GENERAL ASSEMBLY

INTRO TO DATA SCIENCE

CHOOSING A CLASSIFIER, STATISTICS

CHOOSING A CLASSIFIER

Sources:

- http://blog.echen.me/2011/04/27/choosing-a-machine-learning-classifier/
- http://homes.cs.washington.edu/~pedrod/papers/cacm12.pdf

Do you care about accuracy?

1. Test many models
2. Optimize their parameters
3. Use cross-validation
4. Choose the best
5. (Or combine them via an ensemble)

More data > Better algorithm

- Lots of data? The algorithm doesn’t matter as much, so choose based on speed or ease of use

Designing features is key

Do you just want “good enough”?

Small Training Set (generative)

- High bias/low variance classifier (e.g. Naïve Bayes). (Will not overfit as much)

Large Training Set (discriminative)

- Low bias/high variance (e.g. kNN), since high bias classifiers can’t always provide as accurate models given more data. (lower asymptotic error)

Do you just want "good enough"?

Small Training Set (generative)

- High bias/low variance classifier (e.g. Naïve Bayes). (Will not overfit as much)

Large Training Set (discriminative)

- Low bias/high variance (e.g. kNN), since high bias classifiers can't always provide as accurate models given more data. (lower asymptotic error)

MACHINE LEARNING GUIDELINES

LEARNING = REPRESENTATION + EVALUATION + OPTIMIZATION

Representation

- K-nearest neighbor, SVM, Naïve Bayes, Decision trees, etc.

Evaluation

- Accuracy rate, Precision/recall, Squared error, Posterior probability

Optimization

- Greedy search, Gradient descent, Quadratic programming

It’s Generalization That Counts

Always use a test set, or sometimes test and validation sets.

Classifier can be sabotaged via test data, e.g. if used to tune parameters. (Use cross-validation!)

Data alone is not enough.

Machine learning works because we can rely on real-world assumptions about data, beyond just the dataset.

Assumptions: Smoothness, similar examples = similar lasses, limited dependencies, limited complexity

Know probabilistic dependencies? Graphical models.
Know preconditions required per class? IF … THEN rules

Overfitting has many faces

Overfitting: expresses the training data too exactly and fails to generalize it. (Rather have accuracy 100% training/50% test, or 75% training/75% test?)

High Bias: consistently learns the wrong thing (e.g. linear learner can’t induce non-linear)

High Variance: learns random things irrespective of the real signal (e.g. decision trees – different training sets yield different trees)

Combatting Overfitting

- Cross-validation
- Regularization Term: Penalize models with more structure
- Perform statistical significance test (e.g. chi-square) before adding new structure

Intuition Fails in High Dimensions

Curse of Dimensionality

- 100 features space => 2^{100} possibilities, so exponentially more data is necessary
- Similarity-based reasoning breaks down due to extra space around each data point (harder to see clusters)
- Luckily, in real life data is not spread out uniformly through space, but is concentrated on a lower-dimensional manifold, so techniques e.g. PCA can be used

Theoretical Guarantees Not What They Seem

- Often work in limited cases, but not always in real-world data

Feature Engineering is Key

- Most effort may go here. Often the raw data cannot be learned from, but we can extract features that can be learned from.
- Data science an iterative process of running a learner, analyzing results, modifying data/learner, repeating.

More Data Beats a Cleverer Algorithm

Classifiers not learning?

- Design a better learning algorithm, or
- Get more data! (features/examples/etc)

Most models are essentially the same – e.g. neural nets can represent rule-based systems. There are two general types:

- Representation has a fixed size (e.g. linear)
- Representation can grow with data (e.g. Decision tree)

Learn Many Models, Not Just One

Model ensembles often better with little extra effort

Simplicity Does Not Imply Accuracy

Model ensembles often better with little extra effort (intentional dup!)

Representable Does Not Imply Learnable

Each model may not be able to learn all data representations.

e.g. Decision trees cannot learn representations requiring more leaves than training examples!

Correlation Does Not Imply Causation

Keep this in mind when learning models for predicting things!