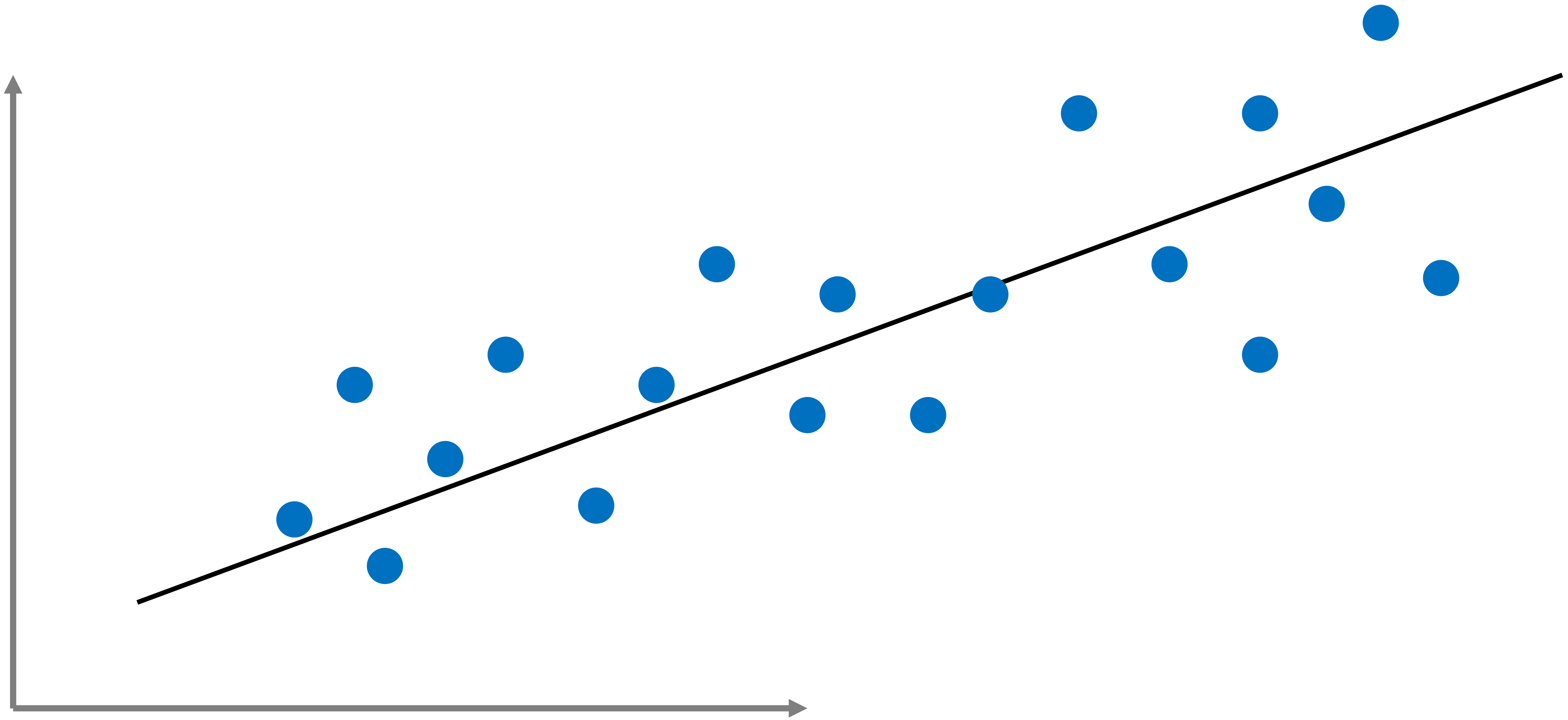
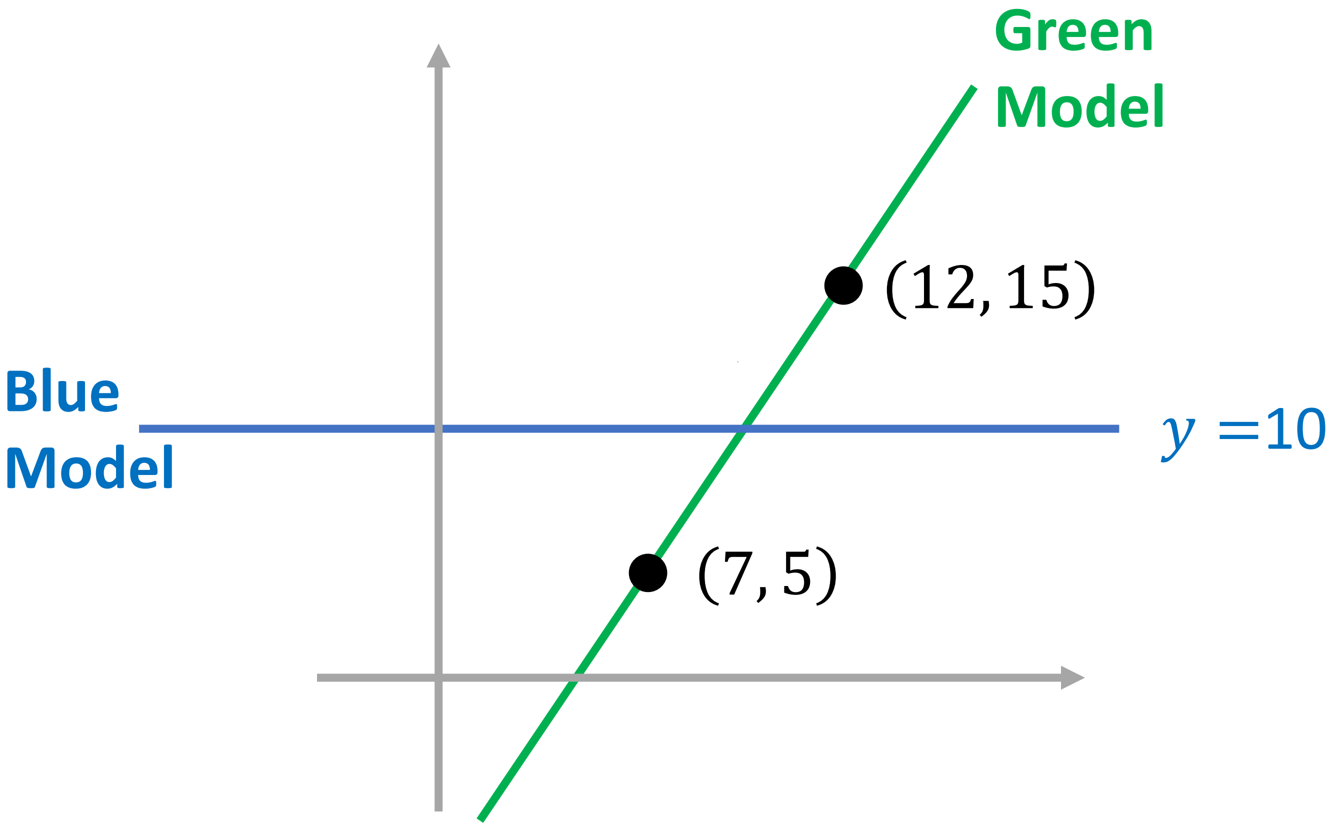
Linear models

Linear regression



Error definition

y_i = Instance
$\hat{y}_i$ = Model

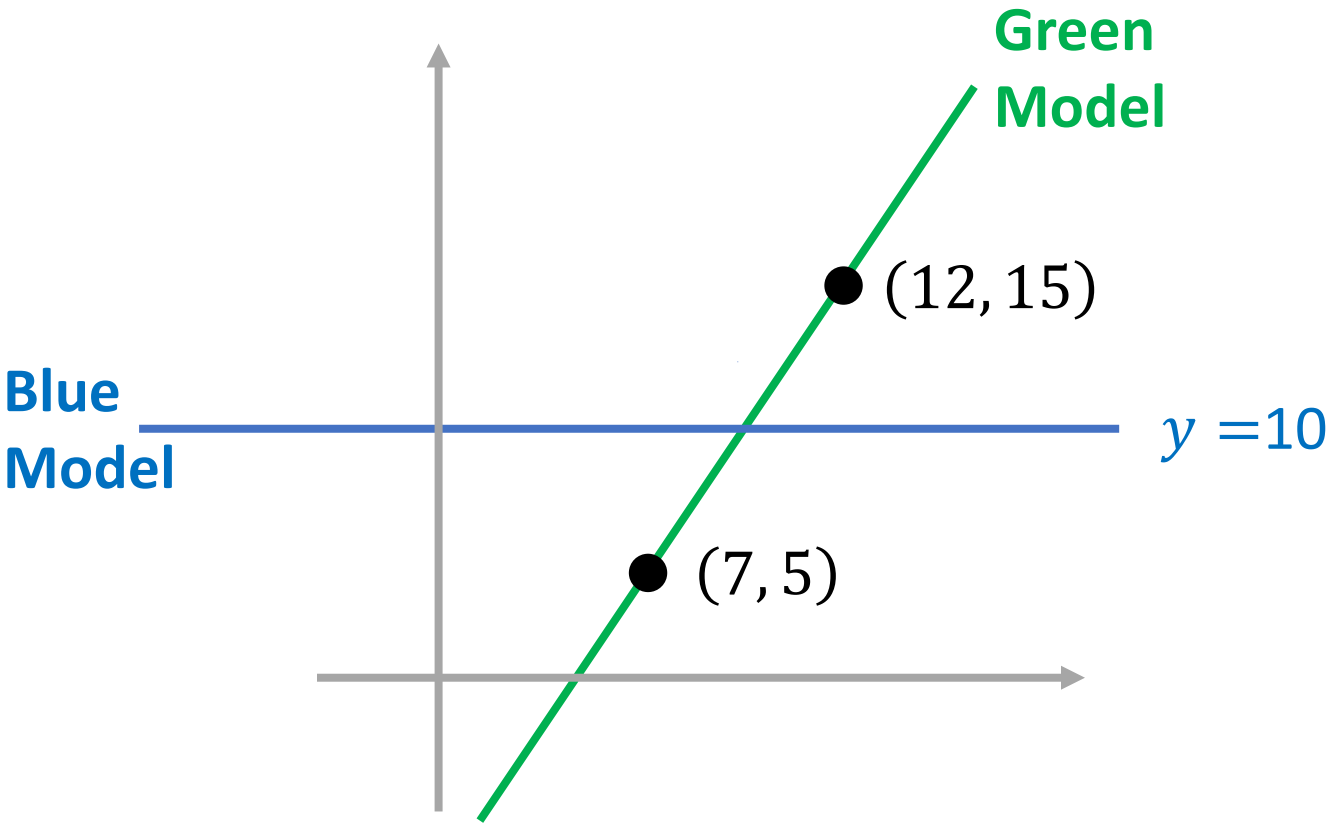


$\varepsilon \propto \hat{y}_i - y_i$

$\varepsilon = 0$

$\varepsilon = 0$

Error definition



$$\varepsilon \propto \hat{y}_i - y_i$$

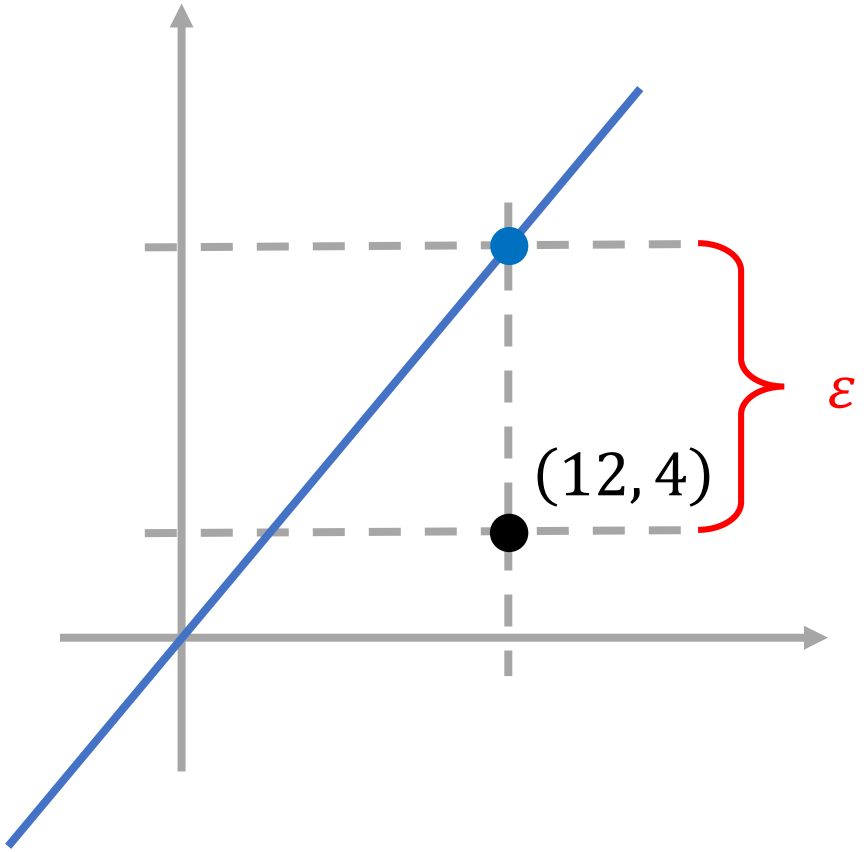
$$\varepsilon = 0$$

$$\varepsilon = 0$$

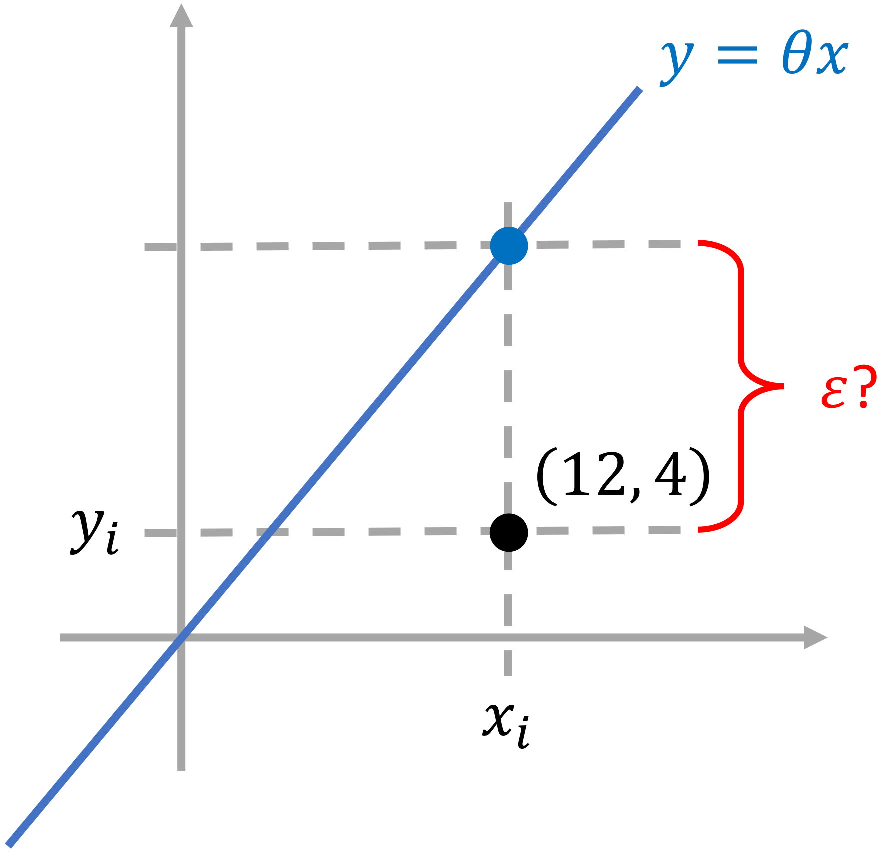
Solution:

$$\varepsilon := \sum_{i=1}^{m} (\hat{y}_i - y_i)^2$$

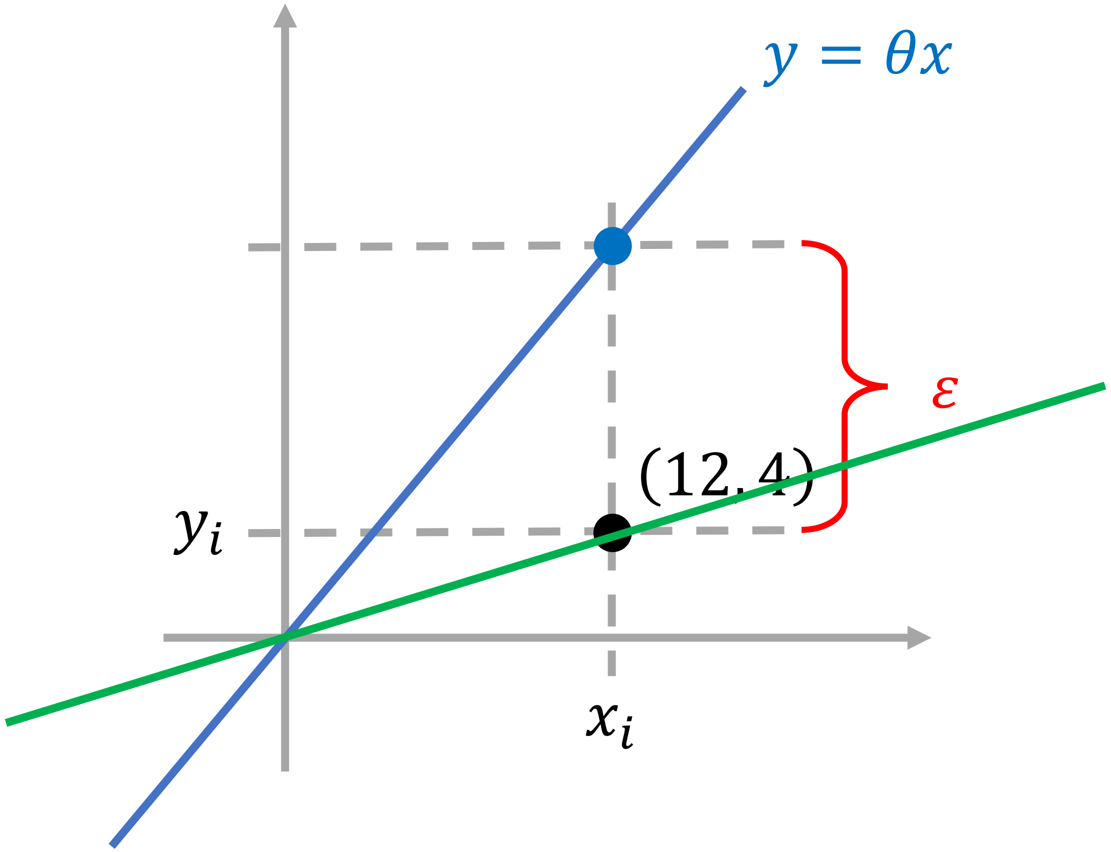
$y = \theta x \quad m = 1$



$$y = \theta x \qquad m = 1$$



$y = \theta x \qquad m = 1$



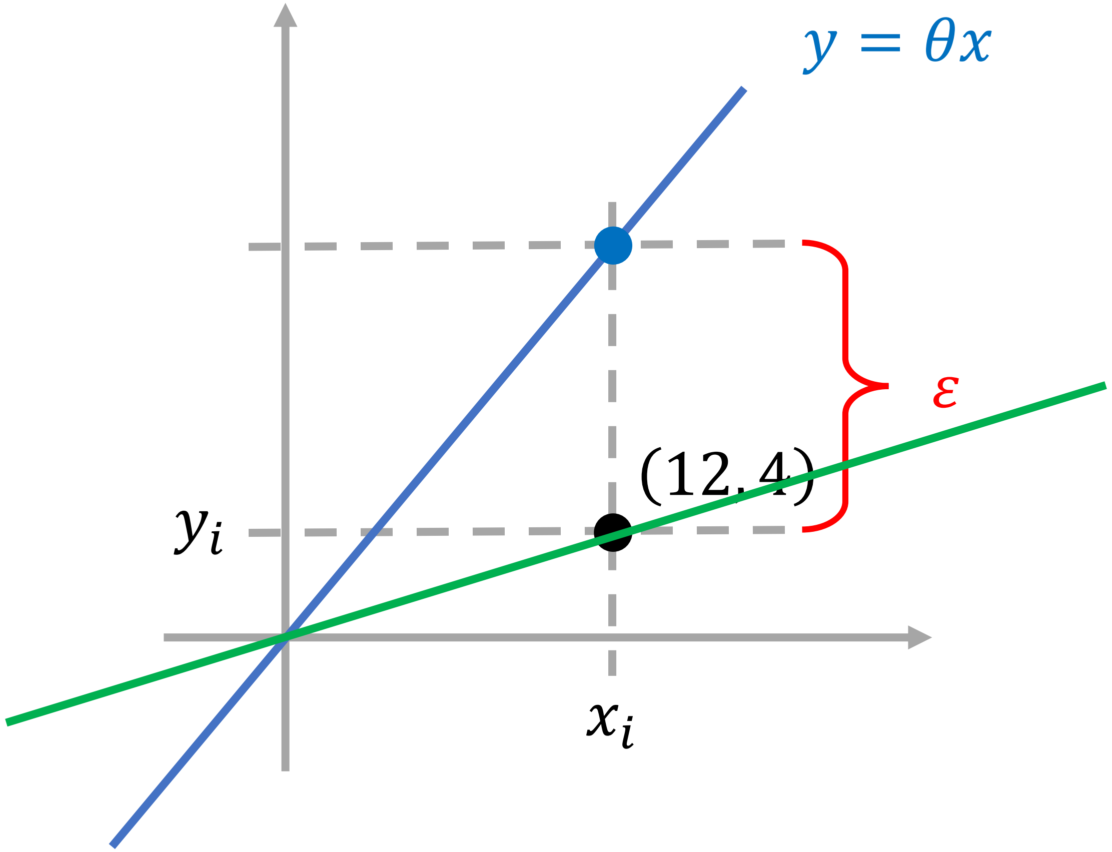
$$\varepsilon = (\ y \ - 4\ )^2$$
$$= (12\theta - 4)^2$$
$$= (x_i\theta - y_i)^2$$

$$\frac{d\varepsilon}{d\theta} = 0 \qquad \Rightarrow \qquad \theta = 4/12 = y_i/x_i$$

$$\therefore \ y = \frac{4}{12}x$$

$$\begin{aligned} y &= f(\theta) \\ \varepsilon &= g(y) \end{aligned} \quad \Rightarrow \varepsilon = h(\theta)$$

$y = \theta x \quad m = 1$

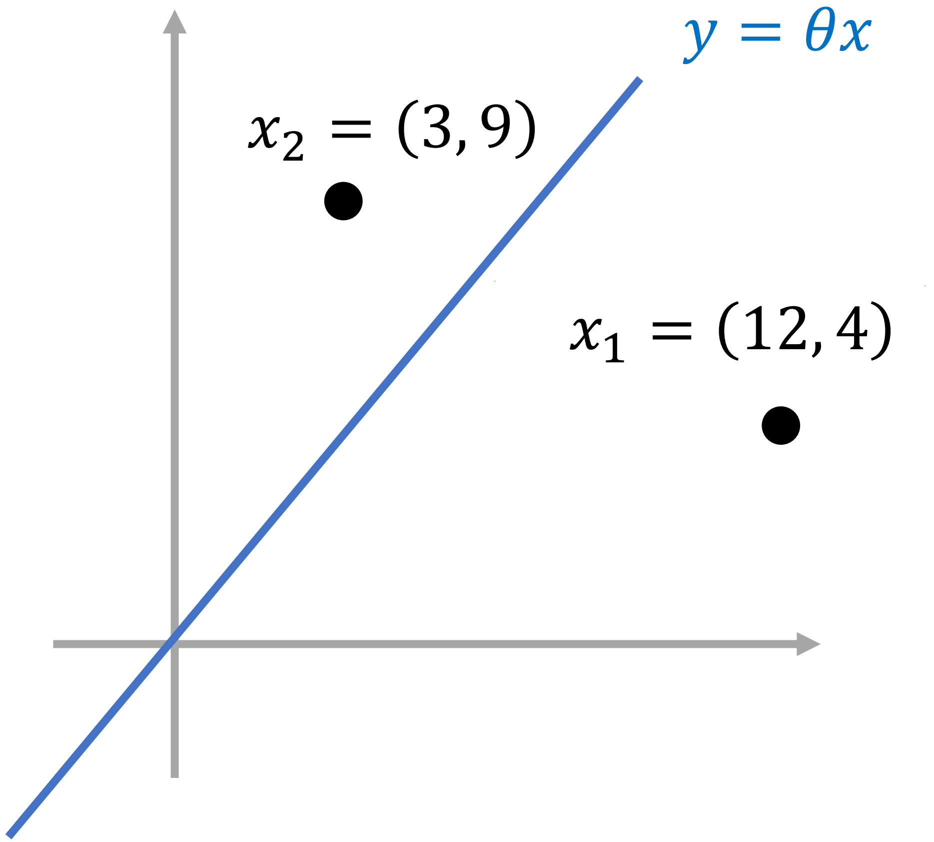


>_ Code

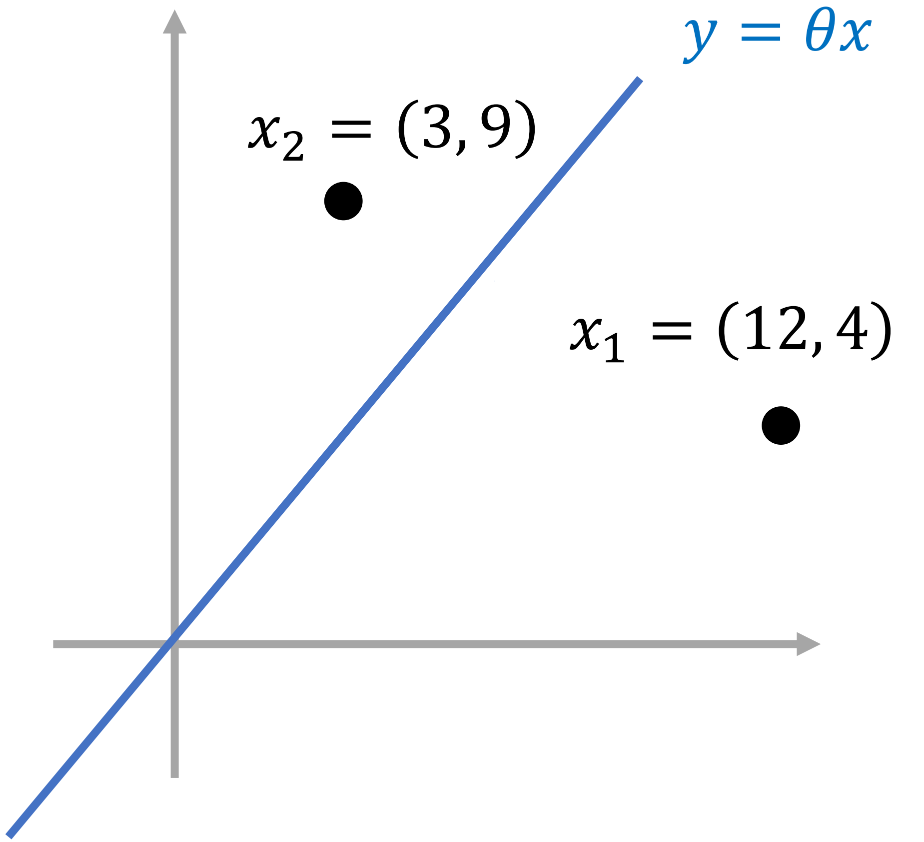
$\therefore \; y = \frac{4}{12} x$

```
import matplotlib.pyplot as plt
xi = 12 # data points o o o o o
yi = 4
theta = yi/xi
plt.scatter(xi,yi)
# model
x = np.linspace(10, 14, num = 5)
y = theta * x
plt.plot(x, y)
plt.axis('equal')
plt.show()
```

$$y = \theta x \qquad m = 2$$

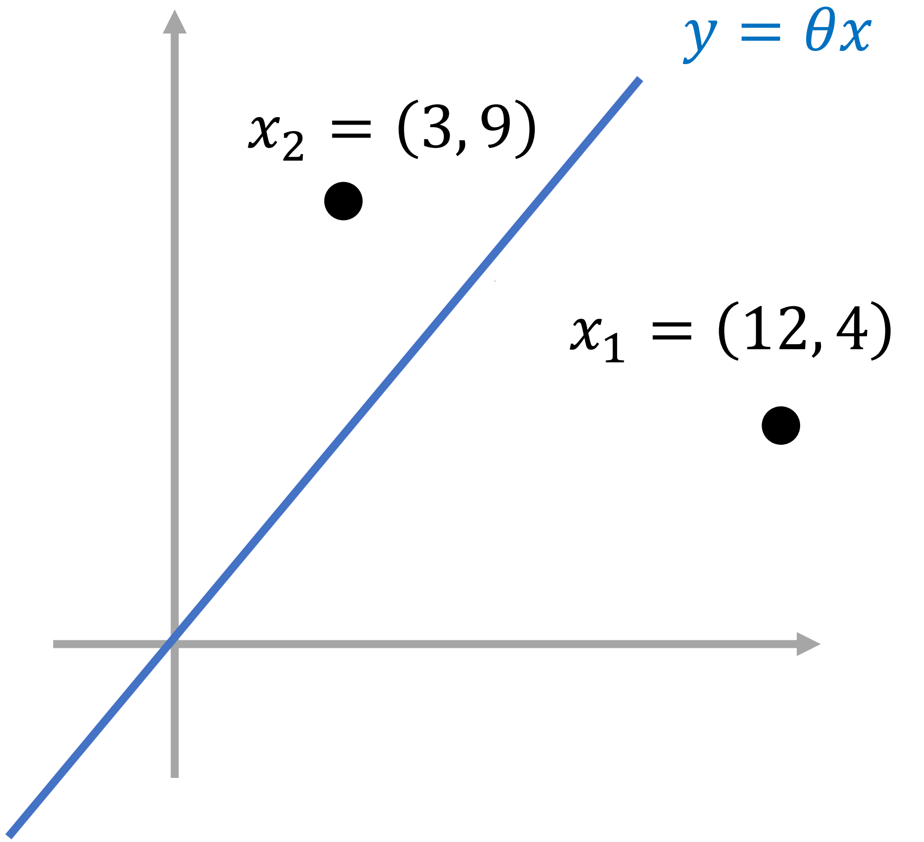


$$y = \theta x \qquad m = 2$$



$$\varepsilon = (\theta x_1 - y_1)^2 + (\theta x_2 - y_2)^2$$

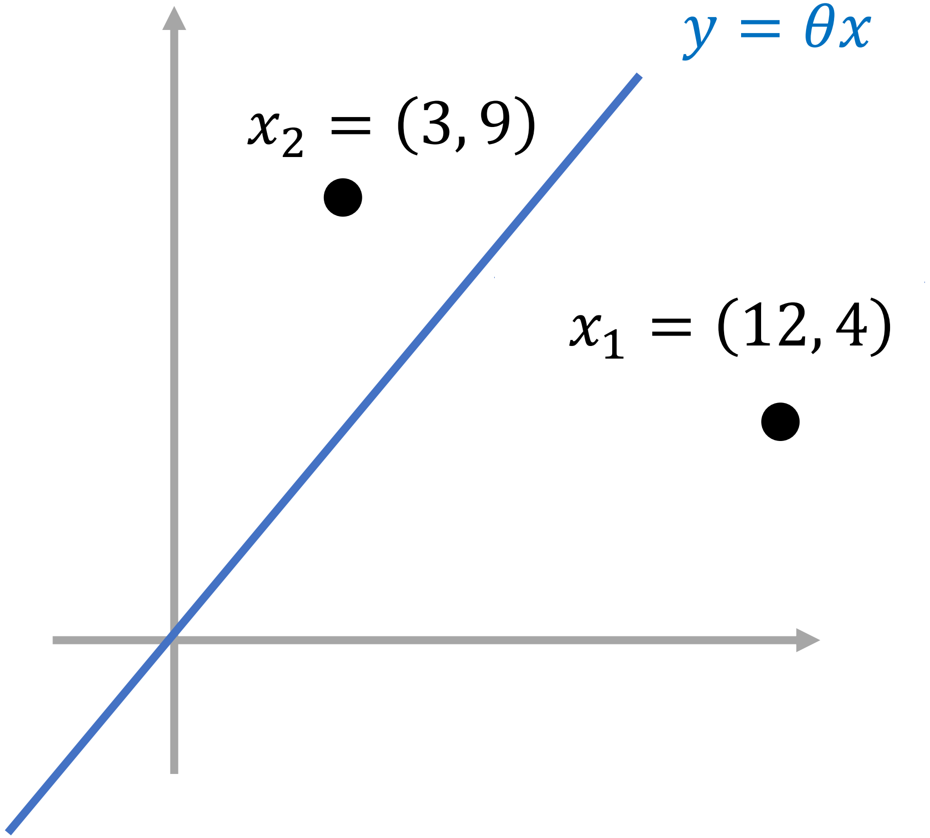
$y = \theta x \qquad m = 2$



$$\varepsilon = (\theta x_1 - y_1)^2 + (\theta x_2 - y_2)^2$$

$$\frac{d\varepsilon}{d\theta} = 0 \quad \Rightarrow \quad \theta = \frac{x_1 y_1 + x_2 y_2}{x_1^2 + x_2^2}$$

$$y = \theta x \qquad m = 2$$



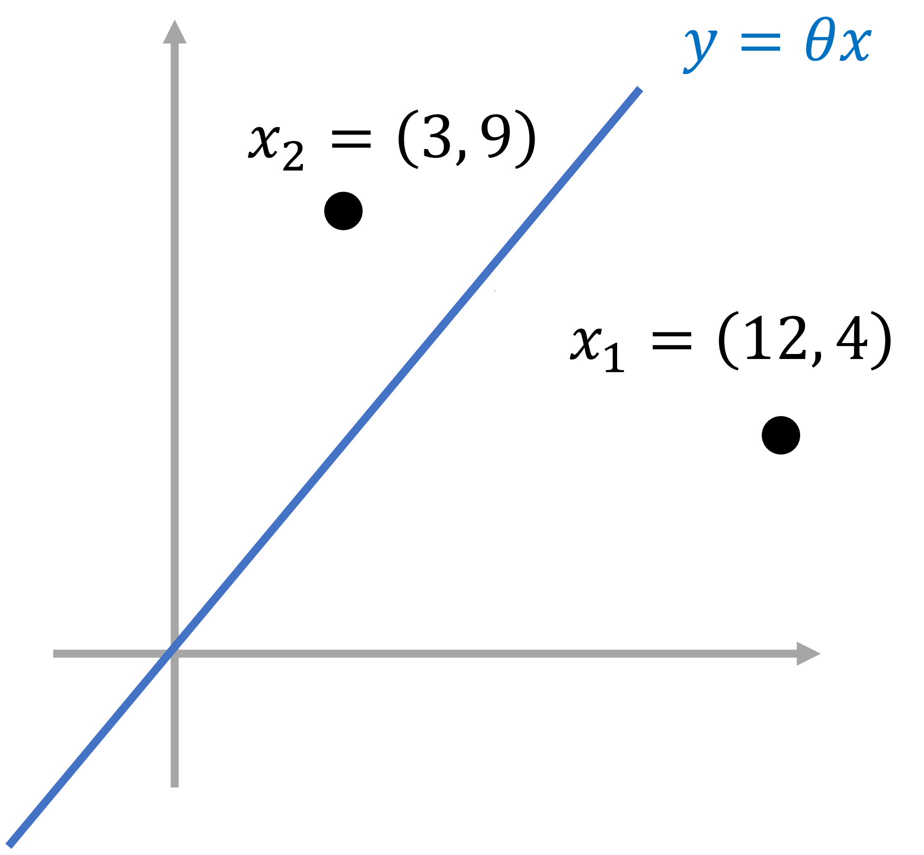
$$\varepsilon = (\theta x_1 - y_1)^2 + (\theta x_2 - y_2)^2$$

$$\frac{d\varepsilon}{d\theta} = 0 \quad \Rightarrow \quad \theta = \frac{x_1 y_1 + x_2 y_2}{x_1^2 + x_2^2}$$

Matrix form for fast vectorized computations

$$\begin{pmatrix} \square & \square \\ \square & \square \end{pmatrix}$$

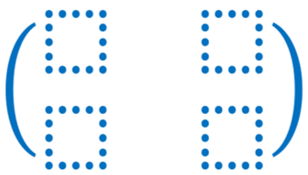
$$y = \theta x \qquad m = 2$$



$$\varepsilon = (\theta x_1 - y_1)^2 + (\theta x_2 - y_2)^2$$

$$\frac{d\varepsilon}{d\theta} = 0 \quad \Rightarrow \quad \theta = \frac{x_1 y_1 + x_2 y_2}{x_1^2 + x_2^2}$$

Matrix form for fast vectorized computations



$$\boldsymbol{x} := \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \qquad \boldsymbol{y} := \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$$

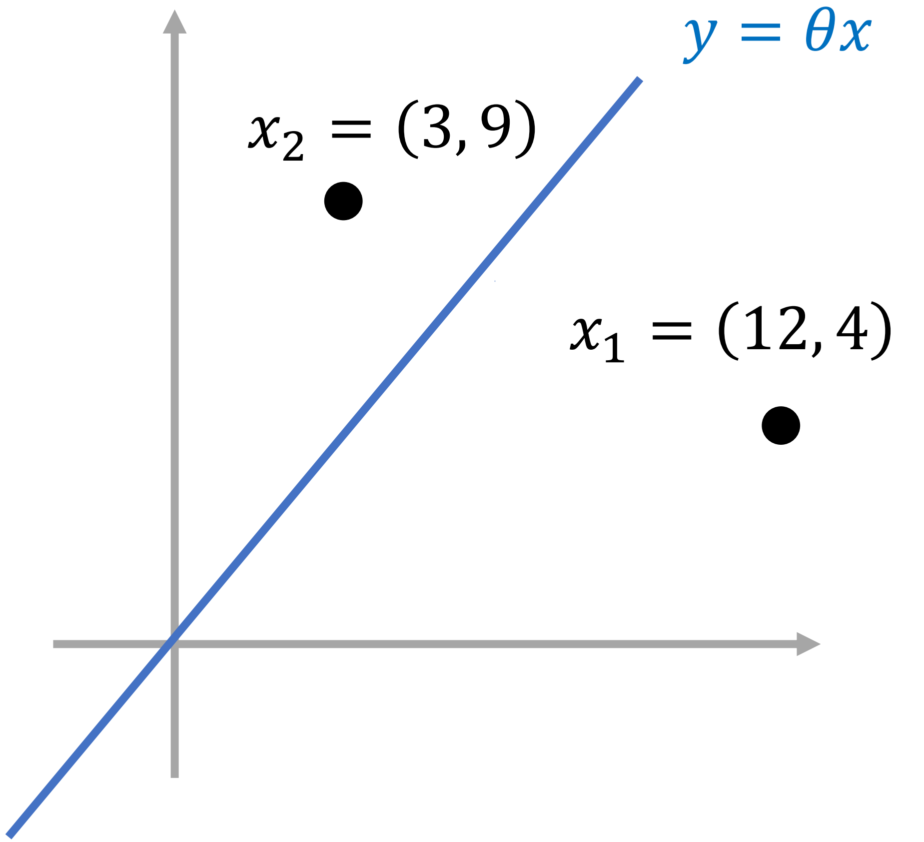
$$\Rightarrow \quad \theta = \frac{(x_1, x_2)^T \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}}{(x_1, x_2)^T \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}} = \boldsymbol{x}^T \boldsymbol{y} / \boldsymbol{x}^T \boldsymbol{x}$$

>_ Code

$y = \theta x \quad m = 2$

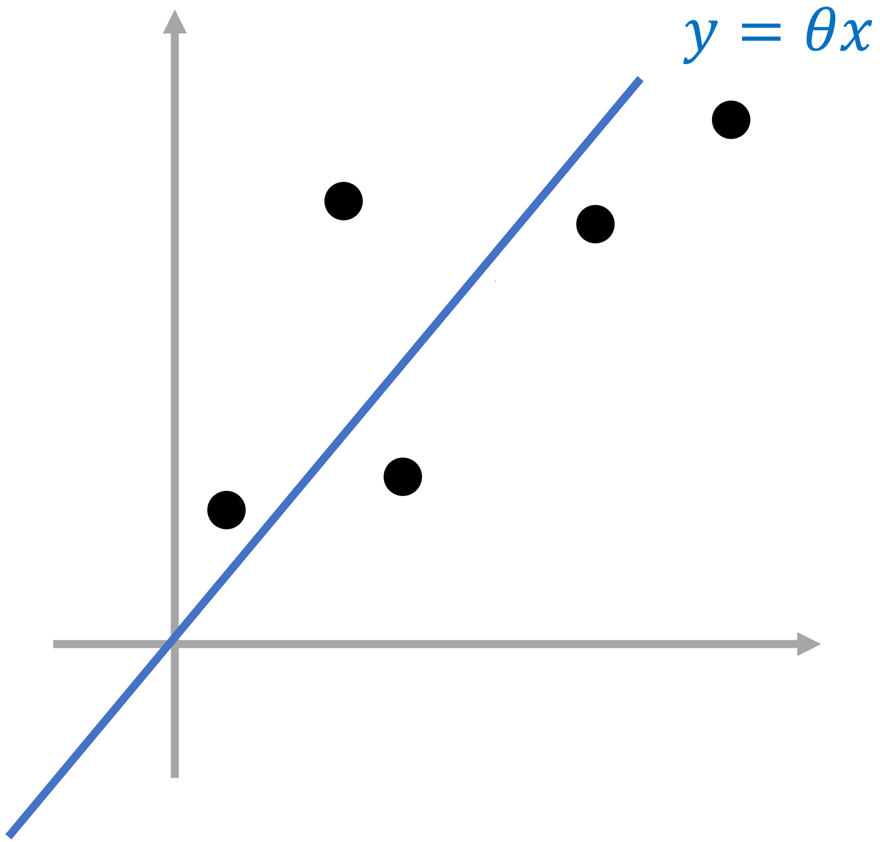
$$\theta = \frac{(x_1, x_2)^T \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}}{(x_1, x_2)^T \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}} = \boldsymbol{x}^T \boldsymbol{y} / \boldsymbol{x}^T \boldsymbol{x}$$

>_ Code

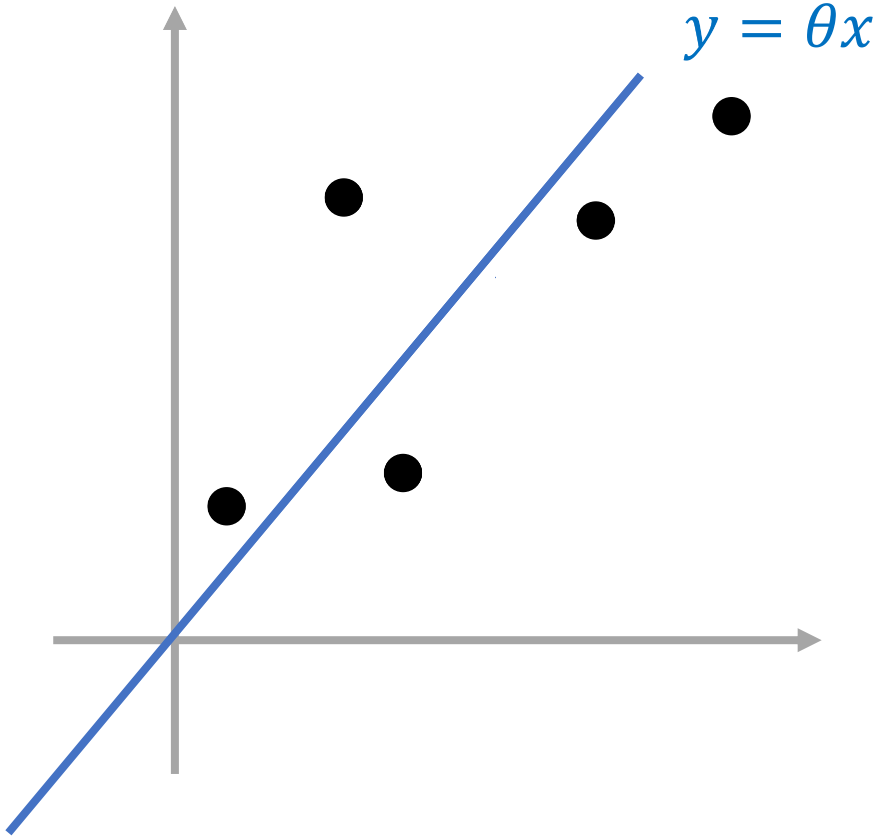



```
import numpy as np
import matplotlib.pyplot as plt
xi = np.array([3, 12]) # data points o o o
yi = np.array([9, 4])
theta = sum(xi * yi) / sum(xi**2)
plt.scatter(xi,yi)
# model
x = np.linspace(min(xi), max(xi), num = 5)
y = theta * x
plt.plot(x, y)
plt.axis('equal')
plt.show()
```

$y = \theta x \qquad m$

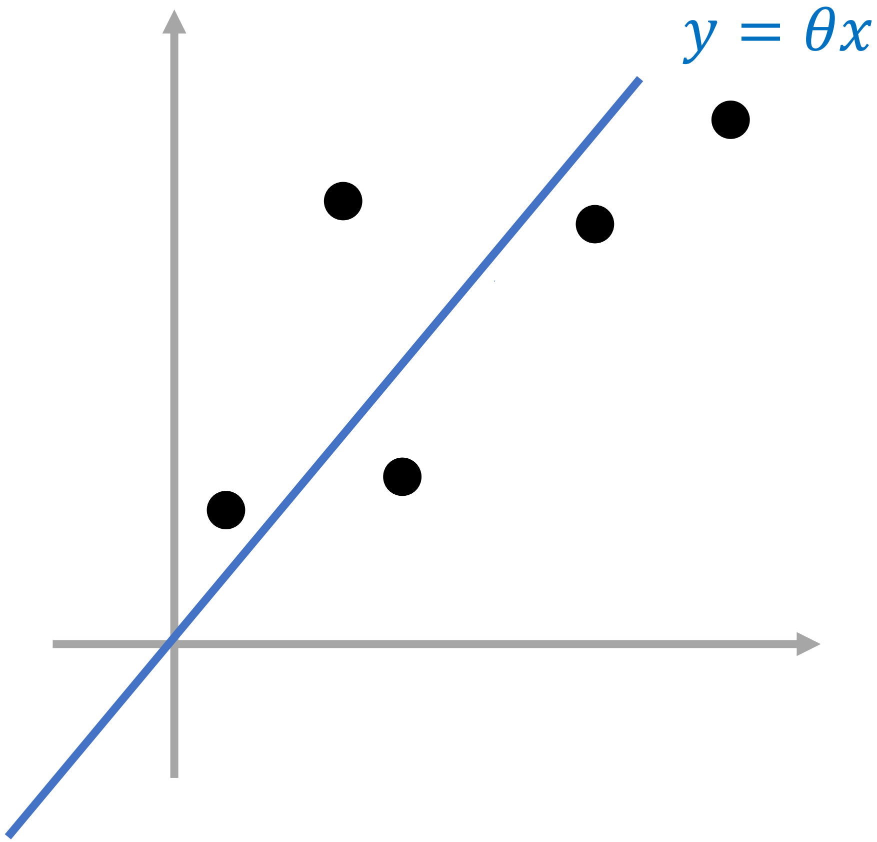


$$y = \theta x \qquad m$$



$$\begin{aligned} \varepsilon = \ & (\theta x_1 - y_1)^2 \\ & +(\theta x_2 - y_2)^2 \\ & \vdots \\ & +(\theta x_m - y_m)^2 \end{aligned}$$

$$y = \theta x \qquad m$$



$$\varepsilon = (\theta x_1 - y_1)^2 + (\theta x_2 - y_2)^2 + \cdots + (\theta x_m - y_m)^2$$

$$\frac{d\varepsilon}{d\theta} = 0 \quad \Rightarrow \quad \theta = \frac{\sum_{i=1}^{m} x_i y_i}{\sum_{i=1}^{m} x_i^2}$$

Matrix form for fast vectorized computations

$$\boldsymbol{x} := \begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix} \qquad \boldsymbol{y} := \begin{pmatrix} y_1 \\ \vdots \\ y_m \end{pmatrix}$$

$$\Rightarrow \quad \theta = \boldsymbol{x}^T \boldsymbol{y} / \boldsymbol{x}^T \boldsymbol{x}$$

>_ Code

Your turn

$$y = A\sin(x)$$