

Linear Regression

I. Lý thuyết:

- 1) So sánh L1 loss (absolute error) và L2 loss (squared error). Nêu ưu nhược điểm của chúng.
- 2) Tìm hiểu về Huber loss và cách Huber loss khắc phục nhược điểm của 2 hàm loss trên.

II. Thực hành code

- 3) Cài đặt linear Regression (vectorization), chạy batch theo 1 sample. Và đánh giá nếu chạy batch theo 1 sample thì sẽ như thế nào?

Lưu ý: Hiện khá nhiều cách biến đổi ma trận, tuy nhiên để được thống nhất ta sẽ biến đổi theo Mindset như sau:

Thường khi đọc dữ liệu thì x và y có dạng (**phù hợp với dữ liệu đầu vào**):

$$x = \begin{bmatrix} \text{sample 1} \\ \text{sample 2} \\ \text{sample 3} \\ \dots \\ \text{sample n} \end{bmatrix} = \begin{bmatrix} x_1^{(1)} & \dots & x_k^{(1)} \\ x_1^{(2)} & \dots & x_k^{(2)} \\ \dots & \dots & \dots \\ x_1^{(N)} & \dots & x_k^{(N)} \end{bmatrix}, y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \dots \\ y_n \end{bmatrix}$$

Cài đặt thuật toán cho m sample:

$$X = \begin{bmatrix} x_1^{(1)} & \dots & x_k^{(1)} & 1 \\ x_1^{(2)} & \dots & x_k^{(2)} & 1 \\ \dots & \dots & \dots & \dots \\ x_1^{(m)} & \dots & x_k^{(m)} & 1 \end{bmatrix} \quad (\text{shape} = m \times (k+1), m < N)$$

$$\vec{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \dots \\ y_m \end{bmatrix} \quad (\text{shape} = m \times 1)$$

$$\vec{w} = \begin{bmatrix} w_1 \\ w_2 \\ \dots \\ w_k \\ b \end{bmatrix} \quad (\text{shape} = (k+1) \times 1)$$

Forward:

$$\hat{y} = X * \vec{w} = \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \hat{y}_3 \\ \dots \\ \hat{y}_m \end{bmatrix} \quad (\text{shape} = (m \times (k+1)) \times ((k+1) \times 1) = m \times 1)$$

Loss:

$$L_{tb} = \frac{1}{m} \sum_{1 \leq i \leq m} (\hat{y}_i - y_i)^2 \quad (\text{shape} = 1 \times 1)$$

Gradient:

$$\frac{dL}{d\vec{w}} = x^T * \frac{2}{m} \sum_{1 \leq i \leq m} (\hat{y}_i - y_i) \quad (\text{shape} = \text{shape}(\vec{w}) = (k+1) \times 1)$$

Update w:

$$\vec{w} = \vec{w} - \eta \frac{dL}{d\vec{w}}$$

DATA:

https://drive.google.com/drive/folders/1MvW5G41n0BGWYVD6U0h_VVf4TDiLhARE?usp=sharing