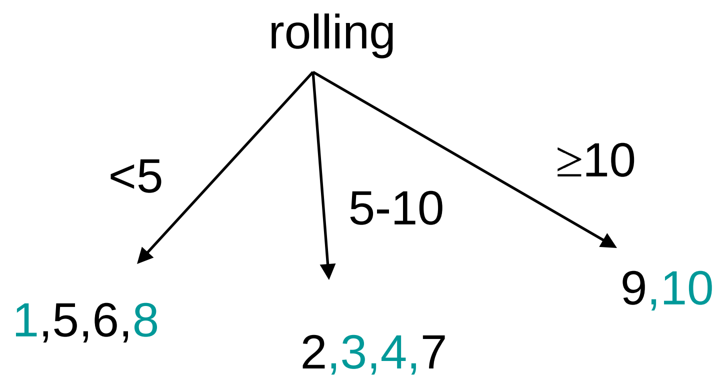
More Machine Learning

(Chapter 18)

Another Information Example

	stock	rolling	the	class
1	0	3	40	other
2	6	8	35	finance
3	7	7	25	other
4	5	7	14	other
5	8	2	20	finance
6	9	4	25	finance
7	5	6	20	finance
8	0	2	35	other
9	0	11	25	finance
10	0	15	28	other



$$Gain(rolling)=1-[4/10H(1/2,1/2)+4/10H(1/2,1/2)+2/10H(1/2,1/2)]=0$$

ML in Practice: General Approach

- Formulate task
- Obtain data
- What representation should be used? (attribute/value pairs)
- Annotate data
- Learn/refine model with data (training)
- Use model for classification or prediction on unseen data (testing)
- Measure accuracy

Issues

- Representation
    - How to map from a representation in the domain to a representation used for learning?
- Training data
    - How can training data be acquired?
- Amount of training data
    - How well does the algorithm do as we vary the amount of data?
- Which attributes influence learning most?
- Does the learning algorithm provide insight into the generalizations made?

Other Decision Tree cases

- What if class is discrete valued, not binary?

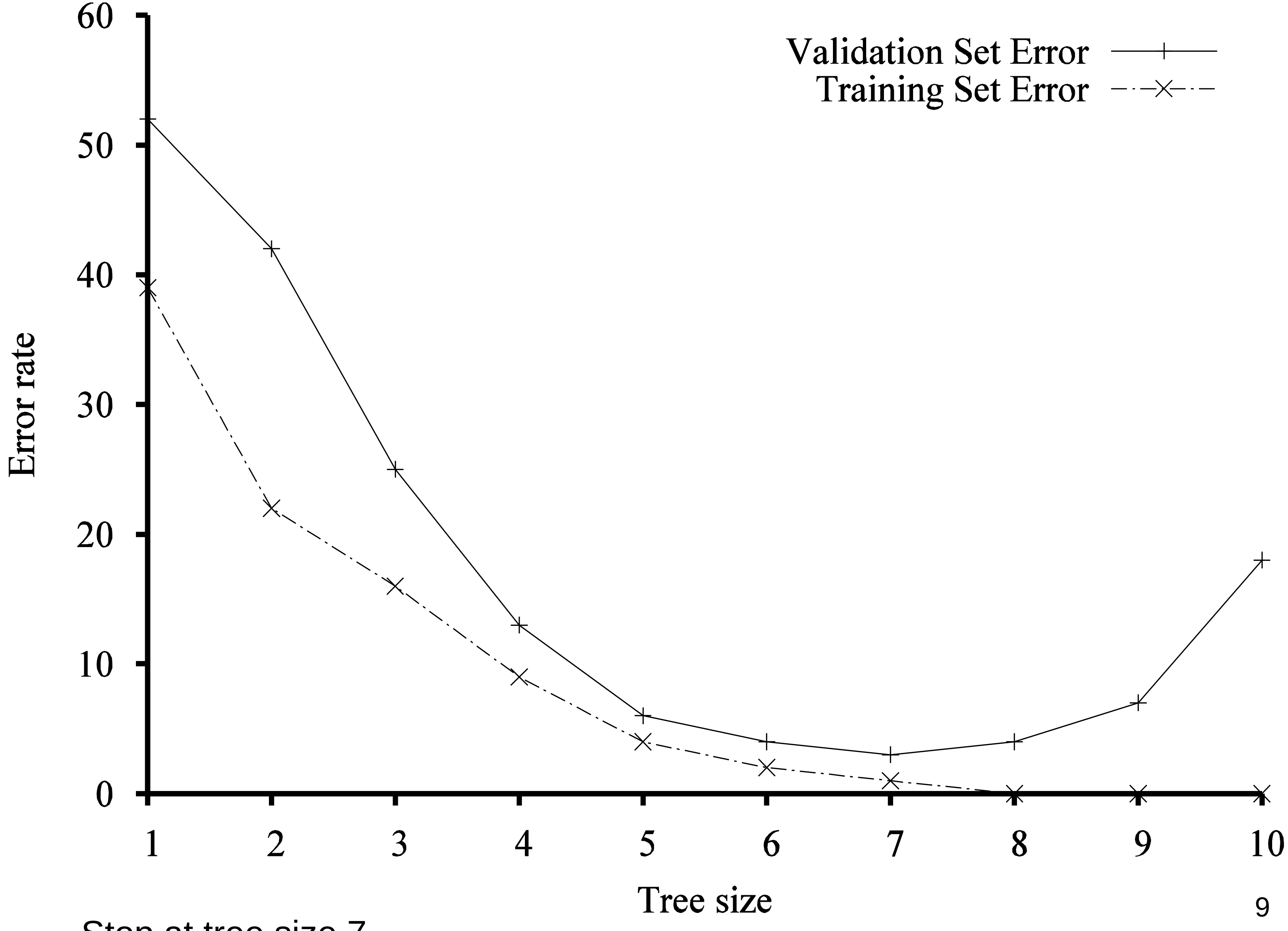
- What if an attribute has many values (e.g., 1 per instance)?

Training vs. Testing

- A learning algorithm is good if it uses its learned hypothesis to make accurate predictions on unseen data
  - Collect a large set of examples (with classifications)
  - Divide into two disjoint sets: the training set and the test set
  - Apply the learning algorithm to the training set, generating hypothesis h
  - Measure the percentage of examples in the test set that are correctly classified by h
  - Repeat for different sizes of training sets and different randomly selected training sets of each size.

Division into 3 sets

- Inadvertent peeking
  - Parameters that must be learned (e.g., how to split values)
  - Generate different hypotheses for different parameter values on training data
  - Choose values that perform best on testing data
  - Why do we need to do this for selecting best attributes?



Stop at tree size 7

Overfitting

- Learning algorithms may use irrelevant attributes to make decisions
  - For news, day published and newspaper
- Decision tree pruning
  - Prune away attributes with low information gain
  - Use statistical significance to test whether gain is meaningful

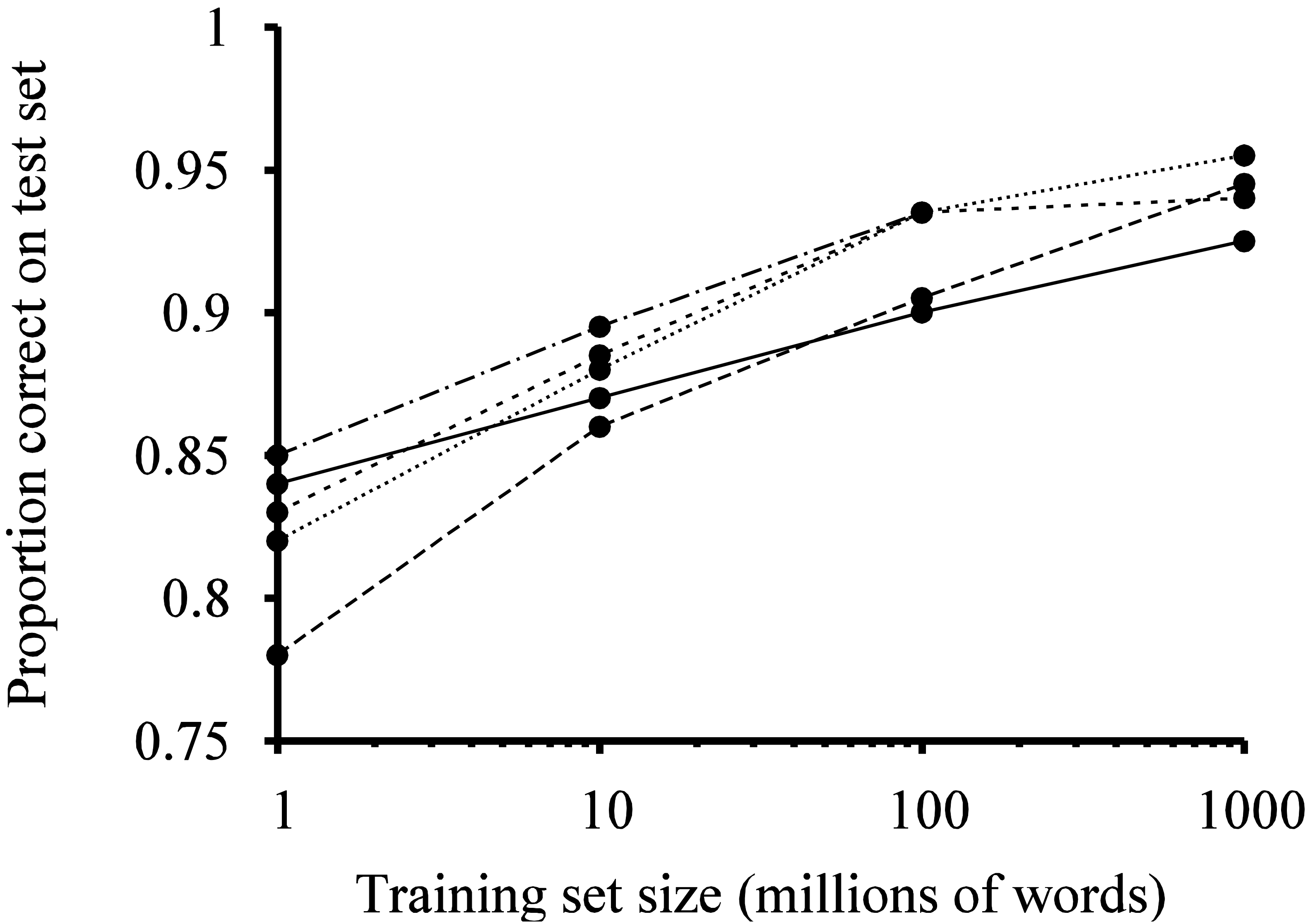
K-fold Cross Validation

- To reduce overfitting
- Run k experiments
  - Use a different 1/k of data for testing each time
  - Average the results
- 5-fold, 10-fold, leave-one-out

Not all errors are always equal

- Express utilities via a loss function
- Other metrics besides accuracy (recall, precision, f-measure)

ML in Practice



More room for improvement by increasing training set rather than improving algorithm!

Ensemble Learning

- Learn from a collection of hypotheses
- Majority voting
- Enlarges the hypothesis space

Boosting

- Uses a weighted training set
  - Each example as an associated weight $w_j \geq 0$
  - Higher weighted examples have higher importance
- Initially, w_j=1 for all examples
- Next round: increase weights of misclassified examples, decrease other weights
- From the new weighted set, generate hypothesis h_2
- Continue until M hypotheses generated
- Final ensemble hypothesis = weighted-majority combination of all M hypotheses
    - Weight each hypothesis according to how well it did on training data

AdaBoost

- If input learning algorithm is a weak learning algorithm
  - L always returns a hypothesis with weighted error on training slightly better than random
- Returns hypothesis that classifies training data perfectly for large enough M
- *Boosts* the accuracy of the original learning algorithm on training data

- To be continued in last set of slides…

Beyond this course

- Read if you are interested
  - Section 18.5 – learning theory
  - Section 18.6-18.9 - beyond decision trees