_0 + w_1 x$$

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

$$\text{Loss} = L = \frac{1}{2n}\sum_{i=1}^{n}(f(x_i) - y_i)^2 = \frac{1}{2n}\sum_{i=1}^{n}((w_0 + w_1 x_i) - y_i)^2$$

Simple Linear Regression: one variable

$y = f(\mathrm{x}) = w_0 + w_1 x$

$$\text{Loss} = L = \frac{1}{2N}\sum_{i=1}^{N}(f(x_i) - y_i)^2 = \frac{1}{2N}\sum_{i=1}^{N}((w_0 + w_1 x_i) - y_i)^2$$

Parameters: w_0, w_1

Goal: minimize Loss function

$$\frac{\partial L}{\partial w_0} = 0, \qquad \frac{\partial L}{\partial w_1} = 0$$

Linear Regression: Learning

Goal: minimize Loss function

$$\frac{\partial L}{\partial w_0} = 0, \qquad \frac{\partial L}{\partial w_1} = 0$$

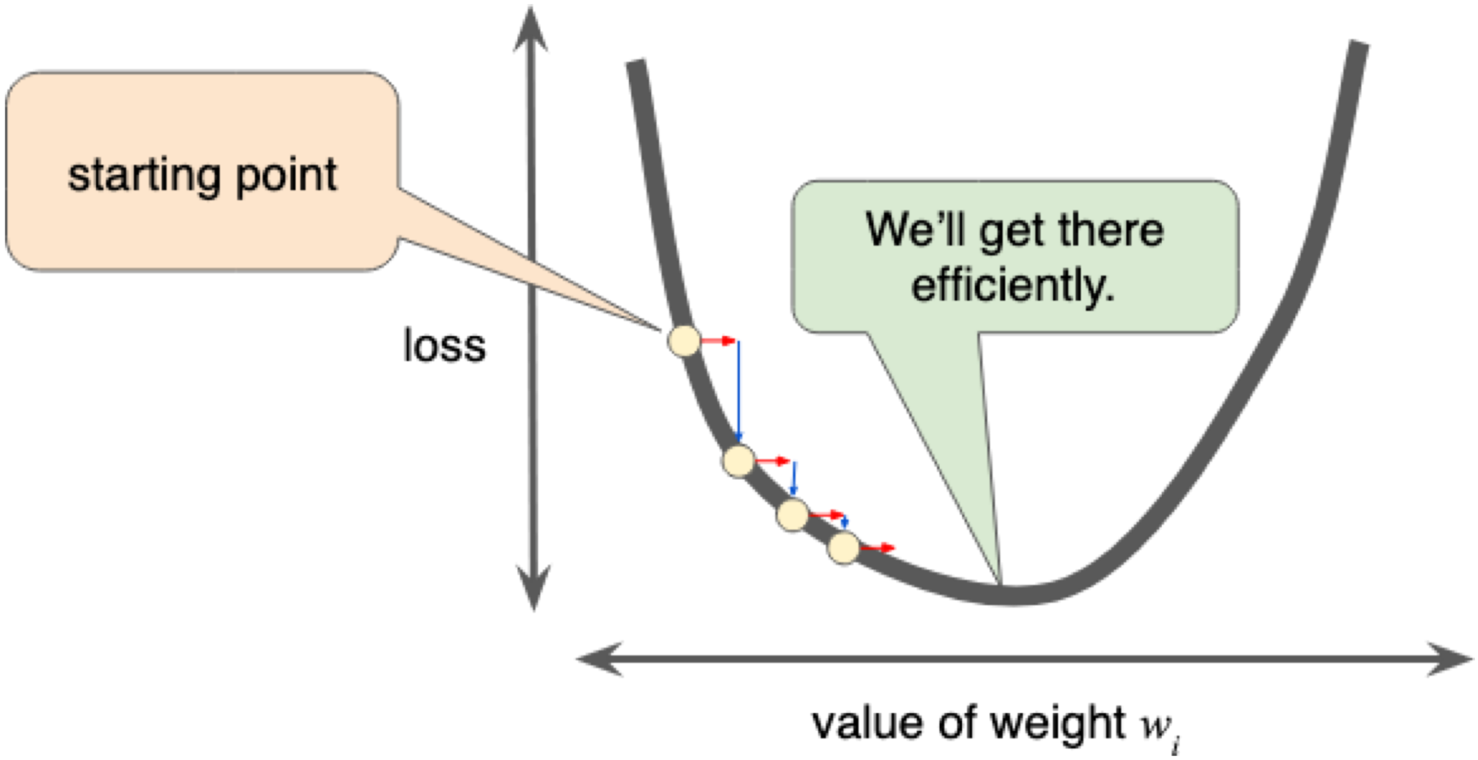
$$L = \frac{1}{2N}\sum_{i=1}^{N}(f(x_{i1}) - y_i)^2 =$$

$$= \frac{1}{2N}\sum_{i=1}^{N}((w_0 + w_1 x_{i1}) - y_i)^2$$

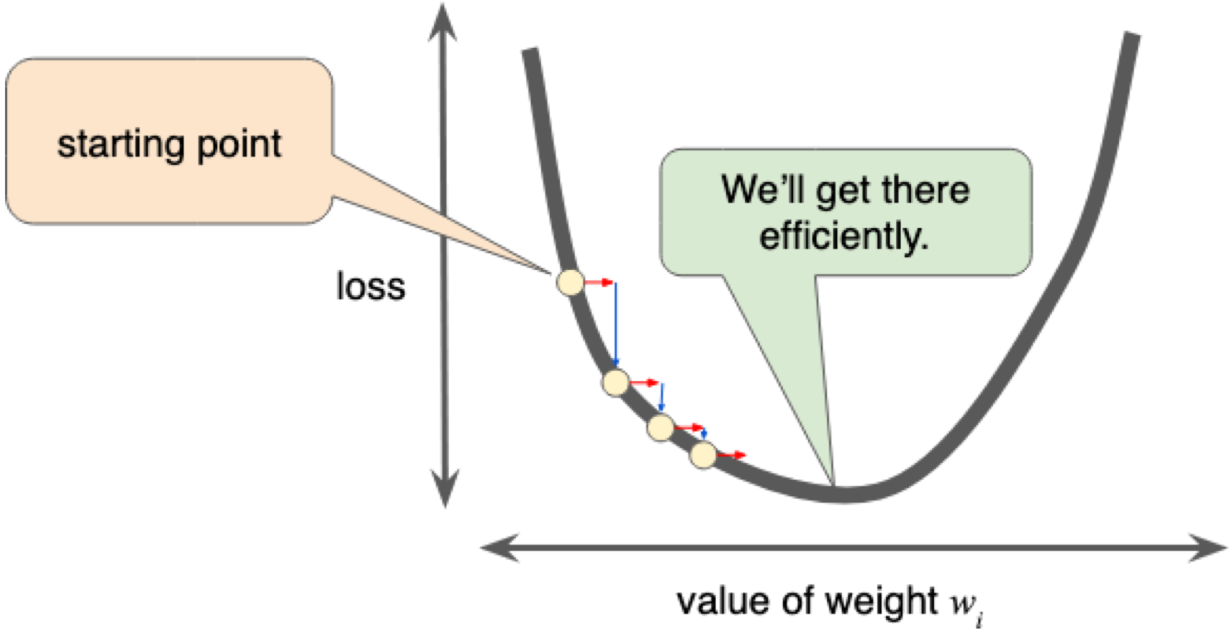
$$\frac{\partial L}{\partial w_0} = \frac{1}{N}\sum_{i=1}^{N}(f(x_{i1}) - y_i)$$

$$\frac{\partial L}{\partial w_1} = \frac{1}{N}\sum_{i=1}^{N}(f(x_{i1}) - y_i)\, x_{i1}$$

Linear Regression: Learning by Gradient Descent



Linear Regression: Learning by Gradient Descent

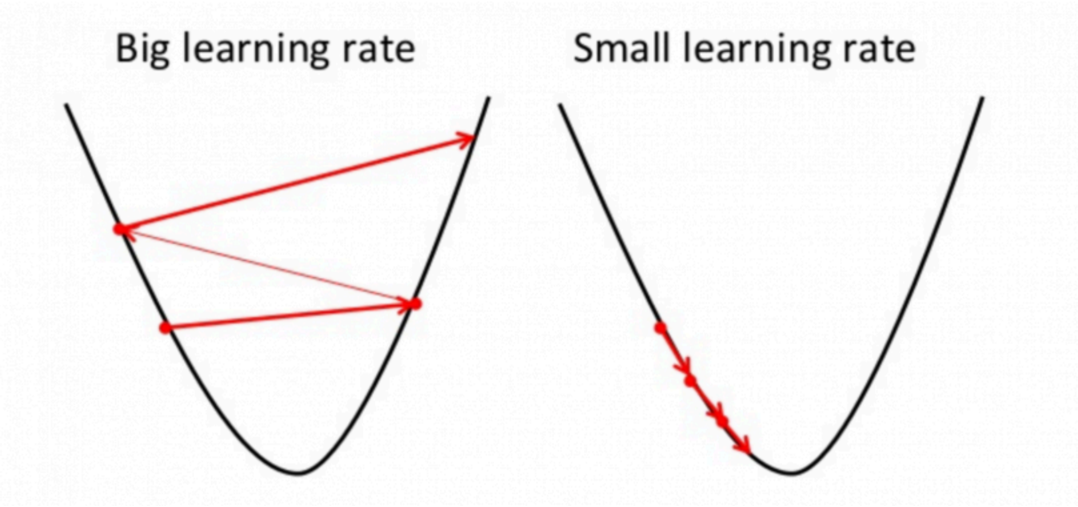


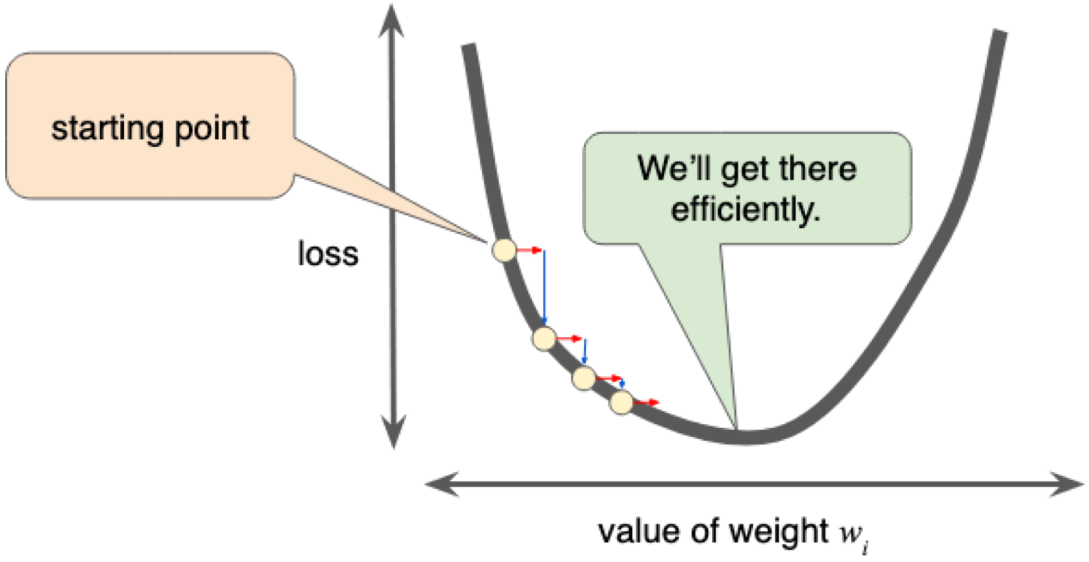
Update parameters:

Learning rate

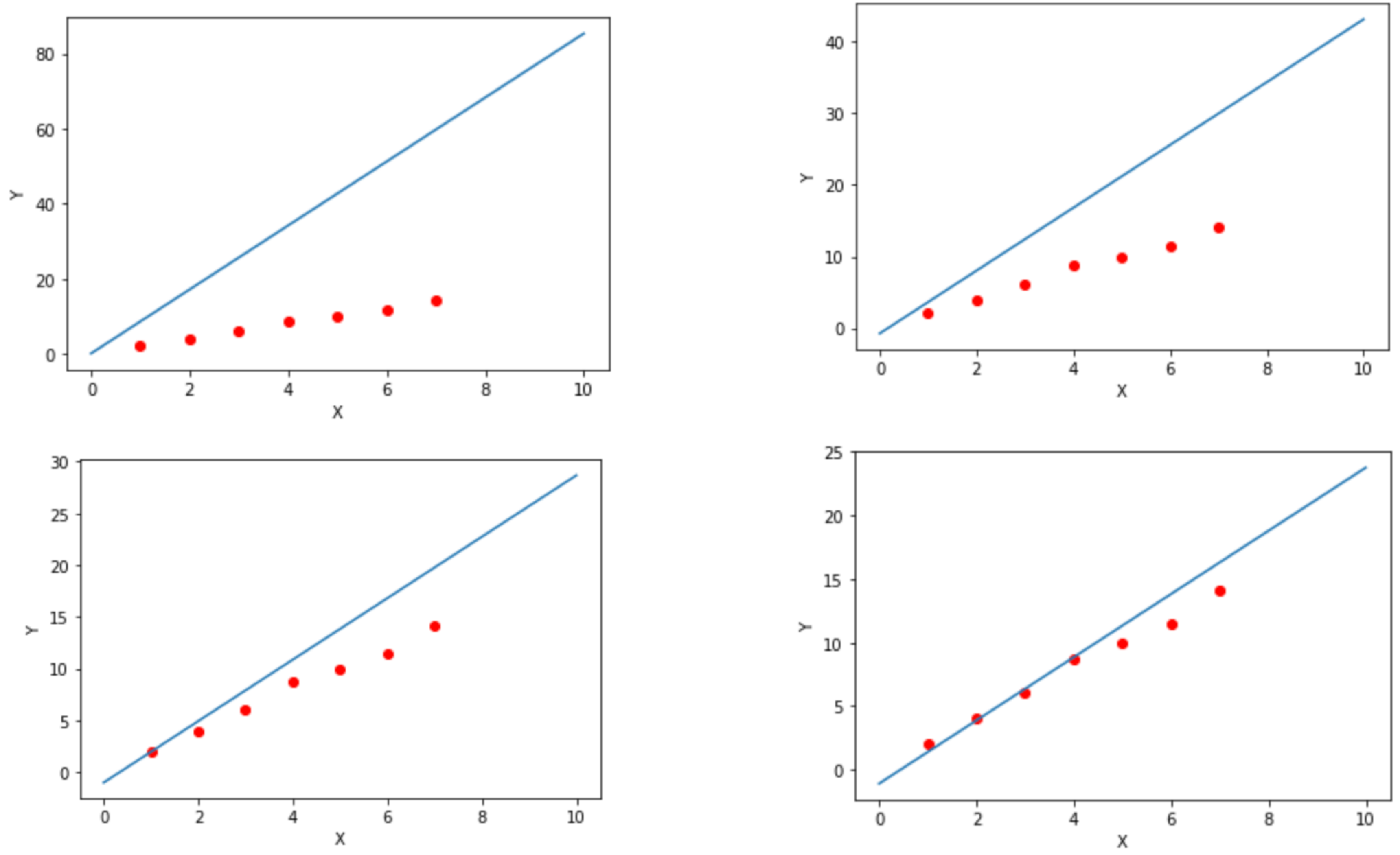
$$w_i = w_i - \mu \frac{\partial L}{\partial w_i}$$

Linear Regression: Learning by Gradient Descent





Learning progress



Multiple Linear Regression

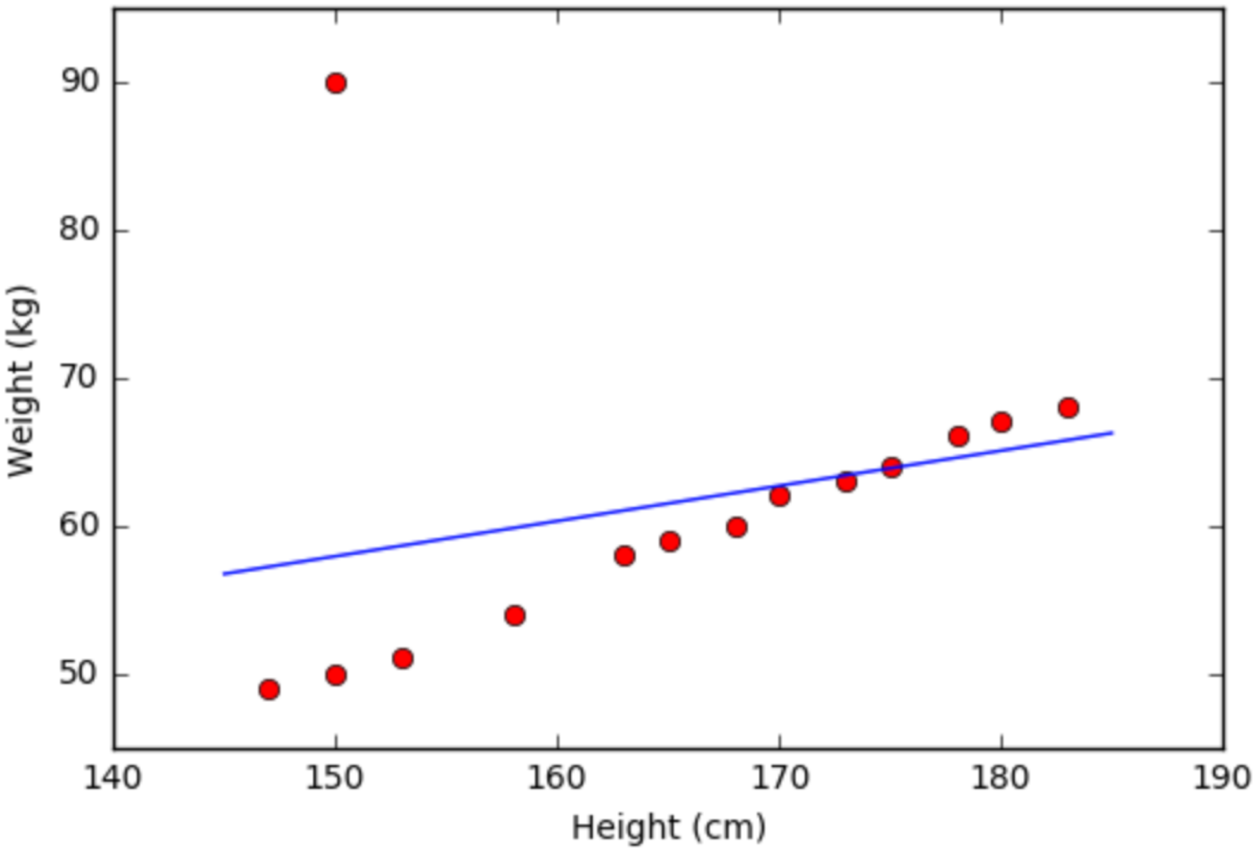
$(X_\mathrm{i}, \mathrm{y_i})$
$X_\mathrm{i} = (x_{i1}, \ldots, x_\mathrm{k})$

$y_\mathrm{i} = f(\mathrm{X_i}) = w_0 x_{\mathrm{i}0} + w_1 x_{\mathrm{i}1} + \ldots + w_\mathrm{k} x_{\mathrm{i}k}$

$$\text{Loss} = L = \frac{1}{2N}\sum_{i=1}^{N}(f(\mathrm{X}) - y_i)^2 = \frac{1}{2N}\sum_{i=1}^{N}((w_{\mathrm{i}0} + w_{\mathrm{i}1}x + \cdots + w_{\mathrm{ik}}x) - y_i)^2 =$$

$$\frac{\partial L}{\partial w_\mathrm{t}} = \frac{1}{N}\sum_{i=1}^{N}(f(\mathrm{X}_i) - y_i)\, x_{i\mathrm{t}}$$

Issues:
Linear Regression is sensitive with outliers



Summary

- Supervised learning vs unsupervised learning.
- Classification vs regression
- Linear regression form.
- Learning Linear Regression model by Gradient Descent algorithm?
- One variable and Multiple variable of Linear Regression

Exercise

- Implement Linear Regression for the problem of house pricing with multiple variables:
  - Firstly derive steps and mathematical formulas in the algorithm.
  - Secondly, implementation using python.

Answer the questions:

1. What is the objective of a supervised learning model? General model?
2. What is linear regression model?
3. Formulating the linear regression model with single variable?
   - Function y = f(x)
   - Loss function
4. Learning parameters by Gradient Descent
   - Update parameters?
   - Learning rate?
5. Implementation
6. What are limitations of linear regression

```python
# hàm y = f(x)
# w: tham số
def f(x,w):
    return w[0]+w[1]*x

# tính hàm loss
def loss(x,y,w):
    d = 0
    for i in range(len(x)):
        d += (y[i] - (w[0] + w[1]*x[i]))**2
    return d/(2*len(x))

# tính đạo hàm tại điểm w[0] và w[1]
def derivative(x,y,w):
    d0=0
    d1=0
    for i in range(len(x)):
        d1 += x[i]*(f(x[i],w)-y[i])
        d0 += f(x[i],w)-y[i]
    return d0/len(x),d1/len(x)
```

```python
# training
epoch = 10
learning_rate = 0.01
w = [1,1] # y = x + 1
los_old = 0
for i in range(epoch):
    # hiển thị đồ thị
    plt.plot(x,y,'ro')
    plt.xlabel('X')
    plt.ylabel('Y')
    x0 = np.linspace(start=1, stop=10, num=50)
    y0 = w[0]+w[1]*x0
    plt.plot(x0,y0)
    plt.show()

    # cập nhật tham số
    los = loss(x,y,w)
    print('epoch_',i,':')
    print(los,' : ',los_old)
    if los>(los_old-0.0001) and i>0:
        break
    los_old = los

    # cập nhật
    a,b = derivative(x,y,w)# cho w0 và w1
    w[0] = w[0]-a*learning_rate
    w[1] = w[1]-b*learning_rate
```

$$f(X) = w_0.x_0 + w_1.x_1 + ... + w_d.x_d$$

$$W = [w_0, w_1, ..., w_d]^T$$

$$X = [x_0, x_1, ..., x_d]$$

$$y = X.W$$

$$Loss = \frac{1}{2}\sum_{i=1}^{n}(f(X_i) - y_i)^2 = \frac{1}{2}\|X.W - y\|^2$$

$$\frac{\partial Loss(W)}{\partial W} = X^T(X.W - y) = 0$$

$$=> X^T.X.W = X^T.y$$

$$X^T.W \triangleq b$$

$$X^T.X.W = b$$

$$A.W = X^T.b$$

$$A^{-1}A.W = A^{-1}.b \ W = A^{-1}.b$$