Machine Learning

Basic Methodology

Joakim Nivre

Uppsala University and Växjö University, Sweden
E-mail: nivre@msi.vxu.se

Evaluation

- Questions of evaluation:
  1. How do we evaluate different hypotheses for a given learning problem?
  2. How do we evaluate different learning algorithms for a given class of problems?
- The No Free Lunch Theorem:
  - Without prior assumptions about the nature of the learning problem, no learning algorithm is superior or inferior to any other (or even to random guessing).

Digression: Random Variables

- A random variable can be viewed as an experiment with a probabilistic outcome. Its value is the outcome of the experiment.
- The probability distribution of a variable Y gives the probability $P(Y = y_i)$ that Y will take on value y_i (for all values y_i).
- Variables may be categorical or numeric.
- Numeric variables may be discrete or continuous.
- We will focus on discrete variables.

Digression: Parameters of Numeric Variables

- The expected value (or mean) $E[Y]$ of a (discrete) numeric variable Y with possible values $y_1, \ldots, y_n$ is:

$$E[Y] \equiv \sum_{i=1}^{n} y_i \cdot P(Y = y_i)$$

- The variance $Var[Y]$ is:

$$Var[Y] \equiv E[(Y - E[Y])^2]$$

- The standard deviation σ_Y is:

$$\sigma_Y \equiv \sqrt{Var[Y]}$$

Digression: Estimation

- An estimator is any random variable used to estimate some parameter of the underlying population from which a sample is drawn.
  1. The estimation bias of an estimator Y for an arbitrary parameter p is $E[Y] - p$, If the estimation bias is 0, then Y is an unbiased estimator for p.
  2. The variance of an estimator Y for an arbitrary parameter p is simply the variance of Y.
- All other things being equal, we prefer the unbiased estimator with the smallest variance.

Digression: Interval Estimation

- A $N\%$ confidence interval for a parameter p is an interval that is expected with probability $N\%$ to contain p. Method:
  1. Identify the population parameter p to be estimated.
  2. Define the estimator Y (preferably a minimum-variance, unbiased estimator).
  3. Determine the probability distribution D_Y that governs the estimator Y, including its mean and variance.
  4. Determine the $N\%$ confidence interval by finding thresholds L and U such that $N\%$ of the mass of the probability distribution D_Y falls between L and U.

Loss Functions

- In order to measure accuracy, we need a loss function for penalizing errors in prediction.
- For regression we may use squared error loss:

$$L(f(x), h(x)) = (f(x) - h(x))^2$$

  where f(x) is the true value of the target function and h(x) is the prediction of a given hypothesis.
- For classification we may use 0-1 loss:

$$L(f(x), h(x)) = \begin{cases} 0 & \text{if f(x) = h(x)} \\ 1 & \text{otherwise} \end{cases}$$

Training and Test Error

- The training error is the mean error over the training sample $D = \langle\langle x_1, f(x_1)\rangle, \ldots, \langle x_n, f(x_n)\rangle\rangle$:

$$err_D(h) \equiv \frac{1}{n} \sum_{i=1}^{n} L(f(x_i), h(x_i))$$

- The test (or generalization) error is the expected prediction error over an independent test sample:

$$Err(h) = E[L(f(x), h(x))]$$

- Problem: Training error is not a good estimator for test error.

Overfitting and Bias-Variance

- Training error can be reduced by making the hypothesis more sensitive to training data, but this may lead to overfitting and poor generalization.
- The bias of a learning algorithm is a measure of its sensitivity to data (low bias implies high sensitivity).
- The variance of a learning algorithm is a measure of its precision (high variance implies low precision).
- Two extremes:
  1. Rote learning: No bias, high variance
  2. Random guessing: High bias, no variance

Model Selection and Assessment

- When we want to estimate test error, we may have two different goals in mind:
  1. Model selection: Estimate the performance of different hypotheses or algorithms in order to choose the (approximately) best one.
  2. Model assessment: Having chosen a final hypothesis or algorithm, estimate its generalization error on new data.

Training, Validation and Test Data

- In a data-rich situation, the best approach to both model selection and model assessment is to randomly divide the dataset into three parts:
  1. A training set used to fit the models.
  2. A validation set (or development test set) used to estimate test error for model selection.
  3. A test set (or evaluation test set) used for assessment of the generalization error of the finally chosen model.

Estimating Hypothesis Accuracy

- Let X be some input space and let $\mathcal{D}$ be the (unknown) probability distribution for the occurrence of instances in X.
- We assume that training examples $\langle x, f(x) \rangle$ of some target function f are sampled according to $\mathcal{D}$.
- Given a hypothesis h and a test set containing n examples sampled according to $\mathcal{D}$:
  1. What is the best estimate of the accuracy of h over future instances drawn from the same distribution?
  2. What is the probable error in this accuracy estimate?

Sample Error and True Error

- The sample test error is the mean error over the test sample $S = \langle\langle x_1, f(x_1)\rangle, \ldots, \langle x_n, f(x_n)\rangle\rangle$ (cf. training error):

$$err_S(h) \equiv \frac{1}{n}\sum_{i=1}^{n} L(f(x_i), h(x_i))$$

- The true test (or generalization) error is the expected prediction error over an arbitrary test sample:

$$Err(h) = E[L(f(x), h(x))]$$

- For classification and 0-1 loss this is the probability of misclassifying an instance drawn from the distribution $\mathcal{D}$.

Interval Estimation

- Statistical theory tells us:
  1. The best estimator for $Err(h)$ is $err_S(h)$.
  2. If $Err(h)$ is normally distributed, then with approximately N% probability $Err(h)$ lies in the interval

$$err_S(h) \pm z_N \frac{\sigma_{Err(h)}}{\sqrt{n}}$$

  where $[-z_N, z_N]$ is the interval that includes $N\%$ of a population with standard normal distribution.
- Problem: $\sigma_{Err(h)}$ is usually unknown.

Approximations

- For regression and squared error loss:
  1. Assume that $Err(h)$ is normally distributed.
  2. Estimate $\sigma_{Err(h)}$ with $\sigma_{err_S(h)}$.
  3. Use the t distribution when deriving the interval (for n $<$ 30).
- For classification and 0-1 loss:
  1. Assume that binomial $Err(h)$ can be approximated with a normal distribution.
  2. Estimate $\sigma_{Err(h)}$ with $\sqrt{err_S(h)(1 - err_S(h))}$.

Normal Approximation of Binomial $Err(h)$

- With normal approximation the interval estimation for classification and 0-1 loss becomes:

$$err_S(h) \pm z_N \sqrt{\frac{err_S(h)(1 - err_S(h))}{n}}$$

- This approximation is very good as long as:
  1. The test sample is a random sample drawn according to $\mathcal{D}$.
  2. The sample size $n \geq 30$.
  3. The sample error $err_S(h)$ is not too close to 0 or 1, so that $n \cdot err_S(h)(1 - err_S(h)) \geq 5$.

Comparing Hypotheses

- Consider two hypotheses h_1 and h_2 for some target function f, with test error $err_{S_1}(h_1)$ and $err_{S_2}(h_2)$, respectively.
  1. We want to test the hypothesis that $Err(h_1) \neq Err(h_2)$.
  2. We compute the probability of observing the test errors when the null hypothesis is true:

$$p = P(err_{S_1}(h_1), err_{S_2}(h_2) \mid Err(h_1) \neq Err(h_2))$$

  3. We conclude that $Err(h_1) \neq Err(h_2)$ (i.e. we reject the null hypothesis) if $p < \alpha$, where α is our chosen significance level.

Statistical Tests

- Independent samples ($S_1 \neq S_2$):
  1. t-test
  2. Mann-Whitney U test
  3. χ^2 tests
- Paired samples ($S_1 = S_2$):
  1. Paired t-test
  2. Wilcoxon ranks test
  3. McNemar's test

Comparing Learning Algorithms

- Comparing two learning algorithms L_1 and L_2, we want to compare their difference in test error in general:

$$E_{D \subset \mathcal{D}}[Err(L_1(D)) - Err(L_2(D))]$$

- In practice, we can measure test error for a single training set D and disjoint test set S:

$$err_S(L_1(D)) - err_S(L_2(D))$$

- However, there are two crucial differences:
  1. We use $err_S(h)$ to estimate $Err(h)$.
  2. We only measure the difference in error for one training set D.

Resampling Methods

- In data-poor situations, it may not be feasible to divide the available data into disjoint sets for training, validation and test.
- An alternative is to use resampling methods, which allow the same data instances to be used for both training and evaluation (although not at the same time).
- Two standard methods:
  - Cross-validation
  - Bootstrap

Cross-Validation

- Partition entire data set D into disjoint subsets $S_1, \ldots, S_k$.
- For i from 1 to k:
  1. $D_i \leftarrow D - S_i$
  2. $\delta_i \leftarrow err_{S_i}(L_1(D_i)) - err_{S_i}(L_2(D_i))$

  Return $\overline{\delta_i} = \frac{1}{k} \sum_i \delta_i$
- Every instance used exactly once for testing; number of test instances bounded by the size of D.
- Commonly used valued for k are 10 (10-fold cross-validation) and n (leave-one-out).

Bootstrap

- Let D be a data set and $rnd_k(D)$ a method for drawing a random sample of size $\frac{n}{k}$ from D.
- For i from 1 to n:
  1. $S_i \leftarrow rnd_k(D)$
  2. $D_i \leftarrow D - S_i$
  3. $\delta_i \leftarrow err_{S_i}(L_1(D_i)) - err_{S_i}(L_2(D_i))$

  Return $\overline{\delta_i} = \frac{1}{n} \sum_i \delta_i$
- Number of test instances not bounded by the size of D; instances may be used repeatedly for testing.

Caveat: Model Selection and Assessment

- Cross-validation and bootstrap are primarily methods for model selection (validation), not for model assessment (final testing).
- Regardless of validation method (held-out data, cross-validation, etc.), repeated testing on final test set should be avoided.

Caveat: Statistical Tests

- All statistical tests presuppose that the data set is a random sample. This can be problematic for linguistic data sets.
- Many statistical tests presuppose a particular type of distribution. Always consider alternative distribution-free tests.