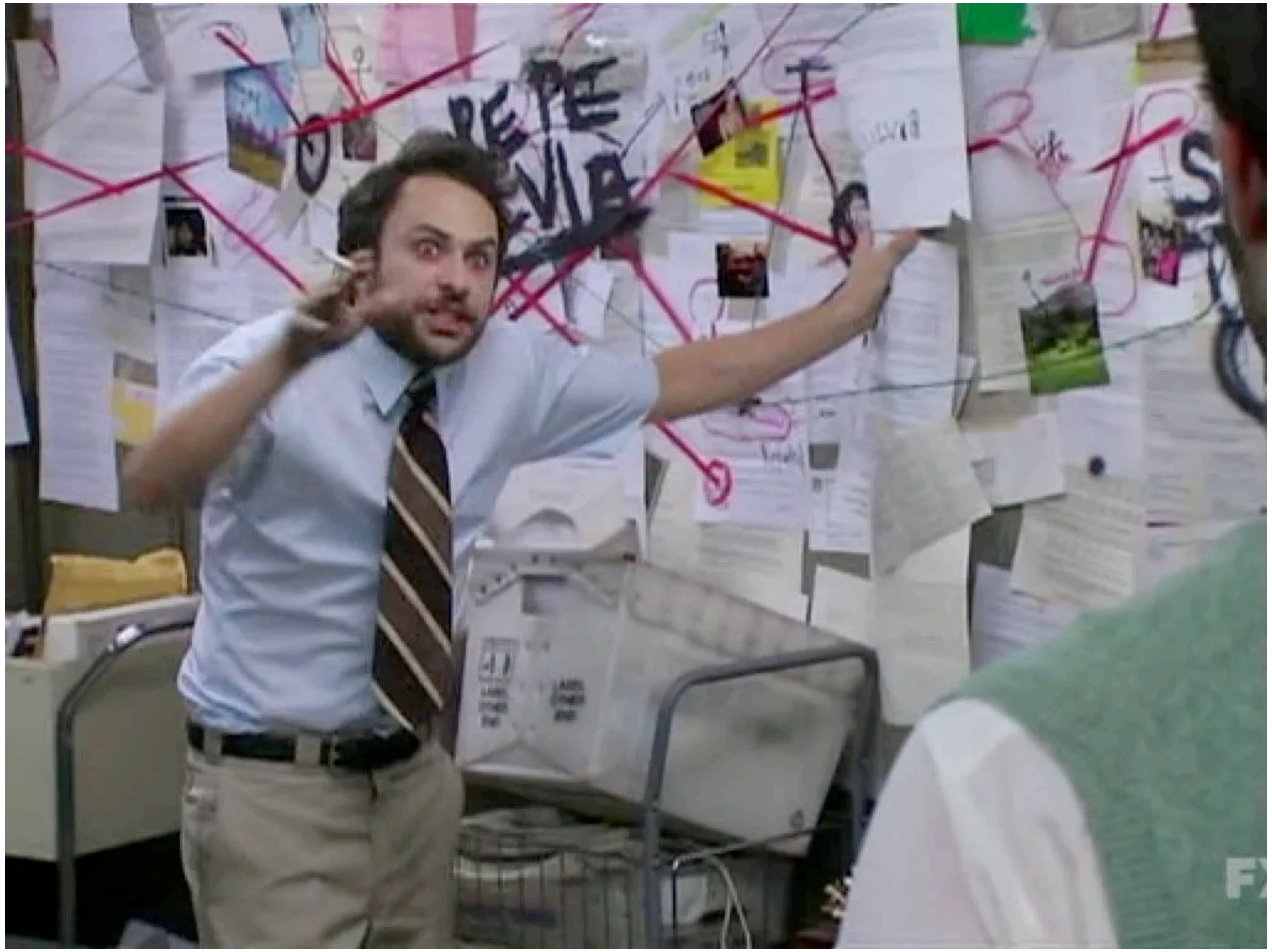
Math for ML

Finale: Course Overview

By: Samuel Deng

Lesson Overview



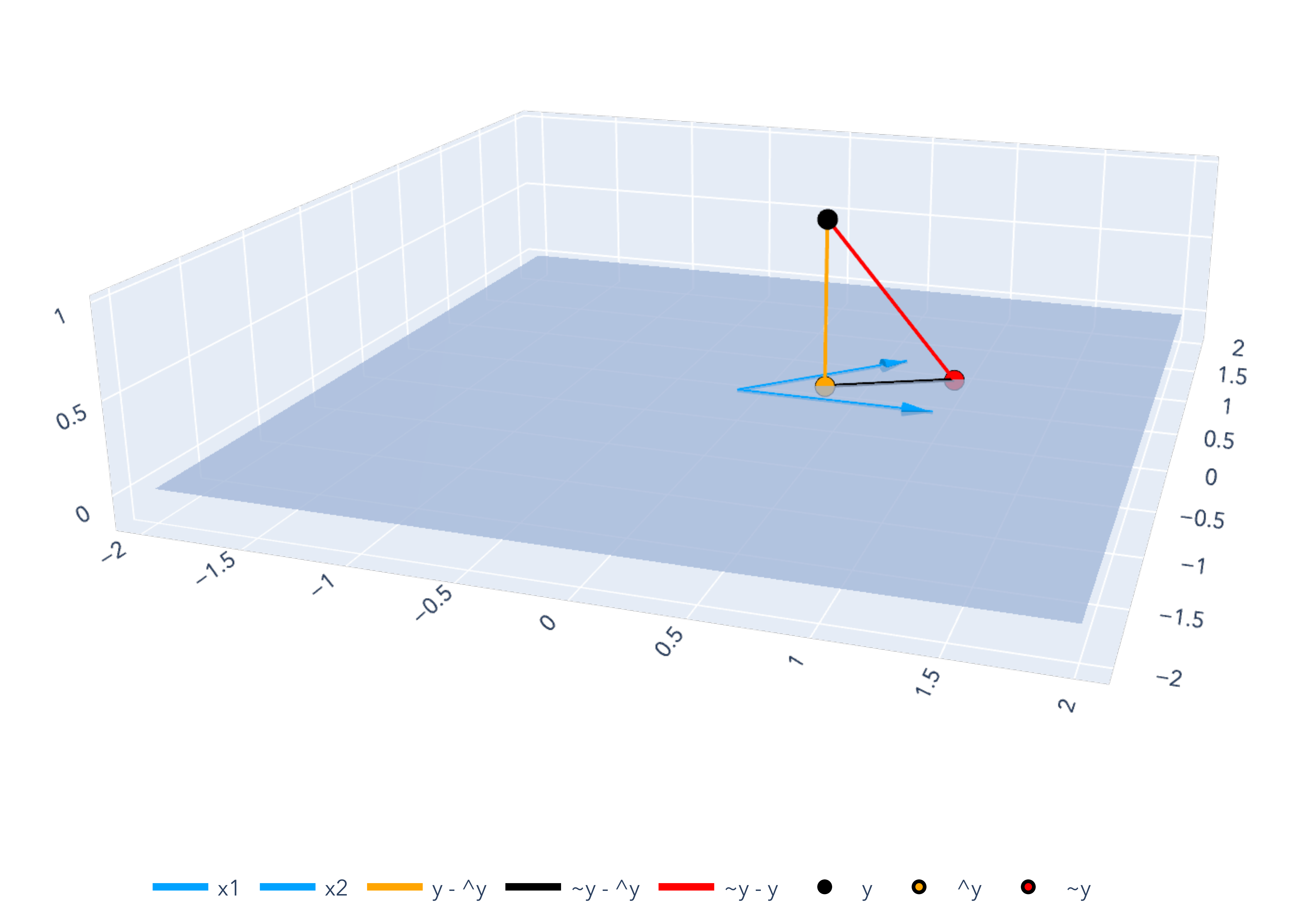
Week 1.1

Vectors, matrices, and least squares regression

Vectors, matrices, and least squares regression

Big Picture: Least Squares

Linear independence, span, and rank allowed us to get $(\mathbf{X}^\top\mathbf{X})^{-1}$ from $\text{rank}(\mathbf{X}^\top\mathbf{X}) = \text{rank}(\mathbf{X})$ sketching our first OLS theorem:



Theorem (OLS solution). If $n \geq d$ and $\text{rank}(\mathbf{X}) = d$, then: $\hat{\mathbf{w}} = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{y}$.

Vectors, matrices, and least squares regression

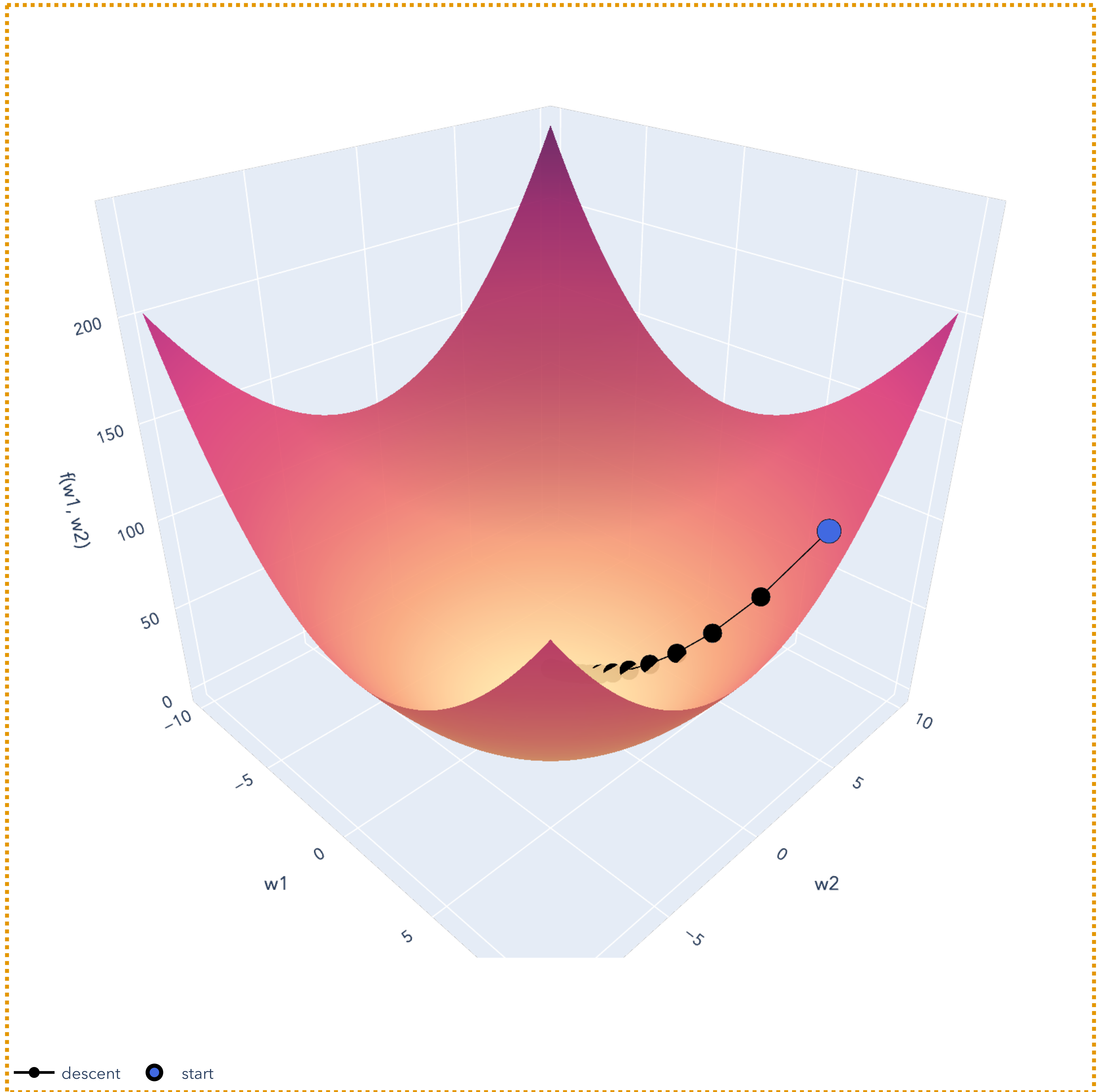
Big Picture: Gradient Descent

Using norm to rewrite the sum of squared residual errors,

$$f(\mathbf{w}) = \sum_{i=1}^{n} (\mathbf{w}^\top \mathbf{x}_i - y_i)^2$$

we got a function measuring how "badly" each $\mathbf{w}$ does:

$$f(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2.$$



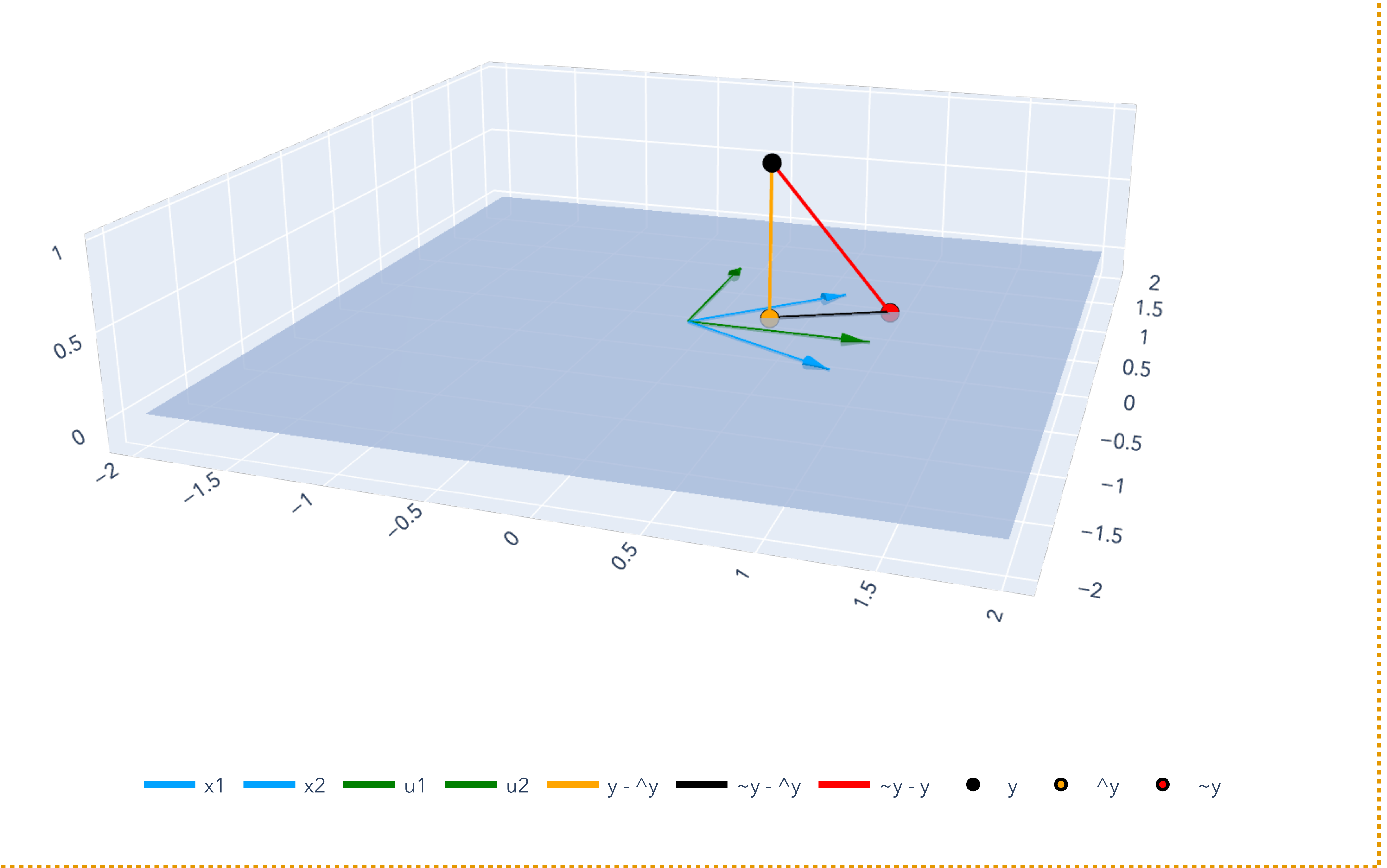
Week 1.2

Bases, subspaces, and orthogonality

Bases, subspaces, and orthogonality

Big Picture: Least Squares

We formally defined subspace, a basis, the columnspace, and orthogonal basis. This filled in the gaps to get **Theorem (invertibility of $\mathbf{X}^\top\mathbf{X}$)** and **Theorem (Pythagorean Theorem).**



Using our new notion of orthogonality, we could simplify the OLS solution if we had an ONB.

Theorem (OLS solution with ONB). If $n \geq d$ and $\mathrm{rank}(\mathbf{X}) = d$ and $\mathbf{U} \in \mathbb{R}^{d \times d}$ an ONB: $\hat{\mathbf{w}} = \mathbf{U}^\top \mathbf{y}$.

Bases, subspaces, and orthogonality

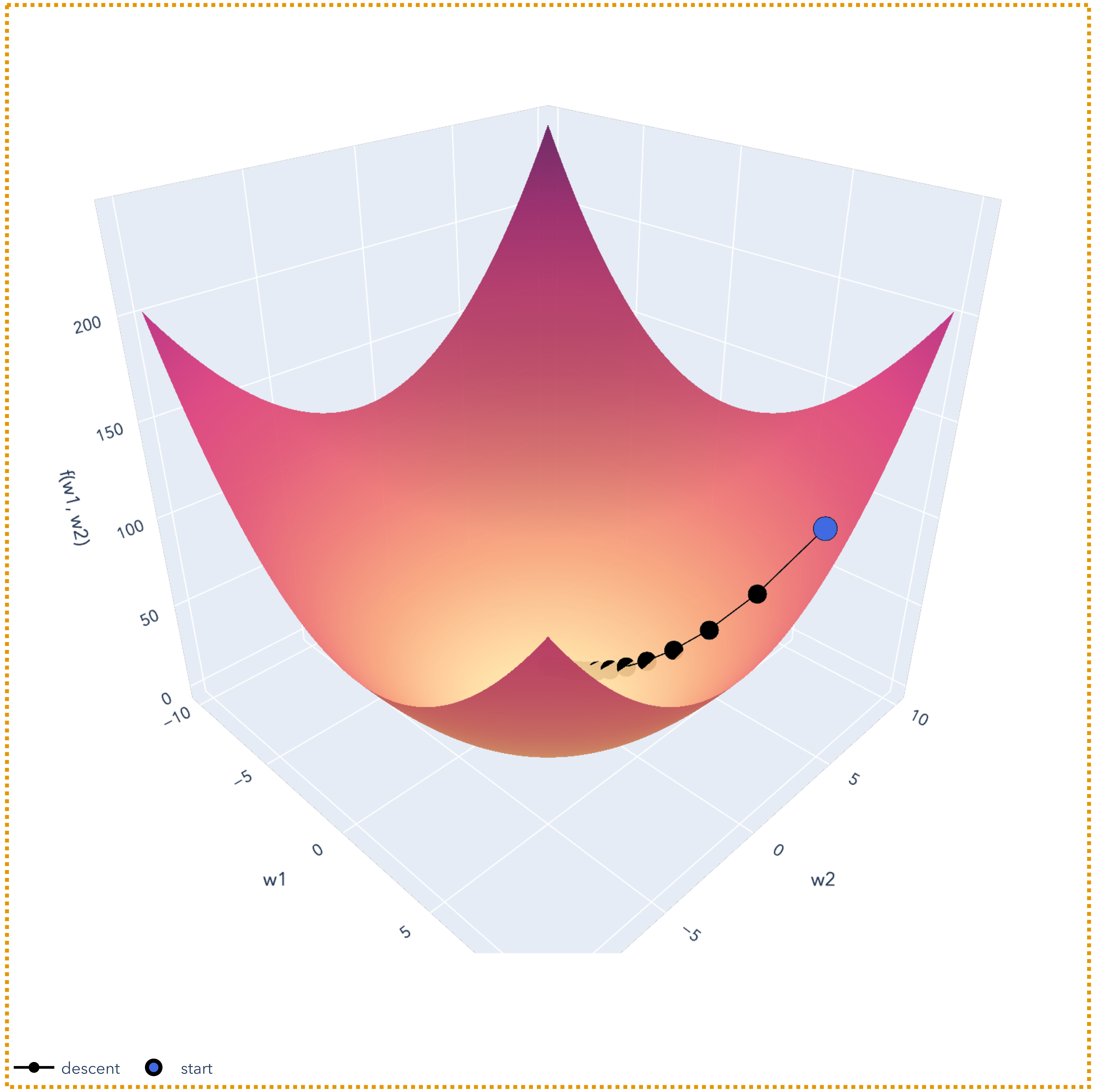
Big Picture: Gradient Descent

Using norm to rewrite the sum of squared residual errors,

$$f(\mathbf{w}) = \sum_{i=1}^{n} (\mathbf{w}^\top \mathbf{x}_i - y_i)^2$$

we got a function measuring how "badly" each $\mathbf{w}$ does:

$$f(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2.$$

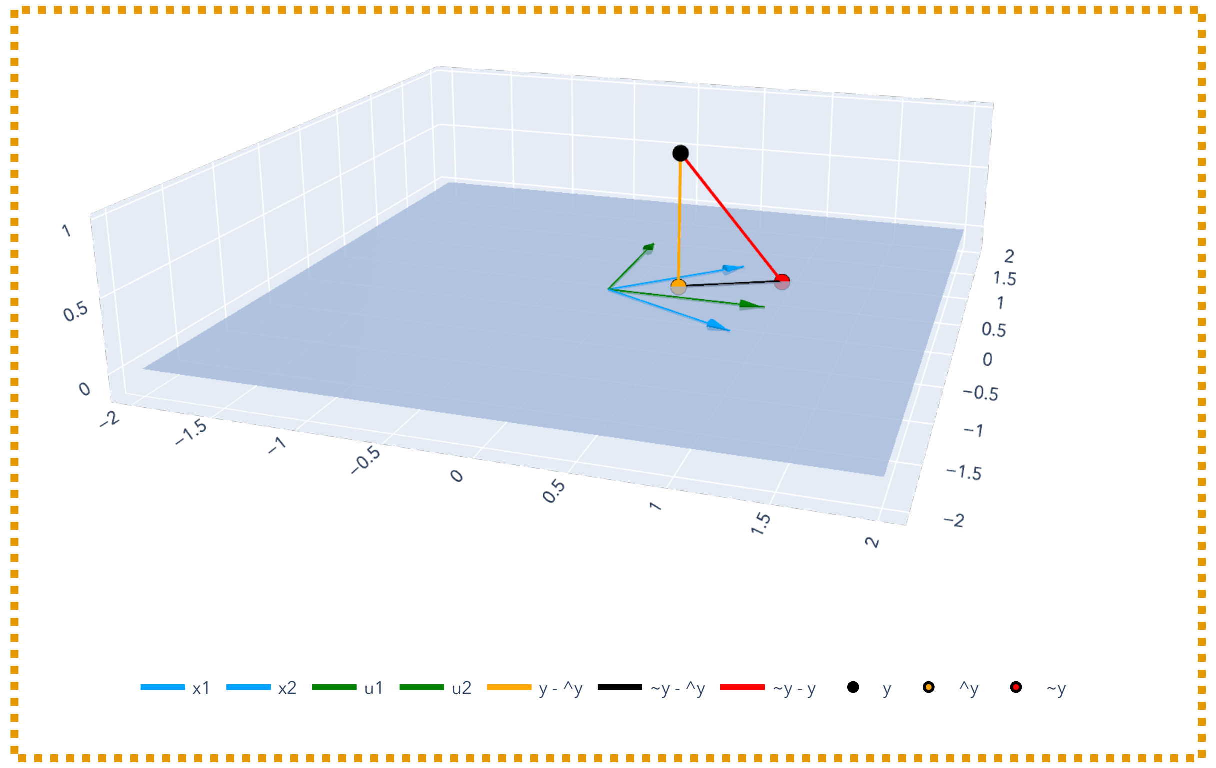


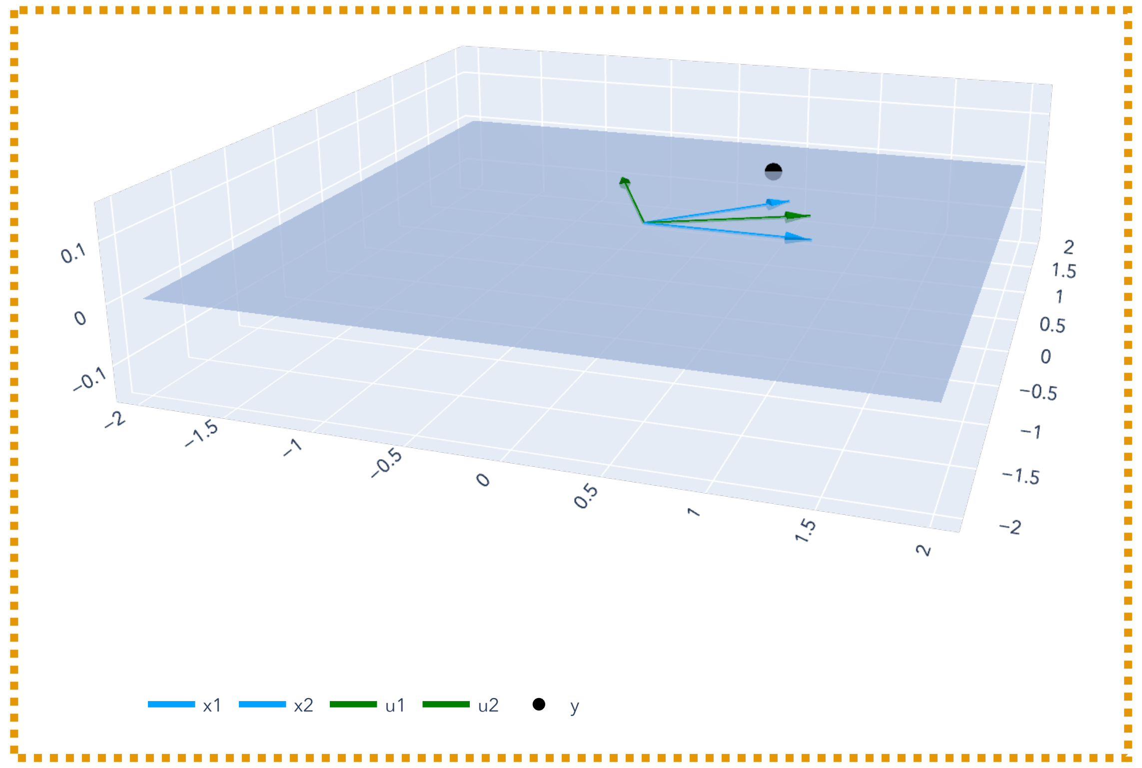
Week 2.1

Singular Value Decomposition

Singular Value Decomposition

Big Picture: Least Squares





We defined orthogonal complements and projection matrices to solve the best-fitting 1D subspace problem, leading to SVD:

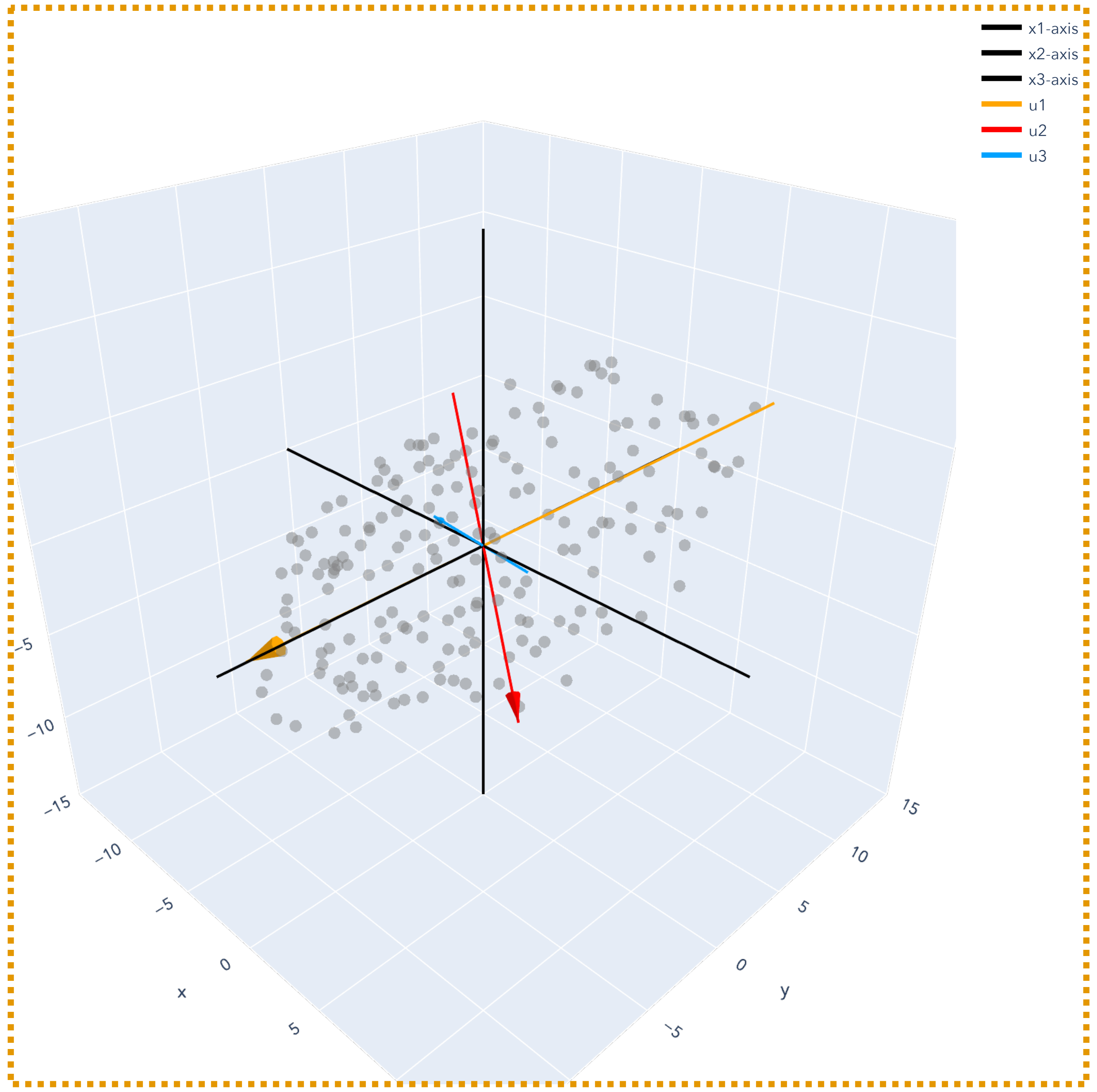
$$\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top$$

The SVD defined the pseudoinverse which unified OLS:

Theorem (OLS solution with pseudoinverse). Let $\mathbf{X} \in \mathbb{R}^{n \times d}$ have pseudoinverse $\mathbf{X}^+ \in \mathbb{R}^{d \times n}$. Then: $\hat{\mathbf{w}} = \mathbf{X}^+\mathbf{y}$.

If $n \geq d$, then $\hat{\mathbf{w}}$ minimizes $\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$.

If $d > n$, then $\hat{\mathbf{w}}$ is the exact solution $\mathbf{X}\hat{\mathbf{w}} = \mathbf{y}$ with min. norm.



Singular Value Decomposition

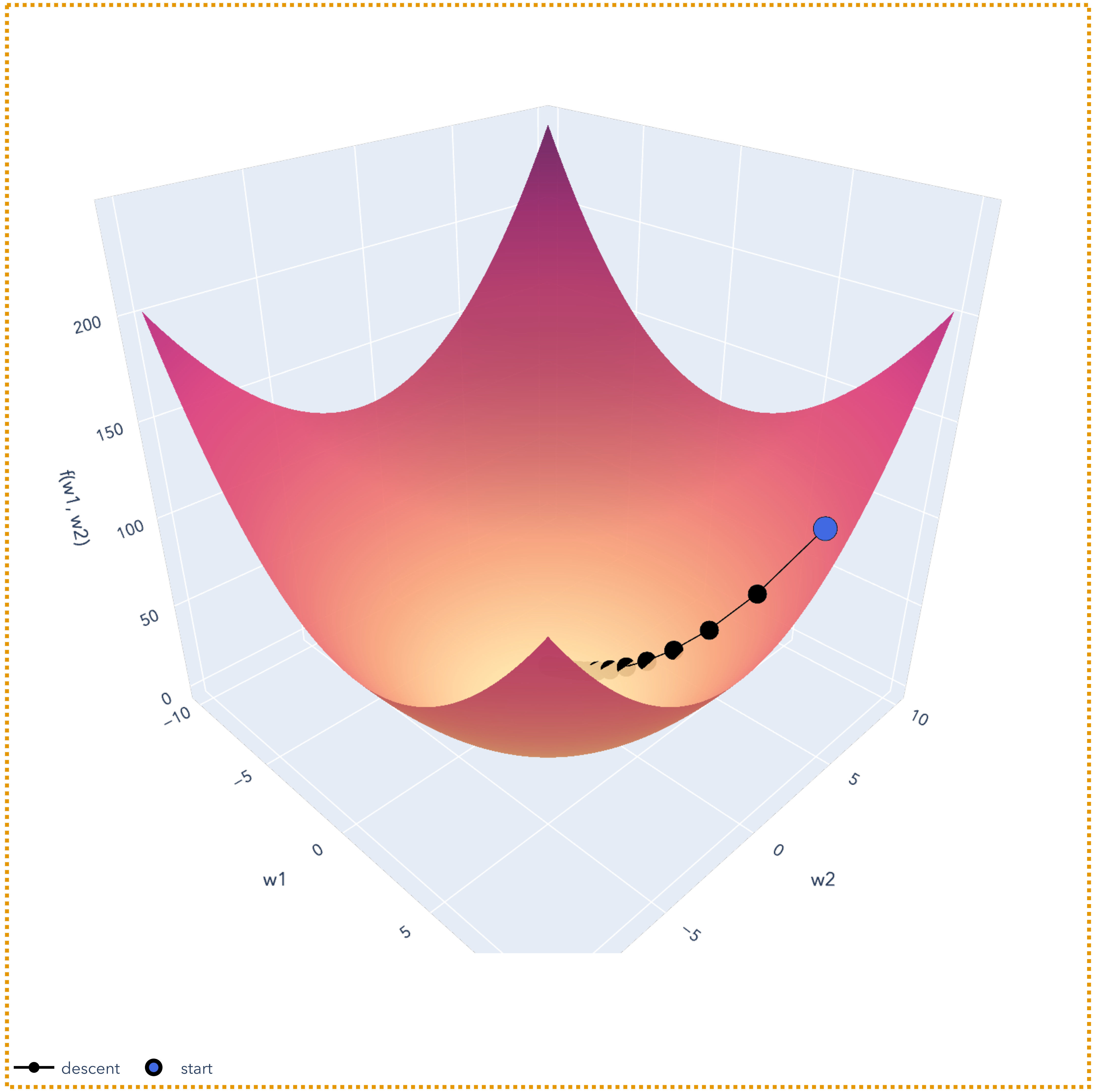
Big Picture: Gradient Descent

Using norm to rewrite the sum of squared residual errors,

$$f(\mathbf{w}) = \sum_{i=1}^{n} (\mathbf{w}^\top \mathbf{x}_i - y_i)^2$$

we got a function measuring how "badly" each $\mathbf{w}$ does:

$$f(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2.$$



Week 2.2

Eigendecomposition and PSD Matrices

Eigendecomposition and PSD Matrices

Big Picture: Least Squares

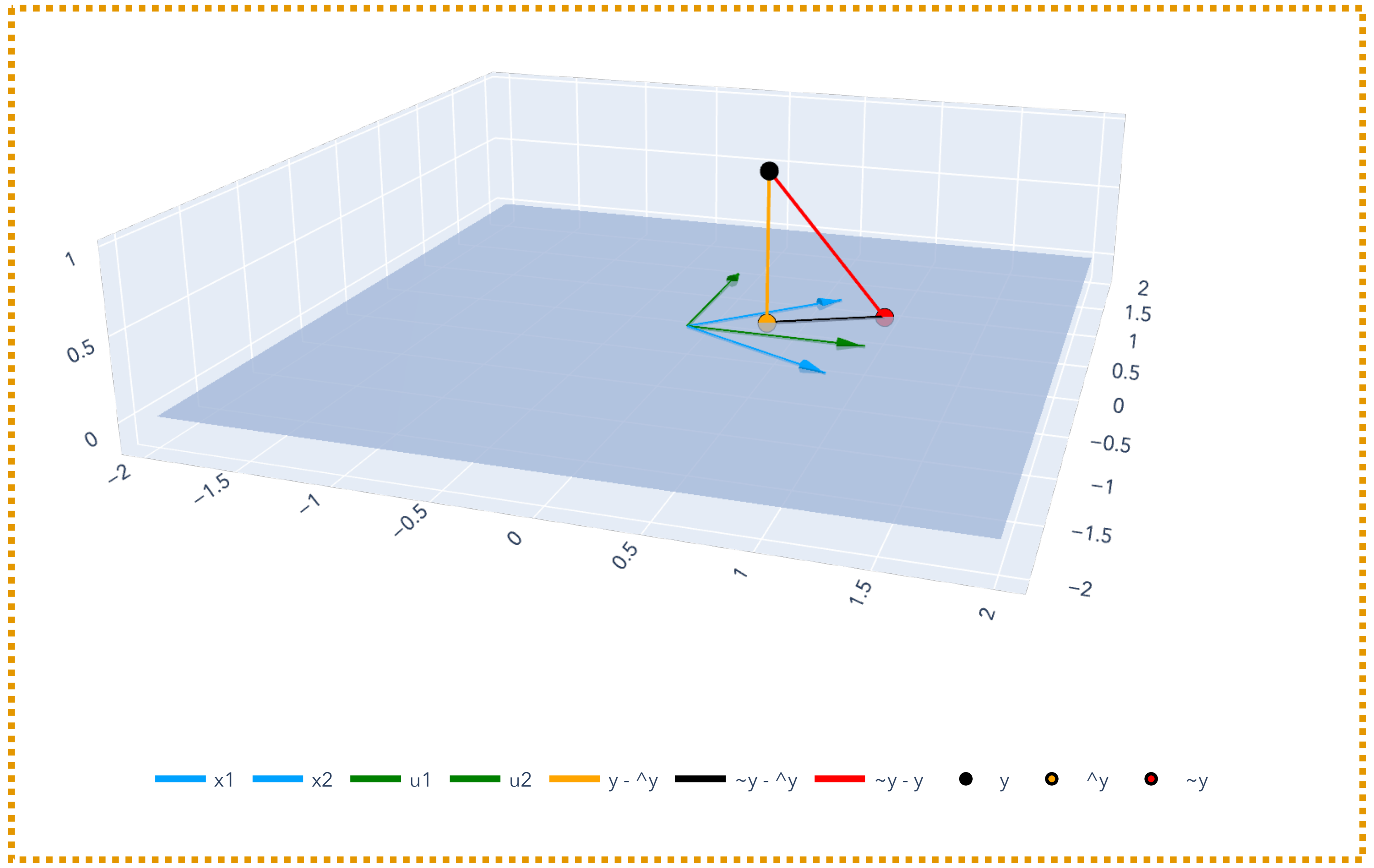
We defined eigenvectors and eigenvalues of square matrices. When a square matrix is diagonalizable:

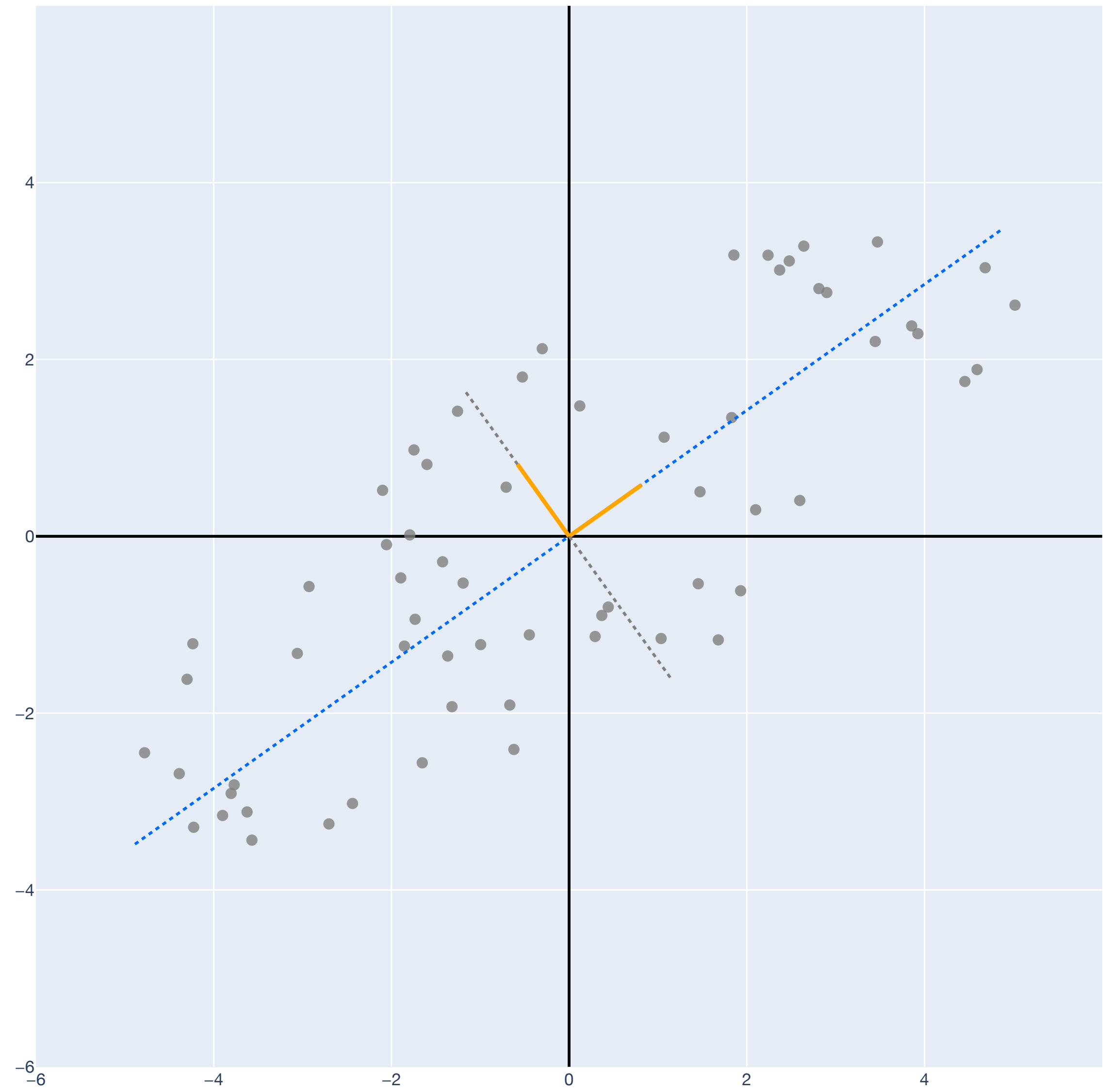
$$\mathbf{X} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^\top$$

The spectral theorem tells us that symmetric matrices are diagonalizable.

One example of a symmetric matrix is $\mathbf{X}^\top\mathbf{X}$, so we did a rudimentary eigenvector/eigenvalue analysis of $(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{y}$ in the error model:

$$\mathbf{y} = \mathbf{X}\mathbf{w}^* + \epsilon.$$





Eigendecomposition and PSD Matrices

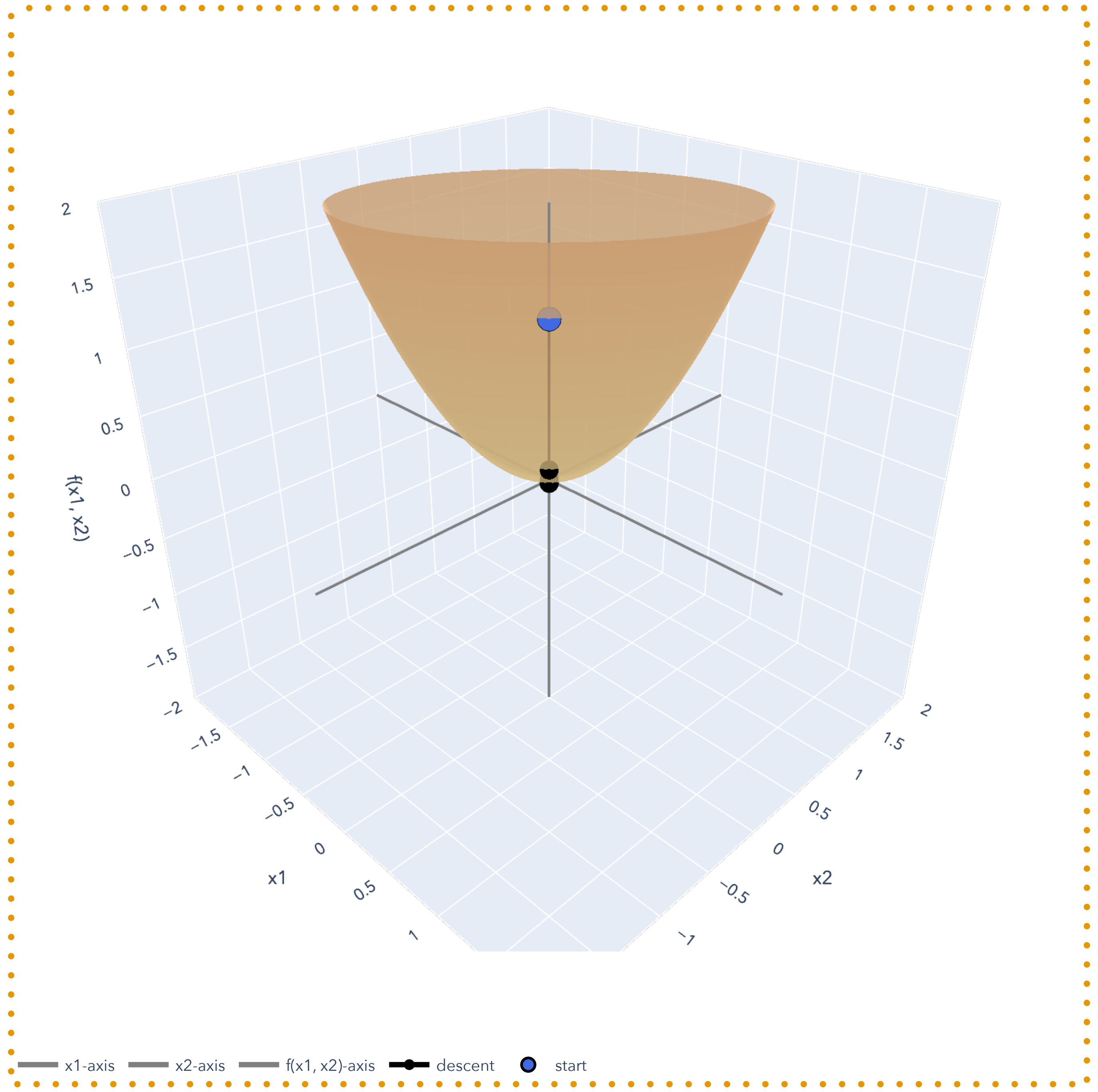
Big Picture: Gradient Descent

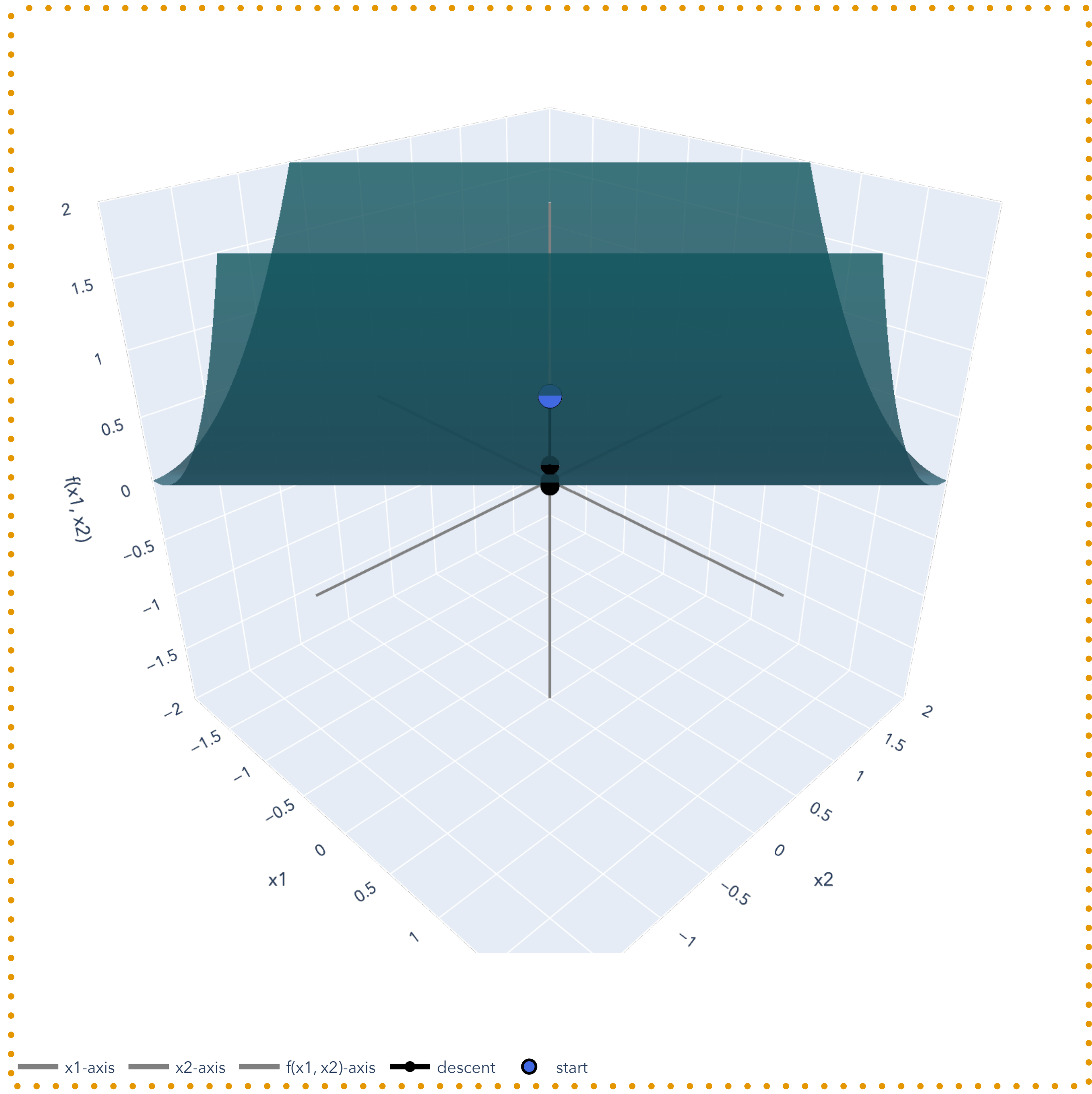
Defined an important class of square, symmetric matrices, positive semidefinite (PSD) matrices.

PSD matrices are always associated with functions called quadratic forms

$$f(\mathbf{x}) := \mathbf{x}^\top \mathbf{A}\mathbf{x},$$

which look "bowl" or "envelope" shaped.





Week 3.1

Differentiation and vector calculus

Differentiation and vector calculus

Big Picture: Least Squares

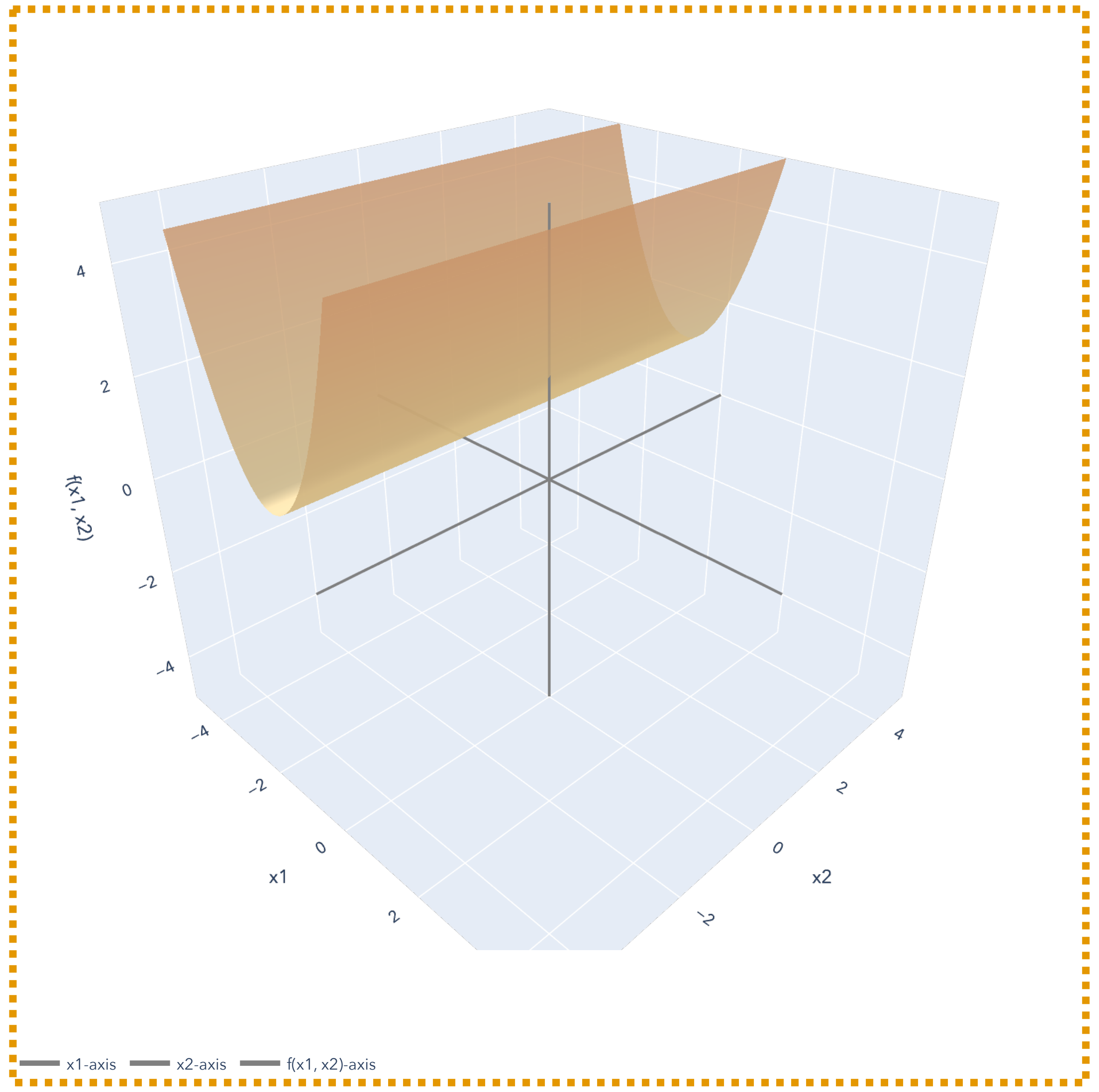
The directional, partial, and total derivatives are summarized with the gradient and Jacobian.

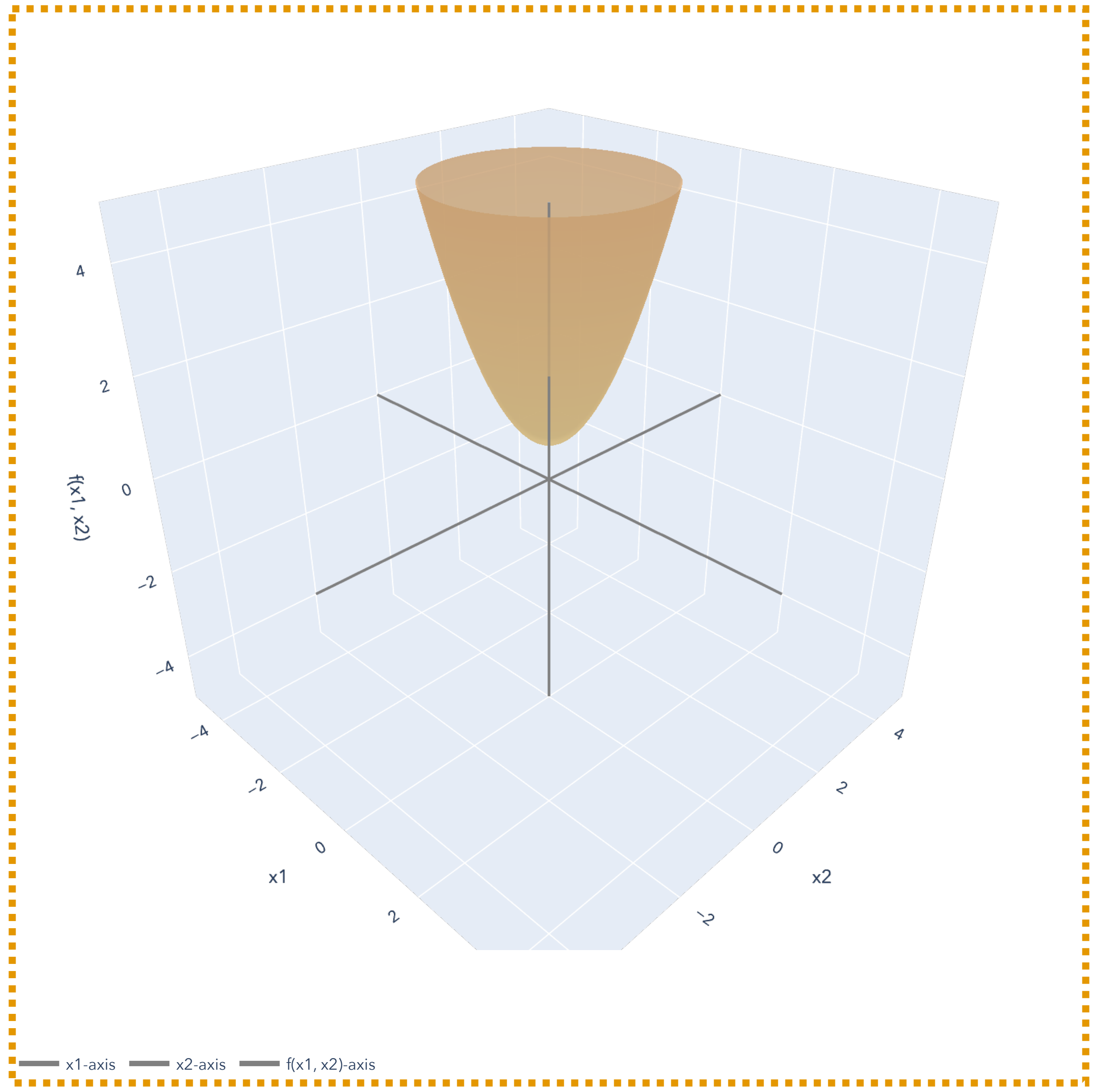
Using *analogy* to single variable calculus optimization, we treated

$$f(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$$

as a function to optimize and proved the same theorem, from a calculus/optimization perspective.

Theorem (OLS solution). If $n \geq d$ and $\text{rank}(\mathbf{X}) = d$, then: $\hat{\mathbf{w}} = (\mathbf{X}^\top \mathbf{X})^{-1}\mathbf{X}^\top \mathbf{y}$.



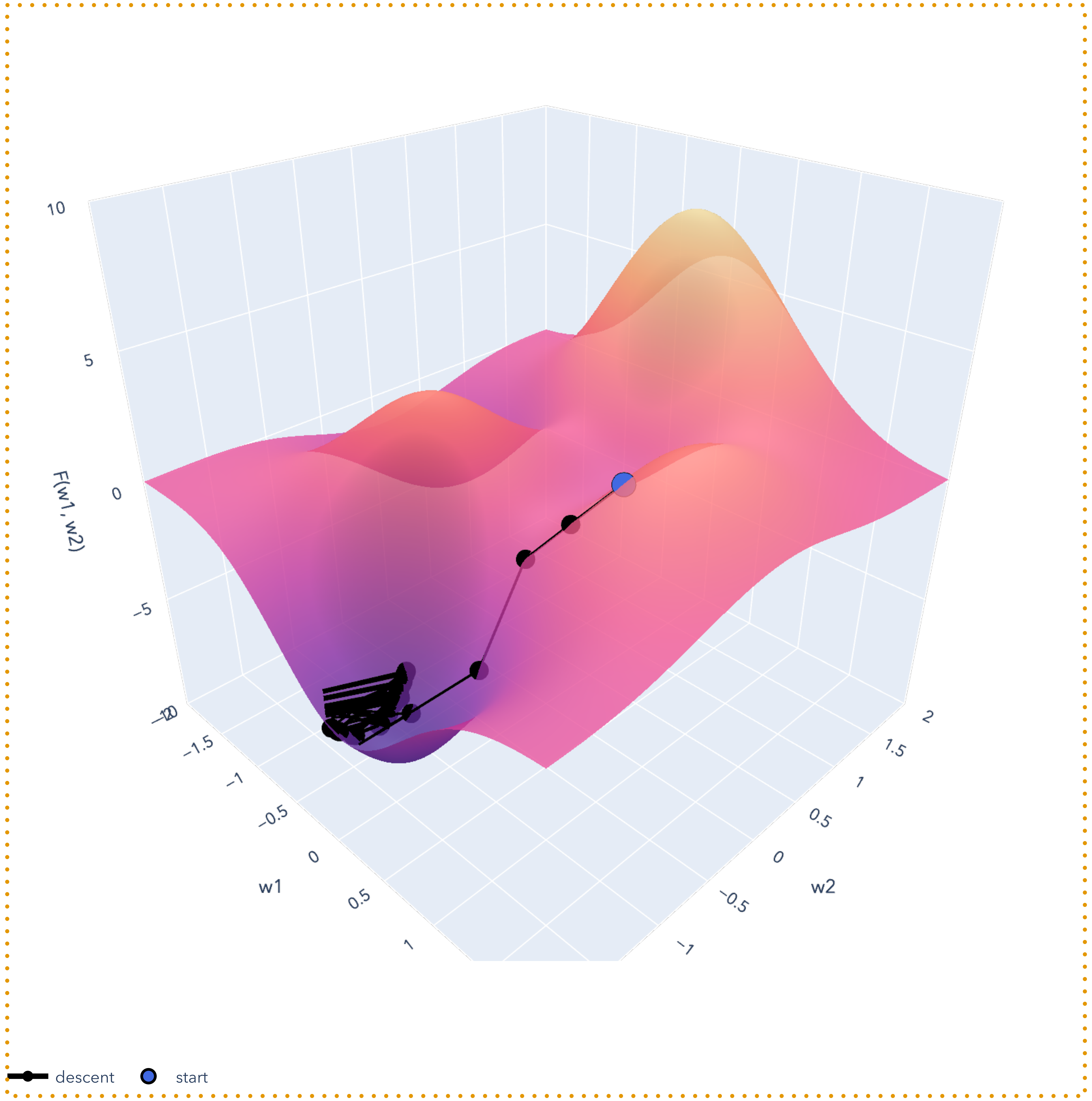


Differentiation and vector calculus

Big Picture: Gradient Descent

The gradient points in the direction of steepest ascent. This lets us write out the algorithm for gradient descent:

$$\mathbf{w}_t \leftarrow \mathbf{w}_{t-1} - \eta \nabla f(\mathbf{w}_{t-1}).$$



Week 3.2

Linearization and Taylor series

Linearization and Taylor series

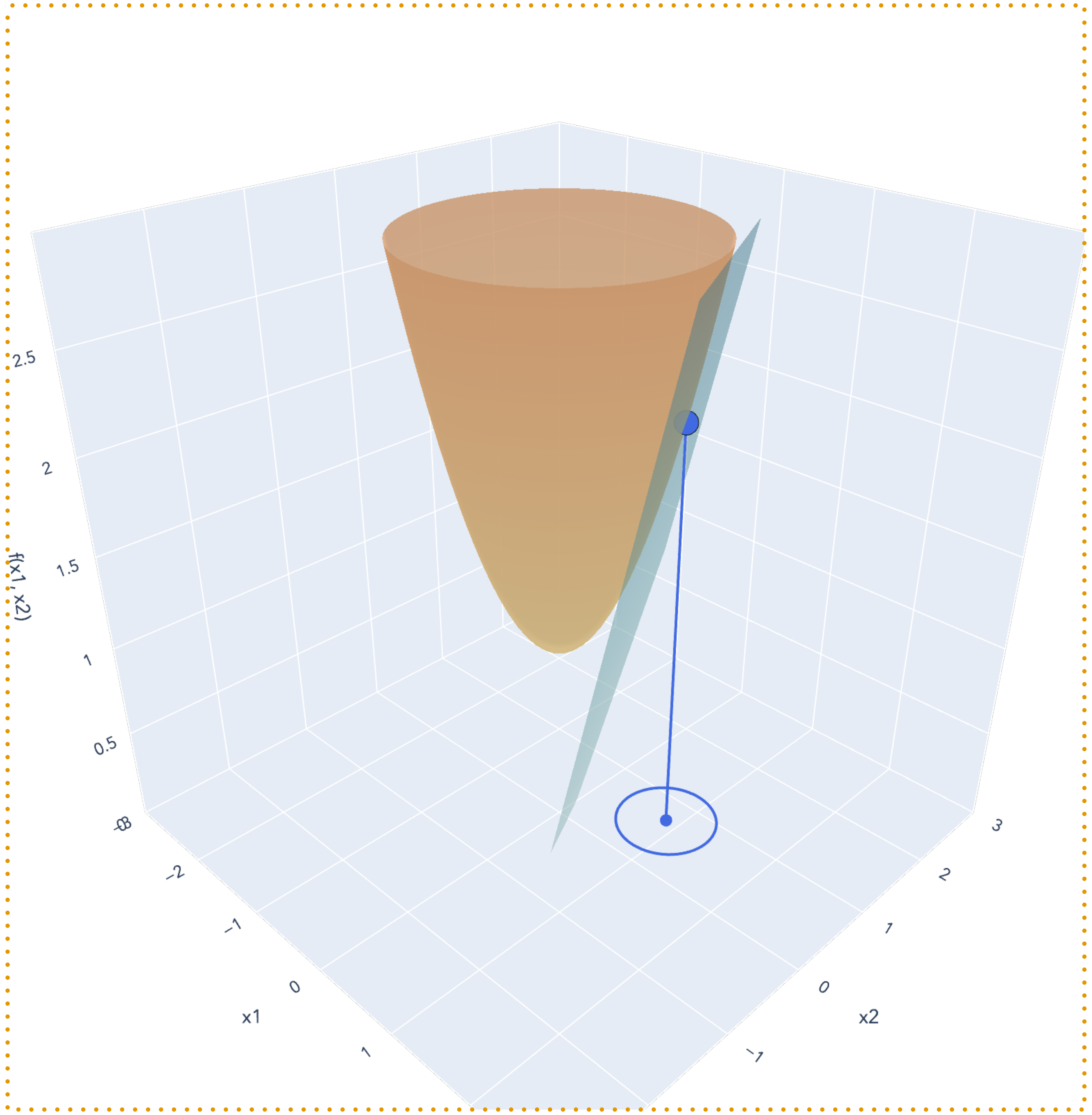
Big Picture: Least Squares

We discussed linearization, a main motivation for the techniques of multivariable calculus:

$$f(\mathbf{x}) \approx f(\mathbf{x}_0) + \nabla f(\mathbf{x}_0)^\top (\mathbf{x} - \mathbf{x}_0)$$

This is a "part" of the Taylor series of a function. We quantified the approximation error of a Taylor series through Taylor's Theorem.

The error term in the first-order Taylor expansion was a function of the Hessian, which is always a symmetric matrix for $\mathscr{C}^2$ functions.



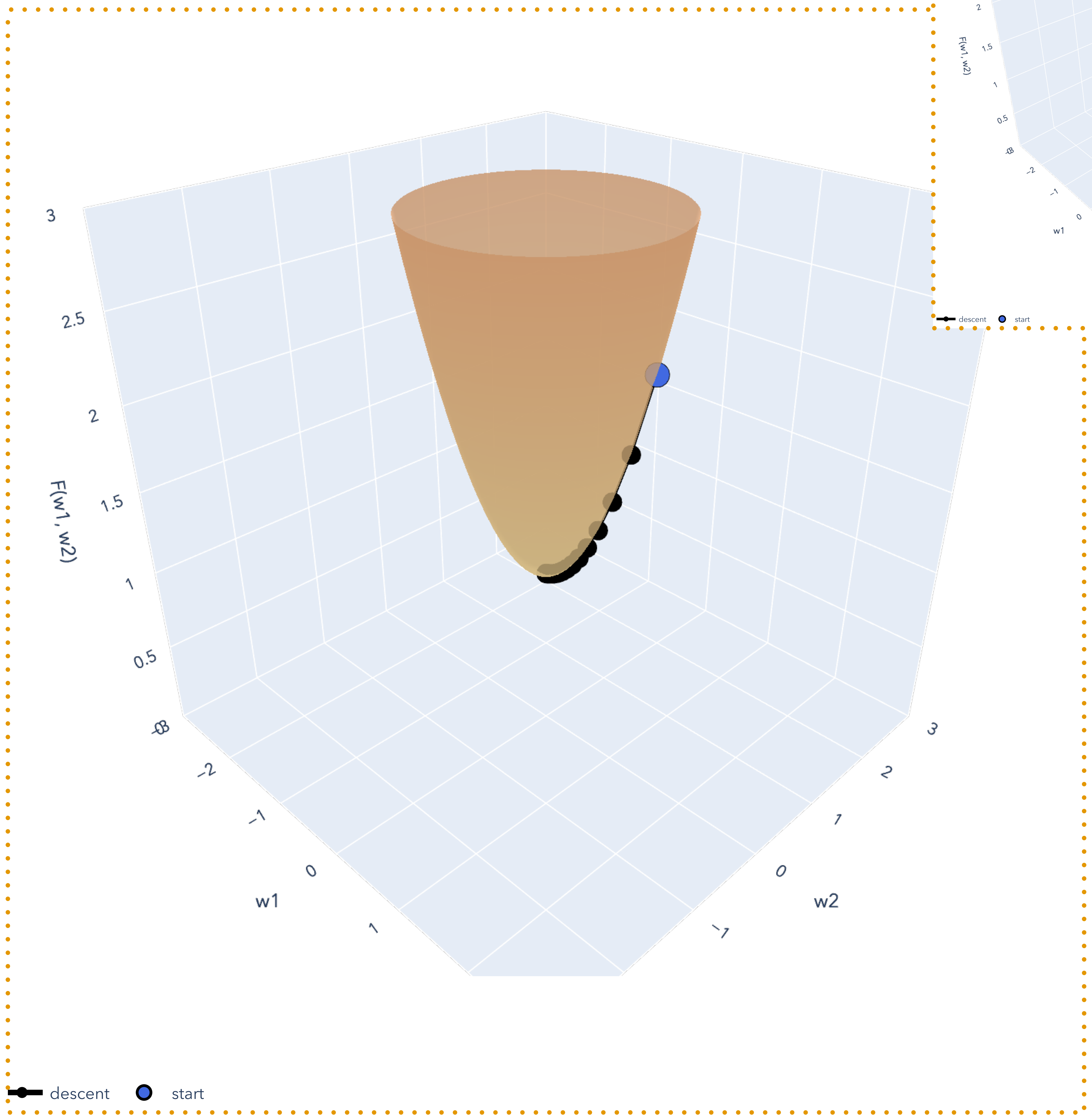
Linearization and Taylor series

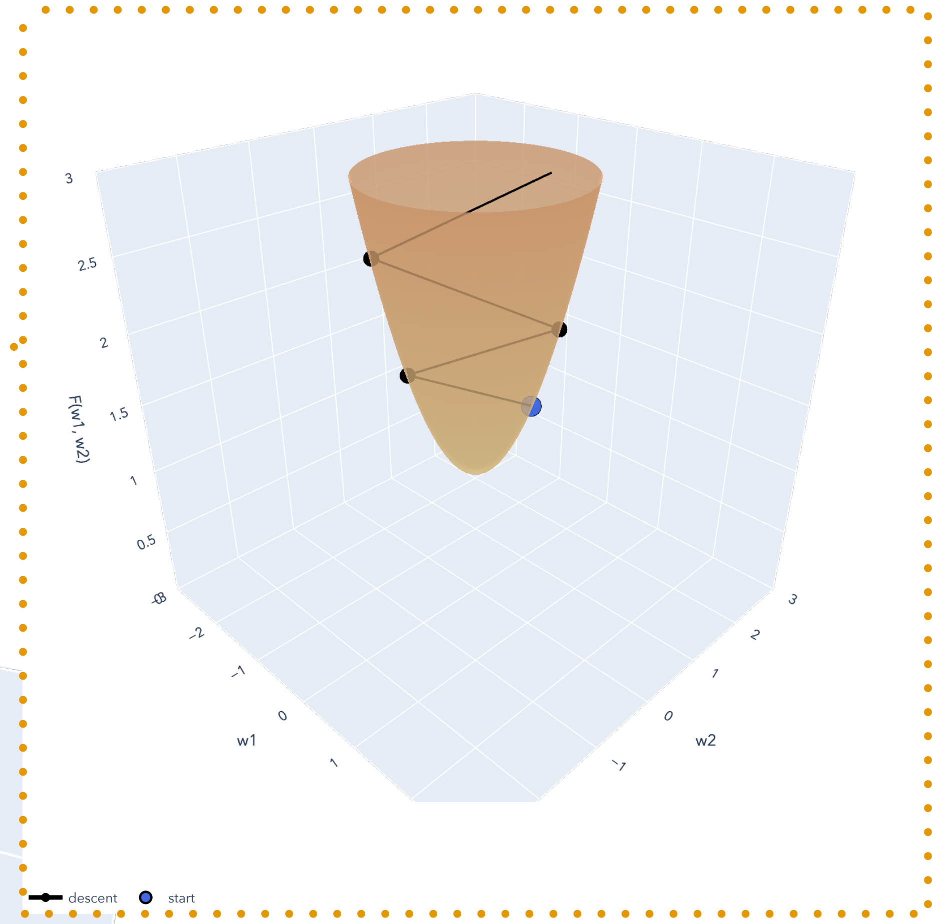
Big Picture: Gradient Descent

Taylor's Theorem and smoothness of the Hessian allowed us to analyze the first-order Taylor approximation to get our first GD theorem:

Theorem (Descent Lemma). If $f \in \mathscr{C}^2$ and is β-smooth, then with $\eta = 1/\beta$, for any $\mathbf{w} \in \mathbb{R}^d$,

$$f(\mathbf{w} - \eta \nabla f(\mathbf{w})) \leq f(\mathbf{w}) - \frac{1}{2\beta} \|\nabla f(\mathbf{w})\|^2.$$





Week 4.1

Optimization and the Lagrangian

Optimization and the Lagrangian

Big Picture: Least Squares

$$\begin{aligned} \underset{\mathbf{x}\in\mathbb{R}^d}{\text{minimize}} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & \mathbf{x} \in \mathscr{C} \end{aligned}$$

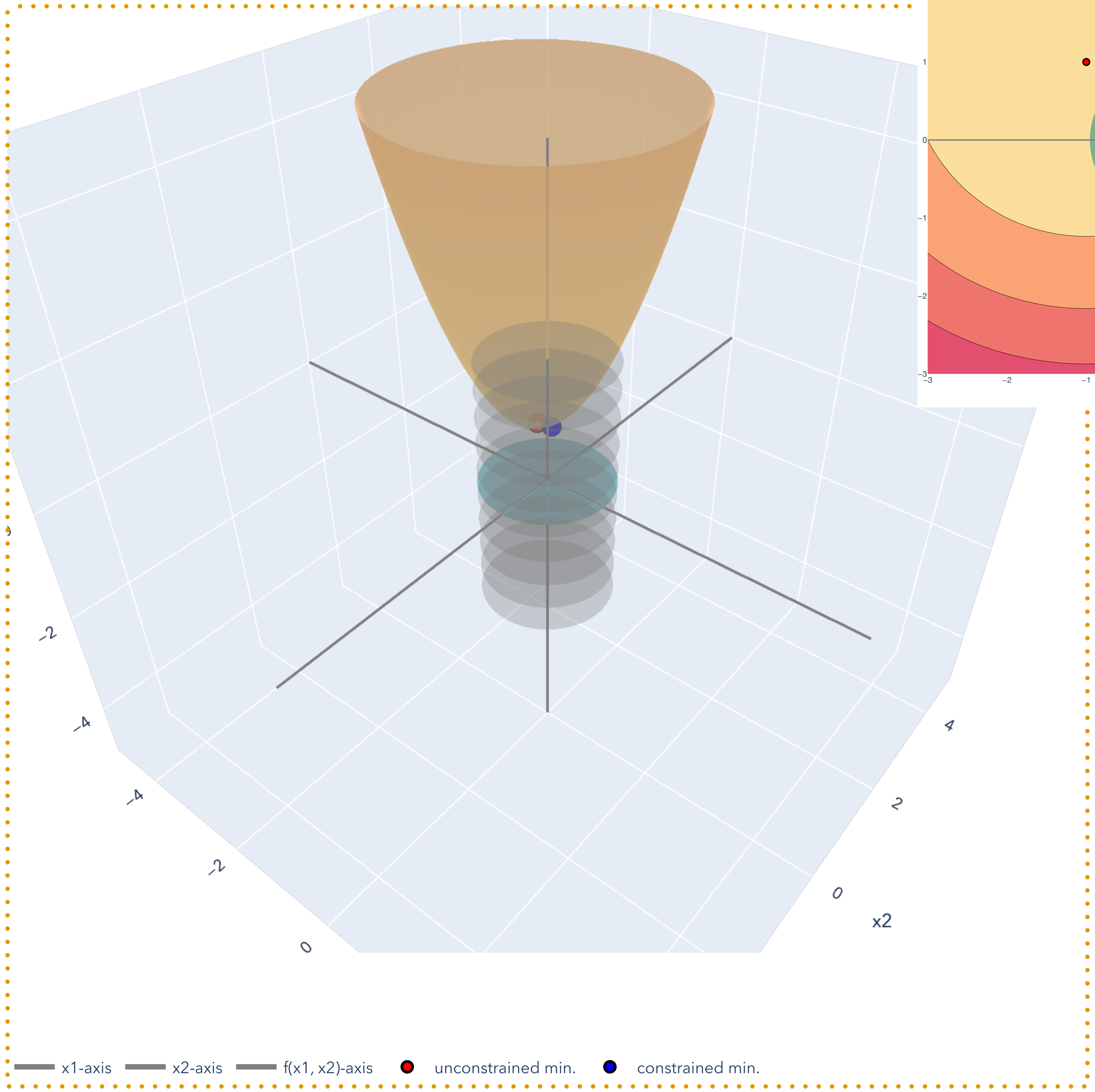
Gave the necessary conditions for unconstrained local minima, filled in gaps in OLS proof.

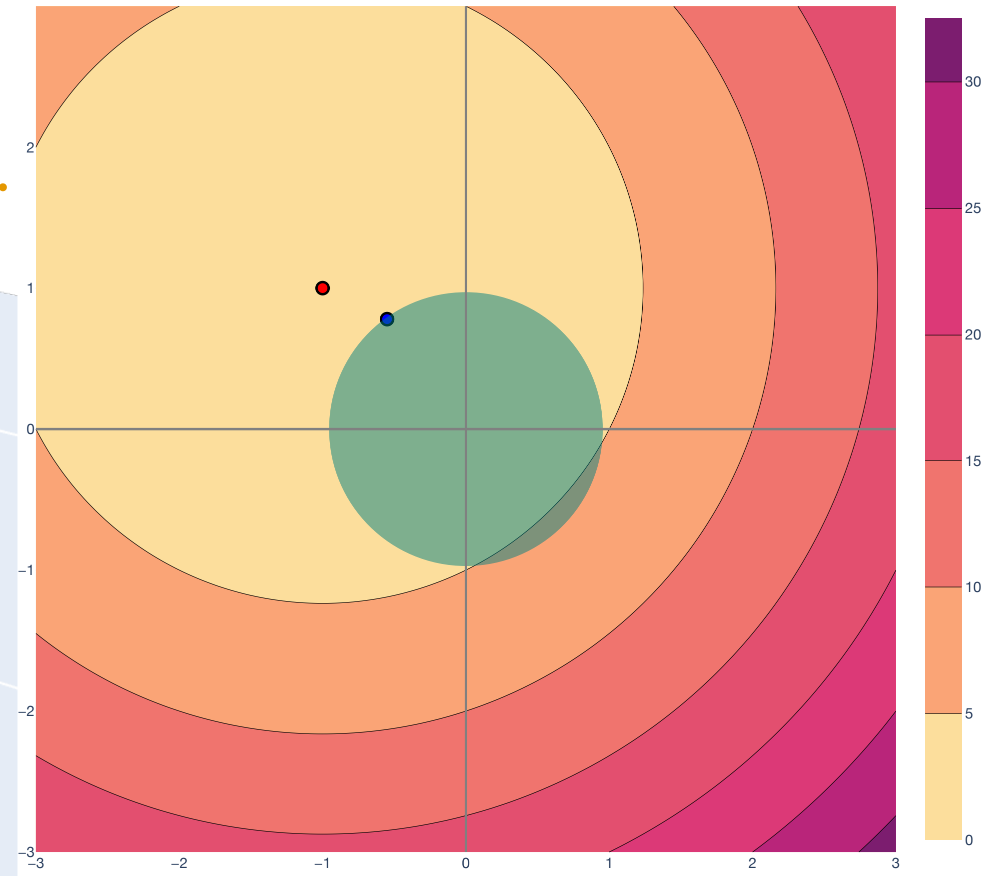
Defined the Lagrangian $L(\mathbf{x}, \lambda)$, which helped us solve constrained optimization problems by "unconstraining."

Two constrained problems related to OLS:

1. Least norm solution. $\hat{\mathbf{w}} = \mathbf{X}^{+}\mathbf{y}$.

2. Ridge regression. $\hat{\mathbf{w}} = (\mathbf{X}^\top\mathbf{X} + \gamma\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{y}$





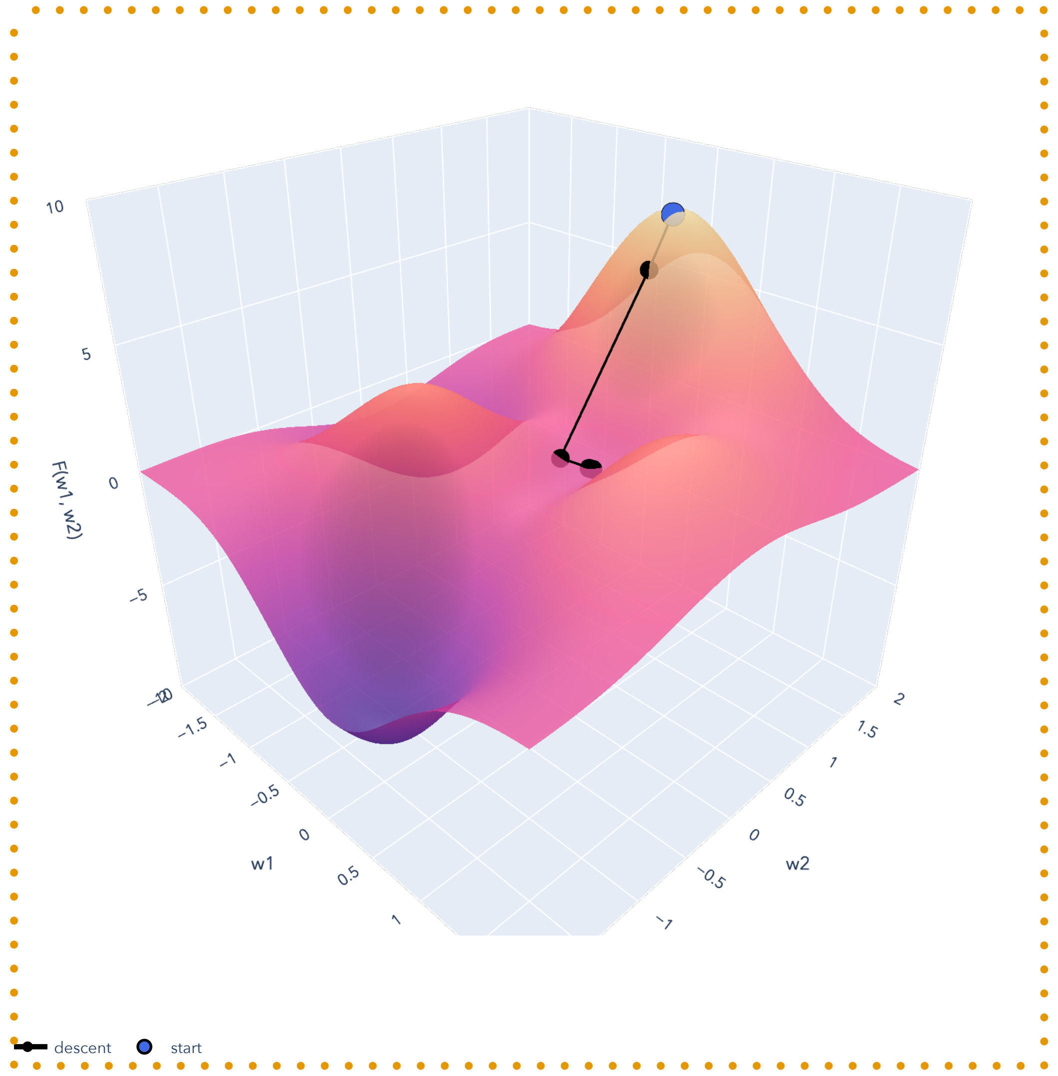
Optimization and the Lagrangian

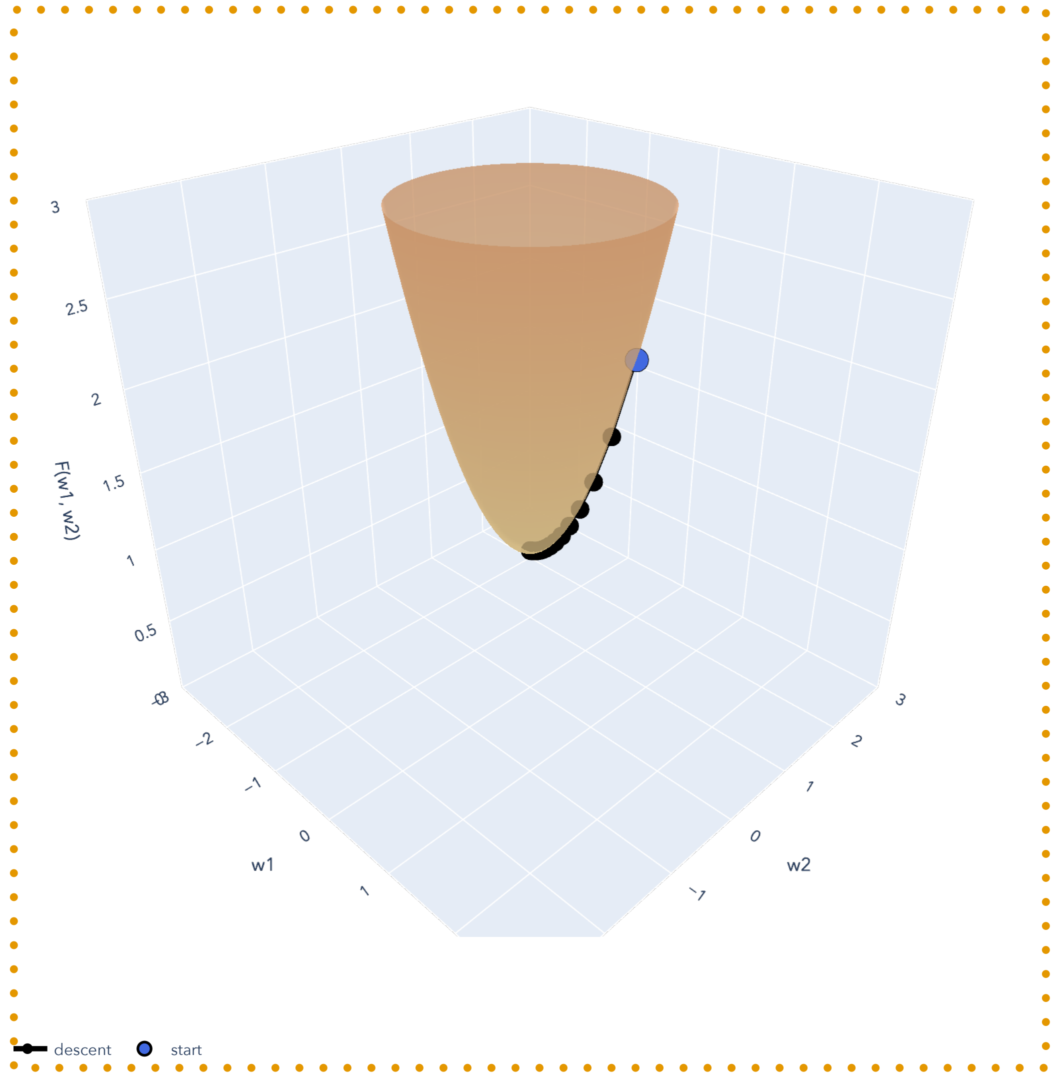
Big Picture: Gradient Descent

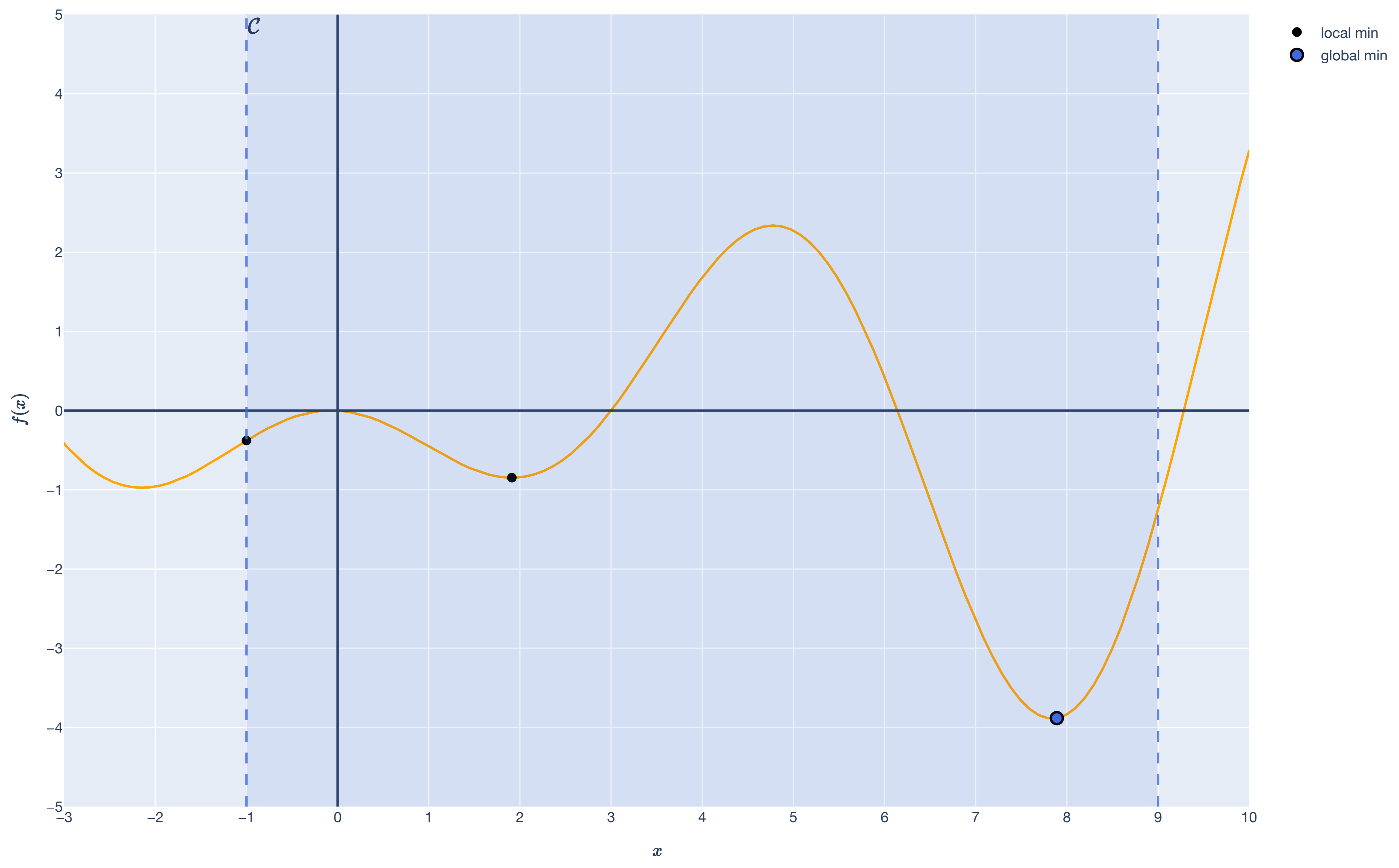
Classified the types of minima we can hope for in an optimization problem: unconstrained local minima, constrained local minima, and global minima.

We want global minima but GD and the descent lemma only says something about getting to the local minima.

$$f(\mathbf{w} - \eta \nabla f(\mathbf{w})) \leq f(\mathbf{w}) - \frac{1}{2\beta} \|\nabla f(\mathbf{w})\|^2$$







Week 4.2

Basics of convex optimization

Basics of convex optimization

Big Picture: Least Squares

Convexity of functions and sets. Convex functions satisfy:

$$f(\alpha \mathbf{x} + (1-\alpha)\mathbf{y}) \leq \alpha f(\mathbf{x}) + (1-\alpha) f(\mathbf{y})\,.$$

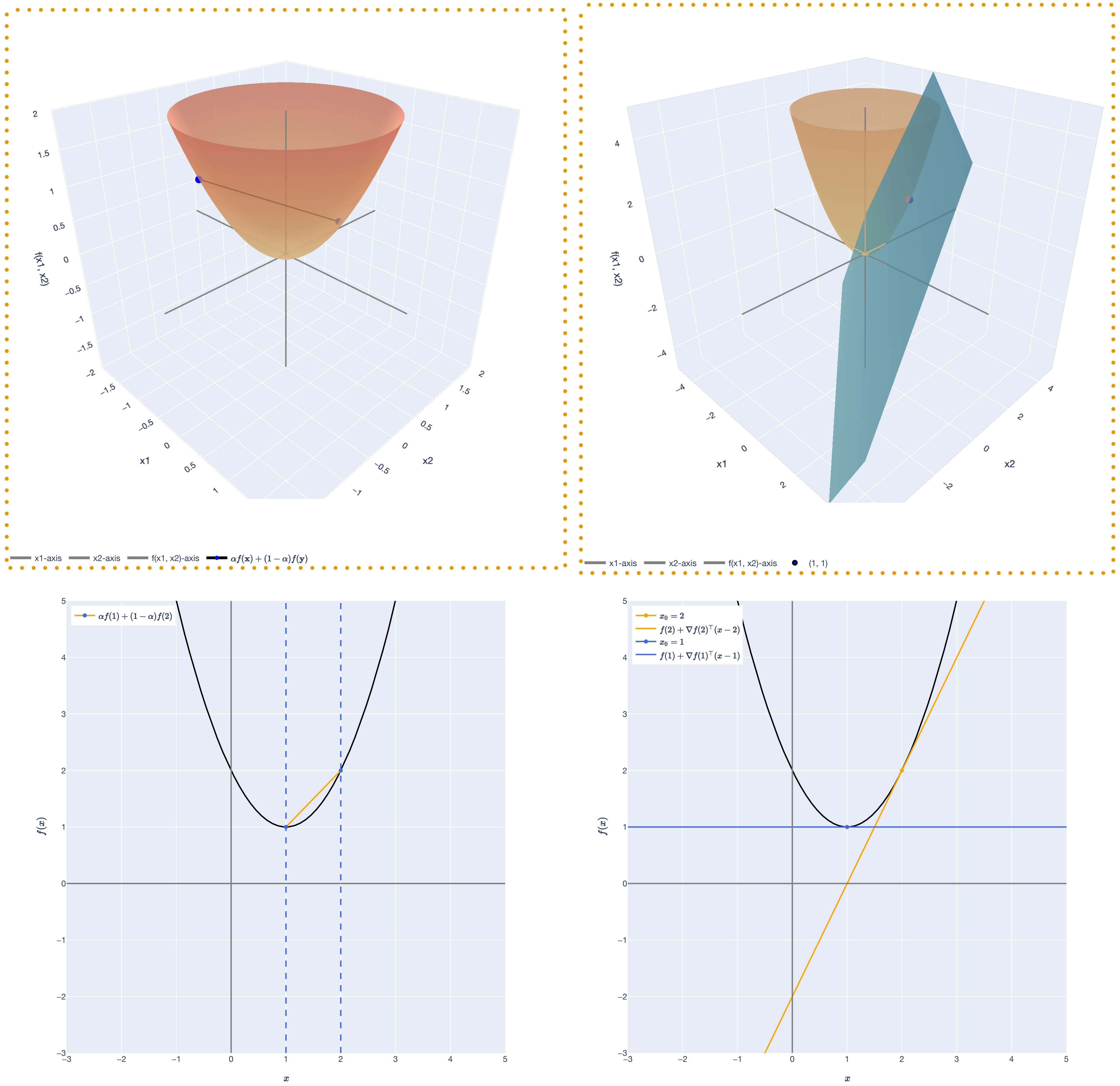
$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla_{\mathbf{x}} f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x})\,.$$

$\nabla^2 f(\mathbf{x})$ is positive semidefinite.

The key property we proved is that for **convex functions, all local minima are global minima.**

We verified that the OLS objective is convex:

$f(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$ is convex.



Basics of convex optimization

Big Picture: Gradient Descent

Assured that for convex functions, **all local minima are global minima**, we proved *global* convergence for GD:

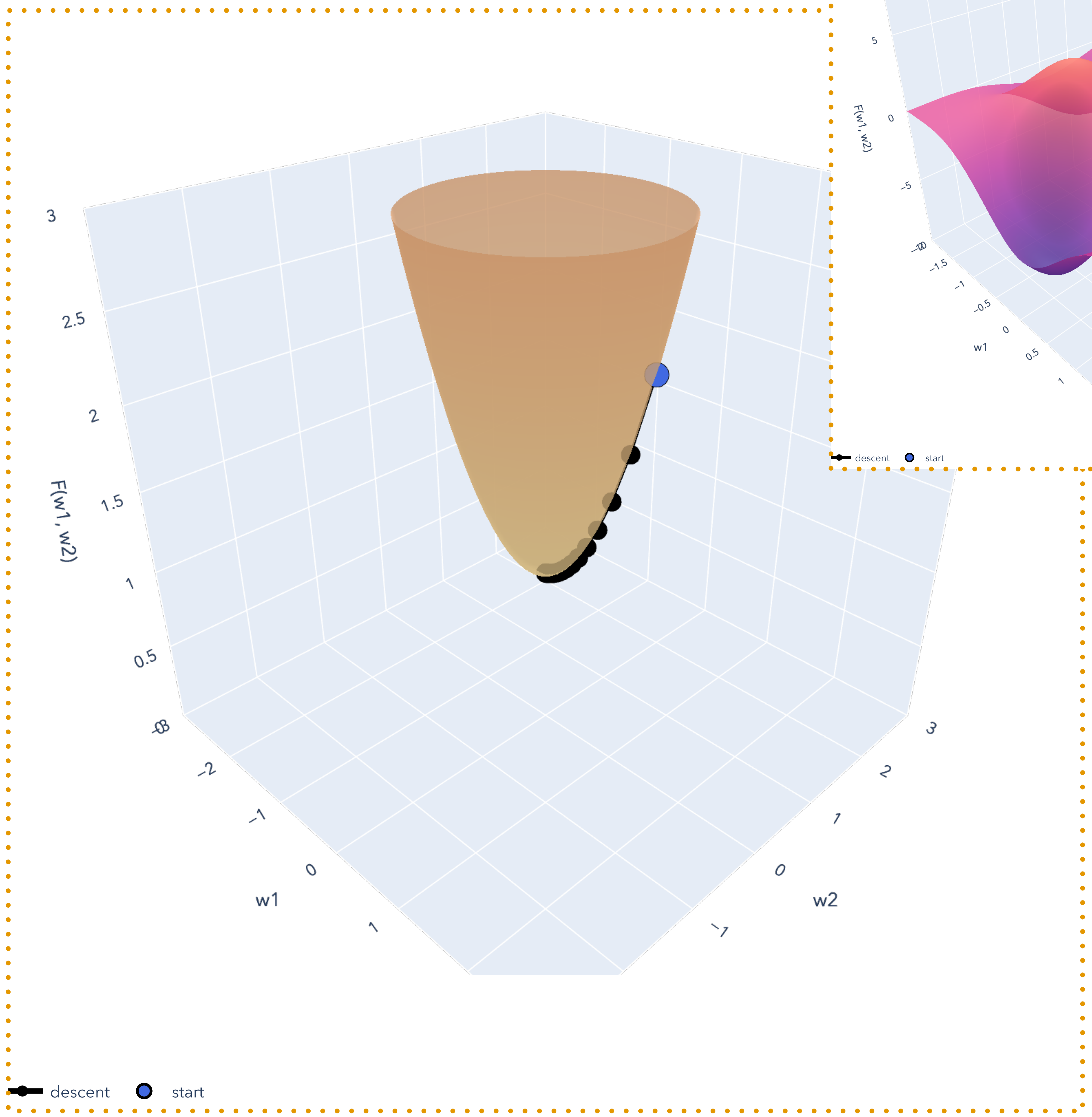
Theorem (Convergence of GD for smooth, convex functions).

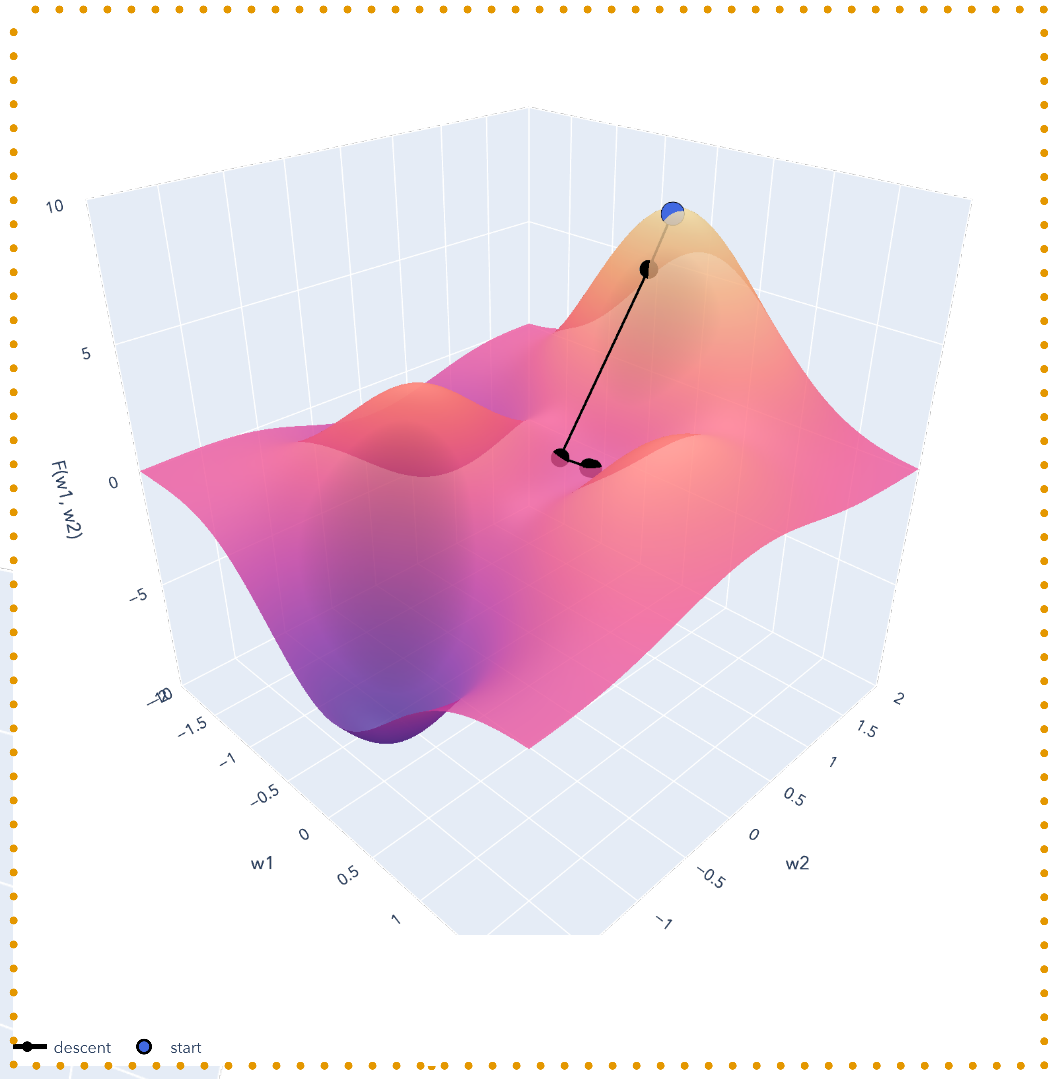
$$f(\mathbf{x}_T) - f(\mathbf{x}^*) \leq \frac{\beta}{2T}\left(\|\mathbf{x}_0 - \mathbf{x}^*\|^2 - \|\mathbf{x}_T - \mathbf{x}^*\|^2\right),$$

after T iterations of our algorithm.

As a corollary, we were able to unite the two stories of our course and **apply GD to OLS** to get:

$$\|\mathbf{X}\mathbf{w}_T - \mathbf{y}\|^2 - \|\mathbf{X}\mathbf{w}^* - \mathbf{y}\|^2 \leq \frac{\beta}{2T}\left(\|\mathbf{w}_0 - \mathbf{w}^*\|^2 - \|\mathbf{w}_T - \mathbf{w}^*\|^2\right).$$



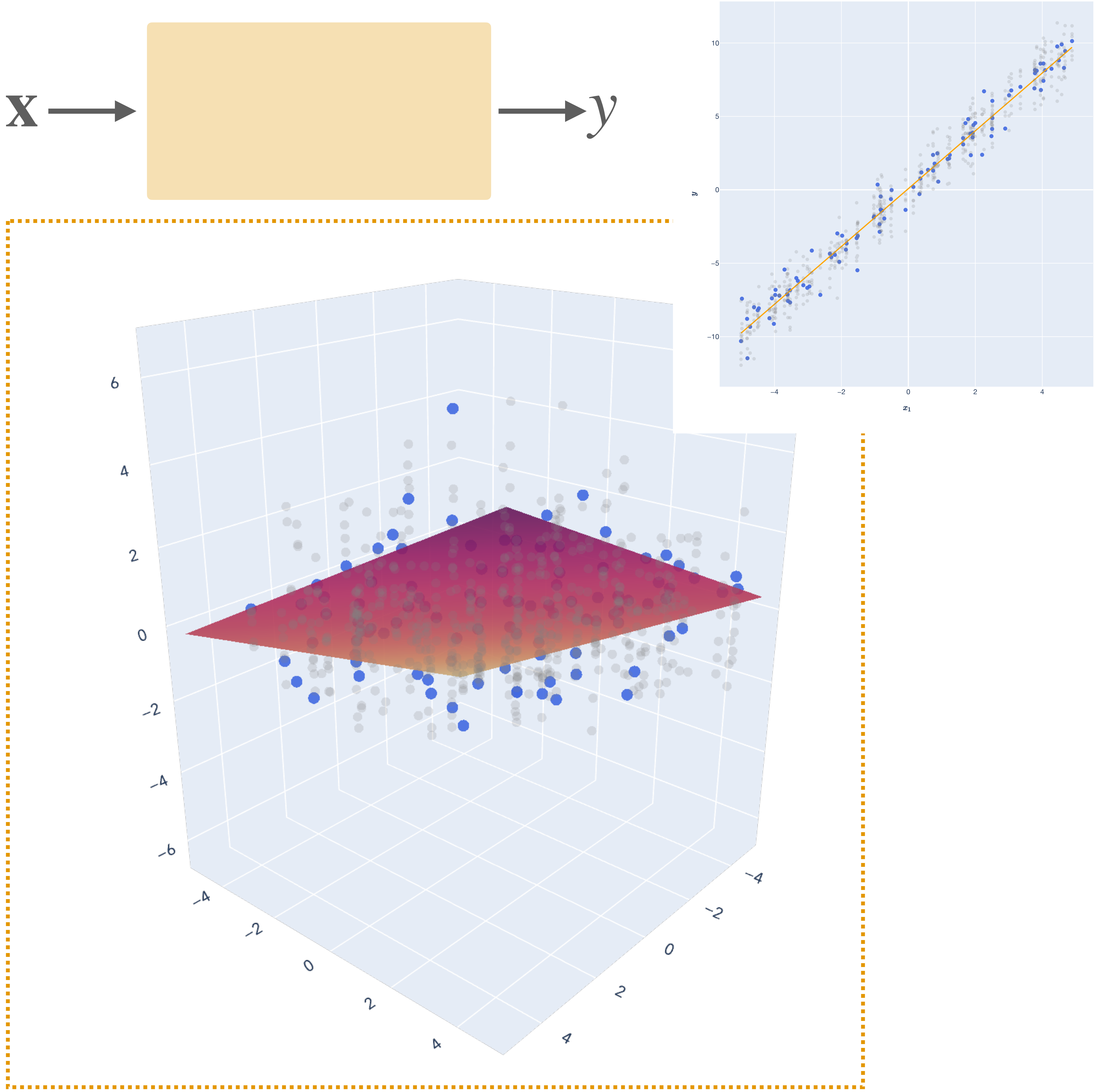


Week 5.1

Probability Theory, Models, and Data

Probability Theory, Models, and Data

Big Picture: Least Squares



Defined probability spaces and random variables.
Random variables come with a CDF and a PMF/PDF.
Most important stats: expectation and variance.

Random vector variances are given in a covariance matrix. This framework allowed us to define the random error model:

$\mathbf{y} = \mathbf{X}\mathbf{w}^* + \epsilon$, where $\mathbb{E}[\epsilon] = 0$ and ϵ_i i.i.d. and indep. of $\mathbf{X}$.

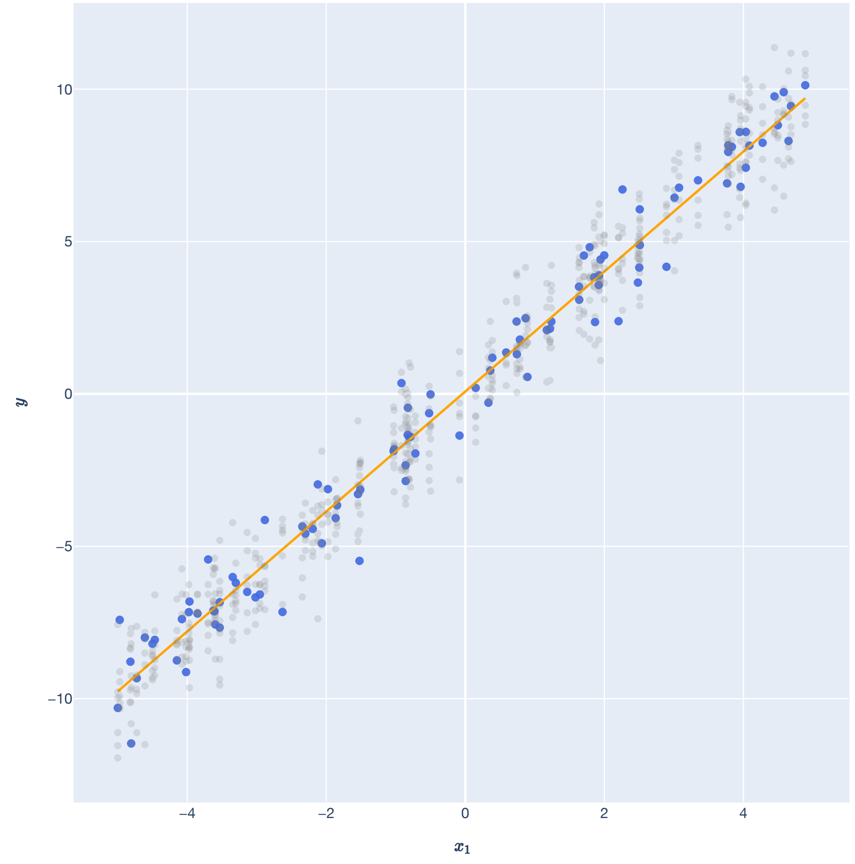
Now, $\hat{\mathbf{w}} = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{y}$ is a random variable with

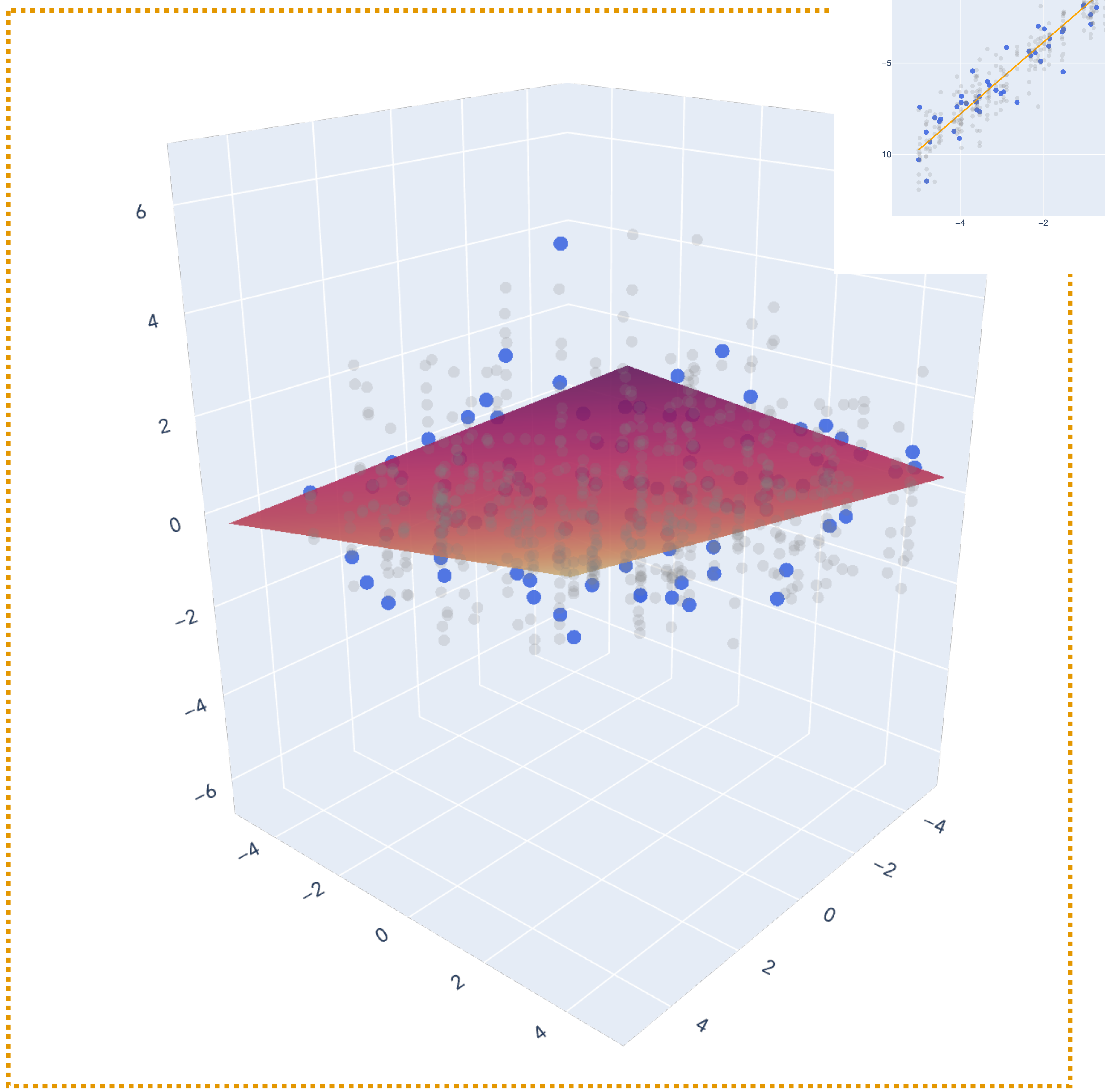
Expectation: $\mathbb{E}[\hat{\mathbf{w}}] = \mathbf{w}^*$.

Variance: $\mathrm{Var}[\hat{\mathbf{w}}] = \sigma^2\mathbb{E}[(\mathbf{X}^\top\mathbf{X})^{-1}]$.

Probability Theory, Models, and Data

Big Picture: Least Squares





Proof of OLS expectation and variance relied heavily on conditioning.

The conditional expectation of a random variable can be thought of as a "best guess" at a random variable given the information of *an event* or *another random variable*.

$$\mathbb{E}[X \mid A], \text{ for } A \subseteq \Omega.$$

$$\mathbb{E}[X \mid Y], \text{ for } Y : \Omega \to \mathbb{R}.$$

Week 5.2

Law of large numbers and statistical estimators

Law of large numbers and statistical estimators

Big Picture: Least Squares

The sample average of i.i.d. random variables:

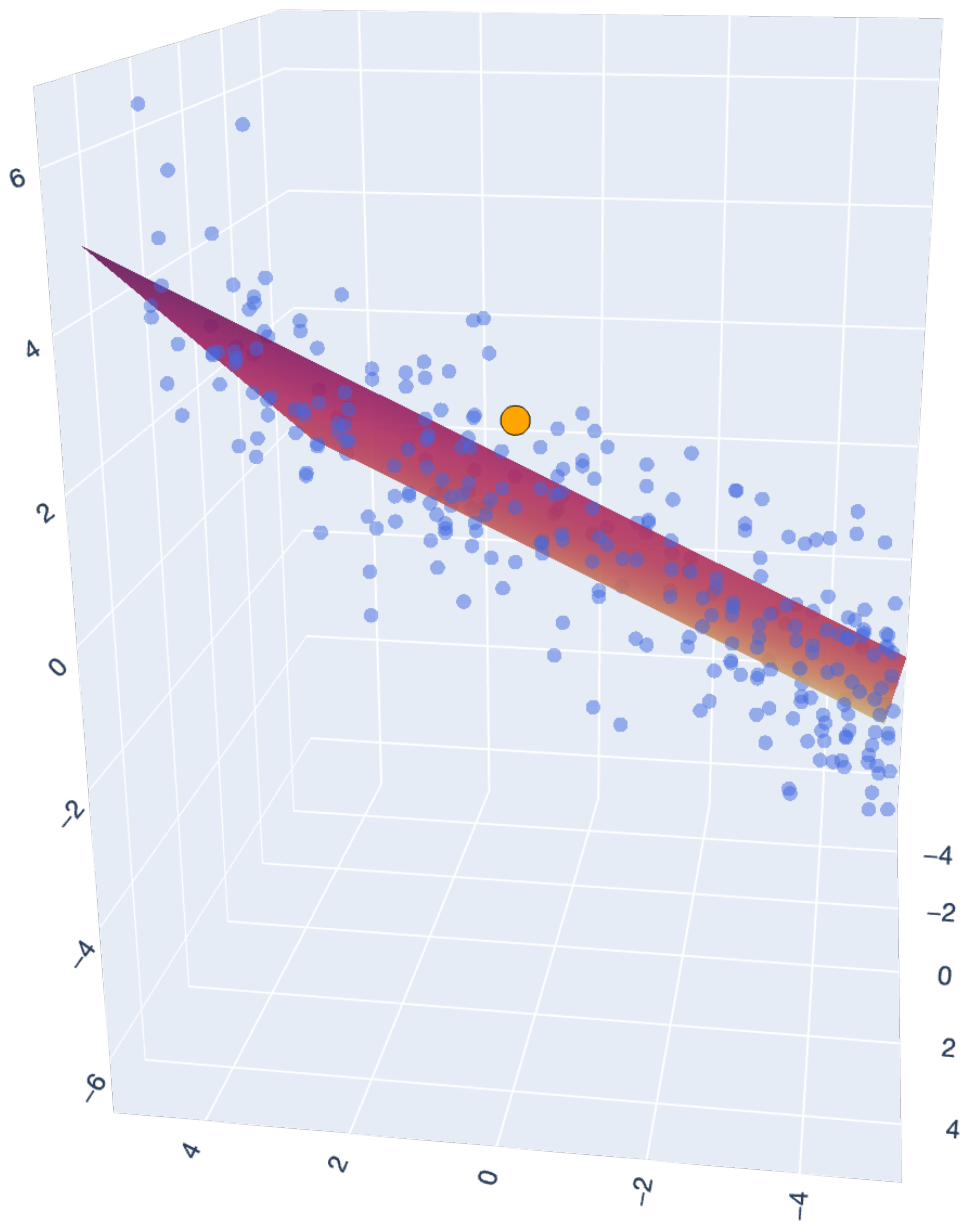
$$\overline{X}_n := \frac{1}{n}\sum_{i=1}^{n} X_i.$$

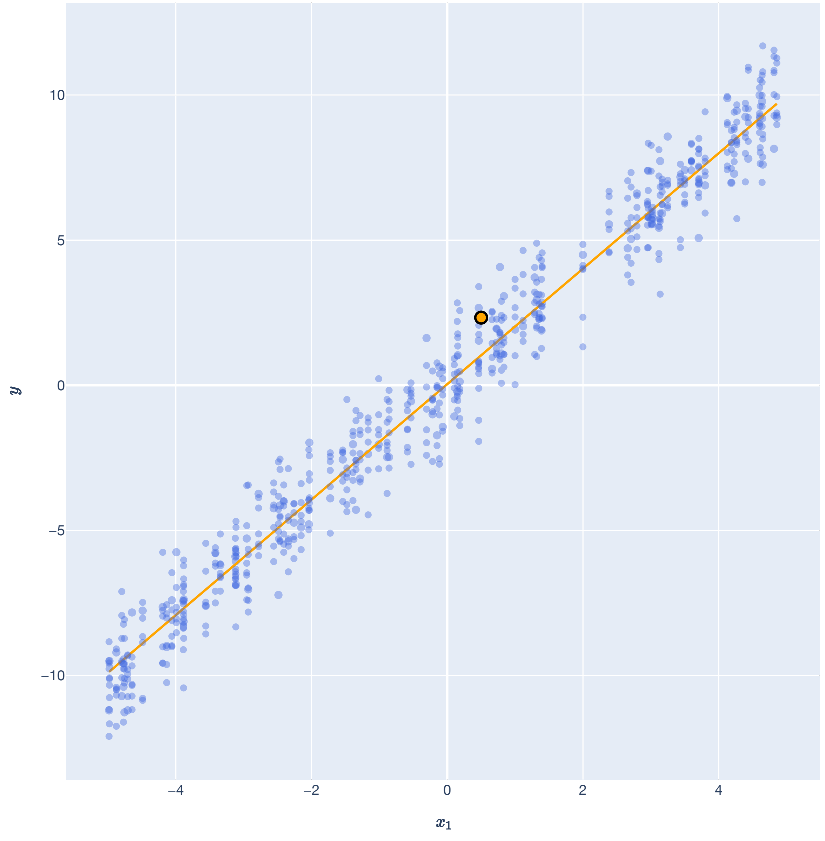
Chebyshev's inequality proved the (Weak) Law of Large Numbers : sample averages approach true means.

Sample average is a statistical estimator of the mean. Estimators have bias and variance connected through the bias-variance decomposition of mean-squared error.

We found that OLS, as a random variable and estimator of $\mathbf{w}^*$ is unbiased, has variance $\mathrm{Var}[\hat{\mathbf{w}}] = \sigma^2\mathbb{E}[(\mathbf{X}^\top\mathbf{X})^{-1}]$, and risk

$$R(\hat{\mathbf{w}}) = \mathbb{E}[(\hat{\mathbf{w}}^\top\mathbf{x}_0 - y_0)^2] \approx \sigma^2 + \frac{\sigma^2 d}{n}.$$





Law of large numbers and statistical estimators

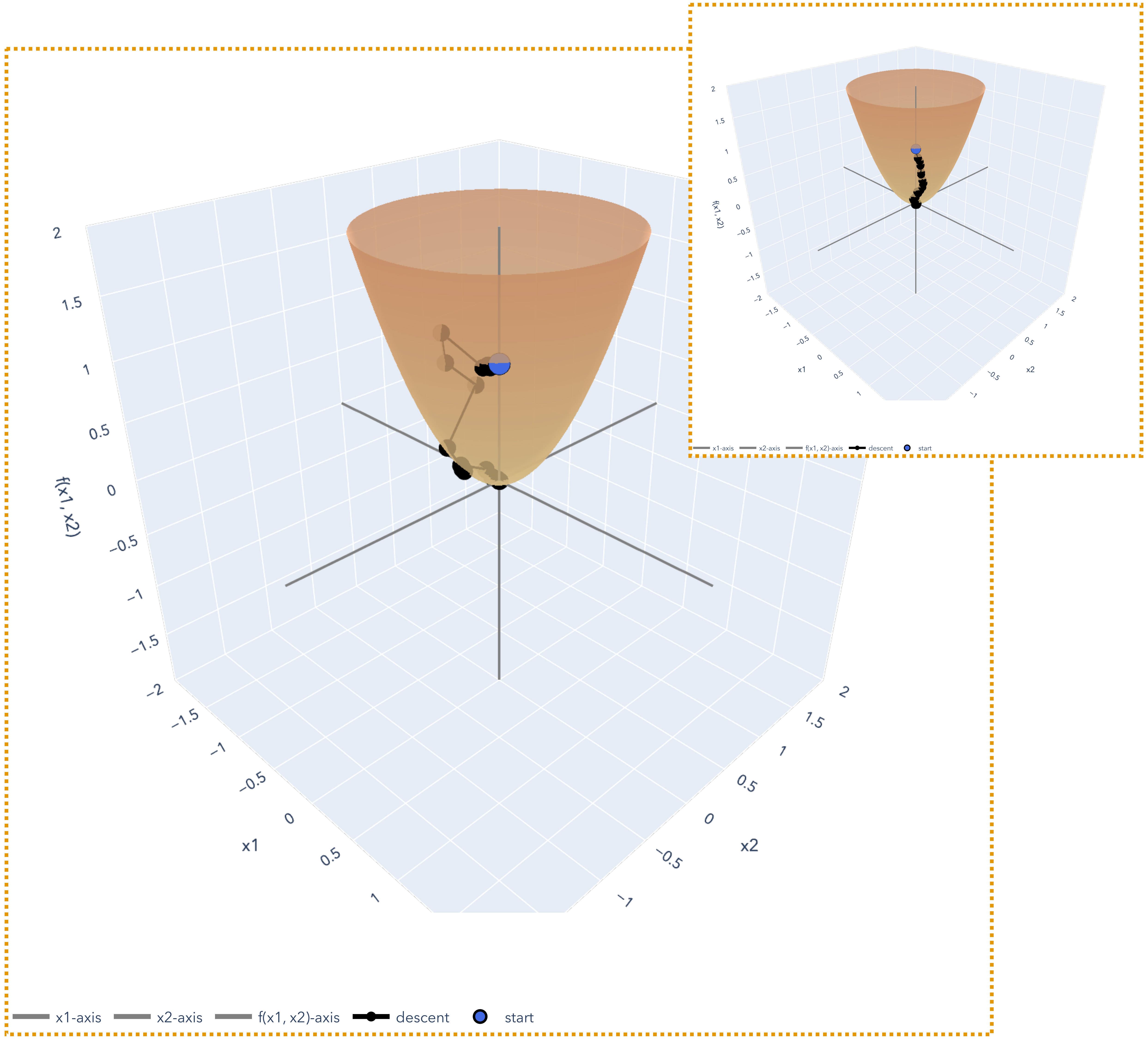
Big Picture: Gradient Descent

We closed the story of gradient descent with stochastic gradient descent (SGD): instead of taking the gradient over *all* the samples $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$, we used an unbiased estimator of the gradient:

Estimand: $\nabla f(\mathbf{w}) = \dfrac{1}{n} \sum_{i=1}^{n} \nabla_{\mathbf{w}} (\mathbf{w}^\top \mathbf{x}_i - y_i)^2.$

Estimator: Sample a single example i uniformly from $1, \ldots, n$ and take the gradient:

$$\widehat{\nabla f(\mathbf{w})} = \nabla_{\mathbf{w}} (\mathbf{w}^\top \mathbf{x}_i - y_i)^2.$$



Week 6.1

Central Limit Theorem, Distributions, and MLE

Central Limit Theorem, Distributions, and MLE

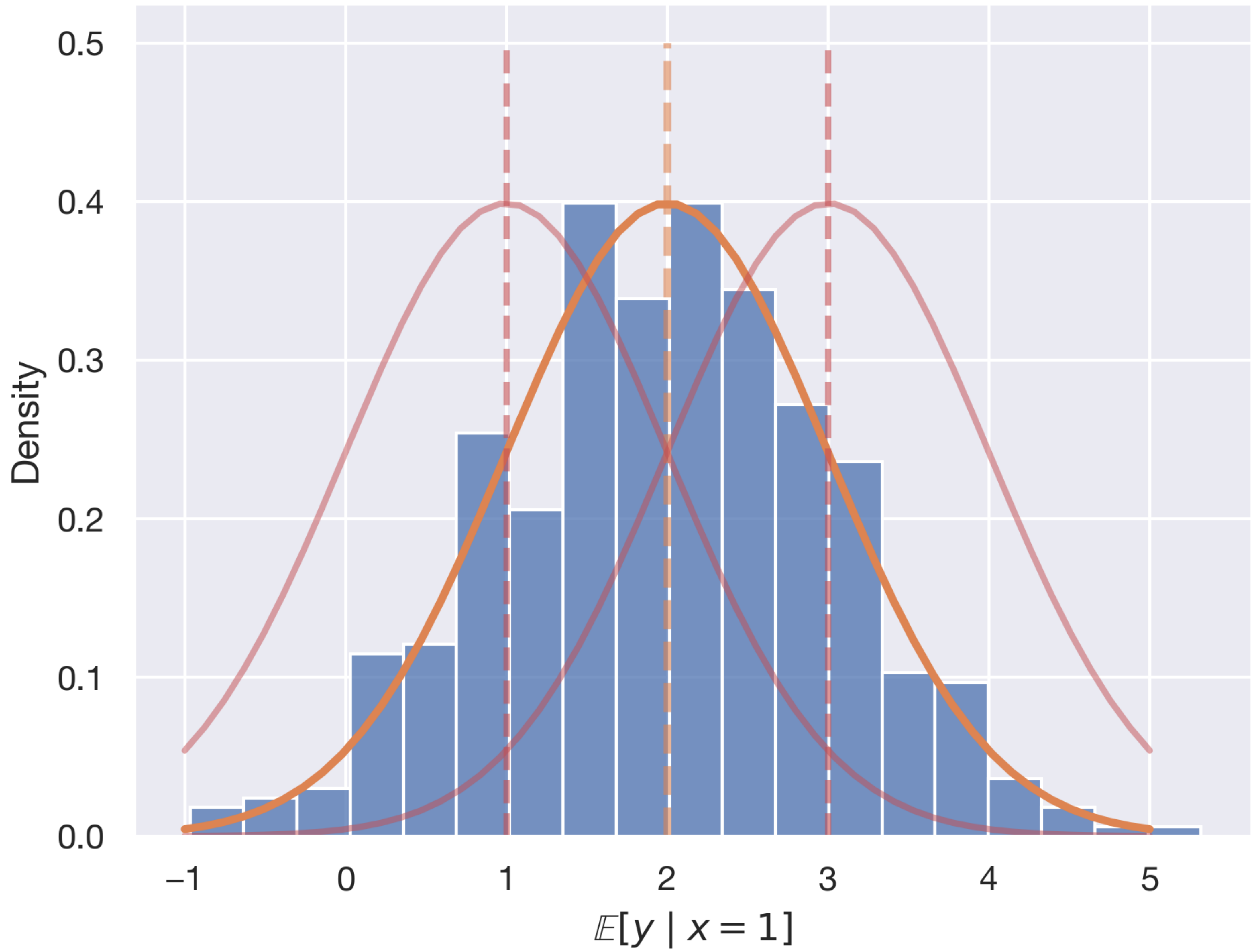
Big Picture: Least Squares

We introduced the Gaussian distribution, and we motivated its importance by the Central Limit Theorem. The Gaussian distribution is just one of many "named distributions" that model common phenomena.

When we have a guess at a parametrized model or generating our i.i.d. data $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$, an alternative perspective on our problem of finding a good model is maximum likelihood estimation (MLE).

This let us prove that, under the Gaussian error model, maximizing the likelihood for the conditional distribution $y \mid \mathbf{x}$ *again* gives us back the **OLS estimator:**

$$\hat{\mathbf{w}}_{MLE} = \arg\max L_n(\mathbf{w}) = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$$



Week 6.2

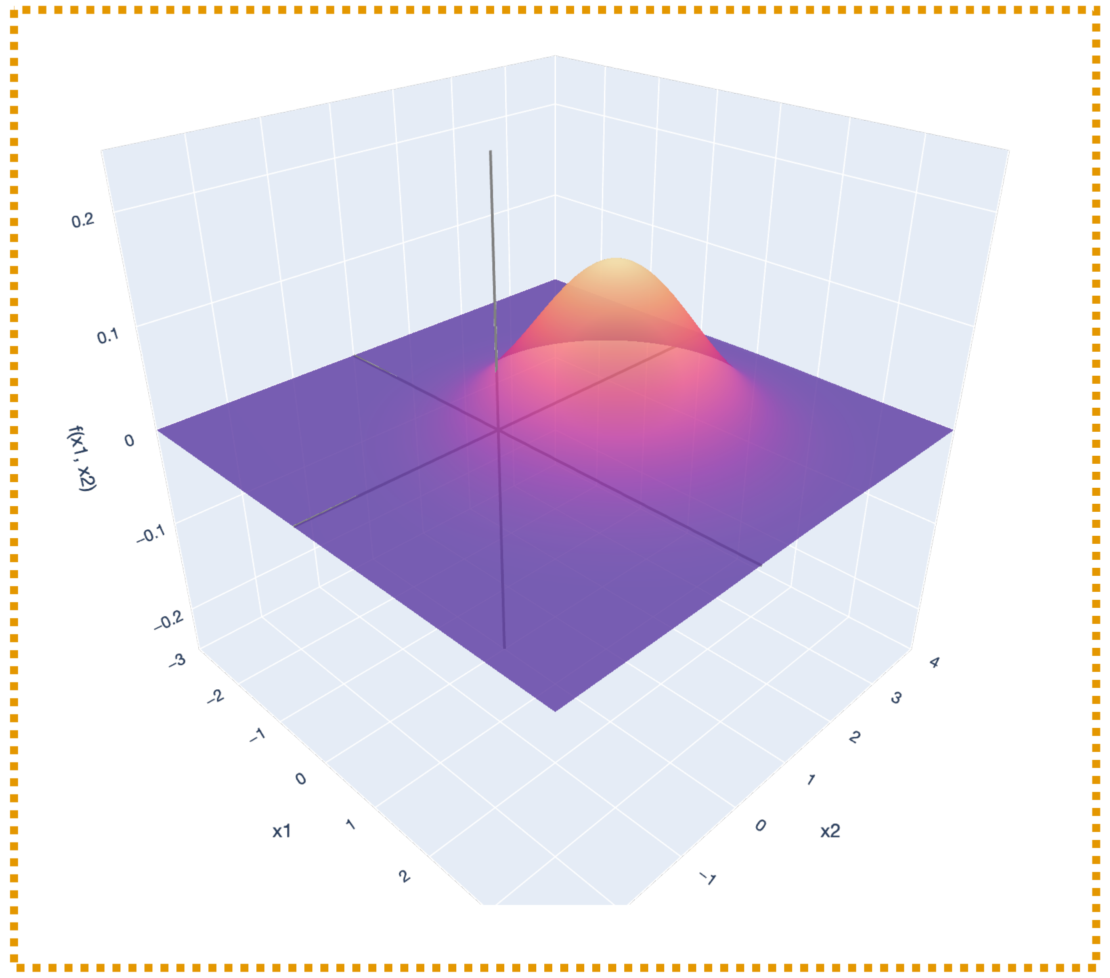
Multivariate Gaussian Distribution

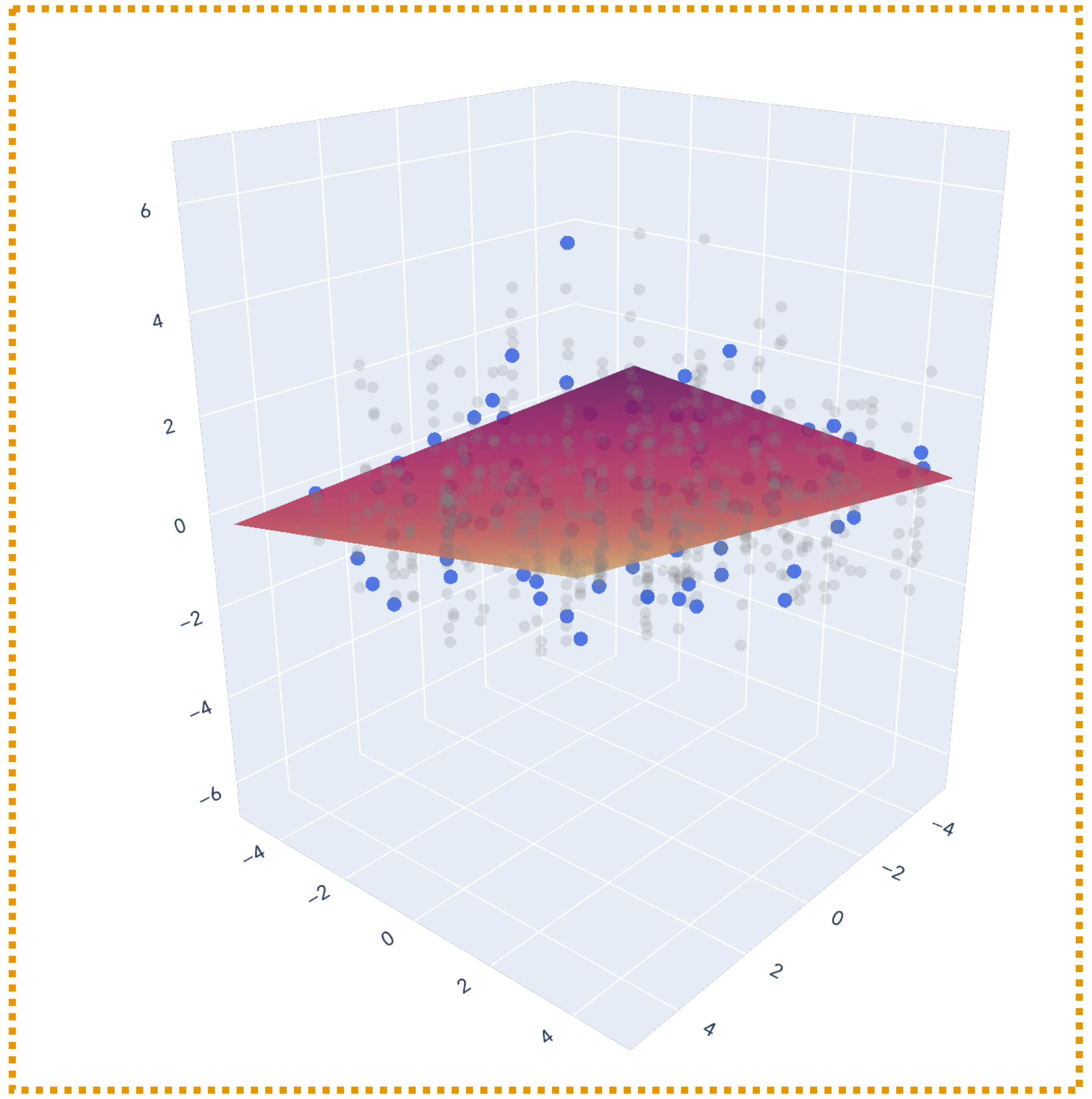
Multivariate Gaussian Distribution

Big Picture: Least Squares

We found that, under the Gaussian error model, the distribution of the OLS estimator *itself* is multivariate Normal/Gaussian.

$$\hat{\mathbf{w}} \sim N(\mathbf{w}^*, \sigma^2(\mathbf{X}^\top \mathbf{X})^{-1})$$





This motivated our study for the MVN distribution, which has properties:

1. Factorization under diagonal covariance.

2. Ellipsoidal geometry from eigendecomposition.

3. Affine transformations bridge standard MVN and general MVN.

What about the rest of ML?

OLS and GD as a “Home Base”

What about the rest of ML?

OLS and GD as a "Home Base"

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

$$\mathbf{w}_t \leftarrow \mathbf{w}_{t-1} - \eta \nabla f(\mathbf{w}_{t-1})$$

Extension 1: Nonlinear Models

Feature transformations

Nonlinear Models

Feature Transformations

Now, consider the following nonlinear function, $\phi : \mathbb{R}^2 \to \mathbb{R}^3$

$$\phi(x_1, x_2) = (x_1^2, x_1 x_2, x_2^2).$$

Because $\phi(\cdot,\cdot)$ takes inputs in $\mathbb{R}^2$, we can feed it each row (sample) in our data matrix. This allows us to "transform" our data matrix to a new data matrix, $\mathbf{X}' \in \mathbb{R}^{5\times 3}$ by applying $\phi(\cdot,\cdot)$ row by row. By doing so, we are constructing 3 new features from the $d = 2$ old features.

Problem 4(e) [4 points] Find the transformed data matrix $\mathbf{X}' \in \mathbb{R}^{5\times 3}$ obtained by applying $\phi(\cdot,\cdot)$ to each of the 5 rows. Find $\mathbf{w} \in \mathbb{R}^d$ by least squares regression on $\mathbf{X}'$ and the original $\mathbf{y}$. Also compute the sum of squared residuals error of your solution, $\text{err}(\mathbf{w})$ (you should find that, now, $\text{err}(\mathbf{w}) = 0$). You may use `numpy` or any other

It turns out that the true relationship between y_i and $\mathbf{x}_i = (x_{i1}, x_{i2})$ for the data in (14) is actually:

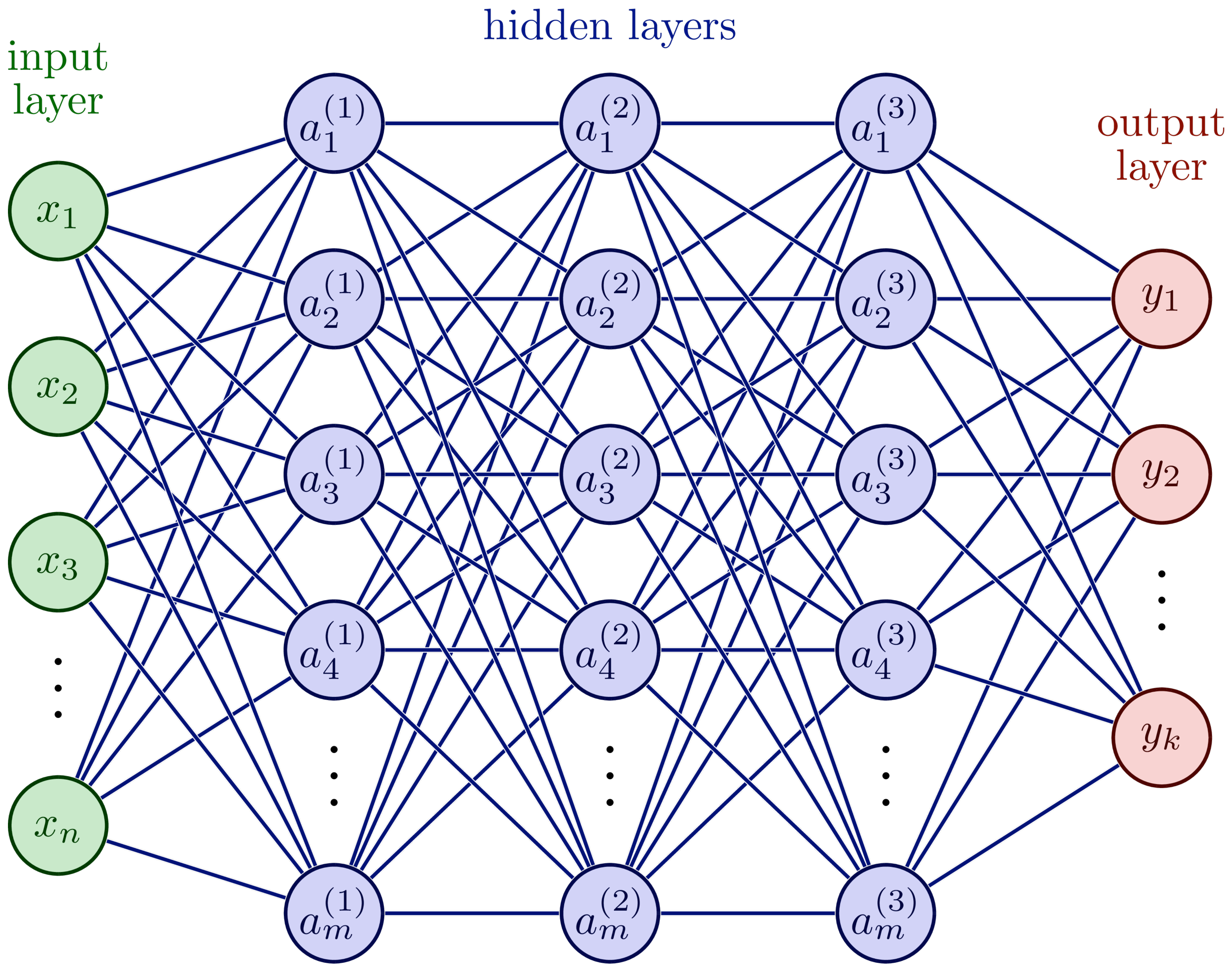
$$y_i = x_{i1}^2 + 2x_{i1}x_{i2} - x_{i2}^2 \quad \text{for all } i \in [n]. \tag{16}$$

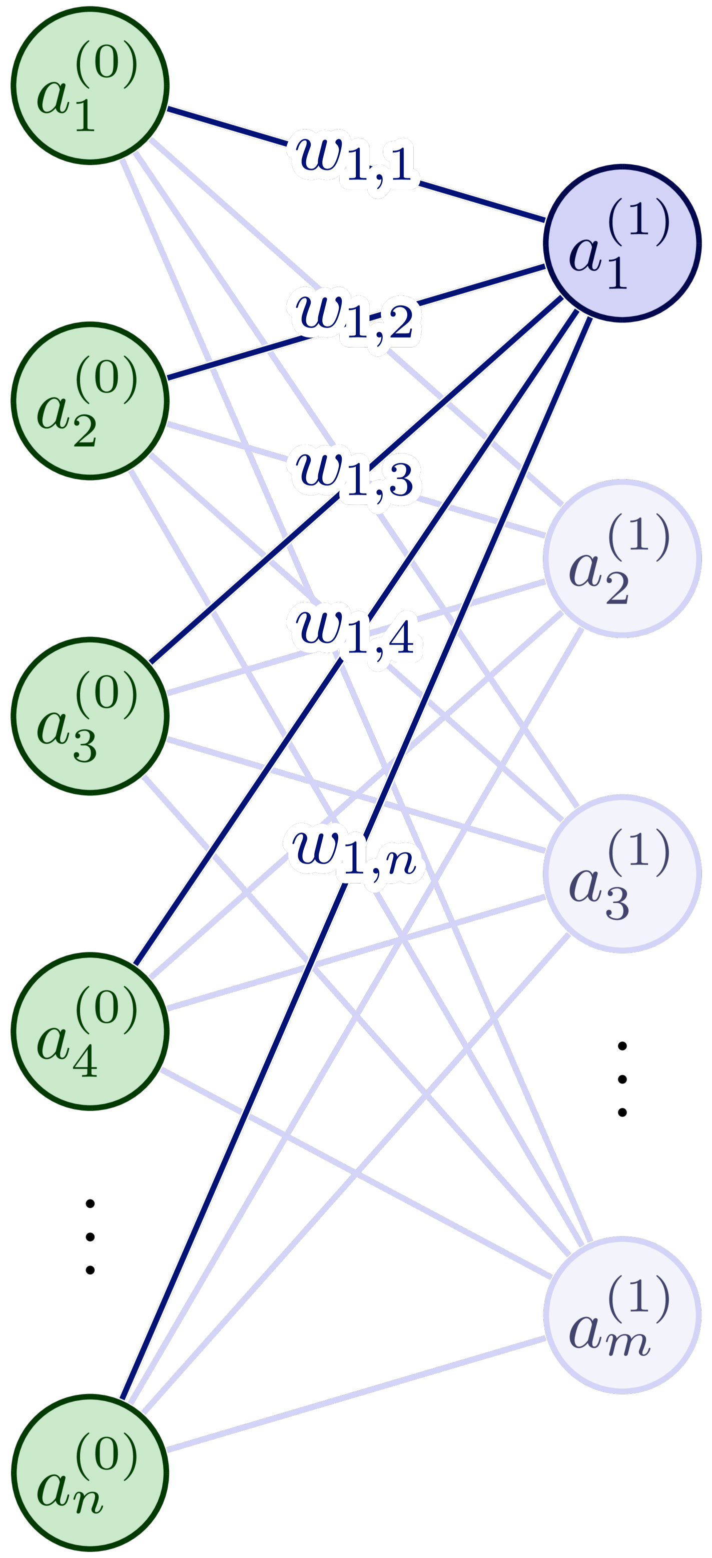
By finding the feature transformation $\phi(\cdot,\cdot)$ above, we turned a problem with a nonlinear relationship into a problem where a linear model is again useful (and, in fact, perfectly fits $\mathbf{X}'$). We are back in our ideal scenario in Equation (12), but there now exists some $\mathbf{w}^* \in \mathbb{R}^d$ such that

$$y_i = (\mathbf{w}^*)^\top \phi(\mathbf{x}_i).$$

Nonlinear Models

Neural Networks





$$= \sigma\left(w_{1,1}a_1^{(0)} + w_{1,2}a_2^{(0)} + \ldots + w_{1,n}a_n^{(0)} + b_1^{(0)}\right)$$

$$= \sigma\left(\sum_{i=1}^{n} w_{1,i}a_i^{(0)} + b_1^{(0)}\right)$$

$$\begin{pmatrix} a_1^{(1)} \\ a_2^{(1)} \\ \vdots \\ a_m^{(1)} \end{pmatrix} = \sigma\left[\begin{pmatrix} w_{1,1} & w_{1,2} & \ldots & w_{1,n} \\ w_{2,1} & w_{2,2} & \ldots & w_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{m,1} & w_{m,2} & \ldots & w_{m,n} \end{pmatrix} \begin{pmatrix} a_1^{(0)} \\ a_2^{(0)} \\ \vdots \\ a_n^{(0)} \end{pmatrix} + \begin{pmatrix} b_1^{(0)} \\ b_2^{(0)} \\ \vdots \\ b_m^{(0)} \end{pmatrix}\right]$$

$$\mathbf{a}^{(1)} = \sigma\left(\mathbf{W}^{(0)}\mathbf{a}^{(0)} + \mathbf{b}^{(0)}\right)$$

Extension 2: Loss Functions

Beyond squared loss

Loss Functions

Beyond Squared Loss

Extension 3: Algorithms

Beyond gradient descent

Algorithms

Beyond Gradient Descent

Extension 4: Learning Theory

Other issues in generalization

Learning Theory

Other issues in generalization

Thank you for listening!

Hope you enjoyed the class :)