Statistical Significance Testing

Machine Learning Lab, ASU

Surendra Singhi

April 29, 2005

Outline

Problem Statement & Assumptions

Problem

Comparing algorithms Given two learning algorithms A and B and a dataset we have to decide which algorithm is better?

Assumptions

- Assuming classification task, the ideas can be easily extended to regression problem.
- Everyone is familiar with probability theory 101 and engineering statistics 101.

Some Definitions

Definition (Null Hypothesis)

It is a hypothesis that the parameters, or mathematical characteristics, of two or more populations are identical.

Definition (Type I error)

This occurs when the null hypothesis is rejected when it is in fact true; that is, H_0 is wrongly rejected.

Definition (Type II error)

This occurs when the null hypothesis H_0, is not rejected when it is in fact false.

Some Definitions

Definition (Alternative Hypothesis)

The hypothesis which we will accept if the observed data values are sufficiently improbable under the null hypothesis.

Definition (Degrees of Freedom)

Degrees of freedom are the number of values in probability distributions that are free to be varied.

Sources of Variation

Four important points

To design and evaluate statistical tests, it is important to identify the sources of variation that each test must control.

- Random variation in test data used to evaluate algorithm. On any particular randomly-drawn test data set, one classifier may outperform another, even though, on the whole population the two classifiers will perform identically. This is pressing problem for small test data sets.
- Random variation due to selection of training data. On any particular randomly-drawn training data set, one classifier may outperform another, even though, on average, the two algorithm have same accuracy.

Four Sources of Variation continued..

- Variation due to internal randomness in the learning algorithm. For example consider back-propagation algorithm for training feed-forward neural networks. It is initialized with random weights, and the learned network depends critically on the random starting state.
- Variation due to random classification error. If a fixed fraction η of the test data points are randomly mislabeled, then no learning algorithm can achieve an error rate of less than η.

Properties of Good Test

- High power, low Type II error.
- Low Type I error.
- Good replicability.

McNemar's Test

- Data is divided into training set R and test set T.
- Algorithms A and B are trained on R and give classifiers $\hat{f}_A$ and $hatf_B$.
- Construct table as shown below:

number of examples misclassified by both $\hat{f}_A$ and $hatf_B$ (n_{00})	number of examples misclassified by $\hat{f}_A$ but not by $hatf_B$ (n_{01})
number of examples misclassified by $\hat{f}_B$ but not by $hatf_A$ (n_{10})	number of examples misclassified neither by $hatf_A$ nor $\hat{f}_B$ (n_{11})

- Under null hypothesis two algorithms should have same error rate, which means that $n_{01} = n_{10}$.
- McNemar's test is based on a χ^2 test for goodness-of-fit that compares the distribution of counts under null hypothesis to the observed counts.

Discussion Continued...

- The expected counts under null hypothesis are:

n_{00}	$(n_{01}+n_{10})/2$
$(n_{01}+n_{10})/2$	n_{11}

- The following statistic is distributed (approximately) as χ^2 with 1 degree of freedom; it incorporates a "continuity correction" term (of -1 in the numerator) to account for the fact that the statistic is discrete while the χ^2 distribution is continuous:
$$\frac{(|n_{01}-n_{10}|-1)^2}{n_{01}+n_{10}}$$
- If null hypothesis is correct, than the probability that this quantity is greater than $\chi^2_{1,0.95} = 3.841459$ is less than 0.5.

Discussion Continued...

- Doesn't measure variability due to choice of training set or internal randomness of learning algorithm.
- A single training set R is chosen, and the algorithms are only compared using that training set.
- This test should only be applied when we believe that sources of variability are small.
- It does not compare performance of algorithm on entire training set but only a fraction, which must be substantially small to ensure a large test set.

Resampled paired t test

- In each trial, the available sample S is randomly divided into a training set R of specified size and a test set T.
- Algorithms are trained on R and resulting classifiers tested on T. Let $p_A^{(i)}$ (respectively $p_B^{(i)}$) be the proportion of test samples misclassified by algorithm A (respectively B) during trial i.
- If we assume that the differences $p^{(i)} = p_A^{(i)} - p_B^{(i)}$ were drawn independently from a normal distribution, then we can apply Student's t test, by computing the statistic:
$$t = \frac{\bar{p}.\sqrt{n}}{\sqrt{\frac{\sum_{i=1}^{n}(p^{(i)}-\bar{p})^2}{n-1}}}$$

Discussion continued. . .

- Under null hypothesis, this statistic has t distribution with $n - 1$ degrees of freedom.
- Individual differences will not have normal distribution, because $p_A^{(i)}$ and $p_B^{(i)}$ are not independent.
- The $p^{(i)}$'s are not independent because the test sets in the trial overlap (and so does the training set).
- Unacceptably high Type I error. This is due to underestimation of variance because the samples are not independent.

Discussion continued. . .

- Let n_1 fraction of data used for training and n_2 fraction of data used for testing. Corrected resampling
  $t = \frac{\overline{p}.\sqrt{n}}{\sqrt{(\frac{1}{n}+\frac{n_2}{n_1})(\sum_{i=1}^{n}(p^{(i)}-\overline{p})^2)}}$
- Still suffers from the problem of low replicability.

k-fold Cross-Validated Paired t Test

- Divide S into k disjoint sets of equal size, $T_1, \ldots, T_k$. Then k trials are conducted, in each trial the test set is T_i, and the training set is the union of all the other T_j, $j \neq i$. The t statistic is calculated in the same way as the resampled paired test.
- Advantage is that each test set is independent of others. But the training sets still overlap. This overlap may prevent the test from obtaining a good estimate of the amount of variation that would be observed if each training set were completely independent of previous training sets.
- The variance in the t statistic maybe sometimes underestimated, the means are occasionally poorly estimated, and this may result in large t values.

Discussion continued. . .

- Slightly elevated Type I error.
- Good power, low Type II error.
- Low replicability.

5*2 CV Paired t Test

- 5 replications of 2-fold CV. In each replication, data is partitioned into two equal-sized sets S_1 and S_2. Each learning algorithm is trained on each set and tested on the other set.
- Produces four error estimate: $p_A^{(1)}$ and $p_B^{(1)}$ (trained on S_1 and tested on S_2) and $p_A^{(2)}$ and $p_B^{(2)}$ (trained on S_2 and tested on S_1).
- Subtracting error differences gives us estimated differences $p^{(1)} = p_A^{(1)} - p_B^{(1)}$ and $p^{(2)} = p_A^{(2)} - p_B^{(2)}$. From these differences the estimated variance is $s^2 = (p^1 - \overline{p})^2 + (p^2 - \overline{p})^2$ where $\overline{p} = (p^{(1)} + p^{(2)})/2$.

Discussion continued. . .

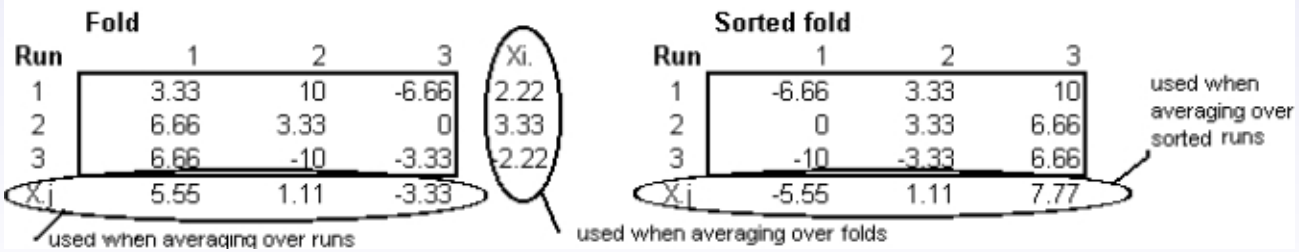
- Let s_i^2 be the variance computed from the i-th replication, and let $p_1^{(1)}$ be the $p^{(1)}$ from the very first of the five replications. Then define the following statistic:
  $$\tilde{t} = \frac{p_1^{(1)}}{\sqrt{\frac{1}{5}\sum_{i=1}^{5} s_i^2}}$$
- Low Type I error.
- Low power, or high Type II error.
- Low Replicability.

Different Approaches

Say we are doing $10 * 10$-fold cross validation.

- *Use all data* approach - considers all 100 outcomes as independent samples.
- *Mean Folds* test - averages the cells for a single 10-fold cross validation and considers these averages as samples.
- *Mean Runs* test - averages over the cells with the same fold number

Figure 1. Example illustrating the data used for averaging over folds, over runs and over sorted runs.



Different Approaches Continued

- *Mean Sorted Runs* test- sort the folds before averaging. Use averages over the runs of the sorted folds.
- *Mean Folds Average Var* - Same as mean folds but instead of estimating variance directly from these numbers, more accurate estimated is obtained by averaging variances obtained from the data of each of the 10-fold CV experiment.
- Similar extensions can be made to mean run and mean sorted run, giving rise to *mean run averaged var* and *mean sorted run averaged var* tests.
- *Mean Folds Average T* test - Instead of averaging just the variances, it averages over the test statistic that would be obtained from each individual 10-fold experiment.

Different Approaches Continued

- Similar extensions can be made to mean run and mean sorted run tests as well, giving rise to *mean run averaged T* and *mean sorted run averaged T* tests.

Table 1. Overview of hypothesis tests based on multiple run k-fold cv.

Test	mean m	variance $\hat{\sigma}^2$	df	Z
Use all data	$\frac{1}{k}\sum_{i=1}^{k}\frac{1}{r}\sum_{j=1}^{r}x_{ij}$	$\frac{\sum_{i=1}^{k}\sum_{j=1}^{r}(x_{ij}-m)^2}{k.r-1}$	$k.r-1$	$\frac{m}{\sqrt{\hat{\sigma}^2}/\sqrt{df+1}}$
folds	"	$\frac{\sum_{j=1}^{r}(x_{.j}-m)^2}{r-1}$	$r-1$	"
folds averaged var	"	$\frac{1}{r}\sum_{j=1}^{r}\hat{\sigma}^2_{.j}$	$r-1$	"
runs	"	$\frac{\sum_{i=1}^{k}(x_{i.}-m)^2}{k-1}$	$k-1$	"
runs averaged var	"	$\frac{1}{k}\sum_{i=1}^{k}\hat{\sigma}^2_{i.}$	$k-1$	"
sorted runs	"	$\frac{\sum_{j=1}^{r}(x_{\theta(ij)}-m)^2}{k-1}$	$k-1$	"
sorted runs averaged var	"	$\frac{1}{k}\sum_{i=1}^{k}\hat{\sigma}^2_{i.}(\theta)$	$k-1$	"
folds averaged T	$Z=\frac{1}{r}\sum_{j=1}^{r}\frac{x_{.j}}{\sqrt{\hat{\sigma}^2_{.j}/df+1}}$		df=$k-1$	
runs averaged T	$Z=\frac{1}{k}\sum_{i=1}^{k}\frac{x_{i.}}{\sqrt{\hat{\sigma}^2_{i.}/df+1}}$		df=$r-1$	
sorted runs averaged T	$Z=\frac{1}{k}\sum_{i=1}^{k}\frac{x_{i.}}{\sqrt{\hat{\sigma}^2_{i.}(\theta)/df+1}}$		df=$r-1$	

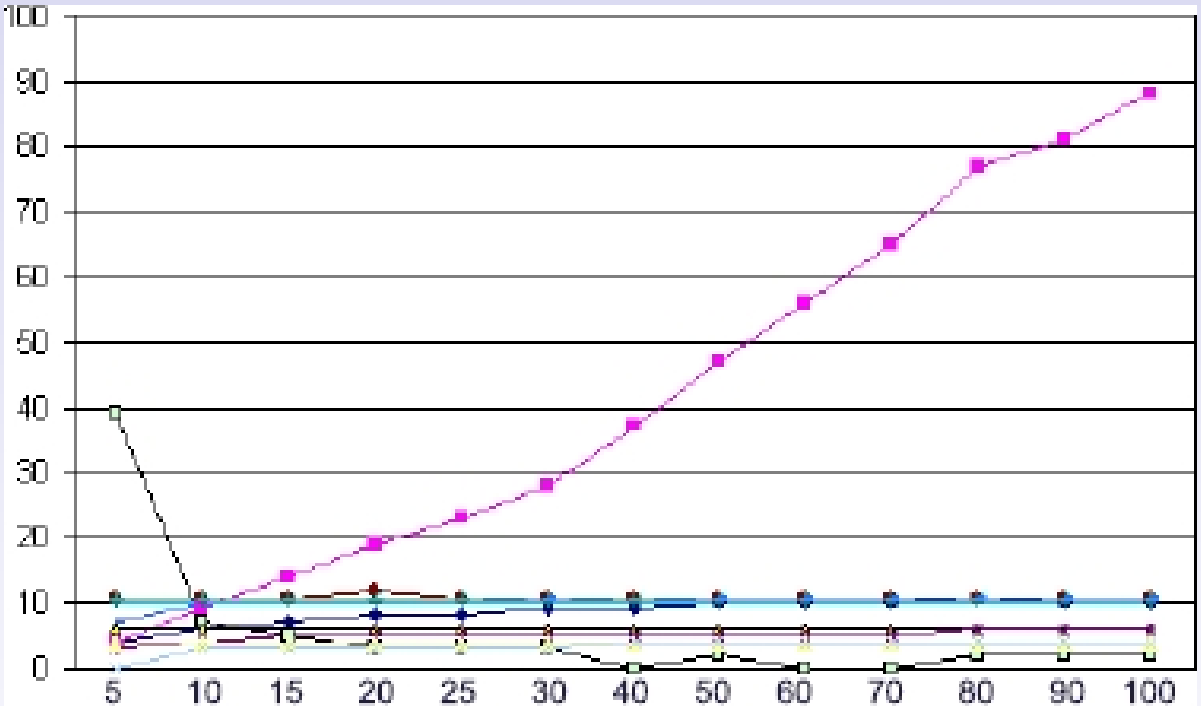
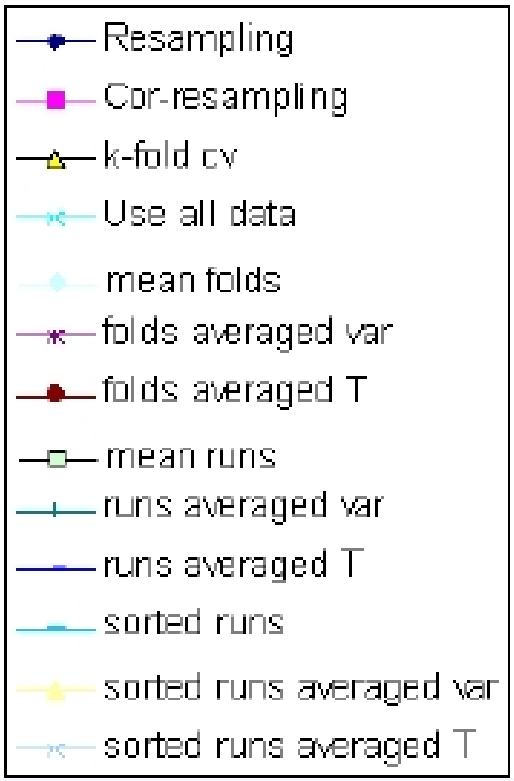


Figure 3. Legend for Figure 2 and 4.



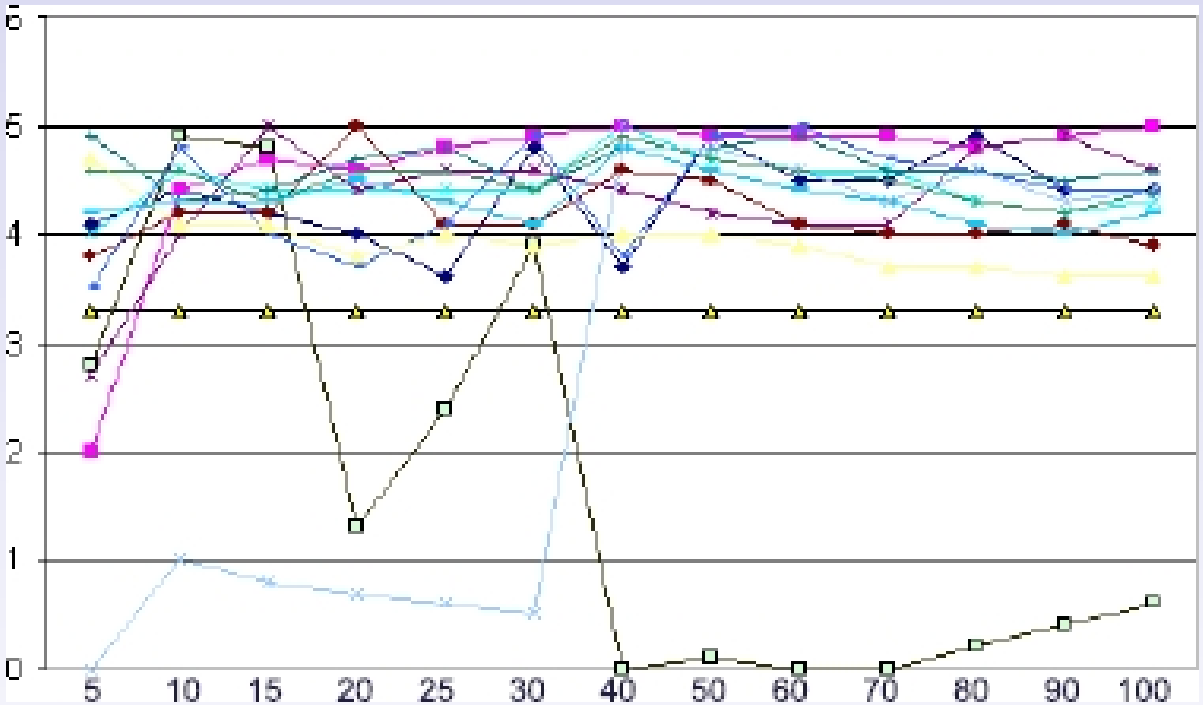


Figure 3. Legend for Figure 2 and 4.

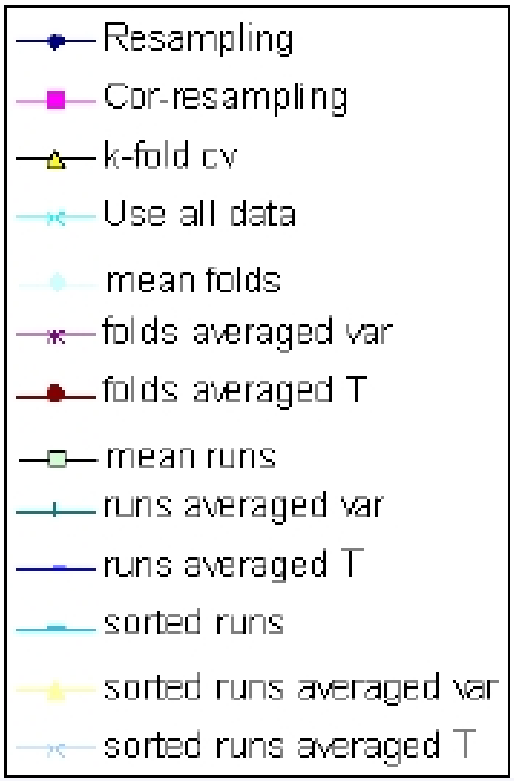


Table 2. Type 1 error for a range of class probabilities (in percentages)

test	0.05	0.1	0.2	0.3	0.4	0.5
Resampling	0.0	0.0	0.4	0.8	3.2	4.3
Cor-resampling	0.0	0.0	0.4	0.8	3.1	4.2
k-fold cv	0.0	0.0	0.0	0.4	2.6	2.5
Use all data	0.0	0.0	0.4	1.0	3.8	5.0
folds averaged var	0.0	0.0	0.2	0.7	3.5	4.8
folds averaged T	0.0	0.0	0.2	0.8	3.8	5.4
runs averaged var	0.0	0.0	0.4	0.8	3.9	5.1
runs averaged T	0.0	0.0	0.3	0.9	4.0	6.0
sorted runs	0.0	0.0	0.6	1.0	4.0	4.7

Table 4. Performance at various significance levels (in percentages).

Type 1 error (using Set 1)				
test	1%	2.5%	5%	10%
Resampling	1.1	2.1	4.4	10.1
Cor-resampling	1.5	2.2	4.4	9.2
k-fold cv	0.4	1.1	3.3	6.7
Use all data	0.6	2.0	4.6	9.6
folds averaged var	0.6	1.9	4.2	9.2
folds averaged T	0.6	1.8	4.3	10.2
runs averaged var	0.8	2.0	4.6	9.8
runs averaged T	1.0	2.1	4.8	10.5
sorted runs	0.5	2.0	4.3	9.7

Table 5. Replicability defined as fraction of tests having the same outcome 10 times on 10 different partitionings (in percentages). All tests calibrated except 5 x 2 cv.

	Set 1	Set 2	Set 3	Set 4	min.
Resampling	72.5	48.5	30.1	42.0	30.1
Cor-resampling	73.2	48.8	30.3	41.4	30.3
k-fold cv	81.8	71.3	42.6	51.9	42.6
5 x 2 cv	72.3	71.2	63.5	16.9	16.9
Use all data	92.8	80.9	76.6	98.5	76.6
folds averaged var	92.9	81.9	75.2	98.1	75.2
folds averaged T	91.6	80.3	74.2	98.3	74.2
runs averaged var	92.2	80.0	76.5	98.7	76.5
runs averaged T	91.2	78.2	73.2	98.6	73.2
sorted runs	92.5	81.7	75.0	98.3	75.0

Conclusion - Choosing between two algorithms!!!

Use 10 times repeated 10 fold cross validation test where all 100 individual accuracies are used to estimate mean and variance and with 10 degrees of freedom for binary data.

Future Work

Checking if the algorithm generalizes to non-binary data.
Multiple Comparisons in Induction Algorithm.

References


Y. Bengio and Y. Granvalet.
No unbiased estimator of the variance of k-fold cross-validation.
In *NIPS*.


R. R. Bouckaert.
Choosing between two learning algorithms based on calibrated tests.
In *Proceedings of the Twentieth International Conference on Machine Learning*, Washington DC, 2003.


T. G. Dietterich.
Approximate statistical tests for compairing supervised classification learning algorithm.
Neural Computation, 10(7):1895–1924, 1998.


C. Nadeau and Y. Bengio.
Inference for generalization error.
Machine Learning, 2003.

Thank you.