1 Evaluating a Learning Algorithm

1.1 Deciding What to Try Next

There's often a huge difference between someone that really knows how to powerfully and effectively apply that algorithm, versus someone that's less familiar with some doesn't really understand how to apply these algorithms and can end up wasting a lot of their time trying things out that don't really make sense.

Practical suggestions, advice, guidelines on how to choose one of the most promising avenues to spend your time pursuing when developing machine learning systems.

There are many things that one can think of that could improve the performance of the learning algorithm:

- Get more training examples(But sometimes getting more training data doesn't actually help)

- Try a smaller set of features

- Get additional features

- Adding polynomial features things

- Increase or decrease the regularization: λ

We need **evaluate learning algorithms** to rule out half of the things on this list as being potentially promising things to pursue. The techniques are called **Machine learning diagnostics.**

Machine learning diagnostics: A test that you can run to gain insight what is/isn't working with a learning algorithm, and gain guidance as to how best to improve its performance

1.2 Evaluating a hypothesis

1.2.1 How?

Split the data we have into two portions: The first portion is going to be our usual **training set** and the second portion is going to be our **test set**

m: denote number of training dataset (70%)

m_{test}: denote number of testset (30%)

The new procedure using these two sets is then:

1. Learn Θ and minimize $J_{train}(\Theta)$ using the training set

2. Compute the test set error $J_{test}(\Theta)$

1.2.2 The test set error

For linear regression: $J_{test}(\Theta) = \dfrac{1}{2m_{test}} \sum_{i=1}^{m_{test}} (h_\Theta(x_{test}^{(i)}) - y_{test}^{(i)})^2$

For classification   Misclassification error (aka 0/1 misclassification error):

$$err(h_\Theta(x), y) = \begin{matrix} 1 \\ 0 \end{matrix} \quad \begin{matrix} \text{if } h_\Theta(x) \geq 0.5 \text{ and } y = 0 \text{ or } h_\Theta(x) < 0.5 \text{ and } y = 1 \\ otherwise \end{matrix}$$

This gives us a binary 0 or 1 error result based on a misclassification. The average test error for the test set is:

$$\text{Test Error} = \frac{1}{m_{test}} \sum_{i=1}^{m_{test}} err(h_\Theta(x_{test}^{(i)}), y_{test}^{(i)})$$