Tóm tắt Linear Regression

Công thức Gradient Descent

$$\frac{dJ}{dw} = X^T * (\hat{y} - y)$$

MSE Loss fucntion

$$MSE = \frac{\sum_{i=1}^{n}(y - \hat{y})^2}{n}$$

Giải bài toán bằng đại số tuyến tính

$w = [w0, w_1, w_2, \dots, w_m]^T =$ là vector hệ số cần phải tối ưu, w0 thường được gọi là bias.

⇨ *Giải thích : w là ma trận (m+1) hàng và 1 cột để lưu các hệ số cần tìm*

$x^{(i)} = [1, x_1^i, x_2^i, \dots, x_m^i]$ là hàng dữ liệu thứ i trong bộ n số lượng dữ liệu quan sát được, mỗi dữ liệu có m giá trị .

⇨ *Giải thích : X là ma trận đầu vào có n mẫu dữ liệu và m biến đầu vào (feature) thì ta sẽ tạo ra ma trận n hàng (m + 1) cột (do thêm cột 1 ở đầu cho tham số bias w0)*

⇨ *Ta sẽ có ma trận X =* $\begin{bmatrix} 1 & x_1^1 & \dots & x_m^1 \\ 1 & x_1^2 & \dots & x_m^2 \\ \dots & \dots & \dots & \dots \\ 1 & x_1^n & \dots & x_m^n \end{bmatrix}$ *(ma trận n hàng và (m+1) cột) (i chạy từ 1 đến n)*

Ta có : Giá trị dự đoán y hat của thàng dữ liệu thứ i là $\widehat{y^{(i)}} = x^{(i)}w$

$X \in R^{n*(m+1)}, w \in R^{(m+1)*1}, y \in R^{n*1}$

$$Ta\ có :\ X = \begin{bmatrix} x^{(1)} \\ x^{(2)} \\ \dots \\ x^{(n)} \end{bmatrix} = \begin{bmatrix} 1 & x_1^1 & \dots & x_m^1 \\ 1 & x_1^2 & \dots & x_m^2 \\ \dots & \dots & \dots & \dots \\ 1 & x_1^n & \dots & x_m^n \end{bmatrix}, w = \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ \dots \\ w_m \end{bmatrix}, \hat{y} = X.w = \begin{bmatrix} w_0 + w_1x_1^1 + w_2x_2^1 + \dots + w_mx_m^1 \\ w_0 + w_1x_1^2 + w_2x_2^2 + \dots + w_mx_m^2 \\ \dots \\ w_0 + w_1x_1^n + w_2x_2^n + \dots + w_mx_m^n \end{bmatrix}$$

$$y = \begin{bmatrix} y^1 \\ y^2 \\ \dots \\ y^n \end{bmatrix}, y - \hat{y} = \begin{bmatrix} y^1 - (w_0 + w_1x_1^1 + w_2x_2^1 + \dots + w_mx_m^1) \\ y^2 - (w_0 + w_1x_1^2 + w_2x_2^2 + \dots + w_mx_m^2) \\ \dots \\ y^n - (w_0 + w_1x_1^n + w_2x_2^n + \dots + w_mx_m^n) \end{bmatrix}$$

⇨ *Giải thích : X là ma trận* $n \times (m + 1)$ *, w là ma trận* $(m + 1) \times 1, \hat{y} = X.w$ *sẽ là ma trận* $n \times 1$

Theo như hàm mất mát MSE Loss function :

$MSE = \frac{\sum_{i=1}^{n}(y - \hat{y})^2}{n}$ **hoặc** $MSE = \frac{1}{2} * \frac{\sum_{i=1}^{n}(y - \hat{y})^2}{n}$

Ta có : $L = \frac{1}{2} * \frac{1}{n} * \sum_{i=1}^{n}\left(y^i - \widehat{y^i}\right)^2 = \frac{1}{2} * \frac{1}{n} * \sum_{i=1}^{n}\left(y^i - x^{(i)}.w\right)^2$

⇨ *Giải thích : MES cho ma trận X là mất mát của n hàng dữ liệu*

Định nghĩa Euclideannorm : $\|z\|_2 = (z_1^2 + z_2^2 + \dots + z_n^2)^{\frac{1}{2}} \Longrightarrow \|z\|_2^2 = (z_1^2 + z_2^2 + \dots + z_n^2)$

- ⇨ *Giải thích : Trong công thức Euclideannorm thì* $\|z\|_2^2$ *, số 2 ở dưới là chuẩn 2 tức là căn bậc 2 và số 2 ở trên là mũ 2 và mũ hai của căn bậc 2 tức là hết căn có nghĩa là đơn giản ta chỉ bình phương của căn thôi .*

Mặc khác ta có :

$$L = \frac{1}{2} * \frac{1}{n} * \sum_{i=1}^{n} \left(\mathrm{y}^{\mathrm{i}} - \widehat{y^{\iota}}\right)^2 = \frac{1}{2} * \frac{1}{n} * \left[\left(\mathrm{y}^1 - \widehat{y^1}\right)^2 + \left(\mathrm{y}^2 - \widehat{y^2}\right)^2 + \cdots + \left(\mathrm{y}^{\mathrm{n}} - \widehat{y^n}\right)^2\right]$$

Đặt $\left(\mathrm{y}^{\mathrm{i}} - \widehat{y^{\iota}}\right)^2 = z_i^2$ Ta được $L = \frac{1}{2} * \frac{1}{n} * (z_1^2 + z_2^2 + \cdots + z_n^2) = \frac{1}{2} * \frac{1}{n} * \|z\|_2^2$

Viết lại

$$L = \frac{1}{2} * \frac{1}{n} * \sum_{i=1}^{n} \left(\mathrm{y}^{\mathrm{i}} - \widehat{y^{\iota}}\right)^2 = \frac{1}{2} * \frac{1}{n} * \left[\left(\mathrm{y}^1 - \widehat{y^1}\right)^2 + \left(\mathrm{y}^2 - \widehat{y^2}\right)^2 + \cdots + \left(\mathrm{y}^{\mathrm{n}} - \widehat{y^n}\right)^2\right]$$

$$= \frac{1}{2} * \frac{1}{n} * \|y - \hat{y}\|_2^2 = \frac{1}{2} * \frac{1}{n} * \|y - Xw\|_2^2 = \frac{1}{2} * \frac{1}{n} * (y - \hat{y})^T * (y - \hat{y})$$

- ⇨ *Giải thích :* $\|y - \hat{y}\|_2^2 = \left(\mathrm{y}^1 - \widehat{y^1}\right)^2 + \left(\mathrm{y}^2 - \widehat{y^2}\right)^2 + \cdots + \left(\mathrm{y}^{\mathrm{n}} - \widehat{y^n}\right)^2$
- ⇨ *Ta thấy :* $y - \hat{y} = \begin{bmatrix} y^1 - \widehat{y^1} \\ y^2 - \widehat{y^2} \\ \cdots \\ y^n - \widehat{y^n} \end{bmatrix} \Longrightarrow (y - \hat{y})^T = \begin{bmatrix} y^1 - \widehat{y^1} & y^2 - \widehat{y^2} & \ldots & y^n - \widehat{y^n} \end{bmatrix}$
- ⇨ *Mà* $(y - \hat{y})^T * (y - \hat{y}) = \begin{bmatrix} y^1 - \widehat{y^1} & y^2 - \widehat{y^2} & \ldots & y^n - \widehat{y^n} \end{bmatrix} * \begin{bmatrix} y^1 - \widehat{y^1} \\ y^2 - \widehat{y^2} \\ \cdots \\ y^n - \widehat{y^n} \end{bmatrix}$

  $= (y^2 - \widehat{y^2}) * (y^2 - \widehat{y^2}) + (y^2 - \widehat{y^2}) * (y^2 - \widehat{y^2}) + \cdots + (y^n - \widehat{y^n}) * (y^n - \widehat{y^n})$

  $= \left(\mathrm{y}^1 - \widehat{y^1}\right)^2 + \left(\mathrm{y}^2 - \widehat{y^2}\right)^2 + \cdots + \left(\mathrm{y}^{\mathrm{n}} - \widehat{y^n}\right)^2$ *Nhân 2 ma trận 1×n và n×1*

Từ đó suy ra :

$$\boldsymbol{L} = \frac{\mathbf{1}}{\mathbf{2}} * \frac{\mathbf{1}}{\boldsymbol{n}} * \|\boldsymbol{y} - \hat{\boldsymbol{y}}\|_2^2 = \frac{\mathbf{1}}{\mathbf{2}} * \frac{\mathbf{1}}{\boldsymbol{n}} * \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\|_2^2 = \frac{\mathbf{1}}{\mathbf{2}} * \frac{\mathbf{1}}{\boldsymbol{n}} * (\boldsymbol{y} - \hat{\boldsymbol{y}})^{\boldsymbol{T}} * (\boldsymbol{y} - \hat{\boldsymbol{y}})$$

Bây giờ tìm đi tìm w sao cho L (giá trị mất mát) là nhỏ nhất , như thường lệ ta sẽ khảo sát , nghĩa là ta sẽ đạo hàm L theo w và tìm nghiệm w để cho $L' = 0$

Lưu ý :

+ X là một ma trận bất kì thì ta có $A = X^T * X$ thì A sẽ luôn là một ma trận đối xứng nghĩa là $A^T = A$

+ $(A * B)^T = B^T * A^T$ (công thức liên quan đến ma trận chuyển vị trong đại số tuyến tính)

Giải bài toán $L' = 0$

L = $\frac{1}{2} * \frac{1}{n} * (y - \hat{y})^T * (y - \hat{y})$. Ta có thể cho L = $(y - \hat{y})^T * (y - \hat{y})$ vì $\frac{1}{2} * \frac{1}{n}$ là hằng

L = $(y - \hat{y})^T * (y - \hat{y}) = (y - X.w)^T * (y - X.w)$

= $(y^T - w^T X^T) * (y - Xw) = \boldsymbol{y^T y} - \boldsymbol{y^T X w} - \boldsymbol{w^T X^T y} + \boldsymbol{w^T X^T X w}$

Đạo hàm :

$L'_w = -\,y^TX\; - [(Xw)^Ty + w^TX^TXw]'_w = -y^TX - [(Xw)^T(y^T)^T + w^TX^TXw]'_w$

$= -\,y^TX - [(y^TXw)^T + w^TX^TXw]'_w \;\; = -\,y^TX - [(y^TXw)^T + w^TX^TXw\,]'_w$

$= -\,y^TX - [(y^TXw)^T\,]'_w + [w^TX^TXw\,]'_w$

$= -\,y^TX - [(y^TX)w]'_w + [w^TX^TXw\,]'_w$

$= -\,y^TX - y^TX + [w^TX^TXw\,]'_w$

$= -2y^TX + [w^TX^TXw\,]'_w$

⇨ Cần đi tìm $[w^TX^TXw\,]'_w$

Tìm $[w^TX^TXw\,]'_w$:

$$+\,x^{(i)}w = [1 \quad x_1^i \quad \dots \quad x_m^i] * \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ \dots \\ w_m \end{bmatrix} = w_0 + w_1x_1^i + w_2x_2^i + \dots + w_mx_m^i$$

$$\Longrightarrow \frac{d\left(x^{(i)}w\right)}{dw} = \left[\frac{d\left(x^{(i)}w\right)}{dw0} \quad \frac{d\left(x^{(i)}w\right)}{dw_1} \quad \dots \quad \frac{d\left(x^{(i)}w\right)}{dw_m}\right] = [1 \quad x_1^1 \quad \dots \quad x_m^i] = x^{(i)}$$

$$\Longrightarrow \frac{d(Xw)}{dw} = \begin{bmatrix} \frac{d\left(x^{(1)}w\right)}{dw0} & \frac{d\left(x^{(1)}w\right)}{dw_1} & \dots & \frac{d\left(x^{(1)}w\right)}{dw_m} \\ \frac{d\left(x^{(2)}w\right)}{dw0} & \frac{d\left(x^{(2)}w\right)}{dw_1} & \dots & \frac{d\left(x^{(2)}w\right)}{dw_m} \\ \dots & \dots & \dots & \dots \\ \frac{d\left(x^{(n)}w\right)}{dw0} & \frac{d\left(x^{(n)}w\right)}{dw_1} & \dots & \frac{d\left(x^{(n)}w\right)}{dw_m} \end{bmatrix} = \begin{bmatrix} 1 & x_1^1 & \dots & x_m^1 \\ 1 & x_1^2 & \dots & x_m^2 \\ \dots & \dots & \dots & \dots \\ 1 & x_1^n & \dots & x_m^n \end{bmatrix} = X$$

$$+\,wx^{(i)} = \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ \dots \\ w_m \end{bmatrix} * [1 \quad x_1^i \quad \dots \quad x_m^i] = \color{red}{w_0 + w_1x_1^i + w_2x_2^i + \dots + w_mx_m^i}$$

$$= \color{green}{\begin{bmatrix} w_0 & w_0x_1^i & \dots & w_0x_m^i \\ w_1 & w_1x_1^i & \dots & w_1x_m^i \\ \dots & \dots & \dots & \dots \\ w_m & w_mx_1^i & \dots & w_mx_m^i \end{bmatrix}}$$

$$\Longrightarrow \frac{d\left(wx^{(i)}\right)}{dw} = \begin{bmatrix} \frac{d\left(wx^{(i)}\right)}{dw0} \\ \frac{d\left(wx^{(i)}\right)}{dw_1} \\ \dots \\ \frac{d\left(wx^{(i)}\right)}{dw_m} \end{bmatrix} = \begin{bmatrix} 1 \\ x_1^i \\ \dots \\ x_m^i \end{bmatrix} = (x^{(i)})^T$$

$$\Longrightarrow \frac{d(Xw)}{dw} = \begin{bmatrix} \frac{d(wx^{(1)})}{dw_0} & \frac{d(x^{(2)}w)}{dw_0} & \dots & \frac{d(x^{(n)}w)}{dw_0} \\ \frac{d(wx^{(1)})}{dw_1} & \frac{d(x^{(2)}w)}{dw_1} & \dots & \frac{d(x^{(n)}w)}{dw_1} \\ \dots & \dots & \dots & \dots \\ \frac{d(wx^{(1)})}{dw_m} & \frac{d(x^{(2)}w)}{dw_m} & \dots & \frac{d(x^{(n)}w)}{dw_m} \end{bmatrix} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ x_1^1 & x_1^2 & \dots & x_1^n \\ \dots & \dots & \dots & \dots \\ x_m^1 & x_m^2 & \dots & x_m^n \end{bmatrix} = X^T$$

$+ \frac{d(w^TXw)}{dw} = w^TX + \frac{d((Xw)^Tw)}{dw} = w^TX + w^TX^T = w^T(X + X^T) = 2w^TX$ (Nếu X đối xứng thì $X^T = X$)

Quay lại bài toán $L'_w = 0$

Giải thích : Đặt $X^TX = H$. Mà H là ma trận đối xứng .

Áp dụng kết quả đã được chúng minh ở trên $\frac{d(w^TXw)}{dw} = 2w^TX \leftrightarrow \frac{d(w^THw)}{dw} = 2w^TH = 2w^TX^TX$

Ta có :

$$L'_w = -2y^TX + [w^TX^TXw]'_w = -2y^TX + \frac{d(w^TX^TXw)}{dw} = -2y^TX + 2w^TX^TX = 0$$

- ⇨ $2w^TX^TX = 2y^TX$
- ⇨ $w^TX^TX = y^TX$
- ⇨ $(Xw)^TX = (X^Ty)^T$
- ⇨ $[X^T(Xw)]^T = (X^Ty)^T$
- ⇨ $X^T(Xw) = X^Ty$
- ⇨ $(X^TX)w = X^Ty$
- ⇨ $\boldsymbol{w = (X^TX)^{-1}X^Ty}$

Kết luận : Nếu X^TX khả nghịch (có ma trận nghịch đảo) , thì L có nghiệm duy nhất : $\boldsymbol{w = (X^TX)^{-1}X^Ty}$

Nếu không khả nghịch, ta có thể sử dụng khái niệm giả nghịch đảo.

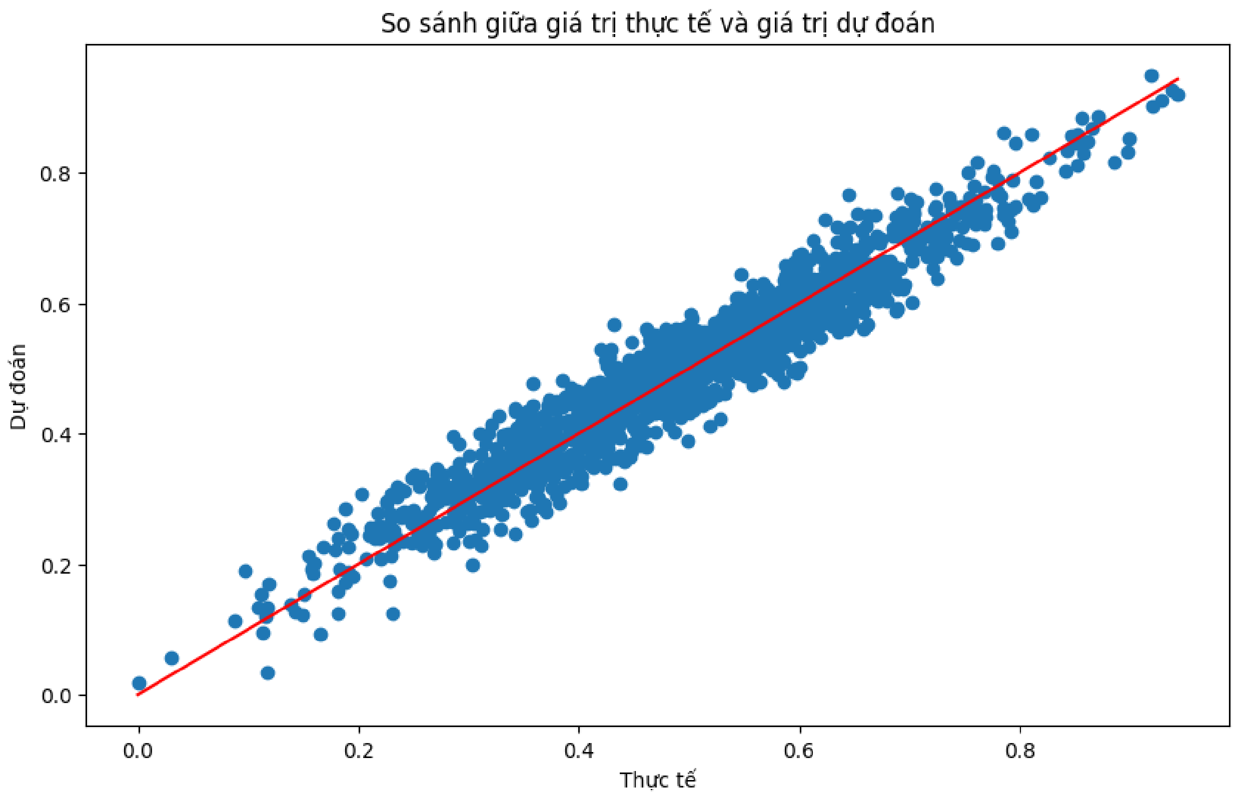
Như vậy ta đã có hàm Gradient descent $\frac{\mathbf{dJ}}{\mathbf{dw}} = \mathbf{X^T} * (\hat{\mathbf{y}} - \mathbf{y})$ và nghiệm $\boldsymbol{w = (X^TX)^{-1}X^Ty}$

- ⇨ Sử dụng công thức w = w – learning_rate * gradient sẽ thu được w tối ưu sau **epochs** lần lặp
- ⇨ Hoặc có thể cho nó dừng sớm nếu w_{old} và w_{new} chênh lệch không quá **epsilon** .

Code Linear Regression

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

def matrix_transformation(x, y) :
    x = np.concatenate((np.ones((x.shape[0], 1)), x), axis=1)
    y = np.array(y).reshape(-1, 1)
    return x, y

def linear_regression(X, y, lr=0.05, epsilon=1e-10, epochs=100000):
```

```python
    ep = 0
    N = X.shape[0]
    w = np.linalg.pinv(X.T.dot(X)).dot(X.T).dot(y)
    while ep < epochs:
        gradient = np.dot(X.T, (np.dot(X, w) - y)) / N
        w = w - lr * gradient
        if np.linalg.norm(gradient) < epsilon:
            break
        ep += 1
    return w

def predict(X, w):
    predictions = X.dot(w)
    return predictions

data = pd.read_csv("USA_Housing.csv")
data.head()
X = data[['Avg. Area Income', 'Avg. Area House Age', 'Avg. Area Number of Rooms',
'Avg. Area Number of Bedrooms', 'Area Population']]
y = data['Price']
np.random.seed(42)
random_indices = np.random.permutation(len(X))
train_size = int(0.7 * len(X))

X_train = X.iloc[random_indices[:train_size]]
y_train = y[random_indices[:train_size]]
X_test = X.iloc[random_indices[train_size:]]
y_test = y[random_indices[train_size:]]

X_train, y_train = matrix_transformation(X_train, y_train)
X_test, y_test = matrix_transformation(X_test, y_test)
w = linear_regression(X_train, y_train, lr=0.05, epsilon=1e-10, epochs=100000)
predictions = predict(X_test, w)

plt.figure(figsize=(10, 6))
plt.scatter(y_test, predictions)
plt.plot([y_test.min(), y_test.max()], [y_test.min(), y_test.max()], color='red')
plt.xlabel('Thực tế')
plt.ylabel('Dự đoán')
plt.title('So sánh giữa giá trị thực tế và giá trị dự đoán')
plt.show()
```



So sánh giữa giá trị thực tế và giá trị dự đoán
Dự đoán
Thực tế
0.0
0.2
0.4
0.6
0.8