# 10-301/601: Introduction to Machine Learning Lecture 11 – Linear Regression

Henry Chai

5/20/25

# Recall: Regression

- Learning to diagnose heart disease

  as a **(supervised)** **<u>regression</u> task**

features / targets / data points

| $x_1$ Family History | $x_2$ Resting Blood Pressure | $x_3$ Cholesterol | $y$ Heart Disease? |
|---|---|---|---|
| Yes | Low | Normal | \$0 |
| No | Medium | Normal | \$20 |
| No | Low | Abnormal | \$30 |
| Yes | Medium | Normal | \$100 |
| Yes | High | Abnormal | \$5000 |

# Decision Tree Regression

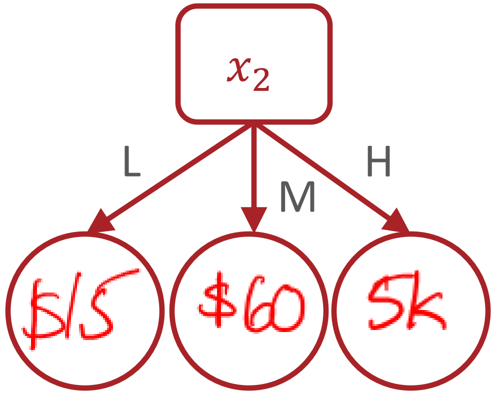
- Learning to diagnose heart disease

as a **(supervised)** **<u>regression</u> task**

features | targets

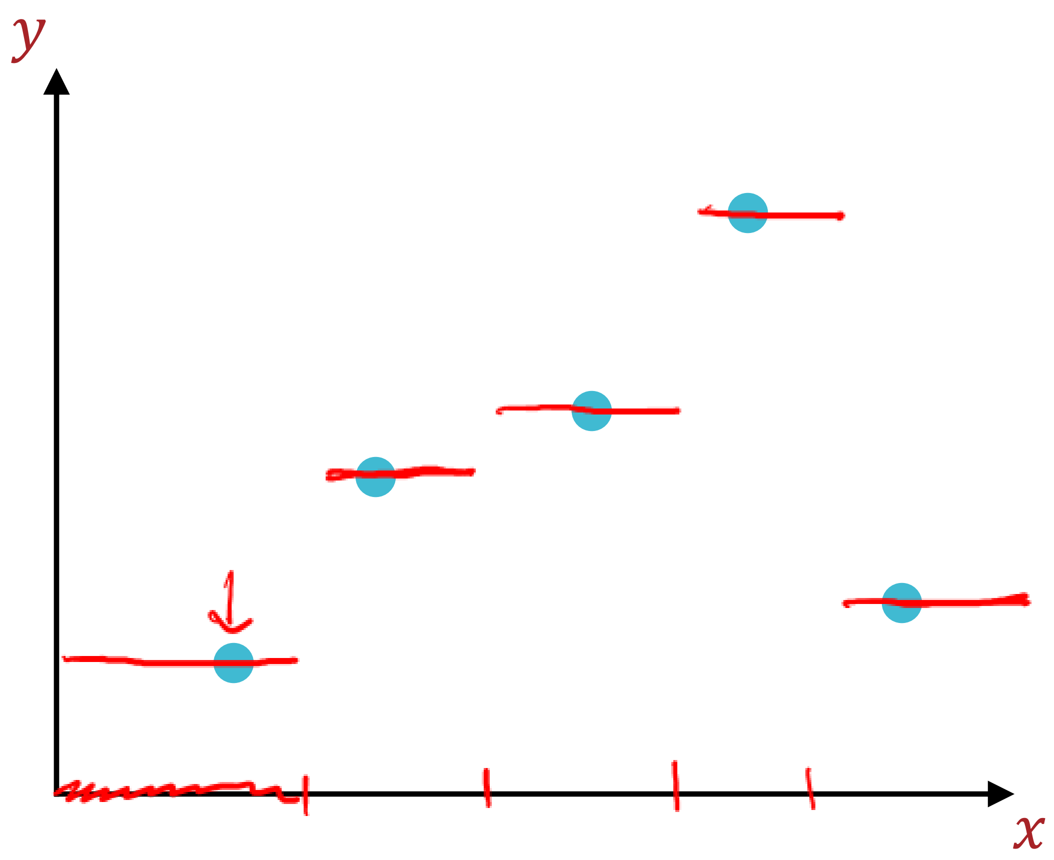
data points

| $x_1$ Family History | $x_2$ Resting Blood Pressure | $x_3$ Cholesterol | $y$ Heart Disease? |
|---|---|---|---|
| Yes | Low | Normal | \$0 |
| No | Medium | Normal | \$20 |
| No | Low | Abnormal | \$30 |
| Yes | Medium | Normal | \$100 |
| Yes | High | Abnormal | \$5000 |



# 1-NN Regression?

- Suppose we have real-valued targets $y \in \mathbb{R}$ and one-dimensional inputs $x \in \mathbb{R}$



# Linear Regression

- Suppose we have real-valued targets $y \in \mathbb{R}$ and $D$-dimensional inputs $\boldsymbol{x} = [1, x_1, \ldots, x_D]^T \in \mathbb{R}^{D+1}$

- **Assume**

$$y = \boldsymbol{\theta}^T \boldsymbol{x} = [w_0 \ \boldsymbol{w}]^T \begin{bmatrix} 1 \\ \boldsymbol{x} \end{bmatrix}$$

- Notation: given training data $\mathcal{D} = \left\{\left(\boldsymbol{x}^{(n)}, y^{(n)}\right)\right\}_{n=1}^{N}$

Design matrix: $X = \begin{bmatrix} 1 & \vec{x}^{(1)T} \\ 1 & \vec{x}^{(2)T} \\ \vdots & \vdots \\ 1 & \vec{x}^{(N)T} \end{bmatrix} = \begin{bmatrix} 1 & x_1^{(1)} & x_2^{(1)} & \cdots & x_D^{(1)} \\ & & \vdots & & \\ 1 & x_1^{(N)} & & \cdots & x_D^{(N)} \end{bmatrix} \in \mathbb{R}^{N \times (D+1)}$

Target vector: $\vec{y} = [y^{(1)}, y^{(2)}, \ldots, y^{(N)}]^T$

# General Recipe for Machine Learning

- Define a model and model parameters

- Write down an objective function

- Optimize the objective w.r.t. the model parameters

# Recipe for Linear Regression

- Define a model and model parameters
  - Assume $y = \vec{\theta}^T \vec{x}$
  - Parameters: $\vec{\theta} = [w_0, w_1, w_2, \ldots, w_D]^T$
- Write down an objective function
  - Minimize the mean squared error
  $$\ell_D(\vec{\theta}) = \frac{1}{N} \sum_{n=1}^{N} \left(\vec{\theta}^T \vec{x}^{(n)} - y^{(n)}\right)^2$$
- ★ Optimize the objective w.r.t. the model parameters
  1. Solve in closed form using the critical point method OR
  2. Gradient descent

$$f(x) = cx \Rightarrow \frac{df}{dx} = c$$

$$f(x) = ax^2 \Rightarrow \frac{df}{dx} = 2ax$$

# Minimizing the Squared Error

$$\ell_{\mathcal{D}}(\boldsymbol{\theta}) = \frac{1}{N}\sum_{n=1}^{N}\left(\boldsymbol{\theta}^T\boldsymbol{x}^{(n)} - y^{(n)}\right)^2 = \frac{1}{N}\sum_{n=1}^{N}\left(\vec{x}^{(n)T}\vec{\theta} - y^{(n)}\right)^2$$

$$= \frac{1}{N}\underbrace{(X\vec{\theta} - \vec{y})}_{\text{"residuals"}}^T(X\vec{\theta} - \vec{y})$$

$$= \frac{1}{N}\left(\vec{\theta}^T X^T X\vec{\theta} - 2\vec{\theta}^T X^T\vec{y} + \vec{y}^T\vec{y}\right)$$

$$\nabla_{\vec{\theta}}\,\ell_{\mathcal{D}}(\vec{\theta}) = \frac{1}{N}\left(2X^TX\vec{\theta} - 2X^T\vec{y} + \vec{0}\right) = \begin{bmatrix} \partial\ell_{\mathcal{D}}/\partial w_0 \\ \partial\ell_{\mathcal{D}}/\partial w_1 \\ \vdots \end{bmatrix}$$

$$\Rightarrow \frac{1}{N}\left(2X^TX\hat{\vec{\theta}} - 2X^T\vec{y}\right) = 0$$

$$\Rightarrow 2X^TX\hat{\vec{\theta}} = 2X^T\vec{y}$$

$$\Rightarrow \hat{\vec{\theta}} = (X^TX)^{-1}X^T\vec{y}$$

$$\hat{\boldsymbol{\theta}} = (X^T X)^{-1} X^T \boldsymbol{y}$$

# Closed Form Solution

1. Is $X^T X$ invertible?

2. If so, how computationally expensive is inverting $X^T X$?

# Is $X^T X$ always invertible?

Yes

No

Unsure

# If $X^T X$ is invertible, how computationally expensive is it to invert?

- $O(N^2)$
- $O(D^2)$
- $O(ND)$
- $O(N^3)$
- $O(D^3)$

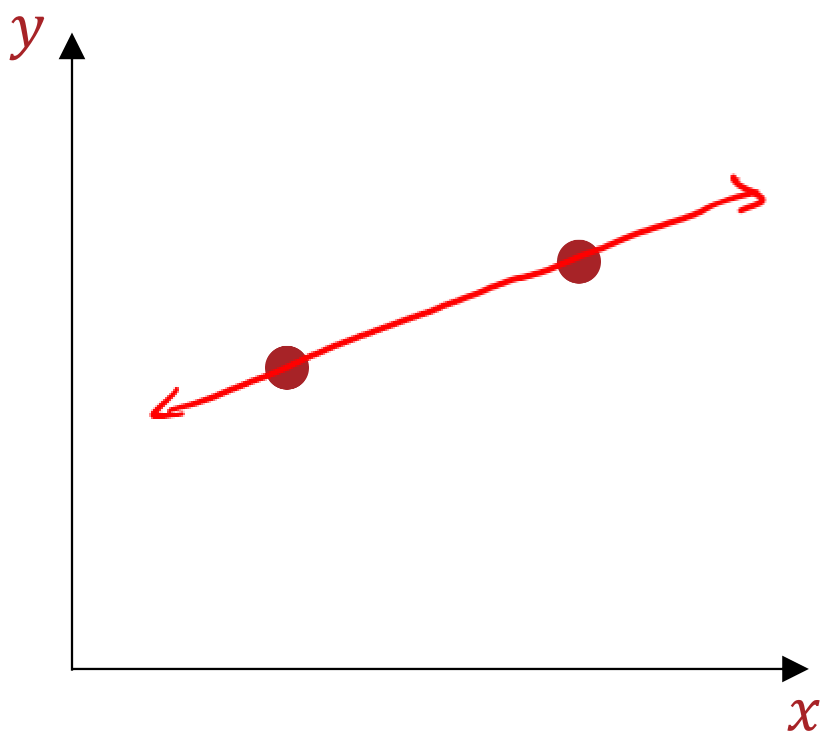
$$\hat{\boldsymbol{\theta}} = (X^T X)^{-1} X^T \boldsymbol{y}$$

# Closed Form Solution

1. Is $X^T X$ invertible?
   - When $N \gg D + 1$, $X^T X$ is (almost always) full rank and therefore, invertible!
   - If $X^T X$ is not invertible (occurs when one of the features is a linear combination of the others), what does that imply about our problem?
2. If so, how computationally expensive is inverting $X^T X$?
   - $X^T X \in \mathbb{R}^{D+1 \times D+1}$ so inverting $X^T X$ takes $O(D^3)$ time…
     - Computing $X^T X$ takes $O(ND^2)$ time
   - What alternative optimization method(s) can we use to minimize the mean squared error?

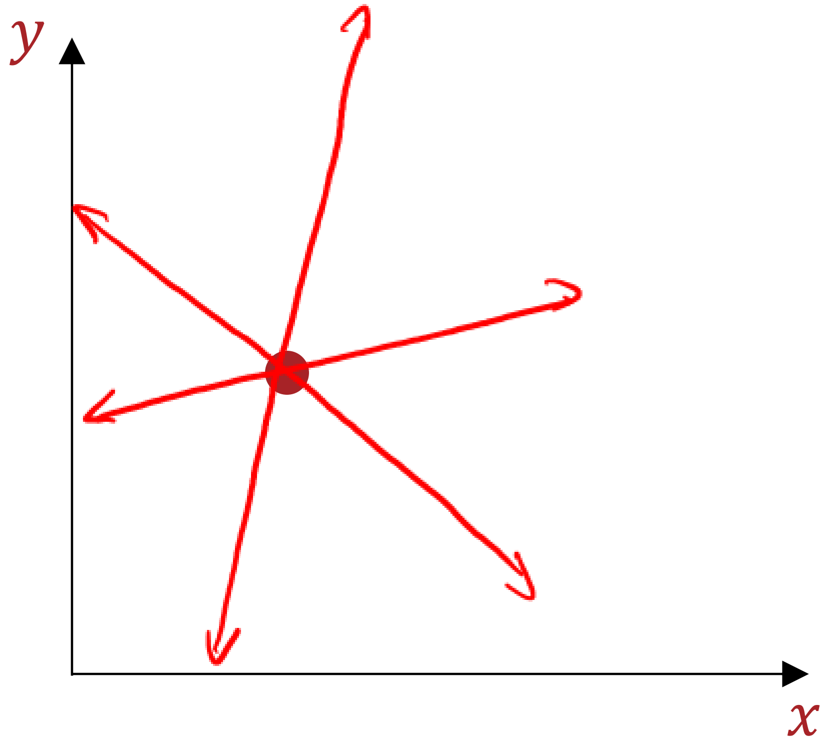
# Linear Regression: Uniqueness

- Consider a 1D linear regression model trained to minimize the mean squared error: how many optimal solutions (i.e., sets of parameters $\boldsymbol{\theta}$) are there for the given dataset?



# Linear Regression: Uniqueness

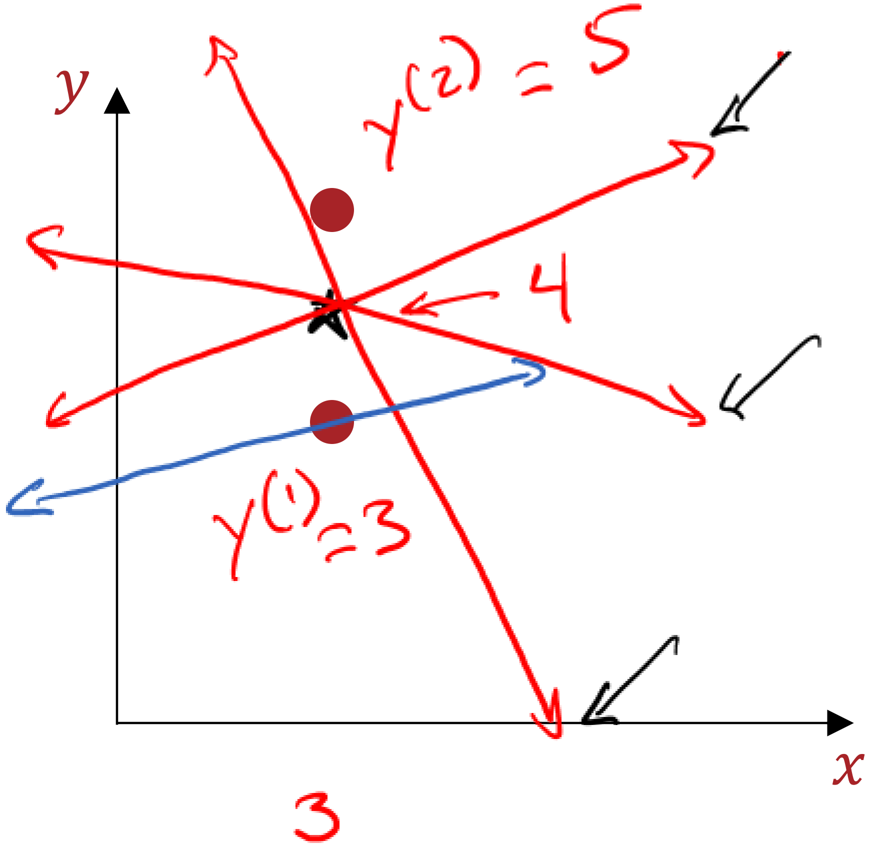
- Consider a 1D linear regression model trained to minimize the mean squared error: how many optimal solutions (i.e., sets of parameters $\boldsymbol{\theta}$) are there for the given dataset?



$$\frac{1}{2}\left((4-3)^2 + (4-5)^2\right) = 1$$

# Linear Regression: Uniqueness

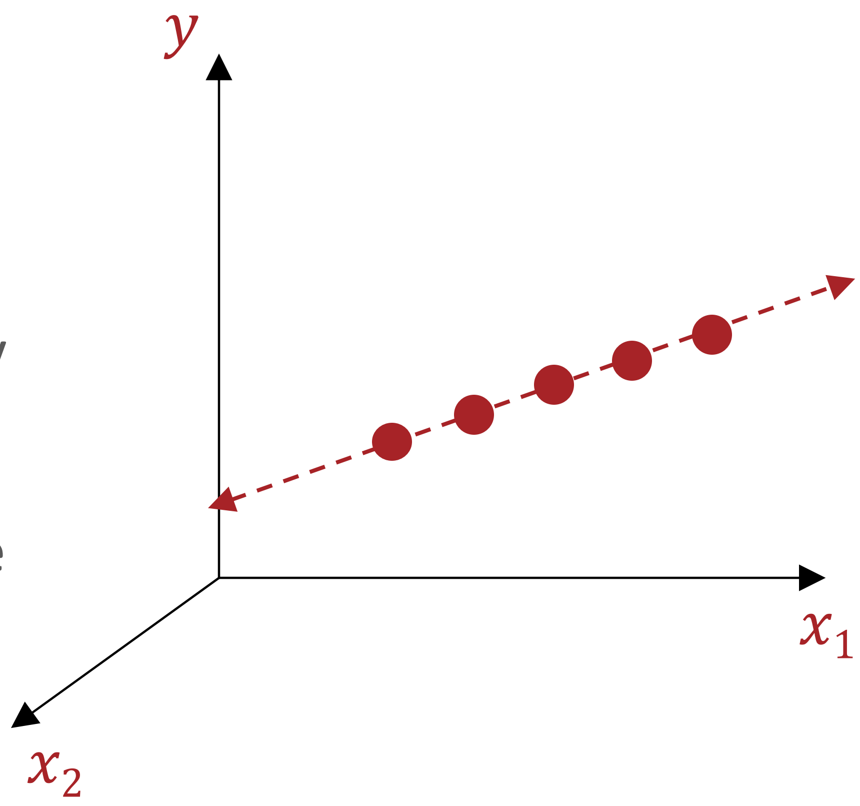
- Consider a 1D linear regression model trained to minimize the mean squared error: how many optimal solutions (i.e., sets of parameters $\boldsymbol{\theta}$) are there for the given dataset?




$$\frac{1}{2}\left((3-3)^2 + (3-5)^2\right) = 2$$

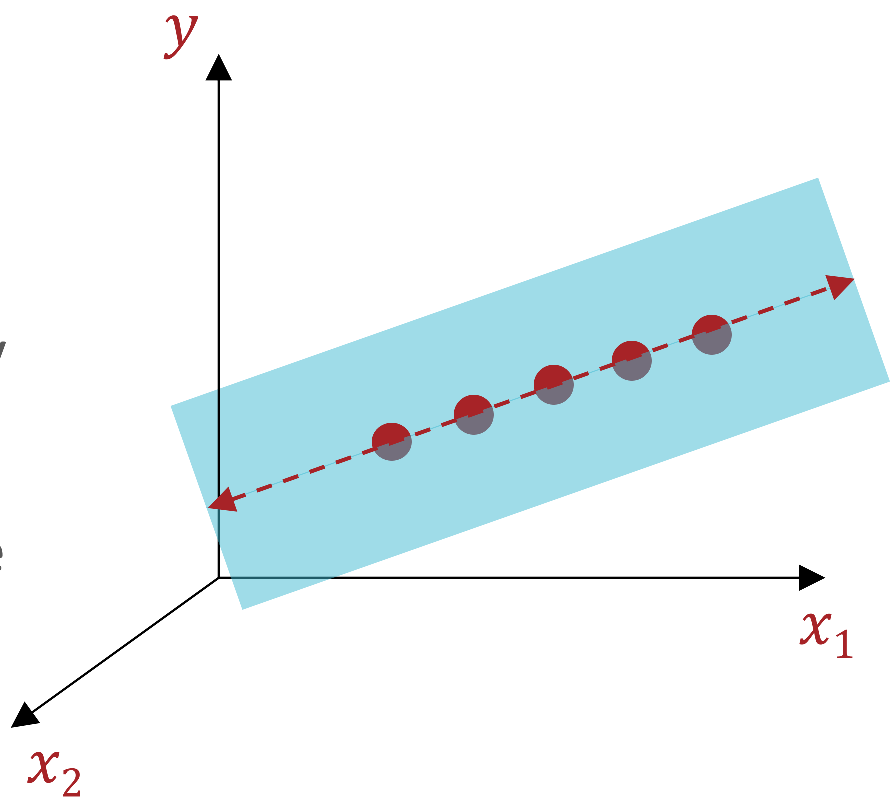
# Linear Regression: Uniqueness

- Consider a 2D linear regression model trained to minimize the mean squared error: how many optimal solutions (i.e., sets of parameters $\boldsymbol{\theta}$) are there for the given dataset?



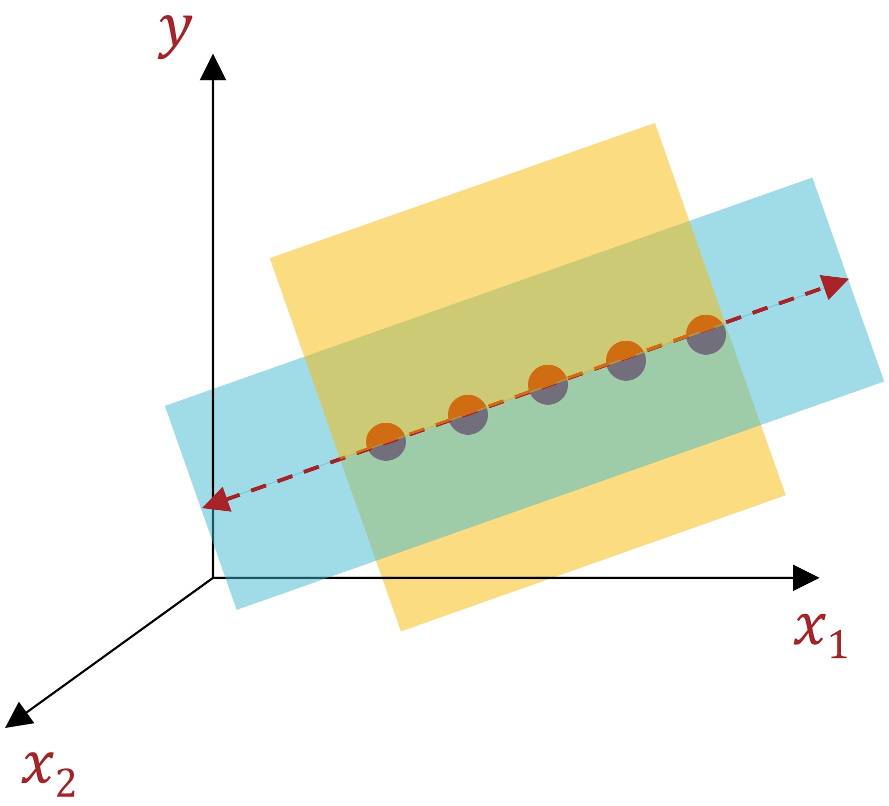
# Linear Regression: Uniqueness

- Consider a 2D linear regression model trained to minimize the mean squared error: how many optimal solutions (i.e., sets of parameters $\boldsymbol{\theta}$) are there for the given dataset?



# Linear Regression: Uniqueness

- Consider a 2D linear regression model trained to minimize the mean squared error: how many optimal solutions (i.e., sets of parameters $\boldsymbol{\theta}$) are there for the given dataset?



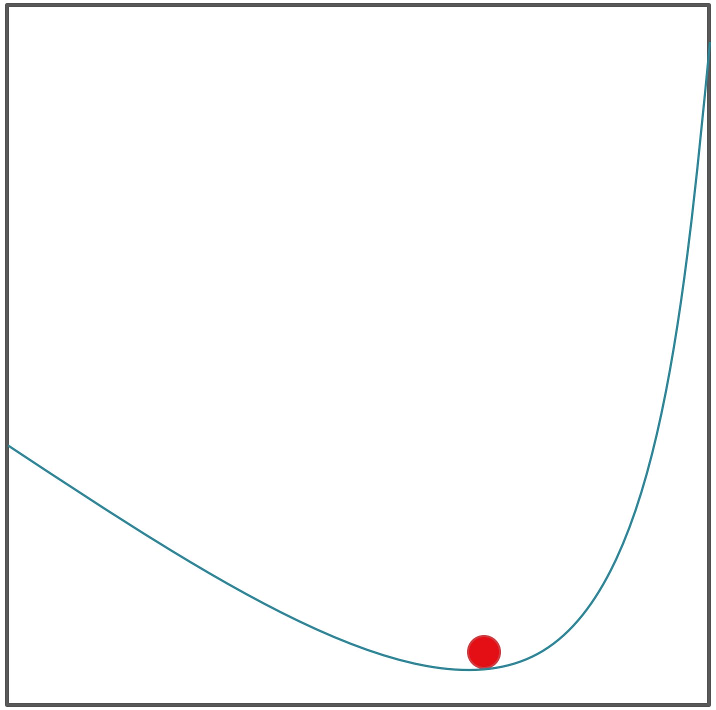
$$\hat{\boldsymbol{\theta}} = (X^T X)^{-1} X^T \boldsymbol{y}$$

# Closed Form Solution

1. Is $X^T X$ invertible?
   - When $N \gg D + 1$, $X^T X$ is (almost always) full rank and therefore, invertible!
   - If $X^T X$ is not invertible (occurs when one of the features is a linear combination of the others) then there are infinitely many solutions.
2. If so, how computationally expensive is inverting $X^T X$?
   - $X^T X \in \mathbb{R}^{D+1 \times D+1}$ so inverting $X^T X$ takes $O(D^3)$ time…
     - Computing $X^T X$ takes $O(ND^2)$ time
   - Can use gradient descent to (potentially) speed things up when $N$ and $D$ are large!

# Gradient Descent: Intuition

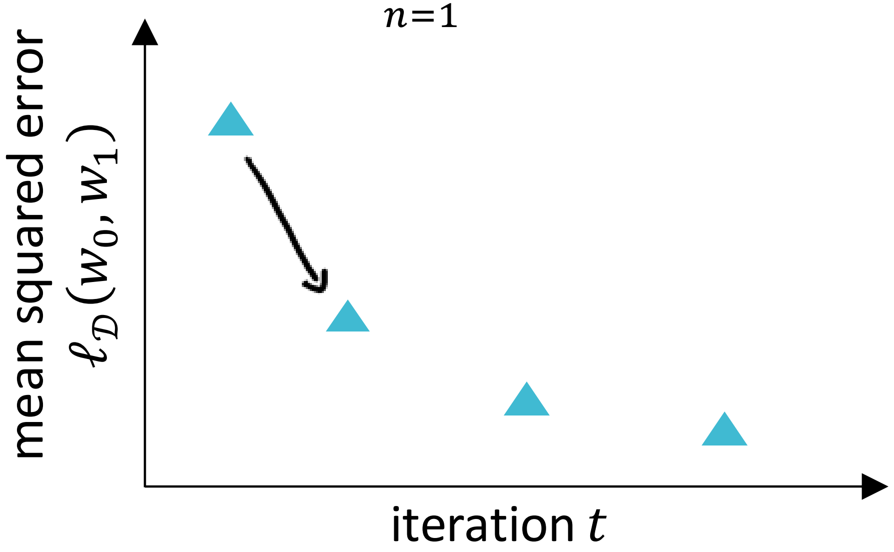
- An iterative method for minimizing functions
- Requires the gradient to exist everywhere

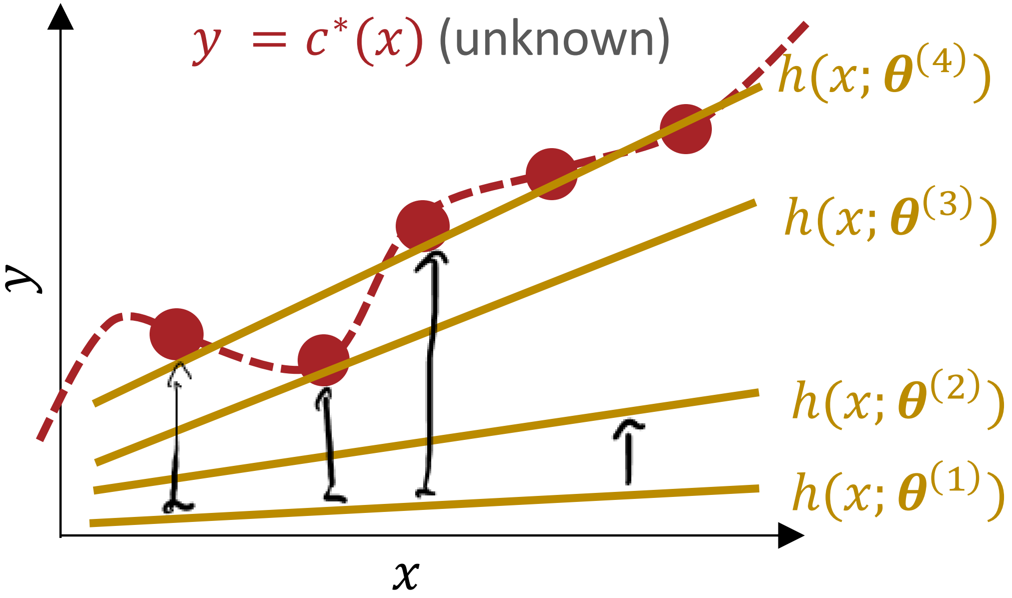


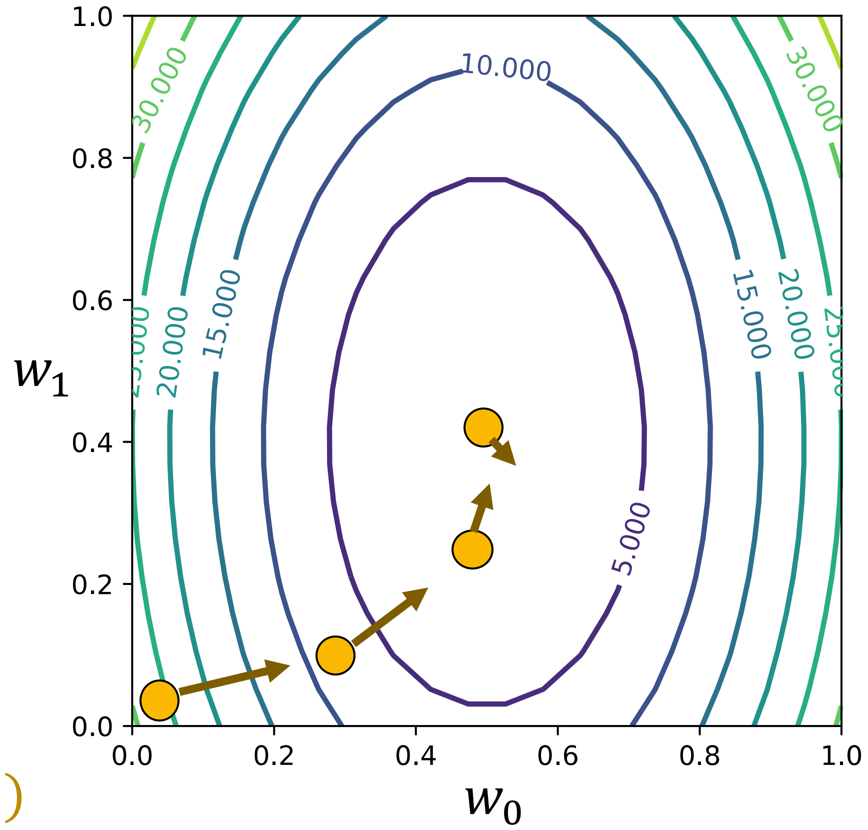
- Good news: the squared error is also convex!

# Gradient Descent for Linear Regression

$$\ell_{\mathcal{D}}(w_0, w_1) = \frac{1}{N}\sum_{n=1}^{N}\left(w_1 x^{(n)} + w_0 - y^{(n)}\right)^2$$



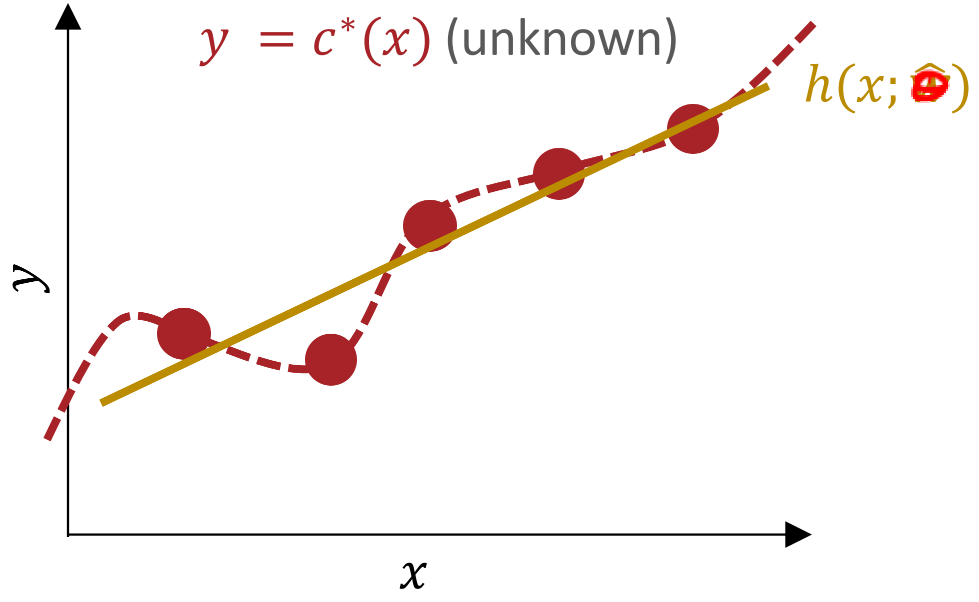



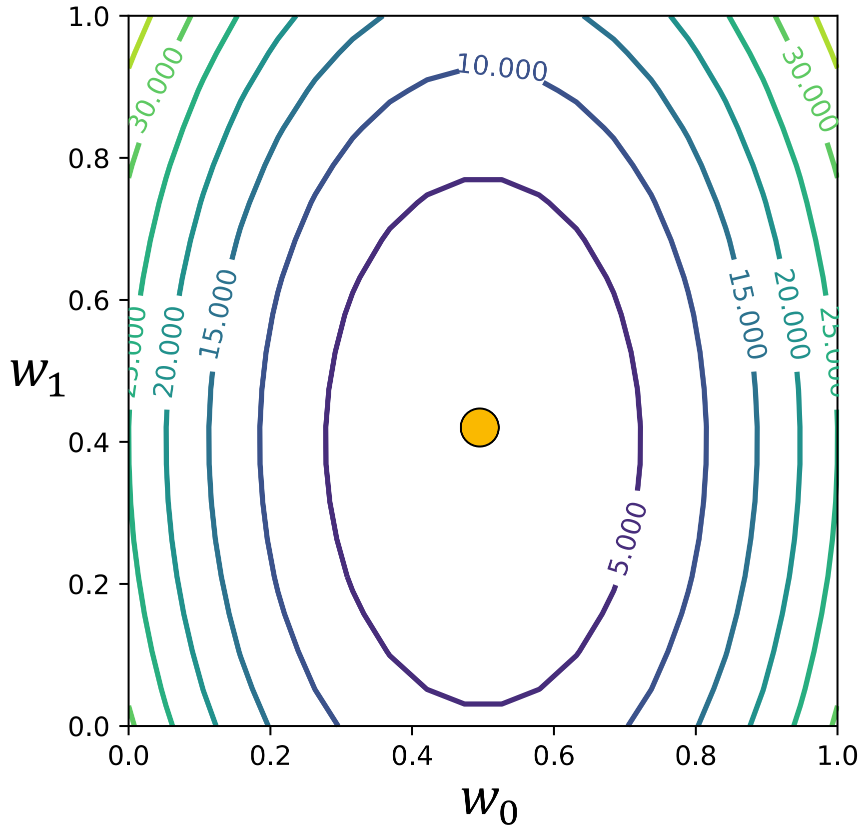




| $t$ | $w_0$ | $w_1$ | $\ell_{\mathcal{D}}(w_0, w_1)$ |
|---|---|---|---|

# Closed Form Optimization

$$\hat{\boldsymbol{\theta}} = (X^T X)^{-1} X^T \boldsymbol{y}$$




| $t$ | $w_0$ | $w_1$ | $\ell_{\mathcal{D}}(w_0, w_1)$ |
|---|---|---|---|
| 1 | 0.59 | 0.43 | 0.2 |

# Key Takeaways

- Decision tree and $k$NN regression
- Closed form solution for linear regression
  - Setting partial derivative/gradients to 0 and solving for critical points
  - Potential issues with the closed form solution: invertibility and computational costs