UNIVERSITY AT ALBANY
STATE UNIVERSITY OF NEW YORK

CSI 401 (Fall 2025)

Numerical Methods

Lecture 9: Least Squares & Linear Regression

Chong Liu

Department of Computer Science

Sep 29, 2025

Agenda

- Linear model of housing price prediction
- Why least squares?
- Linear regression problem
- How to solve linear regression?
  - Direct solver
  - Gradient Descent

Case study: Housing price

- Suppose we would like to build a model predicting house prices.
  - The model takes features of a house as inputs, and outputs predicted price.

- Discussion:
  - What are the factors (features) of a house that affects its price?

- For example,
  - 8 features:

```
- MedInc        median income in block group
- HouseAge      median house age in block group
- AveRooms      average number of rooms per household
- AveBedrms     average number of bedrooms per household
- Population    block group population
- AveOccup      average number of household members
- Latitude      block group latitude
- Longitude     block group longitude
```

  - 1 label: house price

Linear model

- Take input feature vector
  - $\text{Price}(x) = w_1 x_1 + w_2 x_2 + w_3 x_3 + w_4 x_4 + \cdots$
  - x_1: median income
  - x_2: median house age
  - x_3: average number of rooms
  - x_4: average number of bedrooms
  - …
- Label space is the real number space R

Linear model

- In vector form:
  - $\text{Price}(x) = x^{\text{T}} w$
  - $x = [x_1, x_2, \ldots, x_8]$: feature vector
  - $w = [w_1, w_2, \ldots, x_8]$: parameter vector

- As long as we find a good w, we have a good linear model.

- Goal: Find a good w.

In a general form

- Xw
  - X is a $n \times d$ matrix.
    - n is the number of houses.
    - d is the number of features for describing the house.
  - w is a d-dimensional vector.

- Discussion: What else do we need to learn w?
- We need prices of these n houses!

In a general form

- Suppose y is an n-dimensional vector describing the house prices.

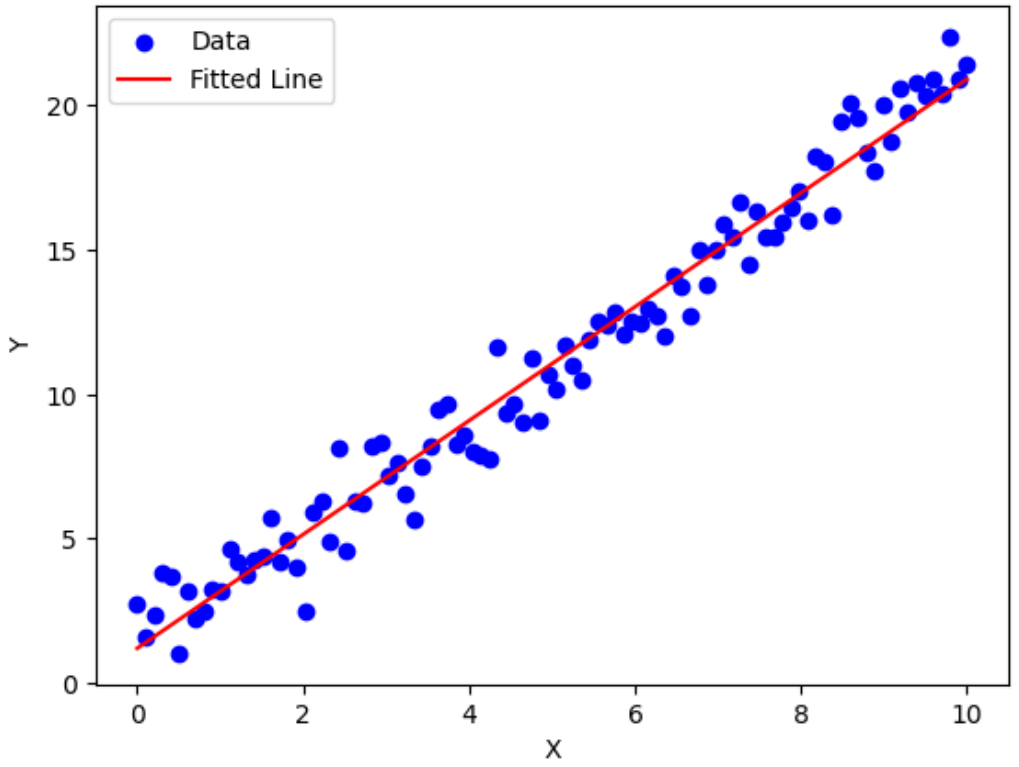
- Our goal: Solve $Xw = y$
  - X is a $n \times d$ matrix.
    - n is the number of houses.
    - d is the number of features for describing the house.
  - w is a d-dimensional vector.

- What's this?

Considering conditions of linear systems

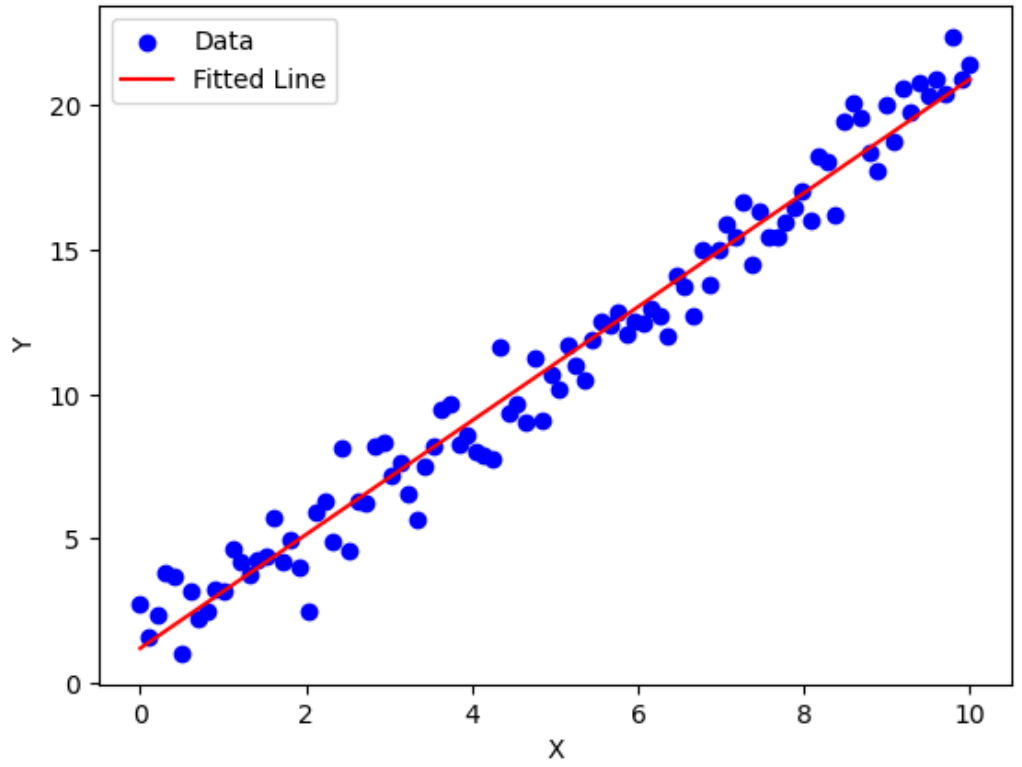
- In real-world applications, there are many challenges.
  - No solution
  - Noisy data
  - Overdetermined systems (most common case)
    - Fitting a hyperplane (a line in 2-d) to too many data points.

- Right figure:
  - x is a feature of the house
  - y is the price.



Considering conditions of linear systems

- In real-world applications, there are many challenges.
  - No solution
  - Noisy data
  - Overdetermined systems (most common case)
    - Fitting a hyperplane (a line in 2-d) to too many data points.
- So our goal reduces to find the an approximate w that best describes the data!
- How?



The objective function for learning linear regression under square loss

- $\widehat{w} = \text{argmin}_w \frac{1}{n} \sum_{i=1}^{n} (x_i^T w - y_i)^2 = \text{argmin}_w \|Xw - y\|_2^2$
  - aka: Ordinary Least Square (OLS)
- In-class exercise: solve this optimization problem by setting gradient of the objective function to 0.

Regression for different problems

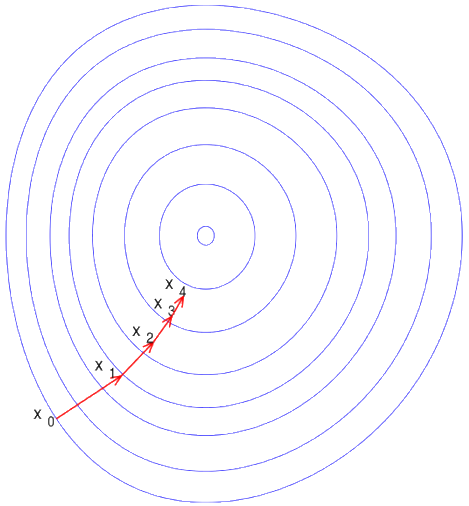
- Prediction problem
  - How well can one predict label y?
    - In housing price example: how well can one predict price given a house?

- Estimation / inference problem
  - How well can one estimate the true function?
    - Actually the true function may not be a linear function.
    - In housing price example: how well can one learn the price generating function?

Detour: How do we optimize a continuously differentiable function in general?

- The problem: $\min_{\theta} f(\theta)$
- Discussion: How do you solve this optimization problem?
- Gradient descent in iterations

$$\theta_{t+1} = \theta_t - \eta_t \nabla f(\theta_t)$$



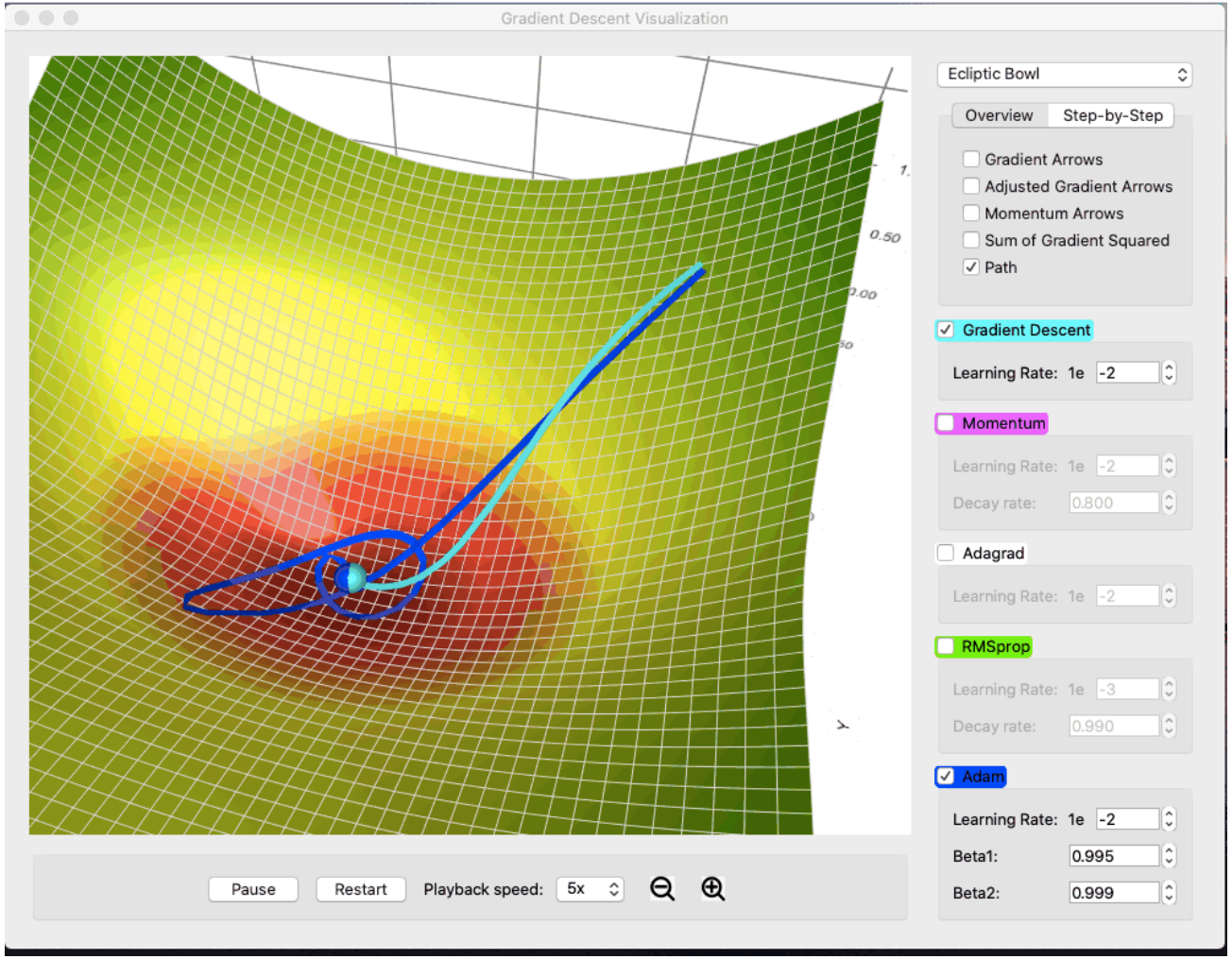
In-class exercise: gradient descent

- $\min f(x) = x^2$

1. Find x_2 given $x_0 = 2, \eta = 0.1$
2. Find x_2 given $x_0 = 2, \eta = 0.4$
3. Find x_2 given $x_0 = 4, \eta = 0.4$
4. Find x_2 given $x_0 = 2, \eta = 1.5$

Gradient Descent Demo in 2-D

- An excellent demo tool:
  - https://github.com/lilipads/gradient_descent_viz



Back to linear regression: How to solve it using Gradient Descent?

- $\widehat{w} = \text{argmin}_w \frac{1}{n} \sum_{i=1}^{n} (x_i^T w - y_i)^2 = \text{argmin}_w \|Xw - y\|_2^2$

- In-class exercise: Write the GD updating rule for solving w.
  - $w \leftarrow w - 2\eta X^T (Xw - y)$

Summary

- Least square:
  - Heavily used in practice, due to
    - Large datasets (many data points)
    - Noisy data
    - No solution based on conditions of linear systems
- Linear regression
  - $\widehat{w} = \text{argmin}_w \frac{1}{n} \sum_{i=1}^{n} (x_i^T w - y_i)^2 = \text{argmin}_w \|Xw - y\|_2^2$
  - Direct solver: $\widehat{w} = (X^T X)^{-1} X^T y$
  - GD: $w \leftarrow w - 2\eta X^T (Xw - y)$