Linear Regression

Prepared by: Joseph Bakarji

Data

Given a table of numbers, what can you do?

Living area (feet^2)	#bedrooms	Price (1000$s)
2104	3	400
1600	3	330
2400	3	369
1416	2	232
3000	4	540
⋮	⋮	⋮

.....

→

	x_1	x_2	x_3	x_4	.....
Apt. 1	$x_1^{(1)}$	$x_2^{(1)}$	$x_3^{(1)}$	$x_4^{(1)}$	
Apt. 2	$x_1^{(2)}$	$x_2^{(2)}$	$x_3^{(2)}$	$x_4^{(2)}$	⋮
Apt. 3	$x_1^{(3)}$	$x_2^{(3)}$	$x_3^{(3)}$	$x_4^{(3)}$	
Apt. 4	$x_1^{(4)}$	$x_2^{(4)}$	$x_3^{(4)}$	$x_4^{(4)}$	
	⋮	⋮	⋮	⋮	

- Visualization: Look at it!
- Find statistical features: Mean, median, outliers etc.
- Clean it: missing values, …

What are the inputs and outputs?

- **Inputs**: quantities that are typically **given**
- **Outputs**: quantities we want to **predict**

Living area (feet^2)	#bedrooms	Price (1000\$s)
2104	3	400
1600	3	330
2400	3	369
1416	2	232
3000	4	540
$\vdots$	$\vdots$	$\vdots$

	Inputs		Outputs		
	x_1	x_2	x_3	x_4	.....
Apt. 1	$x_1^{(1)}$	$x_2^{(1)}$	$x_3^{(1)}$	$x_4^{(1)}$	
Apt. 2	$x_1^{(2)}$	$x_2^{(2)}$	$x_3^{(2)}$	$x_4^{(2)}$	$\vdots$
Apt. 3	$x_1^{(3)}$	$x_2^{(3)}$	$x_3^{(3)}$	$x_4^{(3)}$	
Apt. 4	$x_1^{(4)}$	$x_2^{(4)}$	$x_3^{(4)}$	$x_4^{(4)}$	
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	

Given inputs, predict outputs

	Inputs		Outputs	
	x_1	x_2	y_1	y_2
Apt. 1	$x_1^{(1)}$	$x_2^{(1)}$	$y_1^{(1)}$	$y_2^{(1)}$
Apt. 2	$x_1^{(2)}$	$x_2^{(2)}$	$y_1^{(2)}$	$y_2^{(2)}$
Apt. 3	$x_1^{(3)}$	$x_2^{(3)}$	$y_1^{(3)}$	$y_2^{(3)}$
Apt. 4	$x_1^{(4)}$	$x_2^{(4)}$	$y_1^{(4)}$	$y_2^{(4)}$
	$\vdots$	$\vdots$	$\vdots$	$\vdots$
Given Input	$x_1^{(n)}$	$x_2^{(n)}$	??	**Unknown output**

Supervised Learning

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \xrightarrow{h} \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

Given the data,
find a **function** h, a.k.a a **hypothesis**,
that predicts outputs, given inputs

$$\mathbf{y} = h(\mathbf{x})$$

Assume multiple inputs, 1 output

x_1	x_2	y	
Living area (feet^2)	#bedrooms	Price (1000\$s)	.....
2104	3	400	
1600	3	330	
2400	3	369	
1416	2	232	
3000	4	540	
$\vdots$	$\vdots$	$\vdots$	

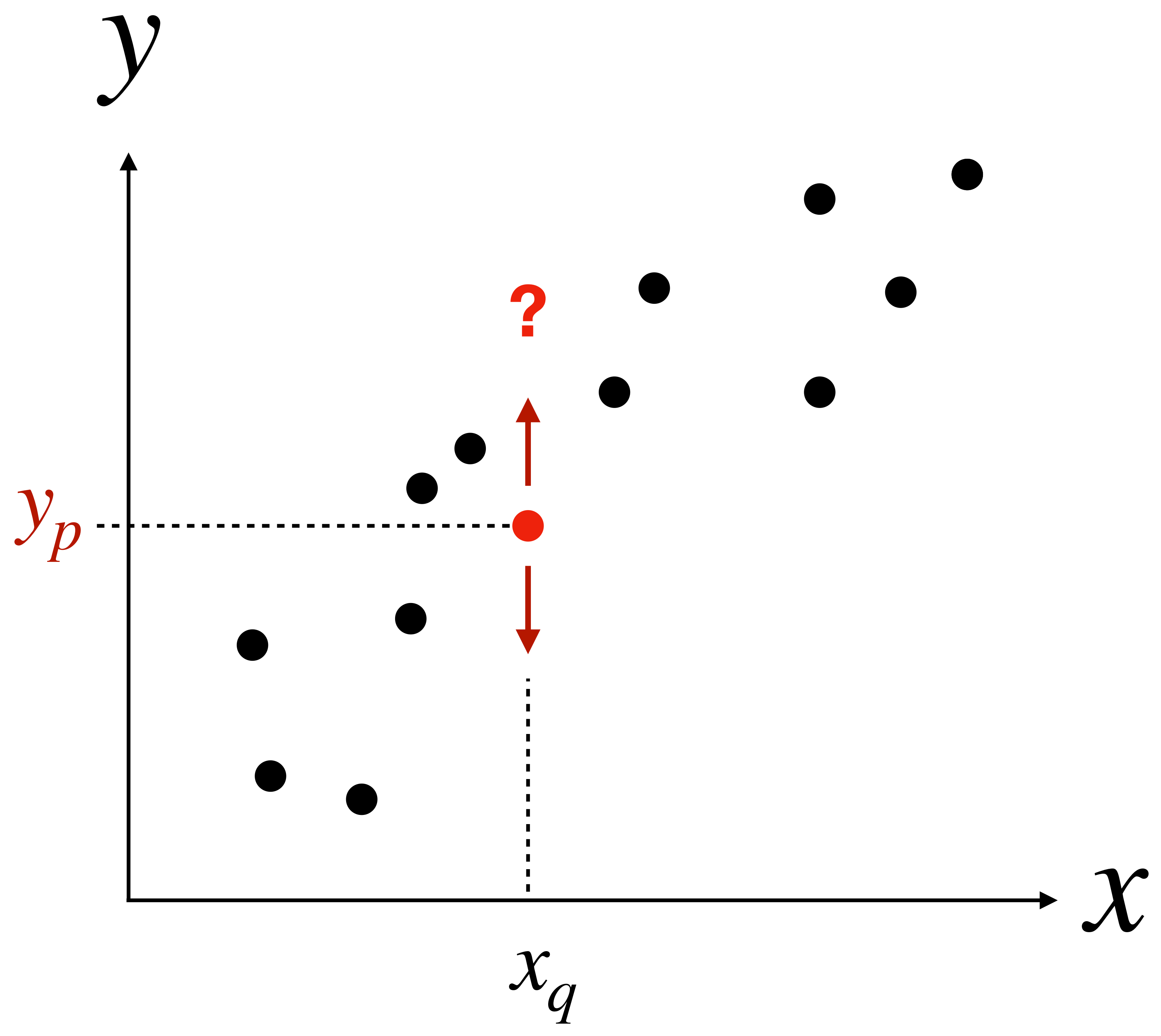
→

Inputs		Output
x_1	x_2	y
$x_1^{(1)}$	$x_2^{(1)}$	$y^{(1)}$
$x_1^{(2)}$	$x_2^{(2)}$	$y^{(2)}$
$x_1^{(3)}$	$x_2^{(3)}$	$y^{(3)}$
$x_1^{(4)}$	$x_2^{(4)}$	$y^{(4)}$
$\vdots$	$\vdots$	$\vdots$

Given new input, what's the output?

(size) Input	(price) Output
x	y
$x^{(1)}$	$y^{(1)}$
$x^{(2)}$	$y^{(2)}$
$x^{(3)}$	$y^{(3)}$
$x^{(4)}$	$y^{(4)}$
$\vdots$	$\vdots$

Query x_q $\qquad$ y_p ?? Prediction



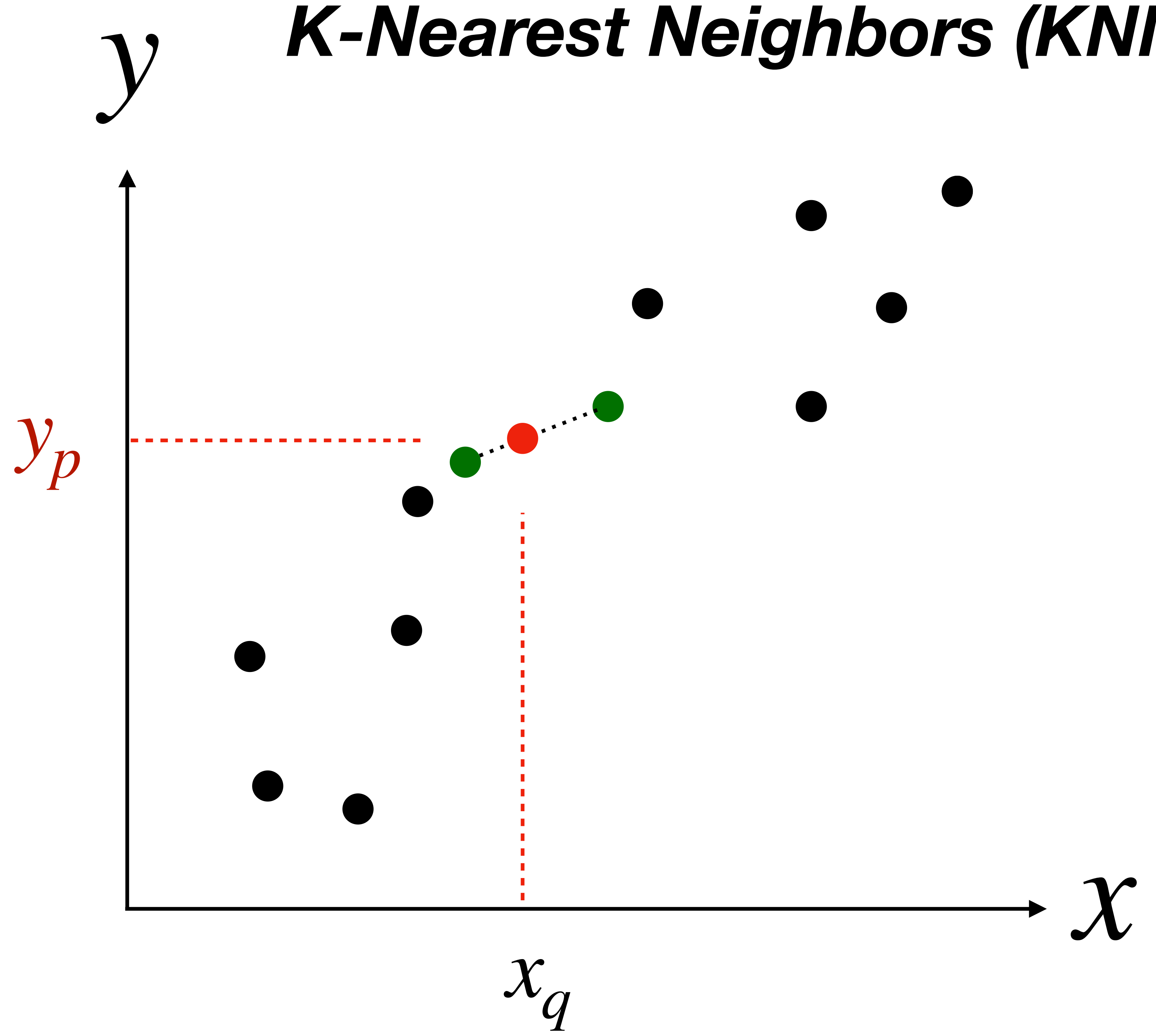
Given new input, what's the output?

Input	Output
x	y
$x^{(1)}$	$y^{(1)}$
$x^{(2)}$	$y^{(2)}$
$x^{(3)}$	$y^{(3)}$
$x^{(4)}$	$y^{(4)}$
$\vdots$	$\vdots$

Query x_q **??** **Prediction**

Weighted average of surrounding points

K-Nearest Neighbors (KNN)



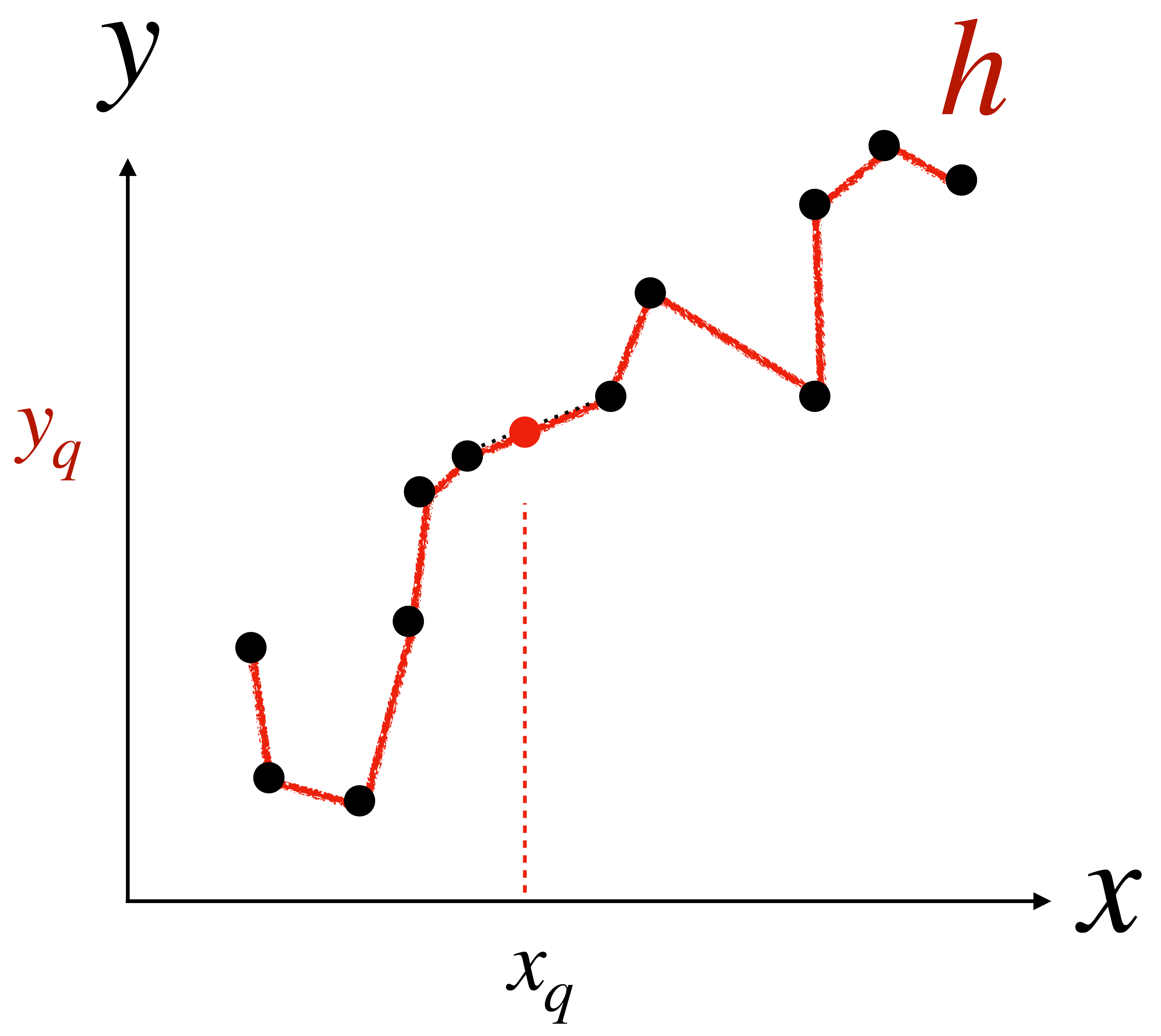
Given new input, what's the output?

$$x \xrightarrow{h} y$$

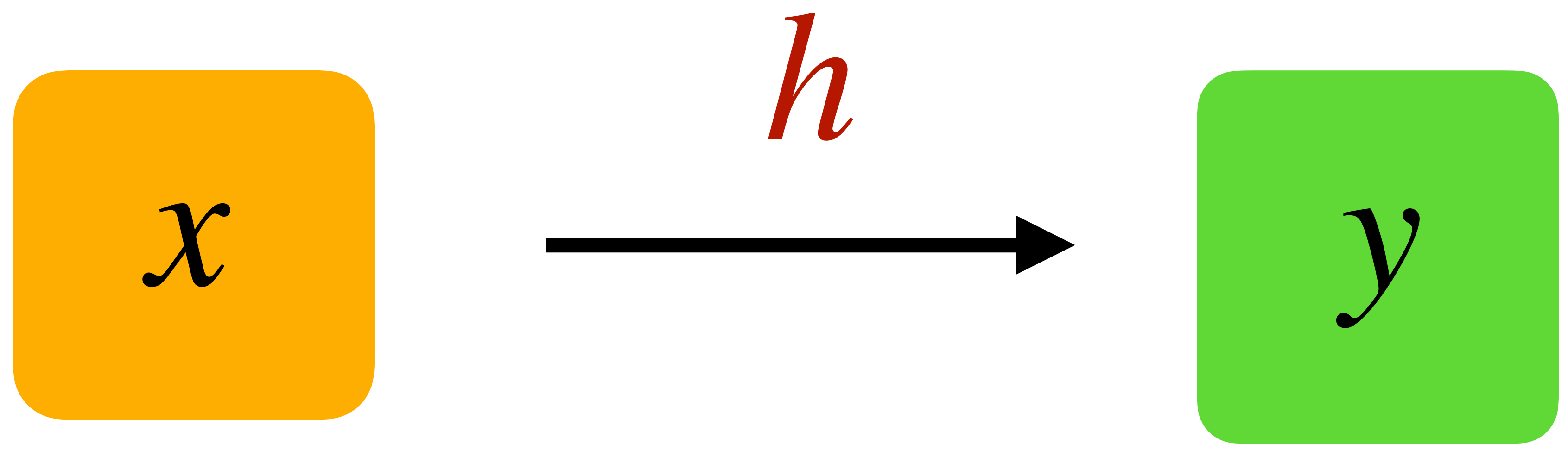
$$h : \mathbb{R} \to \mathbb{R}$$

Given the data,
find a **function** h, a.k.a **hypothesis,**
that predicts an output, given an input

Linear Interpolation

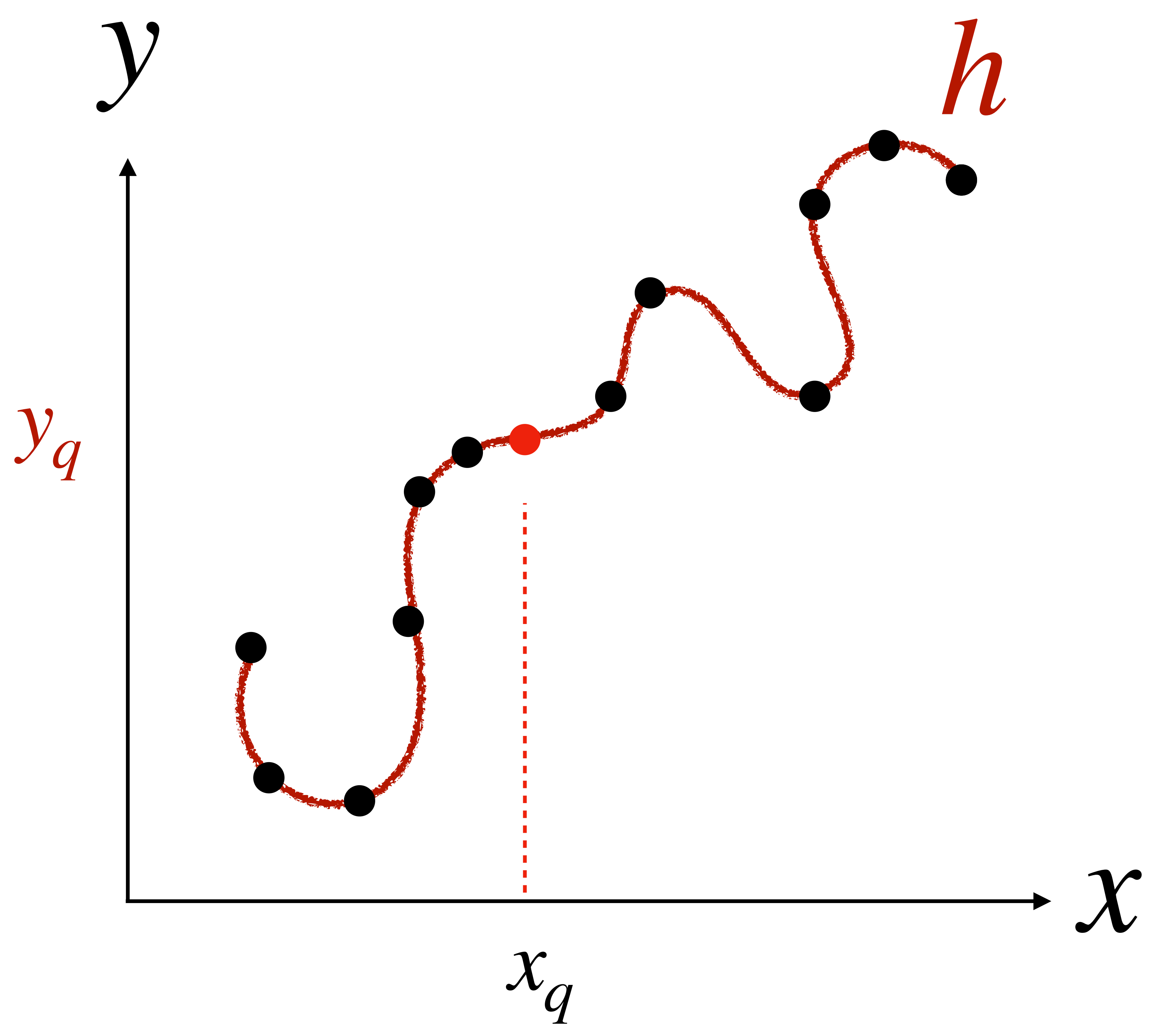


Given new input, what's the output?

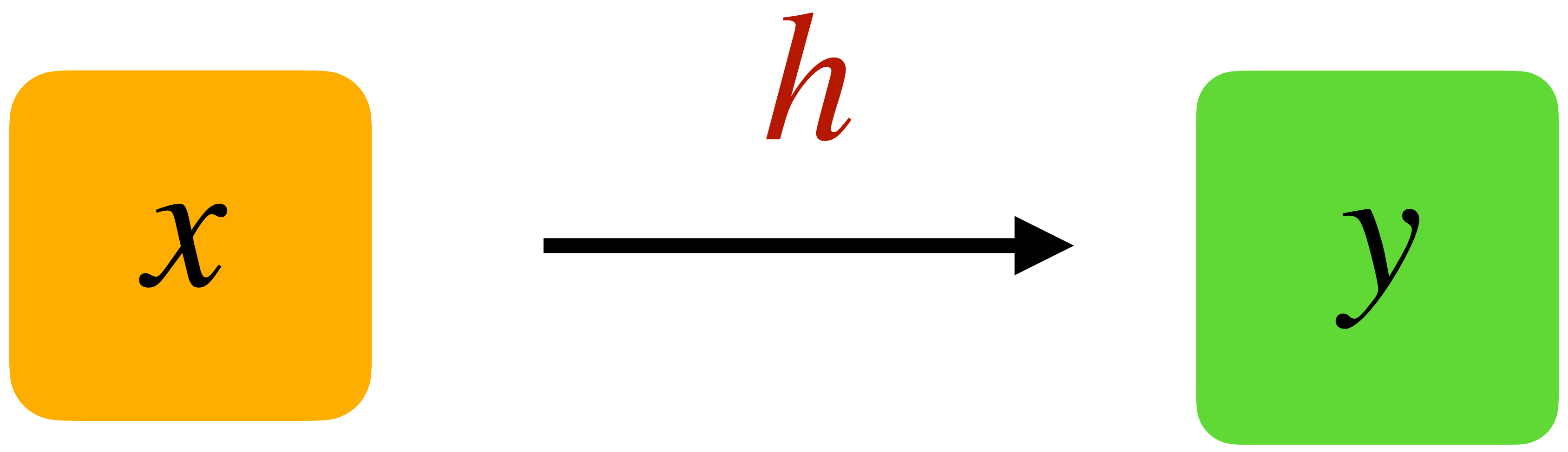


Given the data,
find a **function** h, a.k.a **hypothesis,**
that predicts an output, given an input

Polynomial Interpolation

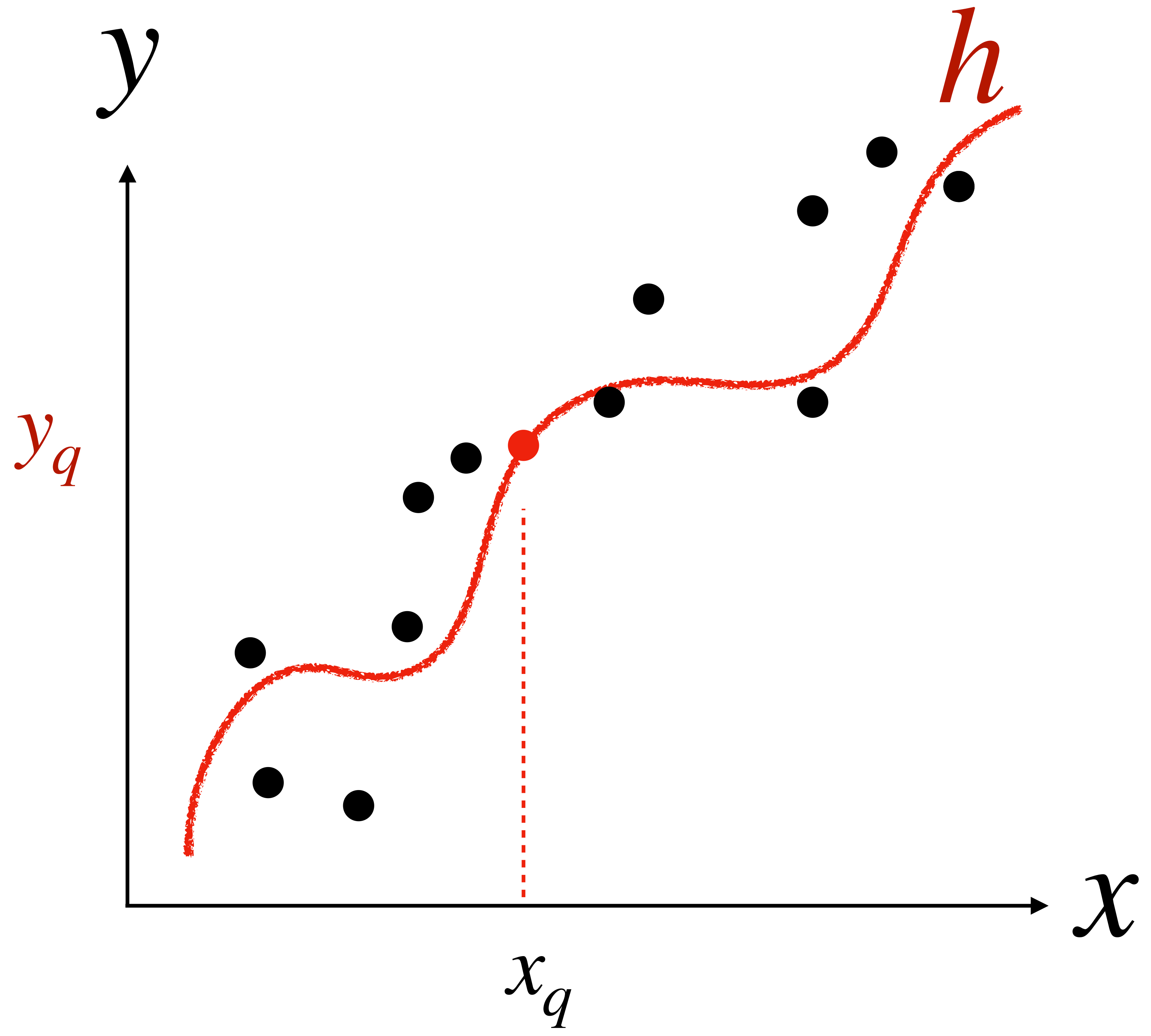


Given new input, what’s the output?



Given the data,
find a **function** h, a.k.a **hypothesis,**
that predicts an output, given an input

Some other function?



Given new input, what's the output?

Assume a linear hypothesis

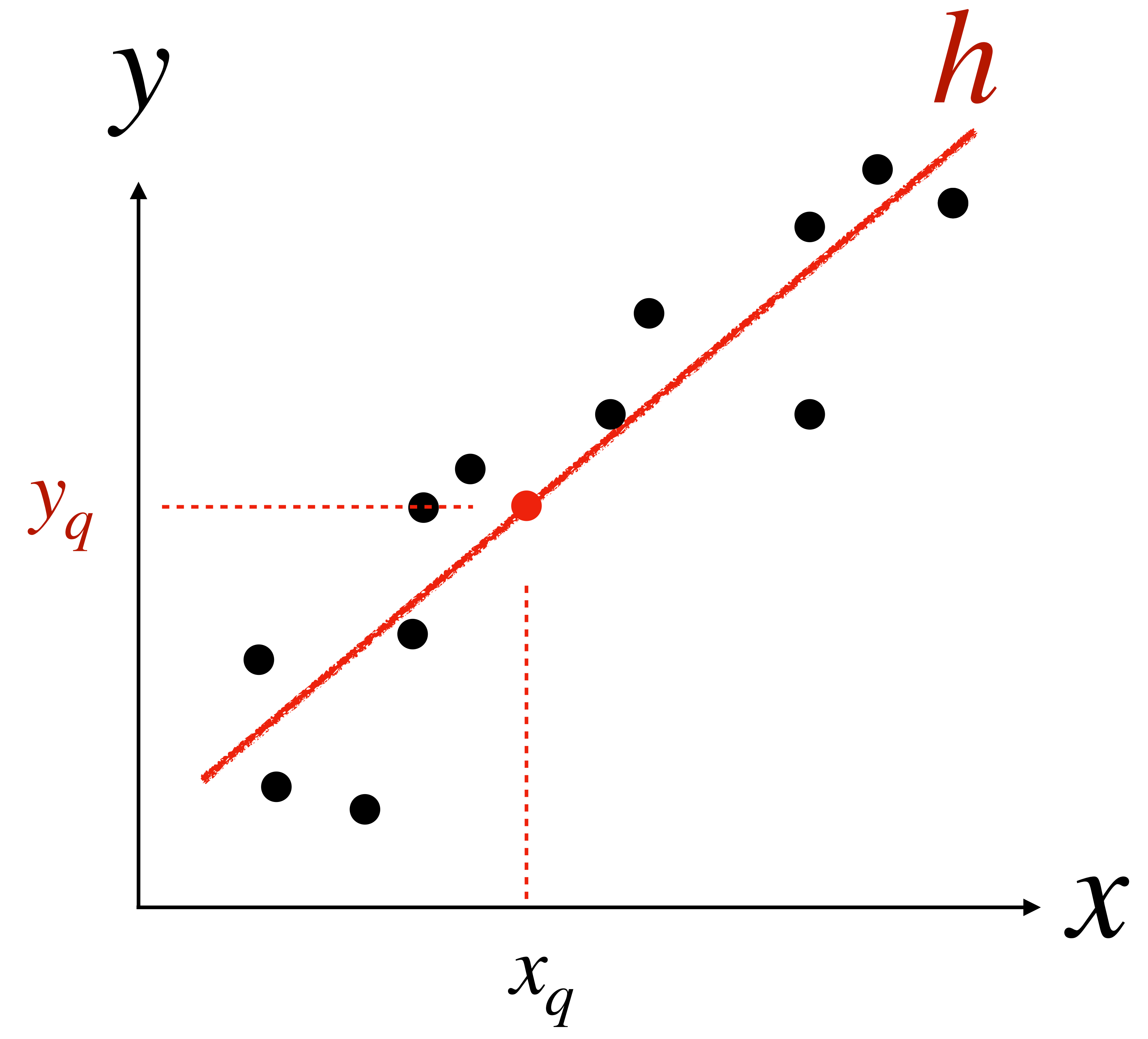
$$x \xrightarrow{h} y$$

$$h(x) = ax + b$$

What are the best a and b that fit the data?

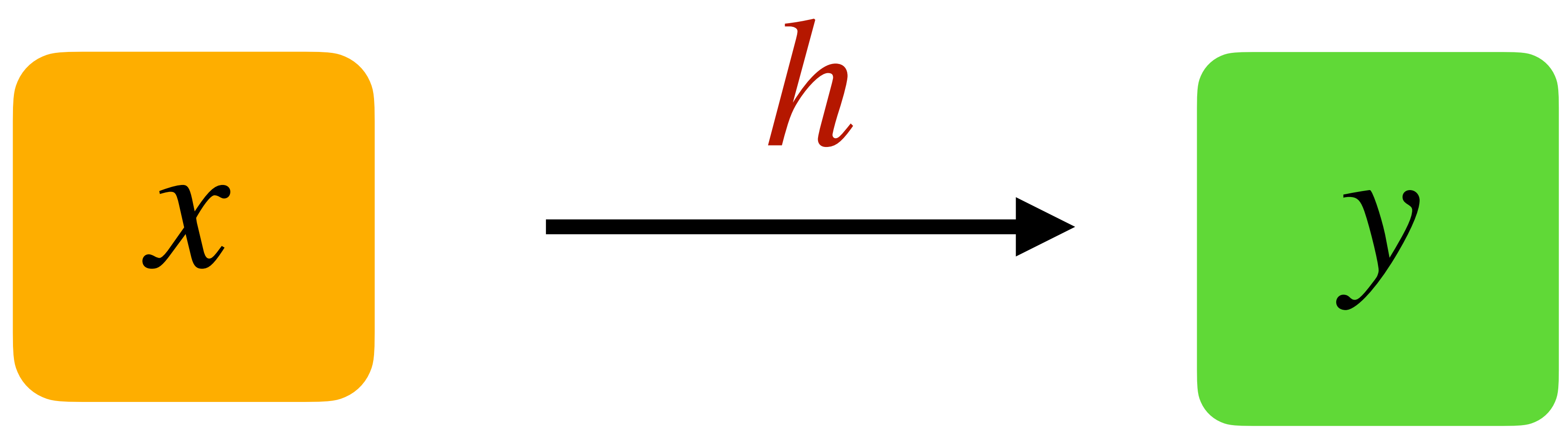
a, b **are fitting parameters**

Linear function



Assume a linear hypothesis

Assume a linear hypothesis



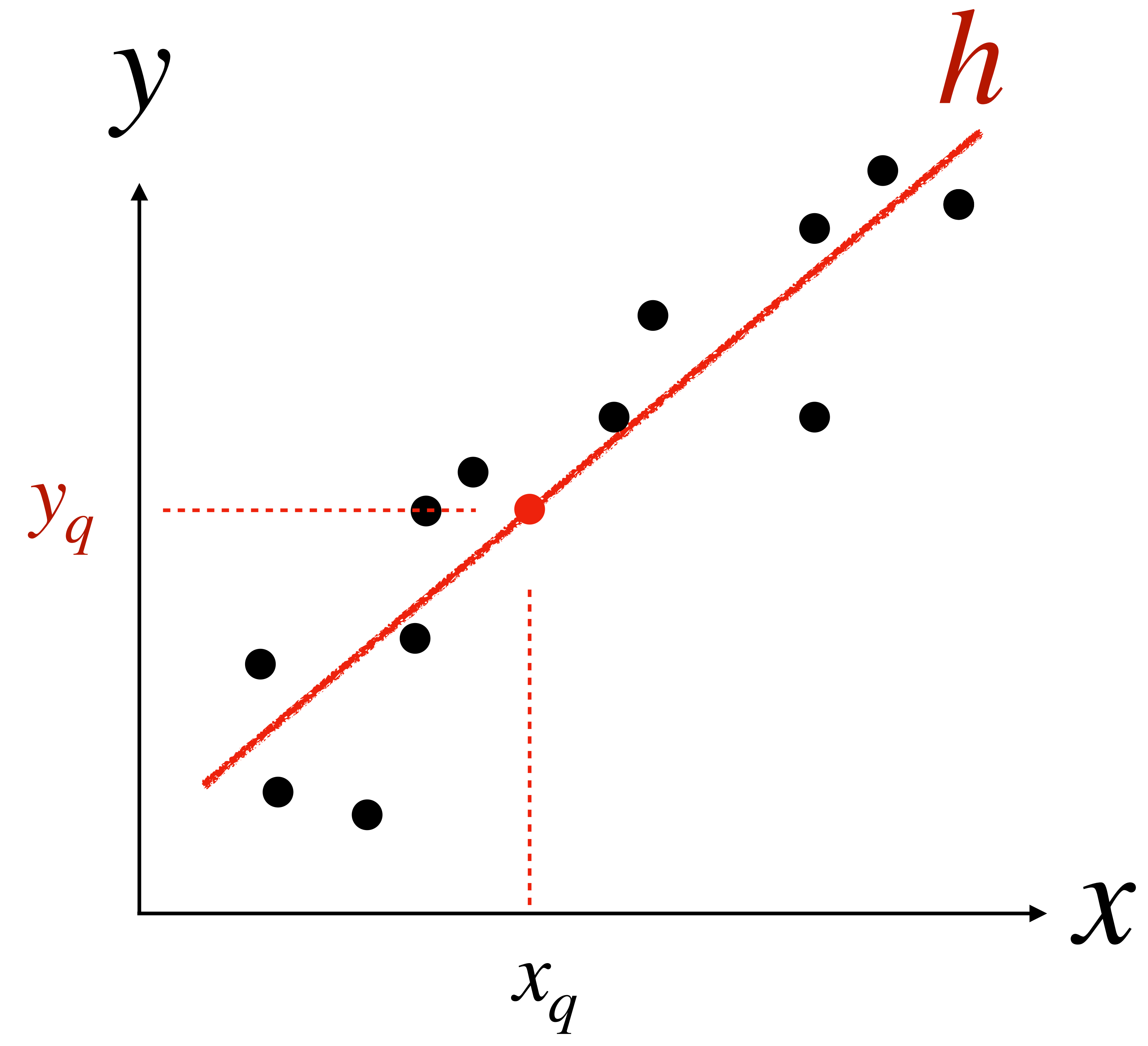
$$h_\theta(x) = \theta_0 + \theta_1 x$$

$$h_\theta(x) = [\theta_0, \theta_1] \cdot [1, x]$$

Unknown parameters | Input features

$$\theta \cdot \mathbf{x}$$

Linear function



What's the best $\theta = [\theta_0, \theta_1]$, given the data ?

What happens if we have **more inputs**?

Assume a linear hypothesis

$$\mathbf{x} \xrightarrow{h} y$$

$$h : \mathbb{R}^n \to \mathbb{R}$$

$$h_\theta(\mathbf{x}) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 + \ldots$$

$$h_\theta(\mathbf{x}) = \underbrace{[\theta_0, \theta_1, \theta_2, \theta_3, \ldots]}_{\theta} \cdot \underbrace{[1, x_1, x_2, x_3, \ldots]}_{\mathbf{x}}$$

weights

inputs

$$h_\theta(\mathbf{x}) = \theta \cdot \mathbf{x} = \theta^\top \mathbf{x}$$

Inputs		Output
x_1	x_2	y
$x_1^{(1)}$	$x_2^{(1)}$	$y^{(1)}$
$x_1^{(2)}$	$x_2^{(2)}$	$y^{(2)}$
$x_1^{(3)}$	$x_2^{(3)}$	$y^{(3)}$
$x_1^{(4)}$	$x_2^{(4)}$	$y^{(4)}$
$\vdots$	$\vdots$	$\vdots$

How do we pick the **best** parameters θ ?

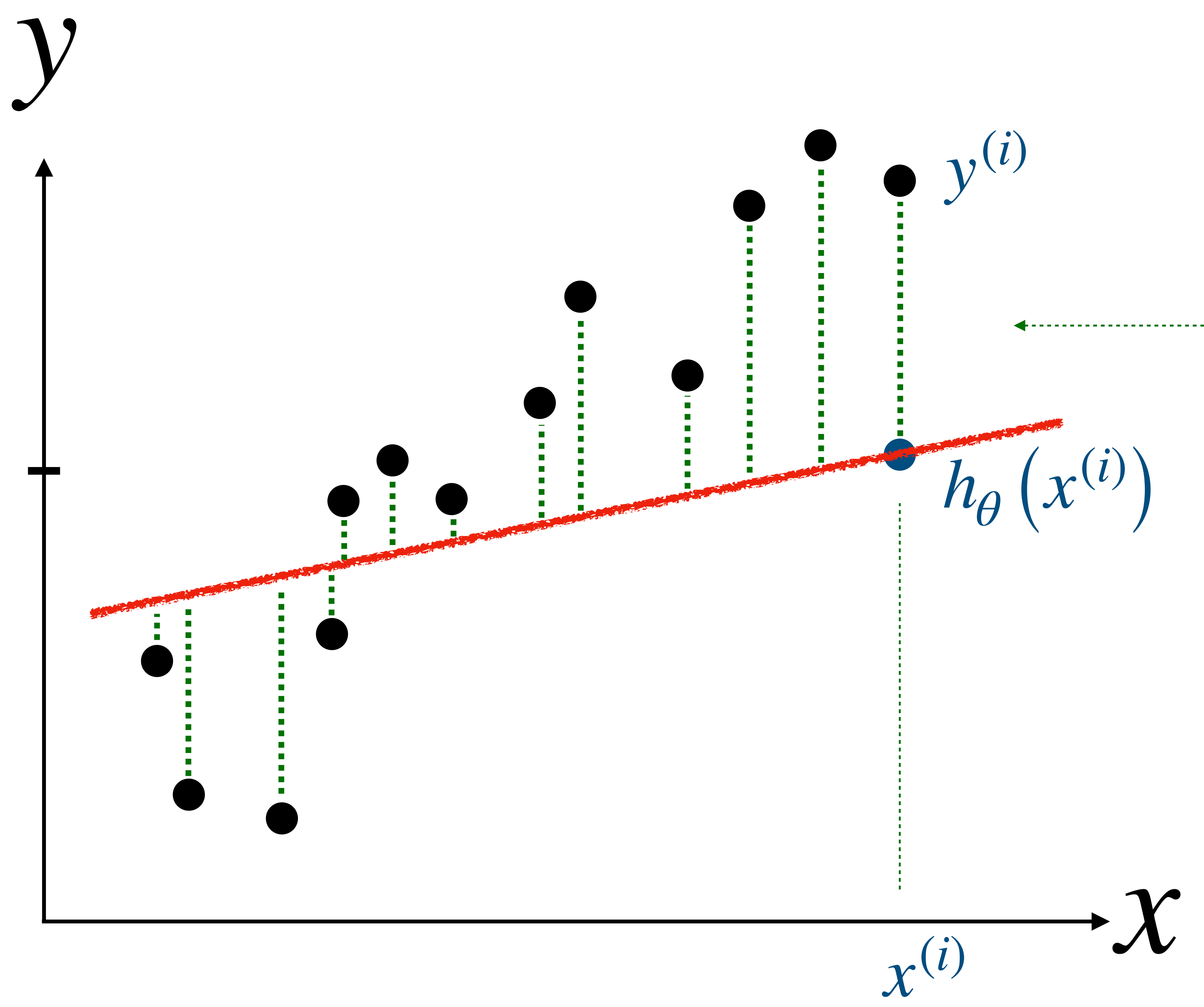
$$h_\theta(\mathbf{x}) = \theta^\top \mathbf{x} = \sum_{i=0}^{d} \theta_i x_i$$

Cost function

$$J(\theta) = \frac{1}{2} \sum_{i=1}^{d} \left(h_\theta\left(x^{(i)}\right) - y^{(i)} \right)^2$$

$$= \frac{1}{2} \sum_{i=1}^{d} \left(\theta^\top \mathbf{x}^{(i)} - y^{(i)} \right)^2$$

Ordinary least squares



distance $\left(h_\theta\left(x^{(i)}\right), y^{(i)} \right)$

$h_\theta\left(x^{(i)}\right) - y^{(i)}$	Residuals
$\left\lvert h_\theta\left(x^{(i)}\right) - y^{(i)} \right\rvert$	Absolute loss
$\left(h_\theta\left(x^{(i)}\right) - y^{(i)} \right)^2$	Square loss

Interactive Demo

https://colab.research.google.com/drive/1jEMvm_qlLneIeOFDC5Andr7JstIetVet?usp=sharing

Choose θ to minimize $J(\theta)$

Cost function

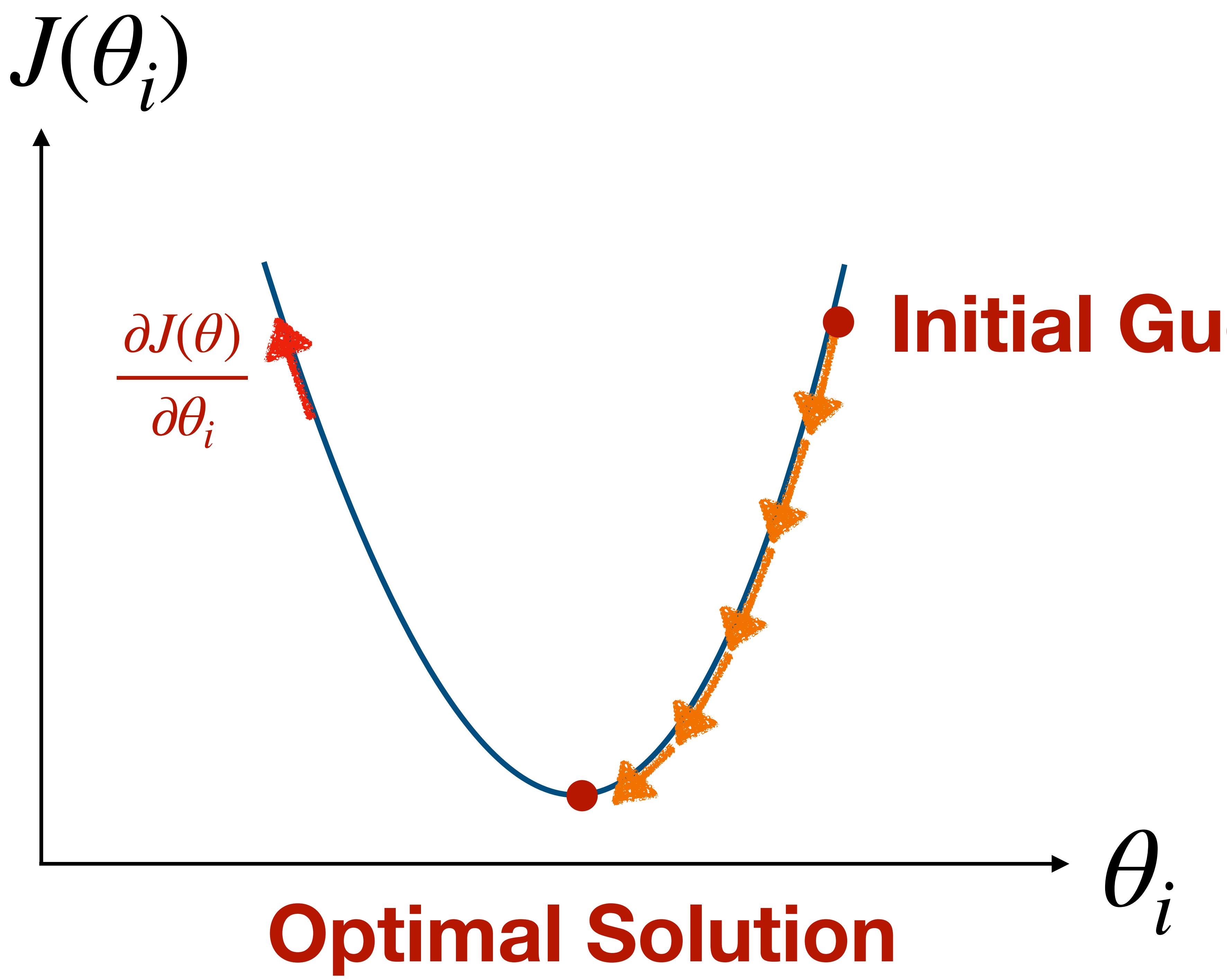
$$J(\theta) = \frac{1}{2}\sum_{i=1}^{d}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2$$

Gradient Descent Update

while not converged:

$$\theta_i := \theta_i - \alpha \frac{\partial J(\theta)}{\partial \theta_i}$$

Learning Rate



Gradient can be computed explicitly

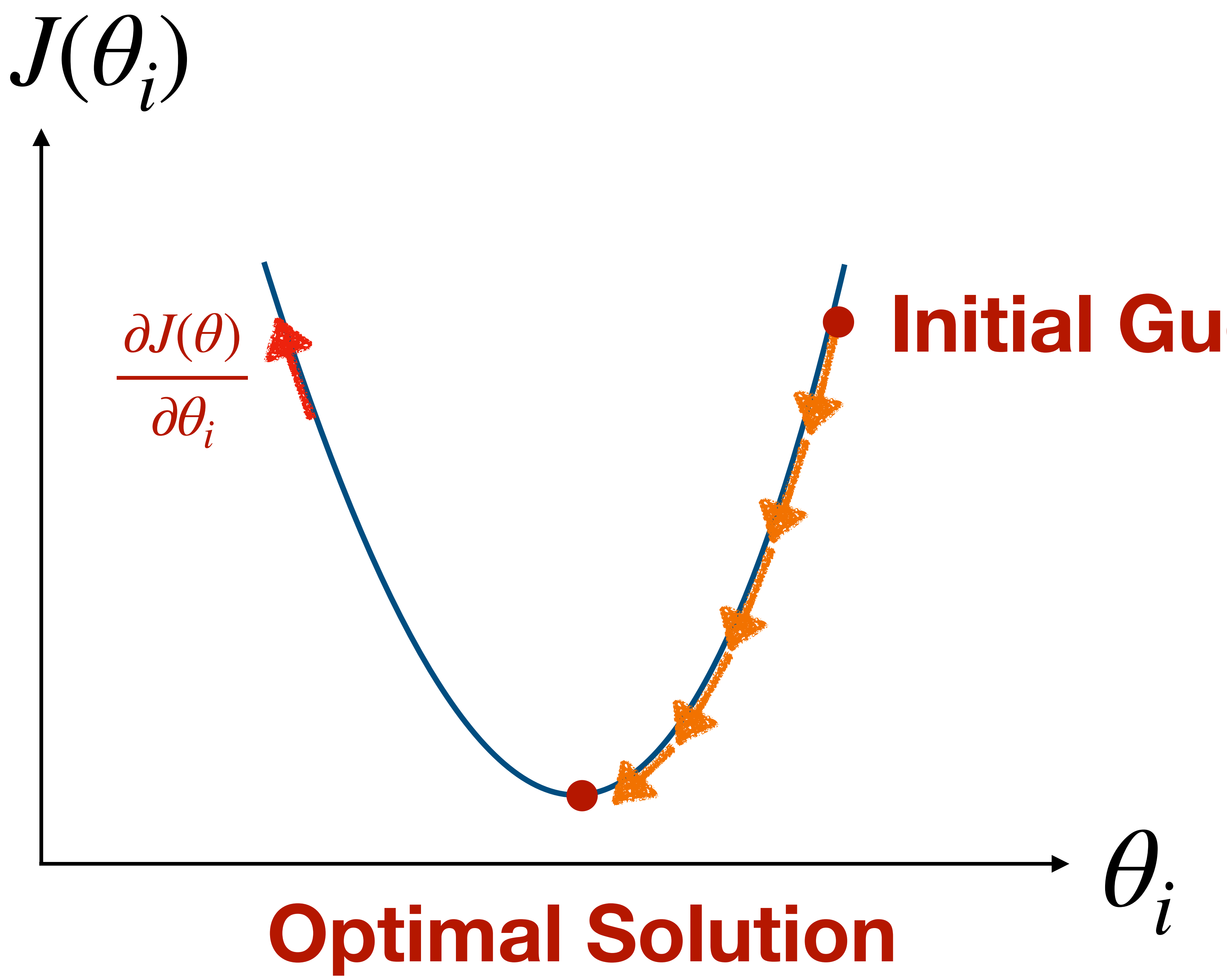
while not converged:

$$\theta_i := \theta_i - \alpha \frac{\partial J(\theta)}{\partial \theta_i}$$

Learning Rate

Derive $\dfrac{\partial J(\theta)}{\partial \theta_i}$ explicitly, for one (x, y) pair

Assume $y = \theta_0 x + \theta_1$



Least Mean Squares (LMS)

A.K.A **Widrow-Hoff** learning rule

For a single training example $\left(x^{(i)}, y^{(i)}\right)$:

$$\theta_j := \theta_j - \alpha\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)x_j^{(i)}$$

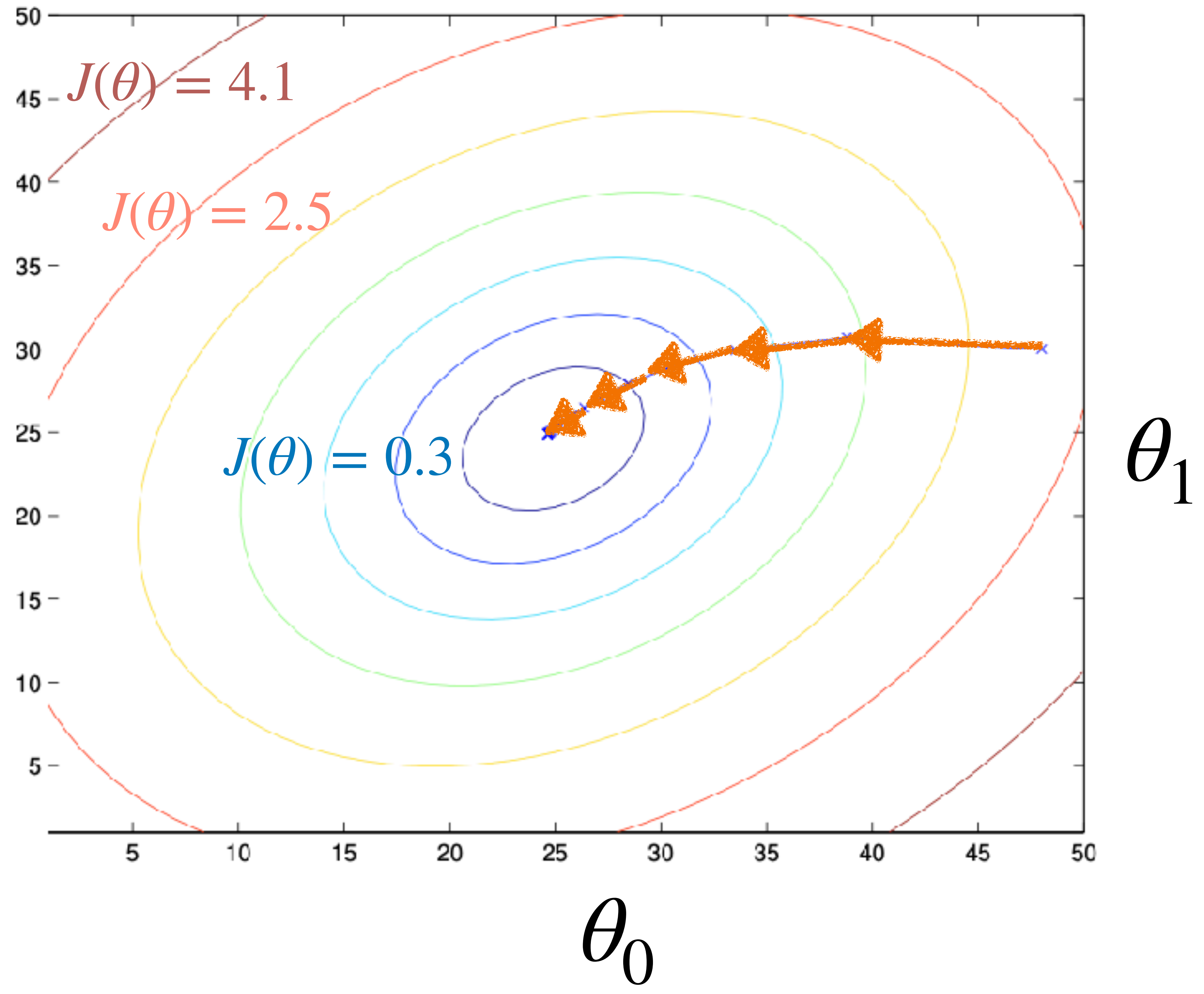
Batch Gradient Descent

for t = 1…T: (Epochs)

for all parameters j:

$$\theta_j := \theta_j - \alpha\sum_{i=1}^{n}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)x_j^{(i)}$$

Contour Plot



Stack and vectorize

Least Mean Squares (LMS)

Batch Gradient Descent (vectorized)

for t = 1...T:

$$\theta := \theta - \alpha \sum_{i=1}^{n} \left(h_\theta \left(x^{(i)} \right) - y^{(i)} \right) x^{(i)}$$

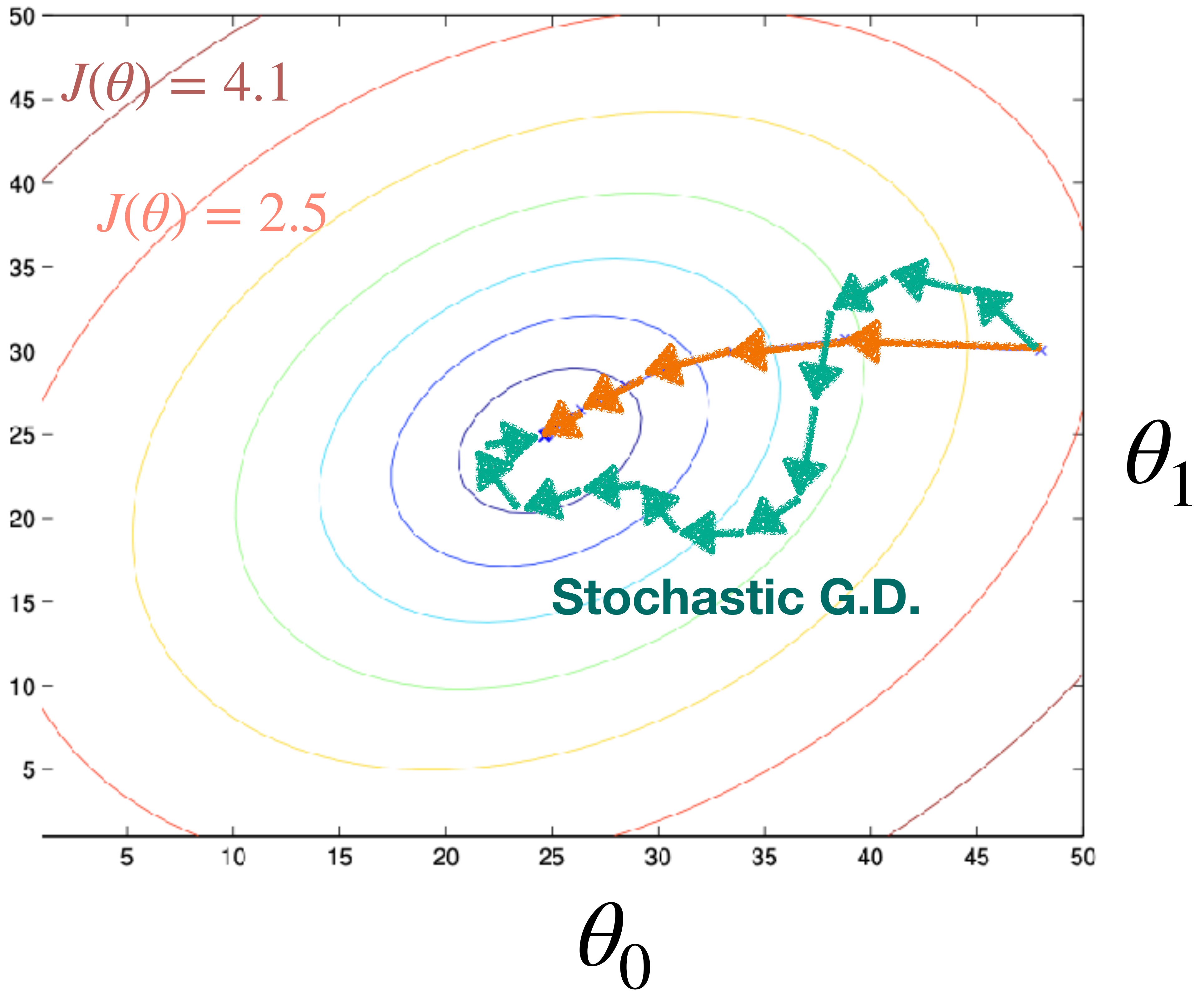
Stochastic Gradient Descent

for t = 1...T:

for i = 1...n:

$$\theta := \theta - \alpha \left(h_\theta \left(x^{(i)} \right) - y^{(i)} \right) x^{(i)}$$

Contour Plot



Summary

1. Assume a linear hypothesis

$$h_\theta(\mathbf{x}) = \theta^\top \mathbf{x} = \sum_{i=0}^{d} \theta_i x_i$$

2. Cost function

$$J(\theta) = \frac{1}{2} \sum_{i=1}^{d} \left(h_\theta \left(x^{(i)} \right) - y^{(i)} \right)^2$$

3. Minimize: Gradient Descent

$$\theta := \theta - \alpha \nabla_\theta J(\theta)$$

SGD

for t = 1…T:

  for i = 1…n:

$$\theta := \theta - \alpha \left(h_\theta \left(x^{(i)} \right) - y^{(i)} \right) x^{(i)}$$

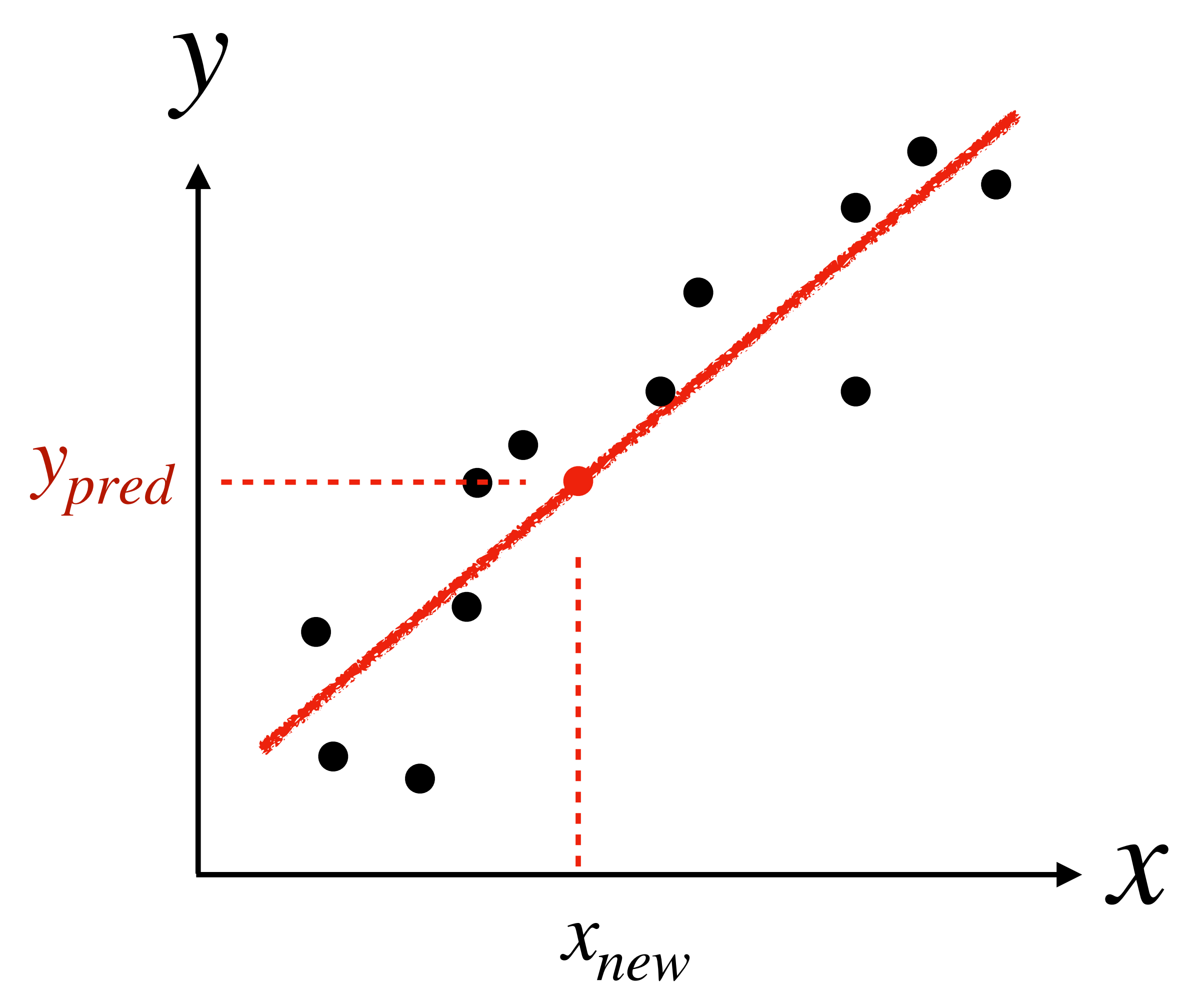
$\hat{\theta}$

4. Optimal predictor

$$y = h_{\hat{\theta}}(x)$$

5. Predict unseen data

$$y_{pred} = h_{\hat{\theta}}(x_{new})$$



Can you find the minimum analytically?

Design matrix

$$X = \begin{bmatrix} -- & x^{(1)} & -- \\ -- & x^{(2)} & -- \\ -- & \vdots & -- \\ -- & x^{(n)} & -- \end{bmatrix}$$

Parameters

$$\vec{\theta} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_m \end{bmatrix}$$

Output

$$\vec{y} = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(n)} \end{bmatrix}$$

Minimize

$$J(\theta) = \frac{1}{2} \| X\theta - \vec{y} \|_2^2 \longrightarrow \nabla_\theta J(\theta) = 0$$

Normal Equation

$$\theta = \left(X^\top X\right)^{-1} X^\top \vec{y}$$

Probabilistic Interpretation

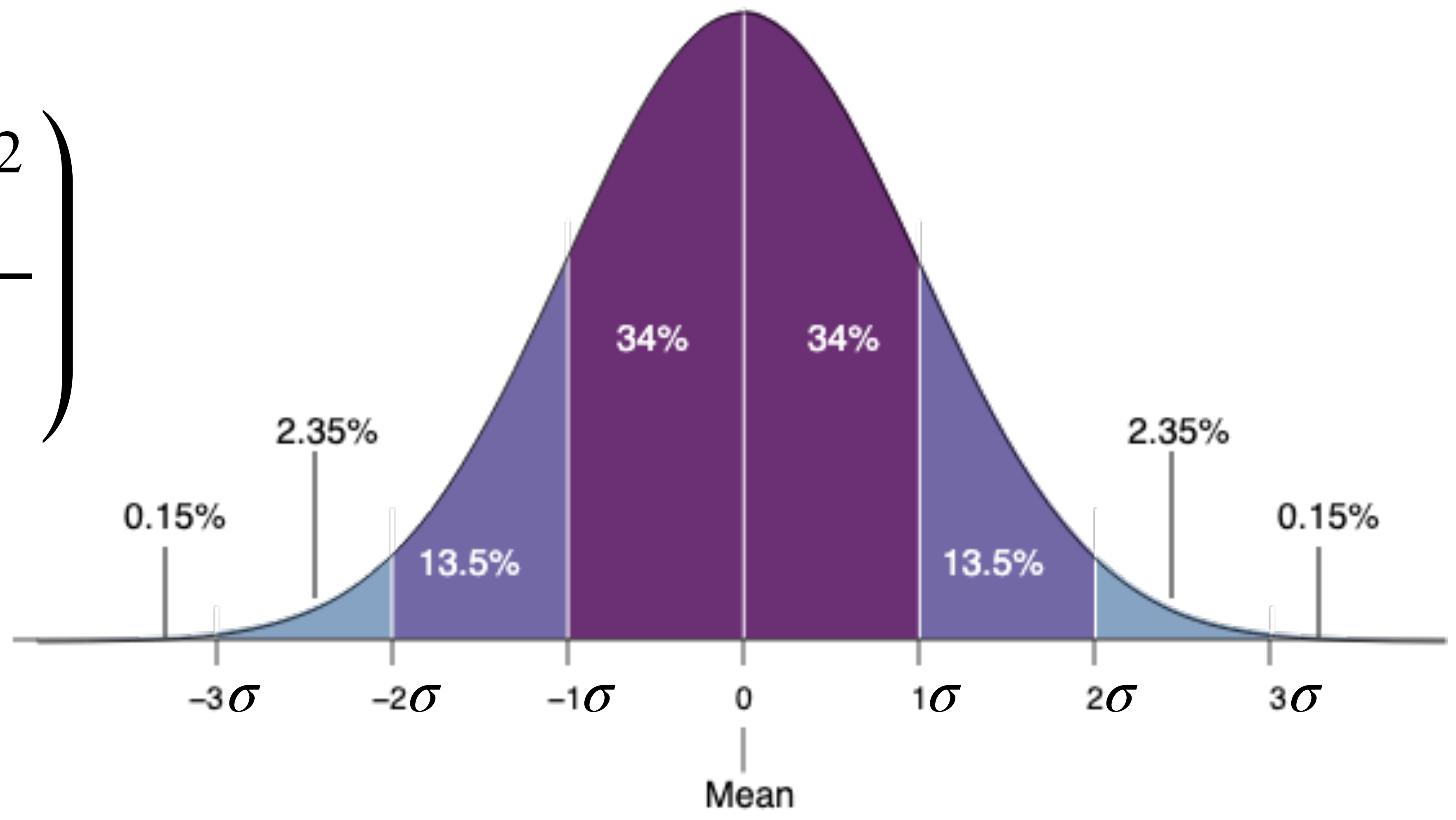
Assume noise is normally distributed around model

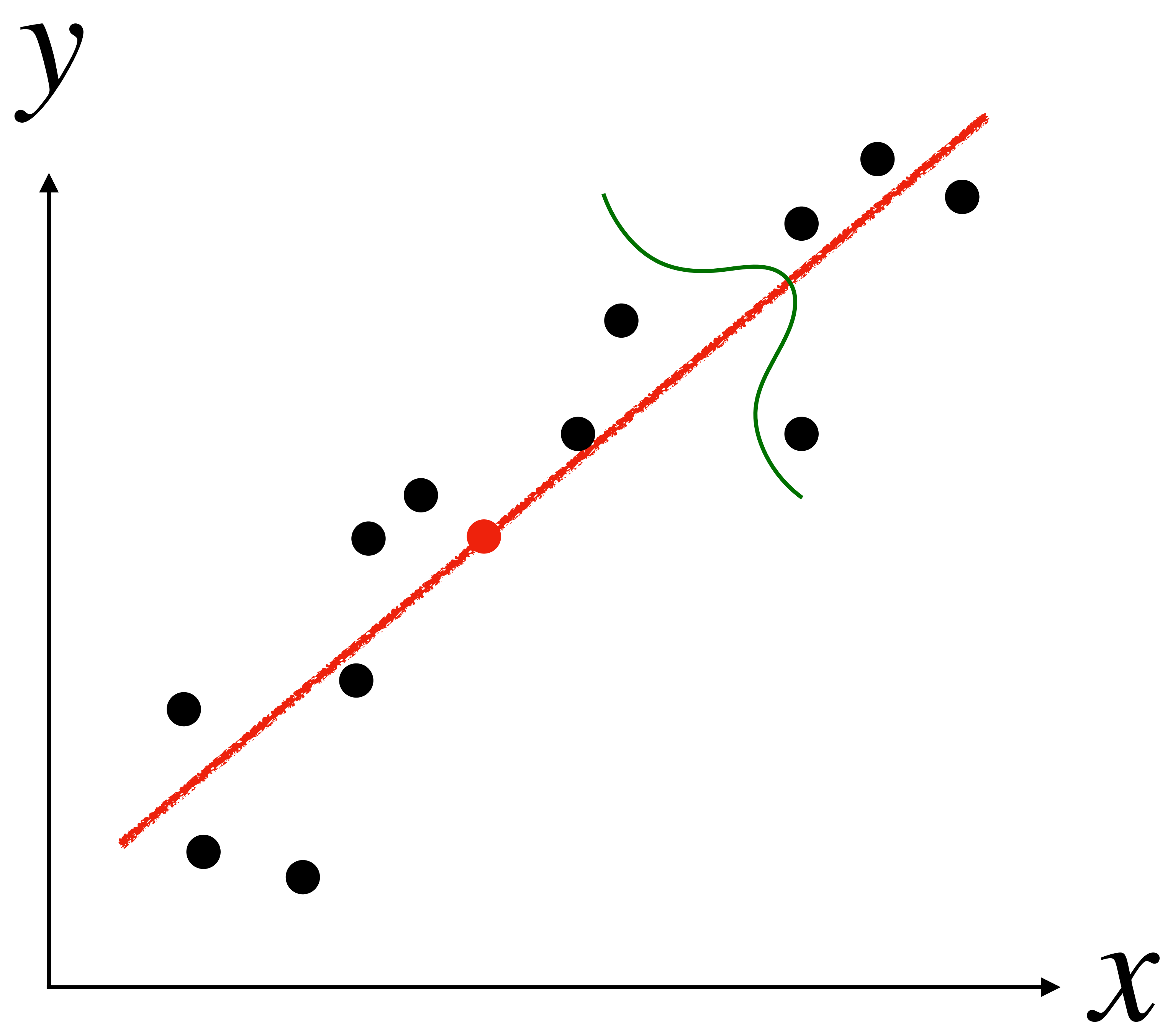
$$y^{(i)} = \theta^\top x^{(i)} + \varepsilon^{(i)}$$

Normally distributed

$$\mathcal{N}\left(0, \sigma^2\right)$$

$$p\left(\varepsilon^{(i)}\right) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\left(\varepsilon^{(i)}\right)^2}{2\sigma^2}\right)$$





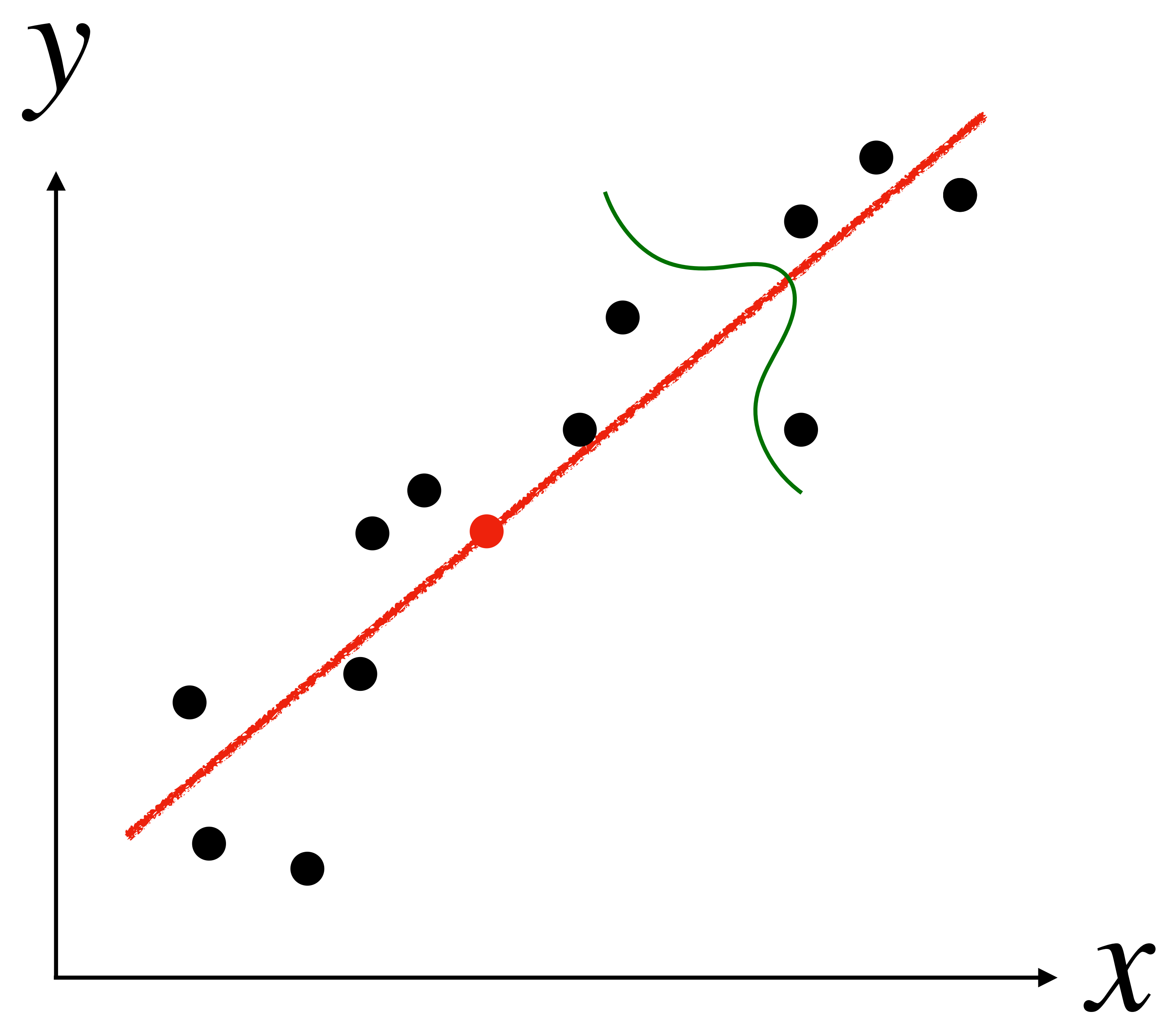
Probabilistic Interpretation

Assume noise is normally distributed around model

$$y^{(i)} = \theta^\top x^{(i)} + \varepsilon^{(i)}$$

$$p\left(\varepsilon^{(i)}\right) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\left(\varepsilon^{(i)}\right)^2}{2\sigma^2}\right)$$

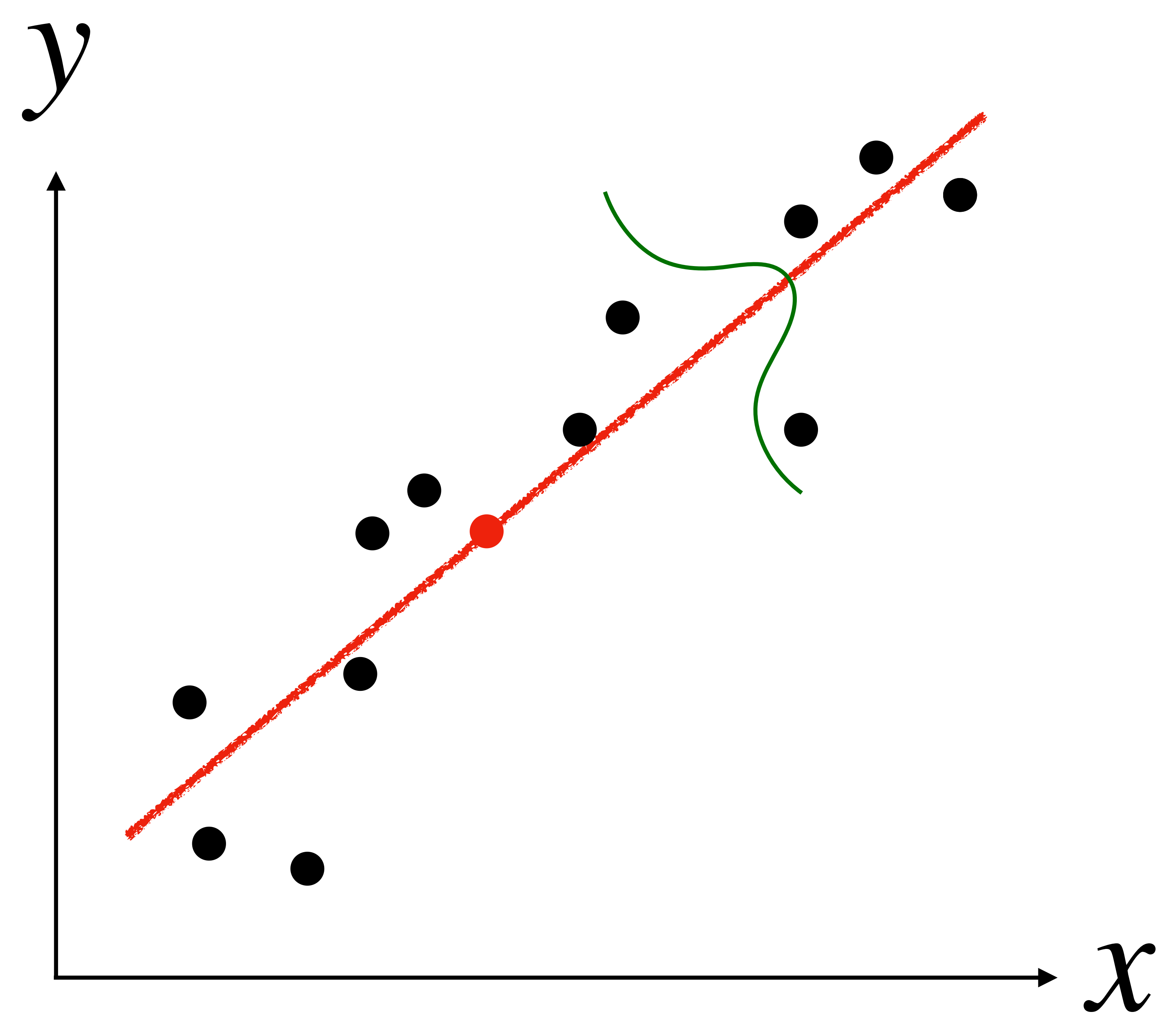
$$p\left(y^{(i)} \mid x^{(i)}; \theta\right) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\left(y^{(i)} - \theta^\top x^{(i)}\right)^2}{2\sigma^2}\right)$$



Likelihood of output given input

$$L(\theta) = \prod_{i=1}^{n} p\left(y^{(i)} \,|\, x^{(i)}; \theta\right)$$

$$= \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\left(y^{(i)} - \theta^\top x^{(i)}\right)^2}{2\sigma^2}\right)$$



Log-likelihood

$$l(\theta) = \log L(\theta)$$

$$= \log \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\left(y^{(i)} - \theta^\top x^{(i)}\right)^2}{2\sigma^2}\right)$$

$$= \sum_{i=1}^{n} \log \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\left(y^{(i)} - \theta^\top x^{(i)}\right)^2}{2\sigma^2}\right) = n \log \frac{1}{\sqrt{2\pi}\sigma} - \frac{1}{2\sigma^2} \sum_{i=1}^{n} \left(y^{(i)} - \theta^\top x^{(i)}\right)^2$$

Maximize Log-likelihood

$$l(\theta) = \log L(\theta)$$

$$= \sum_{i=1}^{n} \log \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\left(y^{(i)} - \theta^\top x^{(i)}\right)^2}{2\sigma^2}\right)$$

$$= n \log \frac{1}{\sqrt{2\pi}\sigma} - \frac{1}{2\sigma^2} \sum_{i=1}^{n} \left(y^{(i)} - \theta^\top x^{(i)}\right)^2$$

Maximize $l(\theta)$ $\longrightarrow$ **Minimize** $\frac{1}{2} \sum_{i=1}^{n} \left(y^{(i)} - \theta^\top x^{(i)}\right)^2$