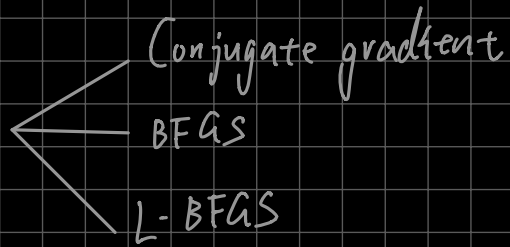


Advanced Optimization Algorithm:

minimize the $J(\theta)$



Advantages: ① No need to choose α
② faster

Disadvantages: ① Complex

How to apply:

Step 1: $\theta = \begin{bmatrix} \theta_0 \\ \vdots \\ \theta_n \end{bmatrix}_{(n+1) \times 1}$

function $[jVal, gradient] = costFunction(\theta)$

$jVal =$ [code to compute $J(\theta)$];

$gradient =$ zero $(n+1, D)$;

$gradient(1) =$ [code to compute $\frac{\partial}{\partial \theta_0} J(\theta)$];

$gradient(2) =$ [code to compute $\frac{\partial}{\partial \theta_1} J(\theta)$];

\vdots
 $gradient(n+1) =$ [code to compute $\frac{\partial}{\partial \theta_n} J(\theta)$];

3 things you need to know about Regularization:

① Regularize item is added on $J(\theta)$

② Regularize item starts from θ_1 ,

$$\frac{\lambda}{2m} \sum_{i=1}^m \theta_i^2$$

③

Regularize item is positive and have $\frac{\lambda}{2m}$ factor.