

# Math for Machine Learning

Week 1.1: Vectors, matrices, and least squares regression

By: Samuel Deng

# Lesson Overview

**Vectors and matrices (an ML view).** A single datapoint/sample in ML is represented as a [vector](#)  $\mathbf{x} \in \mathbb{R}^d$ . A collection of samples is represented as a [matrix](#)  $\mathbf{X} \in \mathbb{R}^{n \times d}$ .

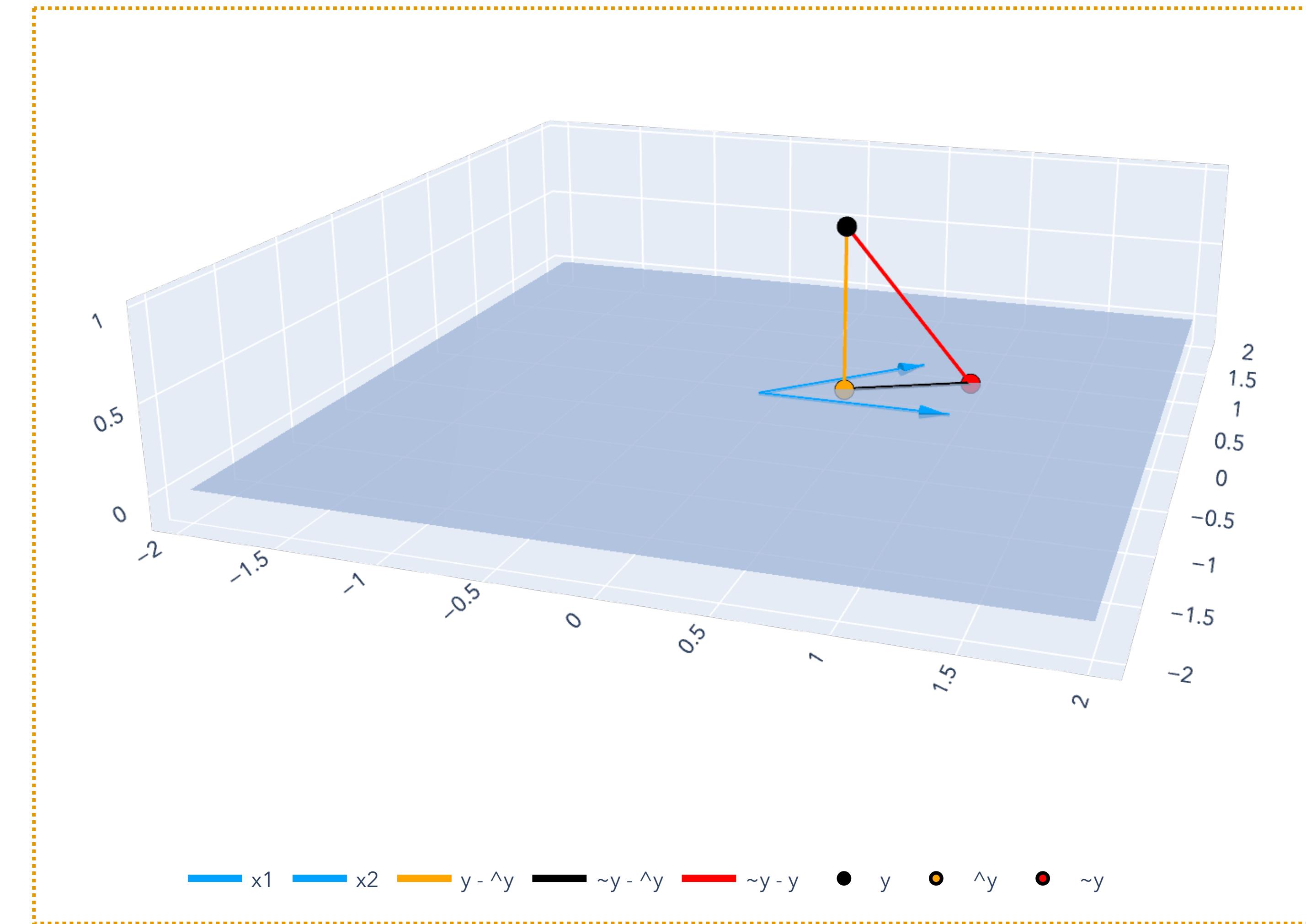
**Regression (the basic ML problem).** The basic problem in machine learning is [regression](#): constructing a “best-fit” model from a collection of observed data  $\mathbf{x} \in \mathbb{R}^d$  and labels  $y \in \mathbb{R}$ :  $(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)$ .

**Linear independence.** [Linearly independent](#) vectors are vectors that are not redundant; linearly dependent vectors can be expressed as simple (linear) combinations of other vectors.

**Span.** The [span](#) of a set of vectors includes all vectors we can form by simple (linear) combinations of the vectors in the set.

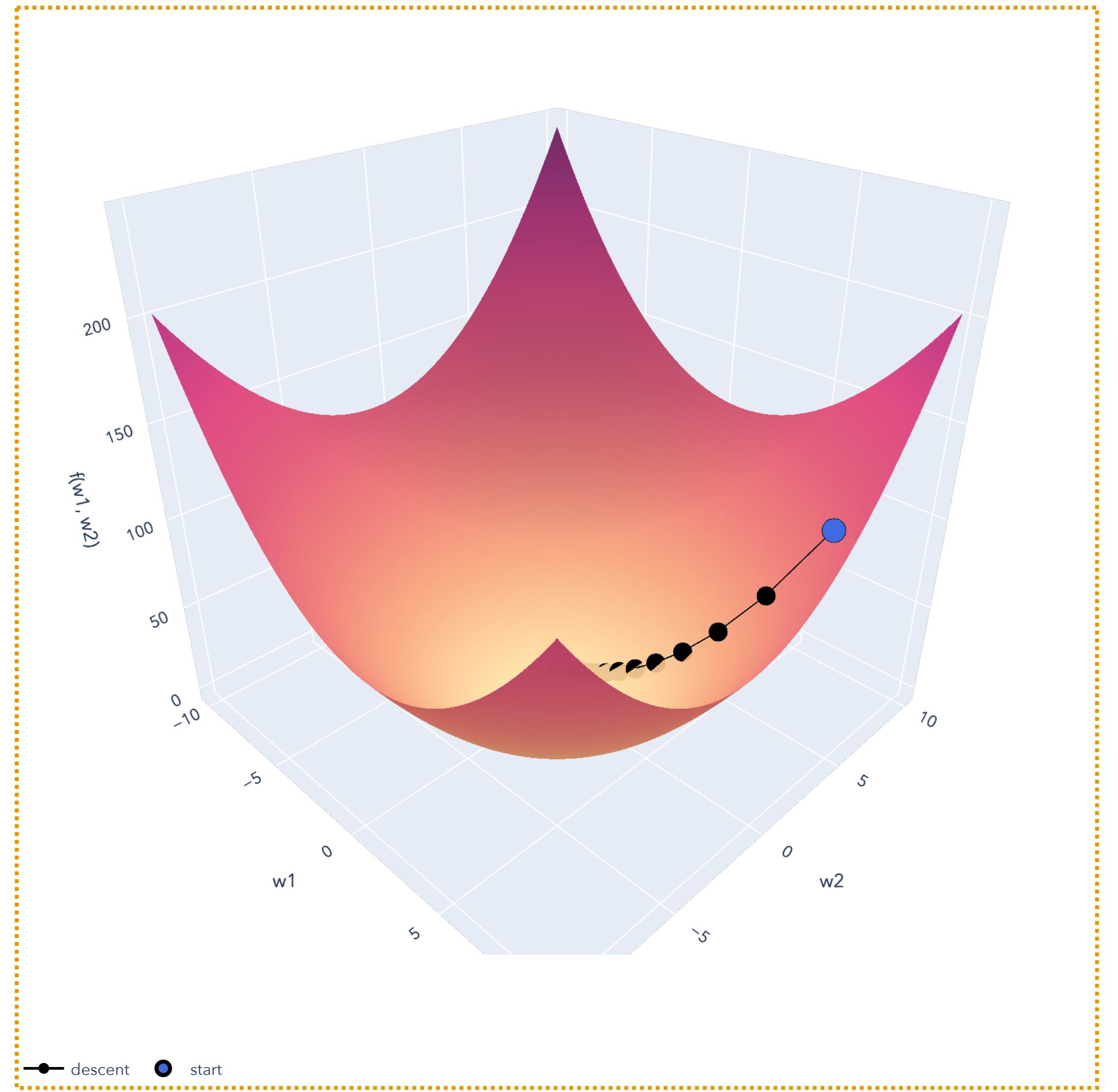
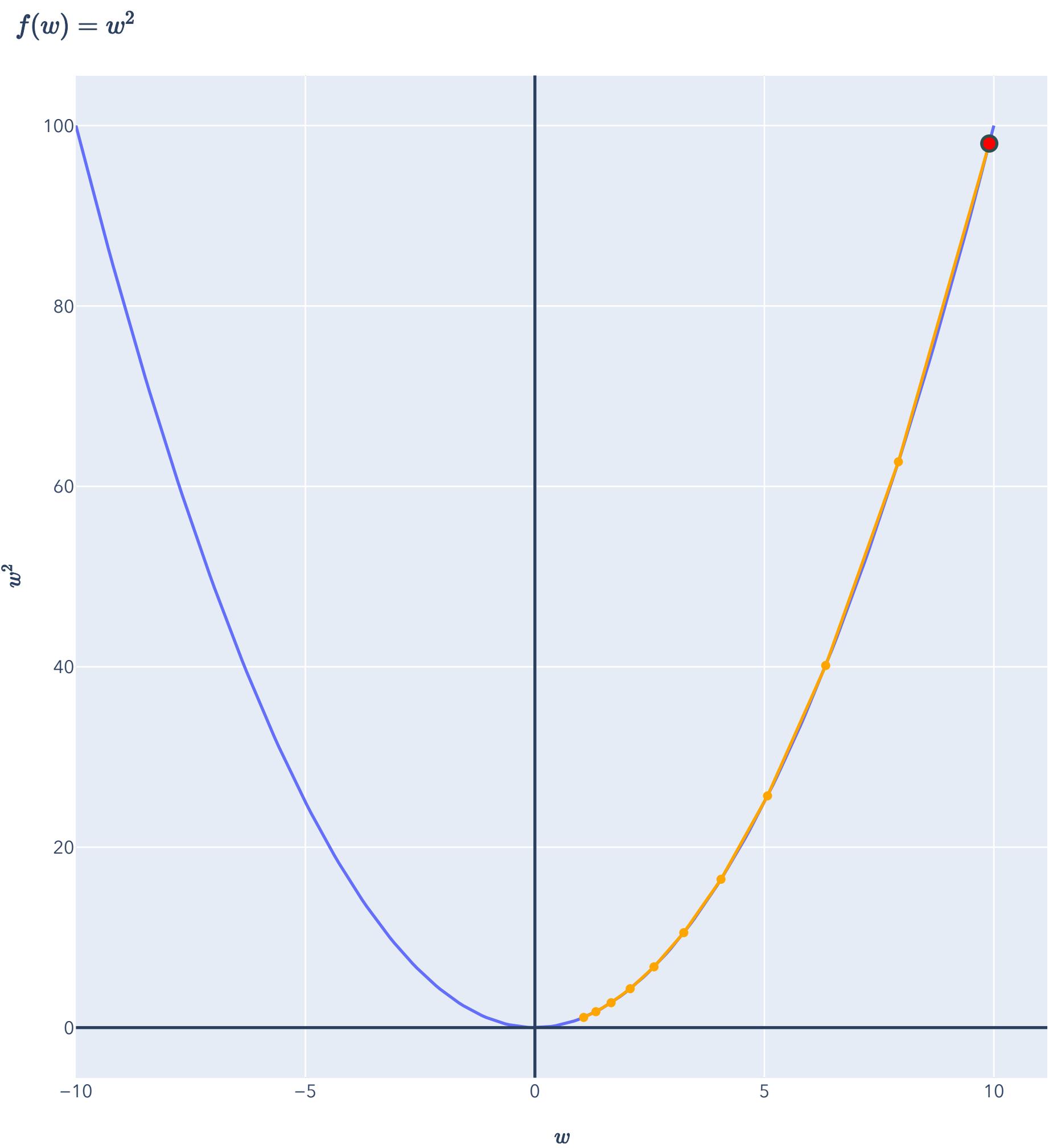
# Lesson Overview

## Big Picture: Least Squares



# Lesson Overview

## Big Picture: Gradient Descent



# Vectors & Matrices

# Vectors

## Review from linear algebra

A vector is a list of numbers. We write  $\mathbf{x} \in \mathbb{R}^d$  as:

$$\mathbf{x} := \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \text{ or } \mathbf{x} := (x_1, \dots, x_d).$$

By convention, our vectors will be *column vectors*. A *row vector* looks like:

$$\mathbf{x}^\top = [x_1 \quad \dots \quad x_d]$$

# Vectors

## Review from linear algebra

In  $\mathbb{R}^n$ , a special set of vectors is the unit basis vectors:

$$\mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \dots, \mathbf{e}_n = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}$$

# Vectors

## Review from linear algebra

Vectors can interchangeably thought of as *points*:

or “*arrows*”:

# Matrices

## Review from linear algebra

A **matrix** is a box of numbers, or a list of vectors. We write  $\mathbf{X} \in \mathbb{R}^{n \times d}$  as:

$$\mathbf{X} = \begin{bmatrix} \uparrow & & \uparrow \\ \mathbf{x}_1 & \dots & \mathbf{x}_d \\ \downarrow & & \downarrow \end{bmatrix} \quad \text{or} \quad \mathbf{X} = \begin{bmatrix} \leftarrow & \mathbf{x}_1^\top & \rightarrow \\ & \vdots & \\ \leftarrow & \mathbf{x}_n^\top & \rightarrow \end{bmatrix}.$$

# Matrices

## Review from linear algebra

A **matrix** is a box of numbers, or a list of vectors. We write  $\mathbf{X} \in \mathbb{R}^{n \times d}$  as:

$$\mathbf{X} = \begin{bmatrix} \uparrow & & \uparrow \\ \mathbf{x}_1 & \dots & \mathbf{x}_d \\ \downarrow & & \downarrow \end{bmatrix} \quad \text{or} \quad \mathbf{X} = \begin{bmatrix} \leftarrow & \mathbf{x}_1^\top & \rightarrow \\ & \vdots & \\ \leftarrow & \mathbf{x}_n^\top & \rightarrow \end{bmatrix}.$$

**Column definition:** stack column vectors  $\mathbf{x}_1, \dots, \mathbf{x}_d \in \mathbb{R}^n$  side-by-side next to each other.

**Row definition:** take (by convention, column) vectors  $\mathbf{x}_1, \dots, \mathbf{x}_n \in \mathbb{R}^d$ , turn them into rows  $\mathbf{x}_1^\top, \dots, \mathbf{x}_n^\top \in \mathbb{R}^{1 \times d}$ , and stack them on top of each other.

# Matrices

## Transpose

For a matrix  $\mathbf{X} \in \mathbb{R}^{n \times d}$ , its transpose is the matrix  $\mathbf{X}^\top \in \mathbb{R}^{d \times n}$  obtained from swapping  $X_{ij}^\top = X_{ji}$  for all  $i \in [d], j \in [n]$ .

$$\mathbf{X} = \begin{bmatrix} \uparrow & & \uparrow \\ \mathbf{x}_1 & \dots & \mathbf{x}_d \\ \downarrow & & \downarrow \end{bmatrix} \quad \text{or} \quad \mathbf{X} = \begin{bmatrix} \leftarrow & \mathbf{x}_1^\top & \rightarrow \\ & \vdots & \\ \leftarrow & \mathbf{x}_n^\top & \rightarrow \end{bmatrix}.$$

$$\mathbf{X}^\top = \begin{bmatrix} \uparrow & & \uparrow \\ \mathbf{x}_1 & \dots & \mathbf{x}_n \\ \downarrow & & \downarrow \end{bmatrix} \quad \text{or} \quad \mathbf{X}^\top = \begin{bmatrix} \leftarrow & \mathbf{x}_1^\top & \rightarrow \\ & \vdots & \\ \leftarrow & \mathbf{x}_d^\top & \rightarrow \end{bmatrix}.$$

# Multiplication

## Vector-vector “multiplication”

Given two vectors  $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ , their dot product (Euclidean inner product) is:

$$\mathbf{x}^\top \mathbf{y} := x_1 y_1 + \dots + x_d y_d.$$

More generally, an inner product between two vectors is written as  $\langle \mathbf{x}, \mathbf{y} \rangle$ . If not specified otherwise, we will use the dot product as default in this course.

# Multiplication

## Properties of the inner product

For any two vectors  $\mathbf{v}, \mathbf{w} \in \mathbb{R}^d$  the inner product obeys the following:

1. Symmetry.  $\langle \mathbf{v}, \mathbf{w} \rangle = \langle \mathbf{w}, \mathbf{v} \rangle$ .

2. Positive definiteness.  $\langle \mathbf{v}, \mathbf{v} \rangle \geq 0$ , and  $\langle \mathbf{v}, \mathbf{v} \rangle = 0$  if and only if  $\mathbf{v} = \mathbf{0}$ .

(note  $\langle \mathbf{v}, \mathbf{v} \rangle = \|\mathbf{v}\|^2$ , the squared norm of any vector)

3. Linearity. Let  $\alpha \in \mathbb{R}$  be a scalar and  $\mathbf{u} \in \mathbb{R}^d$  be another vector. Then:

$$\langle \alpha\mathbf{u} + \mathbf{v}, \mathbf{w} \rangle = \alpha\langle \mathbf{u}, \mathbf{w} \rangle + \langle \mathbf{v}, \mathbf{w} \rangle.$$

# Multiplication

Vector-vector “multiplication”

Example. Compute the dot product between  $\mathbf{x} = (2,5,3)$  and  $\mathbf{y} = (-1,0,3)$ .

# Multiplication

## Matrix-vector multiplication (column view)

To multiply a matrix  $\mathbf{X} \in \mathbb{R}^{n \times d}$  and a vector  $\mathbf{w} \in \mathbb{R}^d$ , we can think of the *column view*:

$$\mathbf{X}\mathbf{w} = \begin{bmatrix} \uparrow & & \uparrow \\ \mathbf{x}_1 & \dots & \mathbf{x}_d \\ \downarrow & & \downarrow \end{bmatrix} \begin{bmatrix} w_1 \\ \vdots \\ w_d \end{bmatrix} = w_1 \begin{bmatrix} \uparrow \\ \mathbf{x}_1 \\ \downarrow \end{bmatrix} + \dots + w_d \begin{bmatrix} \uparrow \\ \mathbf{x}_d \\ \downarrow \end{bmatrix}.$$

The result is  $\mathbf{X}\mathbf{w} \in \mathbb{R}^n$ .

Interpretation:  $\mathbf{X}\mathbf{w}$  is a *linear combination* of the columns of  $\mathbf{X}$ .

# Multiplication

## Matrix-vector multiplication (equation view)

To multiply a matrix  $\mathbf{X} \in \mathbb{R}^{n \times d}$  and a vector  $\mathbf{w} \in \mathbb{R}^d$ , we can think of the equation view:

$$\mathbf{X}\mathbf{w} = \begin{bmatrix} \leftarrow & \mathbf{x}_1^\top & \rightarrow \\ & \vdots & \\ \leftarrow & \mathbf{x}_n^\top & \rightarrow \end{bmatrix} \begin{bmatrix} \uparrow \\ \mathbf{w} \\ \downarrow \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1^\top \mathbf{w} \\ \vdots \\ \mathbf{x}_n^\top \mathbf{w} \end{bmatrix}$$

The result is  $\mathbf{X}\mathbf{w} \in \mathbb{R}^n$ .

Interpretation:  $\mathbf{X}\mathbf{w}$  compiles the “right-hand sides” of a system of linear equations.

# Multiplication

## Matrix-vector multiplication

Example. Compute the matrix-vector product:

$$\mathbf{X}\mathbf{w} = \begin{bmatrix} 1 & -1 & 2 \\ 0 & 2 & 3 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \\ -1 \end{bmatrix}$$

# Multiplication

## Matrix-matrix multiplication (matrix-vector view)

To multiply two matrices  $\mathbf{U} \in \mathbb{R}^{n \times r}$  and  $\mathbf{V} \in \mathbb{R}^{r \times d}$ , we just think of *d different matrix-vector products*:

$$\mathbf{UV} = \mathbf{U} \begin{bmatrix} \uparrow & & \uparrow \\ \mathbf{v}_1 & \dots & \mathbf{v}_d \\ \downarrow & & \downarrow \end{bmatrix} = \begin{bmatrix} \uparrow & & \uparrow \\ \mathbf{U}\mathbf{v}_1 & \dots & \mathbf{U}\mathbf{v}_d \\ \downarrow & & \downarrow \end{bmatrix}$$

The result is  $\mathbf{X} = \mathbf{UV} \in \mathbb{R}^{n \times d}$ .

# Multiplication

Matrix-matrix multiplication (inner product/entry view)

To multiply two matrices  $\mathbf{U} \in \mathbb{R}^{n \times r}$  and  $\mathbf{V} \in \mathbb{R}^{r \times d}$ , we just think of *nd different inner products*:

$$\mathbf{UV} = \begin{bmatrix} \leftarrow & \mathbf{u}_1^\top & \rightarrow \\ & \vdots & \\ \leftarrow & \mathbf{u}_n^\top & \rightarrow \end{bmatrix} \begin{bmatrix} \uparrow & & \uparrow \\ \mathbf{v}_1 & \dots & \mathbf{v}_d \\ \downarrow & & \downarrow \end{bmatrix} = \begin{bmatrix} \mathbf{u}_1^\top \mathbf{v}_1 & \dots & \mathbf{u}_1^\top \mathbf{v}_d \\ \vdots & \ddots & \vdots \\ \mathbf{u}_n^\top \mathbf{v}_1 & \dots & \mathbf{u}_n^\top \mathbf{v}_d \end{bmatrix}$$

$$(\mathbf{UV})_{ij} = \mathbf{u}_i^\top \mathbf{v}_j \text{ for all } i \in [n], j \in [d].$$

The result is  $\mathbf{X} = \mathbf{UV} \in \mathbb{R}^{n \times d}$ .

# Multiplication

## Matrix-matrix multiplication (outer product view)

To multiply two matrices  $\mathbf{U} \in \mathbb{R}^{n \times r}$  and  $\mathbf{V} \in \mathbb{R}^{r \times d}$ , we just think of *summing r different outer products (n × d matrices)*:

$$\mathbf{UV} = \begin{bmatrix} \uparrow & & \uparrow \\ \mathbf{u}_1 & \dots & \mathbf{u}_r \\ \downarrow & & \downarrow \end{bmatrix} \begin{bmatrix} \leftarrow & \mathbf{v}_1^\top & \rightarrow \\ & \vdots & \\ \leftarrow & \mathbf{v}_r^\top & \rightarrow \end{bmatrix} = \begin{bmatrix} \uparrow \\ \mathbf{u}_1 \\ \downarrow \end{bmatrix} [\leftarrow \mathbf{v}_1 \rightarrow] + \dots + \begin{bmatrix} \uparrow \\ \mathbf{u}_r \\ \downarrow \end{bmatrix} [\leftarrow \mathbf{v}_r \rightarrow]$$

The result is  $\mathbf{X} = \mathbf{UV} \in \mathbb{R}^{n \times d}$ .

# Matrices

## Inverses and Identity Matrix

A square matrix  $\mathbf{X} \in \mathbb{R}^{d \times d}$  is invertible if there exists a matrix  $\mathbf{X}^{-1} \in \mathbb{R}^{d \times d}$  (the inverse) such that:

$$\mathbf{X}^{-1}\mathbf{X} = \mathbf{X}\mathbf{X}^{-1} = \mathbf{I},$$

where  $\mathbf{I} \in \mathbb{R}^{d \times d}$  is the identity matrix:

$$\mathbf{I} := \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}.$$

# Regression

# Regression

The main problem of our course

Collect  $d$  measurements  $\mathbf{x}_1, \dots, \mathbf{x}_n \in \mathbb{R}^d$  for  $n$  students...

where  $y_i \in \mathbb{R}$  denotes the test score of a student.

Given the measurements for a new student,  $\mathbf{x}_0 \in \mathbb{R}^d$ , what is their test score?

# Regression

The main problem of our course

We observe  $n$  samples of training (observed) features  $\mathbf{x}_1, \dots, \mathbf{x}_n \in \mathbb{R}^d$ , with labels  $y_1, \dots, y_n \in \mathbb{R}$ .

$$\mathbf{x}_i = \begin{bmatrix} x_{i1} \\ \vdots \\ x_{id} \end{bmatrix}$$

Goal: Given a new unlabelled sample,  $\mathbf{x}_0$ , make a prediction  $\hat{y}$  such that  $\hat{y} \approx y_0$ .

# Regression

The main problem of our course

Goal: Given a new unlabelled sample,  $\mathbf{x}_0$ , make a prediction  $\hat{y}$  such that  $\hat{y} \approx y_0$ .

To do this, we will construct a *model* for the observed data.

A *linear model* is represented with a weight vector  $\mathbf{w} \in \mathbb{R}^d$ . To make a prediction with the weight vector, we take an inner product.

$$\hat{y} = \langle \mathbf{w}, \mathbf{x}_0 \rangle = w_1 x_{01} + \dots + w_d x_{0d}.$$

# Regression

The main problem of our course

How do we construct the weight vector  $\mathbf{w} \in \mathbb{R}^d$ ?

*Learn it from the observed data  $(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)$ .*

For some weight vector  $\mathbf{w} \in \mathbb{R}^d$ , its predictions on the observed data are:

$$\begin{bmatrix} \hat{y}_1 \\ \vdots \\ \hat{y}_n \end{bmatrix} = \hat{\mathbf{y}} = \mathbf{X}\mathbf{w} = \begin{bmatrix} \leftarrow & \mathbf{x}_1^\top & \rightarrow \\ & \vdots & \\ \leftarrow & \mathbf{x}_n^\top & \rightarrow \end{bmatrix} \begin{bmatrix} \uparrow \\ \mathbf{w} \\ \downarrow \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1^\top \mathbf{w} \\ \vdots \\ \mathbf{x}_n^\top \mathbf{w} \end{bmatrix} = \begin{bmatrix} \langle \mathbf{x}_1, \mathbf{w} \rangle \\ \vdots \\ \langle \mathbf{x}_n, \mathbf{w} \rangle \end{bmatrix}$$

# Regression

The main problem of our course

For some weight vector  $\mathbf{w} \in \mathbb{R}^d$ , its predictions on the observed data are:

$$\begin{bmatrix} \hat{y}_1 \\ \vdots \\ \hat{y}_n \end{bmatrix} = \hat{\mathbf{y}} = \mathbf{X}\mathbf{w} = \begin{bmatrix} \leftarrow & \mathbf{x}_1^\top & \rightarrow \\ & \vdots & \\ \leftarrow & \mathbf{x}_n^\top & \rightarrow \end{bmatrix} \begin{bmatrix} \uparrow \\ \mathbf{w} \\ \downarrow \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1^\top \mathbf{w} \\ \vdots \\ \mathbf{x}_n^\top \mathbf{w} \end{bmatrix} = \begin{bmatrix} \langle \mathbf{x}_1, \mathbf{w} \rangle \\ \vdots \\ \langle \mathbf{x}_n, \mathbf{w} \rangle \end{bmatrix}$$

# Regression

The main problem of our course

Goal: Given a new unlabelled sample,  $\mathbf{x}_0$ , make a prediction  $\hat{y}$  such that  $\hat{y} \approx y_0$ .

If the new sample  $(\mathbf{x}_0, y_0)$  is “distributed like” the training samples  $\mathbf{X} \in \mathbb{R}^{n \times d}$  and  $\mathbf{y} \in \mathbb{R}^n$ , then it’s not a bad idea to find  $\mathbf{w} \in \mathbb{R}^d$  so:

$$\mathbf{X}\mathbf{w} = \hat{\mathbf{y}} \approx \mathbf{y}.$$

*This will be our new goal!*

# Regression

## Setup (Example View)

Observed: Matrix of *training samples*  $\mathbf{X} \in \mathbb{R}^{n \times d}$  and vector of *training labels*  $\mathbf{y} \in \mathbb{R}^n$ .

$$\mathbf{X} = \begin{bmatrix} \leftarrow & \mathbf{x}_1^\top & \rightarrow \\ & \vdots & \\ \leftarrow & \mathbf{x}_n^\top & \rightarrow \end{bmatrix} \quad \mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}, \text{ where } \mathbf{x}_1, \dots, \mathbf{x}_n \in \mathbb{R}^d.$$

Unknown: *Weight vector*  $\mathbf{w} \in \mathbb{R}^d$  with weights  $w_1, \dots, w_d$ .

Goal: For each  $i \in [n]$ , we predict:  $\hat{y}_i = \mathbf{w}^\top \mathbf{x}_i = w_1 x_{i1} + \dots + w_d x_{id} \in \mathbb{R}$ .

Choose a weight vector that “fits the training data”:  $\mathbf{w} \in \mathbb{R}^d$  such that  $y_i \approx \hat{y}_i$  for  $i \in [n]$ , or:

$$\mathbf{X}\mathbf{w} = \hat{\mathbf{y}} \approx \mathbf{y}.$$

# Regression

## Setup (Feature View)

Observed: Matrix of *training samples*  $\mathbf{X} \in \mathbb{R}^{n \times d}$  and vector of *training labels*  $\mathbf{y} \in \mathbb{R}^n$ .

$$\mathbf{X} = \begin{bmatrix} \uparrow & & \uparrow \\ \mathbf{x}_1 & \dots & \mathbf{x}_d \\ \downarrow & & \downarrow \end{bmatrix} \quad \mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}, \text{ where } \mathbf{x}_1, \dots, \mathbf{x}_d \in \mathbb{R}^n.$$

Unknown: *Weight vector*  $\mathbf{w} \in \mathbb{R}^d$  with weights  $w_1, \dots, w_d$ .

Choose a weight vector that "fits the training data":  $\mathbf{w} \in \mathbb{R}^d$  such that  $y_i \approx \hat{y}_i$  for  $i \in [n]$ , or:

$$\mathbf{X}\mathbf{w} = \hat{\mathbf{y}} \approx \mathbf{y}.$$

# Regression

## Caveat

Choose a weight vector that "fits the training data":  $\mathbf{w} \in \mathbb{R}^d$  such that  $y_i \approx \hat{y}_i$  for  $i \in [n]$ , or:

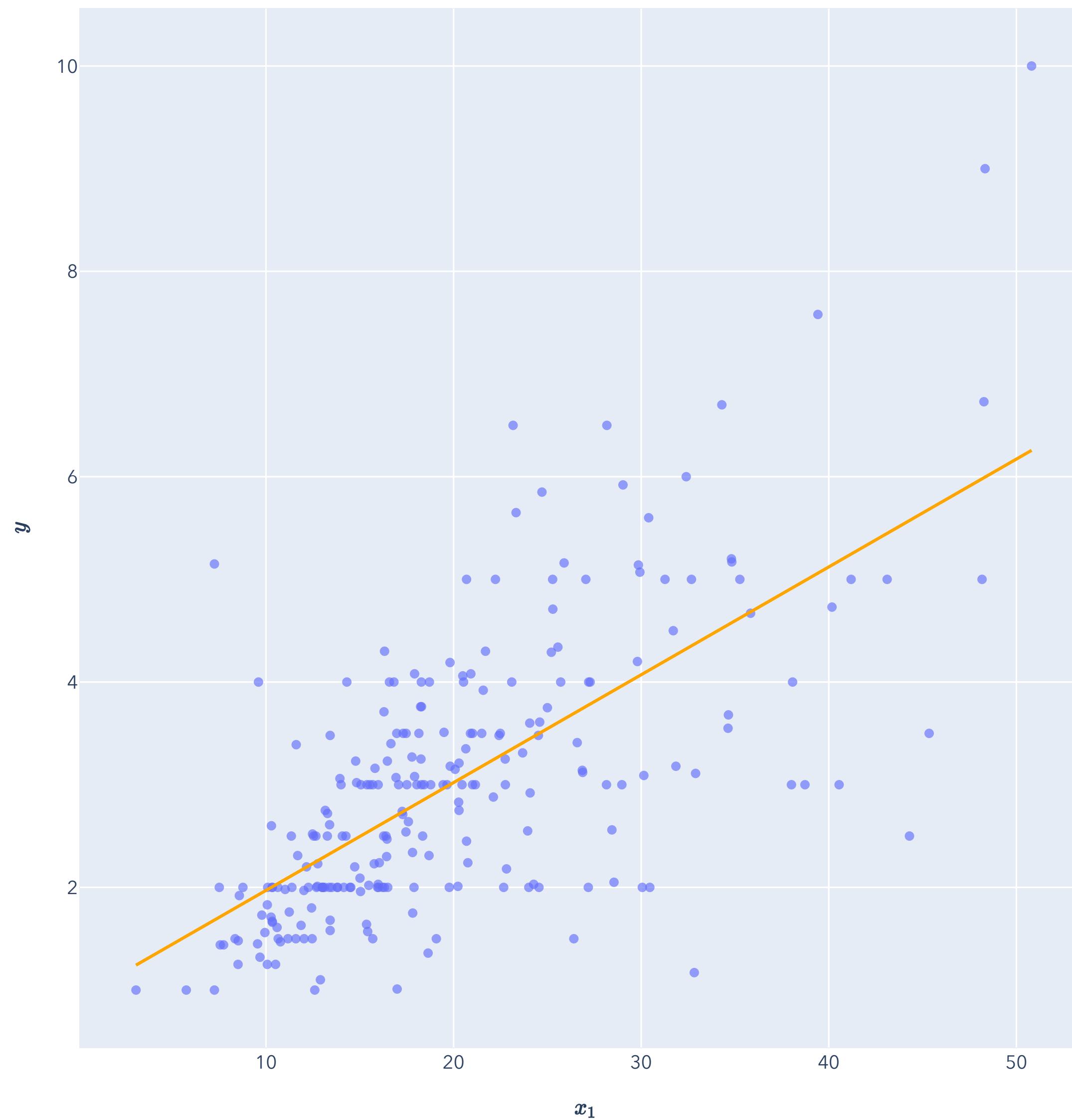
$$\mathbf{X}\mathbf{w} = \hat{\mathbf{y}} \approx \mathbf{y}.$$

In general, it may not be the case that  $\mathbf{y} = \mathbf{X}\mathbf{w}$  for any  $\mathbf{w} \in \mathbb{R}^d$  (the labels  $y_i$  don't have a perfect linear relationship with the  $\mathbf{x}_i$ ).

# Regression

Example:  $d = 1$

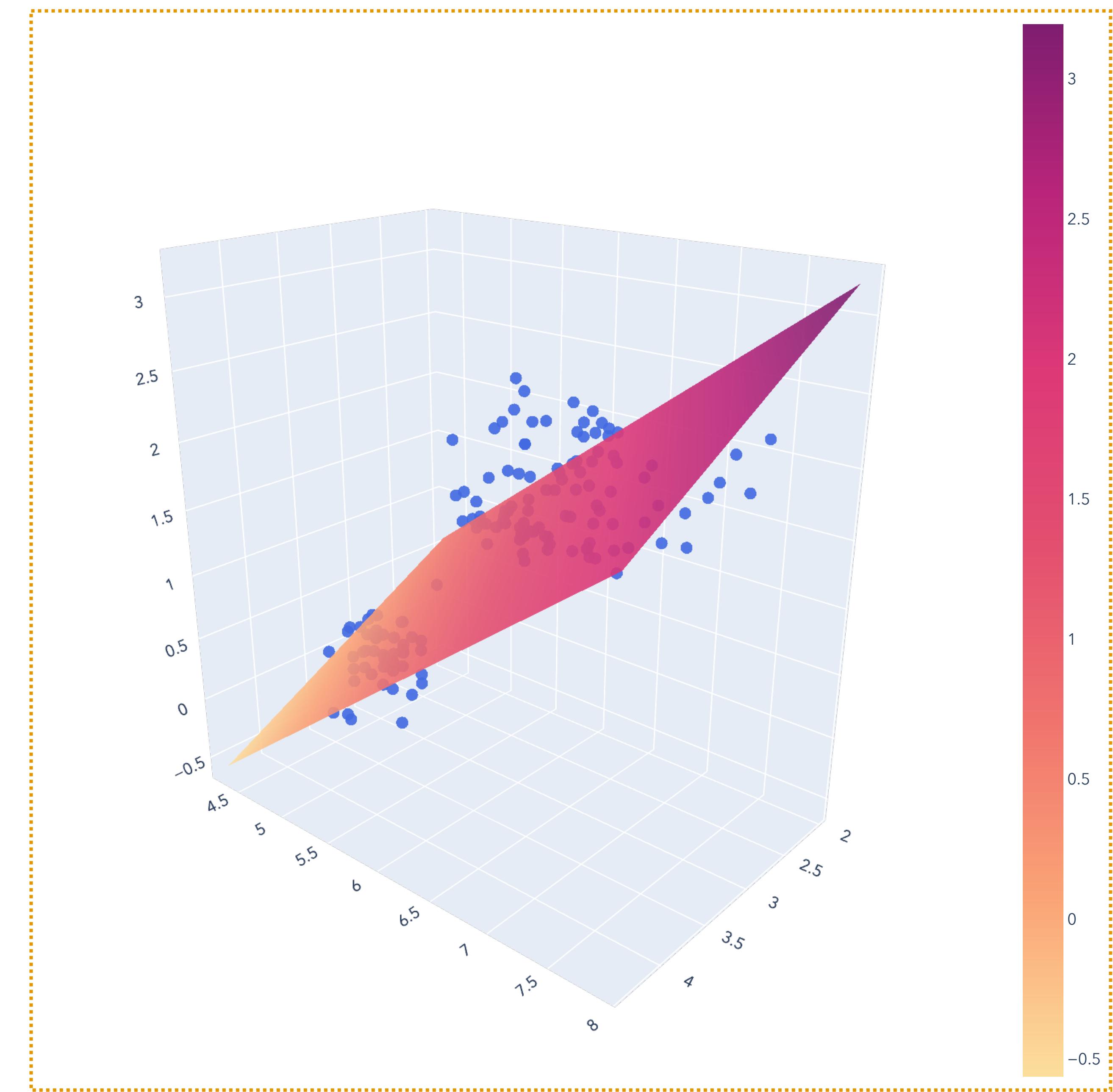
$$\mathbf{X} = \begin{bmatrix} \vdots \\ 14.07 \\ 17.51 \\ 22.42 \\ 26.88 \\ \vdots \end{bmatrix} \quad \mathbf{y} = \begin{bmatrix} \vdots \\ 2.5 \\ 3 \\ 3.48 \\ 3.12 \\ \vdots \end{bmatrix}$$



# Regression

Example:  $d = 2$

$$\mathbf{X} = \begin{bmatrix} \vdots & \vdots \\ 3.4 & 5.4 \\ 2.9 & 6.4 \\ 3.3 & 6.7 \\ 2.6 & 7.7 \\ \vdots & \vdots \end{bmatrix} \quad \mathbf{y} = \begin{bmatrix} \vdots \\ 0.4 \\ 1.3 \\ 2.1 \\ 2.3 \\ \vdots \end{bmatrix}$$



# Least Squares

## A Solution to Regression

# Regression

## Setup (Example View)

Observed: Matrix of *training samples*  $\mathbf{X} \in \mathbb{R}^{n \times d}$  and vector of *training labels*  $\mathbf{y} \in \mathbb{R}^n$ .

$$\mathbf{X} = \begin{bmatrix} \leftarrow & \mathbf{x}_1^\top & \rightarrow \\ & \vdots & \\ \leftarrow & \mathbf{x}_n^\top & \rightarrow \end{bmatrix} \quad \mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}, \text{ where } \mathbf{x}_1, \dots, \mathbf{x}_n \in \mathbb{R}^d.$$

Unknown: *Weight vector*  $\mathbf{w} \in \mathbb{R}^d$  with weights  $w_1, \dots, w_d$ .

Goal: For each  $i \in [n]$ , we predict:  $\hat{y}_i = \mathbf{w}^\top \mathbf{x}_i = w_1 x_{i1} + \dots + w_d x_{id} \in \mathbb{R}$ .

Choose a weight vector that “fits the training data”:  $\mathbf{w} \in \mathbb{R}^d$  such that  $y_i \approx \hat{y}_i$  for  $i \in [n]$ , or:

$$\mathbf{X}\mathbf{w} = \hat{\mathbf{y}} \approx \mathbf{y}.$$

# Ordinary Least Squares

## Notion of Error

In general, it may not be the case that  $\mathbf{y} = \mathbf{X}\mathbf{w}$  for any  $\mathbf{w} \in \mathbb{R}^d$  (the labels  $y_i$  don't have a perfect linear relationship with the  $\mathbf{x}_i$ ).

The residual  $r_i(\mathbf{w})$  of the  $i$ th prediction with  $\mathbf{w} \in \mathbb{R}^d$  is

$$r_i(\mathbf{w}) := \hat{y}_i - y_i = \langle \mathbf{w}, \mathbf{x}_i \rangle - y_i.$$

We can write this as a vector  $\mathbf{r} \in \mathbb{R}^n$ .

The *sum of squared residuals* is

$$SSR := \sum_{i=1}^n r_i(\mathbf{w})^2 = r_1(\mathbf{w})^2 + \dots + r_n(\mathbf{w})^2.$$

# Norms and Inner Products

## Euclidean Norm

Recall the notion of “length” from  $\mathbb{R}^2$ . For a vector  $\mathbf{x} = (x_1, x_2) \in \mathbb{R}^2$ ,

$$\|\mathbf{x}\|_2 := \sqrt{x_1^2 + x_2^2}.$$

Generalizing this, for  $\mathbf{x} \in \mathbb{R}^n$ , the Euclidean norm ( $\ell_2$ -norm) is:

$$\|\mathbf{x}\|_2 := \sqrt{x_1^2 + \dots + x_n^2} = \sqrt{\mathbf{x}^\top \mathbf{x}}.$$

$$\|\mathbf{x}\|_2^2 = \mathbf{x}^\top \mathbf{x}.$$

# Ordinary Least Squares

## Notion of Error

Residual:  $r_i(\mathbf{w}) := \hat{y}_i - y_i = \langle \mathbf{w}, \mathbf{x}_i \rangle - y_i$ , or  $\mathbf{r} \in \mathbb{R}^n$ .

The sum of squared residuals is

$$SSR := \sum_{i=1}^n r_i(\mathbf{w})^2 = r_1(\mathbf{w})^2 + \dots + r_n(\mathbf{w})^2.$$

$$SSR = \|\mathbf{r}\|^2 = \|\hat{\mathbf{y}} - \mathbf{y}\|^2 = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2.$$

# Ordinary Least Squares

## Principle of Least Squares

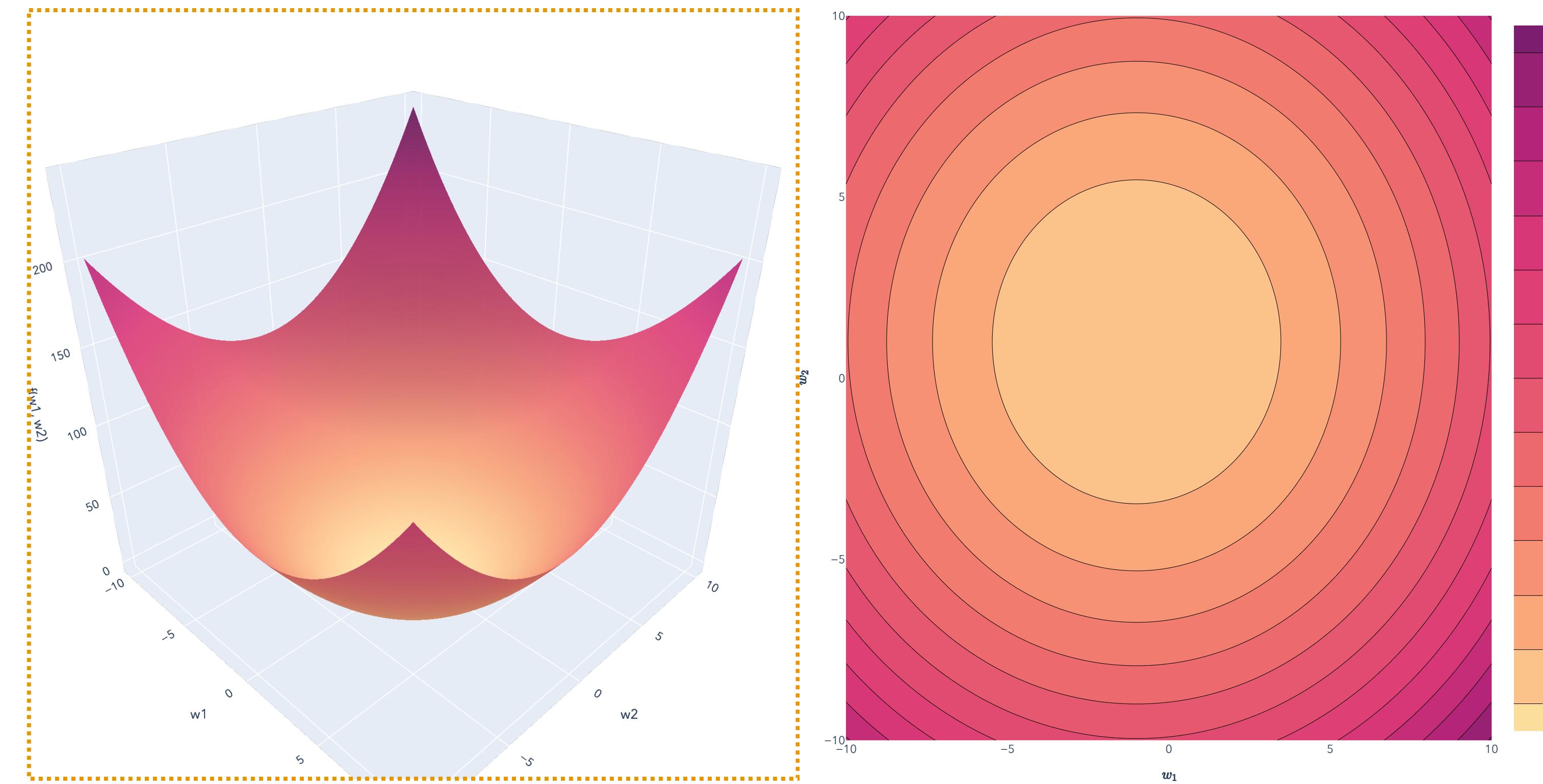
Goal: Find the  $\mathbf{w} \in \mathbb{R}^d$  that minimizes the sum of squared residuals:

$$\|\mathbf{r}\|^2 = \|\hat{\mathbf{y}} - \mathbf{y}\|^2 = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2.$$

# Ordinary Least Squares

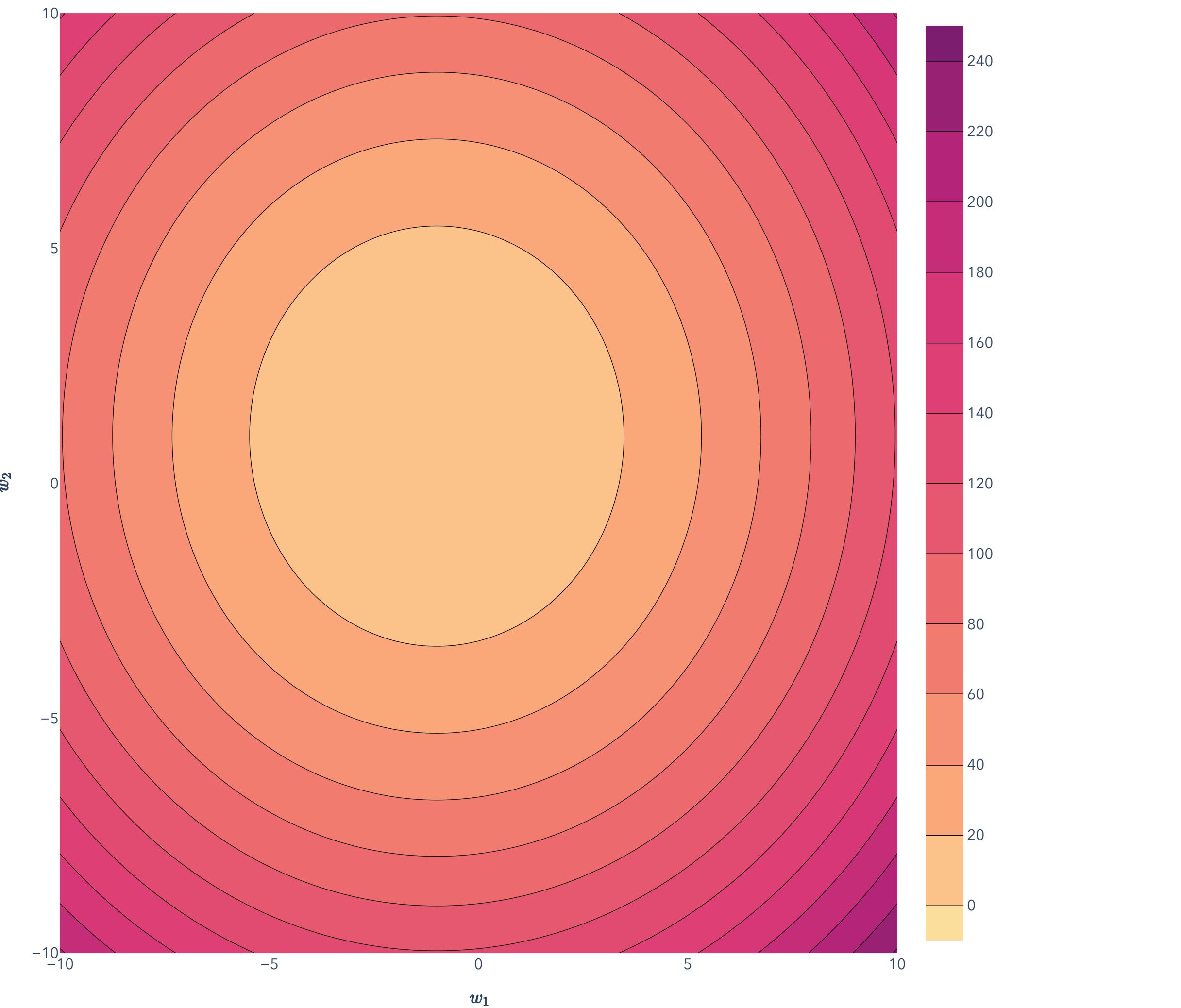
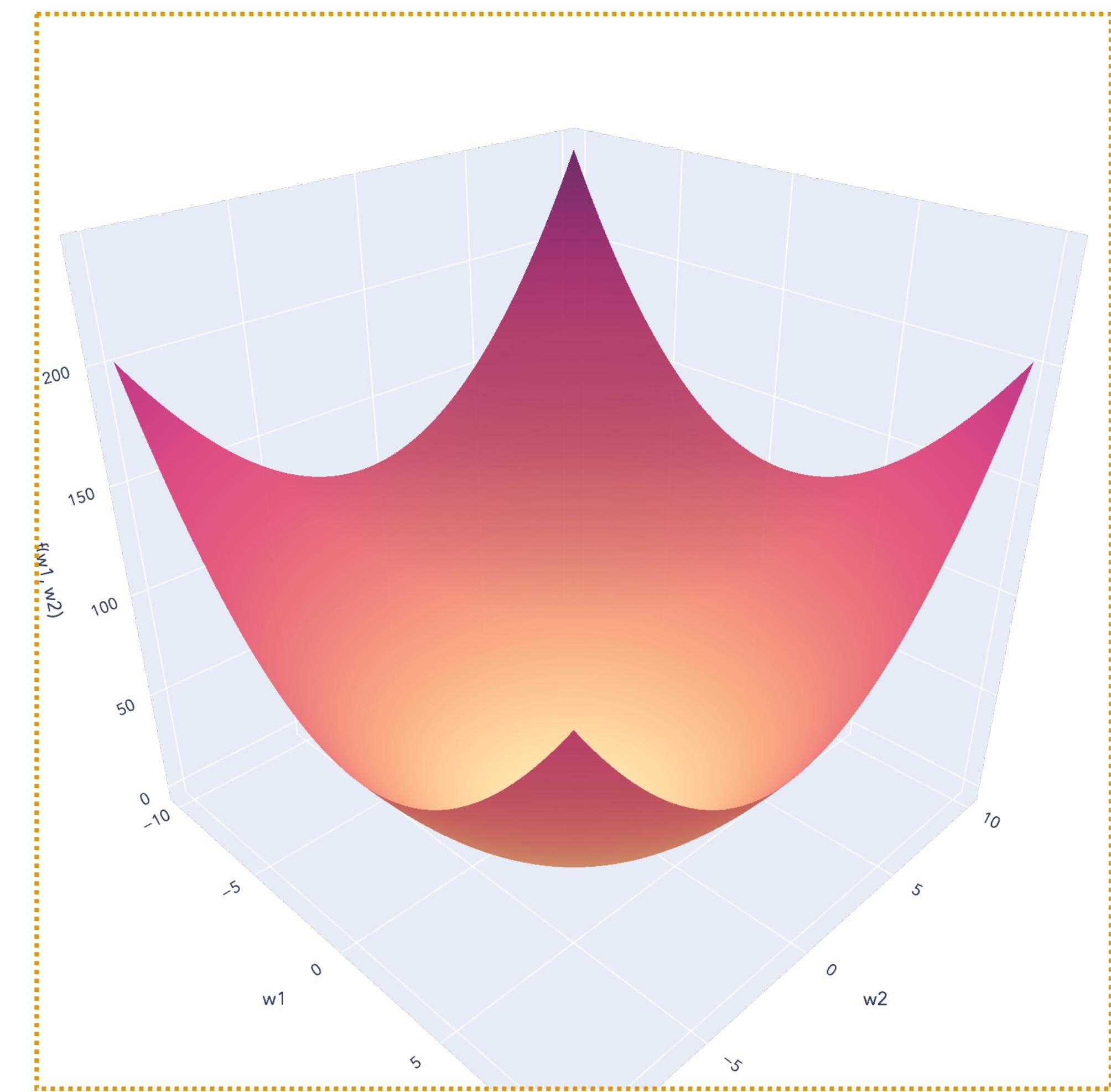
## Sum of Squared Residuals

Example: If  $\mathbf{X} \in \mathbb{R}^{n \times 2}$  and  $\mathbf{y} \in \mathbb{R}^n$ , what can  $SSR(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$  look like?



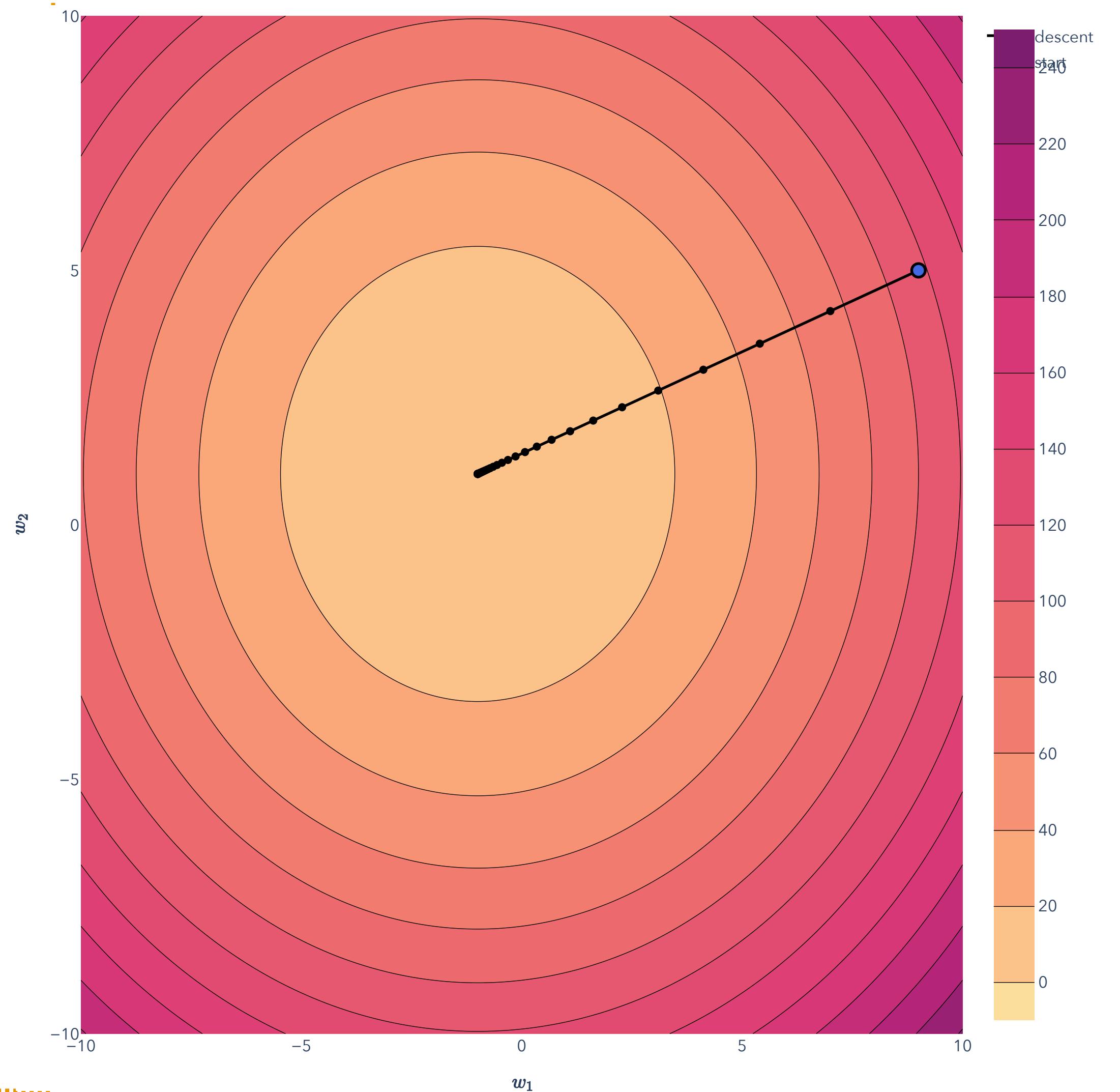
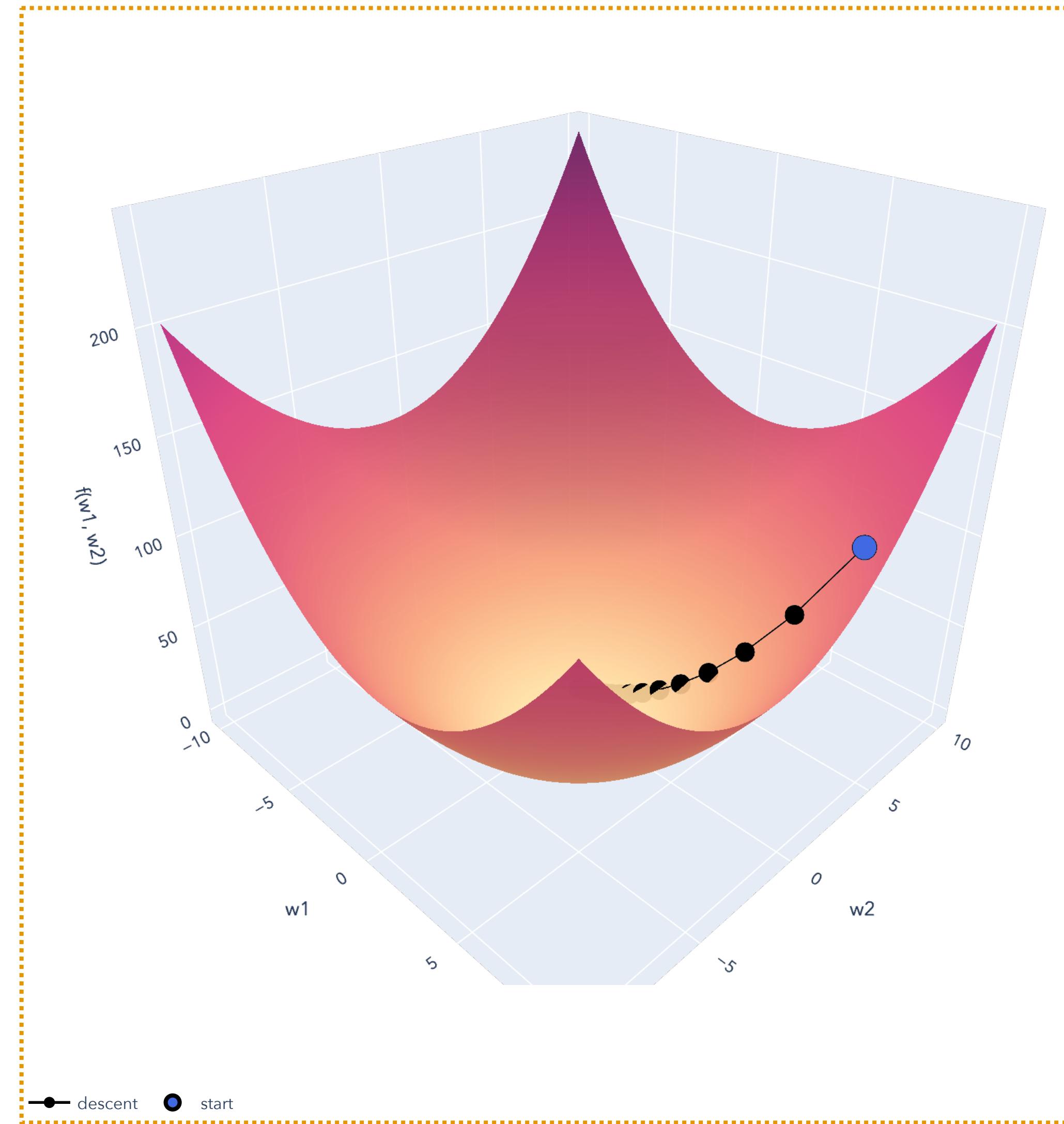
# Ordinary Least Squares

## Sum of Squared Residuals



# Ordinary Least Squares

## Sum of Squared Residuals



# Regression

## Setup (Feature View)

Observed: Matrix of *training samples*  $\mathbf{X} \in \mathbb{R}^{n \times d}$  and vector of *training labels*  $\mathbf{y} \in \mathbb{R}^n$ .

$$\mathbf{X} = \begin{bmatrix} \uparrow & & \uparrow \\ \mathbf{x}_1 & \dots & \mathbf{x}_d \\ \downarrow & & \downarrow \end{bmatrix} \quad \mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}, \text{ where } \mathbf{x}_1, \dots, \mathbf{x}_d \in \mathbb{R}^n.$$

Unknown: *Weight vector*  $\mathbf{w} \in \mathbb{R}^d$  with weights  $w_1, \dots, w_d$ .

Choose a weight vector that "fits the training data":  $\mathbf{w} \in \mathbb{R}^d$  such that  $y_i \approx \hat{y}_i$  for  $i \in [n]$ , or:

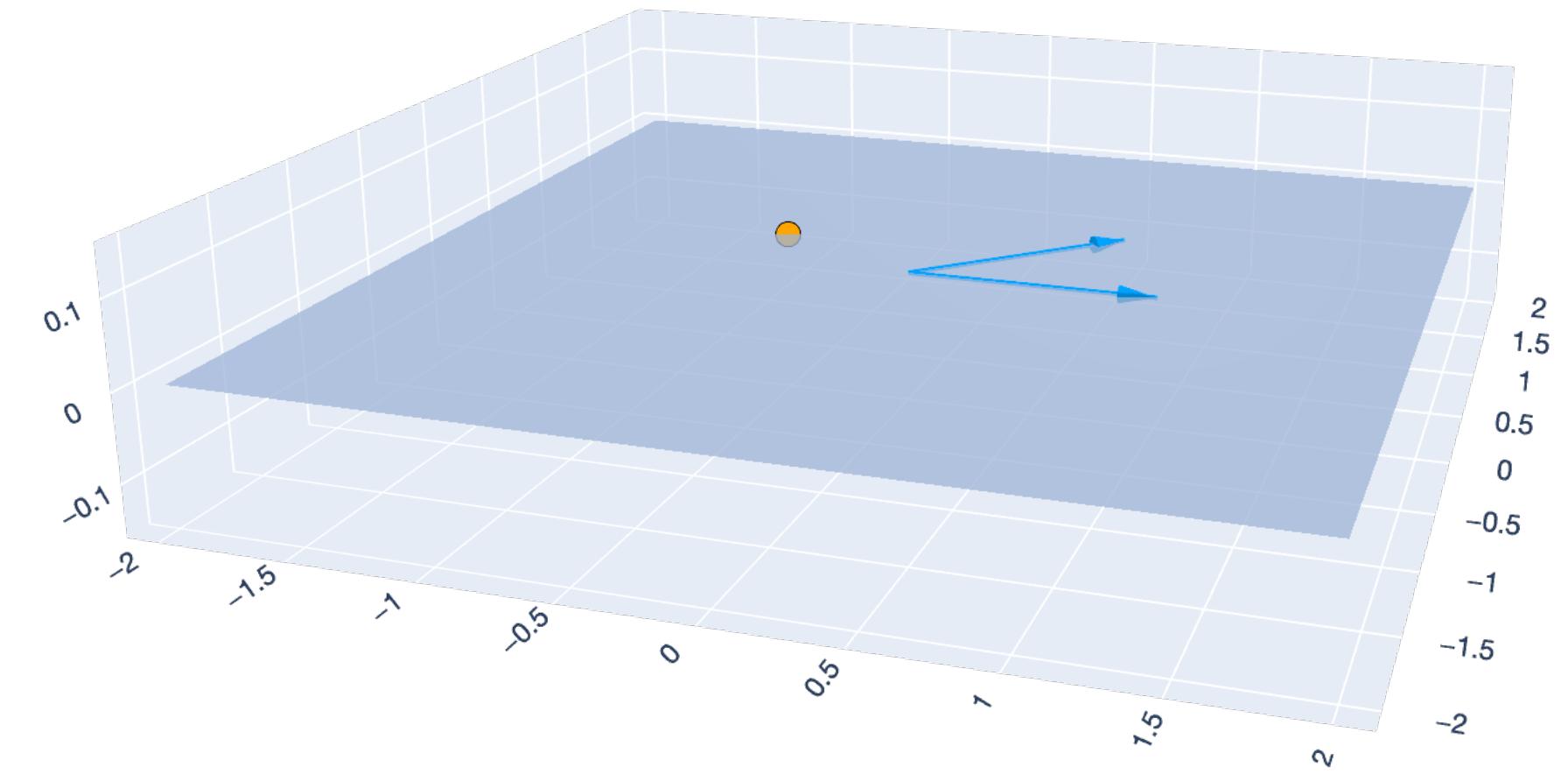
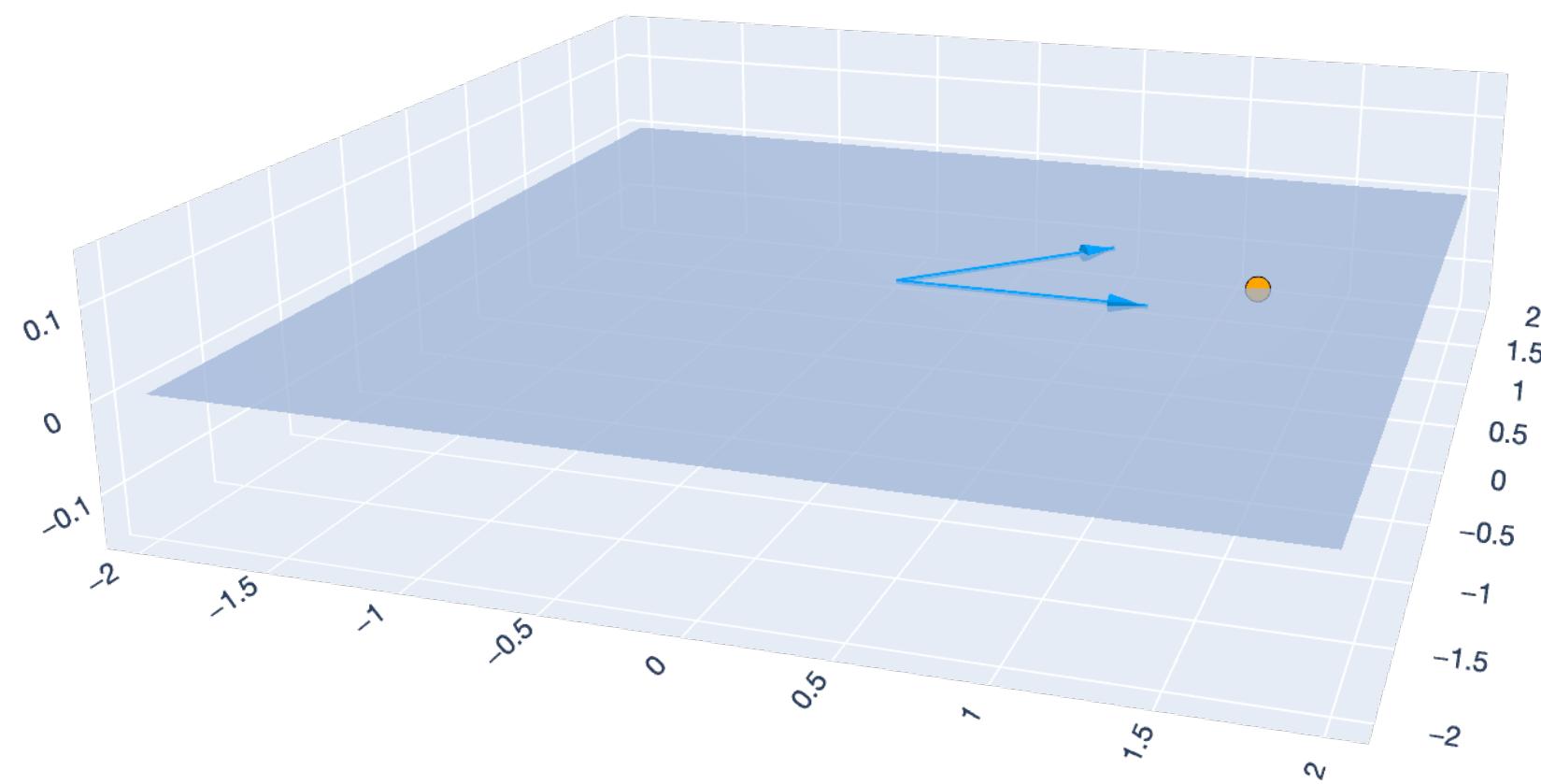
$$\mathbf{X}\mathbf{w} = \hat{\mathbf{y}} \approx \mathbf{y}.$$

# Ordinary Least Squares

## Geometry of Least Squares

Let  $n = 3$  and  $d = 2$ . In this case  $\hat{\mathbf{y}} \in \mathbb{R}^3$  is a *linear combination* of columns  $\mathbf{x}_1$  and  $\mathbf{x}_2$ .

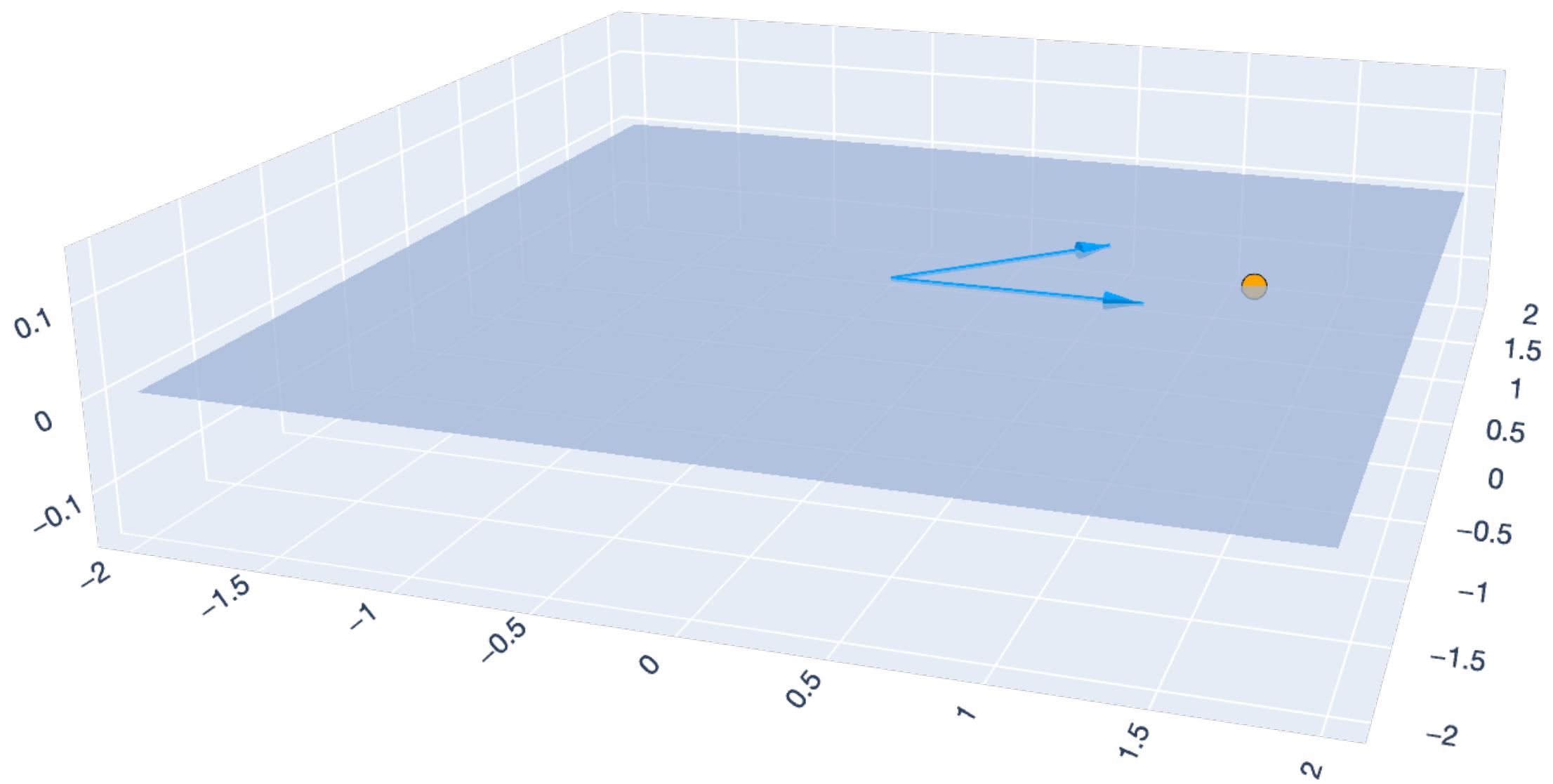
$$\hat{\mathbf{y}} = \mathbf{X}\mathbf{w} = w_1\mathbf{x}_1 + w_2\mathbf{x}_2 \in \mathbb{R}^3$$



# Span

## Idea

For a collection of vectors  $\mathbf{x}_1, \dots, \mathbf{x}_d \in \mathbb{R}^n$ , the **span** is...



# Span

## Definition

For a collection of vectors  $\mathbf{x}_1, \dots, \mathbf{x}_d \in \mathbb{R}^n$ , the **span** is the set of vectors we can attain through linear combinations of  $\mathbf{x}_1, \dots, \mathbf{x}_d$ :

$$\text{span}(\mathbf{x}_1, \dots, \mathbf{x}_d) = \left\{ \mathbf{y} \in \mathbb{R}^n : \mathbf{y} = \sum_{i=1}^d \alpha_i \mathbf{x}_i, \alpha_i \in \mathbb{R} \right\}.$$

# Span

## Examples

$$\text{span} \left( \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right)$$

$$\text{span} \left( \begin{bmatrix} 2 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ -1 \end{bmatrix} \right)$$

$$\text{span} \left( \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ -2 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \right)$$

# Ordinary Least Squares

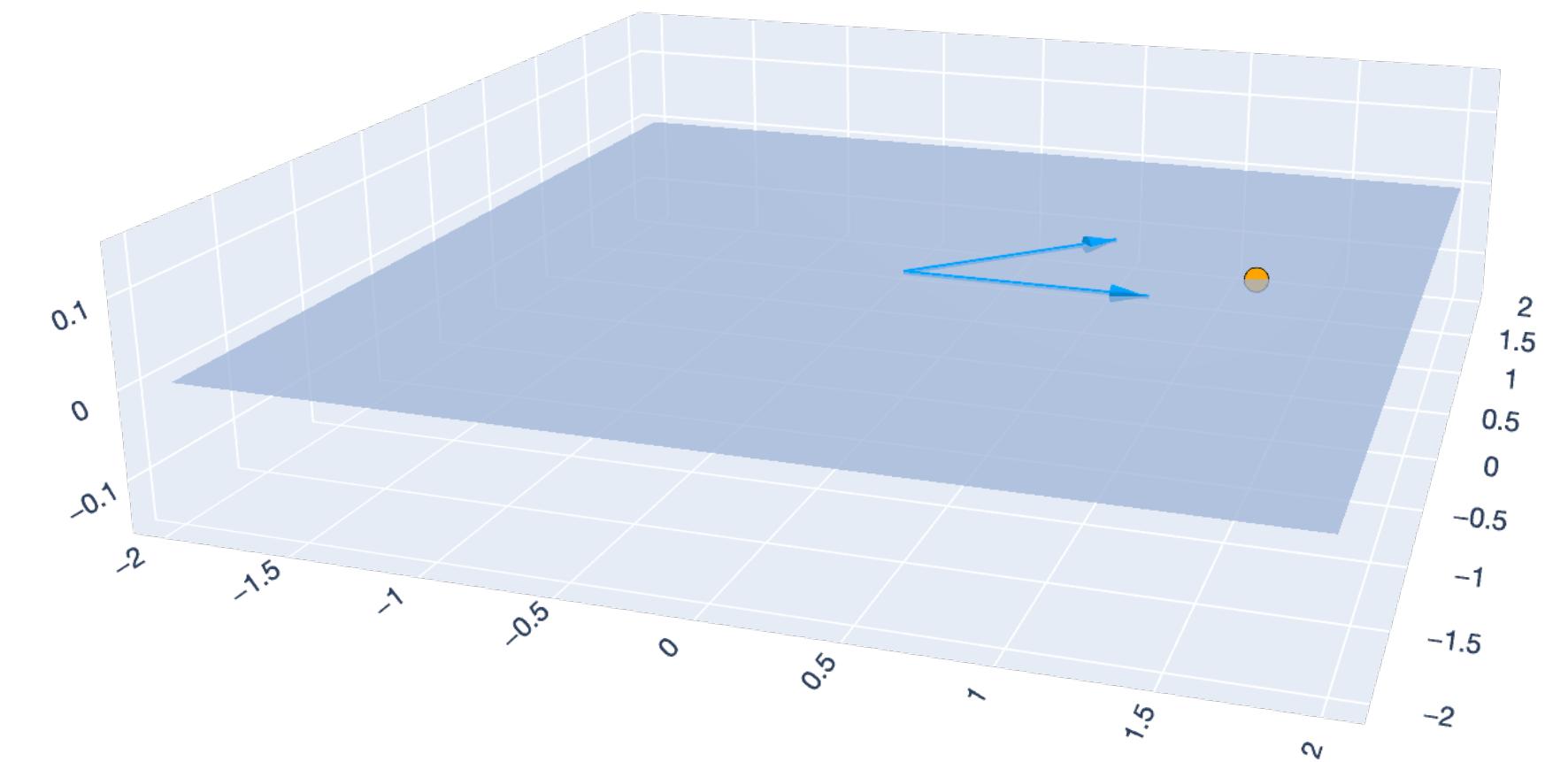
## Geometry of Least Squares

Let  $n = 3$  and  $d = 2$ . In this case  $\hat{\mathbf{y}} \in \mathbb{R}^3$  is a *linear combination* of columns  $\mathbf{x}_1$  and  $\mathbf{x}_2$ .

$$\hat{\mathbf{y}} = \mathbf{X}\mathbf{w} = w_1\mathbf{x}_1 + w_2\mathbf{x}_2 \in \mathbb{R}^3$$

Let  $\text{col}(\mathbf{X}) := \{\mathbf{x}_1, \dots, \mathbf{x}_d\}$  be the *columnspace* of  $\mathbf{X} \in \mathbb{R}^{n \times d}$ . Then,

$$\hat{\mathbf{y}} \in \text{span}(\text{col}(\mathbf{X})).$$



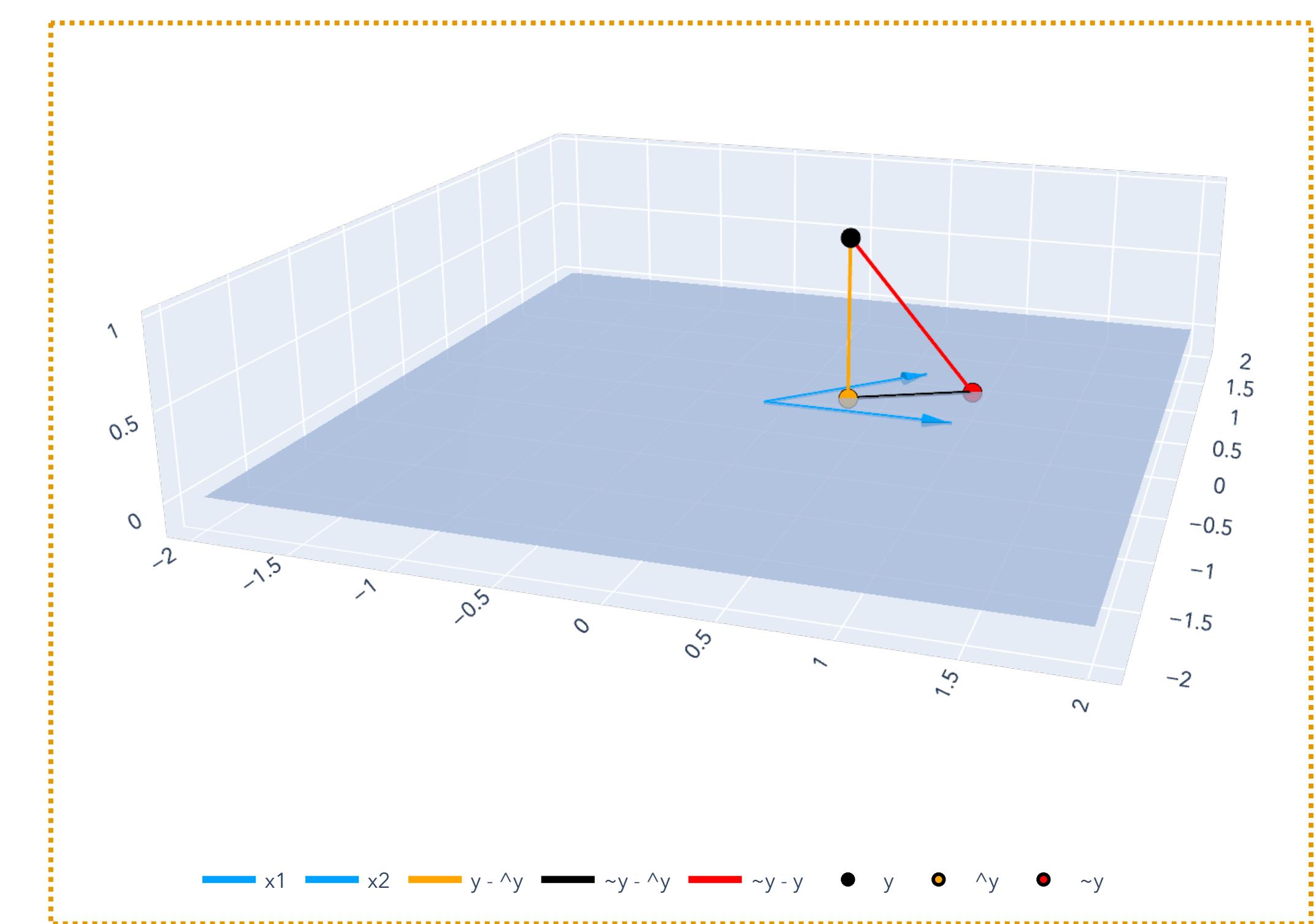
# Ordinary Least Squares

## Geometry of Least Squares

So,  $\hat{\mathbf{y}} = \mathbf{X}\mathbf{w} = w_1\mathbf{x}_1 + w_2\mathbf{x}_2 \in \mathbb{R}^3$ , which we can write as:  $\hat{\mathbf{y}} \in \text{span}(\text{col}(\mathbf{X}))$ .

The true labels  $\mathbf{y} \in \mathbb{R}^n$  might not be in  $\text{span}(\text{col}(\mathbf{X}))$ .

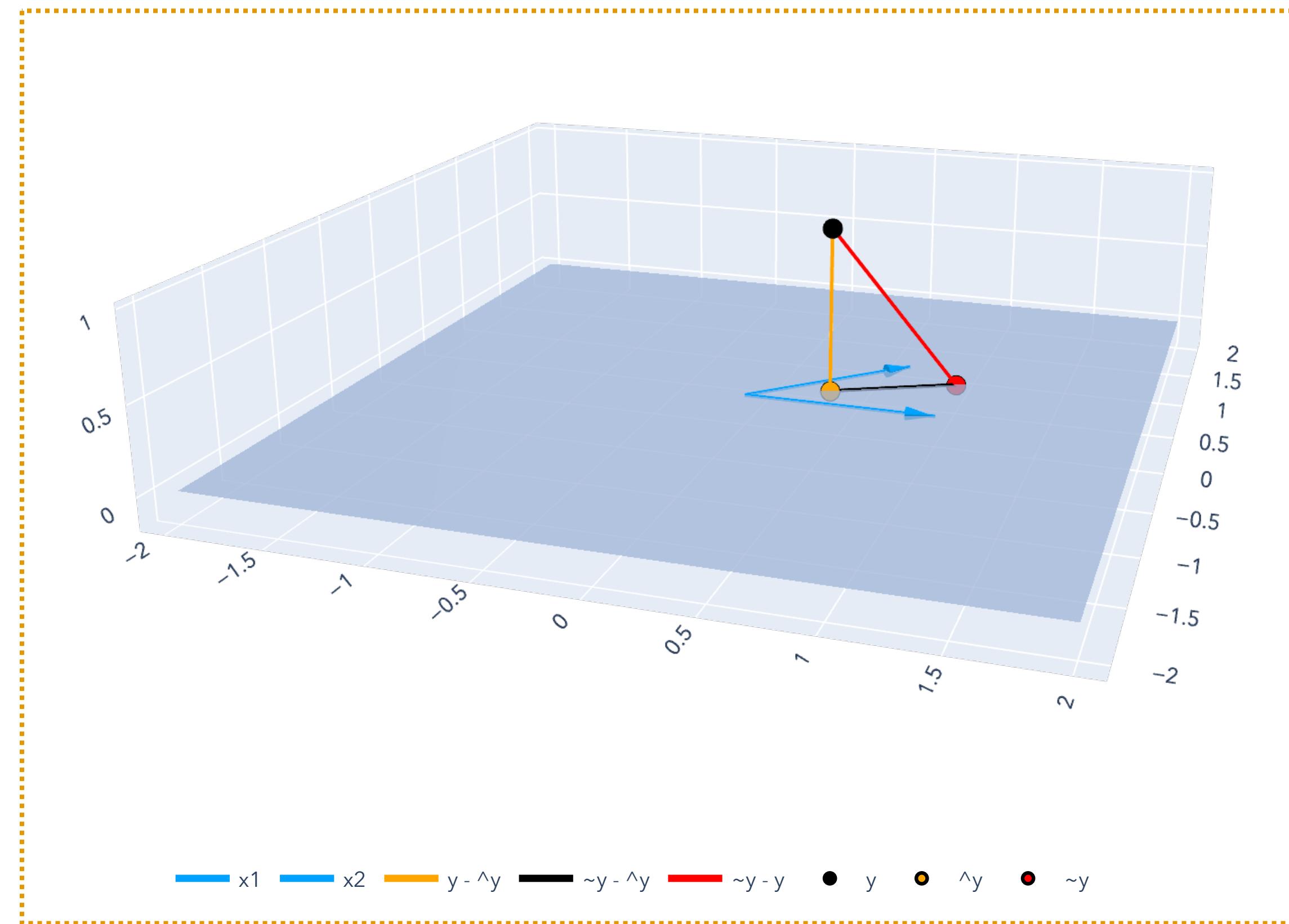
Goal: Find  $\mathbf{w} \in \mathbb{R}^n$  that minimizes  $\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$ .



# Ordinary Least Squares

## Geometry of Least Squares

Goal: Find  $\mathbf{w} \in \mathbb{R}^n$  that minimizes  $\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$ .

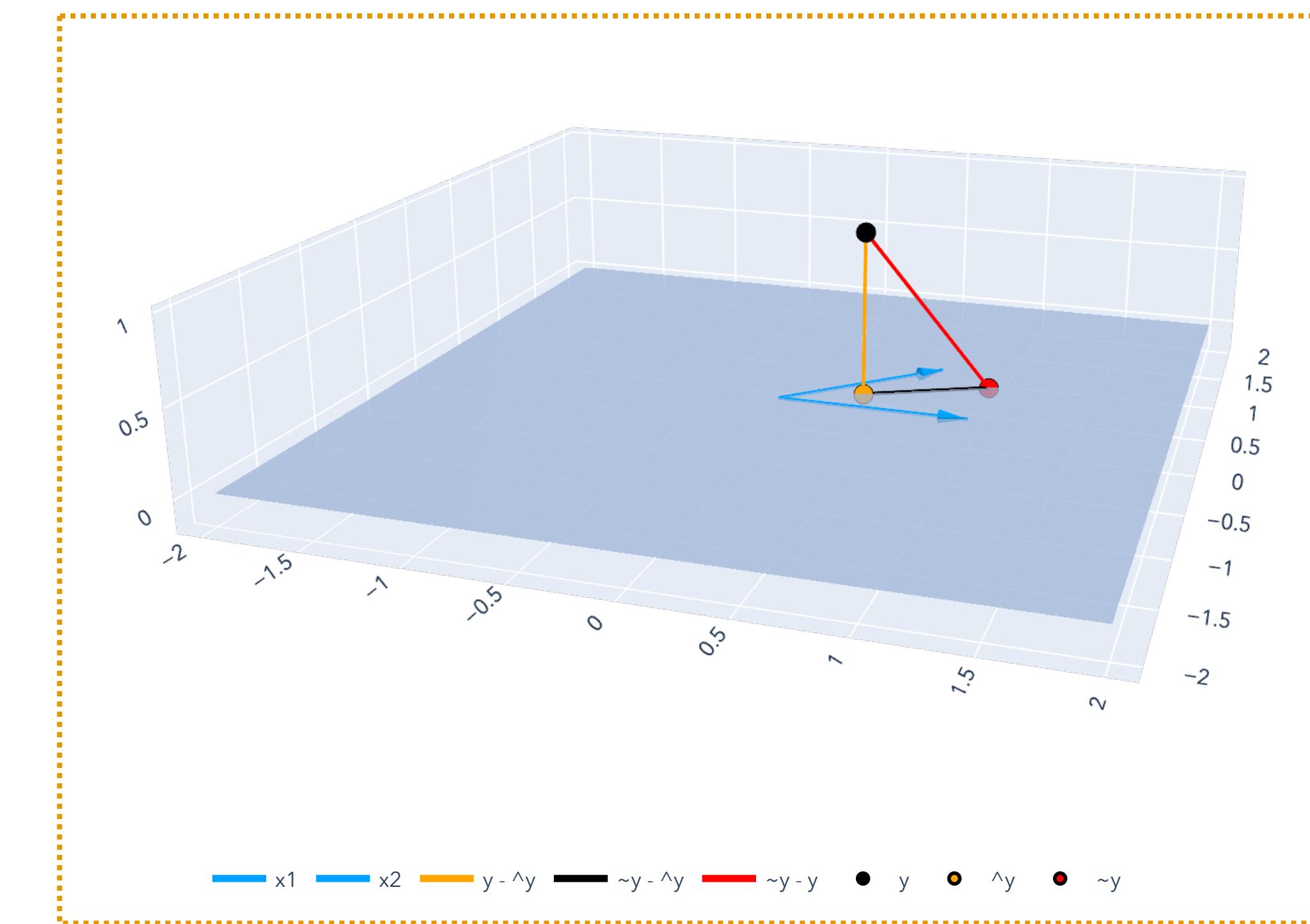


# Ordinary Least Squares

## Geometry of Least Squares

Goal: Find  $\mathbf{w} \in \mathbb{R}^n$  that minimizes  $\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$ .

Which point on  $\text{span}(\text{col}(\mathbf{X}))$  minimizes the distance from  $\mathbf{y}$  to  $\text{span}(\text{col}(\mathbf{X}))$ ?



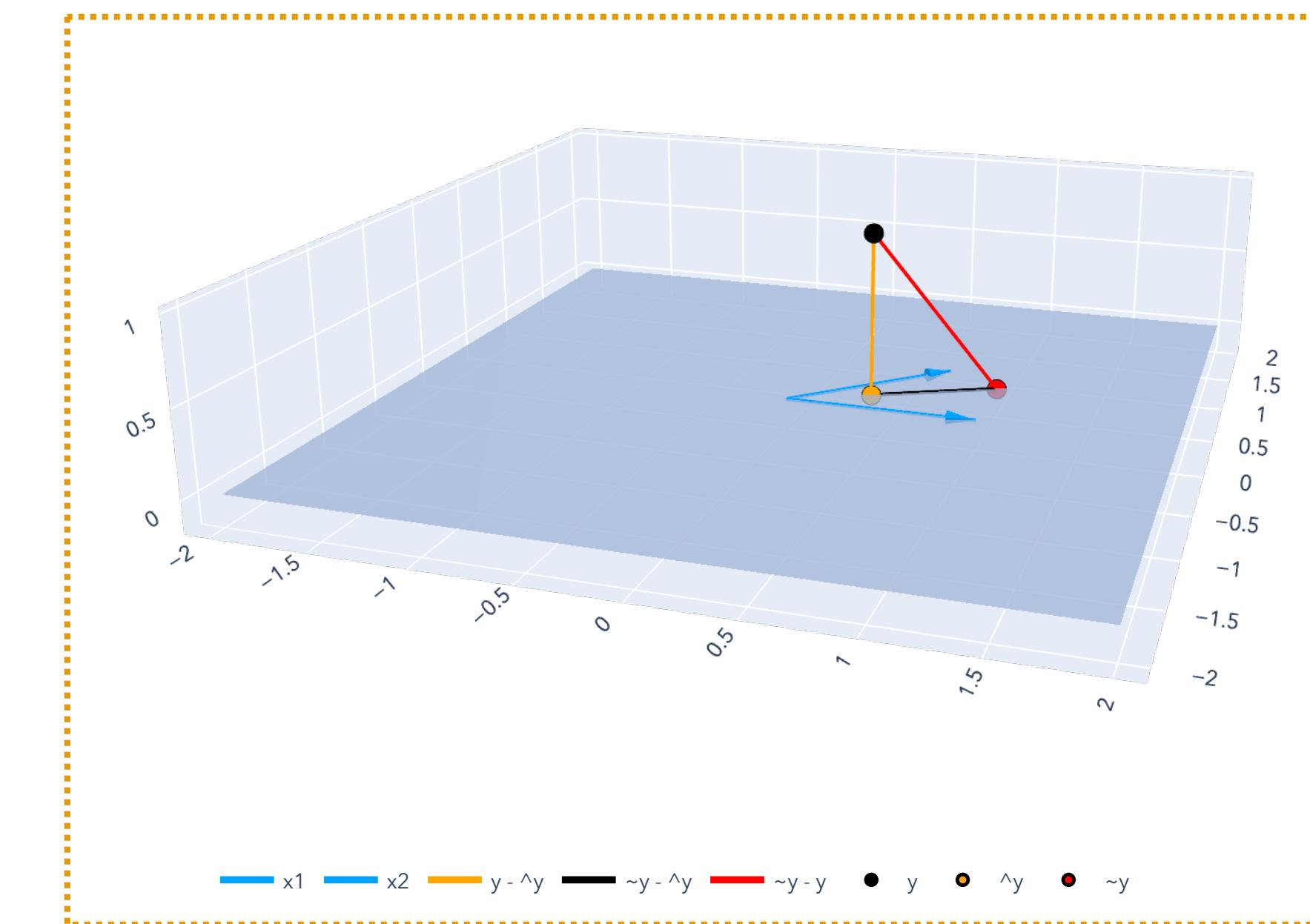
# Ordinary Least Squares

## Geometry of Least Squares

Goal: Find  $\mathbf{w} \in \mathbb{R}^n$  that minimizes  $\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$ .

*Which point on  $\text{span}(\text{col}(\mathbf{X}))$  minimizes the distance from  $\mathbf{y}$  to  $\text{span}(\text{col}(\mathbf{X}))$ ?*

*The point a perpendicular line down to  $\text{span}(\text{col}(\mathbf{X}))$ !*

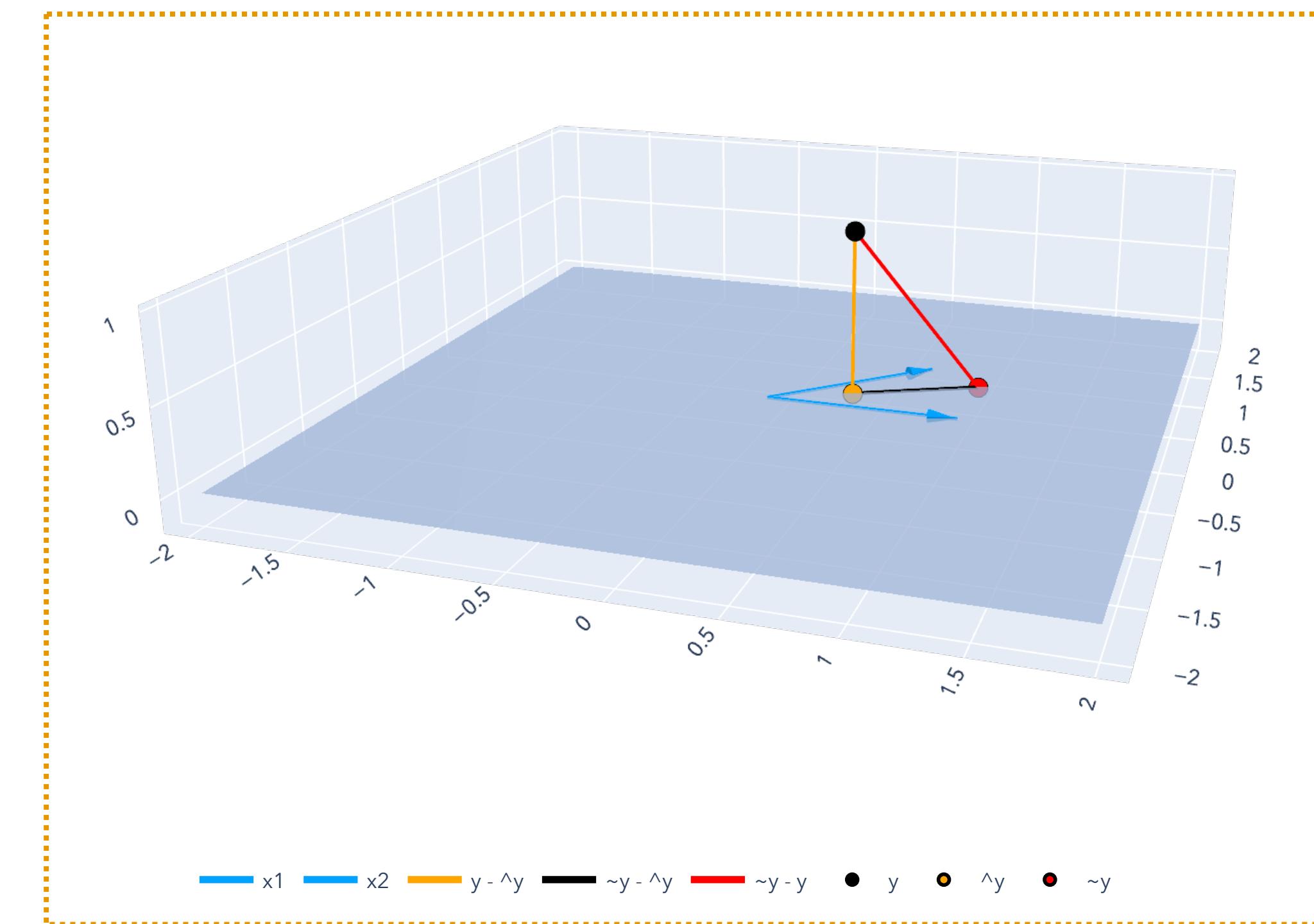


# Ordinary Least Squares

## Geometry of Least Squares

A projection of  $\mathbf{y} \in \mathbb{R}^n$  onto  $\text{span}(\text{col}(\mathbf{X}))$  gives us  $\hat{\mathbf{y}} \in \mathbb{R}^n$ , and  $\mathbf{X}\hat{\mathbf{w}} = \hat{\mathbf{y}}$ .

Let  $\tilde{\mathbf{y}} \in \mathbb{R}^n$  be any other vector in  $\text{span}(\text{col}(\mathbf{X}))$ , written  $\mathbf{X}\tilde{\mathbf{w}} = \tilde{\mathbf{y}}$ .

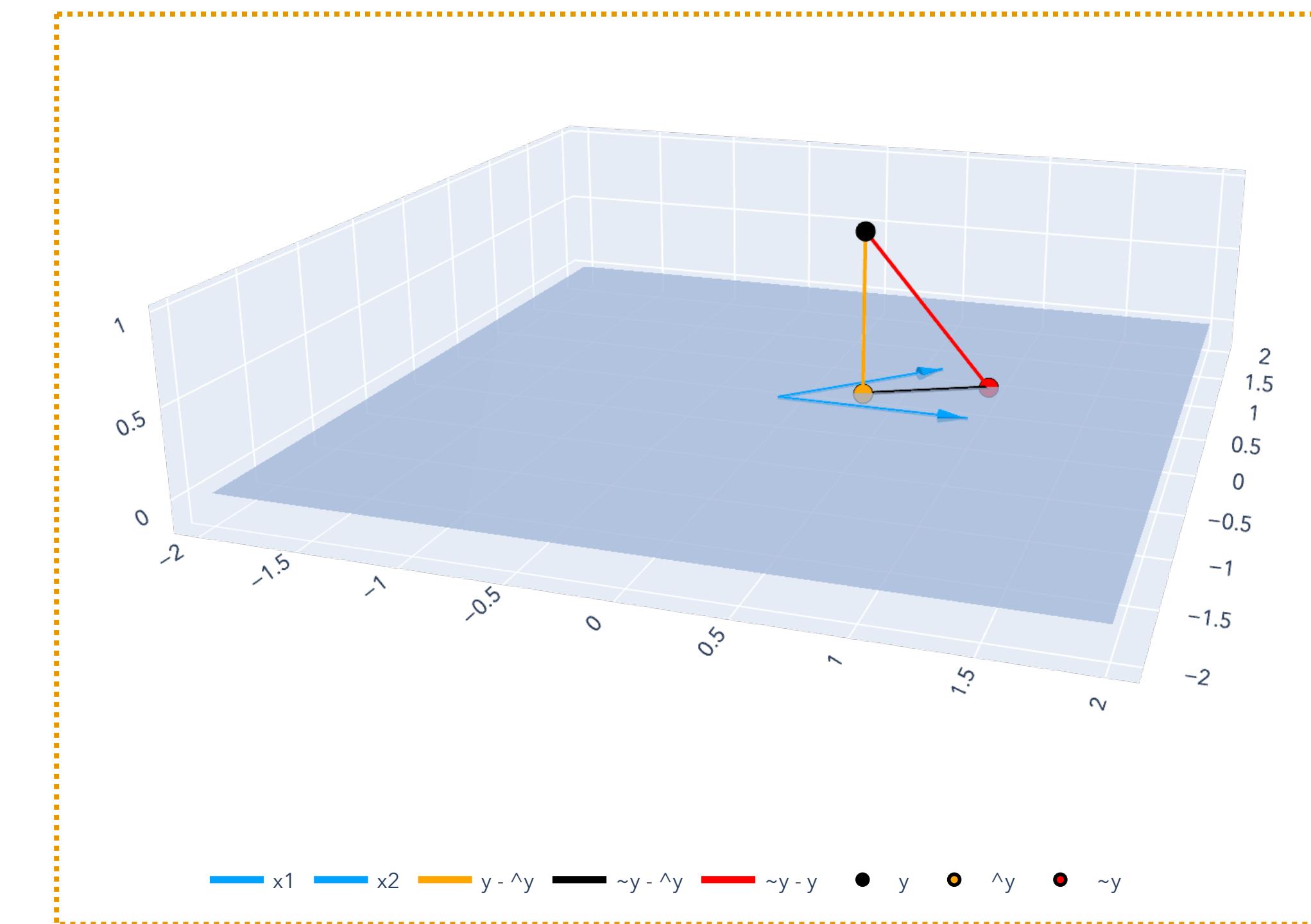


# Ordinary Least Squares

## Geometry of Least Squares

Let  $\hat{\mathbf{y}} = \mathbf{X}\hat{\mathbf{w}}$  be the projection of  $\mathbf{y}$ . Let  $\tilde{\mathbf{y}} = \mathbf{X}\tilde{\mathbf{w}}$  be any other  $\tilde{\mathbf{y}}$ .

The distances  $\|\mathbf{y} - \hat{\mathbf{y}}\|$  and  $\|\mathbf{y} - \tilde{\mathbf{y}}\|$  are the lengths of the residuals  $\|\hat{\mathbf{r}}\|$  and  $\|\tilde{\mathbf{r}}\|$ .



# Ordinary Least Squares

## Geometry of Least Squares

Let  $\tilde{\mathbf{y}} = \mathbf{X}\tilde{\mathbf{w}}$  be any other vector in  $\text{span}(\text{col}(\mathbf{X}))$ .

By the Pythagorean Theorem,

$$\|\hat{\mathbf{r}}\|^2 + \|\tilde{\mathbf{y}} - \hat{\mathbf{y}}\|^2 = \|\tilde{\mathbf{r}}\|^2.$$

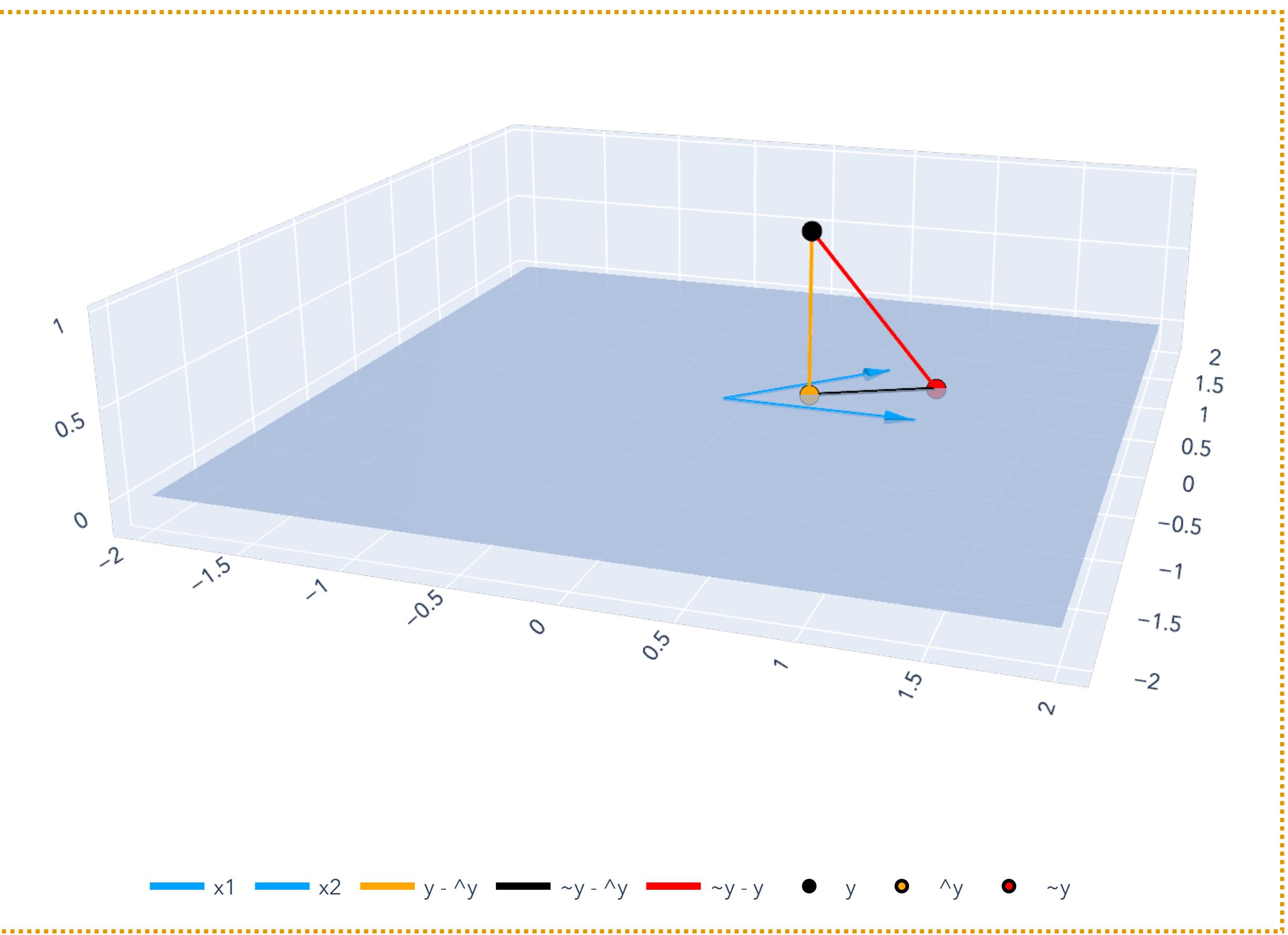
But  $\|\tilde{\mathbf{y}} - \hat{\mathbf{y}}\|^2 \geq 0$ , so:

$$\|\hat{\mathbf{r}}\|^2 \leq \|\tilde{\mathbf{r}}\|^2.$$

By definition,  $\hat{\mathbf{r}} = \mathbf{X}\hat{\mathbf{w}} - \mathbf{y}$  and  $\tilde{\mathbf{r}} = \mathbf{X}\tilde{\mathbf{w}} - \mathbf{y}$ .

Therefore,

$$\|\mathbf{X}\hat{\mathbf{w}} - \mathbf{y}\|^2 \leq \|\mathbf{X}\tilde{\mathbf{w}} - \mathbf{y}\|^2.$$



# Ordinary Least Squares

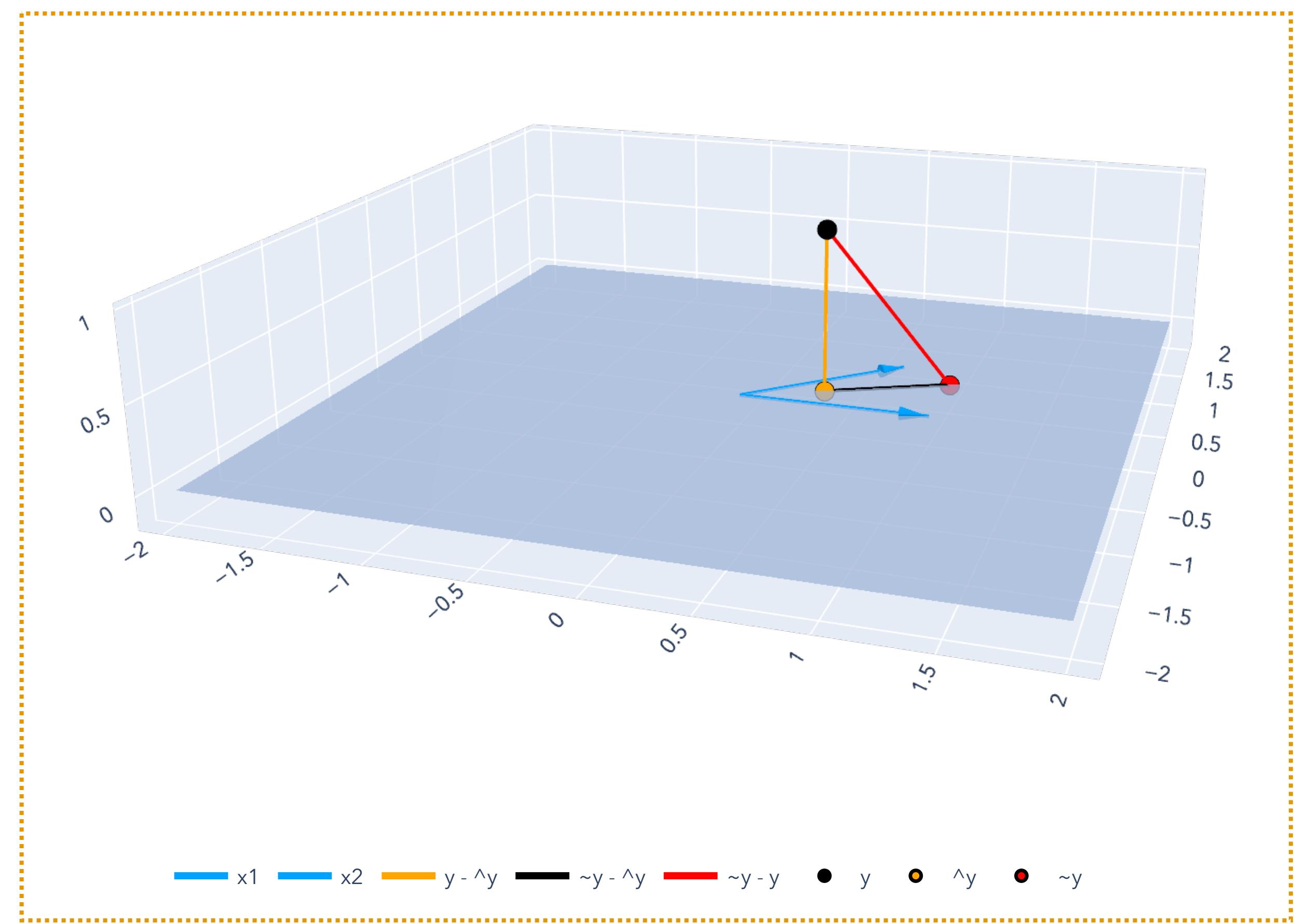
## Geometry of Least Squares

Therefore:

$$\|\mathbf{X}\hat{\mathbf{w}} - \mathbf{y}\|^2 \leq \|\mathbf{X}\tilde{\mathbf{w}} - \mathbf{y}\|^2,$$

where  $\hat{\mathbf{w}} \in \mathbb{R}^d$  is obtained from the *projection*  $\hat{\mathbf{y}}$  of  $\mathbf{y} \in \mathbb{R}^d$  onto  $\text{span}(\text{col}(\mathbf{X}))$ , and  $\tilde{\mathbf{w}} \in \mathbb{R}^d$  is any other vector.

*But what is  $\hat{\mathbf{w}}$ ?*



# Orthogonality

## Definition

Two vectors  $\mathbf{x} \in \mathbb{R}^n$  and  $\mathbf{y} \in \mathbb{R}^n$  are orthogonal if

$$\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^\top \mathbf{y} = 0.$$

So, if a vector  $\mathbf{v} \in \mathbb{R}^n$  is orthogonal to a whole set of vectors  $\{\mathbf{x}_1, \dots, \mathbf{x}_d\}$ , we can write this in matrix form.

$$\mathbf{X} = \begin{bmatrix} \uparrow & & \uparrow \\ \mathbf{x}_1 & \dots & \mathbf{x}_d \\ \downarrow & & \downarrow \end{bmatrix}$$
$$\mathbf{X}^\top \mathbf{v} = \mathbf{0}.$$

# Ordinary Least Squares

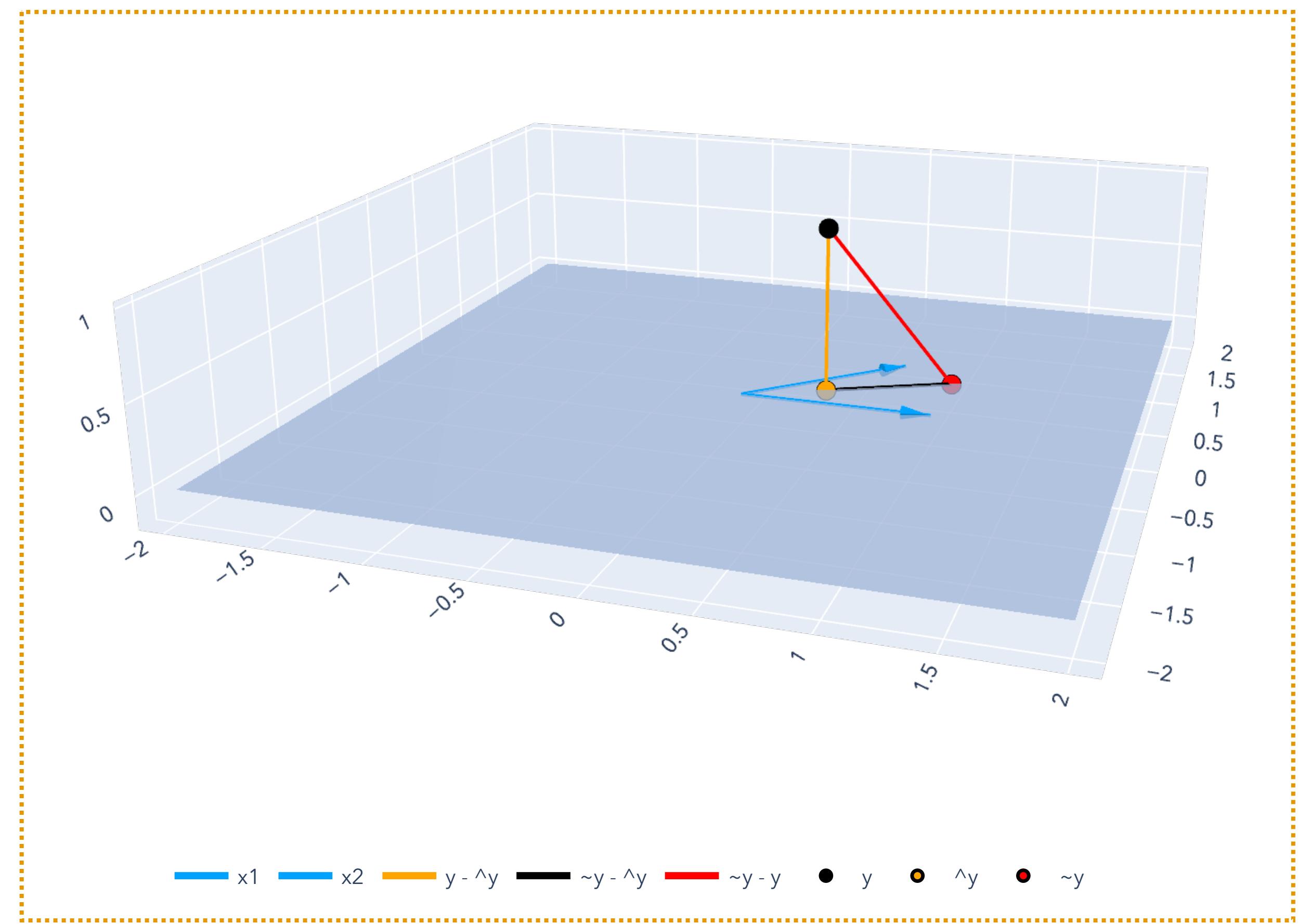
## The Normal Equations

From the picture,  $\hat{\mathbf{r}} = \mathbf{X}\hat{\mathbf{w}} - \mathbf{y}$  is *orthogonal* to  $\text{span}(\text{col}(\mathbf{X}))$ :

$$\mathbf{X}^T \hat{\mathbf{r}} = \mathbf{0} \implies \mathbf{X}^T (\mathbf{X}\hat{\mathbf{w}} - \mathbf{y}) = \mathbf{0}.$$

This gives us the normal equations:

$$\mathbf{X}^T \mathbf{y} = \mathbf{X}^T \mathbf{X} \hat{\mathbf{w}}.$$



# Ordinary Least Squares

## The Normal Equations

Finally, we need to solve the normal equations:

$$\underbrace{\mathbf{X}^\top \mathbf{y}}_{\mathbb{R}^d} = \underbrace{\mathbf{X}^\top \mathbf{X}}_{\mathbb{R}^{d \times d}} \underbrace{\hat{\mathbf{w}}}_{\mathbb{R}^d}.$$

# Linear Independence

## Idea

A collection of vectors  $\mathbf{a}_1, \dots, \mathbf{a}_d \in \mathbb{R}^n$  is linearly independent if there are no redundancies – no vector  $\mathbf{a}_i$  can be written as a linear combination of the others.

# Linear Independence

## Definition

A collection of vectors  $\mathbf{a}_1, \dots, \mathbf{a}_d \in \mathbb{R}^n$  is linearly independent if  $\alpha_1\mathbf{a}_1 + \dots + \alpha_d\mathbf{a}_d = \mathbf{0}$  if and only if  $\alpha_i = 0$  for all  $i \in [d]$ .

Equivalently, there exists  $\mathbf{a}_i$  that can be written in terms of the others:

$$\mathbf{a}_i = \alpha_1\mathbf{a}_1 + \dots + \alpha_{i-1}\mathbf{a}_{i-1} + \alpha_{i+1}\mathbf{a}_{i+1} + \dots + \alpha_d\mathbf{a}_d.$$

# Linear Independence

## Examples

$\left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 2 \\ 2 \\ 0 \end{bmatrix} \right\}$  is *not* linearly independent.

$\left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \right\}$  is linearly independent.

$\left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix} \right\}$  is linearly independent.

# Rank

## Definition

**Rank** is the number of linearly independent columns in a matrix. This is always the same as the number of linearly independent rows in a matrix.

For  $\mathbf{A} \in \mathbb{R}^{n \times d}$ , it is always the case that:  $\text{rank}(\mathbf{A}) \leq \min\{n, d\}$ .

If  $\text{rank}(\mathbf{A}) = \min\{n, d\}$ , then we say  $\mathbf{A}$  is *full rank*.

# Remember this?



# Ordinary Least Squares

## The Normal Equations

Finally, we need to solve the normal equations:

$$\underbrace{\mathbf{X}^\top \mathbf{y}}_{\mathbb{R}^d} = \underbrace{\mathbf{X}^\top \mathbf{X}}_{\mathbb{R}^{d \times d}} \underbrace{\hat{\mathbf{w}}}_{\mathbb{R}^d}.$$

For  $\mathbf{X} \in \mathbb{R}^{n \times d}$ , if  $n \geq d$  and  $\text{rank}(\mathbf{X}) = d$ , then:  $\text{rank}(\mathbf{X}^\top \mathbf{X}) = d \iff \mathbf{X}^\top \mathbf{X}$  has  $d$  linearly independent columns  $\iff (\mathbf{X}^\top \mathbf{X})^{-1}$  exists.

# Ordinary Least Squares

## The Normal Equations

$$\underbrace{\mathbf{X}^\top \mathbf{y}}_{\mathbb{R}^d} = \underbrace{\mathbf{X}^\top \mathbf{X}}_{\mathbb{R}^{d \times d}} \underbrace{\hat{\mathbf{w}}}_{\mathbb{R}^d}.$$

Finally, solving the normal equations:

$$\hat{\mathbf{w}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$$

# Ordinary Least Squares

## Main Theorem

Let  $\mathbf{X} \in \mathbb{R}^{n \times d}$  with  $n \geq d$  and  $\text{rank}(\mathbf{X}) = d$  (the columns of  $\mathbf{X}$  are linearly independent).

Then, the solution  $\hat{\mathbf{w}} \in \mathbb{R}^d$  that minimizes  $\|\mathbf{X}\mathbf{w} - \mathbf{y}\|$ , i.e.

$$\|\mathbf{X}\hat{\mathbf{w}} - \mathbf{y}\| \leq \|\mathbf{X}\mathbf{w} - \mathbf{y}\| \text{ for all } \mathbf{w} \in \mathbb{R}^d,$$

is given by:

$$\hat{\mathbf{w}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}.$$

# Recap

# Lesson Overview

## Takeaways

**Regression.** The basic problem in machine learning is regression. We have *training data* in the form of a data matrix  $\mathbf{X} \in \mathbb{R}^{n \times d}$  and labels  $\mathbf{y} \in \mathbb{R}^n$ . We seek a model  $\hat{\mathbf{w}} \in \mathbb{R}^d$  such that  $\mathbf{X}\hat{\mathbf{w}} \approx \mathbf{y}$ .

**Least squares.** One way to find a model for the data is through *least squares*: choose  $\hat{\mathbf{w}}$  that minimizes  $\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$ .

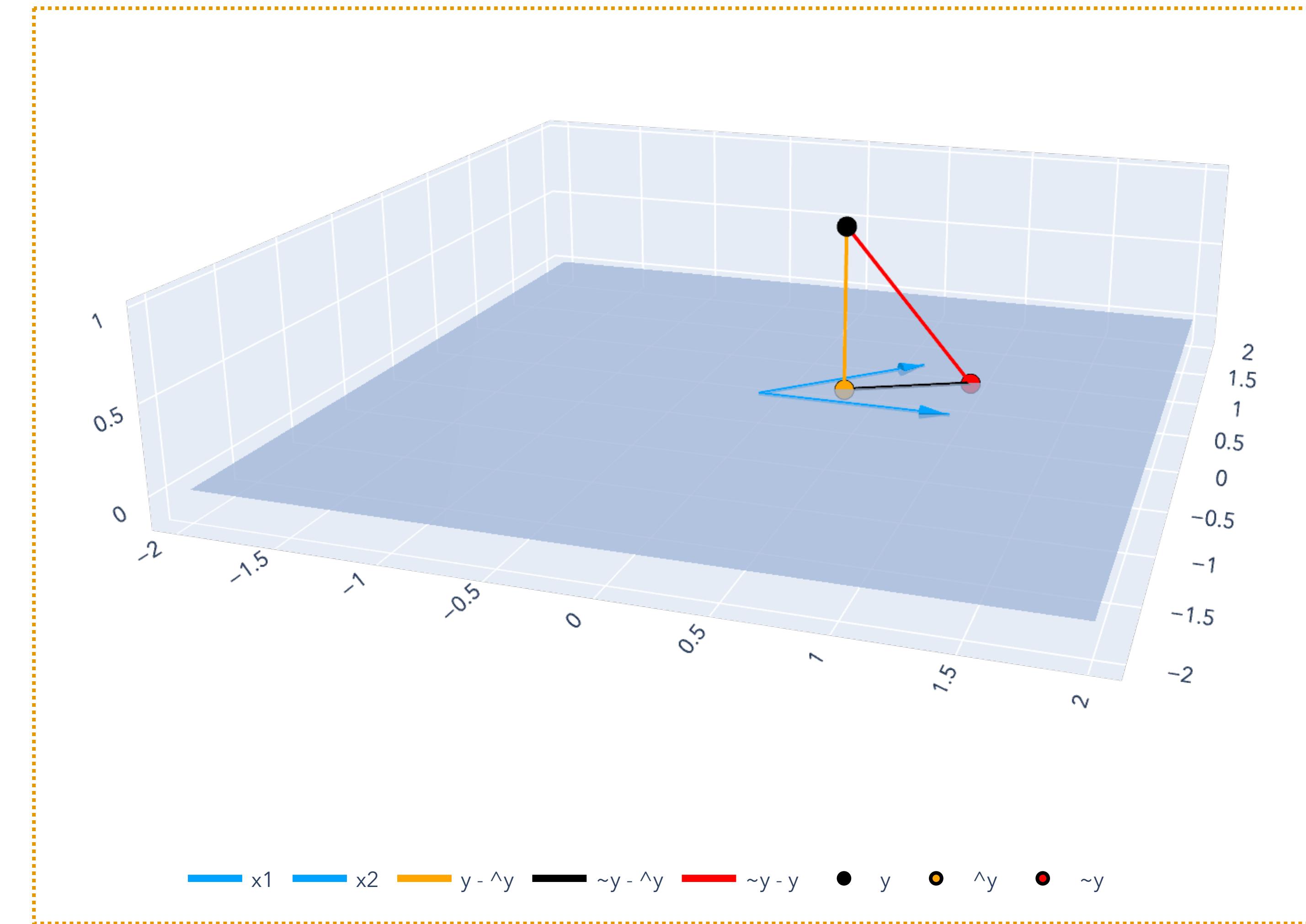
**Span and orthogonality.** We can solve least squares by noticing that  $\mathbf{X}\hat{\mathbf{w}} - \mathbf{y}$  is *orthogonal* to  $\text{span}(\text{cols}(\mathbf{X}))$ . This gives us the normal equations:  $\mathbf{X}^\top \mathbf{X}\hat{\mathbf{w}} = \mathbf{X}^\top \mathbf{y}$ .

**Linear independence.** To solve the normal equations, we need  $\mathbf{X}$  to be full *rank* (its  $d$  columns are *linearly independent*). Then, we can invert and solve the normal equations.

$$\hat{\mathbf{w}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}.$$

# Lesson Overview

## Big Picture: Least Squares



# Lesson Overview

## Big Picture: Gradient Descent

