

Classification and Representation

- ✓

Video: Classification

8 min
- ✓

Reading: Classification

2 min
- ✓

Video: Hypothesis Representation

7 min
- ✓

Reading: Hypothesis Representation

3 min
- ✓

Video: Decision Boundary

14 min
- ✓

Reading: Decision Boundary

3 min

Logistic Regression Model

- ✓

Video: Cost Function

10 min
- ✓

Reading: Cost Function

3 min
- ✓

Video: Simplified Cost Function and Gradient Descent

10 min
- ✓

Reading: Simplified Cost Function and Gradient Descent

3 min
- ▶

Video: Advanced Optimization

14 min
- 📖

Reading: Advanced Optimization

3 min

Multiclass Classification

- ▶

Video: Multiclass Classification: One-vs-all

6 min
- 📖

Reading: Multiclass Classification: One-vs-all

3 min

Review

- ✓

Reading: Lecture Slides

10 min
- 📋

Quiz: Logistic Regression

5 questions

Solving the Problem of Overfitting

Review



Simplified Cost Function and Gradient Descent

Note: [6:53 - the gradient descent equation should have a 1/m factor]

We can compress our cost function's two conditional cases into one case:

$$\text{Cost}(h_{\theta}(x), y) = -y \log(h_{\theta}(x)) - (1 - y) \log(1 - h_{\theta}(x))$$

Notice that when y is equal to 1, then the second term $(1 - y) \log(1 - h_{\theta}(x))$ will be zero and will not affect the result. If y is equal to 0, then the first term $-y \log(h_{\theta}(x))$ will be zero and will not affect the result.

We can fully write out our entire cost function as follows:

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^m [y^{(i)} \log(h_{\theta}(x^{(i)})) + (1 - y^{(i)}) \log(1 - h_{\theta}(x^{(i)}))]$$

A vectorized implementation is:

$$h = g(X\theta)$$
$$J(\theta) = \frac{1}{m} \cdot (-y^T \log(h) - (1 - y)^T \log(1 - h))$$

Gradient Descent

Remember that the general form of gradient descent is:

$$\text{Repeat } \left\{ \begin{aligned} \theta_j &:= \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta) \end{aligned} \right\}$$

We can work out the derivative part using calculus to get:

$$\text{Repeat } \left\{ \begin{aligned} \theta_j &:= \theta_j - \frac{\alpha}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_j^{(i)} \end{aligned} \right\}$$

Notice that this algorithm is identical to the one we used in linear regression. We still have to simultaneously update all values in theta.

A vectorized implementation is:

$$\theta := \theta - \frac{\alpha}{m} X^T (g(X\theta) - \vec{y})$$

✓ Complete

Go to next item

