DLI Accelerated Data Science Teaching Kit

Lecture 14.8 - Bagging

Numerous Possible Classifiers

Classifier	Training time	Cross valid ation	Testing time	Accuracy
kNN classifier	None	Can be slow	Slow	??
Decision trees	Slow	Very slow	Very fast	??
Naive Bayes classifier	Fast	None	Fast	??
…	…	…	…	…

Which Classifier/Model to Choose?

Possible strategies:

- Go from simplest model to more complex model until you obtain desired accuracy
- Discover a new model if the existing ones do not work for you
- Combine all (simple) models
- Common strategy: **bagging**
  - Improve stability and accuracy
  - Reduce variance
  - Reduce overfitting

Common Strategy: Bagging
(Bootstrap Aggregating)

Consider the data set $S = \{(x_i, y_i)\}_{i=1,..,n}$

- Pick a sample S^* with replacement of size n
  (S* called a "bootstrap sample")
- Train on S^* to get a classifier f^*
- Repeat above steps B times to get $f_1, f_2, \ldots, f_B$
- Final classifier $f(x) = \text{majority}\{f_b(x)\}_{j=1,...,B}$

http://statistics.about.com/od/Applications/a/What-Is-Bootstrapping.htm

Bagging decision trees

Consider the data set S

- Pick a sample S^* with replacement of size n
- Grow a decision tree T_b
- Repeat B times to get $T_1,...,T_B$
- The final classifier will be

$$f(x) = \text{majority}\{f_{T_b}(x)\}_{b=1,...,B}$$

Random Forests

Almost identical to bagging decision trees, except we introduce some randomness:

- Randomly pick *m* of the *d* available attributes, at every split when growing the tree (i.e., d-m attributes ignored)

Bagged **random** decision trees

= **Random forests**

DLI Accelerated Data Science Teaching Kit

Thank You