Supervised Machine Learning
Regression & Classification

Supervised Learning

Regression : House Price Prediction

↑

Predict a number, infinitely many possible outputs.

Classification : Cat & Dogs / Breast Cancer detection

↑

Predict categories, small number of possible outputs

Unsupervised Learning

Clustering - Group similar data points together

Anomaly detection

Dimensionality reduction

Linear Regression with One Variable

Terminology

x = input variable / feature

y : output variable / target variable

(x, y) : single training example

$(x^{(i)}, y^{(i)})$: i^{th} training example

$\hat{y}$: estimate / prediction / estimated y

for linear regression,

$$\underbrace{f_{w,b}}_{f(x)}(x) = wx + b \qquad (\underbrace{\text{Univariate}}_{\text{One variable}} \text{ linear regression})$$

Cost function

(Squared error cost function) $\quad \hat{y}^{(i)} = f_{w,b}(x^{(i)})$

$$J(w,b) = \frac{1}{2m} \sum_{i=1}^{m} \left(f_{x,b}(x^{(i)}) - y^{(i)} \right)^2 \qquad f_{w,b}(x^{(i)}) = wx^{(i)} + b$$

for all m (number of training examples)

Gradient Descent

minimize $J(w,b)$

$$tmp\_w = w - \alpha \frac{\partial}{\partial w} J(w,b)$$

$$tmp\_b = w - \alpha \frac{\partial}{\partial b} J(w,b)$$

$$w = tmp\_w$$

$$b = tmp\_b$$

Linear regression model

$$f_{w,b}(x) = wx + b$$

Cost function

$$J(w,b) = \frac{1}{2m} \sum_{i=1}^{m} \left(f_{w,b}(x^{(i)}) - y^{(i)} \right)^2$$

Gradient descent algorithm

repeat until convergence {

$$w = w - \alpha \boxed{\frac{\partial}{\partial w} J(w,b)} \longrightarrow \frac{1}{m} \sum_{i=1}^{m} \left(f_{w,b}(x^{(i)}) - y^{(i)} \right) x^{(i)}$$

$$b = b - \alpha \boxed{\frac{\partial}{\partial b} J(w,b)} \longrightarrow \frac{1}{m} \sum_{i=1}^{m} \left(f_{w,b}(x^{(i)}) - y^{(i)} \right)$$

}

'Batch' gradient descent

Batch : Each step of gradient descent uses all the training examples.

Multiple Features

x_j = j^{th} feature

n : number of features

$\vec{x}^{(i)}$: features of i^{th} trainig example (vector row)

$x_j^{(i)}$: value of feature j in i^{th} training example

$$f_{w,b}(x) = w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4 + b$$

$\vec{w} = [w_1 \ w_2 \ \dots \ w_n]$

b is a number

$\vec{x} = [x_1 \ x_2 \ \dots \ x_n]$

$$f_{\vec{w},b} = \vec{w} \cdot \vec{x} + b$$

Vectorization

$f_{\vec{w},b}(\vec{x}) = \vec{w} \cdot \vec{x} + b$

f = np.dot (w, x) + b

Gradient descent

$w = w - \alpha * d$

Derivative

Gradient Descent for multiple regression

parameters

$\vec{w} = [w_1 \; \cdots \; w_n]$

b

$f_{\vec{w},b}(\vec{x}) = \vec{w} \cdot \vec{x} + b$ ← Model

Cost function $J(\vec{w}, b)$

Gradient descent

$$\text{repeat} \; \{ \quad w_j = w_j - \alpha \frac{\partial}{\partial w_j} J(\vec{w}, b) \quad b = b - \alpha \frac{\partial}{\partial b} J(\vec{w}, b) \quad \}$$

for multiple features

$$w_1 = w_1 - \alpha \underbrace{\frac{1}{m} \sum_{i=1}^{m} \left(f_{\vec{w},b}(x^{(i)}) - y^{(i)}\right) x_1^{(i)}}_{\frac{\partial}{\partial w_1} J(\vec{w}, b)}$$

$\vdots$

$$w_n = w_n - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(f_{\vec{w},b}(x^{(i)}) - y^{(i)}\right) x_n^{(i)}$$

$$b = b - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(f_{\vec{w},b}(x^{(i)}) - y^{(i)}\right)$$

Alternative to gradient descent

Normal equation

$$\theta = \left(X^T X + \lambda \underbrace{\begin{bmatrix} 0 & \\ & 0 \end{bmatrix}}_{(n+1)(n+1)}\right)^{-1} X^T y$$

Feature Scaling

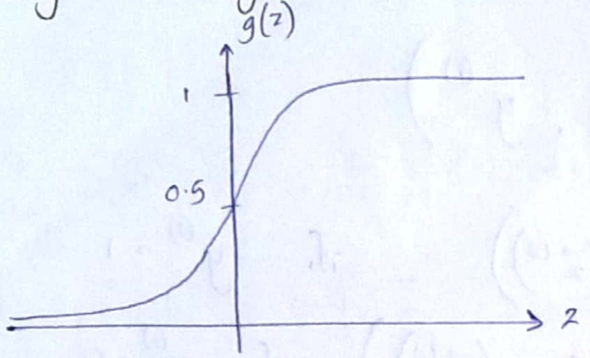
Mean Normalization

$$\frac{x_1^{(i)} - \mu_1}{Range}$$

Z - score normalization

$$\frac{x_1^{(i)} - \mu_1}{\sigma_1}$$

Logistic Regression



$$g(z) = \frac{1}{1 + e^{-z}}$$

$$0 < g(z) < 1$$

sigmoid function

logistic function

$$f_{\vec{w},b}(\vec{x}) \longrightarrow z = \vec{w} \cdot \vec{x} + b$$

$$g(z) = \frac{1}{1 + e^{-z}}$$

$$f_{\vec{w},b}(\vec{x}) = g(\vec{w} \cdot \vec{x} + b) = \frac{1}{1 + e^{-(\vec{w} \cdot \vec{x} + b)}}$$

Cost function for logistic regression

Squared Error

$$J(\vec{w},b) = \frac{1}{m}\sum_{i=1}^{m} \underbrace{\frac{1}{2}\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)^2}_{L\left(f_{\vec{w},b}(\vec{x}^{(i)}),\, y^{(i)}\right)}$$

Logistic loss function

$$J(\vec{w},b) = \frac{1}{m}\sum_{i=1}^{m} L\left(f_{\vec{w},b}(\vec{x}^{(i)}),\, y^{(i)}\right)$$

$$\begin{cases} -\log\left(f_{\vec{w},b}(\vec{x}^{(i)})\right) & \text{if } y^{(i)} = 1 \\ -\log\left(1 - f_{\vec{w},b}(\vec{x}^{(i)})\right) & \text{if } y^{(i)} = 0 \end{cases}$$

Simplified Cost function

$$L\left(f_{\vec{w},b}(x^{(i)}),\, y^{(i)}\right) = -y^{(i)}\log\left(f_{\vec{w},b}(\vec{x}^{(i)})\right) - \left(1-y^{(i)}\right)\log\left(1 - f_{\vec{w},b}(x^{(i)})\right)$$

Loss function

Cost function

$$J(\vec{w},b) = \frac{-1}{m}\sum_{i=1}^{m}\left[y^{(i)}\log\left(f_{\vec{w},b}(x^{(i)})\right) + \left(1-y^{(i)}\right)\log\left(1 - f_{\vec{w},b}(x^{(i)})\right)\right]$$

Gradient Descent

repeat {

$j = 1, \dots, n$

$$w_j = w_j - \alpha \frac{\partial}{\partial w_j} J(\vec{w}, b)$$

$$b = b - \alpha \frac{\partial}{\partial b} J(\vec{w}, b)$$

}

$$\frac{\partial}{\partial w_j} J(\vec{w}, b) = \frac{1}{m} \sum_{i=1}^{m} \left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)} \right) x_j^{(i)}$$

$$\frac{\partial}{\partial b} J(\vec{w}, b) = \frac{1}{m} \sum_{i=1}^{m} \left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)} \right)$$

Overfitting

Underfit $\rightarrow$ High bias

Overfit $\longrightarrow$ High variance

Addressing Overfitting

Collect more training data

Select features to include /exclude

Regularization

CamScanner

Cost function with Regularization

$$J(\vec{w},b) = \frac{1}{2m}\sum_{i=1}^{m}\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)^2 + \frac{\lambda}{2m}\sum_{j=1}^{n} w_j^2$$

$\lambda > 0 \leftarrow$ Regularization parameter

Regularized linear regression

$$J(\vec{w},b) = \frac{1}{2m}\sum_{i=1}^{m}\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)^2 + \frac{\lambda}{2m}\sum_{j=1}^{n} w_j^2$$

Gradient descent

$$w_j = w_j - \alpha \frac{\partial}{\partial w_j} J(\vec{w},b) \qquad = \frac{1}{m}\sum_{i=1}^{m}\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)x_j^{(i)} + \frac{\lambda}{m} w_j$$

$$b = b - \alpha \frac{\partial}{\partial b} J(\vec{w},b) \qquad = \frac{1}{m}\sum_{i=1}^{m}\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)$$

Regularized logistic regression

$$J(\vec{w},b) = \frac{-1}{m}\sum_{i=1}^{m}\left[y^{(i)}\log\left(f_{\vec{w},b}(\vec{x}^{(i)})\right) + (1-y^{(i)})\log\left(1 - f_{\vec{w},b}(\vec{x}^{(i)})\right)\right] + \frac{\lambda}{2m}\sum_{j=1}^{n} w_j^2$$

* Gradient descent is same as linear regression