CS722/822: Machine Learning

Instructor: Jiangwen Sun
Computer Science Department

Last Lecture

- What is regression
- Least squares
- Different regression problems
- Statistical interpretation of least squares
- Solve least squares (to be continued)

Solve least squares

- Least squares with a linear function of $\mathbf{x}$ and parameters $\mathbf{w}$ is called “linear regression”
- Linear regression has a closed-form solution for w

$$\min_{\mathbf{w}} \sum_{i=1}^{N} \left(y_i - \mathbf{x}_i^T \mathbf{w}\right)^2$$
$$= \min_{\mathbf{w}} (\mathbf{y} - \mathbf{X}\mathbf{w})^T (\mathbf{y} - \mathbf{X}\mathbf{w})$$
$$= \min_{\mathbf{w}} E(\mathbf{w})$$

- The minimum is achieved at the zero gradient

The gradient $\frac{\partial E(\mathbf{w})}{\partial \mathbf{w}} = -2\mathbf{X}^T(\mathbf{y} - \mathbf{X}\mathbf{w}) = 0$

$$\mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

Solve least squares

- We can use the following formula

$$\mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

to build a model

1. the model can be a linear function of X
2. the model can be a polynomial of X
3. Actually, the model can be any format of

$$y_i = \phi(\mathbf{x}_i)^T\mathbf{w}$$

Let us try out some examples

Simple examples - linear

- A simple example where we observed three data points
- $(\mathbf{x}^{(1)}, y^{(1)})$, $(\mathbf{x}^{(2)}, y^{(2)})$ and $(\mathbf{x}^{(3)}, y^{(3)})$ where $\mathbf{x}^{(i)}$ is a vector of 2 elements

$$\begin{bmatrix} x_1^{(1)} & x_2^{(1)} \\ x_1^{(2)} & x_2^{(2)} \\ x_1^{(3)} & x_2^{(3)} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \approx \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ y^{(3)} \end{bmatrix} \qquad E(\mathbf{w}) = \sum_{i=1}^{3} \left(y^{(i)} - \mathbf{x}^{(i)^T} \mathbf{w} \right)^2$$

$$\frac{\partial E(\mathbf{w})}{\partial w_1} = \sum_{i=1}^{3} -2 x_1^{(i)} \left(y^{(i)} - \left(x_1^{(i)} w_1 + x_2^{(i)} w_2 \right) \right) = 0$$

$$\Rightarrow \quad \boxed{-2 \sum_{i=0}^{3} x_1^{(i)} y^{(i)} + 2 \sum_{i=0}^{3} x_1^{(i)} x_1^{(i)} \, w_1 + 2 \sum_{i=0}^{3} x_1^{(i)} x_2^{(i)} \, w_2 = 0}$$

$$\frac{\partial E(\mathbf{w})}{\partial w_2} = \boxed{-2 \sum_{i=0}^{3} x_2^{(i)} y^{(i)} + 2 \sum_{i=0}^{3} x_2^{(i)} x_1^{(i)} \, w_1 + 2 \sum_{i=0}^{3} x_2^{(i)} x_2^{(i)} \, w_2 = 0}$$

Simple examples - polynomial

- Our function is no longer linear but a polynomial of $\mathbf{x}$

- Let us assume we have one independent variable x, and we are building a polynomial of order M to approximate y

$$f(x;\mathbf{w}) = w_0 + w_1 x^1 + w_2 x^2 + \cdots + w_M x^M = \sum_{j=0}^{M} w_j x^j$$

$$\begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^M \\ 1 & x_2 & x_2^2 & \cdots & x_2^M \\ 1 & x_3 & x_3^2 & \cdots & x_3^M \end{bmatrix} \begin{bmatrix} w_0 \\ \vdots \\ w_M \end{bmatrix} \approx \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

Simple examples - polynomial

- Similarly, Least Squares has a closed form solution with linear regression, we also have the same closed form solution when the function is a polynomial

$$\mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

- where

$$\mathbf{X} = \begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^M \\ 1 & x_2 & x_2^2 & \cdots & x_2^M \\ 1 & x_3 & x_3^2 & \cdots & x_3^M \end{bmatrix}$$

Design Matrix

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

Solve least squares

- For nonlinear regression, there is no closed-form solution
- Or when the design matrix (i.e., $\mathbf{X}$) is too big, the computation cost of inverse matrix is too high

$$\mathbf{w} = \boxed{(\mathbf{X}^T\mathbf{X})^{-1}}\mathbf{X}^T\mathbf{y}$$

We use the so-called "gradient descent" algorithm

Recall: for linear regression, we set

the gradient $\frac{\partial E(\mathbf{w})}{\partial \mathbf{w}} = -2\mathbf{X}^T(\mathbf{y} - \mathbf{X}\mathbf{w}) = 0$

to obtain the solution

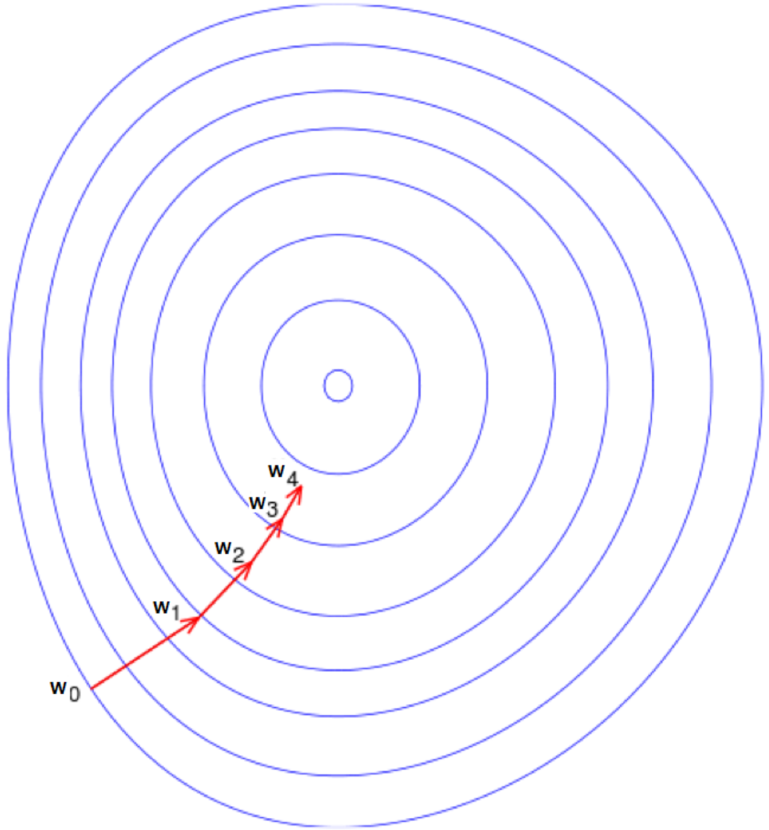
Basic idea of gradient descent

- From $\mathbf{w}_0$, at each iteration, we reduce $E(\mathbf{w})$,

  $$E(\mathbf{w}_0) \geq E(\mathbf{w}_1) \geq E(\mathbf{w}_2) \geq \cdots$$

- $\mathbf{w}_0$ can be any feasible $\mathbf{w}$
- If $E(\mathbf{w})$ is differentiable, then at any point $\mathbf{w}_k$, $E(\mathbf{w})$ decreases fastest along the direction of the negative gradient of $E(\mathbf{w})$ at $\mathbf{w}_k$,

  $$-\frac{\partial E(\mathbf{w}_k)}{\partial \mathbf{w}_k} = 2\mathbf{X}^T(\mathbf{y} - \mathbf{X}\mathbf{w}_k)$$



Algorithm of gradient descent

1. Set iteration $k = 0$, make an initial guess $\mathbf{w}_0$
2. repeat:
3. Compute the negative gradient of $E(\mathbf{w})$ at $\mathbf{w}_k$ and set it to be the search direction $\mathbf{d}_k$
4. Choose a step size α_k to sufficiently reduce $$E(\mathbf{w}_k + \alpha_k \mathbf{d}_k)$$
5. Update $\mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{d}_k$
6. $k = k + 1$
7. Until a termination rule is met