Debugging Learning Algorithms

COMP 135 Intro to Machine Learning
Liping Liu

Why debugging

- Low performance on test set
  - What's wrong?

How to diagnose an algorithm?

- There is one way of being right, but there are many different ways of being wrong

Diagnose a learning algorithm 1

- Q: is the problem with generalization to the test data?
- A: compare training error and test error (with a sense of variance)
  - High training error: do not expect to do better on test set
  - Low training error but high test error: the trained model does not generalize

Diagnose a learning algorithm 2

- Q: do you have train/test mismatch?
  - Do training and test sets have the same distribution?
- A: Shuffle the training and test data together and re-split
  - Error reduced: mismatch problem
  - Error not reduced: generalization problem

Diagnose a learning algorithm 3

- Q: Is your learning algorithm implemented correctly?
- A1: Is the loss or neg. llh decreasing on training set or test set?
- A2: comparing to existing packages
- A3: Sanity check with toy problems whose answer is known.
Bias-Variance Trade-off

• Identify two parts of the error
  - Bias: because there is not a good model
  - Variance: because the learning algorithm does not find a good model

\[
\text{error}(f) = \text{error}(f) - \min_{f^*} \text{error}(f^*) + \min_{f^*} \text{error}(f^*)
\]

Variance: estimation error
Bias: approximation error

Error here is test error, NOT training error

Bias-Variance Decomposition

• Almost impossible to separate the two errors in practice
• But we know the trend
  - Larger model space -> Smaller approximation error (bias), larger estimation error (var)
  - Smaller model space -> Larger approximation error (bias), smaller estimation error (var).

Relation to under/overfitting

• Underfitting ⇔ large approximation error (bias)
• Overfitting ⇔ large estimation error (variance)

Hyper-parameter and model space

• Almost every model has hyper-parameters controlling the size of model space
  - The size of model space is also called model complexity
• Two conceptual extremes
  - Lookup table: memorize training data and classify everything else as 0
  - Trivial classifier: classify everything as 0

Hyper-parameter and model space

• KNN classifier
  - K
  - Larger K => smaller model space

• Naïve Bayes classifier with binary features (project 1)
  - Smooth factor
  - Larger smooth factor => smaller model space

• Linear classifier
  - \( \lambda \)
  - Larger \( \lambda \) value => smaller model space

• Tree classifiers
  - Depth
  - Larger depth value => larger model space
**Hyper-parameter and model space**

- AdaBoost
  - Number of classifiers to combine
  - Larger number $\Rightarrow$ larger model complexity

**Technique of model selection**

- Use a validation set or cross-validation to choose a hyper-parameter
- Better the candidate set of hyper-parameters contains values leading to underfitting and also values leading to overfitting
  - Not to radically miss a good hyper-parameter value