Evaluating a Learning Algorithm

1. How to improve my learning algorithm:
    - Get more training examples: fix high variance
    - Try smaller sets of features: fix high variance
    - Try getting additional features: fix high bias
    - Try add polynomial features: fix high bias
    - Try decrease λ: fix hgih bias
    - Try increasing λ: fix high variance

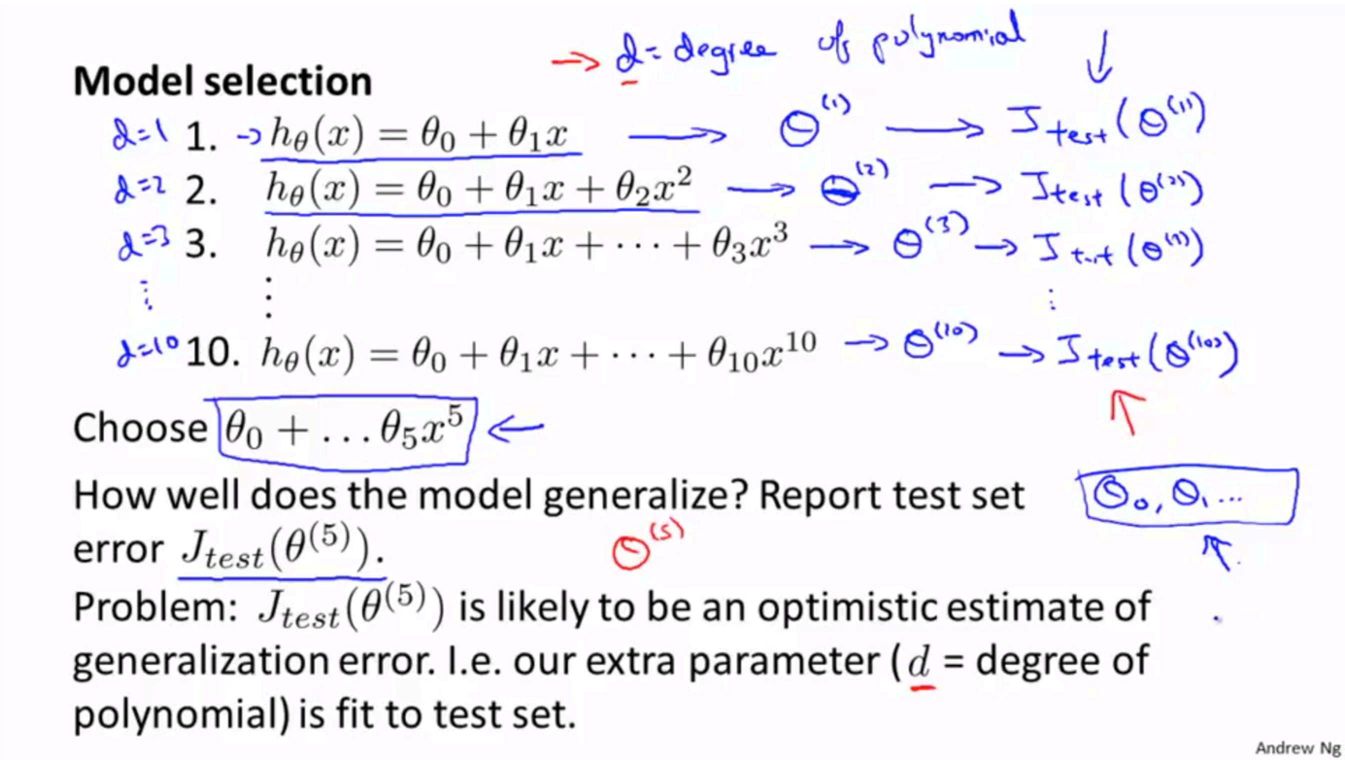
2. Learning Diagnostic:
    - definition: a test you can run to gain insight what is/isn't working;

3. Evaluting a hypothesis:

    - Split dataset into training set and test set(70%/30%, randomize ordering of data);
    - Compute test error: $J_{test}(\Theta) = \frac{1}{2m_{test}}...$ also the case for classificiation problems;
    - Miscalssification error(0/1 misclaffication error): $err(H_\Theta(x), y) = 1/0$ based on if prediction is correct or not; $Test_err = 1/m\sum_{i=1}^{m_{test}} err_i$

4. Model selection and train/validation/test sets:

    - choose different models(polynomial) to see how the test dataset performs:



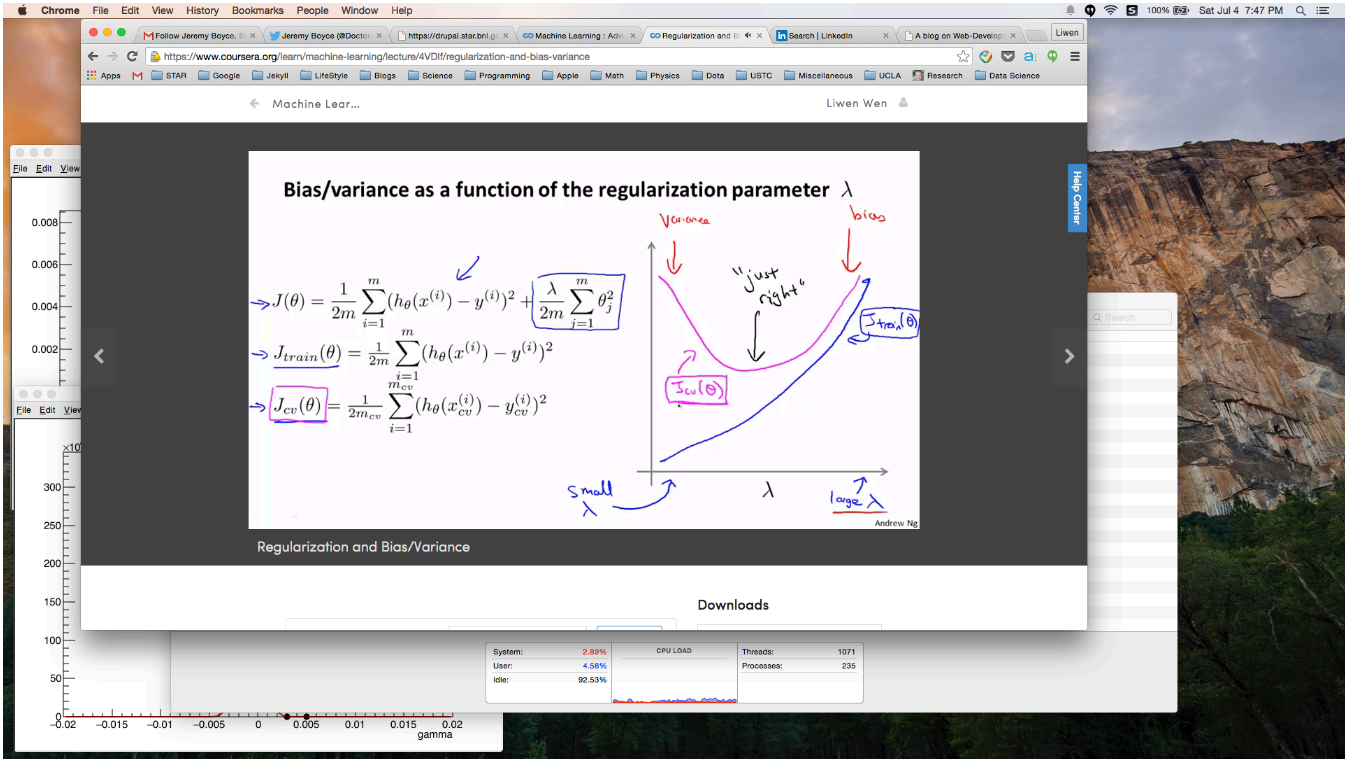
- Training/cross validation/test set(60/20/20, training error/cross validation error/ test error):
  - get hypothesis parameters;
  - apply these hypothesis on cv data set(other wise the degree of polynomial is fittd to test dataset);
  - apply the optimal hypothesis on test data set to estimate generalization error for the model selected;

5. Diagnosing bias vs. variance:

   - plot degree of poly. vs. error:
   - distinguish bias/variance:
     - bias: high-training error, high-cross validation error
     - variance: low-training error, high-cross validation error

6. Regularization and bias/variance:

   - large λ : underfitting;
   - intermediate λ: just right;
   - small λ: overfitting;
   - like in logistics regress fitting, try different λ from 0 to 10(for example) and follow previous procedures:



7. Learning curves:

    - learning curves: error vs. m
    - high bias: converge for both traning error and cv error(no help to increase the sample number)
    - high variance: big gap between training error and cv error(it helps to get more samples)

8. How to choose neural network archetect

    - small: underfitting and computationally cheap;
    - large struture: overfitting and can be fixed by regularization;