Generalization Quiz

1. Generalization is about how well our classifier does on testing data / future data.

2. Overfitting - predictor too flexible
   - can capture every detail of the data
   - Predictor A vs. B.
     $E_{train\,A} < E_{train\,B}$
     $E_{test\,A} > E_{test\,B}$

   Underfitting - predictor not flexible enough
   - cannot capture relevant patterns
   - $E_{train\,A} > E_{train\,B}$
     $E_{test\,A} > E_{test\,B}$

3. Complexity control
   Naive Bayes - no. of atts and limits on distribution parameters
   Decision Trees - no. of nodes and/or pruning confidence
   Lin. Regr - degree of polynomial / no. of attributes.

4. $E_{train} = \frac{1}{n} \sum_{i=1}^{n} error(f_D(\underline{x}_i), y_i)$
   error fn, predictor, trn. instance, label

5. We can't est. gen err from trn err!

6. Gen err - use the estimate to tell us how well we expect to perform on unseen data.
   Testing err - est. gen err.
   Training err - min. to build best predictor

7. If we use N randomly selected testing instances, and get an error rate of E, then the following are true:
   - The error rate on another rand-sel set will be dist. approximately Gaussian w/ mean E and variance $E(1-E)/N$

     $$\mu = \sqrt{\text{var}} \cdot \phi \rightarrow E \pm \sqrt{\frac{E(1-E)}{N}} \cdot \phi$$

   - E is our best estimate of the error rate in any set of ran-sel-set.

8. Properties of confidence intervals
   - CI varies roughly with the square root of the no. of samples
   - CI is fully specified by a confidence level, a mean and an interval either side mean (variance)
   - If N is reasonably large, then 95% CI for Erate in random sample of size N is about 2 SDs on other side of the mean.

9. We use CI to describe the range of error rates we'd expect to see when testing future unseen sets of instances.

10. Suppose we want to decide whether NB or DT are better for classification tasks, which attrs to use (NB), no. of nodes (DT). How would we do it?
    - Divide N instances to 3 sets
        - K training → train on all attrs/nodes
        - L validation → test on this, pick best performance.
        - M testing → error rate, confidence interval

11. Cross validation
    - every instance is used for testing
    - pick subsets in turn, train on other subsets and test on the one we picked, then average.
    - every instance is used for training
    - less likely to get biased testing set
    - each fold has distinct test & training set.

12. Leave one out
    - cross validation: all but one of the instances are used for training.

13. Stratification.
    - Deals w/ problems in k-fold
    - not useful for leave one out
    - ensures test/trn sets have representative balance of classes.