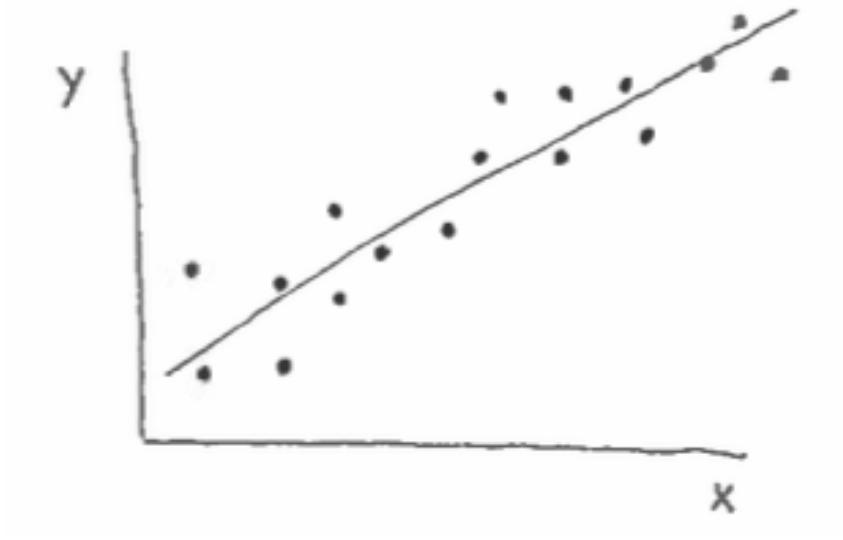
Gradient Descent

E190BO

2026-09-02

Goal: fit regression line to data



Data: (x_i, y_i), $i = 1, 2, \ldots, N$

Model: $\hat{y}_i = ax_i + b$

Loss: $MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$

Q: How to select model weights?

Approach #1: Analytical

-
- only works with

Approach #2: Gradient Descent

- can work with
- only guarantees
-

Gradient descent is an iterative approach to optimization.

1. start with
2. take step in direction of
3. repeat until

Pseudocode

Pick $w^{(0)}$ randomly

```
for i in range(numIterations):
```

$$w^{(i+1)} = w^{(i)} - \gamma \cdot \left.\frac{\partial e}{\partial w}\right|_{w=w^{(i)}}$$

- $w^{(i+1)}$: updated estimate
- $w^{(i)}$: current estimate
- γ: learning rate
- $\frac{\partial e}{\partial w}$: gradient of loss w.r.t. parameters, gradient is steepest ascent, negative gradient is steepest descent
- $\left|_{w=w^{(i)}}\right.$: evaluate gradient at the current estimate

Exercise

Model: $\hat{y}_i = ax_i + b$

Loss: $MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$

(a) Derive gradient
(b) Implement in notebook
(c) Plot MSE vs iteration for $\gamma = 1, 0.1, 0.01$

(a) Gradient Derivation

$$e = \frac{1}{N}\sum_{i=1}^{N}(y_i - ax_i - b)^2$$

$$\frac{\partial e}{\partial a} =$$

$$\frac{\partial e}{\partial b} =$$

(b) Implementation

```
w = [0, 0] # a, b
for i in range(numIterations):
    agrad =
    bgrad =
    w =
```

Goal: Fit regression line to data (m-dim)

	scalar	matrix	comments
Data	$(\bar{x}_i, y_i)$, $i = 1, 2, ..., N$ $\bar{x}_i \in \mathbb{R}^m$, $y_i \in \mathbb{R}$	$X \in \mathbb{R}^{N \times m}$, $\bar{y} \in \mathbb{R}^{N \times 1}$	
Model	$\hat{y}_i = \sum_{j=0}^{m-1} w_j x_{ij}$	$\hat{\bar{y}} = X\bar{w}$, $\bar{w} \in \mathbb{R}^{m \times 1}$	can add constant 1 as feature to
Loss	$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$	$\frac{1}{N}(\bar{y}-\hat{\bar{y}})^T(\bar{y}-\hat{\bar{y}})$	

Derive gradient:

$$e = \frac{1}{N}(\bar{y} - X\bar{w})^T(\bar{y} - X\bar{w})$$

$$= \frac{1}{N}\bar{y}^T\bar{y} - \frac{2}{N}\bar{w}^T X^T\bar{y} + \frac{1}{N}\bar{w}^T X^T X\bar{w}$$

$$\frac{\partial e}{\partial \bar{w}} = -\frac{2}{N}X^T\bar{y} + \frac{2}{N}X^T X\bar{w}$$

Gradient Descent:

initialize w randomly

for i in range(numIterations):

$$\text{wgrad} = -\frac{2}{N} X^T y + \frac{2}{N} X^T X \bar{w}$$

$$w = w - \gamma \cdot \text{wgrad}$$

Key ideas

- gradient descent is a flexible + powerful way to fit a model
- define model & loss $\rightarrow$ derive gradient $\rightarrow$ gradient descent