6.036/6.862: Introduction to Machine Learning

Lecture: starts Tuesdays 9:35am (Boston time zone)

Course website: introml.odl.mit.edu

Who's talking? Prof. Tamara Broderick

Questions? discourse.odl.mit.edu ("Lecture 5" category)

Materials: Will all be available at course website

Last Time(s)

I. Linear logistic classification/logistic regression
II. Gradient descent

Today's Plan

I. Linear regression
II. Ridge regression
III. Gradient descent & stochastic gradient descent

Recall

Recall

Classification

Recall

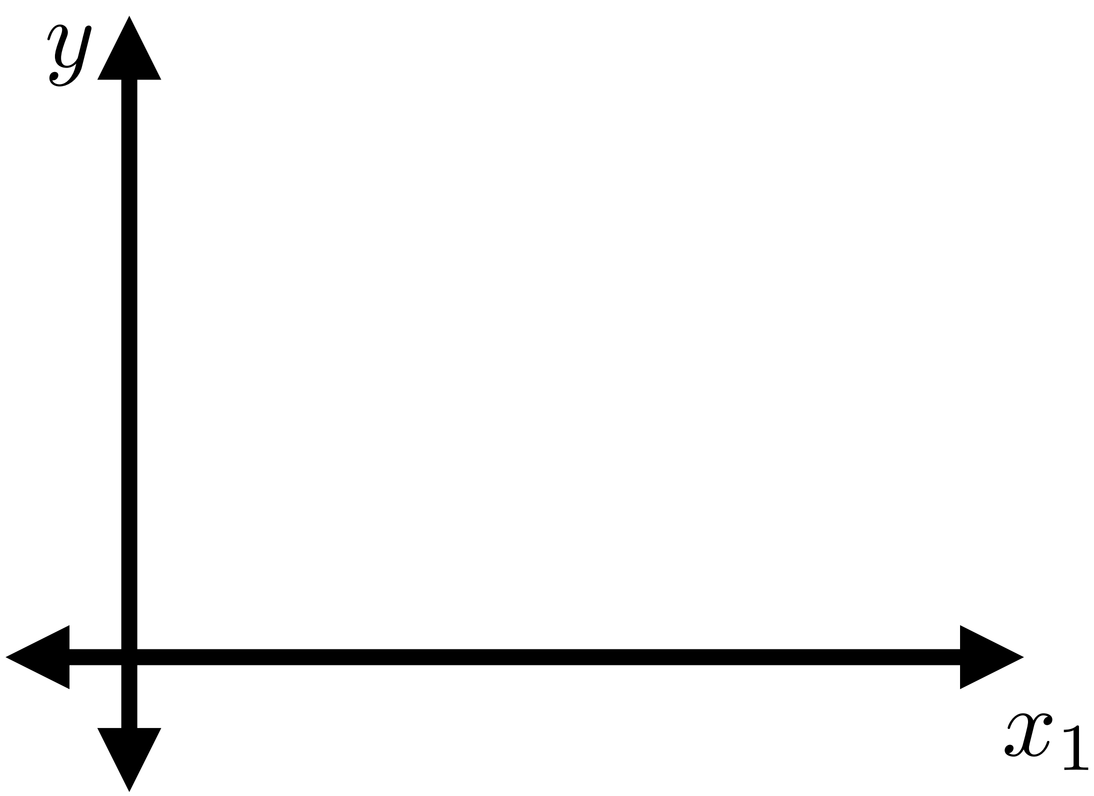
Classification

- Datum i:

Recall

Classification

- Datum i:



Recall

Classification

- Datum i:

Recall

Classification

- Datum i: feature vector

$$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$$

Recall

Classification

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \{-1, +1\}$

Recall

Classification

- Datum i: feature vector

$$x^{(i)} = (x_1^{(i)}, \dots, x_d^{(i)})^\top \in \mathbb{R}^d$$

  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$

Recall

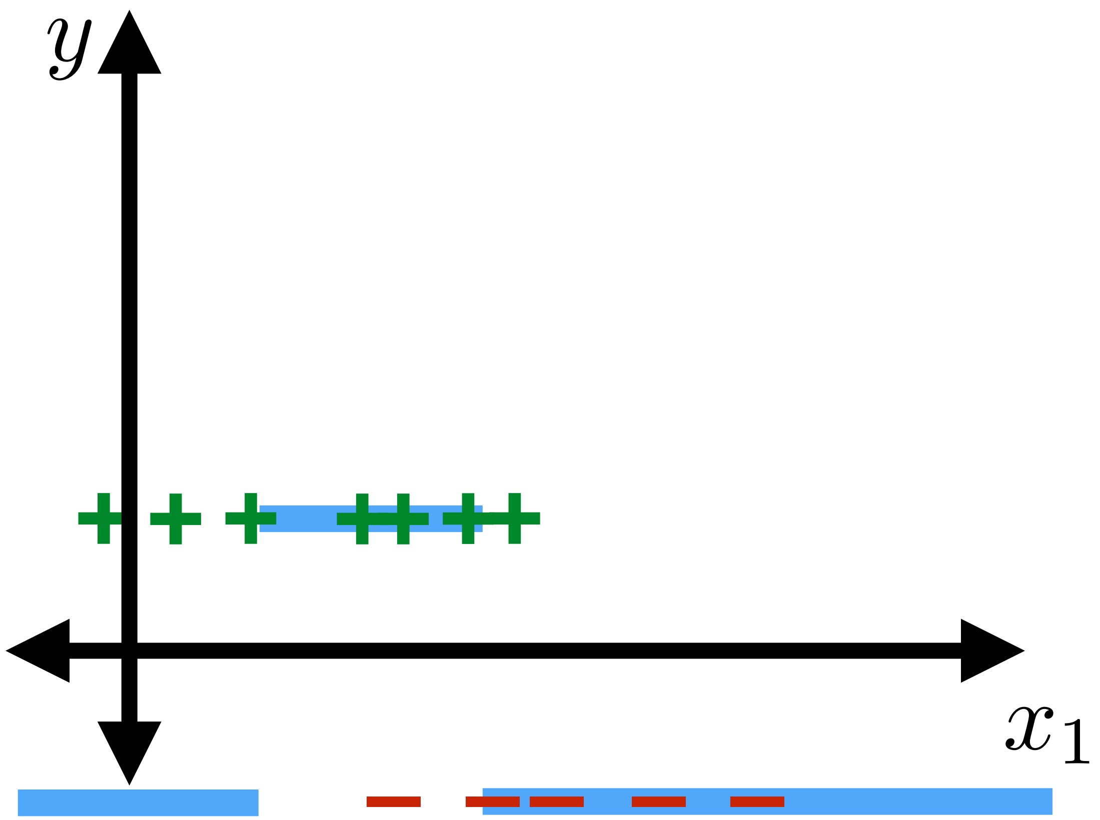
Classification

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \{-1, +1\}$

- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$



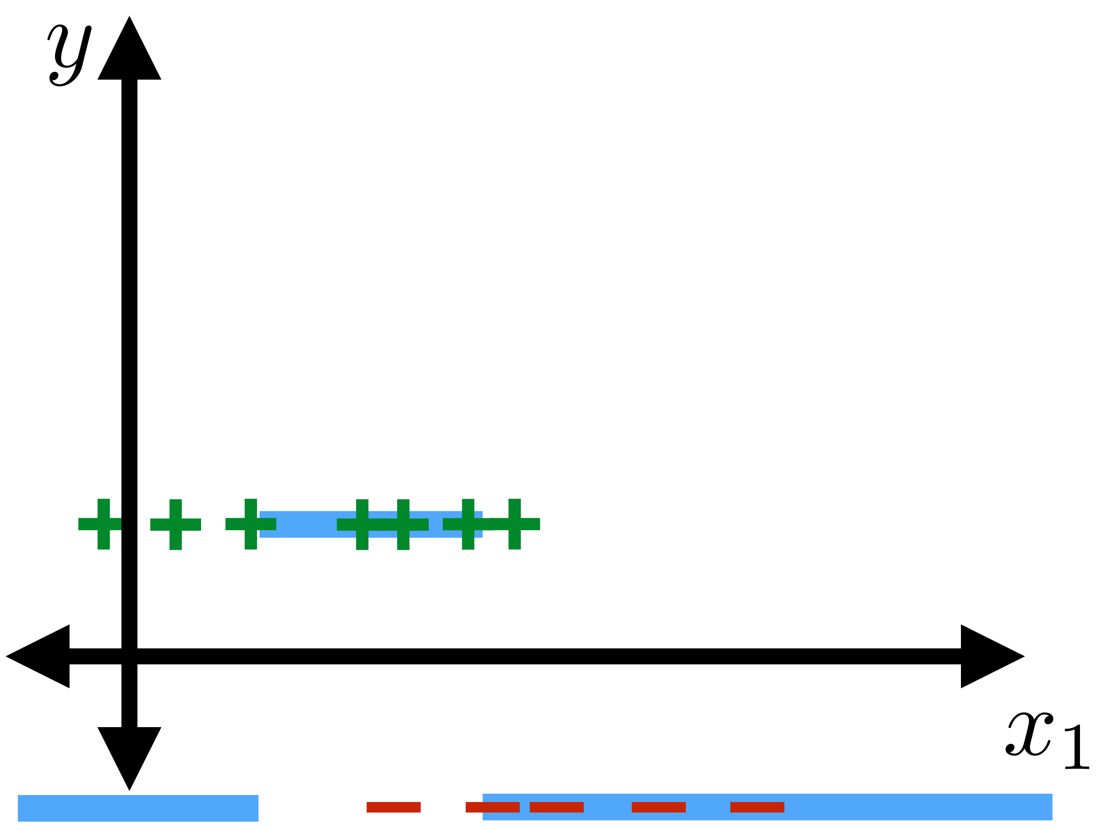
Recall

Classification

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL



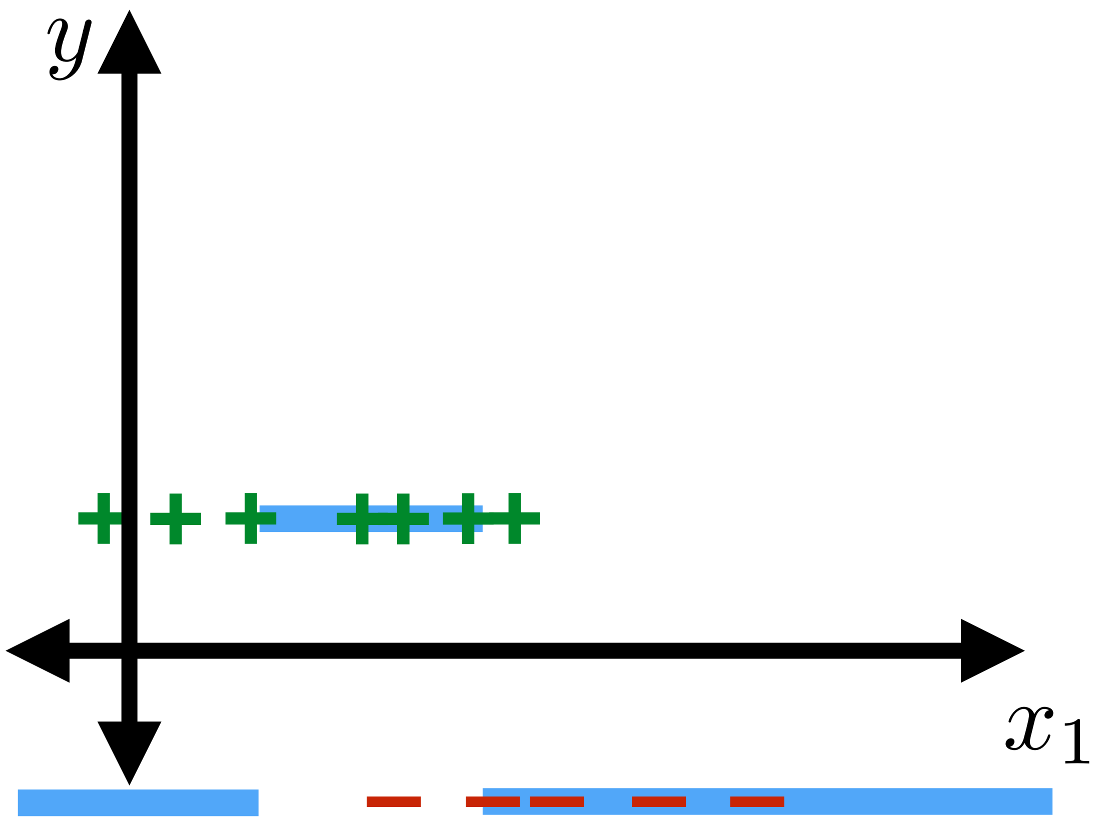
Recall

Classification

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification



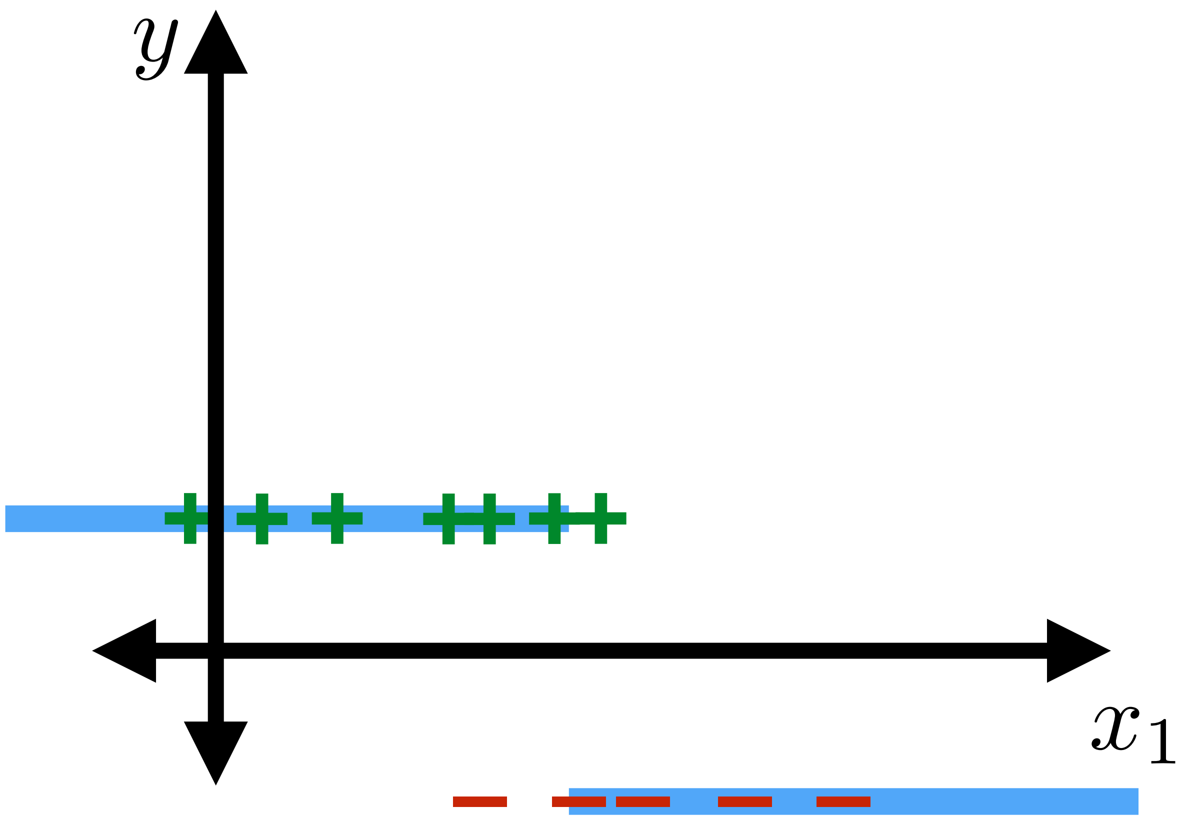
Recall

Classification

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification



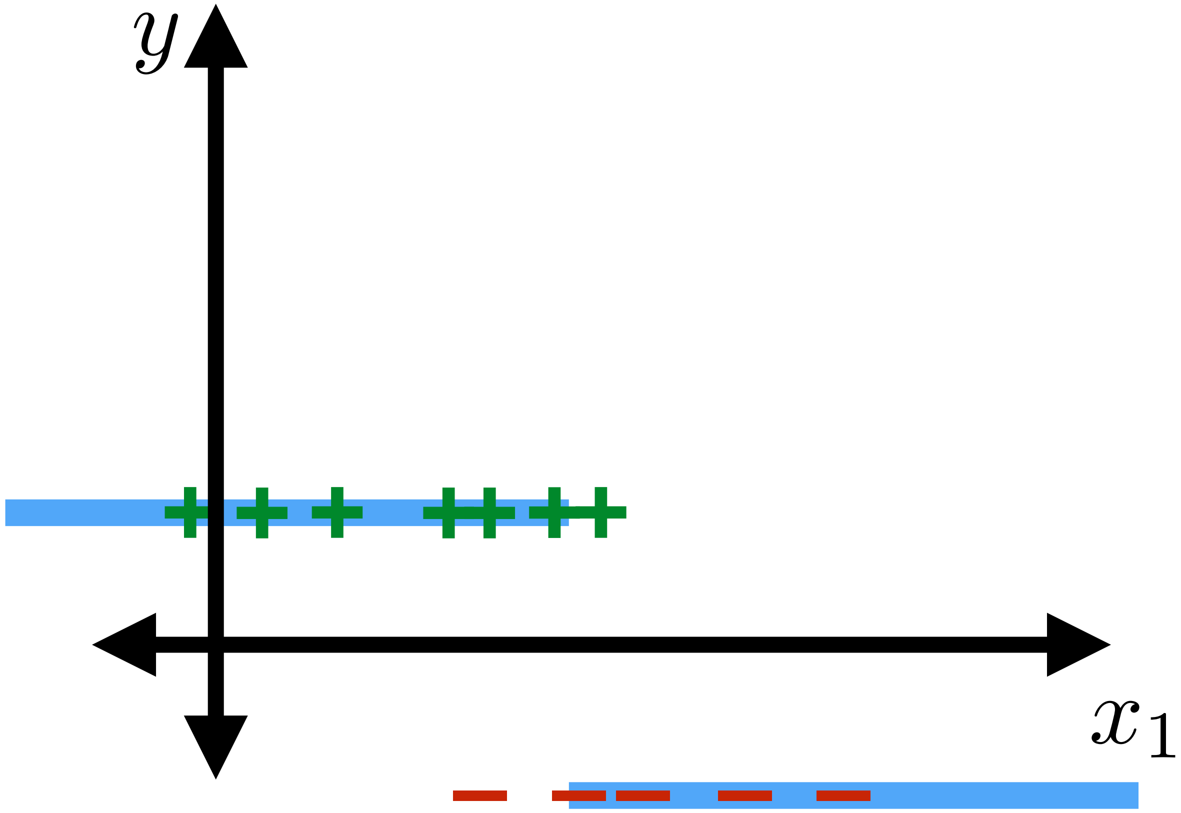
Recall

Classification

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification



Compare

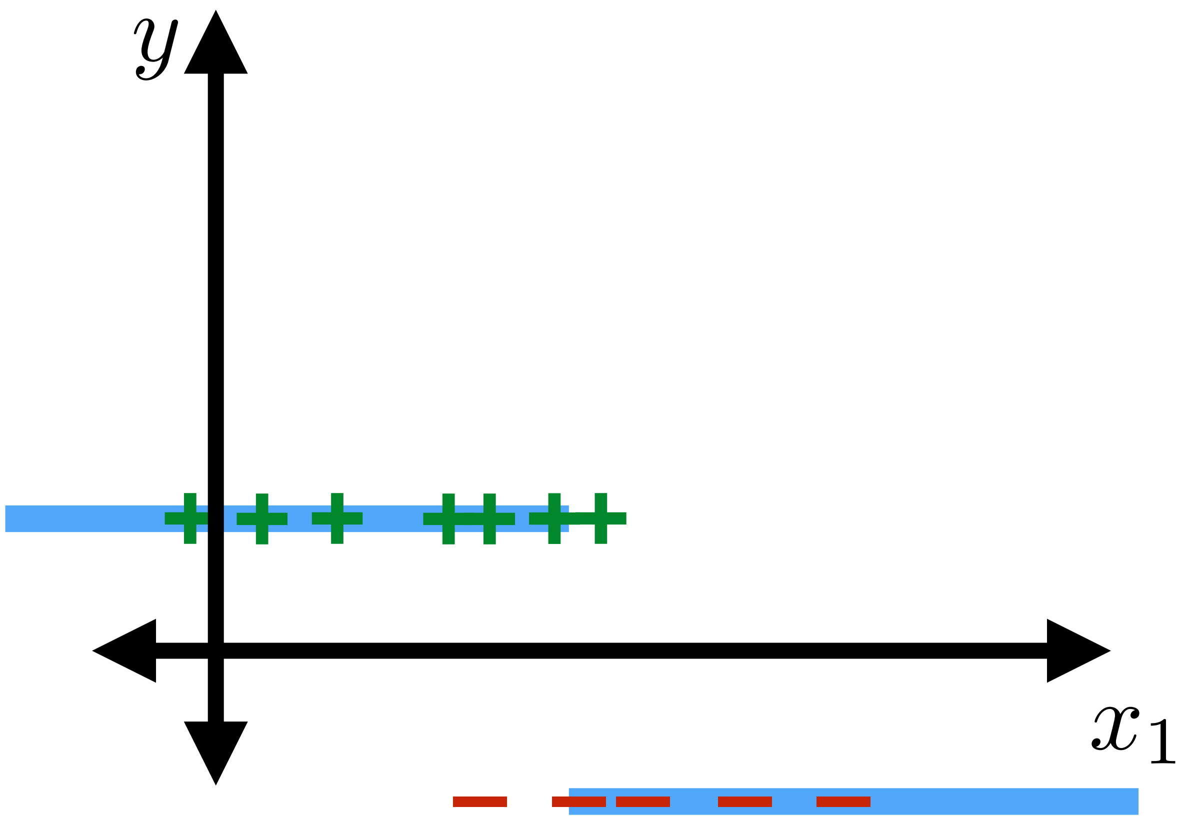
Recall

Classification

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification



Compare

Regression

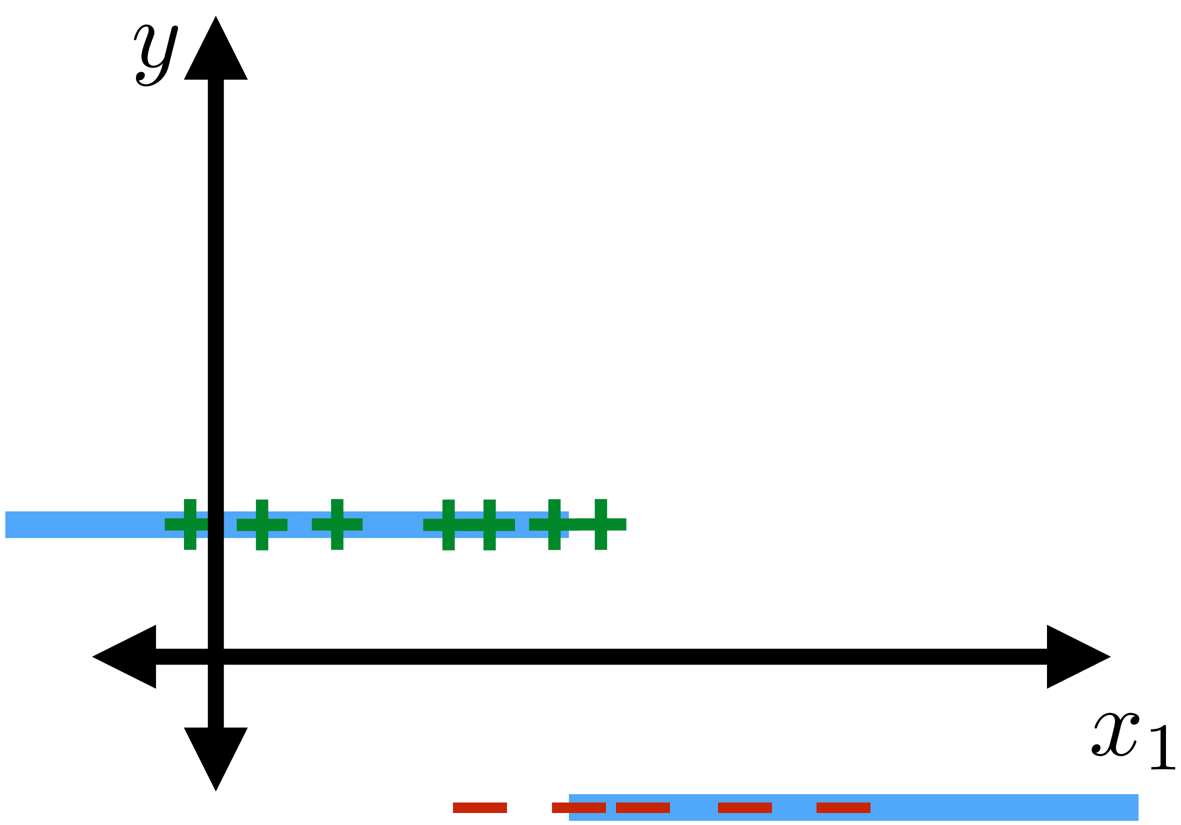
Recall

Classification

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification



Compare

Regression

- Datum i :

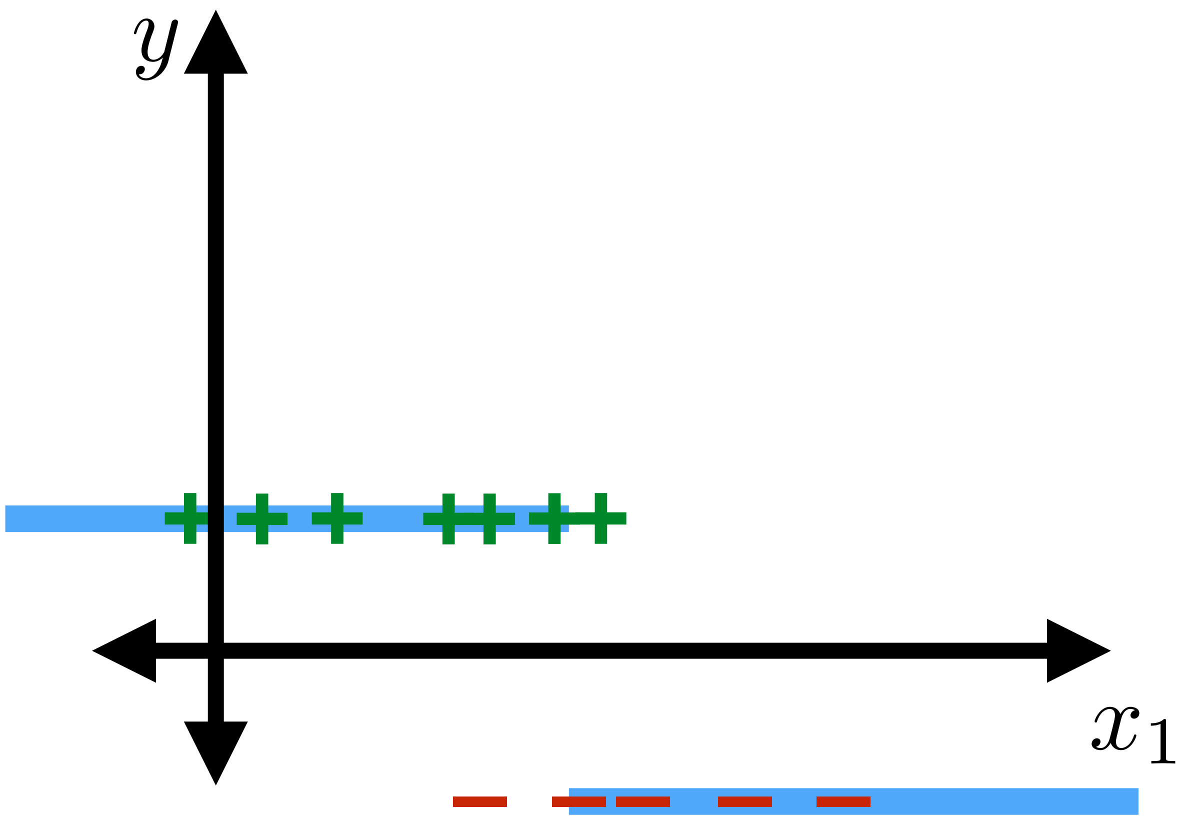
Recall

Classification

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

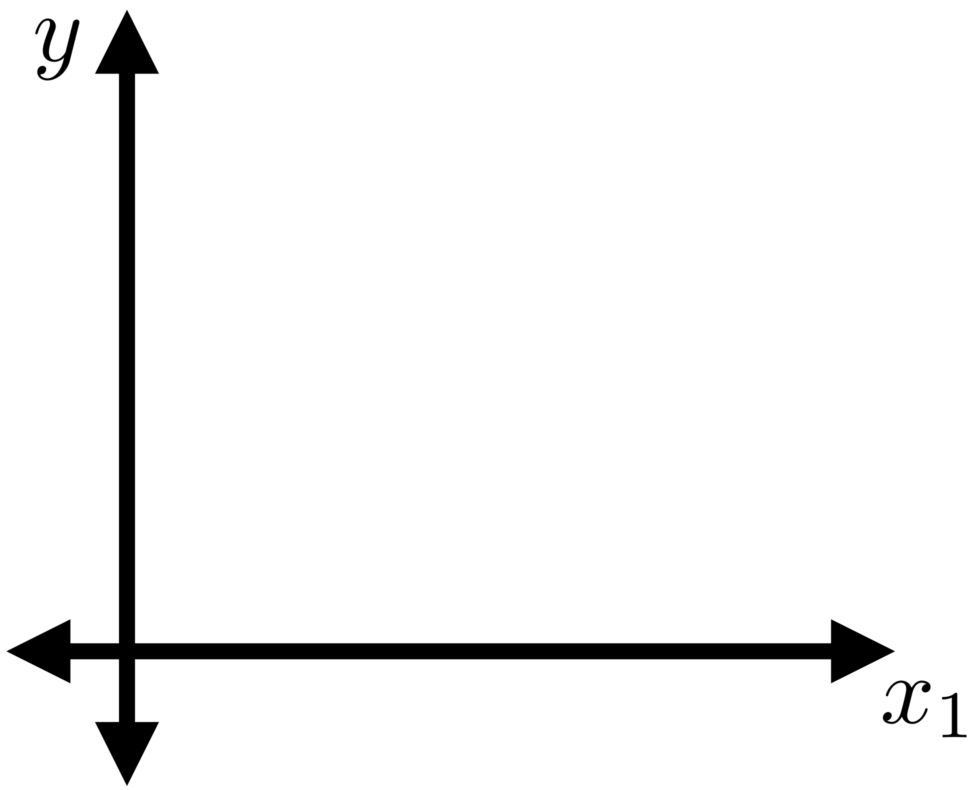
  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification



Compare

Regression

- Datum i:



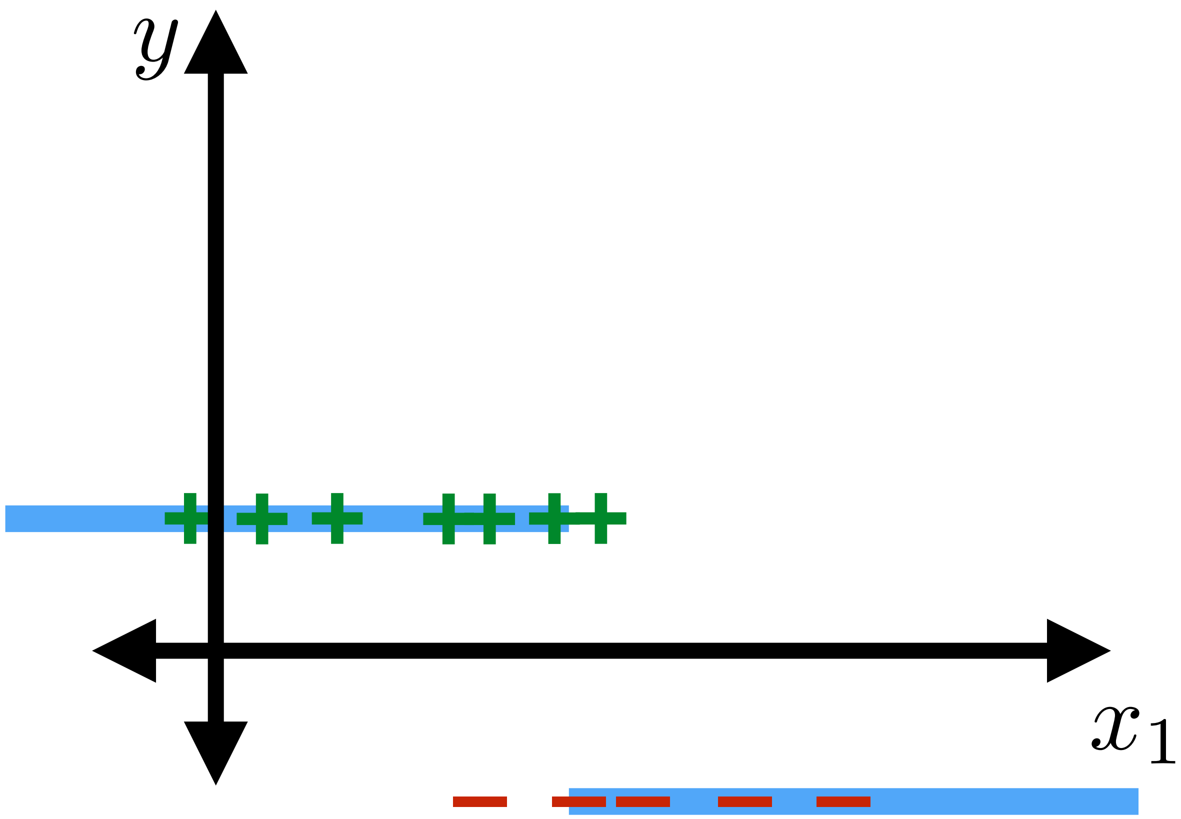
Recall

Classification

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

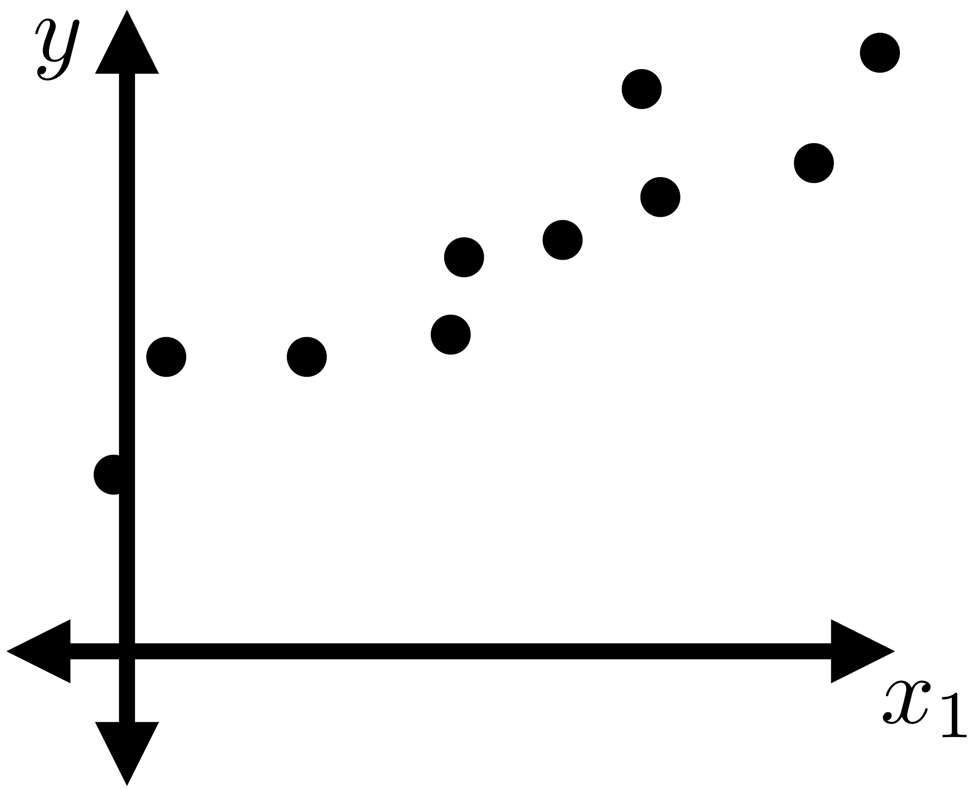
  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification



Compare

Regression

- Datum i:



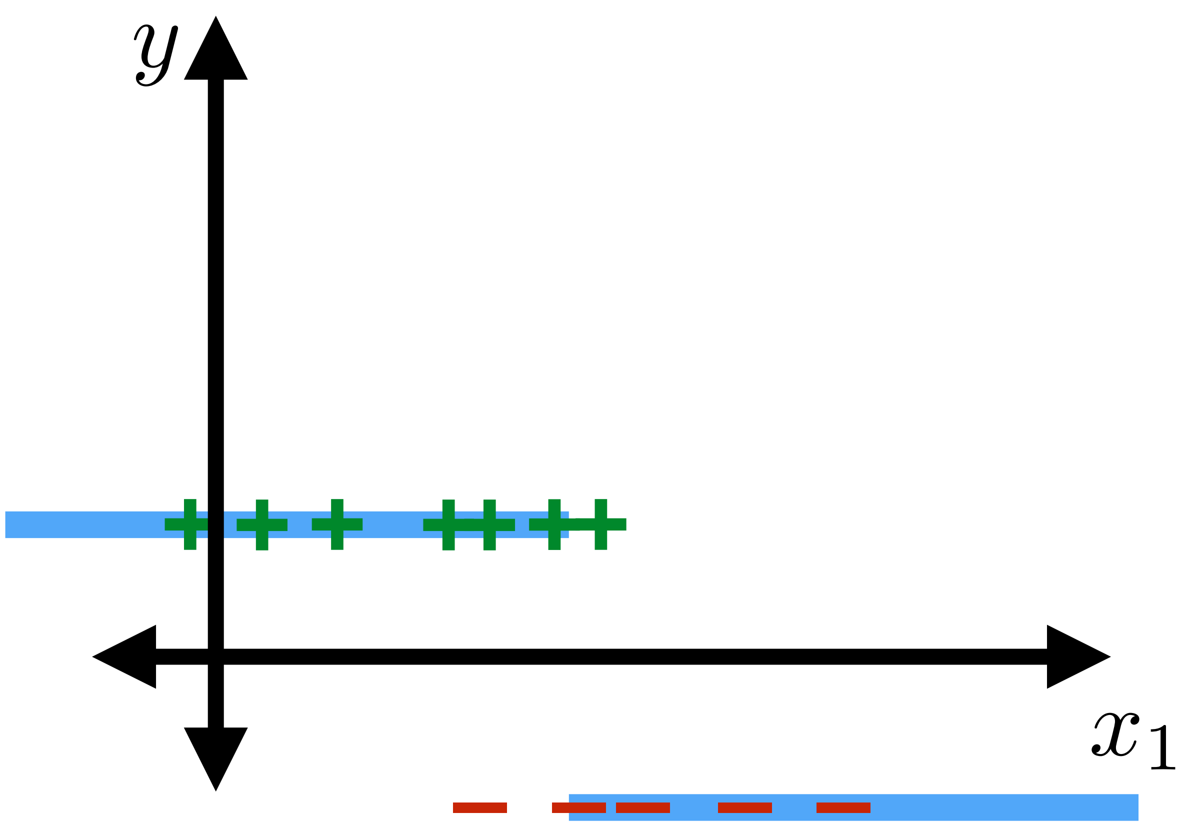
Recall

Classification

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification



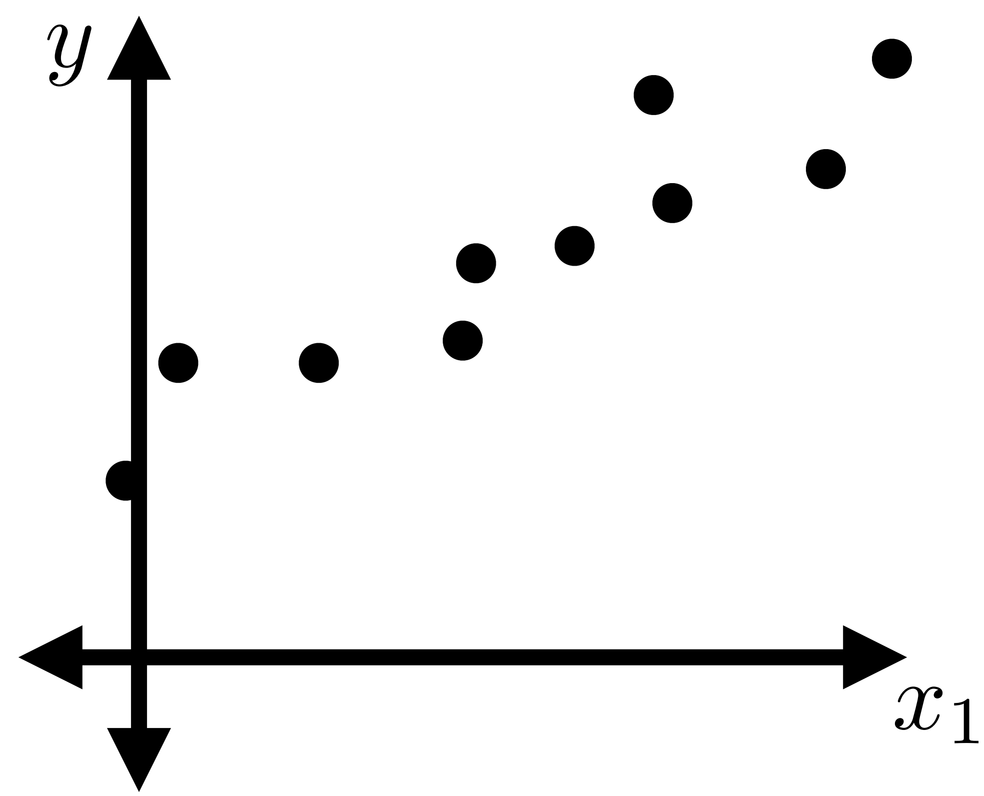
Compare

Regression

- Datum i : feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \mathbb{R}$



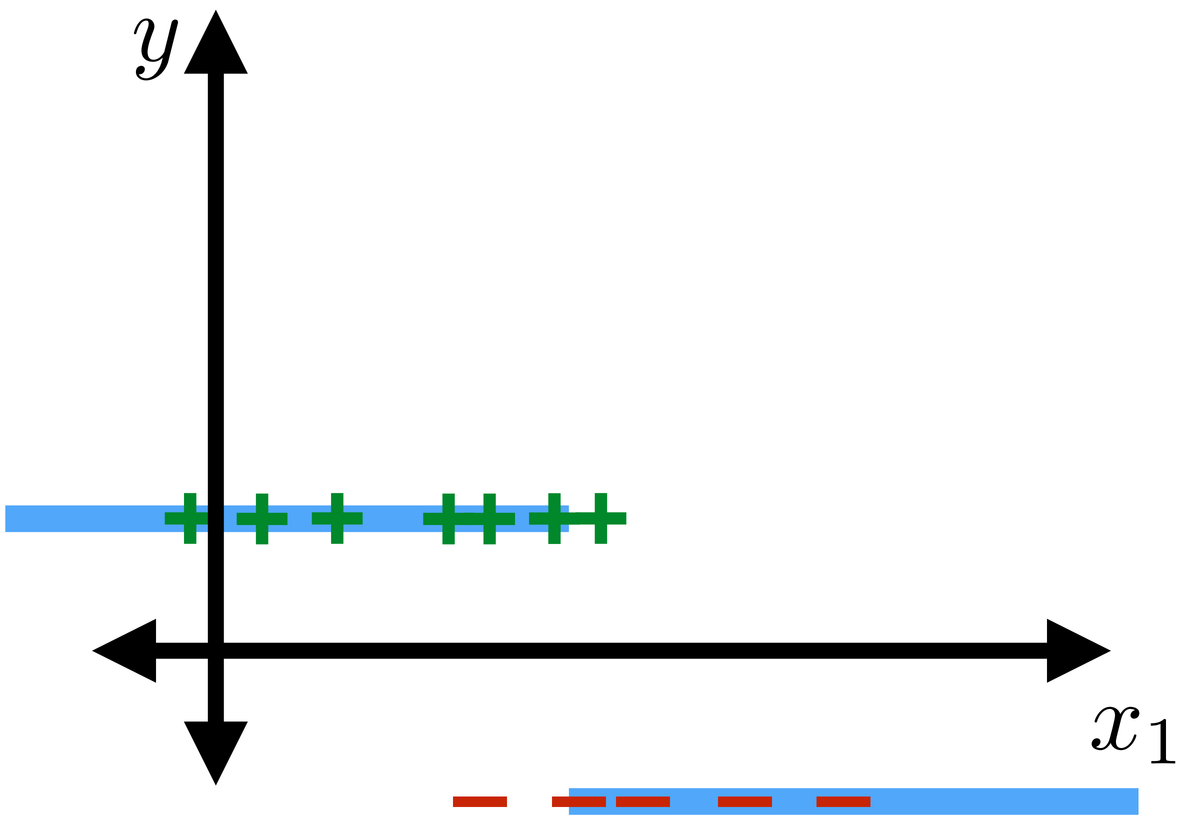
Recall

Classification

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification



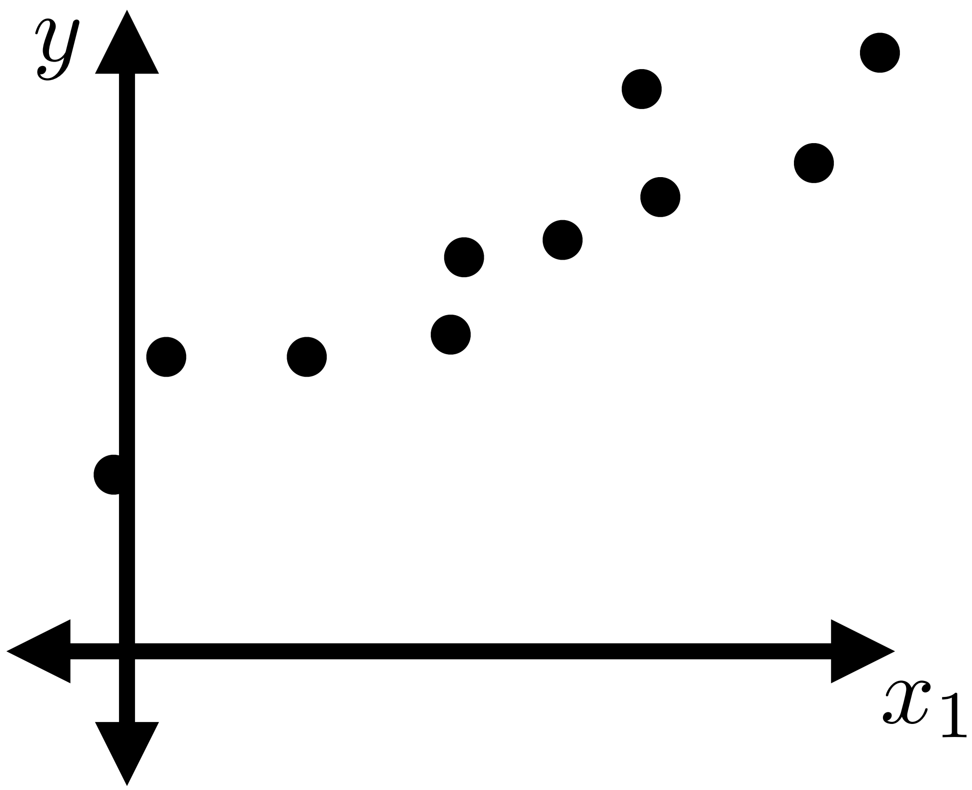
Compare

Regression

- Datum i : feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \mathbb{R}$
- Hypothesis $h : \mathbb{R}^d \to \mathbb{R}$



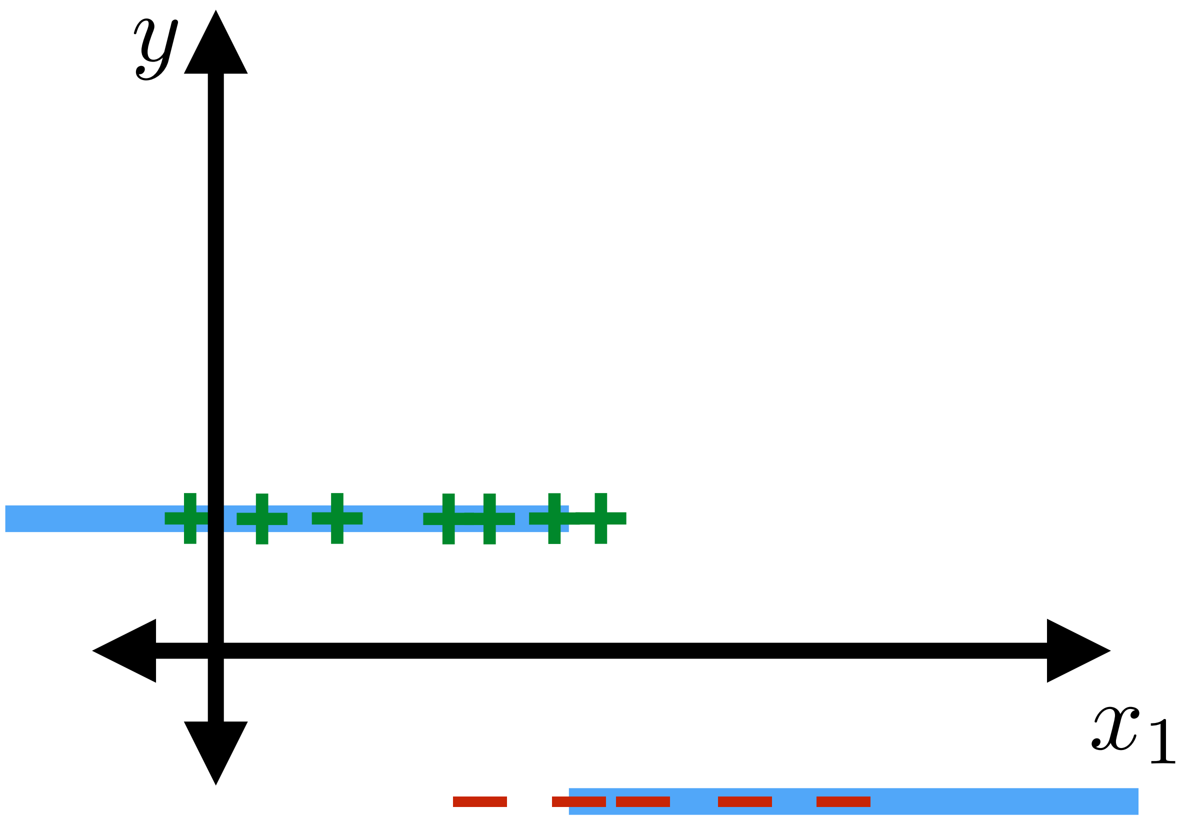
Recall

Classification

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification



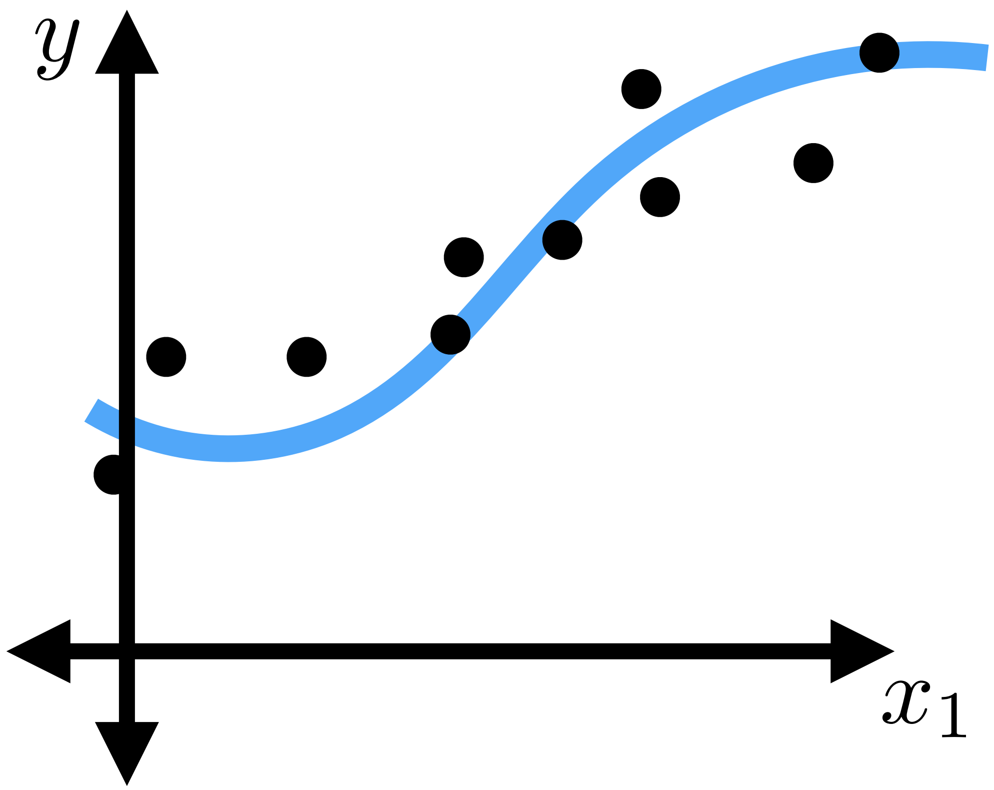
Compare

Regression

- Datum i : feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \mathbb{R}$
- Hypothesis $h : \mathbb{R}^d \to \mathbb{R}$



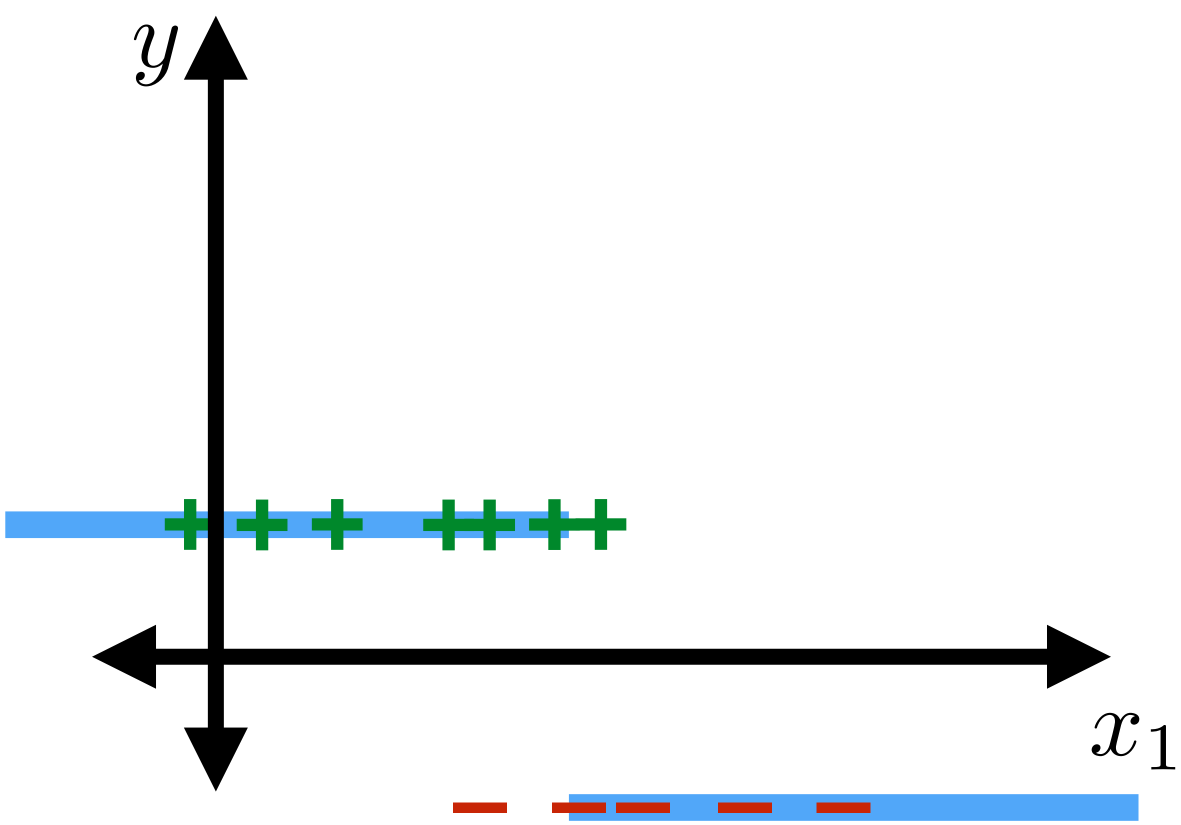
Recall

Classification

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \rightarrow \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification



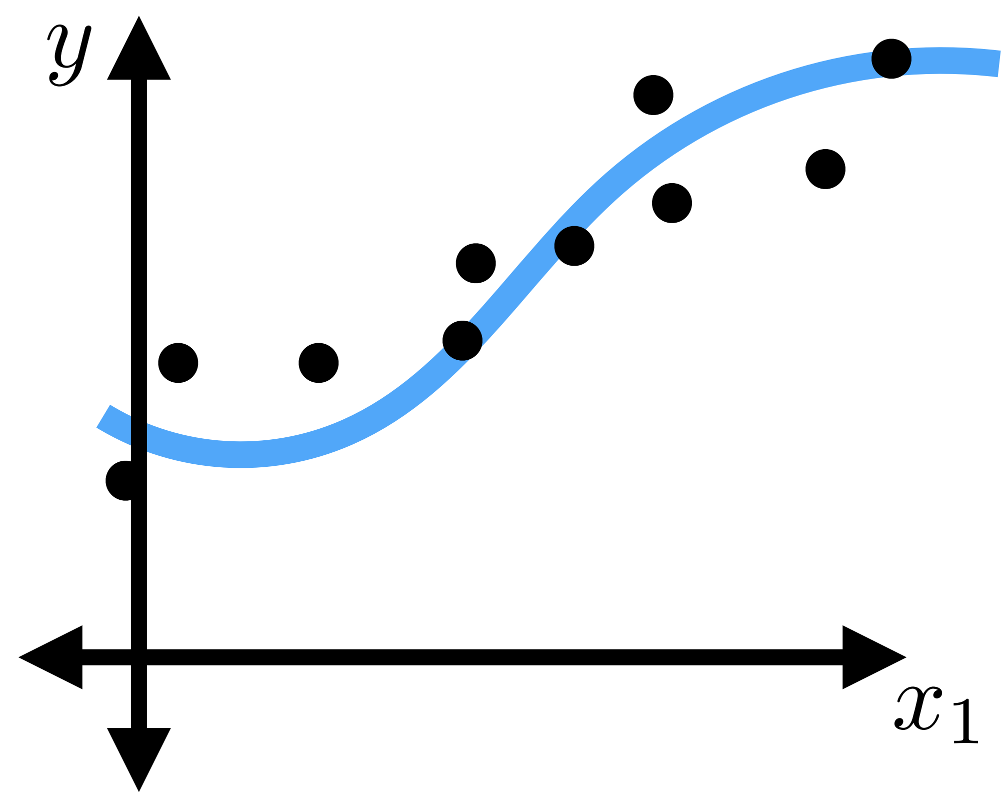
Compare

Regression

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \mathbb{R}$
- Hypothesis $h : \mathbb{R}^d \rightarrow \mathbb{R}$
- Loss:



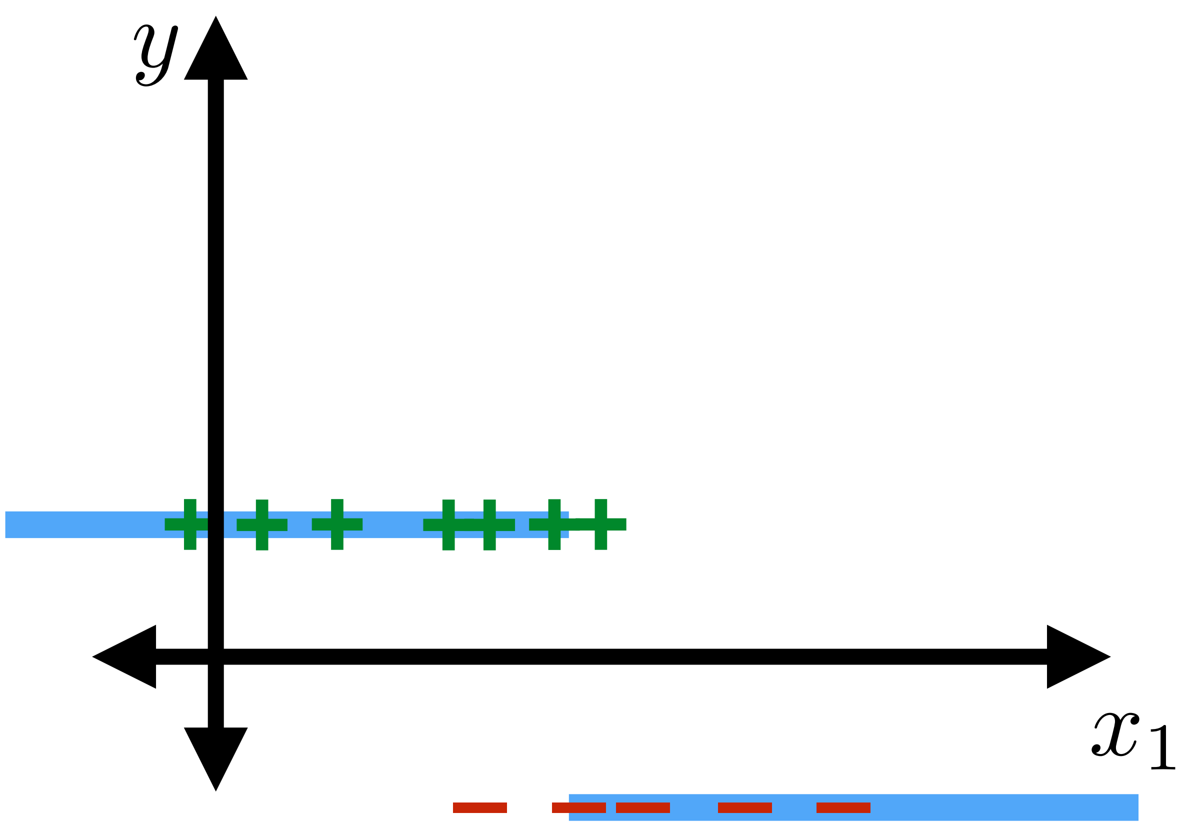
Recall

Classification

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification



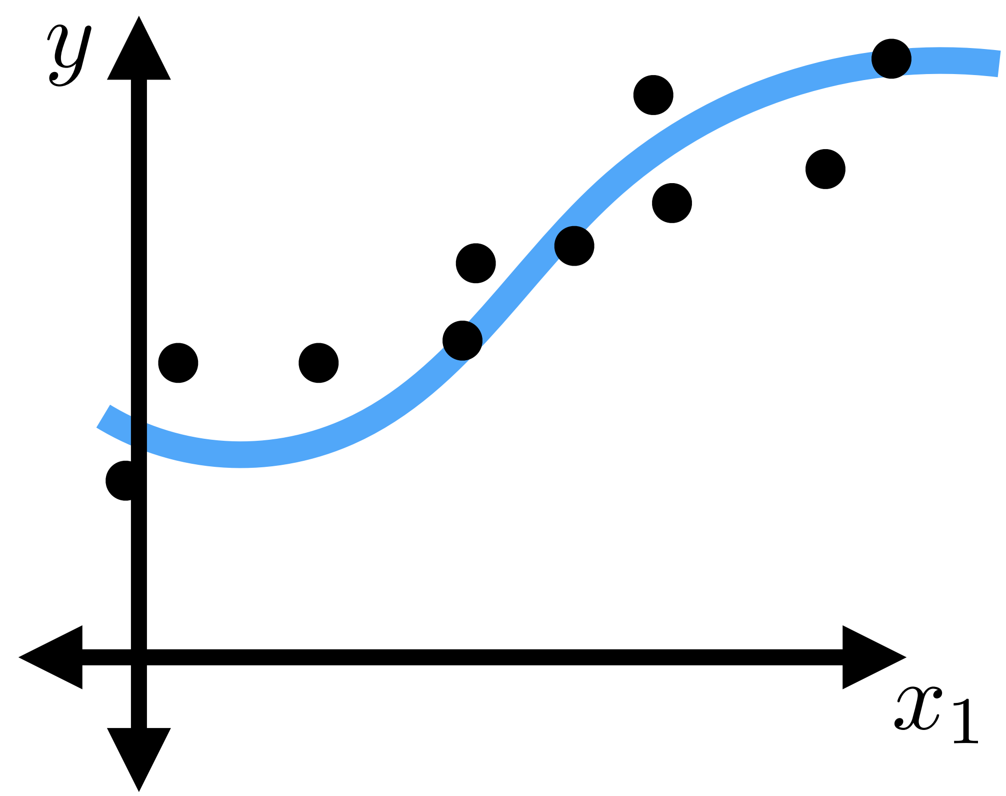
Compare

Regression

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

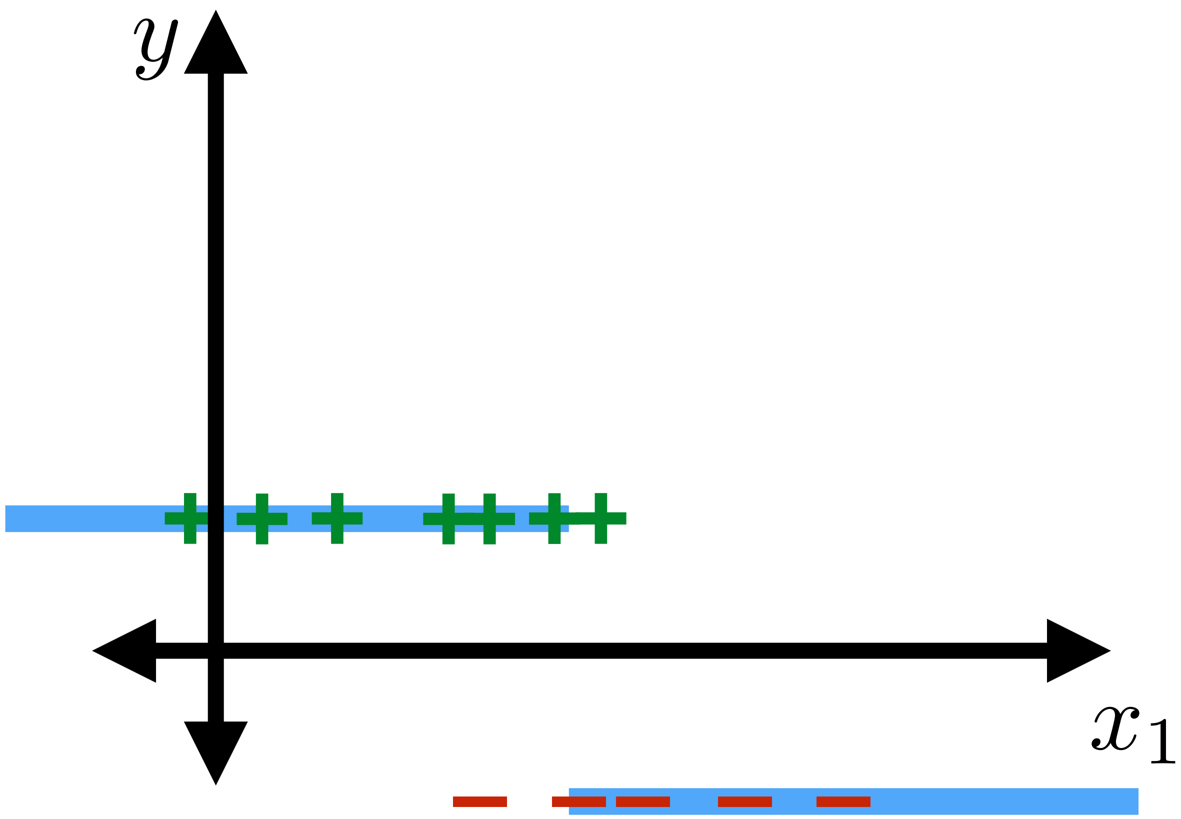
  - Label $y^{(i)} \in \mathbb{R}$
- Hypothesis $h : \mathbb{R}^d \to \mathbb{R}$
- Loss: $L(g, a) = (g - a)^2$



Recall

Classification

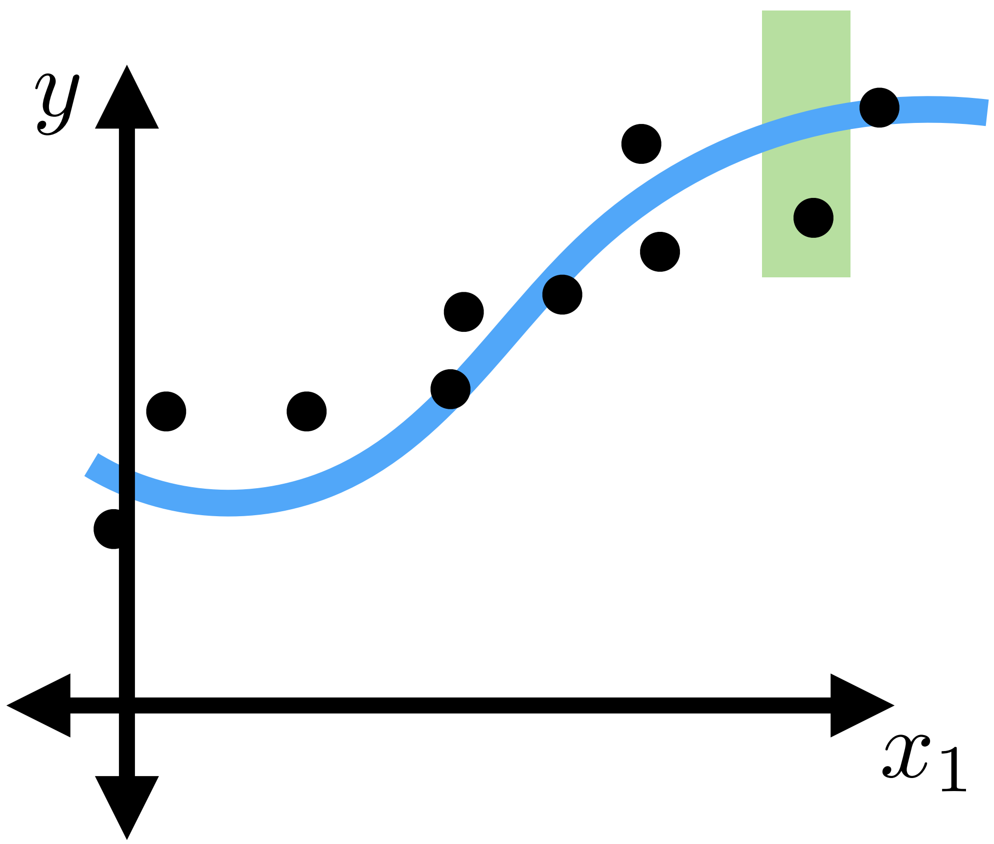
- Datum i: feature vector
$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification



Compare

Regression

- Datum i: feature vector
$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
  - Label $y^{(i)} \in \mathbb{R}$
- Hypothesis $h : \mathbb{R}^d \to \mathbb{R}$
- Loss: $L(g, a) = (g - a)^2$



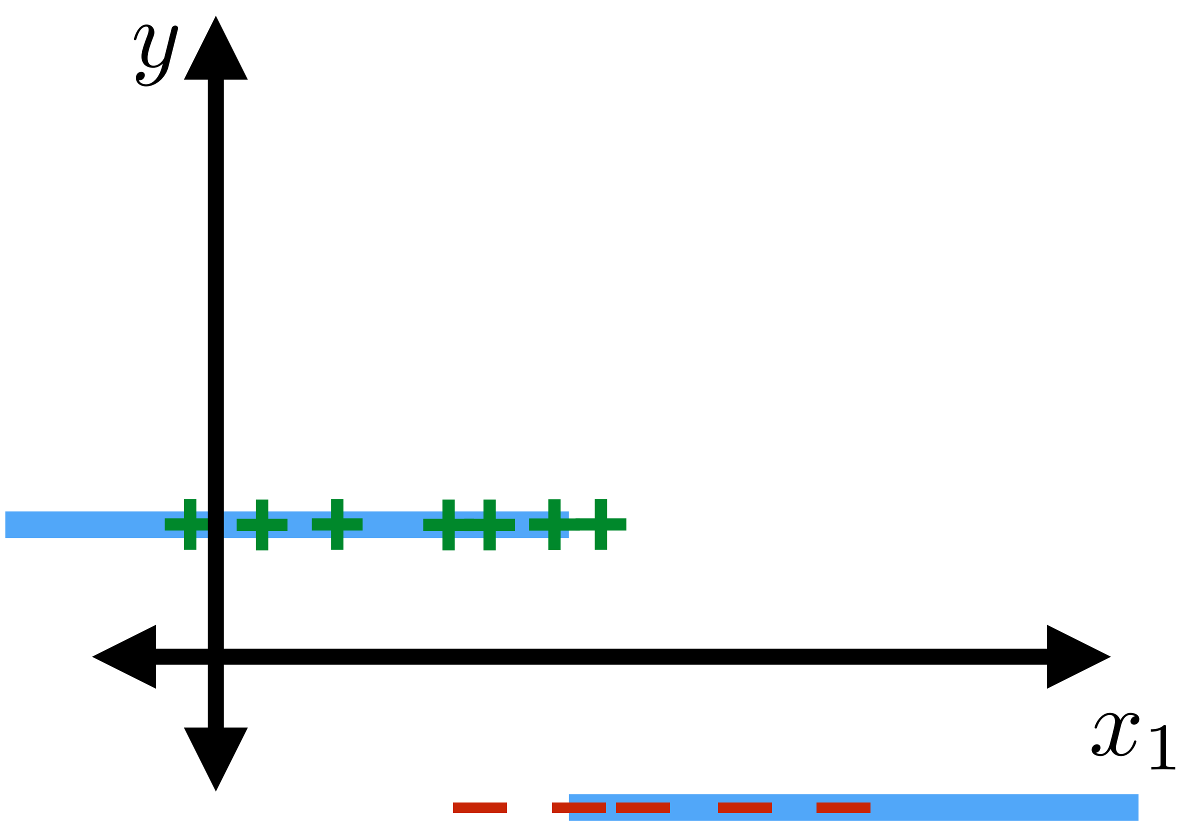
Recall

Classification

- Datum i: feature vector

$$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$$

  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \rightarrow \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification



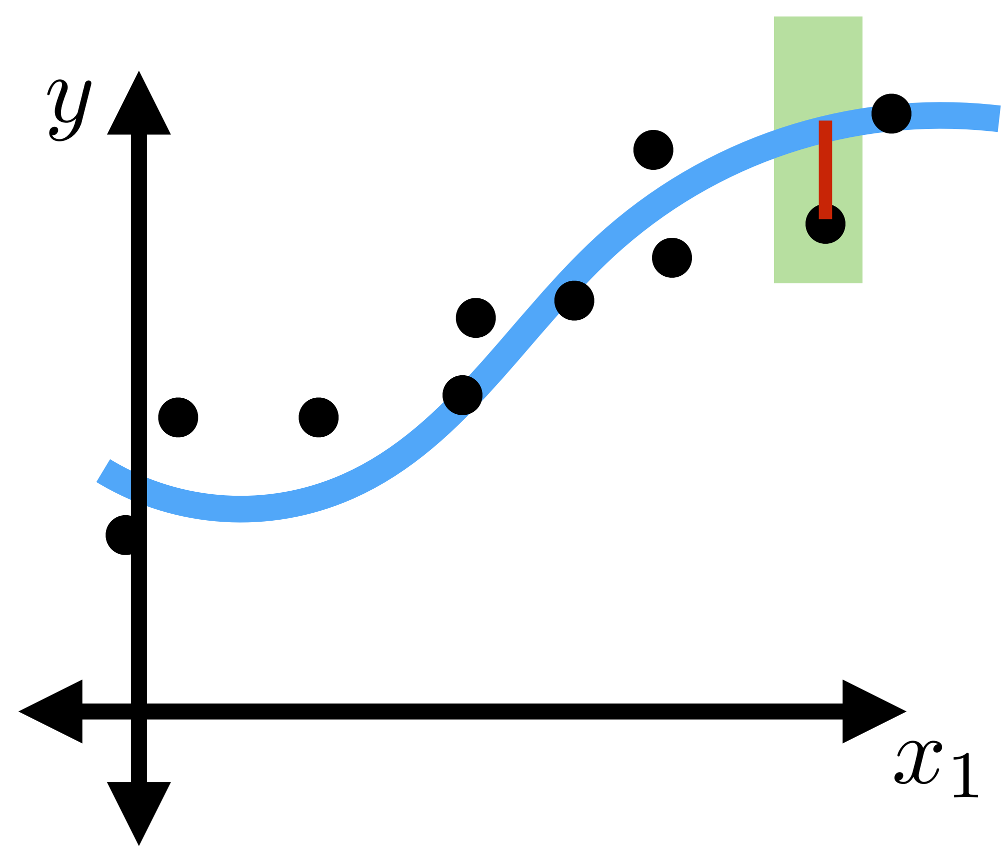
Compare

Regression

- Datum i: feature vector

$$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$$

  - Label $y^{(i)} \in \mathbb{R}$
- Hypothesis $h : \mathbb{R}^d \rightarrow \mathbb{R}$
- Loss: $L(g, a) = (g - a)^2$



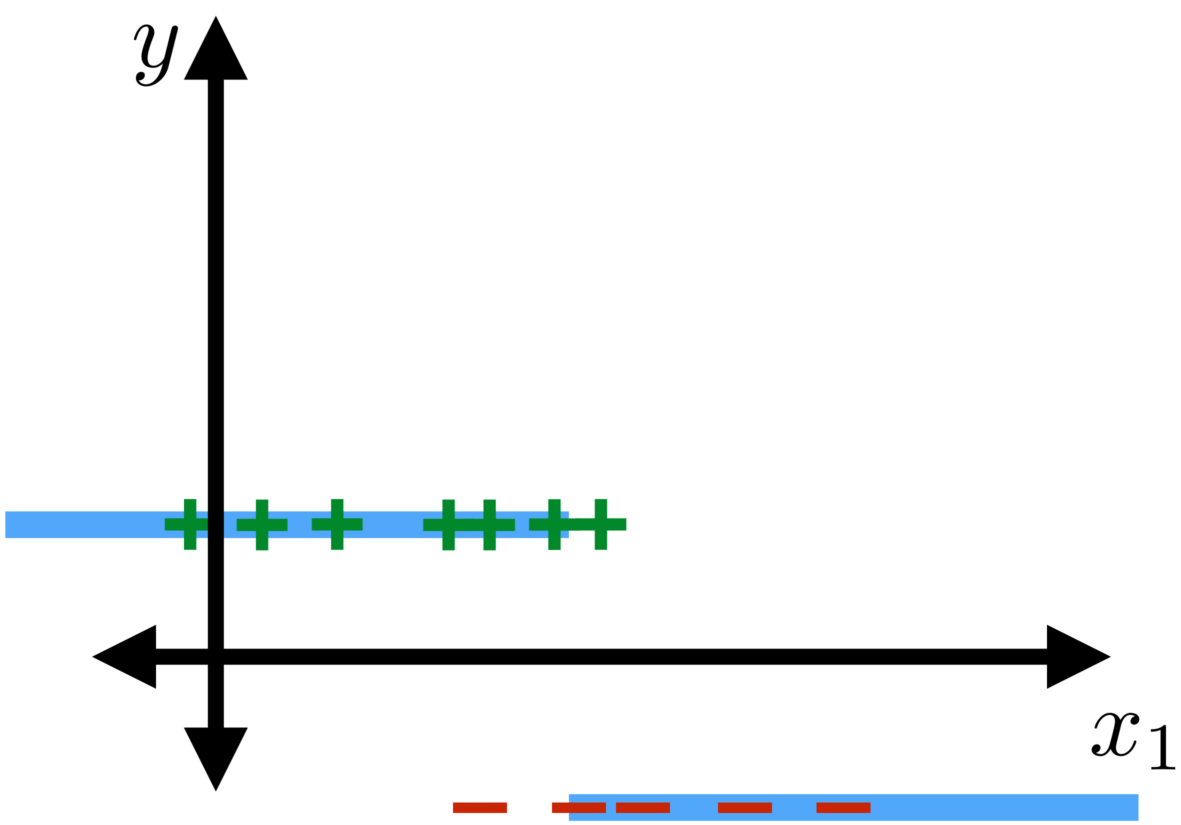
Recall

Classification

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification



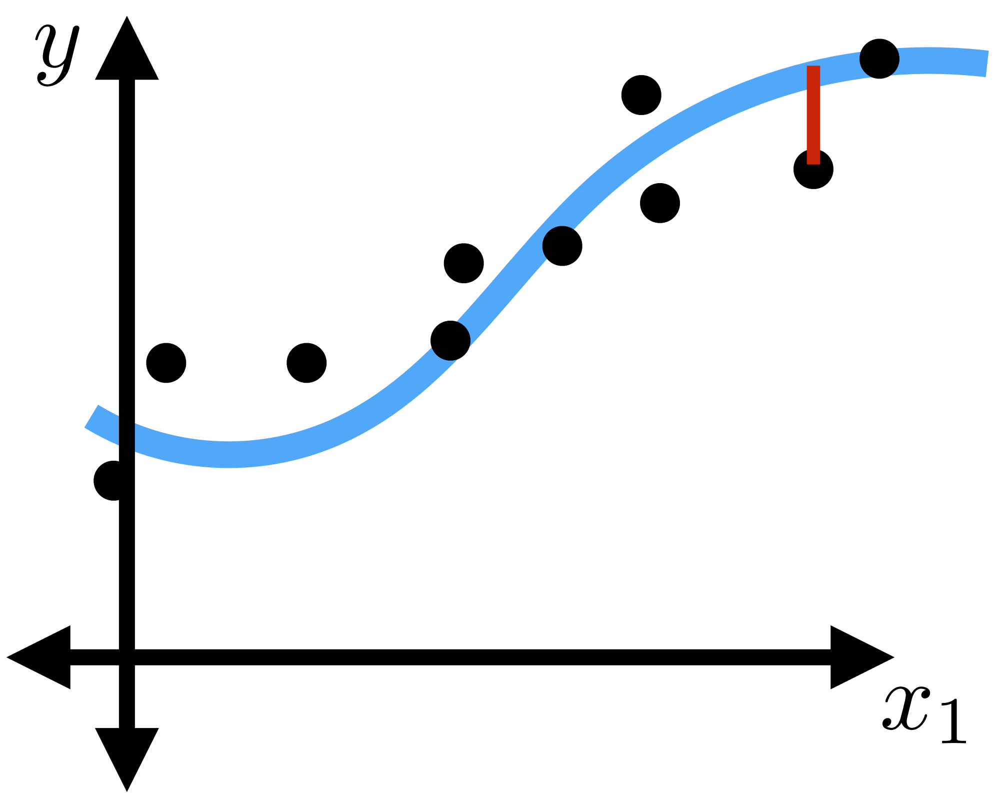
Compare

Regression

- Datum i : feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \mathbb{R}$
- Hypothesis $h : \mathbb{R}^d \to \mathbb{R}$
- Loss: $L(g, a) = (g - a)^2$



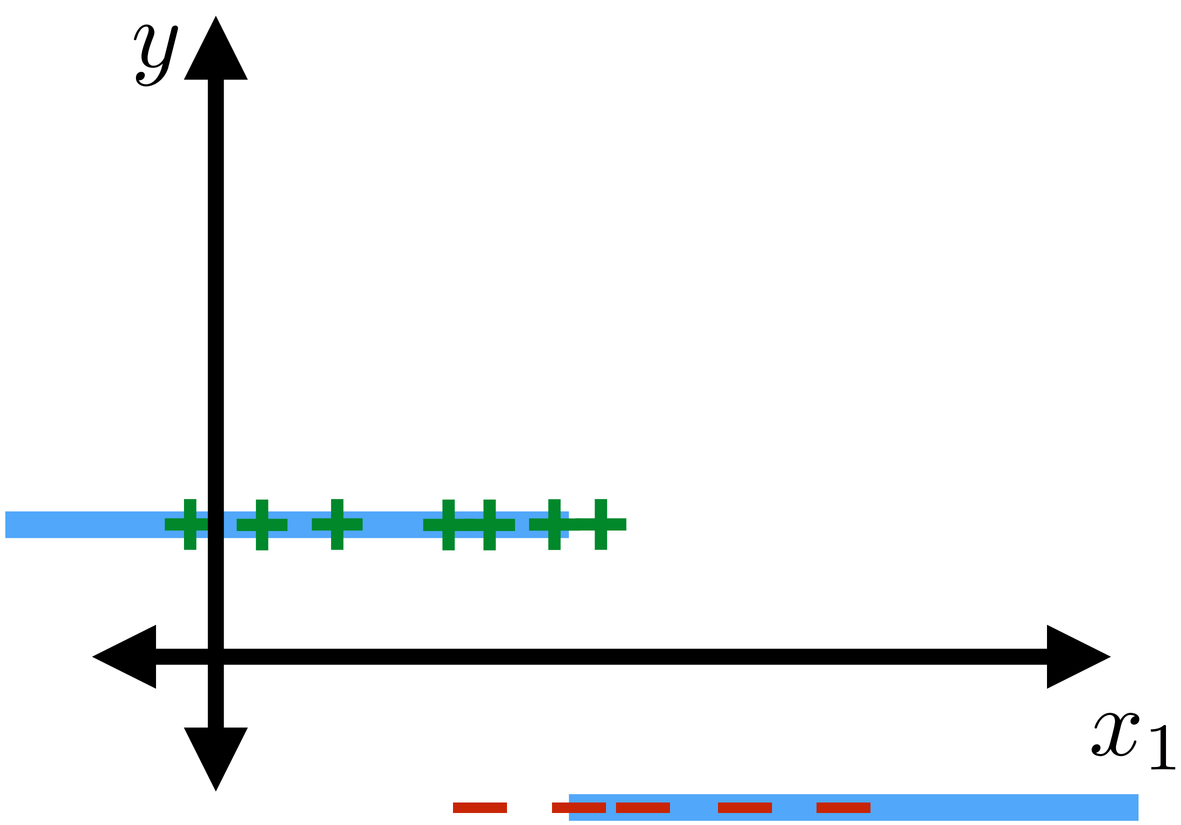
Recall

Classification

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification



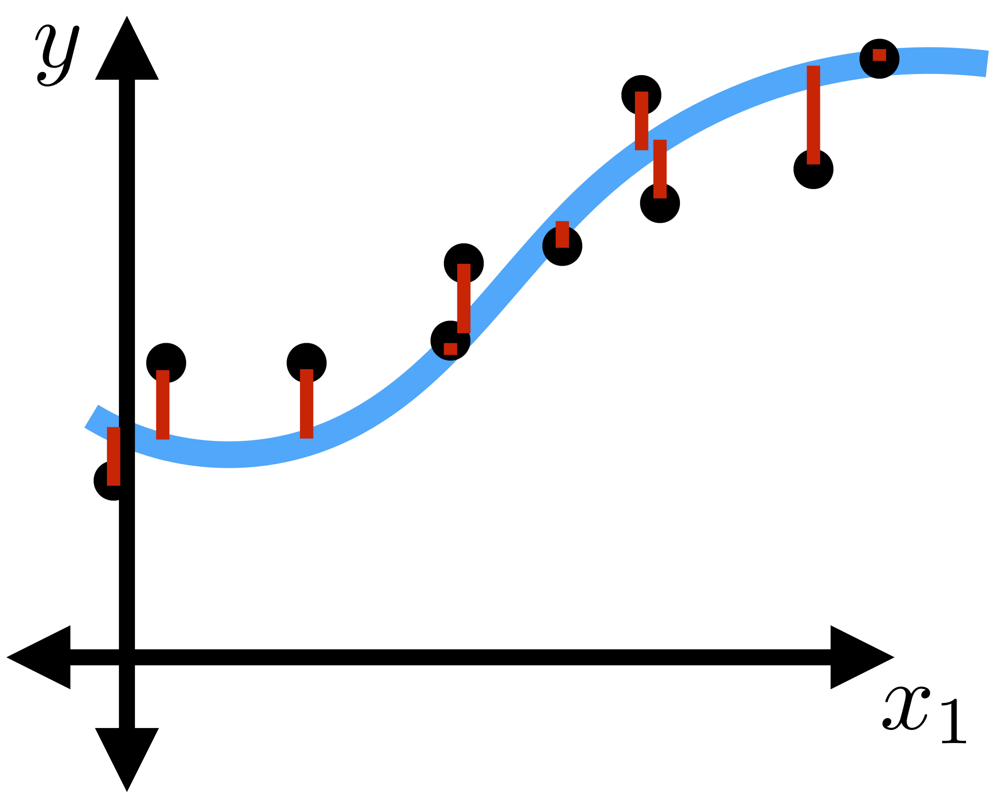
Compare

Regression

- Datum i : feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \mathbb{R}$
- Hypothesis $h : \mathbb{R}^d \to \mathbb{R}$
- Loss: $L(g, a) = (g - a)^2$



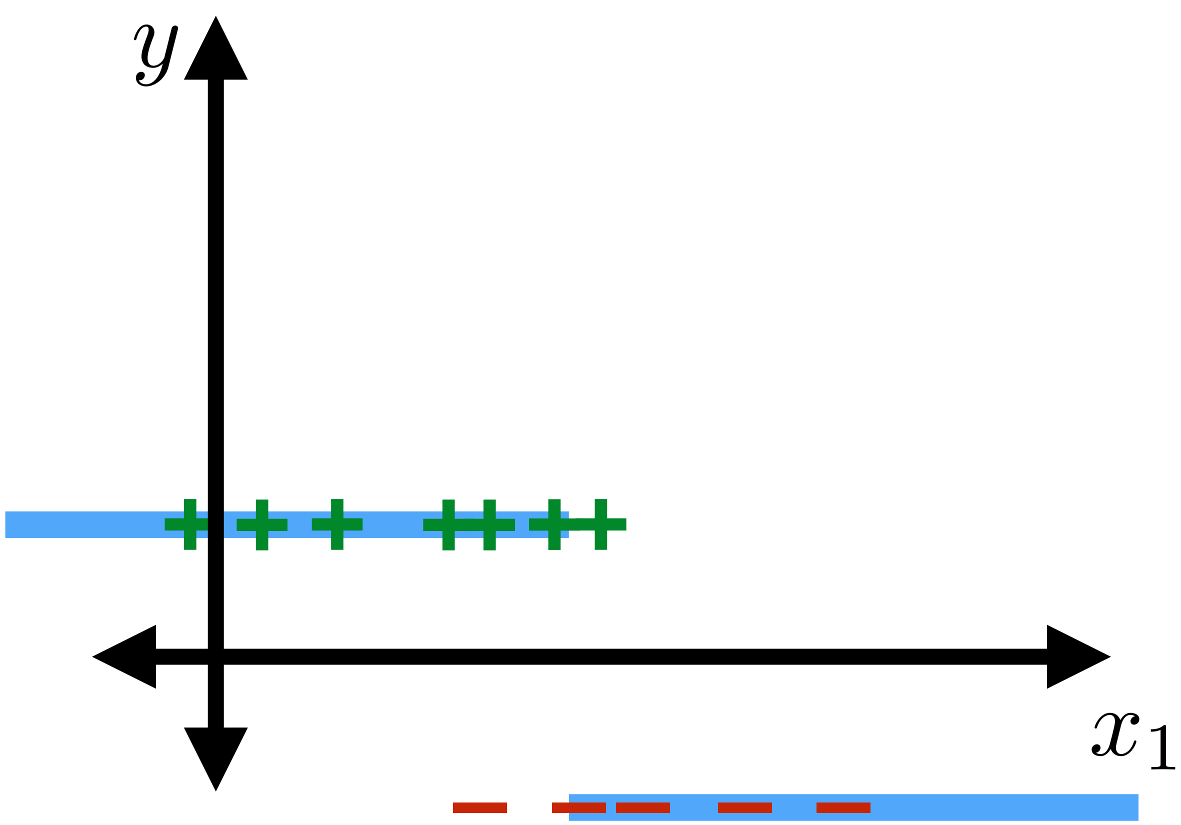
Recall

Classification

- Datum i: feature vector

$$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$$

  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification



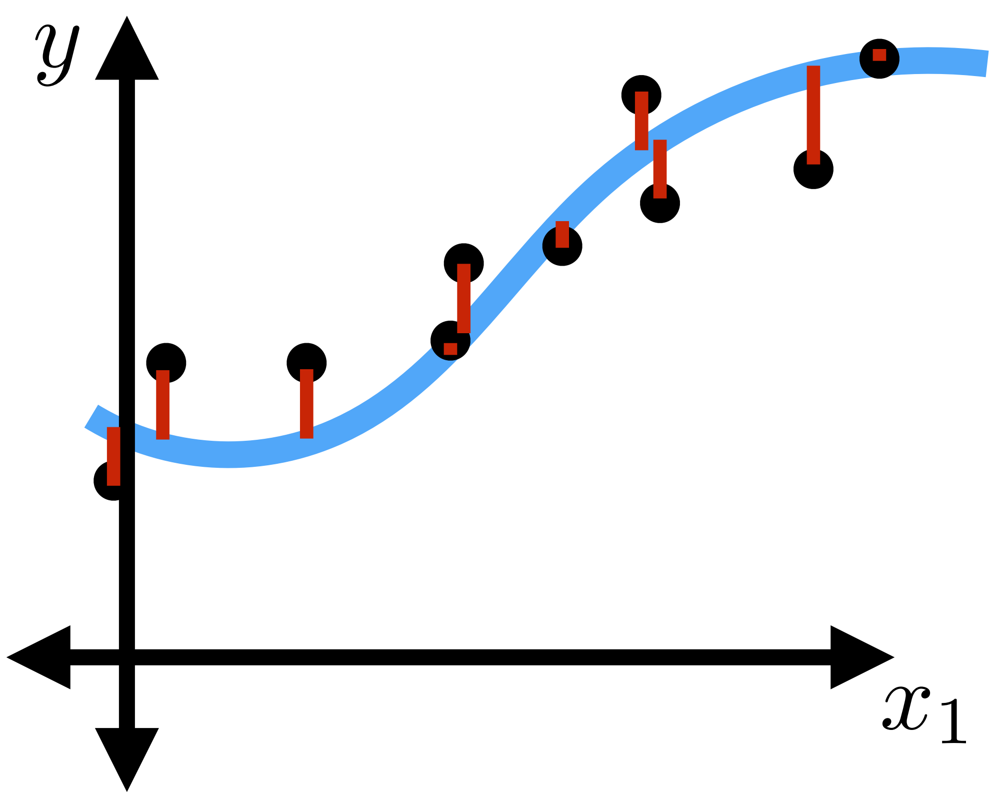
Compare

Regression

- Datum i : feature vector

$$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$$

  - Label $y^{(i)} \in \mathbb{R}$
- Hypothesis $h : \mathbb{R}^d \to \mathbb{R}$
- Loss: $L(g, a) = (g - a)^2$
- Example: linear regression



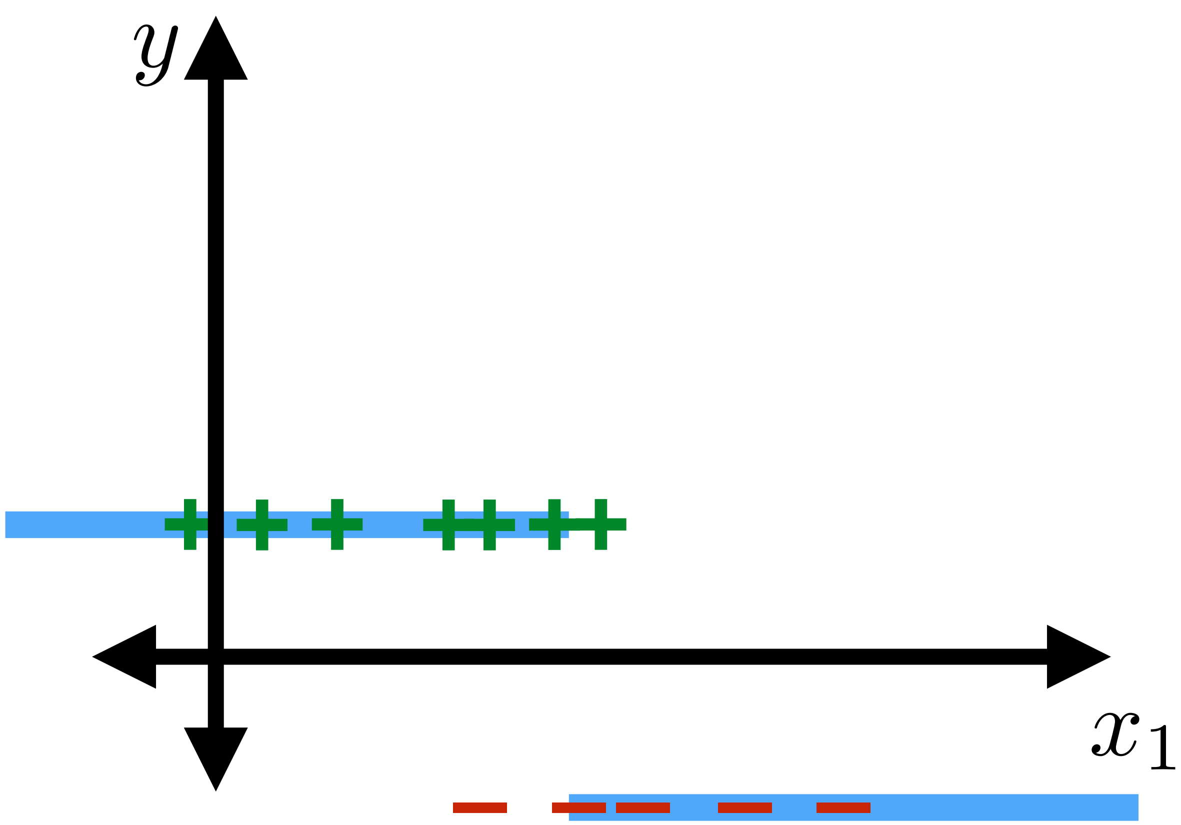
Recall

Classification

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification



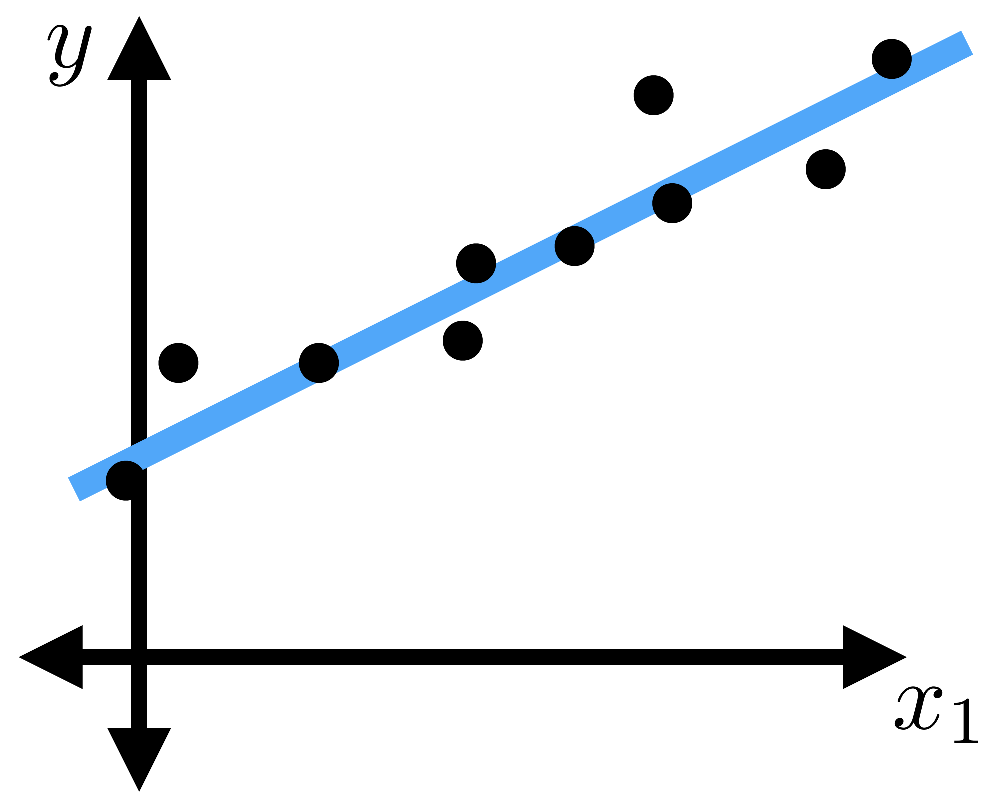
Compare

Regression

- Datum i : feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \mathbb{R}$
- Hypothesis $h : \mathbb{R}^d \to \mathbb{R}$
- Loss: $L(g, a) = (g - a)^2$
- Example: linear regression



Recall

Classification

- Datum i: feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification

Compare

Regression

- Datum i: feature vector $x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$
  - Label $y^{(i)} \in \mathbb{R}$
- Hypothesis $h : \mathbb{R}^d \to \mathbb{R}$
- Loss: $L(g, a) = (g - a)^2$
- Example: linear regression

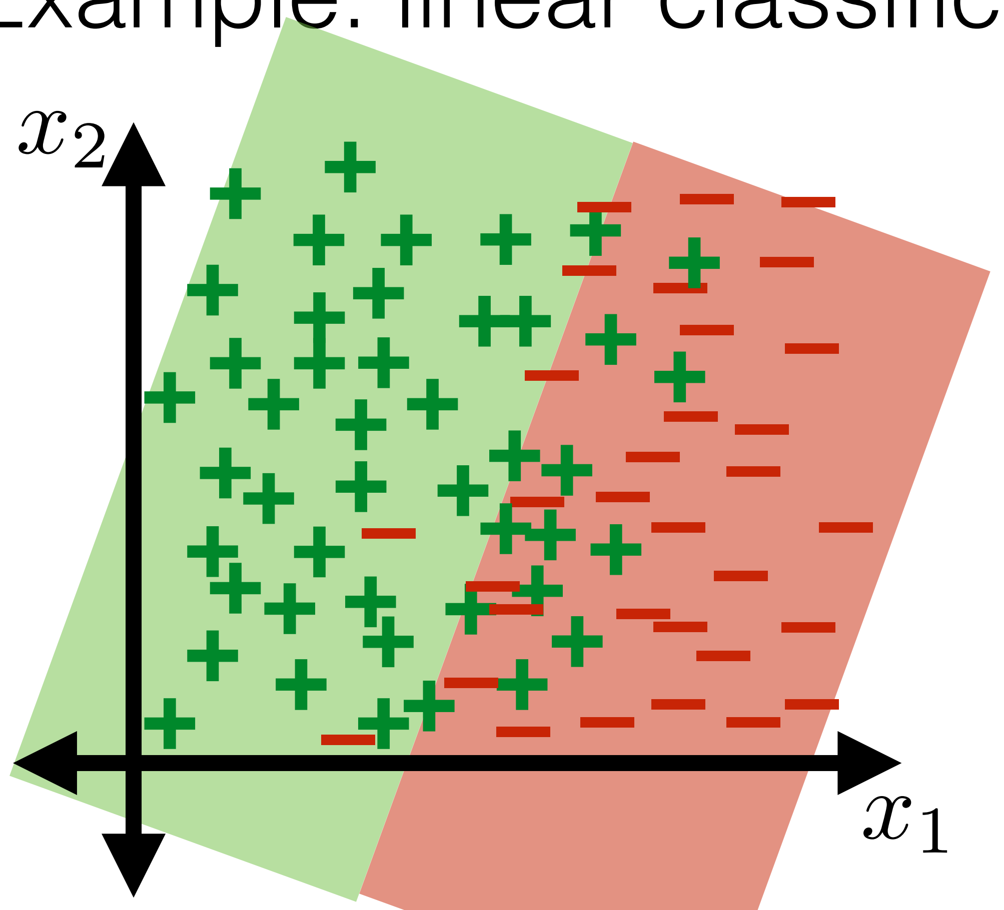
Recall

Classification

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification



Compare

Regression

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \mathbb{R}$
- Hypothesis $h : \mathbb{R}^d \to \mathbb{R}$
- Loss: $L(g, a) = (g - a)^2$
- Example: linear regression

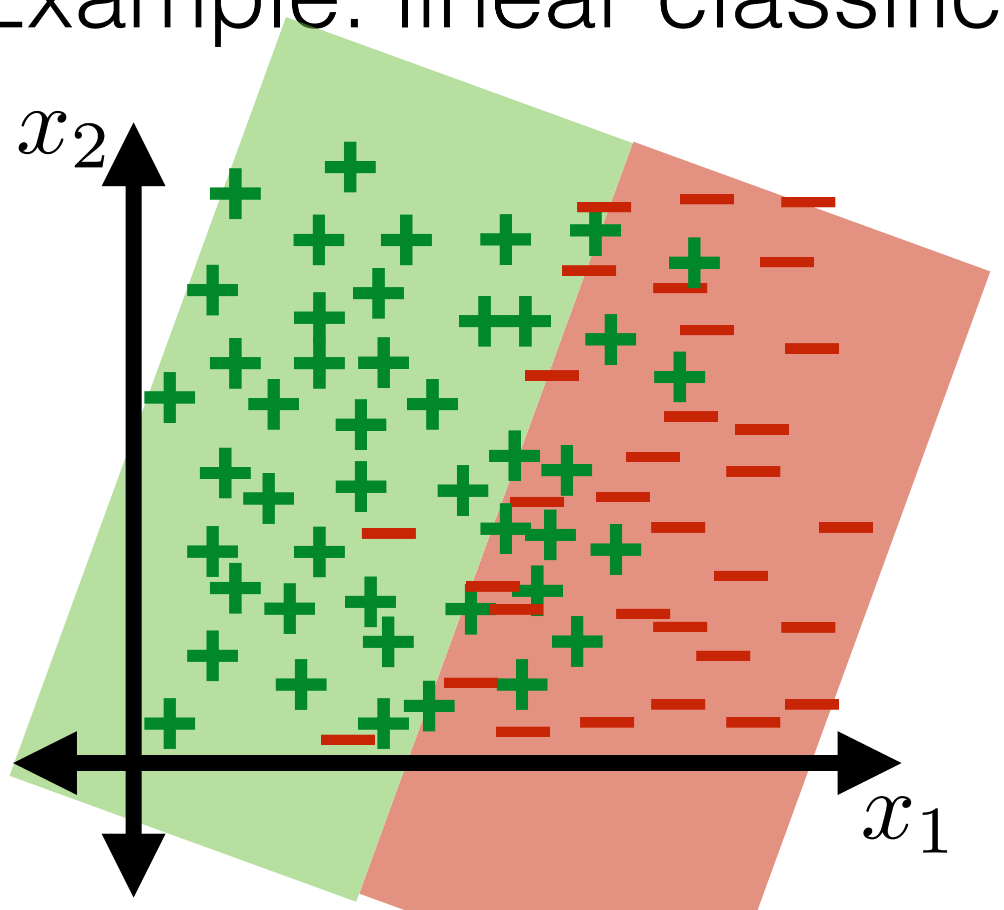
Recall

Classification

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \{-1, +1\}$
- Hypothesis $h : \mathbb{R}^d \to \{-1, +1\}$
- Loss: 0-1, asymmetric, NLL
- Example: linear classification



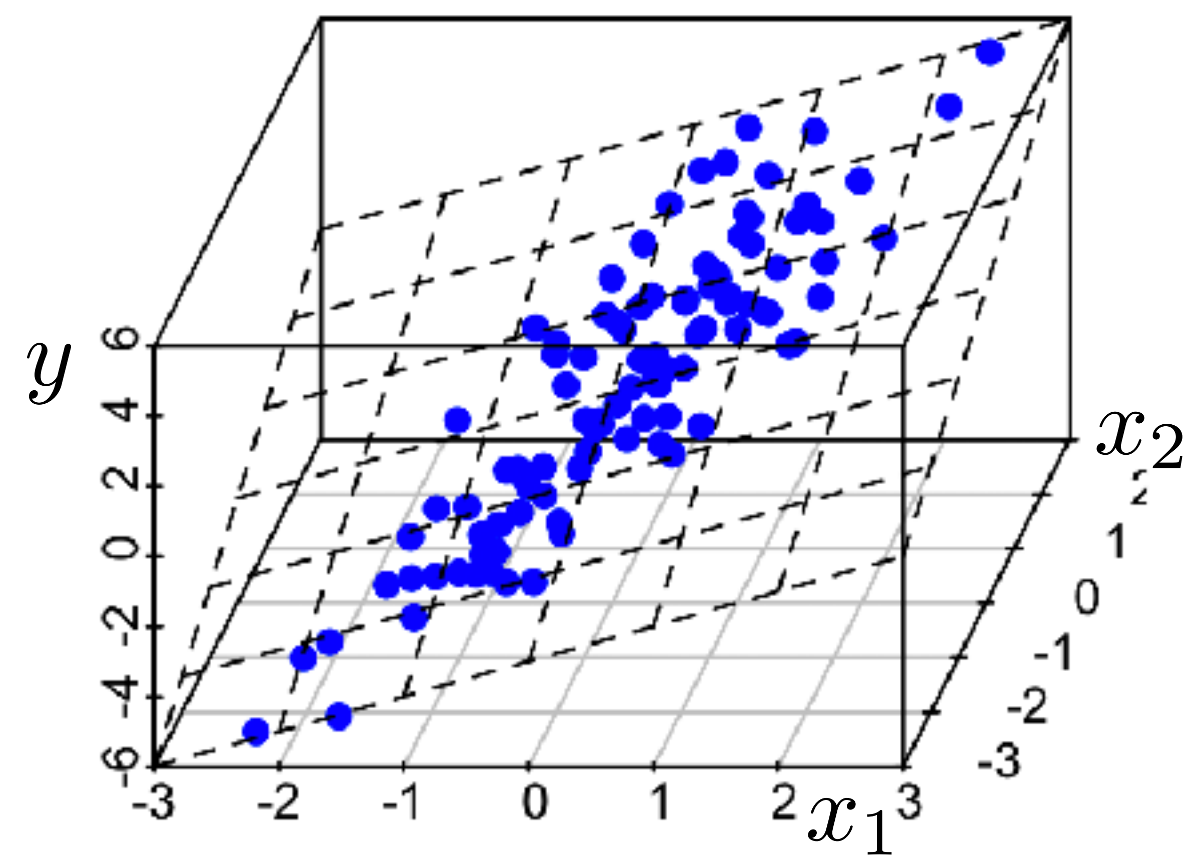
Compare

Regression

- Datum i: feature vector

$x^{(i)} = (x_1^{(i)}, \ldots, x_d^{(i)})^\top \in \mathbb{R}^d$

  - Label $y^{(i)} \in \mathbb{R}$
- Hypothesis $h : \mathbb{R}^d \to \mathbb{R}$
- Loss: $L(g, a) = (g - a)^2$
- Example: linear regression



Linear regression

- Hypotheses for linear regression: $h(x; \theta, \theta_0) = \theta^\top x + \theta_0$

Linear regression

- Hypotheses for linear regression: $h(x; \theta, \theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

Linear regression

- Hypotheses for linear regression: $h(x; \theta, \theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta, \theta_0) = \frac{1}{n} \sum_{i=1}^{n} L(h(x^{(i)}; \theta, \theta_0), y^{(i)})$$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0),y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature “1”

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0),y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} - y^{(i)})^2$$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature “1”

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underset{\text{1xd}}{\theta^\top} x^{(i)} - y^{(i)})^2$$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0),y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} - y^{(i)})^2$$

1xd,dx1

Linear regression

- Hypotheses for linear regression: $h(x; \theta, \theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)}; \theta, \theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{\theta^\top x^{(i)}}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2$$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature “1”

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{\theta^\top x^{(i)}}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2 = \frac{1}{n}\sum_{i=1}^{n}((x^{(i)})^\top\theta - y^{(i)})^2$$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{\theta^\top x^{(i)}}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2 = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{(x^{(i)})^\top}_{\text{1xd}}\theta - y^{(i)})^2$$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{\theta^\top x^{(i)}}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2 = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{(x^{(i)})^\top}_{\text{1xd}}\underbrace{\theta}_{\text{dx1}} - y^{(i)})^2$$

Linear regression

- Hypotheses for linear regression: $h(x; \theta, \theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta, \theta_0) = \frac{1}{n} \sum_{i=1}^{n} L(h(x^{(i)}; \theta, \theta_0), y^{(i)}) = \frac{1}{n} \sum_{i=1}^{n} (\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n} \sum_{i=1}^{n} (\underbrace{\theta^\top x^{(i)}}_{1\text{xd,dx1}} - \underbrace{y^{(i)}}_{1\text{x}1})^2 = \frac{1}{n} \sum_{i=1}^{n} (\underbrace{(x^{(i)})^\top}_{1\text{xd}}\underbrace{\theta}_{\text{dx1}} - \underbrace{y^{(i)}}_{1\text{x}1})^2$$

Linear regression

- Hypotheses for linear regression: $h(x; \theta, \theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta, \theta_0) = \frac{1}{n} \sum_{i=1}^{n} L(h(x^{(i)}; \theta, \theta_0), y^{(i)}) = \frac{1}{n} \sum_{i=1}^{n} (\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n} \sum_{i=1}^{n} (\underbrace{\theta^\top x^{(i)}}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2 = \frac{1}{n} \sum_{i=1}^{n} (\underbrace{(x^{(i)})^\top \theta}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2$$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature “1”

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underset{\text{1xd,dx1}}{\theta^\top x^{(i)}} - \underset{\text{1x1}}{y^{(i)}})^2 = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{(x^{(i)})^\top\theta}_{\text{1xd,dx1}} - \underset{\text{1x1}}{y^{(i)}})^2$$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underset{\text{1xd,dx1}}{\theta^\top x^{(i)}} - \underset{\text{1x1}}{y^{(i)}})^2 = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{(x^{(i)})^\top}_{\text{1xd,}}\underbrace{\theta}_{\text{dx1}} - \underset{\text{1x1}}{y^{(i)}})^2$$

Define $\tilde{X} = \begin{bmatrix} x_1^{(1)} & \cdots & x_d^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(n)} & \cdots & x_d^{(n)} \end{bmatrix}$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0),y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature “1”

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underset{\text{1xd,dx1}}{\theta^\top x^{(i)}} - \underset{\text{1x1}}{y^{(i)}})^2 = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{(x^{(i)})^\top}\underbrace{\theta}_{\text{1xd,dx1}} - \underset{\text{1x1}}{y^{(i)}})^2$$

Define $\underset{\text{nxd}}{\tilde{X}} = \begin{bmatrix} x_1^{(1)} & \cdots & x_d^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(n)} & \cdots & x_d^{(n)} \end{bmatrix}$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underset{\text{1xd,dx1}}{\theta^\top x^{(i)}} - \underset{\text{1x1}}{y^{(i)}})^2 = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{(x^{(i)})^\top}_{\text{1xd}}\underbrace{\theta}_{\text{dx1}} - \underset{\text{1x1}}{y^{(i)}})^2$$

Define $\underset{\text{nxd}}{\tilde{X}} = \begin{bmatrix} x_1^{(1)} & \cdots & x_d^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(n)} & \cdots & x_d^{(n)} \end{bmatrix}$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{\theta^\top x^{(i)}}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2 = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{(x^{(i)})^\top\theta}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2$$

Define $\underset{\text{nxd}}{\tilde{X}} = \begin{bmatrix} x_1^{(1)} & \cdots & x_d^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(n)} & \cdots & x_d^{(n)} \end{bmatrix}$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{\theta^\top x^{(i)}}_{1\text{xd},\text{dx}1} - \underbrace{y^{(i)}}_{1\text{x}1})^2 = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{(x^{(i)})^\top}_{1\text{xd}}\underbrace{\theta}_{\text{dx}1} - \underbrace{y^{(i)}}_{1\text{x}1})^2$$

Define $\underset{\text{nxd}}{\tilde{X}} = \begin{bmatrix} x_1^{(1)} & \cdots & x_d^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(n)} & \cdots & x_d^{(n)} \end{bmatrix}$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{\theta^\top x^{(i)}}_{1\text{xd},\text{dx}1} - \underbrace{y^{(i)}}_{1\text{x}1})^2 = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{\underbrace{(x^{(i)})}\underbrace{{}^\top\theta}}_{1\text{xd},\text{dx}1} - \underbrace{y^{(i)}}_{1\text{x}1})^2$$

Define $\underbrace{\tilde{X}}_{\text{nxd}} = \begin{bmatrix} x_1^{(1)} & \cdots & x_d^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(n)} & \cdots & x_d^{(n)} \end{bmatrix}$ $\qquad \tilde{Y} = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(n)} \end{bmatrix}$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{\theta^\top x^{(i)}}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2 = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{(x^{(i)})^\top \theta}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2$$

Define $\underbrace{\tilde{X}}_{\text{nxd}} = \begin{bmatrix} x_1^{(1)} & \cdots & x_d^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(n)} & \cdots & x_d^{(n)} \end{bmatrix}$ $\qquad \underbrace{\tilde{Y}}_{\text{nx1}} = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(n)} \end{bmatrix}$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature “1”

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{\theta^\top x^{(i)}}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2 = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{(x^{(i)})^\top\theta}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2$$

Define $\underset{\text{nxd}}{\tilde{X}} = \begin{bmatrix} x_1^{(1)} & \cdots & x_d^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(n)} & \cdots & x_d^{(n)} \end{bmatrix}$ $\qquad \underset{\text{nx1}}{\tilde{Y}} = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(n)} \end{bmatrix}$

Linear regression

- Hypotheses for linear regression: $h(x; \theta, \theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta, \theta_0) = \frac{1}{n} \sum_{i=1}^{n} L(h(x^{(i)}; \theta, \theta_0), y^{(i)}) = \frac{1}{n} \sum_{i=1}^{n} (\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n} \sum_{i=1}^{n} (\underset{\text{1xd,dx1}}{\theta^\top x^{(i)}} - \underset{\text{1x1}}{y^{(i)}})^2 = \frac{1}{n} \sum_{i=1}^{n} (\underbrace{(x^{(i)})^\top \theta}_{\text{1xd,dx1}} - \underset{\text{1x1}}{y^{(i)}})^2$$

$$\tilde{X}\theta - \tilde{Y}$$

Define $\underset{\text{nxd}}{\tilde{X}} = \begin{bmatrix} x_1^{(1)} & \cdots & x_d^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(n)} & \cdots & x_d^{(n)} \end{bmatrix}$ $\qquad \underset{\text{nx1}}{\tilde{Y}} = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(n)} \end{bmatrix}$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{\theta^\top x^{(i)}}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2 = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{(x^{(i)})^\top\theta}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2$$

$$\tilde{X}\theta - \tilde{Y}$$

Define $\underset{\text{nxd}}{\tilde{X}} = \begin{bmatrix} x_1^{(1)} & \cdots & x_d^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(n)} & \cdots & x_d^{(n)} \end{bmatrix}$ $\qquad \underset{\text{nx1}}{\tilde{Y}} = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(n)} \end{bmatrix}$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{\theta^\top x^{(i)}}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2 = \frac{1}{n}\sum_{i=1}^{n}(\boxed{\underbrace{(x^{(i)})^\top\theta}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}}})^2$$

$$\underbrace{\tilde{X}}_{\text{nxd}}\theta - \tilde{Y}$$

Define $\underbrace{\tilde{X}}_{\text{nxd}} = \begin{bmatrix} x_1^{(1)} & \cdots & x_d^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(n)} & \cdots & x_d^{(n)} \end{bmatrix}$ $\quad \underbrace{\tilde{Y}}_{\text{nx1}} = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(n)} \end{bmatrix}$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{\theta^\top x^{(i)}}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2 = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{(x^{(i)})^\top\theta}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2$$

$$\underbrace{\tilde{X}\theta}_{\text{nxd,dx1}} - \tilde{Y}$$

Define $\underbrace{\tilde{X}}_{\text{nxd}} = \begin{bmatrix} x_1^{(1)} & \cdots & x_d^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(n)} & \cdots & x_d^{(n)} \end{bmatrix}$ $\quad\underbrace{\tilde{Y}}_{\text{nx1}} = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(n)} \end{bmatrix}$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underset{\text{1xd,dx1}}{\theta^\top x^{(i)}} - \underset{\text{1x1}}{y^{(i)}})^2 = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{(x^{(i)})^\top}_{\text{1xd,dx1}}\theta - \underset{\text{1x1}}{y^{(i)}})^2$$

$$\underset{\text{nxd,dx1}}{\tilde{X}\theta} - \underset{\text{nx1}}{\tilde{Y}}$$

Define $\underset{\text{nxd}}{\tilde{X}} = \begin{bmatrix} x_1^{(1)} & \cdots & x_d^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(n)} & \cdots & x_d^{(n)} \end{bmatrix}$ $\qquad \underset{\text{nx1}}{\tilde{Y}} = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(n)} \end{bmatrix}$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{\theta^\top x^{(i)}}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2 = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{(x^{(i)})^\top \theta}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2$$

$$= \frac{1}{n}\|\underbrace{\tilde{X}\theta}_{\text{nxd,dx1}} - \underbrace{\tilde{Y}}_{\text{nx1}}\|^2$$

Define $\underbrace{\tilde{X}}_{\text{nxd}} = \begin{bmatrix} x_1^{(1)} & \cdots & x_d^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(n)} & \cdots & x_d^{(n)} \end{bmatrix}$ $\qquad \underbrace{\tilde{Y}}_{\text{nx1}} = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(n)} \end{bmatrix}$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{\theta^\top x^{(i)}}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2 = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{(x^{(i)})^\top \theta}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2$$

$$= \frac{1}{n}\|\underbrace{\tilde{X}\theta}_{\text{nxd,dx1}} - \underbrace{\tilde{Y}}_{\text{nx1}}\|^2$$

Define $\underbrace{\tilde{X}}_{\text{nxd}} = \begin{bmatrix} x_1^{(1)} & \cdots & x_d^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(n)} & \cdots & x_d^{(n)} \end{bmatrix}$ $\underbrace{\tilde{Y}}_{\text{nx1}} = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(n)} \end{bmatrix}$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underset{\text{1xd,dx1}}{\theta^\top x^{(i)}} - \underset{\text{1x1}}{y^{(i)}})^2 = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{(x^{(i)})^\top}_{\text{1xd,dx1}}\theta - \underset{\text{1x1}}{y^{(i)}})^2$$

$$= \frac{1}{n}\|\underset{\text{nxd,dx1}}{\tilde{X}\theta} - \underset{\text{nx1}}{\tilde{Y}}\|^2 = \frac{1}{n}(\tilde{X}\theta - \tilde{Y}) \quad (\tilde{X}\theta - \tilde{Y}) \qquad \boxed{?}$$

Define $\underset{\text{nxd}}{\tilde{X}} = \begin{bmatrix} x_1^{(1)} & \cdots & x_d^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(n)} & \cdots & x_d^{(n)} \end{bmatrix}$ $\qquad \underset{\text{nx1}}{\tilde{Y}} = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(n)} \end{bmatrix}$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature "1"

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{\theta^\top x^{(i)}}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2 = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{(x^{(i)})^\top\theta}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2$$

$$= \frac{1}{n}\|\underbrace{\tilde{X}\theta}_{\text{nxd,dx1}} - \underbrace{\tilde{Y}}_{\text{nx1}}\|^2 = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$$

Define $\underbrace{\tilde{X}}_{\text{nxd}} = \begin{bmatrix} x_1^{(1)} & \cdots & x_d^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(n)} & \cdots & x_d^{(n)} \end{bmatrix}$ $\qquad \underbrace{\tilde{Y}}_{\text{nx1}} = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(n)} \end{bmatrix}$

Linear regression

- Hypotheses for linear regression: $h(x;\theta,\theta_0) = \theta^\top x + \theta_0$
- Training error (using squared error loss)

$$J(\theta,\theta_0) = \frac{1}{n}\sum_{i=1}^{n} L(h(x^{(i)};\theta,\theta_0), y^{(i)}) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- Training error if we augment features with feature “1”

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{\theta^\top x^{(i)}}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2 = \frac{1}{n}\sum_{i=1}^{n}(\underbrace{(x^{(i)})^\top\theta}_{\text{1xd,dx1}} - \underbrace{y^{(i)}}_{\text{1x1}})^2$$

$$= \frac{1}{n}\|\underbrace{\tilde{X}\theta}_{\text{nxd,dx1}} - \underbrace{\tilde{Y}}_{\text{nx1}}\|^2 = \frac{1}{n}\underbrace{(\tilde{X}\theta - \tilde{Y})^\top}_{\text{1xn}}\underbrace{(\tilde{X}\theta - \tilde{Y})}_{\text{nx1}}$$

Define $\underset{\text{nxd}}{\tilde{X}} = \begin{bmatrix} x_1^{(1)} & \cdots & x_d^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(n)} & \cdots & x_d^{(n)} \end{bmatrix}$ $\quad \underset{\text{nx1}}{\tilde{Y}} = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(n)} \end{bmatrix}$

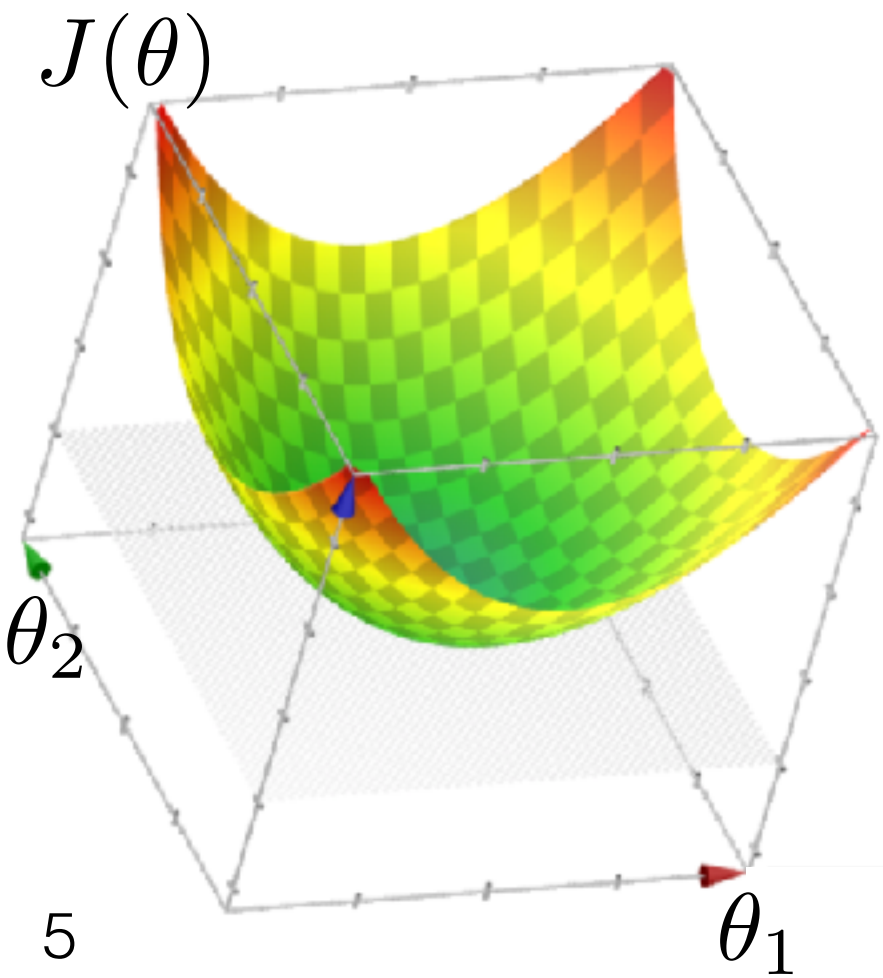
- Goal: minimize $J(\theta) = \dfrac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$

Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$

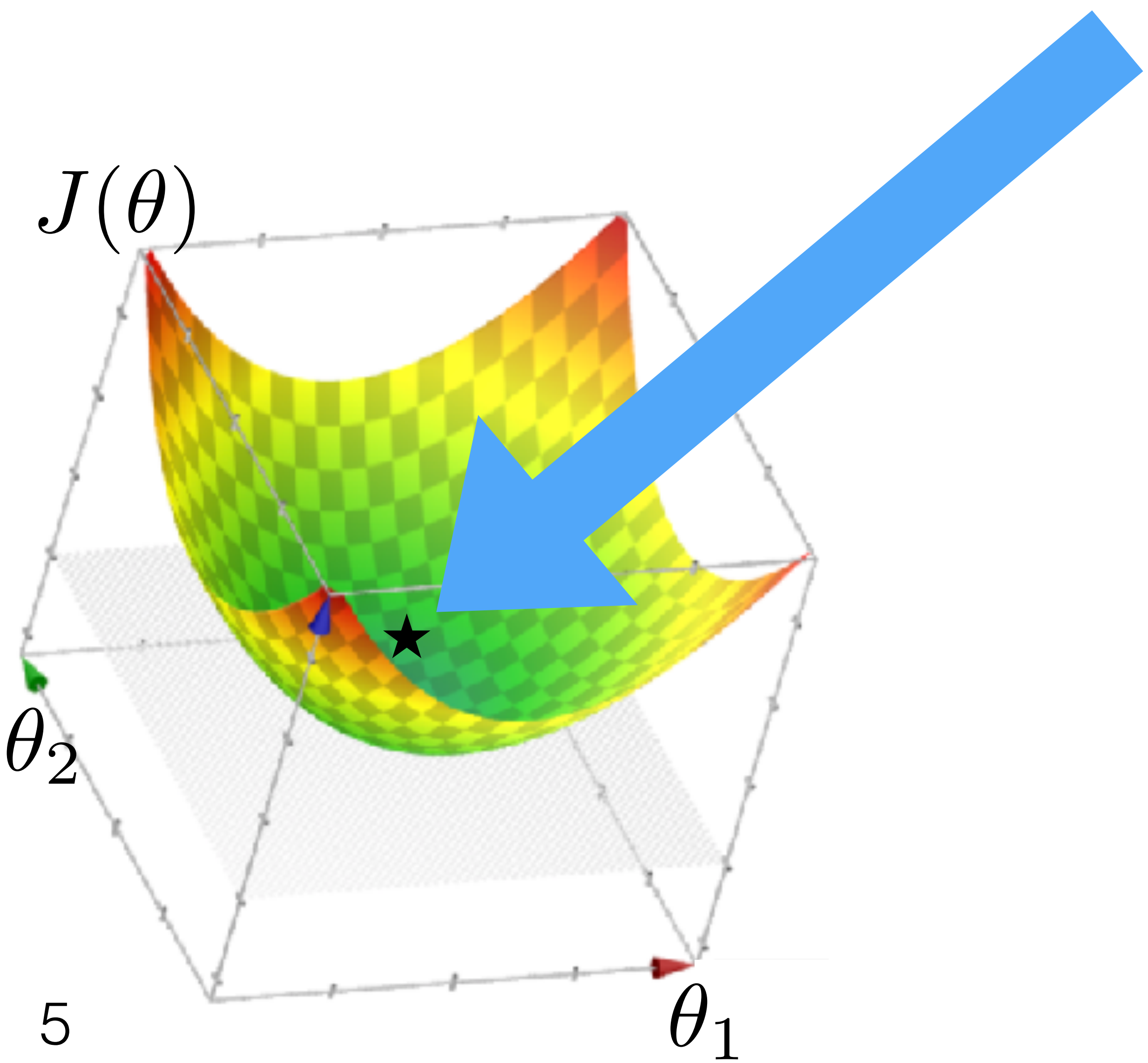
Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$



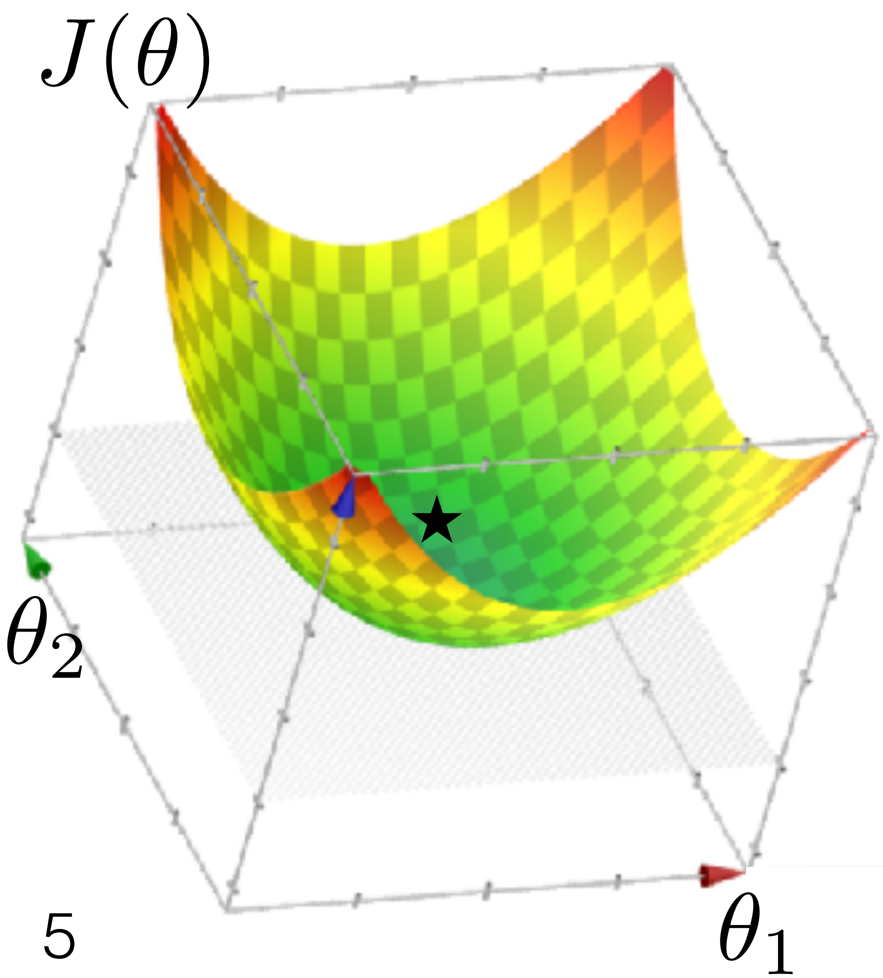
Linear regression: A Direct Solution

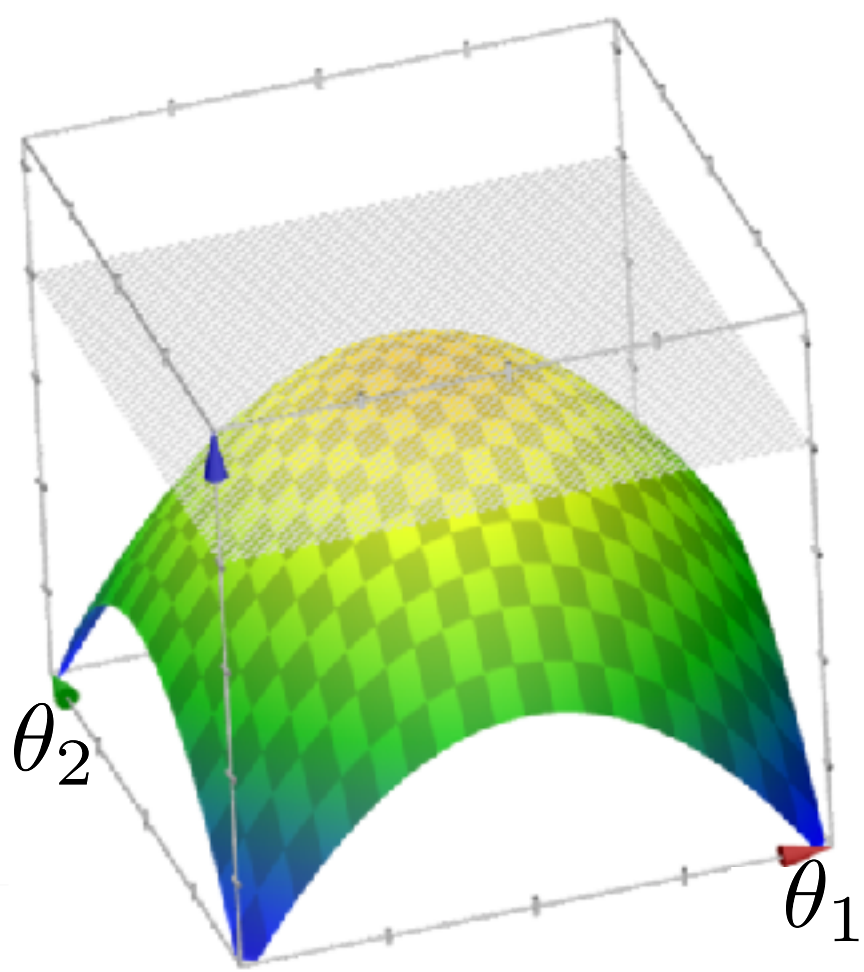
- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$



Linear regression: A Direct Solution

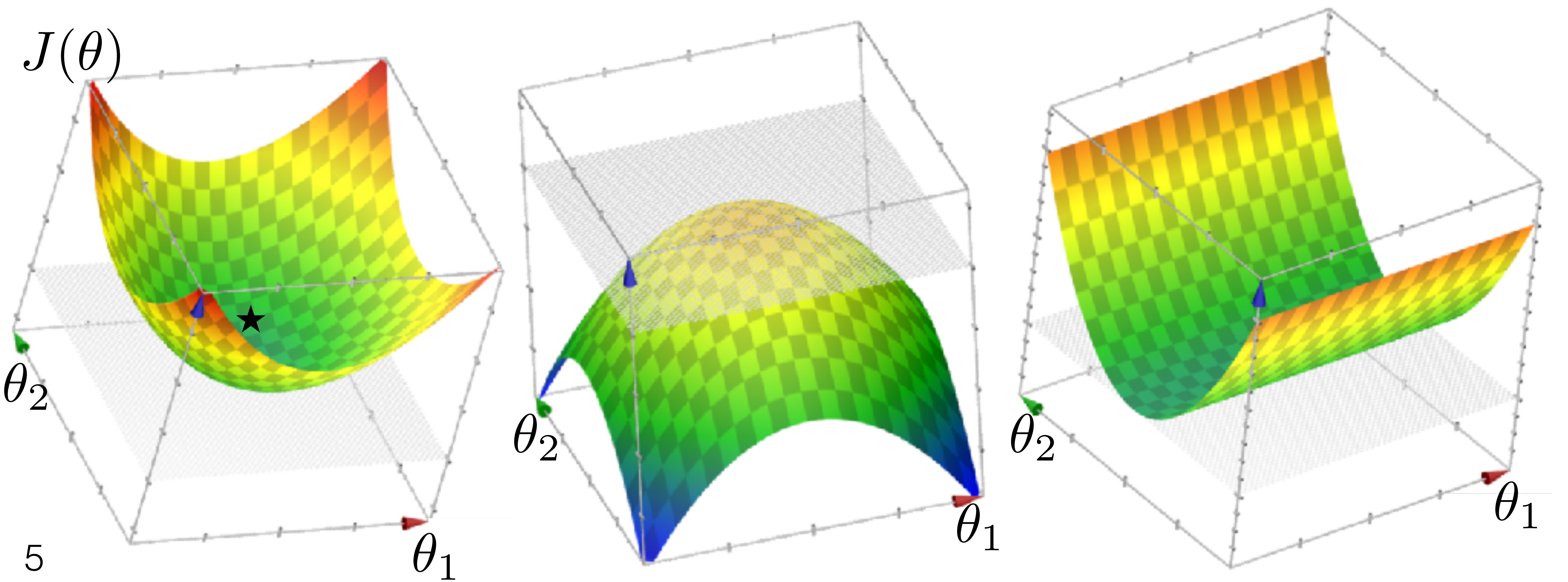
- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$





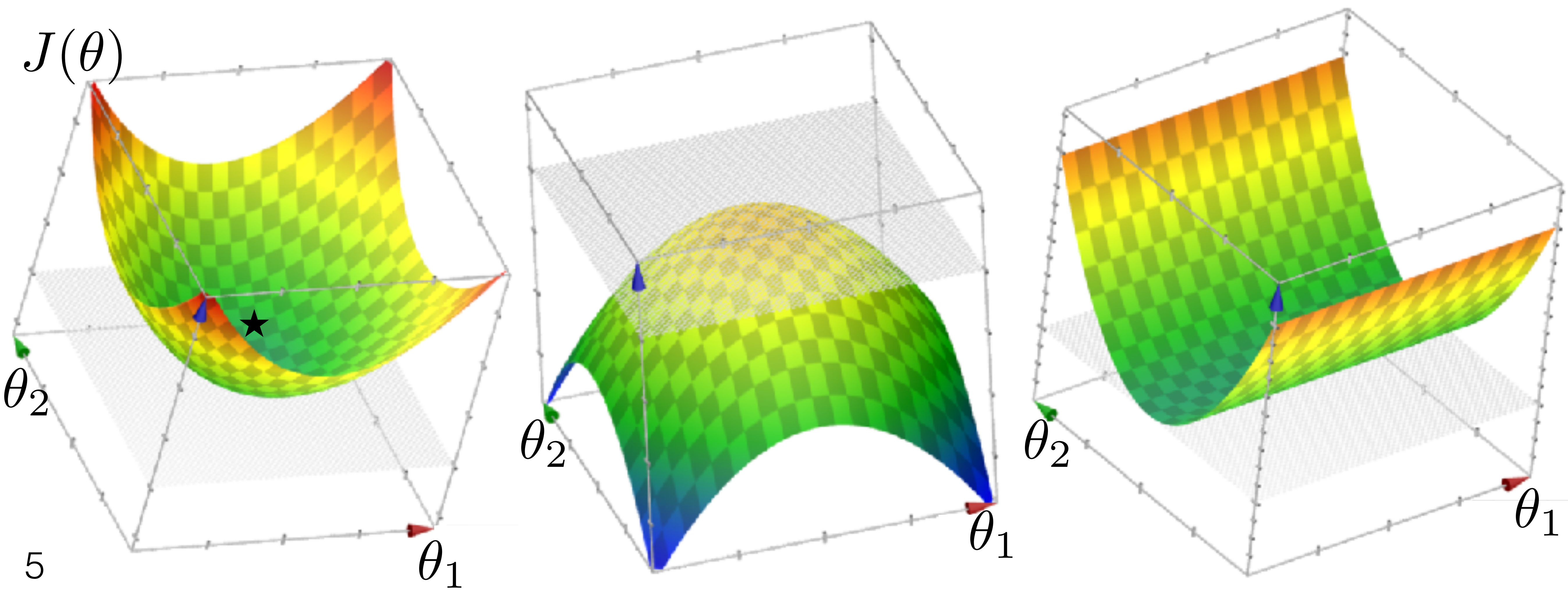
Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$



Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]



Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]

Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient

Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\nabla_\theta J(\theta)$

Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\underbrace{\nabla_\theta J(\theta)}_{\text{dx1}}$

Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\underbrace{\nabla_\theta J(\theta)}_{\text{dx1}} = \frac{2}{n}\tilde{X}^\top(\tilde{X}\theta - \tilde{Y})$

Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\underbrace{\nabla_\theta J(\theta)}_{\text{dx1}} = \frac{2}{n}\tilde{X}^\top(\tilde{X}\theta - \tilde{Y})$

Exercise: check n,d=1

Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\underbrace{\nabla_\theta J(\theta)}_{\text{dx1}} = \frac{2}{n}\tilde{X}^\top(\tilde{X}\theta - \tilde{Y})$

Exercise: check the vector elements

Exercise: check n,d=1

Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\nabla_\theta J(\theta) = \frac{2}{n}\tilde{X}^\top(\tilde{X}\theta - \tilde{Y})$

dx1

nxd

Exercise: check the vector elements

Exercise: check n,d=1

Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\nabla_\theta J(\theta) = \frac{2}{n}\tilde{X}^\top(\tilde{X}\theta - \tilde{Y})$

dx1 nxd,dx1

Exercise: check n,d=1

Exercise: check the vector elements

Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\underbrace{\nabla_\theta J(\theta)}_{d\times 1} = \frac{2}{n}\tilde{X}^\top(\underbrace{\tilde{X}\theta}_{n\times d,\, d\times 1} - \underbrace{\tilde{Y}}_{n\times 1})$

Exercise: check the vector elements

Exercise: check n,d=1

Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\nabla_\theta J(\theta) = \frac{2}{n}\tilde{X}^\top(\tilde{X}\theta - \tilde{Y})$

dx1 dxn nxd,dx1 nx1

Exercise: check the vector elements

Exercise: check n,d=1

Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\underbrace{\nabla_\theta J(\theta)}_{\text{dx1}} = \frac{2}{n}\underbrace{\tilde{X}^\top}_{\text{dxn}}(\underbrace{\tilde{X}\theta}_{\text{nxd,dx1}} - \underbrace{\tilde{Y}}_{\text{nx1}}) \overset{\text{set}}{=} 0$

Exercise: check the vector elements

Exercise: check n,d=1

Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\underbrace{\nabla_\theta J(\theta)}_{dx1} = \frac{2}{n}\underbrace{\tilde{X}^\top}_{dxn}(\underbrace{\tilde{X}\theta}_{nxd,dx1} - \underbrace{\tilde{Y}}_{nx1}) \overset{set}{=} 0$

$$\tilde{X}^\top\tilde{X}\theta - \tilde{X}^\top\tilde{Y} = 0$$

Exercise: check the vector elements

Exercise: check n,d=1

Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\nabla_\theta J(\theta) = \frac{2}{n}\tilde{X}^\top(\tilde{X}\theta - \tilde{Y}) \overset{\text{set}}{=} 0$

dx1 dxn nxd,dx1 nx1

$$\tilde{X}^\top\tilde{X}\theta - \tilde{X}^\top\tilde{Y} = 0$$

$$\tilde{X}^\top\tilde{X}\theta = \tilde{X}^\top\tilde{Y}$$

Exercise: check the vector elements

Exercise: check n,d=1

Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\underbrace{\nabla_\theta J(\theta)}_{\text{dx1}} = \frac{2}{n}\underbrace{\tilde{X}^\top}_{\text{dxn}}(\underbrace{\tilde{X}\theta}_{\text{nxd,dx1}} - \underbrace{\tilde{Y}}_{\text{nx1}}) \overset{\text{set}}{=} 0$

$$\tilde{X}^\top\tilde{X}\theta - \tilde{X}^\top\tilde{Y} = 0$$

$$(\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{X}\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

Exercise: check the vector elements

Exercise: check n,d=1

Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\nabla_\theta J(\theta) = \frac{2}{n}\tilde{X}^\top(\tilde{X}\theta - \tilde{Y}) \overset{\text{set}}{=} 0$

dx1 dxn nxd,dx1 nx1

$$\tilde{X}^\top\tilde{X}\theta - \tilde{X}^\top\tilde{Y} = 0$$

$$(\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{X}\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

$$\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

Exercise: check the vector elements

Exercise: check n,d=1

Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\underbrace{\nabla_\theta J(\theta)}_{\text{dx1}} = \frac{2}{n}\underbrace{\tilde{X}^\top}_{\text{dxn}}(\underbrace{\tilde{X}\theta}_{\text{nxd,dx1}} - \underbrace{\tilde{Y}}_{\text{nx1}}) \overset{\text{set}}{=} 0$

$$\tilde{X}^\top\tilde{X}\theta - \tilde{X}^\top\tilde{Y} = 0$$

$$(\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{X}\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

$$\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

Exercise: check the vector elements

Exercise: check n,d=1

- Matrix of second derivatives $\frac{2}{n}\tilde{X}^\top\tilde{X}$

Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\nabla_\theta J(\theta) = \frac{2}{n}\tilde{X}^\top(\tilde{X}\theta - \tilde{Y}) \overset{\text{set}}{=} 0$

dx1 dxn nxd,dx1 nx1

$$\tilde{X}^\top\tilde{X}\theta - \tilde{X}^\top\tilde{Y} = 0$$

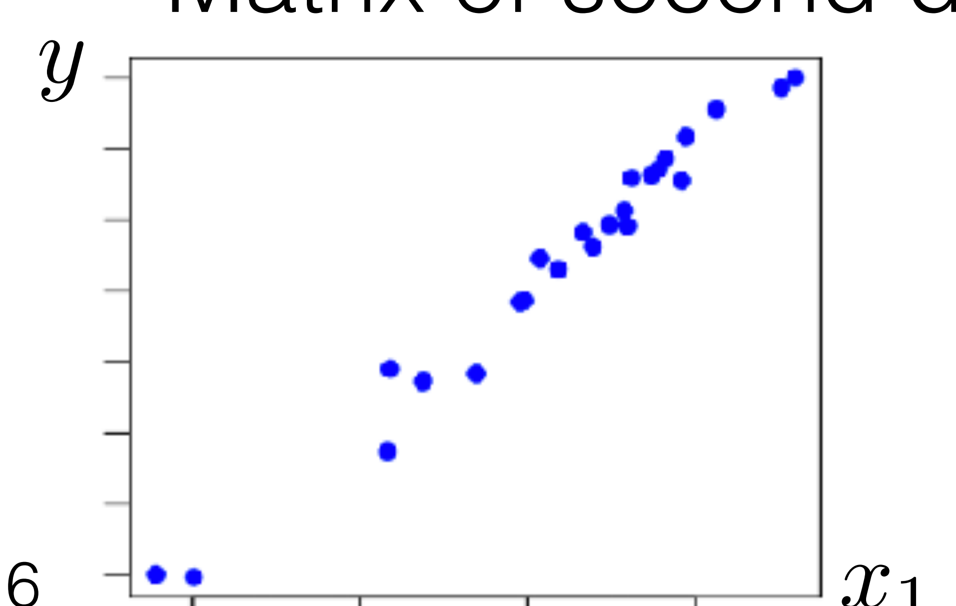
$$(\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{X}\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

$$\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

Exercise: check the vector elements

Exercise: check n,d=1

- Matrix of second derivatives $\frac{2}{n}\tilde{X}^\top\tilde{X}$



Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\nabla_\theta J(\theta) = \frac{2}{n}\tilde{X}^\top(\tilde{X}\theta - \tilde{Y}) \overset{\text{set}}{=} 0$

dx1 dxn nxd,dx1 nx1

$$\tilde{X}^\top\tilde{X}\theta - \tilde{X}^\top\tilde{Y} = 0$$

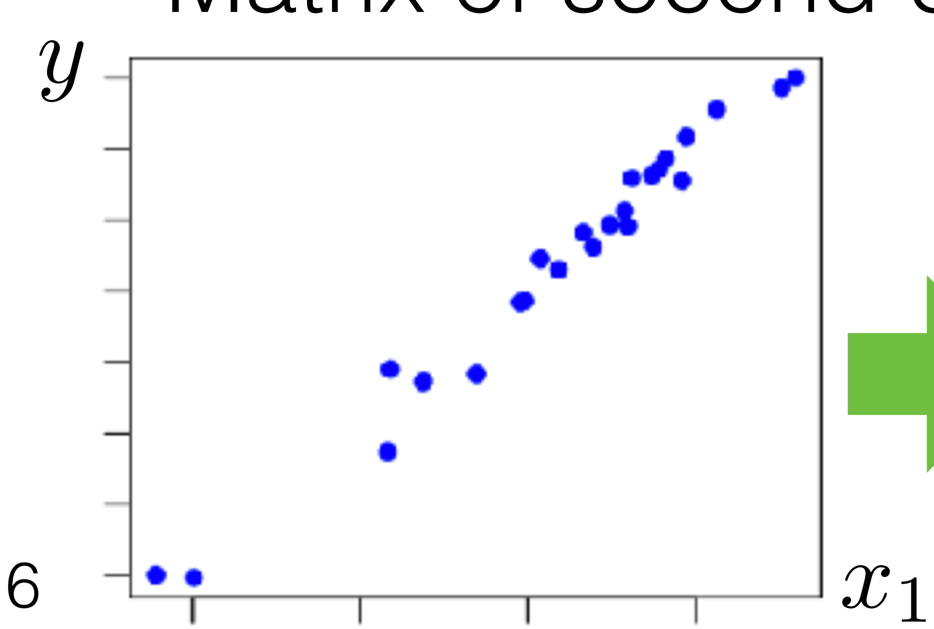
$$(\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{X}\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

$$\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

Exercise: check the vector elements

Exercise: check n,d=1

- Matrix of second derivatives $\frac{2}{n}\tilde{X}^\top\tilde{X}$



Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\underbrace{\nabla_\theta J(\theta)}_{dx1} = \frac{2}{n}\underbrace{\tilde{X}^\top}_{dxn}(\underbrace{\tilde{X}\theta}_{nxd,dx1} - \underbrace{\tilde{Y}}_{nx1}) \overset{\text{set}}{=} 0$

$$\tilde{X}^\top\tilde{X}\theta - \tilde{X}^\top\tilde{Y} = 0$$

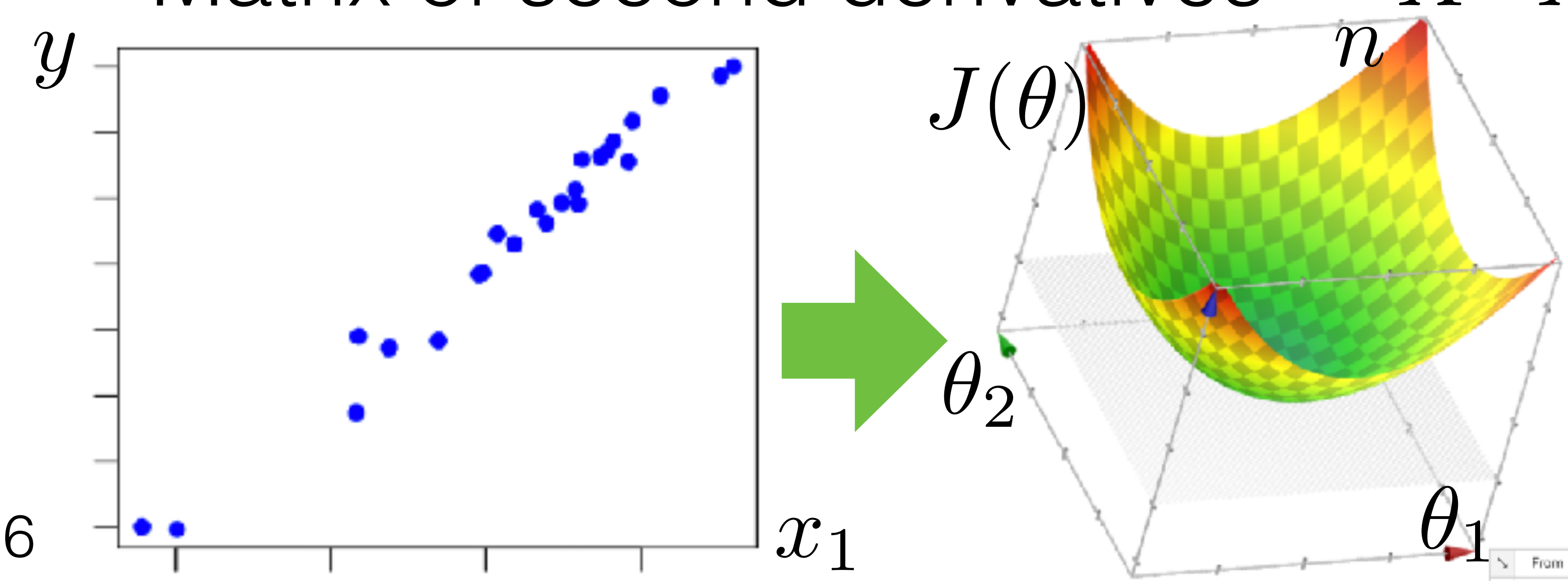
$$(\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{X}\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

$$\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

Exercise: check the vector elements

Exercise: check n,d=1

- Matrix of second derivatives $\frac{2}{n}\tilde{X}^\top\tilde{X}$



Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\underbrace{\nabla_\theta J(\theta)}_{\text{dx1}} = \frac{2}{n}\underbrace{\tilde{X}^\top}_{\text{dxn}}(\underbrace{\tilde{X}\theta}_{\text{nxd,dx1}} - \underbrace{\tilde{Y}}_{\text{nx1}}) \overset{\text{set}}{=} 0$

$$\tilde{X}^\top\tilde{X}\theta - \tilde{X}^\top\tilde{Y} = 0$$

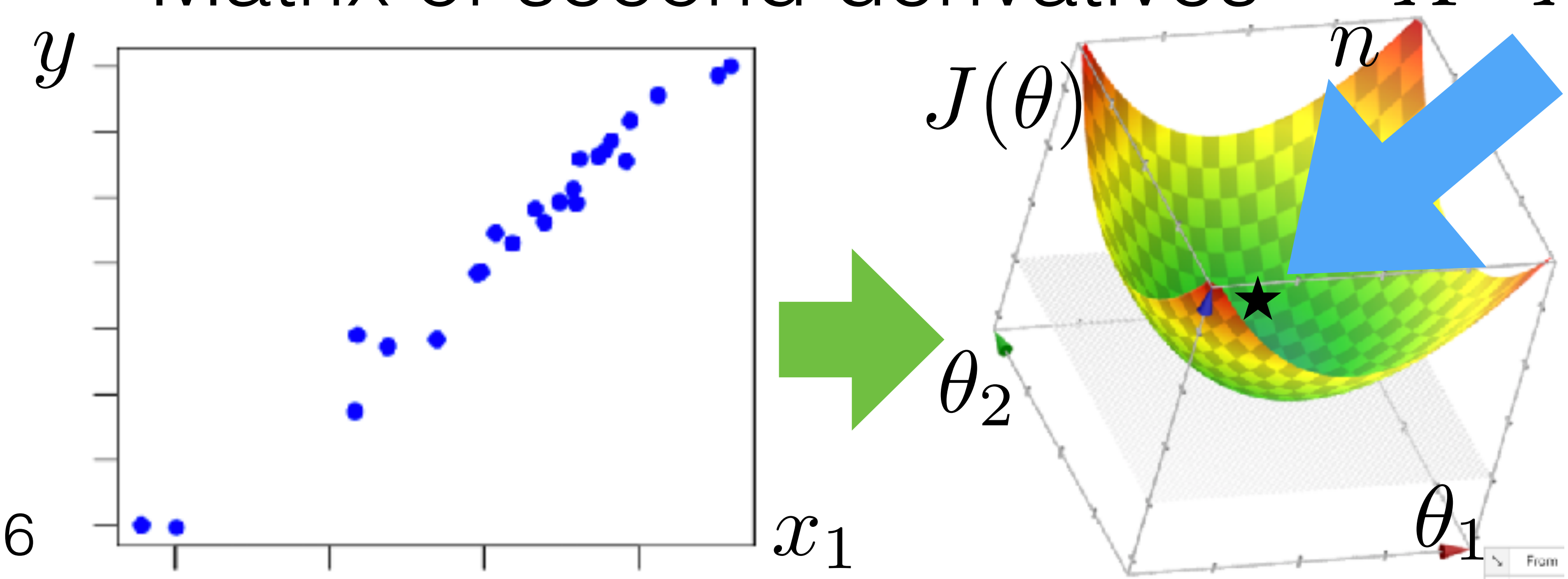
$$(\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{X}\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

$$\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

Exercise: check the vector elements

Exercise: check n,d=1

- Matrix of second derivatives $\frac{2}{n}\tilde{X}^\top\tilde{X}$



Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\nabla_\theta J(\theta) = \frac{2}{n}\tilde{X}^\top(\tilde{X}\theta - \tilde{Y}) \overset{\text{set}}{=} 0$

dx1 dxn nxd,dx1 nx1

Exercise: check the vector elements

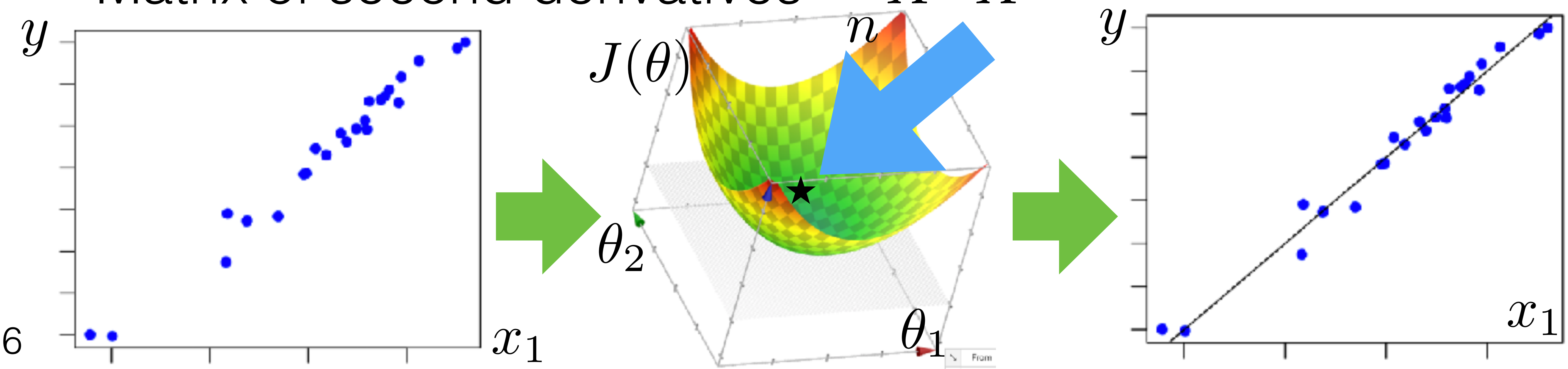
Exercise: check n,d=1

$$\tilde{X}^\top\tilde{X}\theta - \tilde{X}^\top\tilde{Y} = 0$$

$$(\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{X}\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

$$\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

- Matrix of second derivatives $\frac{2}{n}\tilde{X}^\top\tilde{X}$



Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\nabla_\theta J(\theta) = \frac{2}{n}\tilde{X}^\top(\tilde{X}\theta - \tilde{Y}) \overset{\text{set}}{=} 0$

dx1 dxn nxd,dx1 nx1

$$\tilde{X}^\top\tilde{X}\theta - \tilde{X}^\top\tilde{Y} = 0$$

$$(\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{X}\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

$$\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

Exercise: check the vector elements

Exercise: check n,d=1

- Matrix of second derivatives $\frac{2}{n}\tilde{X}^\top\tilde{X}$

Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\underbrace{\nabla_\theta J(\theta)}_{\text{dx1}} = \frac{2}{n}\underbrace{\tilde{X}^\top}_{\text{dxn}}(\underbrace{\tilde{X}\theta}_{\text{nxd,dx1}} - \underbrace{\tilde{Y}}_{\text{nx1}}) \overset{\text{set}}{=} 0$

$$\tilde{X}^\top\tilde{X}\theta - \tilde{X}^\top\tilde{Y} = 0$$

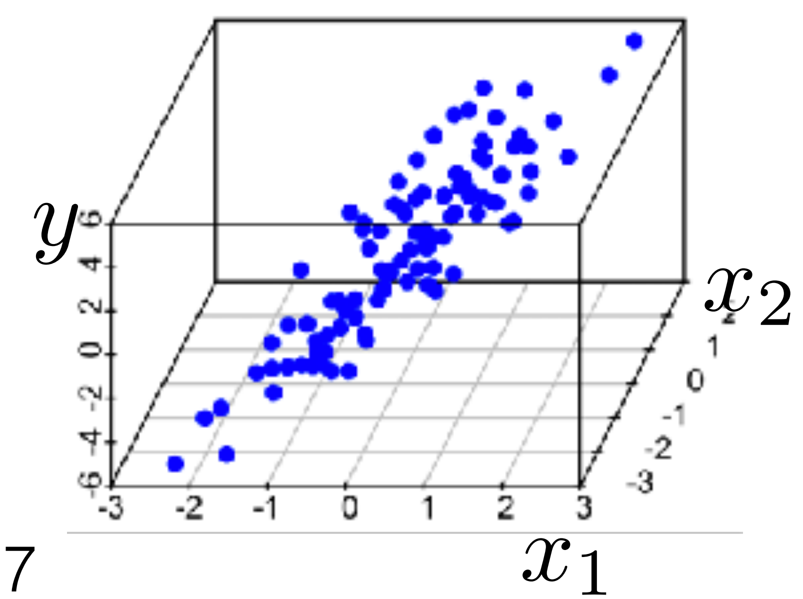
$$(\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{X}\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

$$\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

Exercise: check the vector elements

Exercise: check n,d=1

- Matrix of second derivatives $\frac{2}{n}\tilde{X}^\top\tilde{X}$



Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\underbrace{\nabla_\theta J(\theta)}_{dx1} = \frac{2}{n}\underbrace{\tilde{X}^\top}_{dxn}\underbrace{(\tilde{X}\theta}_{nxd,dx1} - \underbrace{\tilde{Y})}_{nx1} \overset{\text{set}}{=} 0$

$$\tilde{X}^\top\tilde{X}\theta - \tilde{X}^\top\tilde{Y} = 0$$

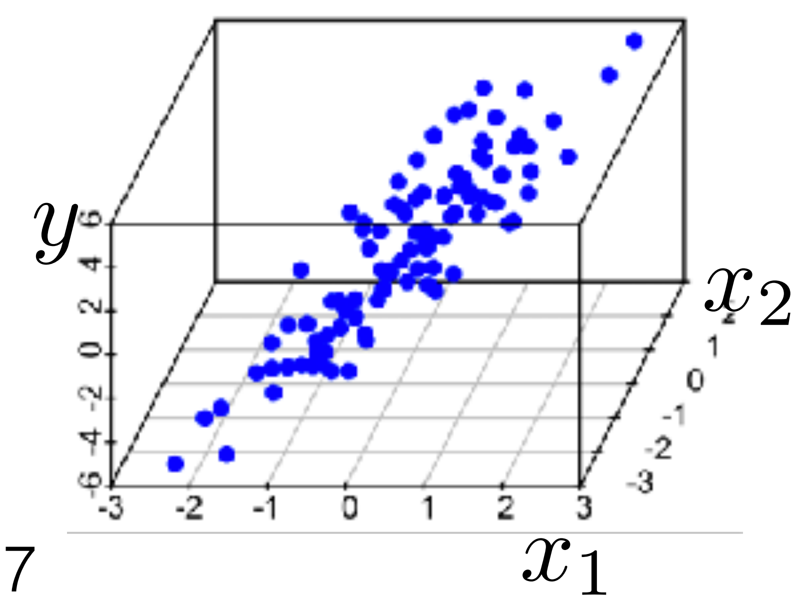
$$(\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{X}\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

$$\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

Exercise: check the vector elements

Exercise: check n,d=1

- Matrix of second derivatives $\frac{2}{n}\tilde{X}^\top\tilde{X}$



Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\underbrace{\nabla_\theta J(\theta)}_{\text{dx1}} = \frac{2}{n}\underbrace{\tilde{X}^\top}_{\text{dxn}}(\underbrace{\tilde{X}\theta}_{\text{nxd,dx1}} - \underbrace{\tilde{Y}}_{\text{nx1}}) \overset{\text{set}}{=} 0$

Exercise: check the vector elements

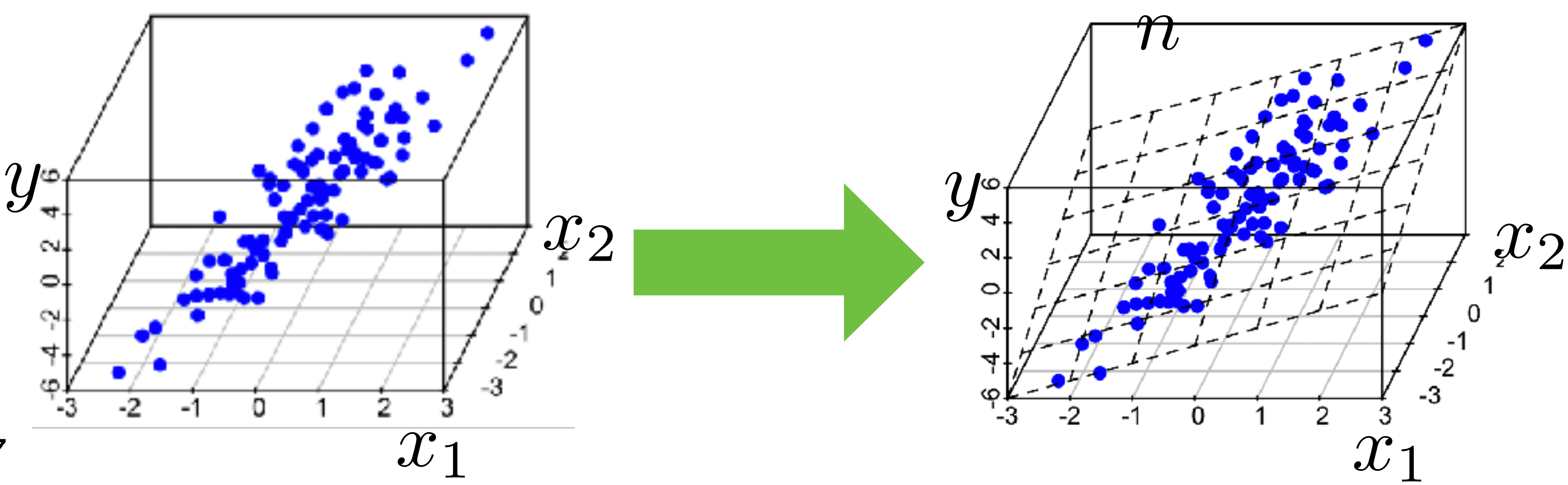
Exercise: check n,d=1

$$\tilde{X}^\top\tilde{X}\theta - \tilde{X}^\top\tilde{Y} = 0$$

$$(\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{X}\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

$$\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

- Matrix of second derivatives $\frac{2}{n}\tilde{X}^\top\tilde{X}$



Linear regression: A Direct Solution

- Goal: minimize $J(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y})$
- Uniquely minimized at a point if gradient at that point is zero and function "curves up" [see linear algebra]
- Gradient $\underset{\text{dx1}}{\nabla_\theta J(\theta)} = \frac{2}{n}\underset{\text{dxn}}{\tilde{X}^\top}\underset{\text{nxd,dx1}}{(\tilde{X}\theta} - \underset{\text{nx1}}{\tilde{Y})} \overset{\text{set}}{=} 0$

$$\tilde{X}^\top\tilde{X}\theta - \tilde{X}^\top\tilde{Y} = 0$$

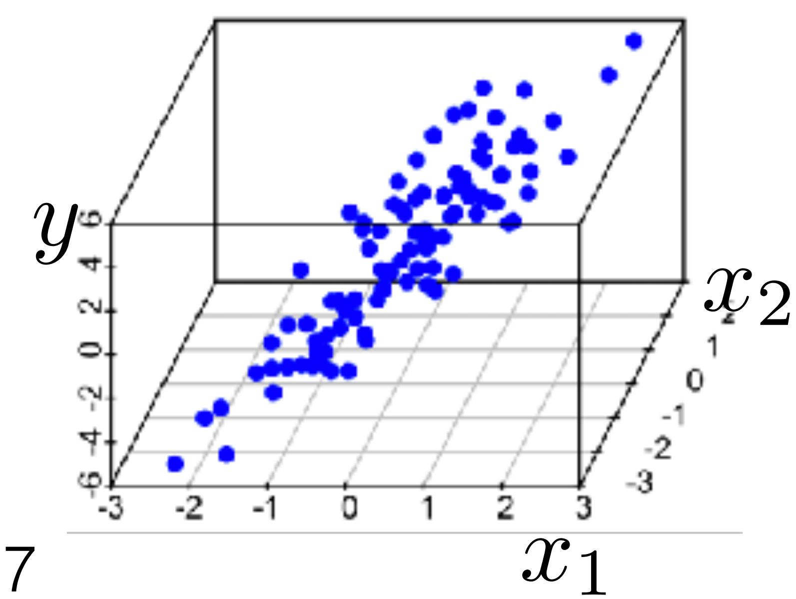
$$(\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{X}\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

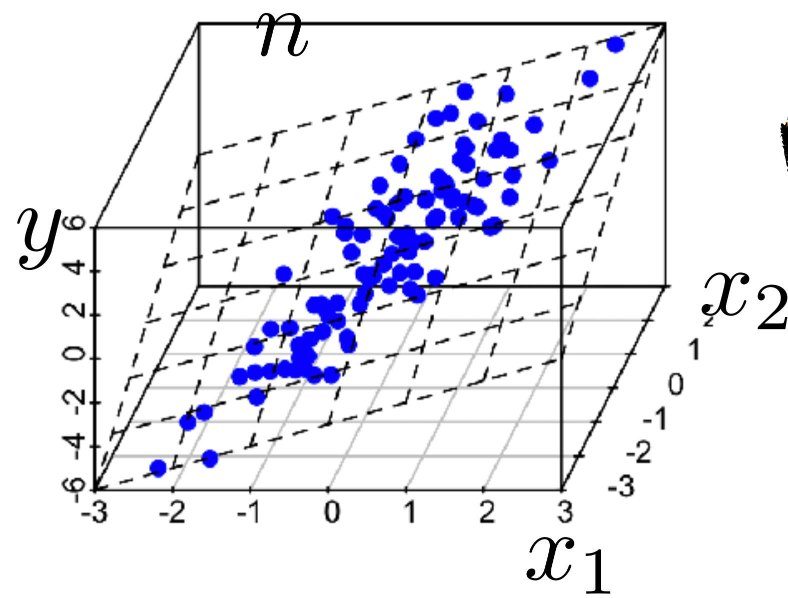
$$\theta = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{Y}$$

Exercise: check the vector elements

Exercise: check n,d=1

- Matrix of second derivatives $\frac{2}{n}\tilde{X}^\top\tilde{X}$





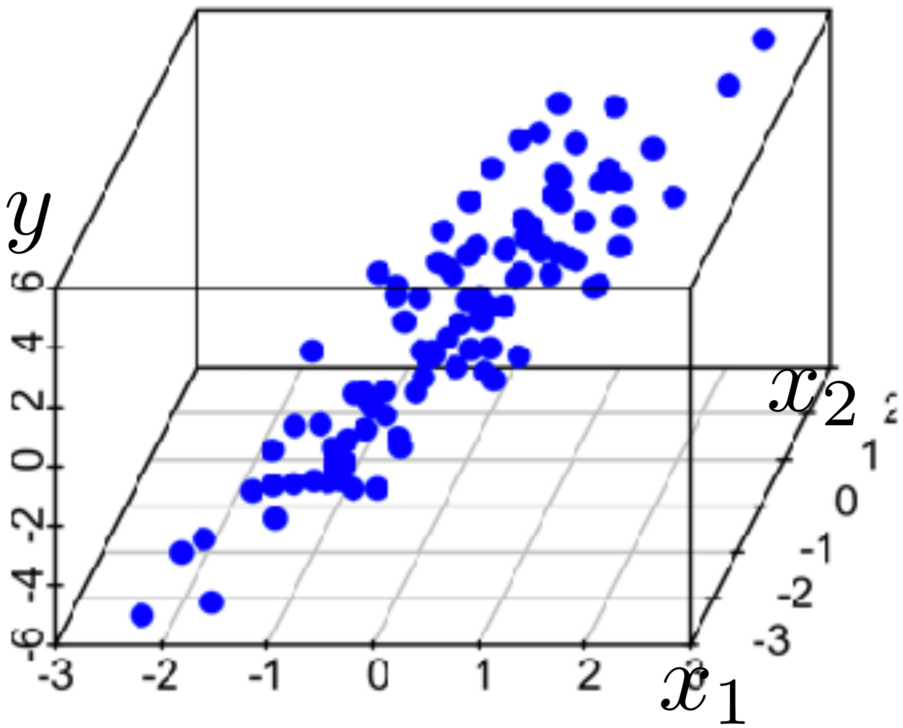
Note: hypothesis is a hyperplane!

What can go wrong in practice?

- Sometimes there isn't a unique best hyperplane

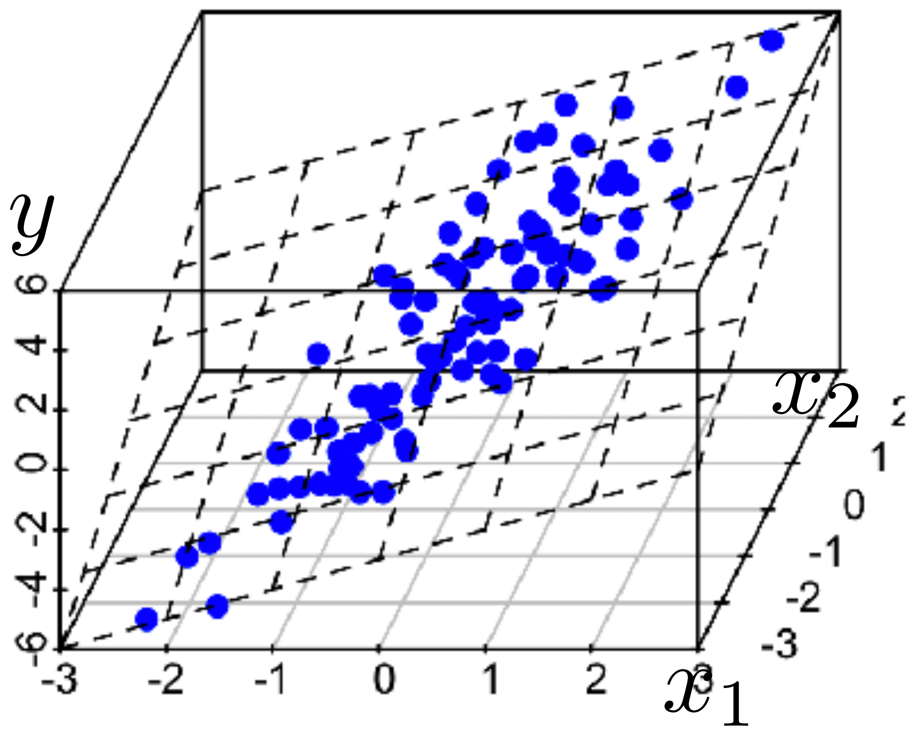
What can go wrong in practice?

- Sometimes there isn't a unique best hyperplane



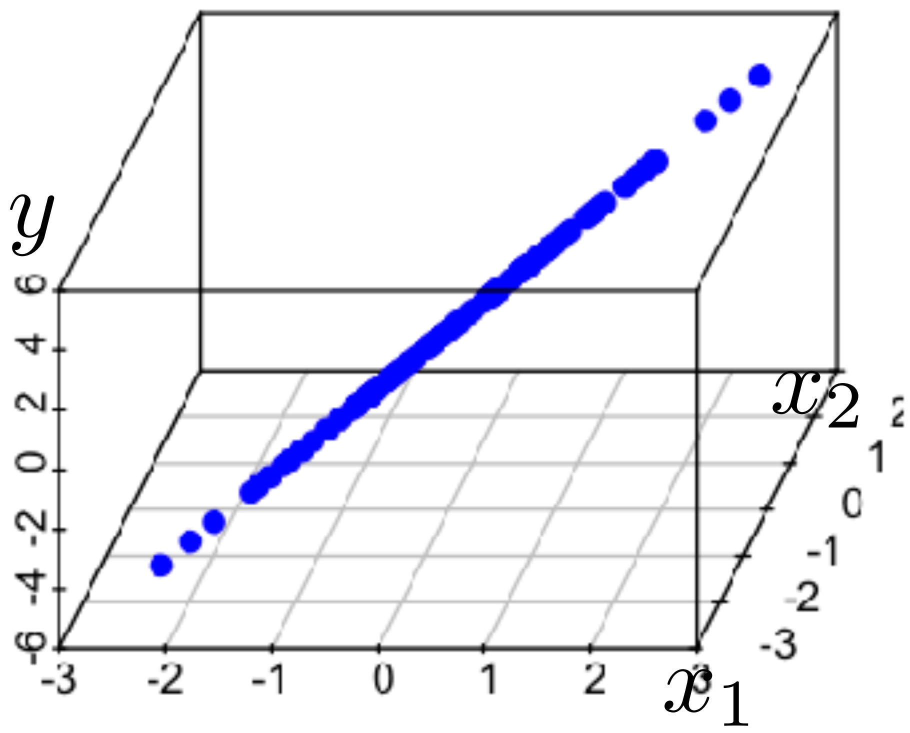
What can go wrong in practice?

- Sometimes there isn't a unique best hyperplane



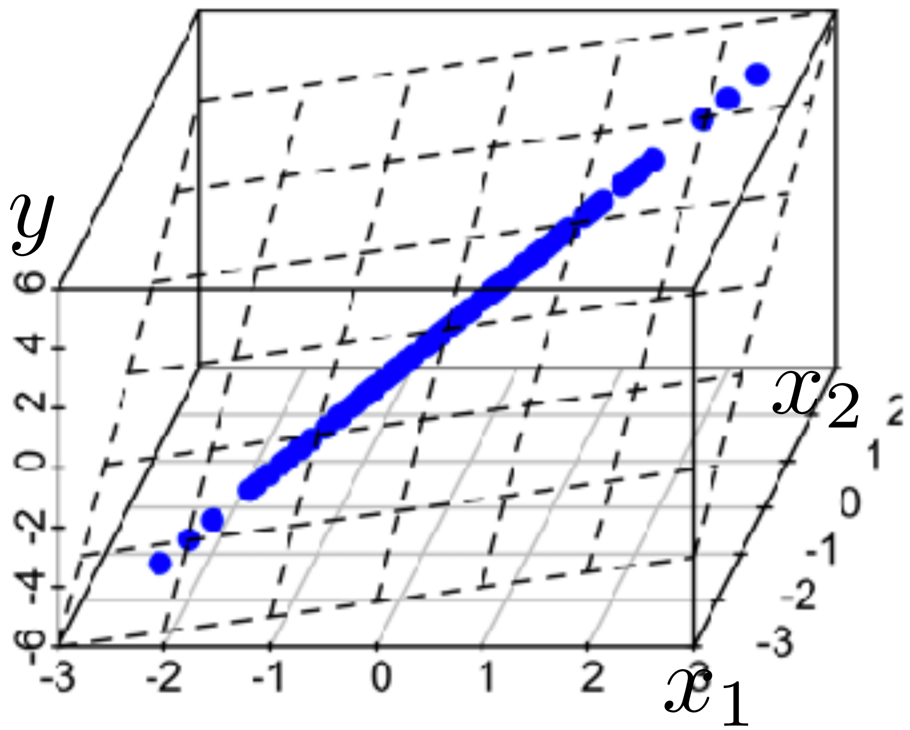
What can go wrong in practice?

- Sometimes there isn't a unique best hyperplane



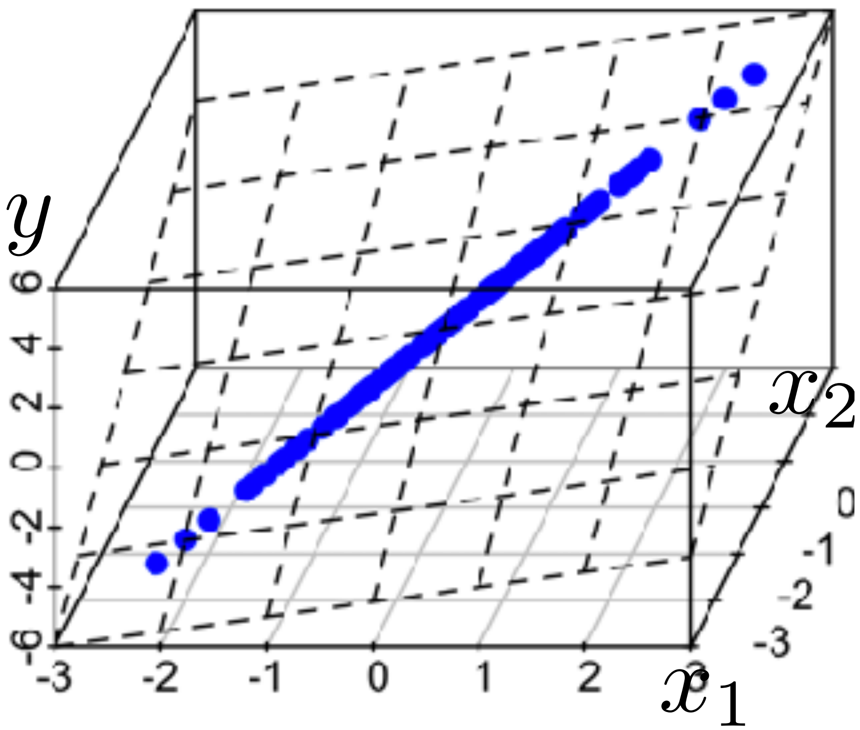
What can go wrong in practice?

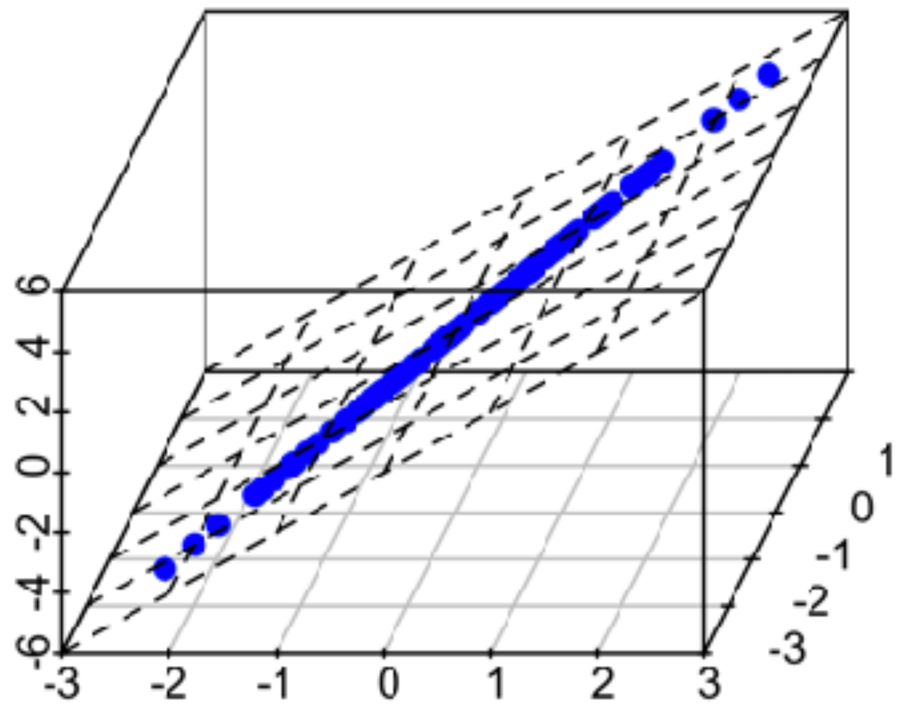
- Sometimes there isn't a unique best hyperplane



What can go wrong in practice?

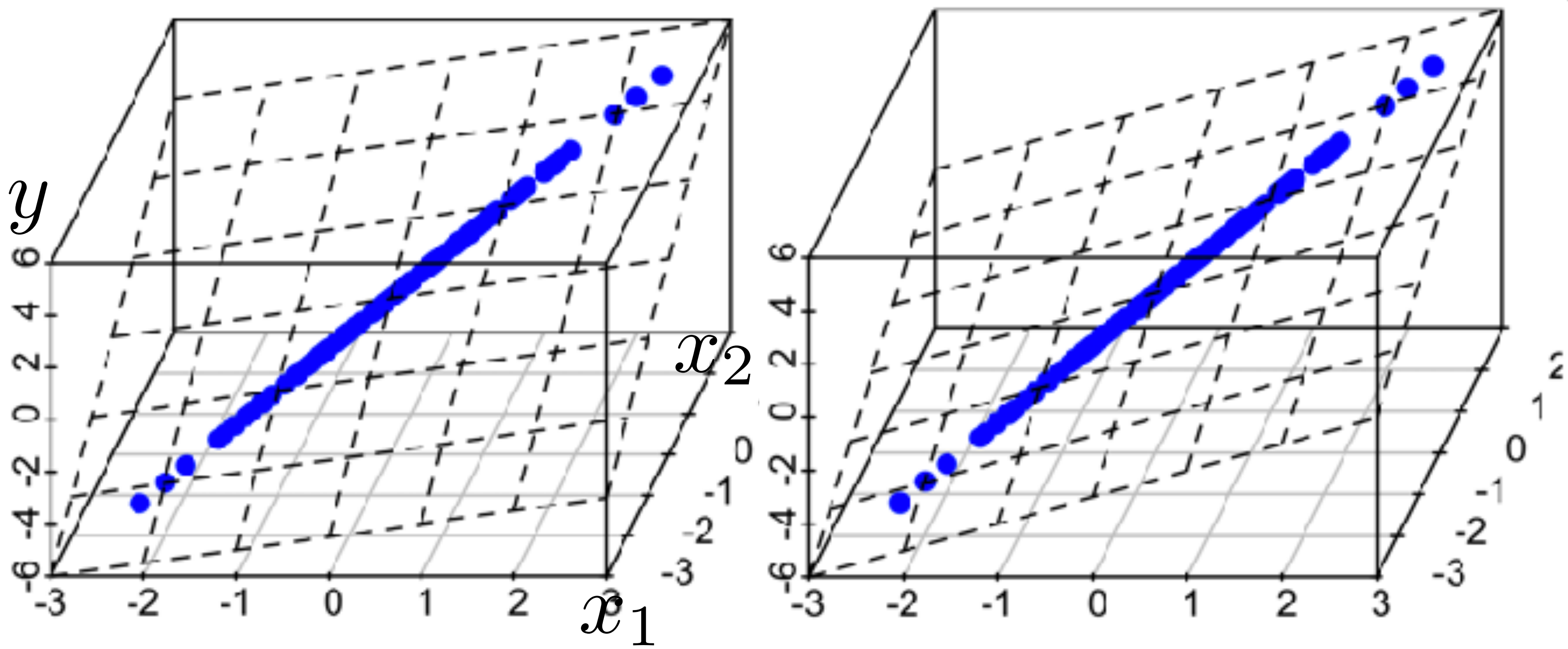
- Sometimes there isn't a unique best hyperplane





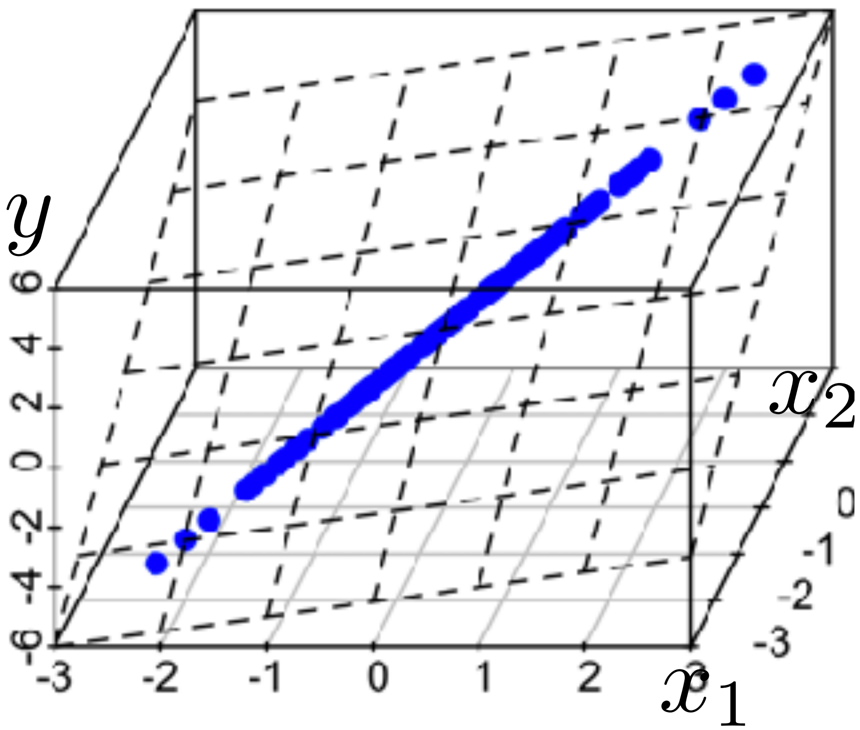
What can go wrong in practice?

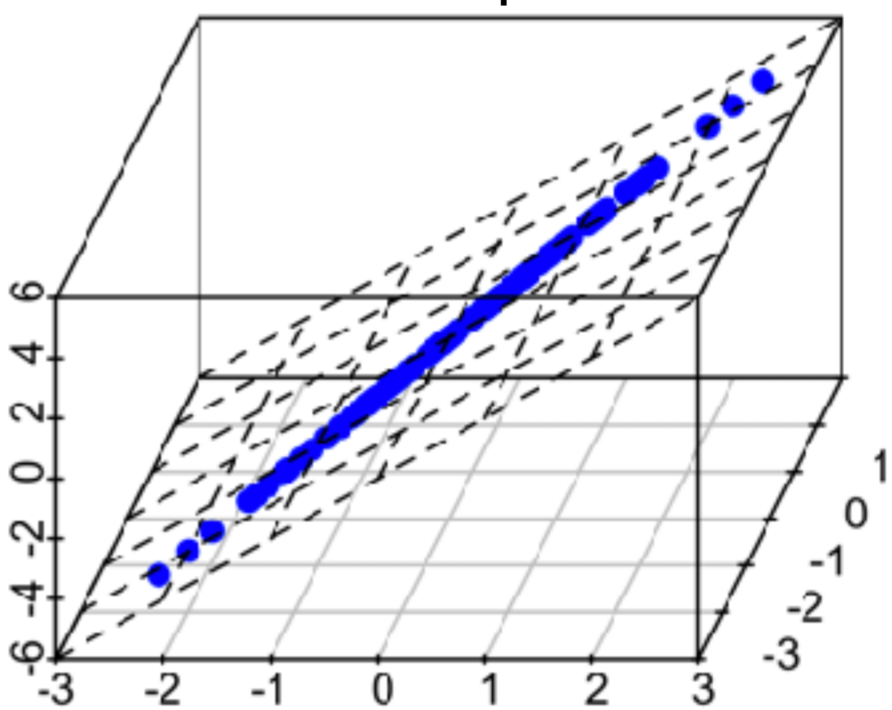
- Sometimes there isn't a unique best hyperplane



What can go wrong in practice?

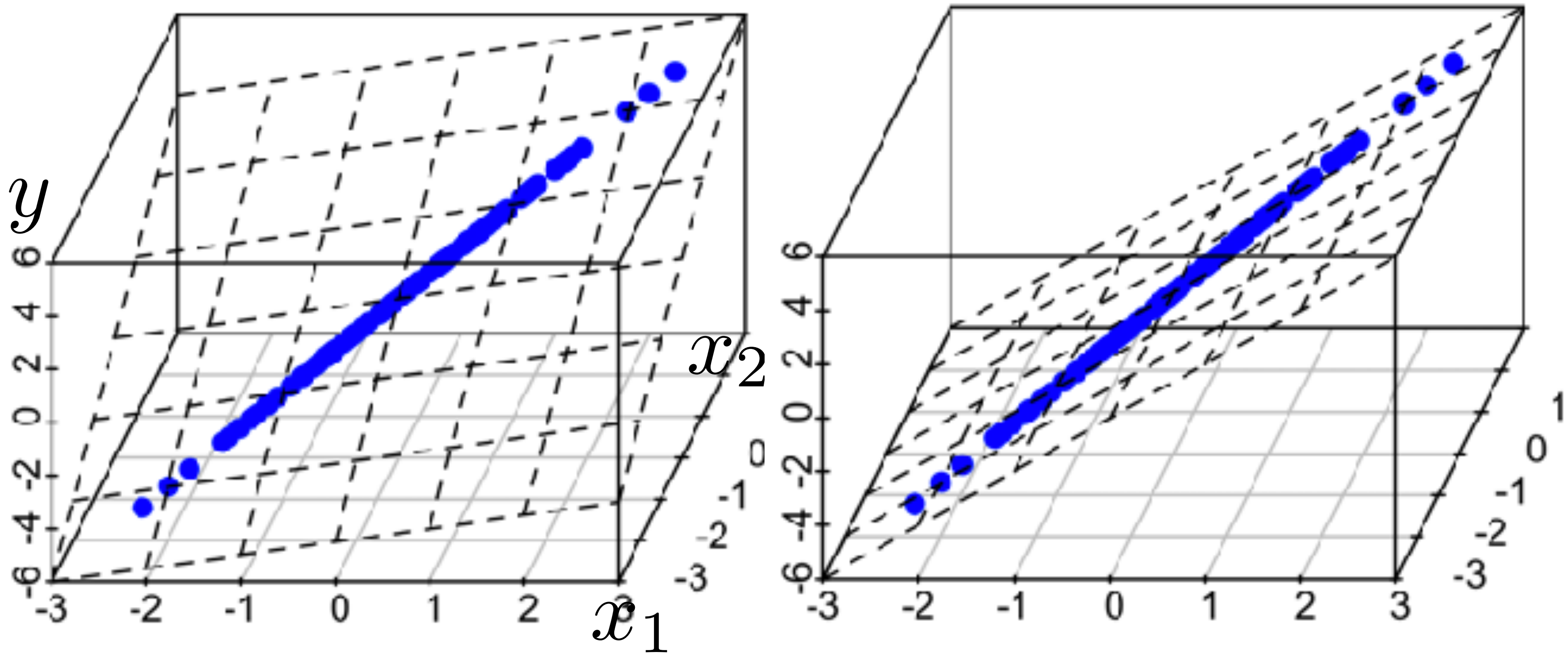
- Sometimes there isn't a unique best hyperplane





What can go wrong in practice?

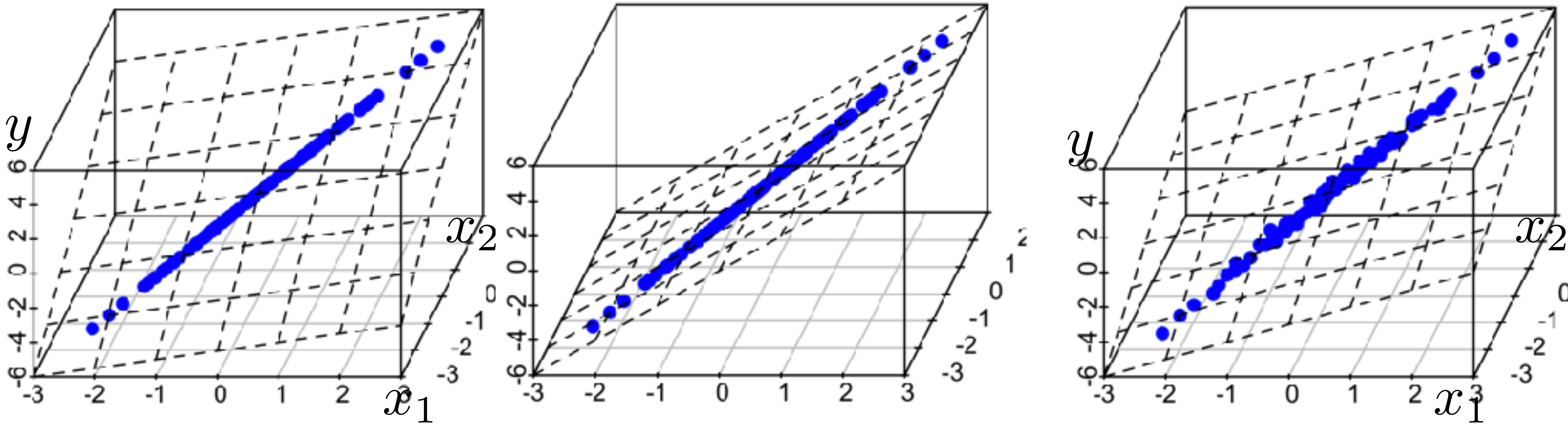
- Sometimes there isn't a unique best hyperplane



- Sometimes there's technically a unique best hyperplane, but just because of noise

What can go wrong in practice?

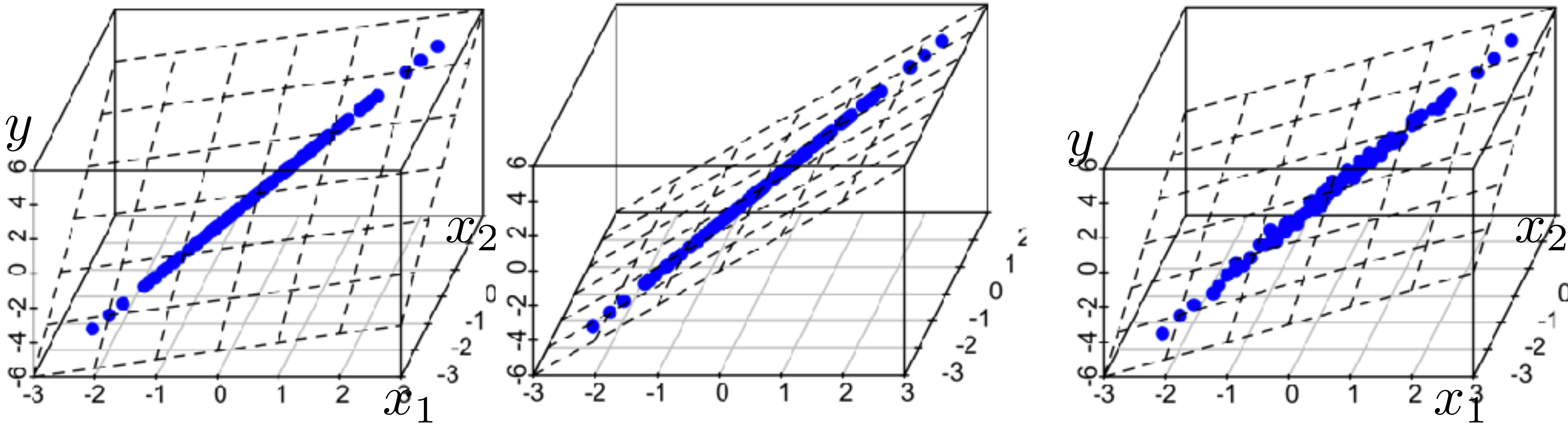
- Sometimes there isn't a unique best hyperplane



- Sometimes there's technically a unique best hyperplane, but just because of noise

What can go wrong in practice?

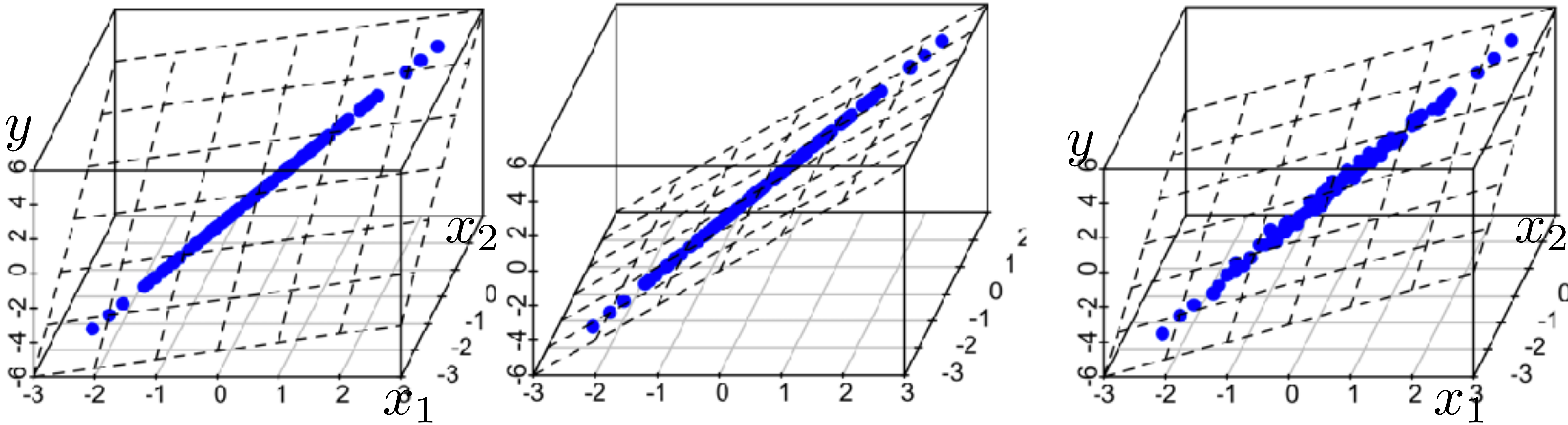
- Sometimes there isn't a unique best hyperplane



- Sometimes there's technically a unique best hyperplane, but just because of noise
- Feature encodings (cf. one-hot)

What can go wrong in practice?

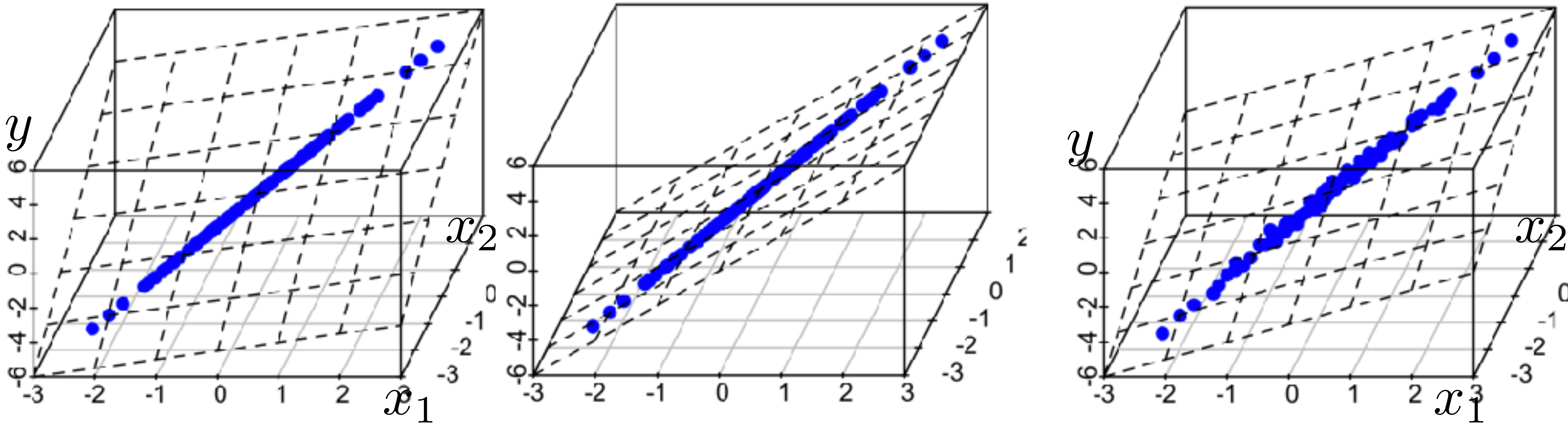
- Sometimes there isn't a unique best hyperplane



- Sometimes there's technically a unique best hyperplane, but just because of noise
- Feature encodings (cf. one-hot);
  real-life features often correlated

What can go wrong in practice?

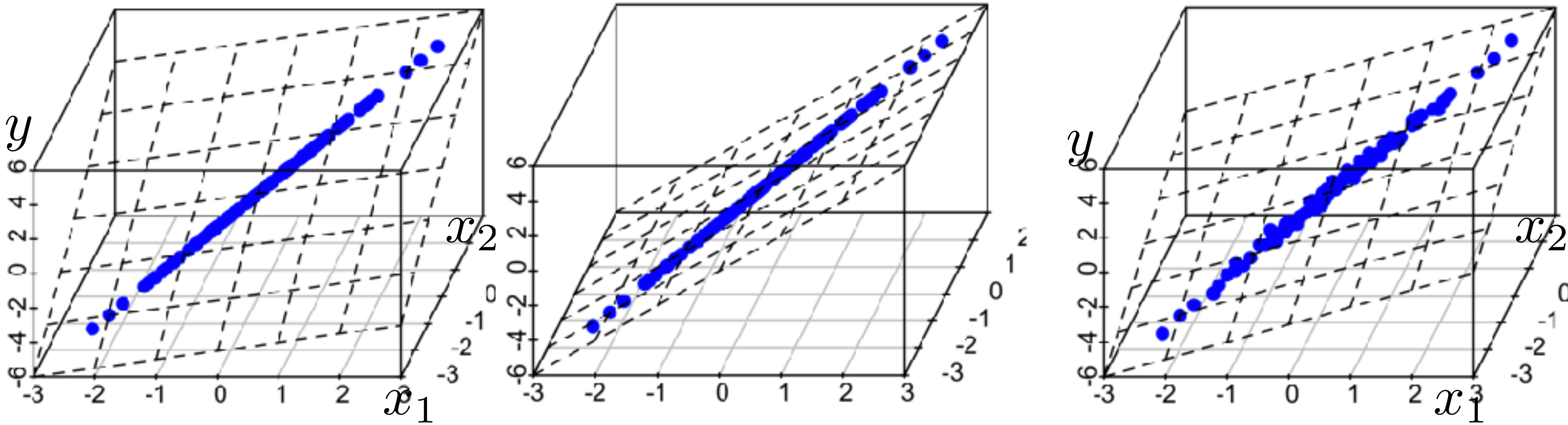
- Sometimes there isn't a unique best hyperplane



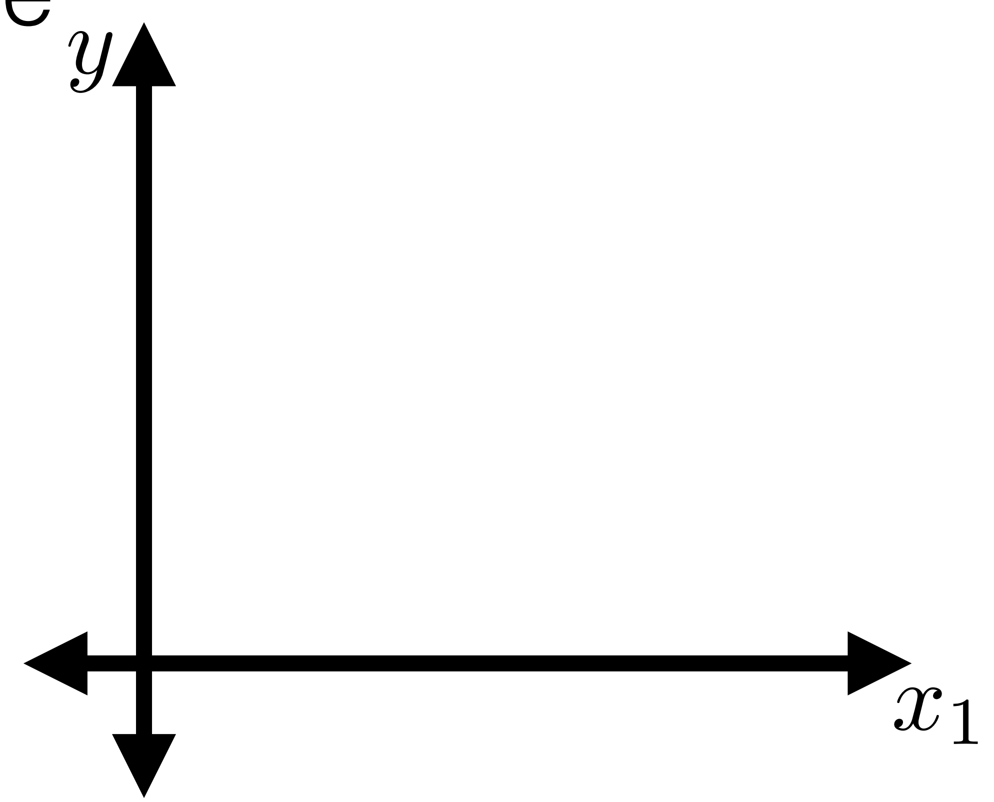
- Sometimes there's technically a unique best hyperplane, but just because of noise
- Feature encodings (cf. one-hot);
  real-life features often correlated;
  many feature dimensions

What can go wrong in practice?

- Sometimes there isn't a unique best hyperplane

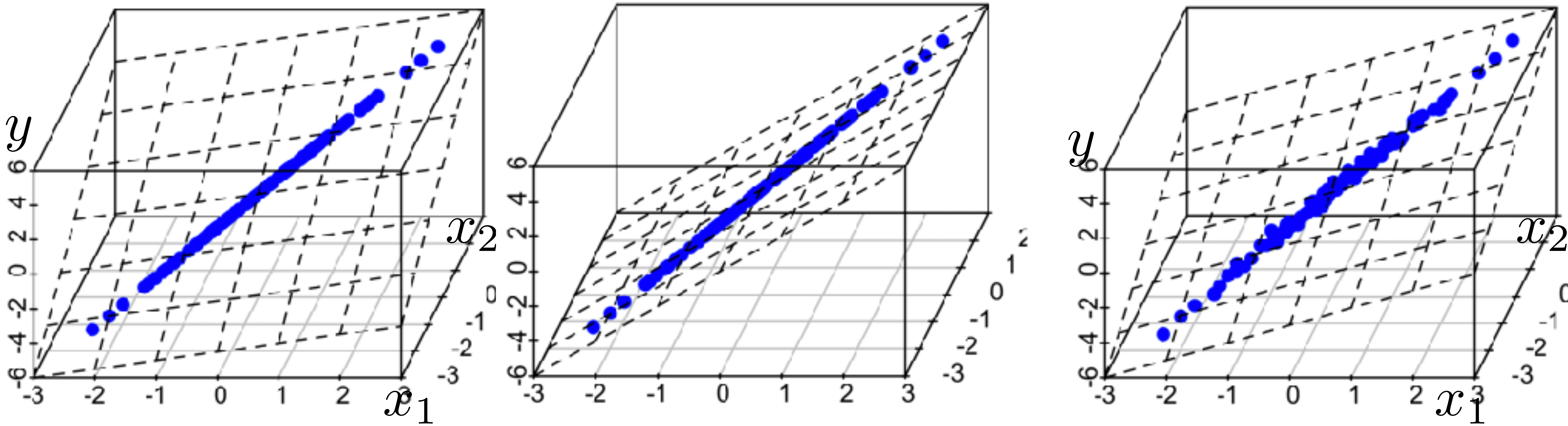


- Sometimes there's technically a unique best hyperplane, but just because of noise
- Feature encodings (cf. one-hot);
  real-life features often correlated;
  many feature dimensions

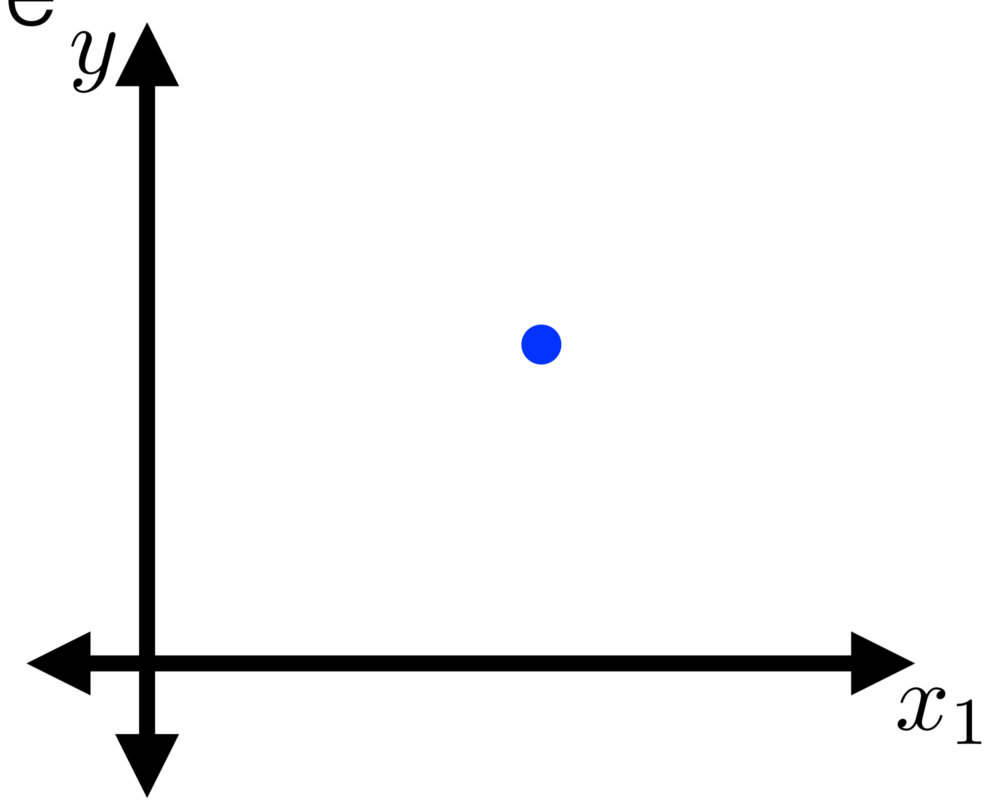


What can go wrong in practice?

- Sometimes there isn't a unique best hyperplane

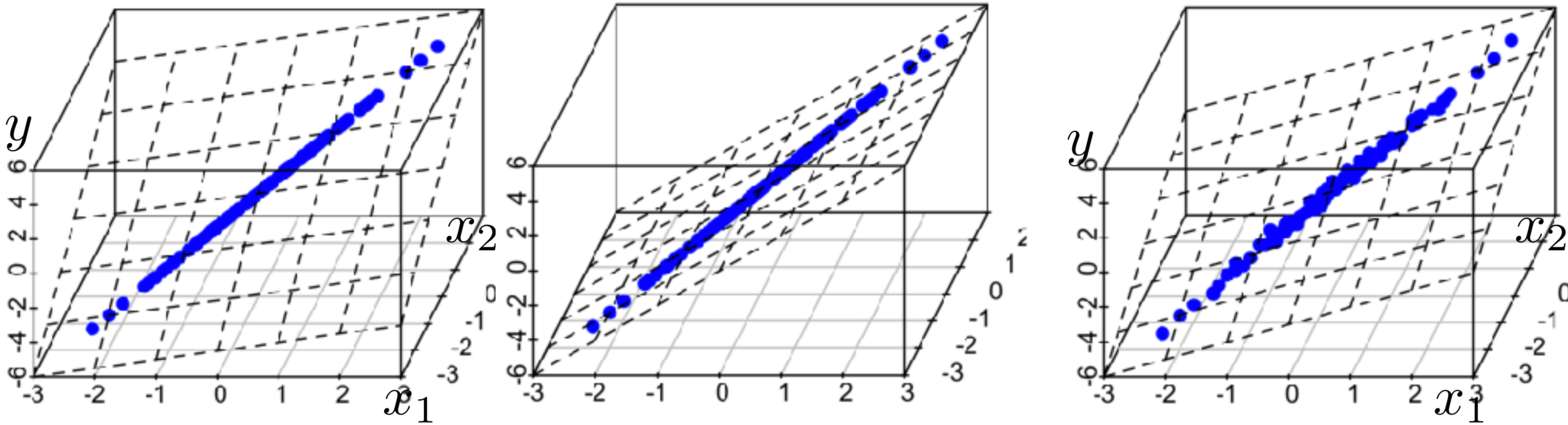


- Sometimes there's technically a unique best hyperplane, but just because of noise
- Feature encodings (cf. one-hot); real-life features often correlated; many feature dimensions

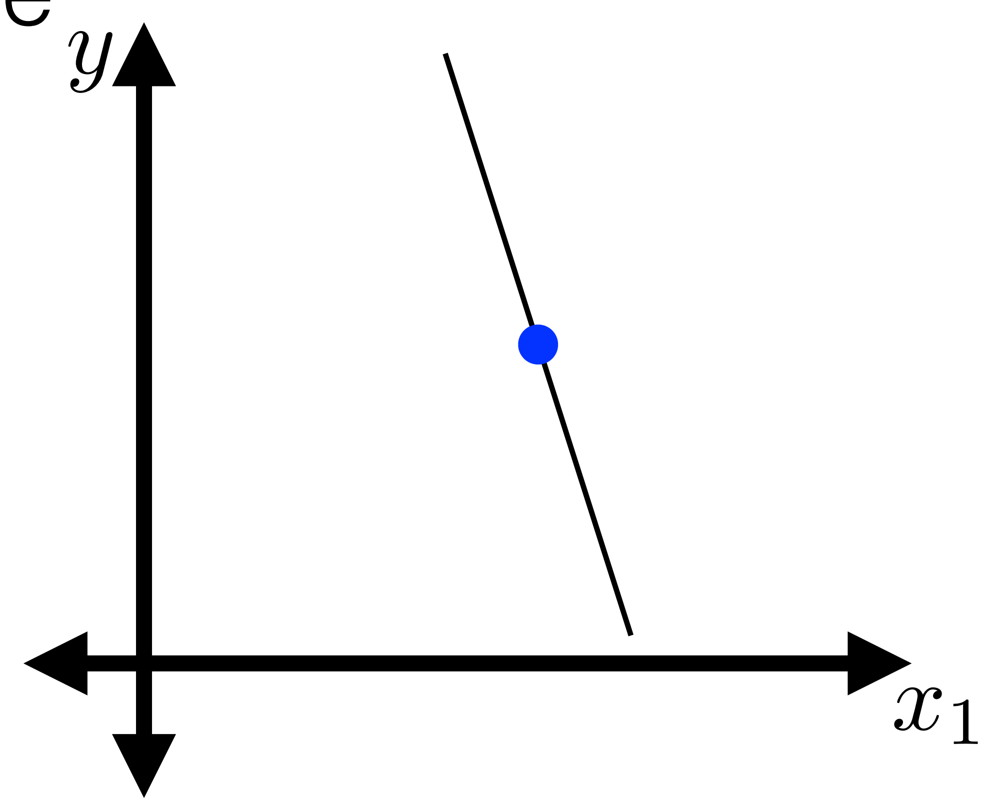


What can go wrong in practice?

- Sometimes there isn't a unique best hyperplane

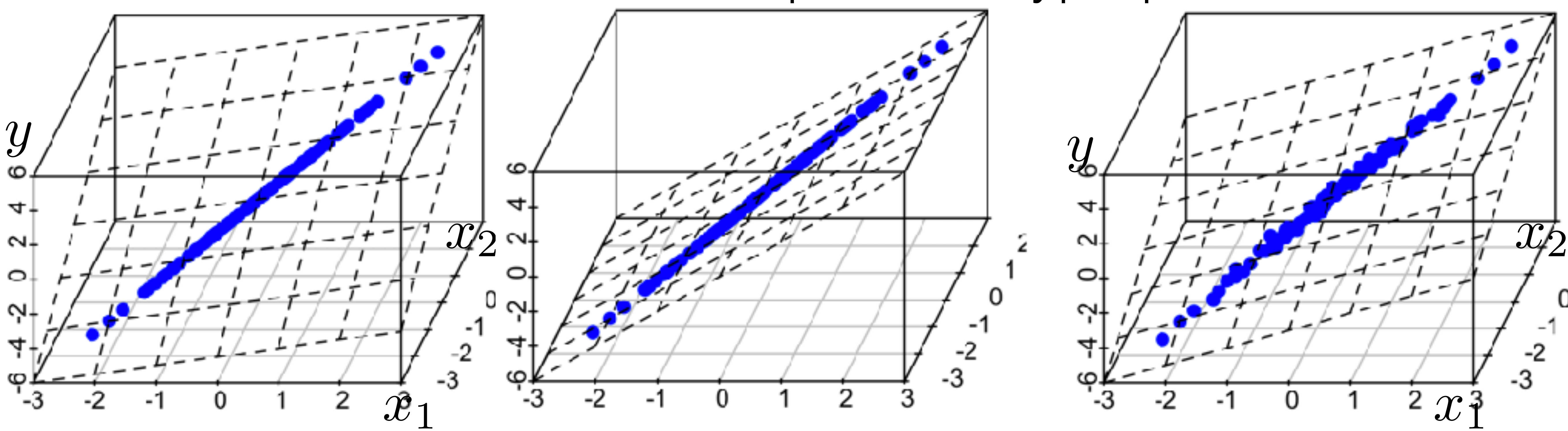


- Sometimes there's technically a unique best hyperplane, but just because of noise
- Feature encodings (cf. one-hot); real-life features often correlated; many feature dimensions

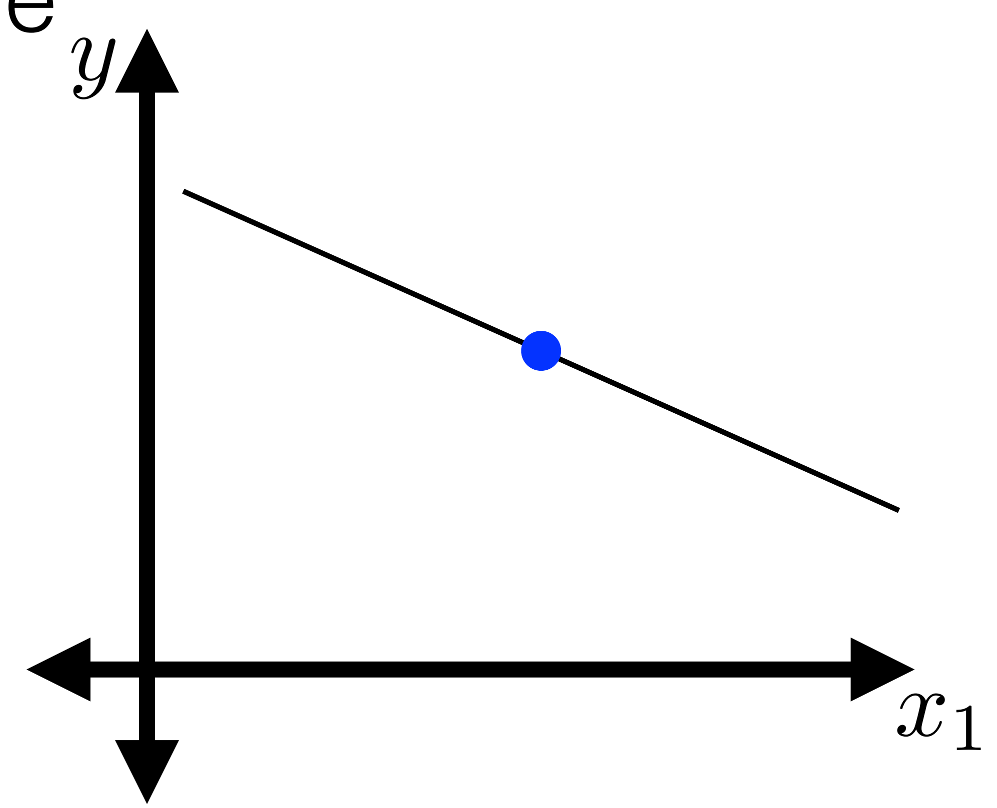


What can go wrong in practice?

- Sometimes there isn't a unique best hyperplane

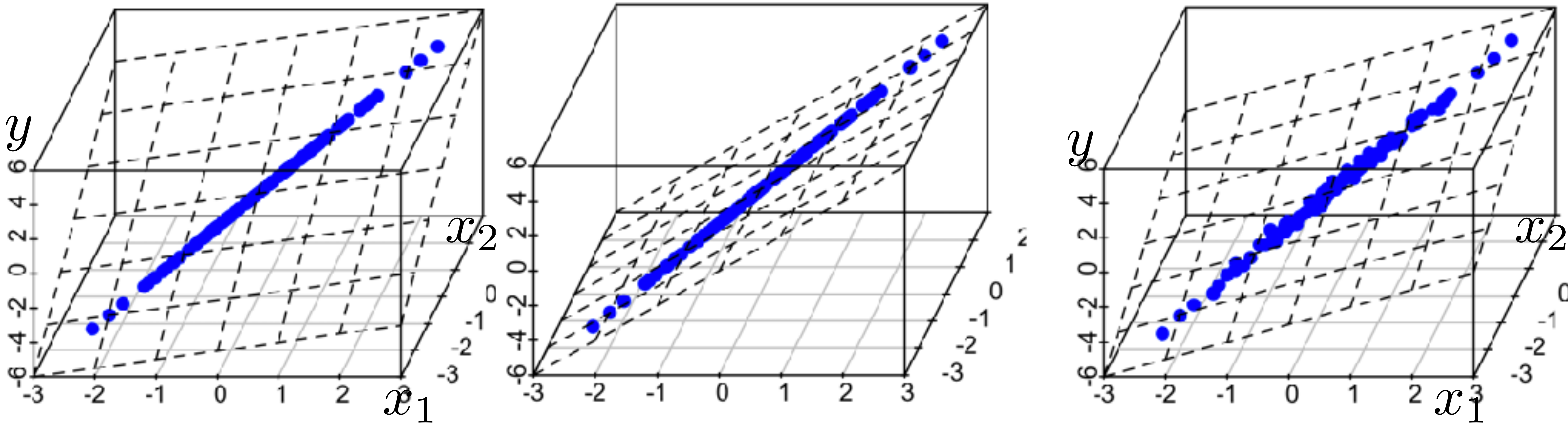


- Sometimes there's technically a unique best hyperplane, but just because of noise
- Feature encodings (cf. one-hot); real-life features often correlated; many feature dimensions

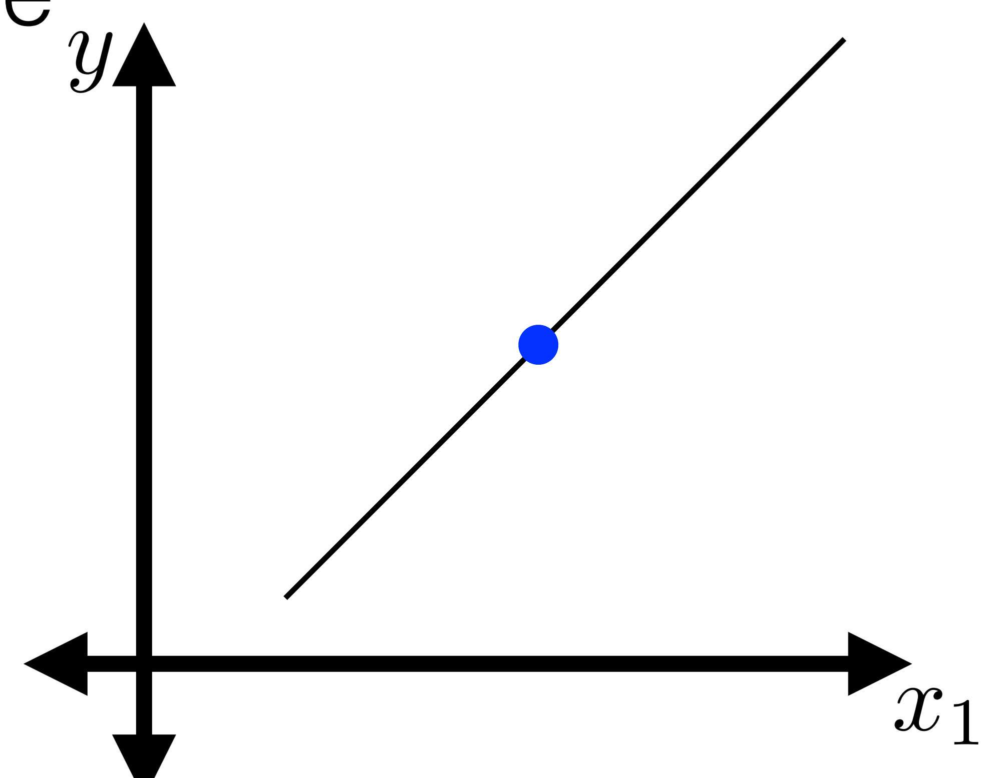


What can go wrong in practice?

- Sometimes there isn't a unique best hyperplane

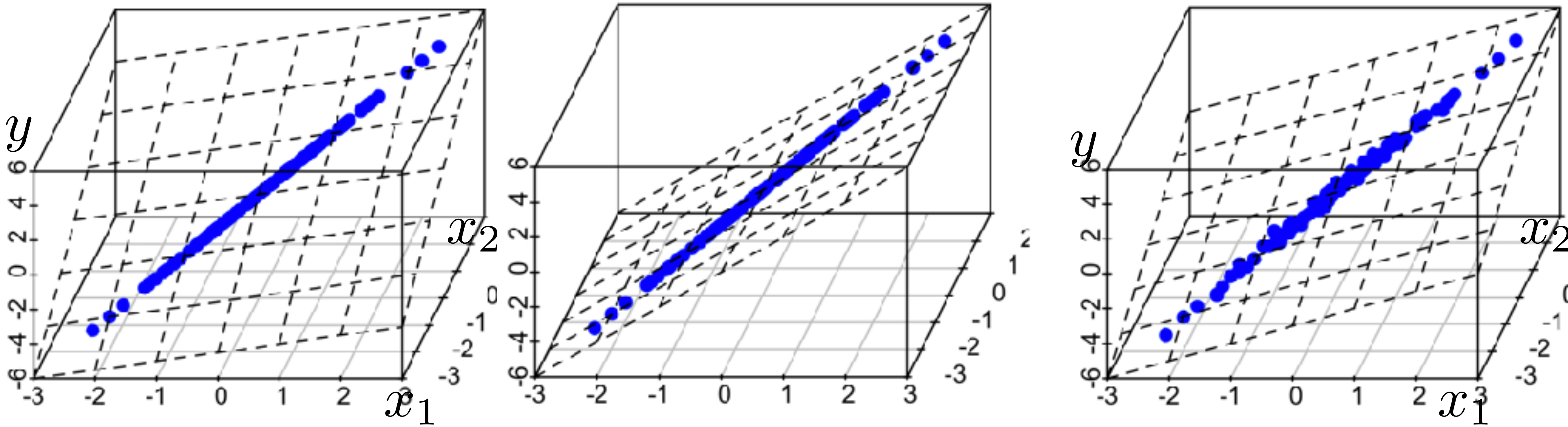


- Sometimes there's technically a unique best hyperplane, but just because of noise
- Feature encodings (cf. one-hot);
real-life features often correlated;
many feature dimensions

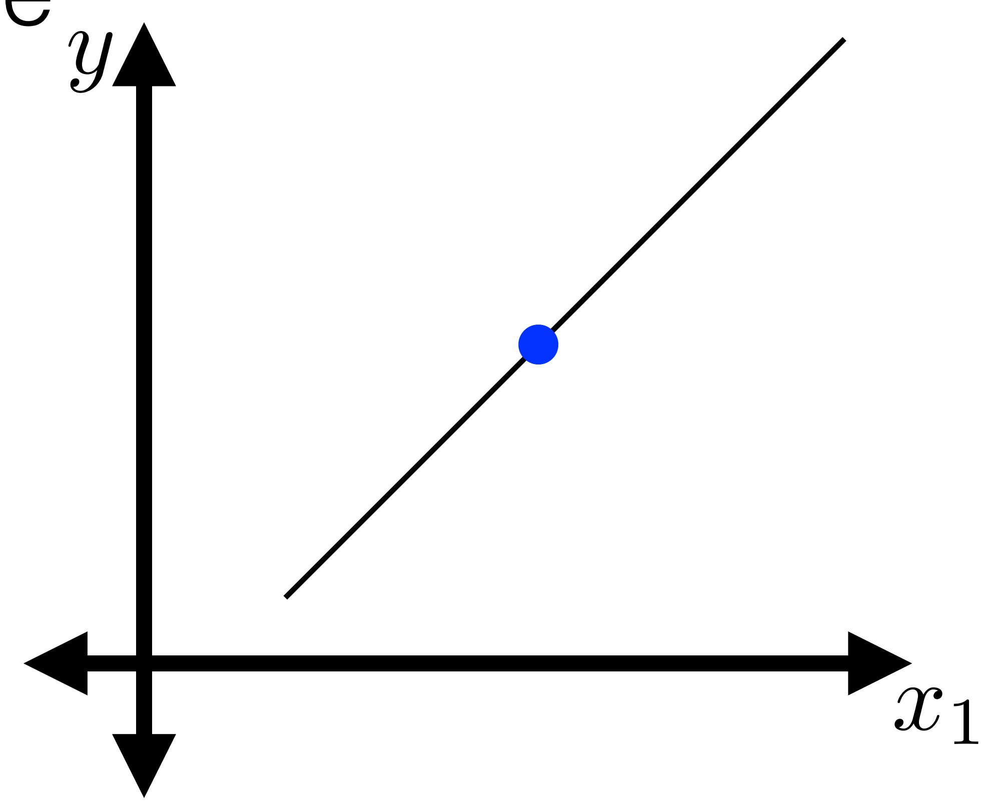


What can go wrong in practice?

- Sometimes there isn't a unique best hyperplane

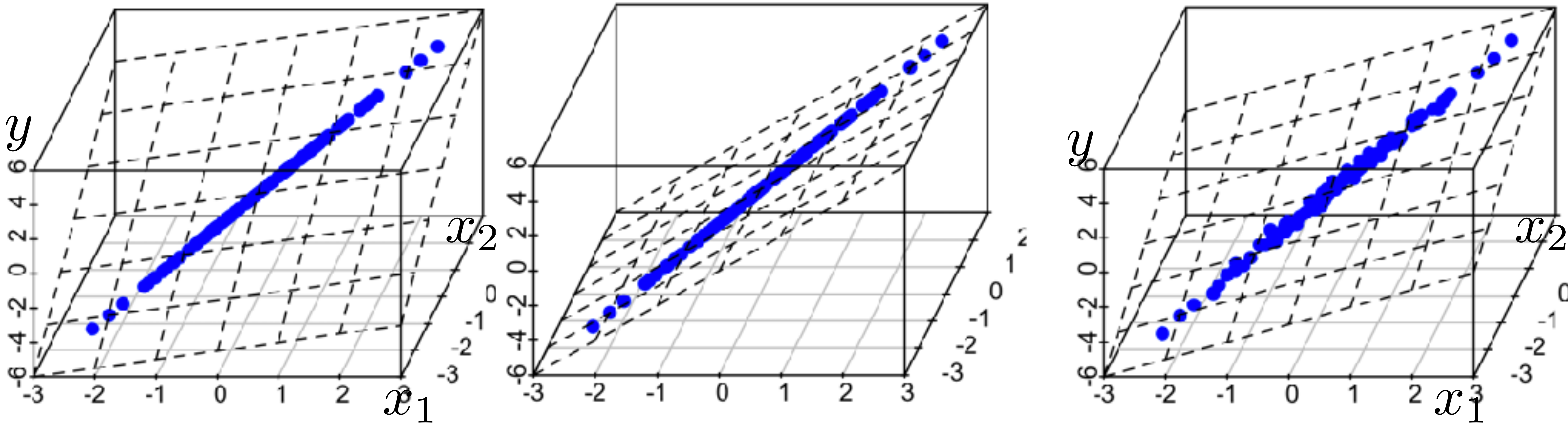


- Sometimes there's technically a unique best hyperplane, but just because of noise
- Feature encodings (cf. one-hot); real-life features often correlated; many feature dimensions
- How to choose among planes?

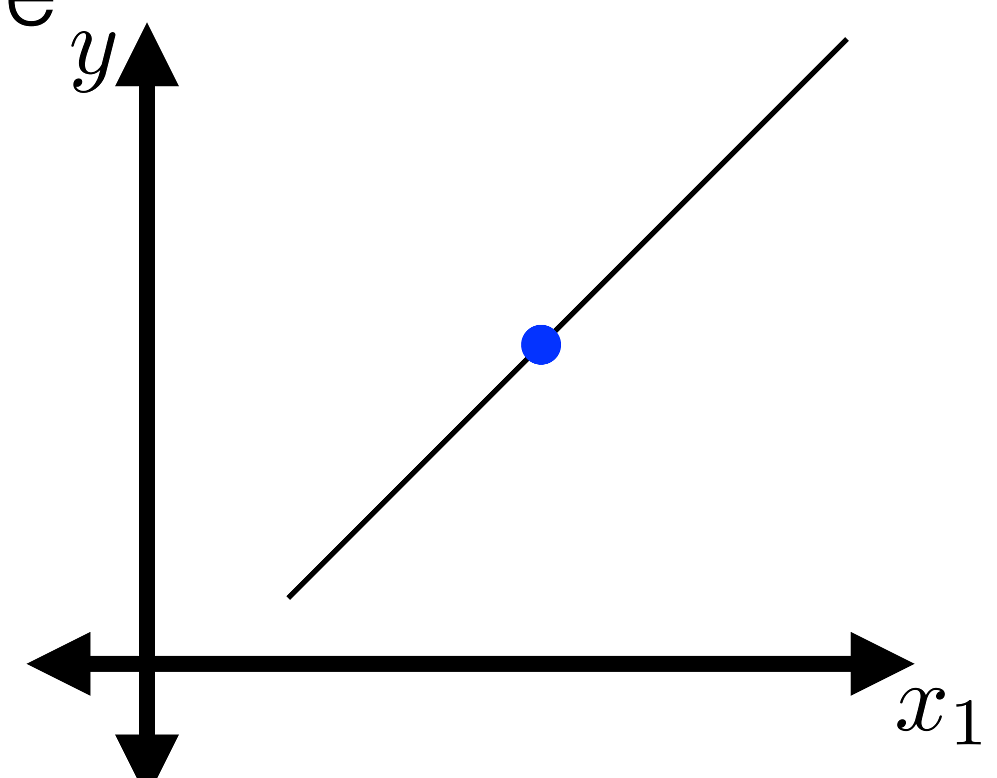


What can go wrong in practice?

- Sometimes there isn't a unique best hyperplane



- Sometimes there's technically a unique best hyperplane, but just because of noise
- Feature encodings (cf. one-hot); real-life features often correlated; many feature dimensions
- How to choose among planes? Preference for θ components being near zero



Regularizing linear regression

Regularizing linear regression

- Linear regression with square penalty: ridge regression

Regularizing linear regression

- Linear regression with square penalty: ridge regression

$$\frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

Regularizing linear regression

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

Regularizing linear regression

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2$$

Regularizing linear regression

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

Regularizing linear regression

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

What happens if $\lambda < 0$?

Regularizing linear regression

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

- Special case: ridge regression with no offset

What happens if $\lambda < 0$?

Regularizing linear regression

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

- Special case: ridge regression with no offset

$$J_{\text{ridge}}(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y}) + \lambda\|\theta\|^2$$

What happens if $\lambda < 0$?

Regularizing linear regression

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

- Special case: ridge regression with no offset

$$J_{\text{ridge}}(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y}) + \lambda\|\theta\|^2$$

$$= \frac{1}{n}\|\tilde{X}\theta - \tilde{Y}\|^2 + \lambda\|\theta\|^2$$

What happens if $\lambda < 0$?

Regularizing linear regression

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

- Special case: ridge regression with no offset

What happens if $\lambda < 0$?

$$J_{\text{ridge}}(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y}) + \lambda\|\theta\|^2$$

$$= \frac{1}{n}\|\tilde{X}\theta - \tilde{Y}\|^2 + \lambda\|\theta\|^2$$

  - Min at: $\nabla_\theta J_{\text{ridge}}(\theta) = 0$

Regularizing linear regression

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

- Special case: ridge regression with no offset

What happens if $\lambda < 0$?

$$J_{\text{ridge}}(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y}) + \lambda\|\theta\|^2$$

$$= \frac{1}{n}\|\tilde{X}\theta - \tilde{Y}\|^2 + \lambda\|\theta\|^2$$

- Min at: $\nabla_\theta J_{\text{ridge}}(\theta) = 0 \;\Rightarrow \theta = (\tilde{X}^\top\tilde{X} + n\lambda I)^{-1}\tilde{X}^\top\tilde{Y}$

Regularizing linear regression

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

- Special case: ridge regression with no offset

$$J_{\text{ridge}}(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y}) + \lambda\|\theta\|^2$$

$$= \frac{1}{n}\|\tilde{X}\theta - \tilde{Y}\|^2 + \lambda\|\theta\|^2$$

What happens if $\lambda < 0$?

- Min at: $\nabla_\theta J_{\text{ridge}}(\theta) = 0 \;\Rightarrow \theta = (\tilde{X}^\top \tilde{X} + n\lambda I)^{-1}\tilde{X}^\top \tilde{Y}$

Regularizing linear regression

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

- Special case: ridge regression with no offset

What happens if $\lambda < 0$?

$$J_{\text{ridge}}(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y}) + \lambda\|\theta\|^2$$

$$= \frac{1}{n}\|\tilde{X}\theta - \tilde{Y}\|^2 + \lambda\|\theta\|^2$$

- Min at: $\nabla_\theta J_{\text{ridge}}(\theta) = 0 \;\Rightarrow \theta = (\tilde{X}^\top \underbrace{\tilde{X}}_{\text{nxd}} + n\lambda I)^{-1}\tilde{X}^\top\tilde{Y}$

Regularizing linear regression

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

- Special case: ridge regression with no offset

What happens if $\lambda < 0$?

$$J_{\text{ridge}}(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y}) + \lambda\|\theta\|^2$$

$$= \frac{1}{n}\|\tilde{X}\theta - \tilde{Y}\|^2 + \lambda\|\theta\|^2$$

- Min at: $\nabla_\theta J_{\text{ridge}}(\theta) = 0 \;\Rightarrow \theta = (\tilde{X}^\top\tilde{X} + n\lambda I)^{-1}\tilde{X}^\top\tilde{Y}$

dxn,nxd

Regularizing linear regression

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

- Special case: ridge regression with no offset

What happens if $\lambda < 0$?

$$J_{\text{ridge}}(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y}) + \lambda\|\theta\|^2$$

$$= \frac{1}{n}\|\tilde{X}\theta - \tilde{Y}\|^2 + \lambda\|\theta\|^2$$

- Min at: $\nabla_\theta J_{\text{ridge}}(\theta) = 0 \;\Rightarrow \theta = (\underbrace{\tilde{X}^\top \tilde{X}}_{\text{dxn,nxd}} + n\lambda \underbrace{I}_{\text{dxd}})^{-1}\tilde{X}^\top \tilde{Y}$

Regularizing linear regression

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

- Special case: ridge regression with no offset

What happens if $\lambda < 0$?

$$J_{\text{ridge}}(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y}) + \lambda\|\theta\|^2$$

$$= \frac{1}{n}\|\tilde{X}\theta - \tilde{Y}\|^2 + \lambda\|\theta\|^2$$

- Min at: $\nabla_\theta J_{\text{ridge}}(\theta) = 0 \;\Rightarrow \theta = (\underbrace{\tilde{X}^\top\tilde{X}}_{\text{dxn,nxd}} + \underbrace{n\lambda I}_{\text{dxd}})^{-1}\tilde{X}^\top\tilde{Y}$

Regularizing linear regression

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

- Special case: ridge regression with no offset

What happens if $\lambda < 0$?

$$J_{\text{ridge}}(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y}) + \lambda\|\theta\|^2$$

$$= \frac{1}{n}\|\tilde{X}\theta - \tilde{Y}\|^2 + \lambda\|\theta\|^2$$

- Min at: $\nabla_\theta J_{\text{ridge}}(\theta) = 0 \;\Rightarrow \theta = (\tilde{X}^\top\tilde{X} + n\lambda I)^{-1}\tilde{X}^\top\tilde{Y}$

  dxn,nxd dxd

  - Matrix of second derivatives: $\tilde{X}^\top\tilde{X} + n\lambda I$ (always “curves up” & invertible when $\lambda > 0$)

Regularizing linear regression

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

- Special case: ridge regression with no offset

$$J_{\text{ridge}}(\theta) = \frac{1}{n}(\tilde{X}\theta - \tilde{Y})^\top(\tilde{X}\theta - \tilde{Y}) + \lambda\|\theta\|^2$$

$$= \frac{1}{n}\|\tilde{X}\theta - \tilde{Y}\|^2 + \lambda\|\theta\|^2$$

What happens if $\lambda < 0$?

  - Min at: $\nabla_\theta J_{\text{ridge}}(\theta) = 0 \;\Rightarrow \theta = (\underbrace{\tilde{X}^\top \tilde{X}}_{\text{dxn,nxd}} + \underbrace{n\lambda I}_{\text{dxd}})^{-1}\tilde{X}^\top\tilde{Y}$
    - Matrix of second derivatives: $\tilde{X}^\top\tilde{X} + n\lambda I$ (always "curves up" & invertible when $\lambda > 0$)
- Can also solve for minimizing parameters in case with offset; just a bit more math

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n} \sum_{i=1}^{n} (\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda \|\theta\|^2 \quad (\lambda > 0)$$

Some notes on features

- Linear regression with square penalty: ridge regression

$$J_{\mathrm{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

Some notes on features

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

- Assumption: features on same scale (cf. standardization)

Some notes on features

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

- Assumption: features on same scale (cf. standardization)
- Featurization still matters! (ridge or "ordinary" regression)

Some notes on features

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

- Assumption: features on same scale (cf. standardization)
- Featurization still matters! (ridge or "ordinary" regression)


The New York Times

A Bitter Election. Accusations of Fraud. And Now Second Thoughts.

A close look at Bolivian election data suggests an initial analysis by the O.A.S. that raised questions of vote-rigging — and helped force out a president — was flawed.

[https://www.nytimes.com/2020/06/07/world/americas/bolivia-election-evo-morales.html/]

Some notes on features

- Linear regression with square penalty: ridge regression

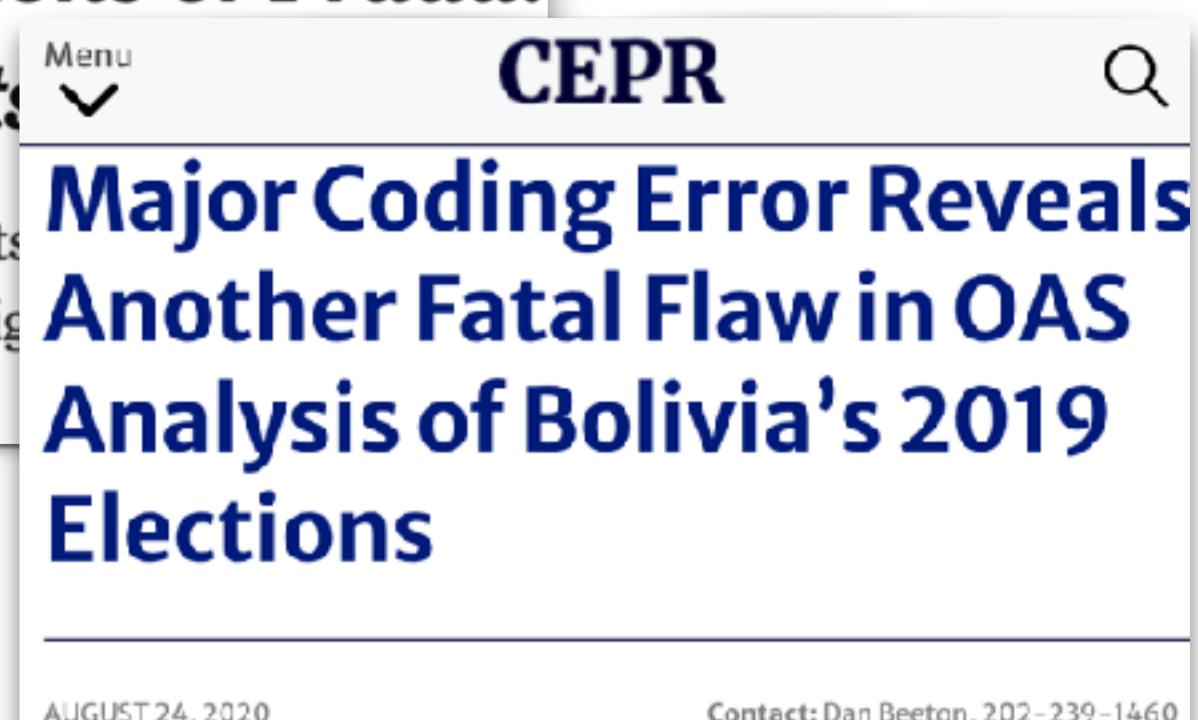
$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

- Assumption: features on same scale (cf. standardization)
- Featurization still matters! (ridge or “ordinary” regression)


The New York Times

A Bitter Election. Accusations of Fraud. And Now Second Thought

A close look at Bolivian election data suggest by the O.A.S. that raised questions of vote-ri force out a president — was flawed.


Menu

CEPR

Major Coding Error Reveals Another Fatal Flaw in OAS Analysis of Bolivia’s 2019 Elections

AUGUST 24, 2020

Contact: Dan Beeton, 202-239-1460

[https://cepr.net/press-release/major-coding-error-reveals-another-fatal-flaw-in-oas-analysis-of-bolivias-2019-elections/]

Some notes on features

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

- Assumption: features on same scale (cf. standardization)
- Featurization still matters! (ridge or "ordinary" regression)

The New York Times

A Bitter Election. Accusations of Fraud. And Now Second Thought

A close look at Bolivian election data suggest by the O.A.S. that raised questions of vote-rig force out a president — was flawed.

Menu

CEPR

Major Coding Error Reveals Another Fatal Flaw in OAS Analysis of Bolivia's 2019 Elections

AUGUST 24, 2020

"time stamps were sorted alphanumerically, instead of chronologically"

[https://cepr.net/press-release/major-coding-error-reveals-another-fatal-flaw-in-oas-analysis-of-bolivias-2019-elections/]

Some notes on features

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda\|\theta\|^2 \quad (\lambda > 0)$$

- Assumption: features on same scale (cf. standardization)
- Featurization still matters! (ridge or "ordinary" regression)

The New York Times

A Bitter Election. Accusations of Fraud. And Now Second Thought

A close look at Bolivian election data suggest by the O.A.S. that raised questions of vote-rig force out a president — was flawed.

Menu

CEPR

Major Coding Error Reveals Another Fatal Flaw in OAS Analysis of Bolivia's 2019 Elections

AUGUST 24, 2020

"time stamps were sorted alphanumerically, instead of chronologically"

- Can never take a data set blindly

[https://cepr.net/press-release/major-coding-error-reveals-another-fatal-flaw-in-oas-analysis-of-bolivias-2019-elections/]

Some notes on features

- Linear regression with square penalty: ridge regression

$$J_{\text{ridge}}(\theta, \theta_0) = \frac{1}{n} \sum_{i=1}^{n} (\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2 + \lambda \|\theta\|^2 \quad (\lambda > 0)$$

- Assumption: features on same scale (cf. standardization)
- Featurization still matters! (ridge or "ordinary" regression)

The New York Times

A Bitter Election. Accusations of Fraud. And Now Second Thought

A close look at Bolivian election data suggests by the O.A.S. that raised questions of vote-rig force out a president — was flawed.

Menu

CEPR

Major Coding Error Reveals Another Fatal Flaw in OAS Analysis of Bolivia's 2019 Elections

AUGUST 24, 2020

"time stamps were sorted alphanumerically, instead of chronologically"

- Can never take a data set blindly
- Share code/data

[https://cepr.net/press-release/major-coding-error-reveals-another-fatal-flaw-in-oas-analysis-of-bolivias-2019-elections/]

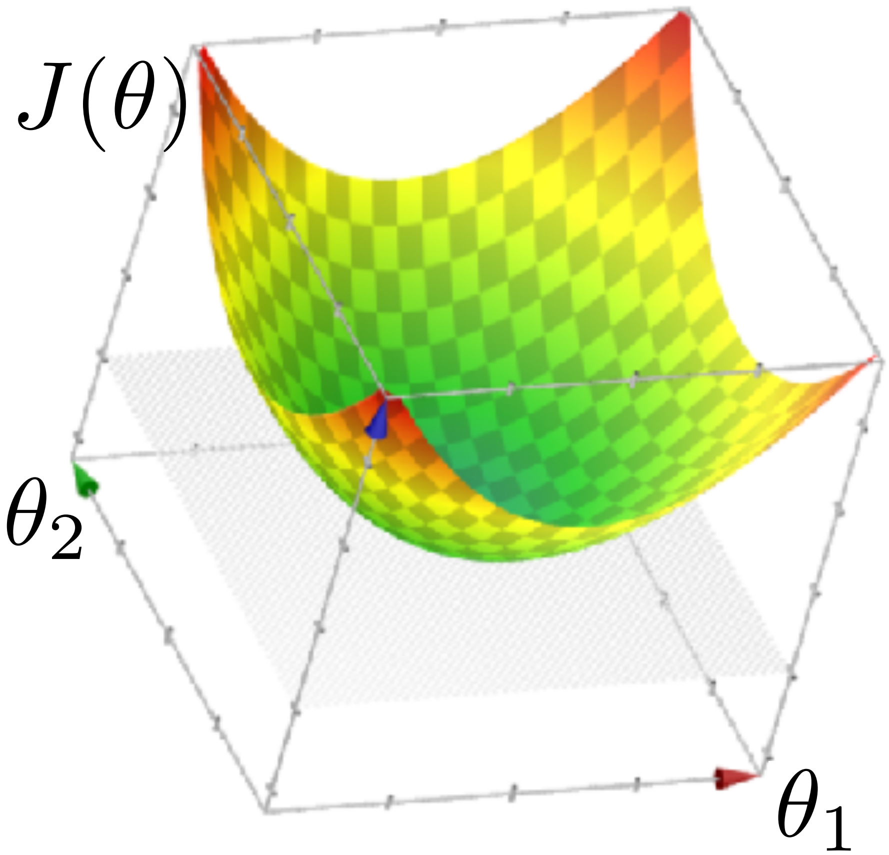
Optimizing linear regression

Optimizing linear regression

- Gradient descent

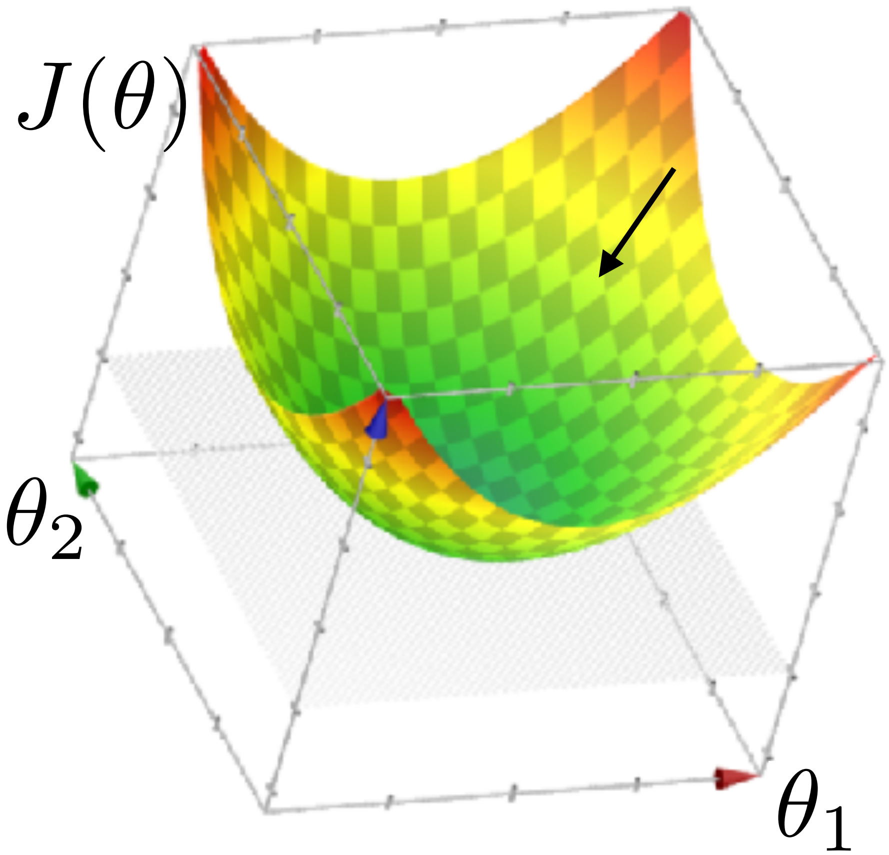
Optimizing linear regression

- Gradient descent



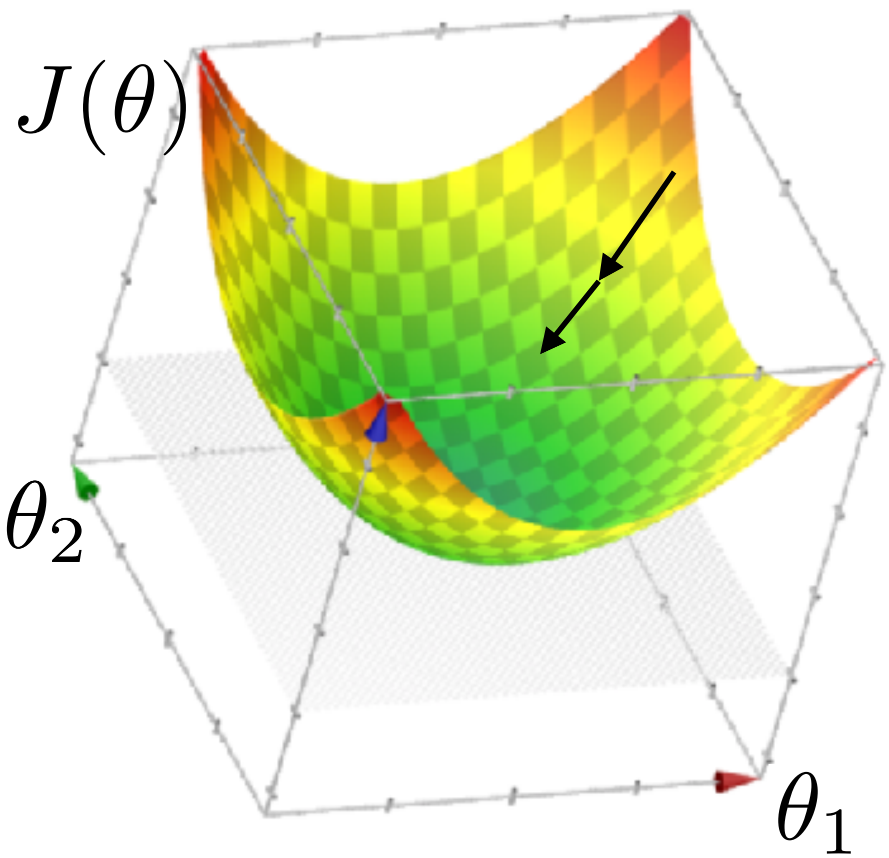
Optimizing linear regression

- Gradient descent



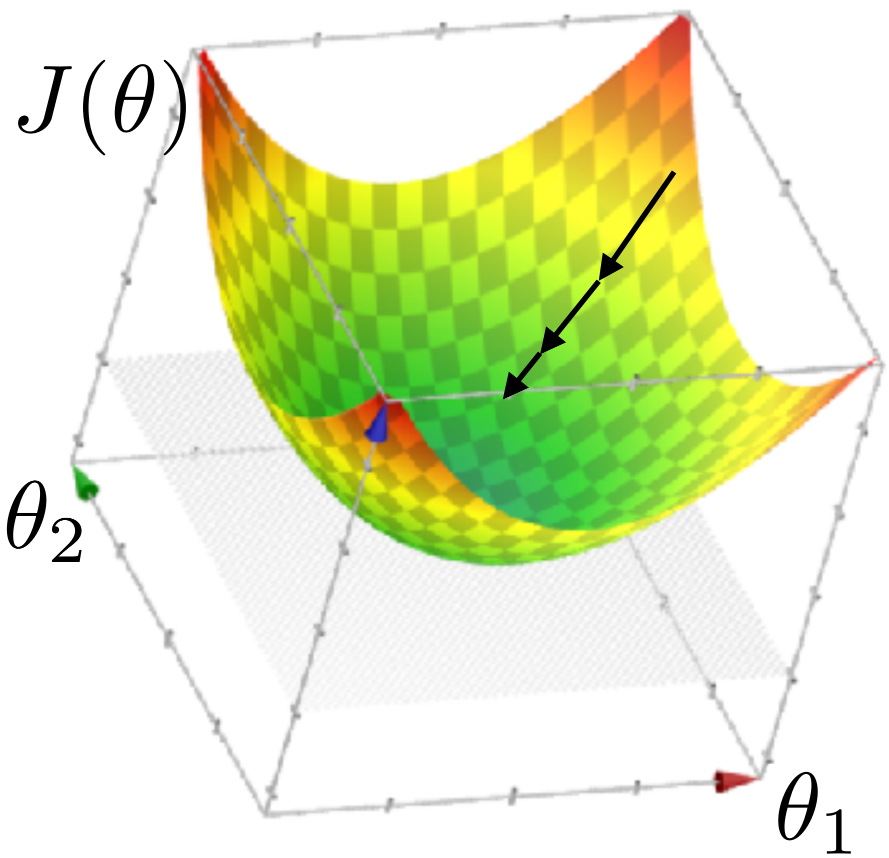
Optimizing linear regression

- Gradient descent



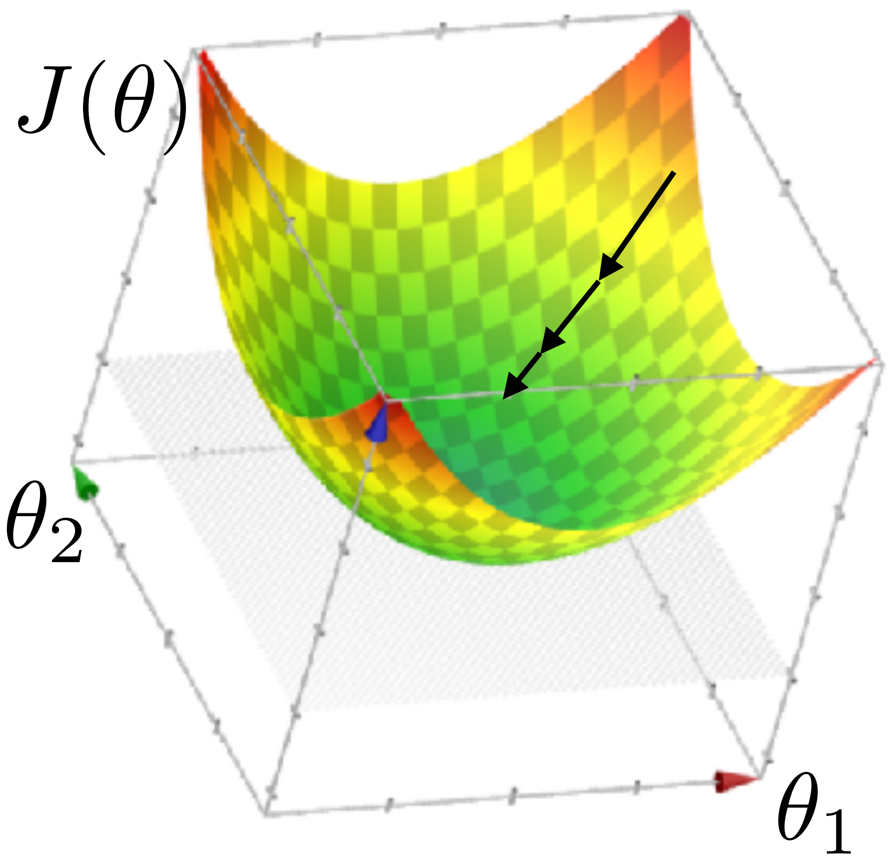
Optimizing linear regression

- Gradient descent



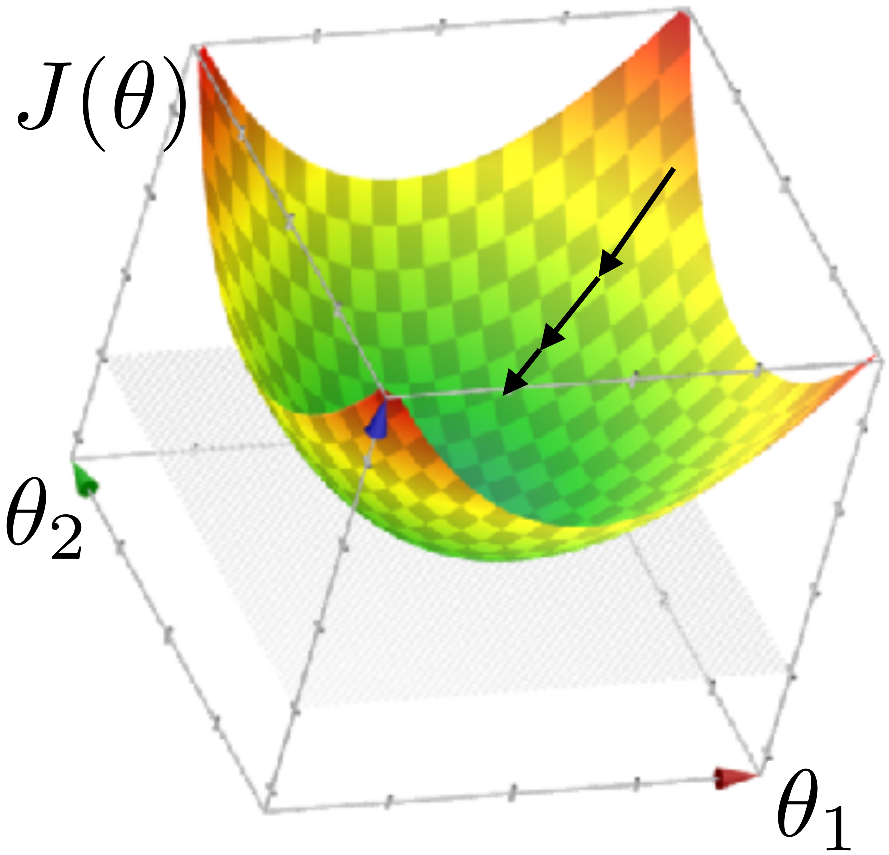
Optimizing linear regression

- Gradient descent



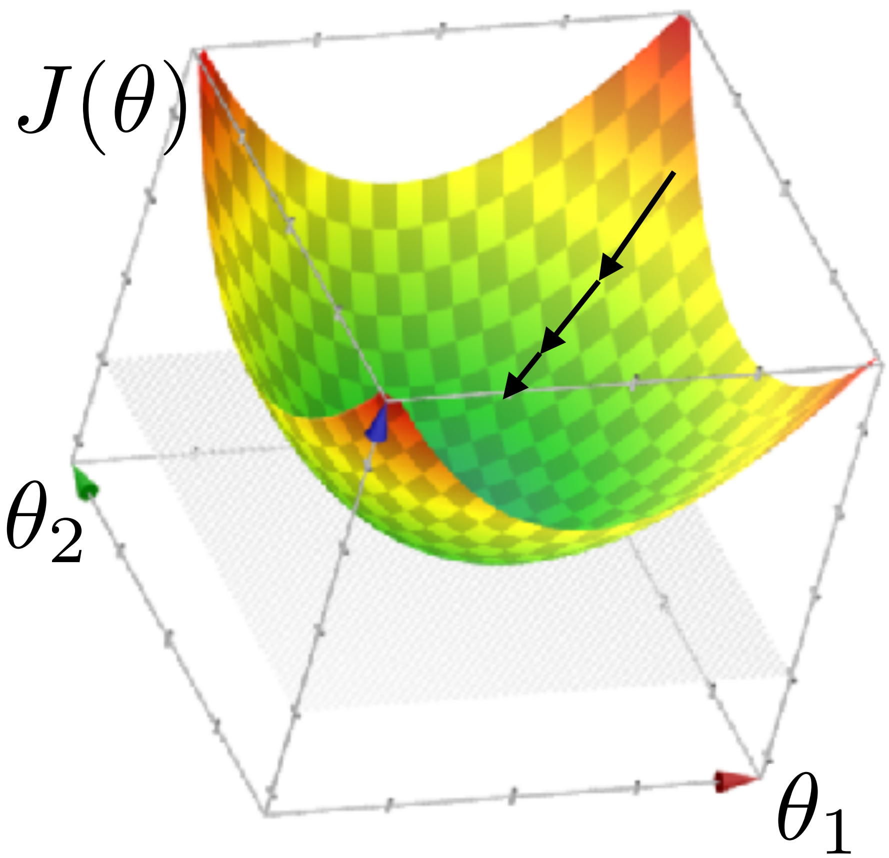
Optimizing linear regression

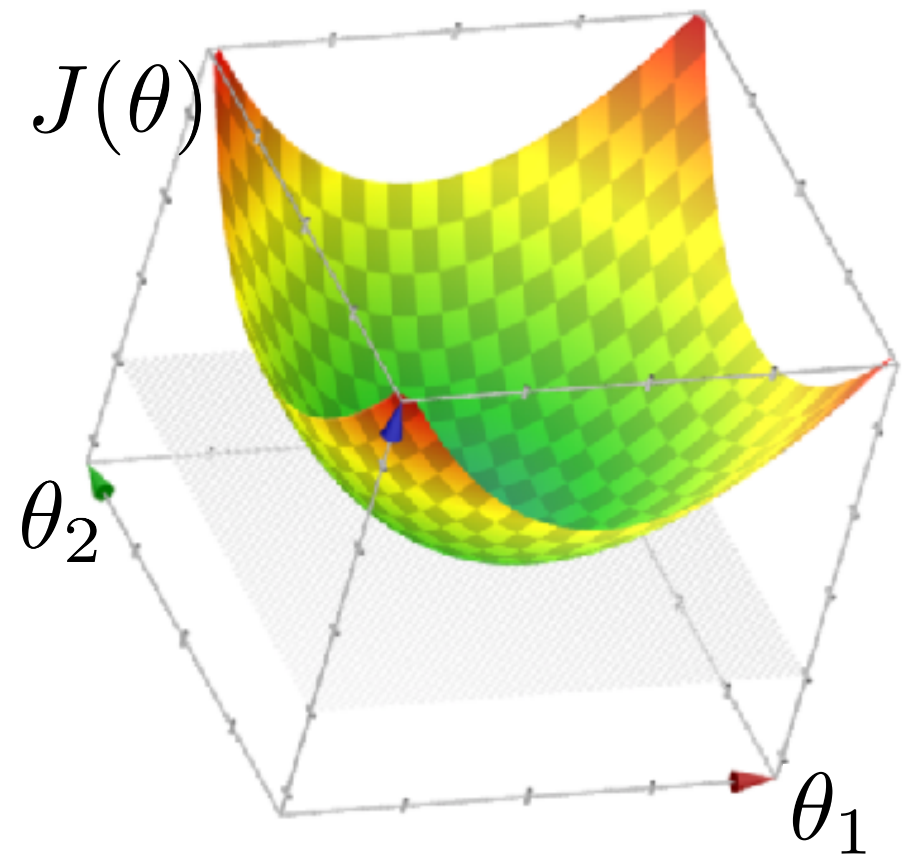
- Gradient descent vs. analytical/closed-form/direct solution



Optimizing linear regression

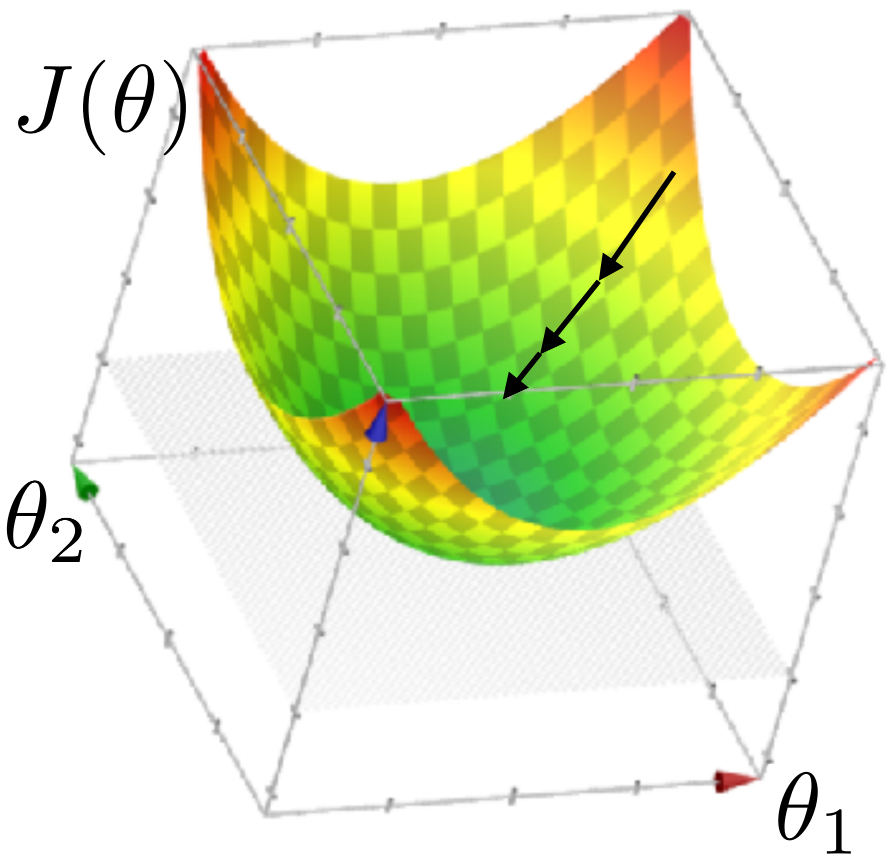
- Gradient descent vs. analytical/closed-form/direct solution

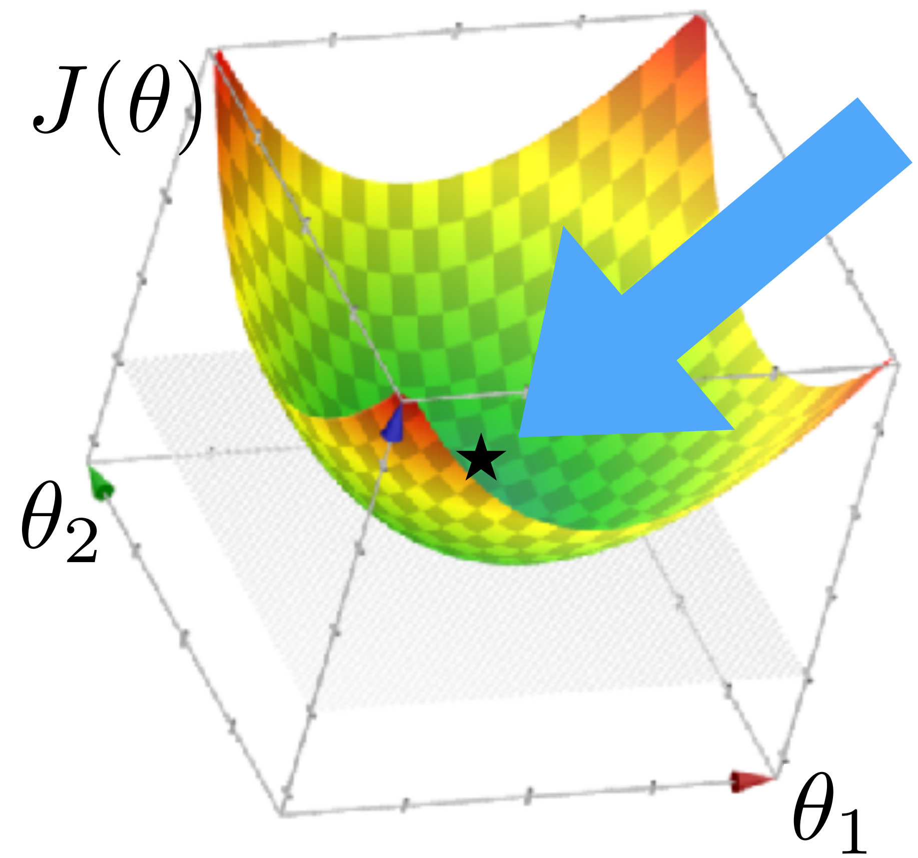




Optimizing linear regression

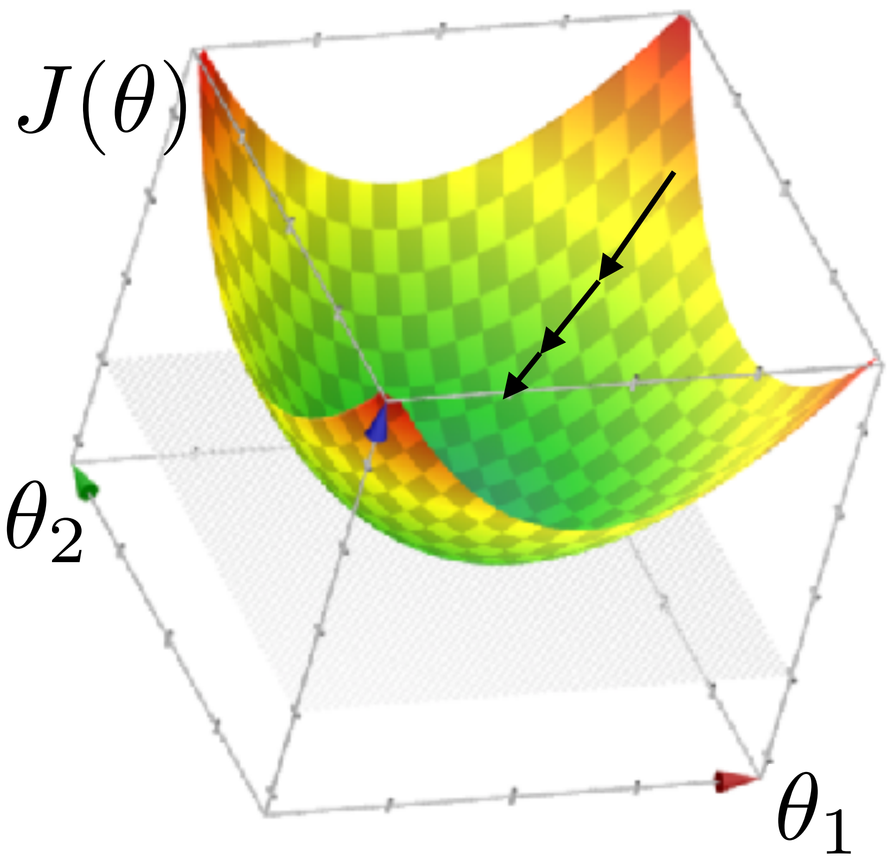
- Gradient descent vs. analytical/closed-form/direct solution

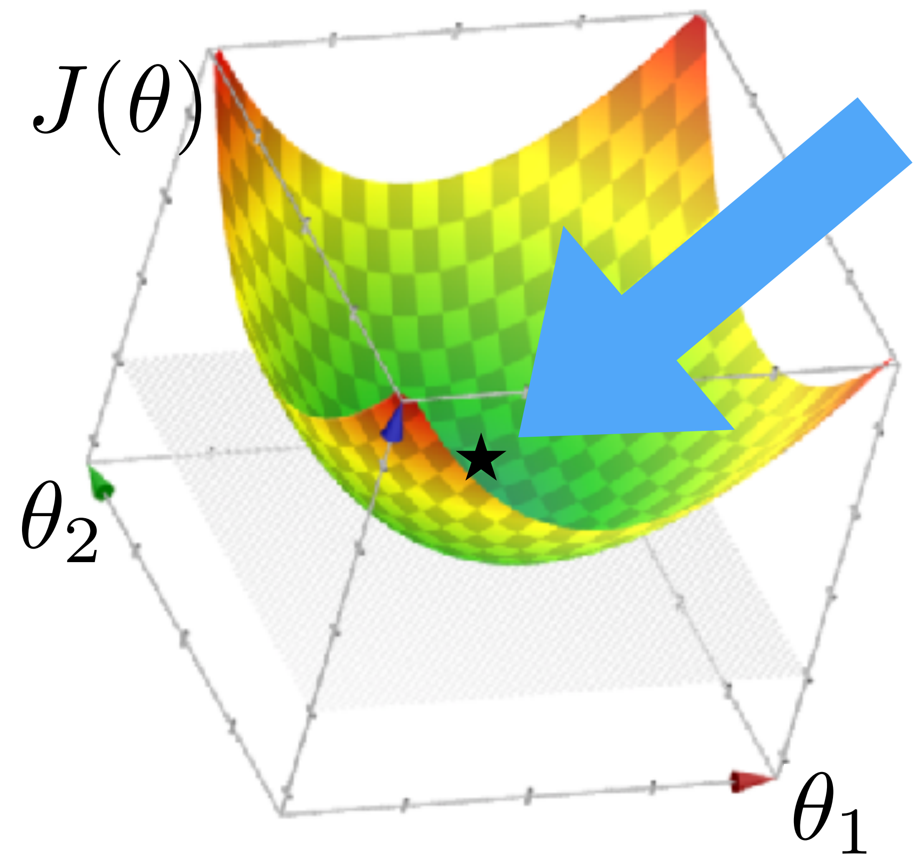




Optimizing linear regression

- Gradient descent vs. analytical/closed-form/direct solution

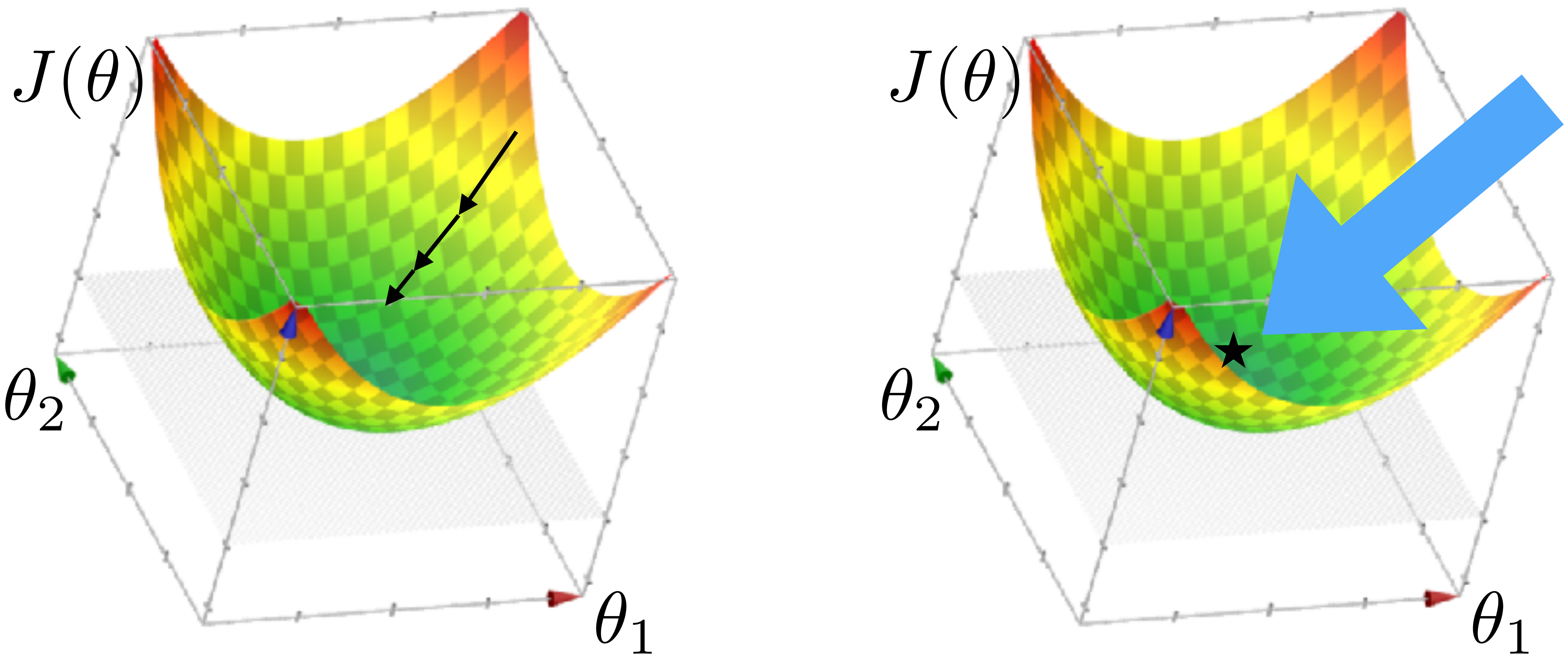




- Accuracy doesn't mean anything without running time

Optimizing linear regression

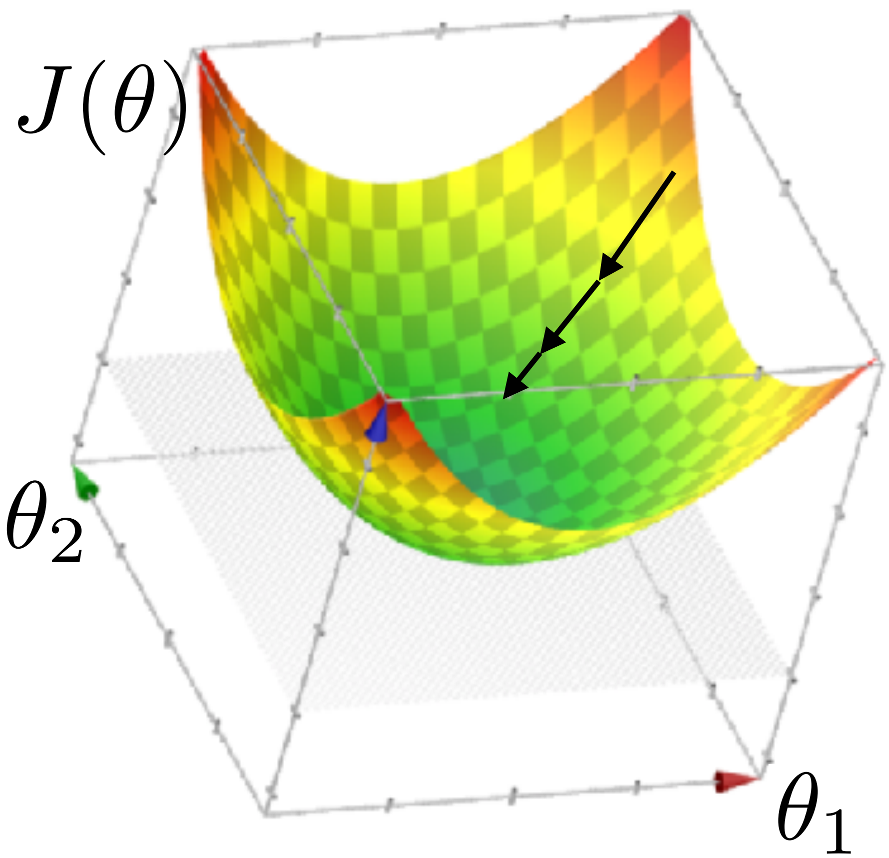
- Gradient descent vs. analytical/closed-form/direct solution

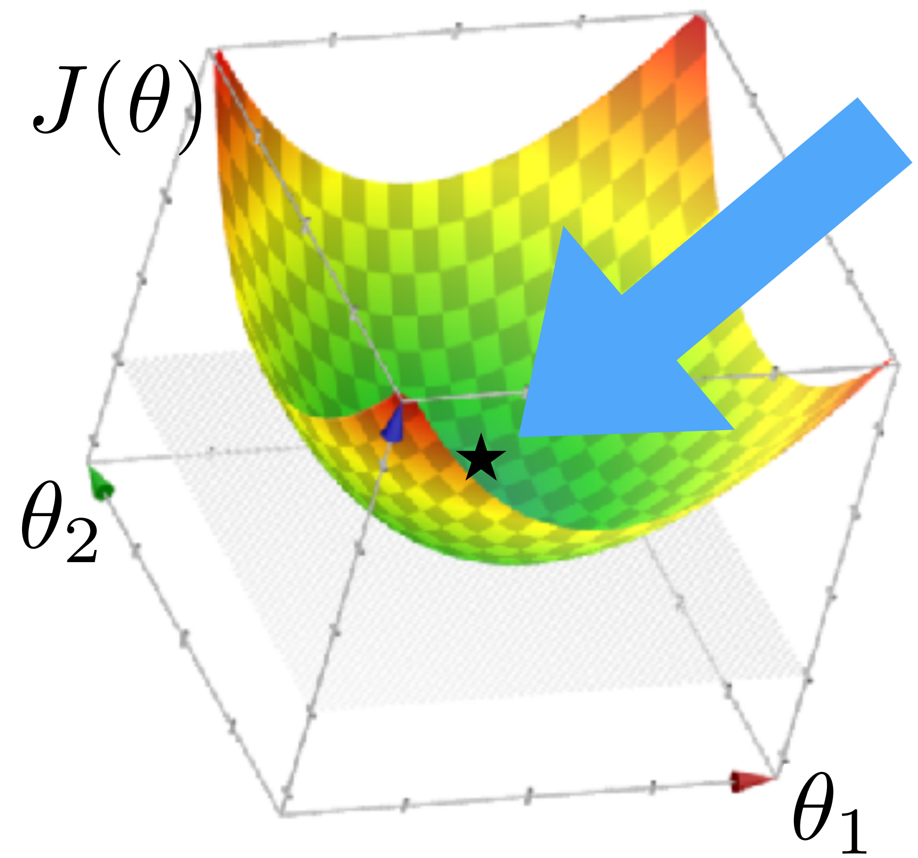


- Accuracy doesn't mean anything without running time
- Running time doesn't mean anything without accuracy

Optimizing linear regression

- Gradient descent vs. analytical/closed-form/direct solution

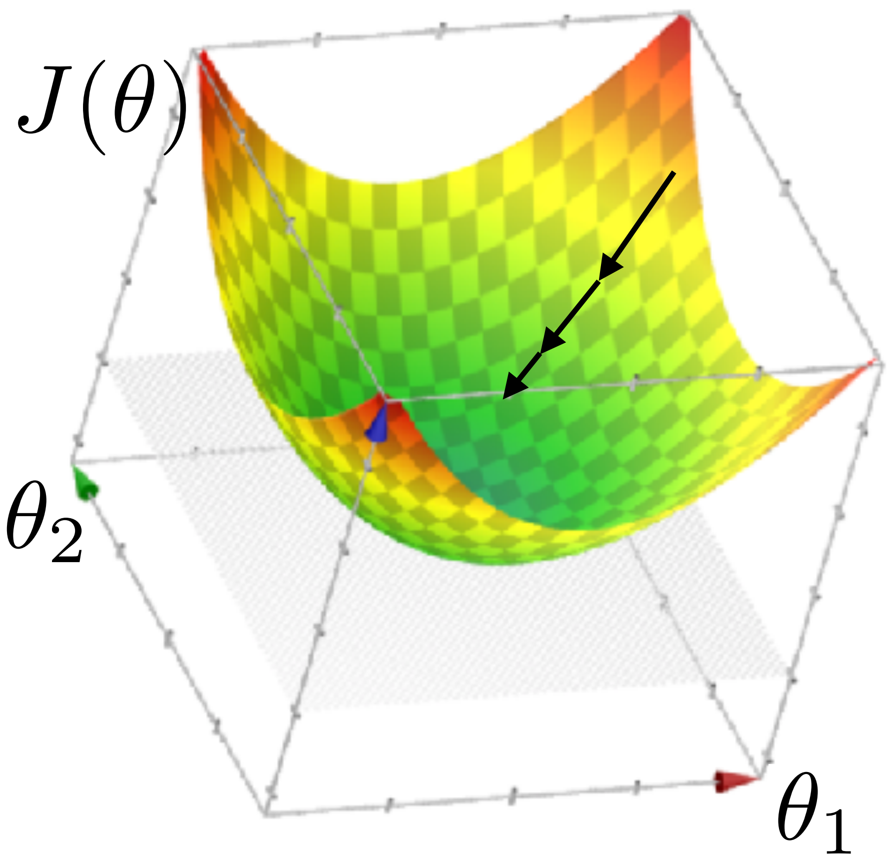


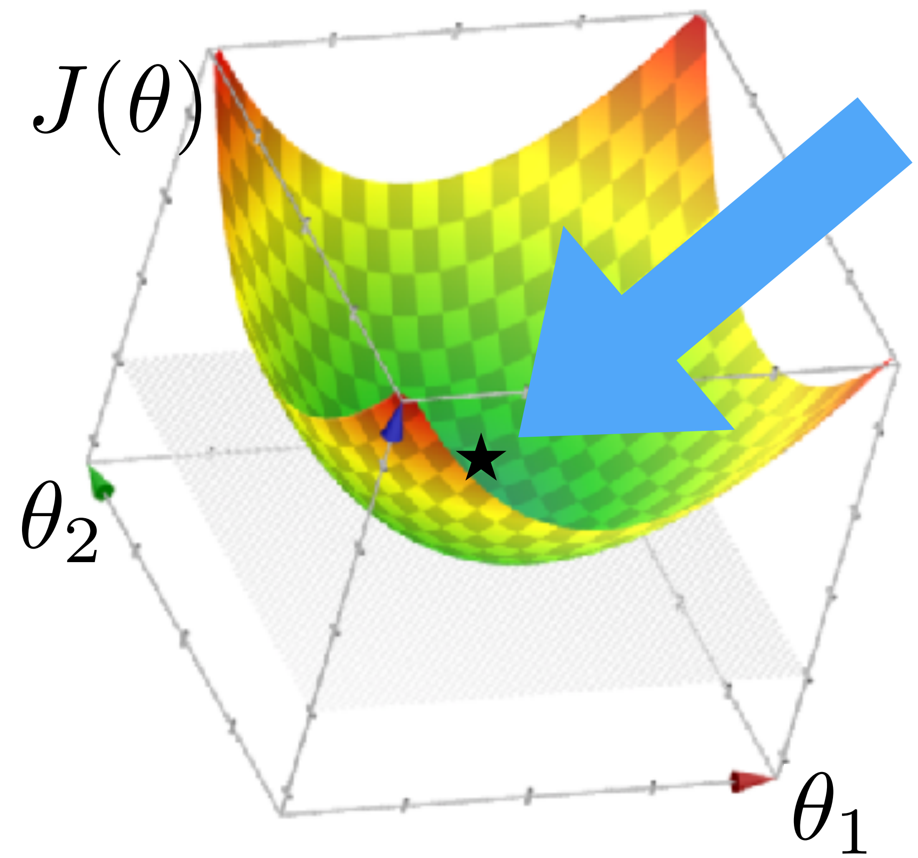


- Accuracy doesn't mean anything without running time
- Running time doesn't mean anything without accuracy
- Need to measure accuracy for the running time we have

Optimizing linear regression

- Gradient descent vs. analytical/closed-form/direct solution



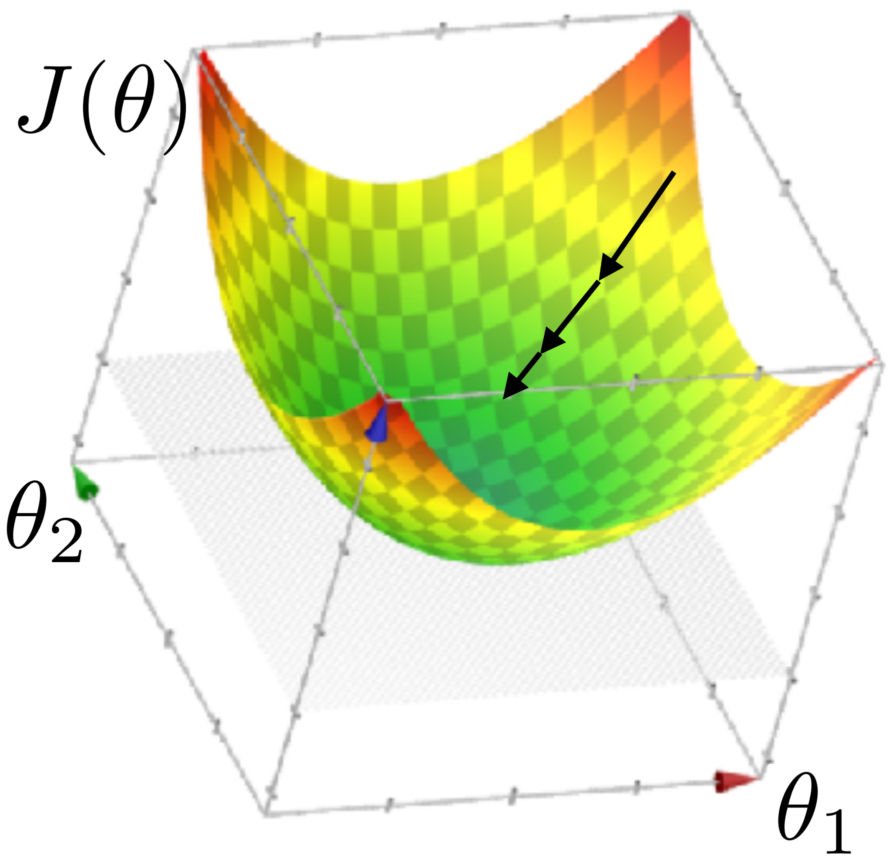


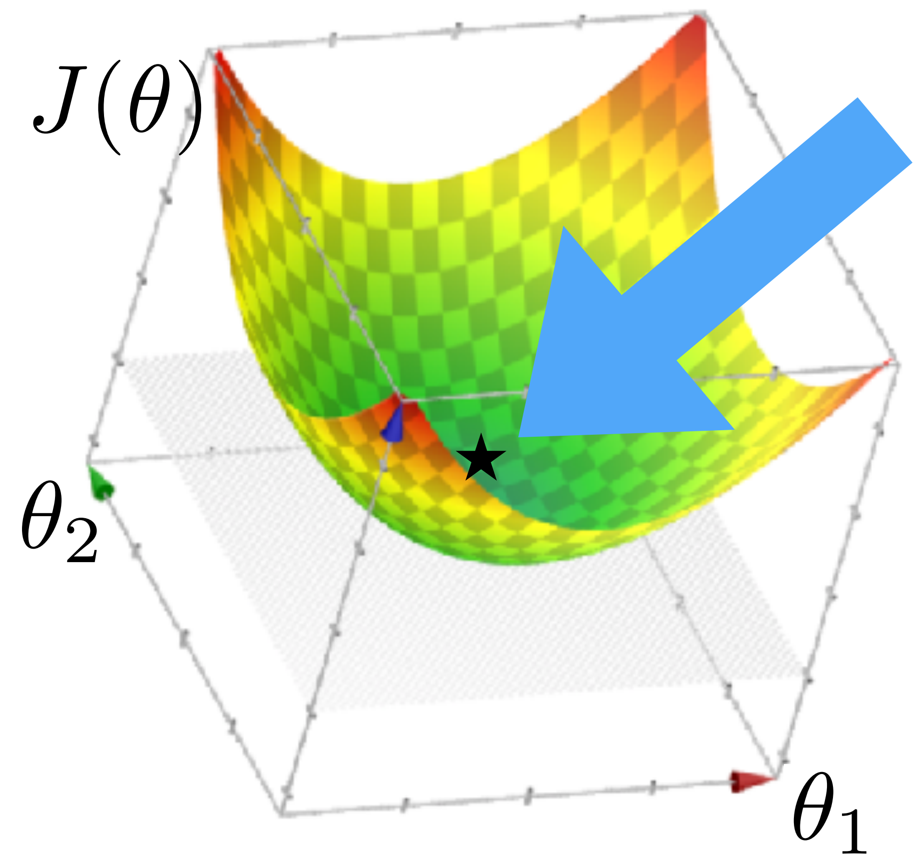
- Accuracy doesn't mean anything without running time
- Running time doesn't mean anything without accuracy
- Need to measure accuracy for the running time we have

$$\theta = (\tilde{X}^\top \tilde{X} + n\lambda I)^{-1} \tilde{X}^\top \tilde{Y}$$

Optimizing linear regression

- Gradient descent vs. analytical/closed-form/direct solution



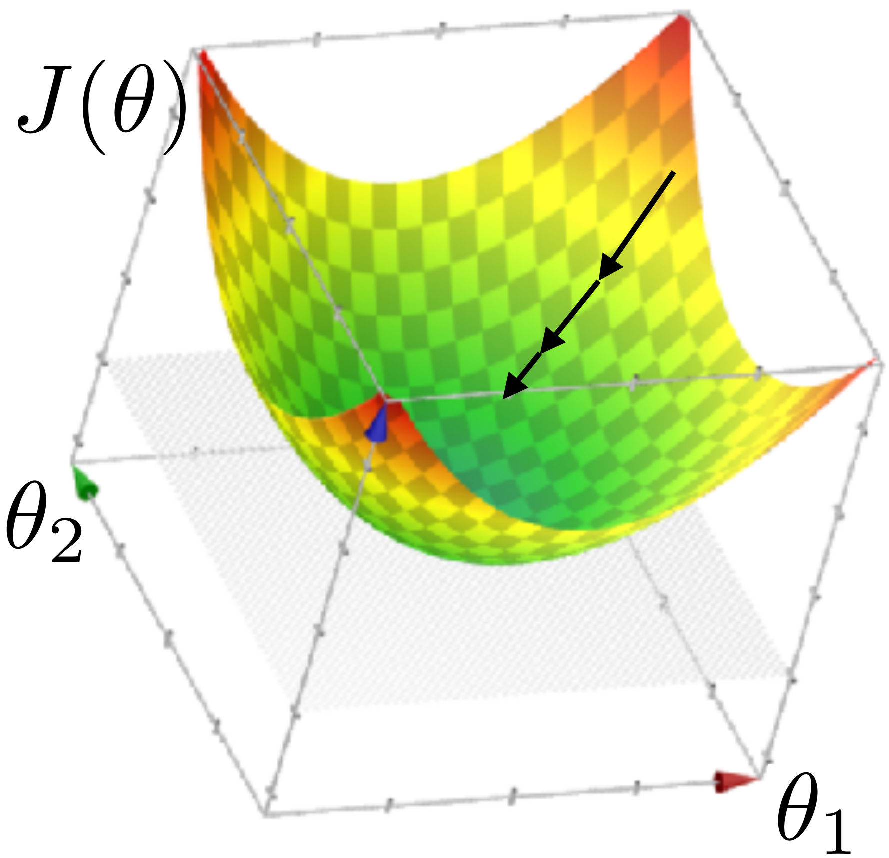


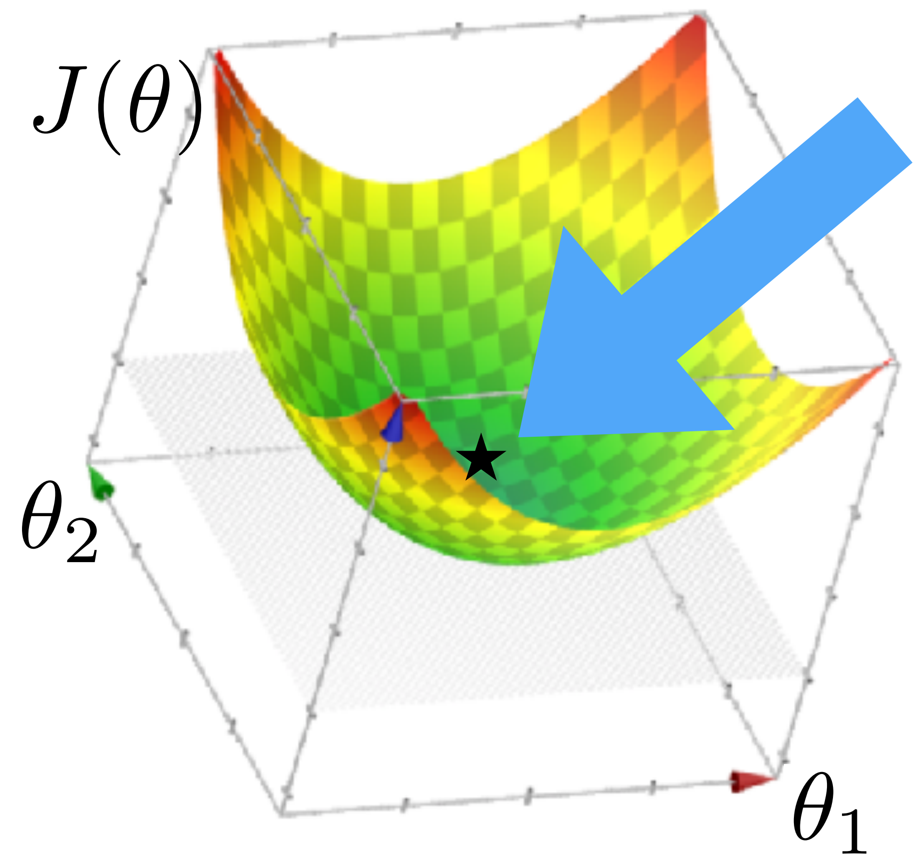
- Accuracy doesn't mean anything without running time
- Running time doesn't mean anything without accuracy
- Need to measure accuracy for the running time we have

$$\theta = \underbrace{(\tilde{X}^\top \tilde{X} + n\lambda I)}_{\text{dxd}}^{-1} \tilde{X}^\top \tilde{Y}$$

Optimizing linear regression

- Gradient descent vs. analytical/closed-form/direct solution





- Accuracy doesn't mean anything without running time
- Running time doesn't mean anything without accuracy
- Need to measure accuracy for the running time we have

$$\theta = \underbrace{(\tilde{X}^\top \tilde{X} + n\lambda I)}_{\text{dxd}}{}^{-1} \tilde{X}^\top \tilde{Y}$$

Matrix inversion: $O(d^3)$

Gradient descent for linear regression

Gradient descent for linear regression

`LinearRegression-Gradient-Descent(` $\theta_{\text{init}}, \theta_{0,\text{init}}, \eta, T$ `)`

Gradient descent for linear regression

`LinearRegression-Gradient-Descent(` $\theta_{\text{init}}, \theta_{0,\text{init}}, \eta, T$ `)`

Exactly gradient descent with *f* given by linear regression objective

Gradient descent for linear regression

`LinearRegression-Gradient-Descent(` $\theta_{\text{init}}, \theta_{0,\text{init}}, \eta, T$ `)`

`Initialize` $\theta^{(0)} = \theta_{\text{init}}$

`Initialize` $\theta_0^{(0)} = \theta_{0,\text{init}}$

Exactly gradient descent with *f* given by linear regression objective

Gradient descent for linear regression

```
LinearRegression-Gradient-Descent( θ_init, θ_0,init, η, T )
  Initialize θ^(0) = θ_init
  Initialize θ_0^(0) = θ_0,init
  for t = 1 to T
```

$\text{LinearRegression-Gradient-Descent}(\theta_{\text{init}}, \theta_{0,\text{init}}, \eta, T)$

Initialize $\theta^{(0)} = \theta_{\text{init}}$

Initialize $\theta_0^{(0)} = \theta_{0,\text{init}}$

for t = 1 **to** T

Exactly gradient descent with *f* given by linear regression objective

Gradient descent for linear regression

```
LinearRegression-Gradient-Descent( $\theta_{\text{init}}, \theta_{0,\text{init}}, \eta, T$ )
  Initialize $\theta^{(0)} = \theta_{\text{init}}$
  Initialize $\theta_0^{(0)} = \theta_{0,\text{init}}$
  for t = 1 to T
```

$$\theta^{(t)} = \theta^{(t-1)} - \eta \left\{ \frac{2}{n} \sum_{i=1}^{n} \left[\theta^{(t-1)\top} x^{(i)} + \theta_0^{(t-1)} - y^{(i)} \right] x^{(i)} + 2\lambda\theta^{(t-1)} \right\}$$

Exactly gradient descent with *f* given by linear regression objective

Gradient descent for linear regression

```
LinearRegression-Gradient-Descent( θ_init, θ_0,init, η, T )
  Initialize θ^(0) = θ_init
  Initialize θ_0^(0) = θ_0,init
  for t = 1 to T
```

$$\theta^{(t)} = \theta^{(t-1)} - \eta\left\{\frac{2}{n}\sum_{i=1}^{n}\left[\theta^{(t-1)\top}x^{(i)} + \theta_0^{(t-1)} - y^{(i)}\right]x^{(i)} + 2\lambda\theta^{(t-1)}\right\}$$

$$\theta_0^{(t)} = \theta_0^{(t-1)} - \eta\left\{\frac{2}{n}\sum_{i=1}^{n}\left[\theta^{(t-1)\top}x^{(i)} + \theta_0^{(t-1)} - y^{(i)}\right]\right\}$$

Exactly gradient descent with *f* given by linear regression objective

Gradient descent for linear regression

```
LinearRegression-Gradient-Descent( θ_init, θ_0,init, η, T )
  Initialize θ^(0) = θ_init
  Initialize θ_0^(0) = θ_0,init
  for t = 1 to T
```

$\texttt{LinearRegression-Gradient-Descent(}\ \theta_{\text{init}}, \theta_{0,\text{init}}, \eta, T\ \texttt{)}$

Initialize $\theta^{(0)} = \theta_{\text{init}}$

Initialize $\theta_0^{(0)} = \theta_{0,\text{init}}$

for t = 1 **to** T

$$\theta^{(t)} = \theta^{(t-1)} - \eta\left\{\frac{2}{n}\sum_{i=1}^{n}\left[\theta^{(t-1)\top}x^{(i)} + \theta_0^{(t-1)} - y^{(i)}\right]x^{(i)} + 2\lambda\theta^{(t-1)}\right\}$$

$$\theta_0^{(t)} = \theta_0^{(t-1)} - \eta\left\{\frac{2}{n}\sum_{i=1}^{n}\left[\theta^{(t-1)\top}x^{(i)} + \theta_0^{(t-1)} - y^{(i)}\right]\right\}$$

Return $\theta^{(t)}, \theta_0^{(t)}$

Exactly gradient descent with *f* given by linear regression objective

Stochastic gradient descent

Stochastic gradient descent

- Linear regression objective with $\lambda = 0$:

Stochastic gradient descent

- Linear regression objective with $\lambda = 0$:

$$J_{\text{linreg}}(\theta, \theta_0) = \frac{1}{n} \sum_{i=1}^{n} (\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

Stochastic gradient descent

- Linear regression objective with $\lambda = 0$:

$$J_{\text{linreg}}(\Theta) = J_{\text{linreg}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

Stochastic gradient descent

- Linear regression objective with $\lambda = 0$:

$$J_{\text{linreg}}(\Theta) = J_{\text{linreg}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- A common machine learning objective:

$$f(\Theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\Theta)$$

Stochastic gradient descent

- Linear regression objective with $\lambda = 0$:

$$J_{\text{linreg}}(\Theta) = J_{\text{linreg}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- A common machine learning objective:

$$f(\Theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\Theta)$$

- Also recall logistic regression with $\lambda = 0$

Stochastic gradient descent

- Linear regression objective with $\lambda = 0$:

$$J_{\text{linreg}}(\Theta) = J_{\text{linreg}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- A common machine learning objective:

$$f(\Theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\Theta)$$

- Also recall logistic regression with $\lambda = 0$

`Stochastic-Gradient-Descent(`$\Theta_{\text{init}}, \eta, T$`)`

Stochastic gradient descent

- Linear regression objective with $\lambda = 0$:

$$J_{\text{linreg}}(\Theta) = J_{\text{linreg}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- A common machine learning objective:

$$f(\Theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\Theta)$$

- Also recall logistic regression with $\lambda = 0$

```
Stochastic-Gradient-Descent(Θ_init, η, T)
  Initialize Θ^(0) = Θ_init
```

Stochastic gradient descent

- Linear regression objective with $\lambda = 0$:

$$J_{\text{linreg}}(\Theta) = J_{\text{linreg}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- A common machine learning objective:

$$f(\Theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\Theta)$$

- Also recall logistic regression with $\lambda = 0$

Stochastic-Gradient-Descent($\Theta_{\text{init}}, \eta, T$)

  Initialize $\Theta^{(0)} = \Theta_{\text{init}}$

  for t = 1 **to** T

Stochastic gradient descent

- Linear regression objective with $\lambda = 0$:

$$J_{\text{linreg}}(\Theta) = J_{\text{linreg}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- A common machine learning objective:

$$f(\Theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\Theta)$$

- Also recall logistic regression with $\lambda = 0$

```
Stochastic-Gradient-Descent(Θ_init, η, T)
  Initialize Θ^(0) = Θ_init
  for t = 1 to T
    randomly select i from {1,…,n} (with equal probability)
```

Stochastic gradient descent

- Linear regression objective with $\lambda = 0$:

$$J_{\text{linreg}}(\Theta) = J_{\text{linreg}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- A common machine learning objective:

$$f(\Theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\Theta)$$

- Also recall logistic regression with $\lambda = 0$

Stochastic-Gradient-Descent($\Theta_{\text{init}}, \eta, T$)

  Initialize $\Theta^{(0)} = \Theta_{\text{init}}$

  for t = 1 **to** T

    randomly select i from {1,…,n} (with equal probability)

    $\Theta^{(t)} = \Theta^{(t-1)} - \eta(t)\nabla_\Theta f_i(\Theta^{(t-1)})$

Stochastic gradient descent

- Linear regression objective with $\lambda = 0$:

$$J_{\text{linreg}}(\Theta) = J_{\text{linreg}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- A common machine learning objective:

$$f(\Theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\Theta)$$

- Also recall logistic regression with $\lambda = 0$

```
Stochastic-Gradient-Descent(Θ_init, η, T)
  Initialize Θ^(0) = Θ_init
  for t = 1 to T
    randomly select i from {1,…,n} (with equal probability)
    Θ^(t) = Θ^(t−1) − η(t)∇_Θ f_i(Θ^(t−1))
```

$\texttt{Stochastic-Gradient-Descent}(\Theta_{\text{init}}, \eta, T)$

$\quad\texttt{Initialize}\ \Theta^{(0)} = \Theta_{\text{init}}$

$\quad\textbf{for}\ t = 1\ \textbf{to}\ T$

$\qquad\texttt{randomly select i from \{1,…,n\} (with equal probability)}$

$$\Theta^{(t)} = \Theta^{(t-1)} - \boxed{\eta(t)}\nabla_\Theta f_i(\Theta^{(t-1)})$$

Stochastic gradient descent

- Linear regression objective with $\lambda = 0$:

$$J_{\text{linreg}}(\Theta) = J_{\text{linreg}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- A common machine learning objective:

$$f(\Theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\Theta)$$

- Also recall logistic regression with $\lambda = 0$

Stochastic-Gradient-Descent($\Theta_{\text{init}}, \eta, T$)

Initialize $\Theta^{(0)} = \Theta_{\text{init}}$

for t = 1 **to** T

randomly select i from {1,…,n} (with equal probability)

$\Theta^{(t)} = \Theta^{(t-1)} - \eta(t)\nabla_\Theta f_i(\Theta^{(t-1)})$

Stochastic gradient descent

- Linear regression objective with $\lambda = 0$:

$$J_{\text{linreg}}(\Theta) = J_{\text{linreg}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- A common machine learning objective:

$$f(\Theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\Theta)$$

- Also recall logistic regression with $\lambda = 0$

Stochastic-Gradient-Descent($\Theta_{\text{init}}, \eta, T$)

  Initialize $\Theta^{(0)} = \Theta_{\text{init}}$

  for t = 1 **to** T

    randomly select i from {1,…,n} (with equal probability)

    $\Theta^{(t)} = \Theta^{(t-1)} - \eta(t)\nabla_\Theta f_i(\Theta^{(t-1)})$

Stochastic gradient descent

- Linear regression objective with $\lambda = 0$:

$$J_{\text{linreg}}(\Theta) = J_{\text{linreg}}(\theta, \theta_0) = \frac{1}{n}\sum_{i=1}^{n}(\theta^\top x^{(i)} + \theta_0 - y^{(i)})^2$$

- A common machine learning objective:

$$f(\Theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\Theta)$$

- Also recall logistic regression with $\lambda = 0$

Stochastic-Gradient-Descent($\Theta_{\text{init}}, \eta, T$)

  Initialize $\Theta^{(0)} = \Theta_{\text{init}}$

  for t = 1 **to** T

    randomly select i from {1,…,n} (with equal probability)

    $\Theta^{(t)} = \Theta^{(t-1)} - \eta(t)\nabla_\Theta f_i(\Theta^{(t-1)})$

  Return $\Theta^{(t)}$