Choosing between two learning algorithms based on calibrated tests

Remco R. Bouckaert[1,2] RRB@XM.CO.NZ, REMCO@CS.WAIKATO.AC.NZ

1. Xtal Mountain Information Technology, Auckland 2. Computer Science Department, University of Waikato, Hamilton, New Zealand

Abstract

Designing a hypothesis test to determine the best of two machine learning algorithms with only a small data set available is not a simple task. Many popular tests suffer from low power (5x2 cv [2]), or high Type I error (Weka's 10x10 cross validation [11]). Furthermore, many tests show a low level of replicability, so that tests performed by different scientists with the same pair of algorithms, the same data sets and the same hypothesis test still may present different results. We show that 5x2 cv, resampling and 10 fold cv suffer from low replicability.

The main complication is due to the need to use the data multiple times. As a consequence, independence assumptions for most hypothesis tests are violated. In this paper, we pose the case that reuse of the same data causes the effective degrees of freedom to be much lower than theoretically expected. We show how to calibrate the effective degrees of freedom empirically for various tests. Some tests are not calibratable, indicating another flaw in the design. However the ones that are calibratable all show very similar behavior. Moreover, the Type I error of those tests is on the mark for a wide range of circumstances, while they show a power and replicability that is a considerably higher than currently popular hypothesis tests.

1. Introduction

Choosing between two learning algorithms given a single dataset is not a trivial task [10]. The most straightforward approach is to use a hypothesis test of some kind to decide whether we can reject the null hypothesis that the two algorithms perform the same. However, because there is often just a limited amount of data available, we have to reuse the data more than once to get a number of samples x. Here x provides an indication of the difference in accuracy of two algorithms. We can use the values of x to calculate some sort of statistic T. For that statistic to be used in a hypothesis test we normally assume that the samples x on which T is based are independent. However, we know that they are not be completely independent because they are based on partly the same data. In practice, the result is that when performing a test the Type I error differs considerably from the desired significance level, as observed in previous work [2, 7].

One of the effects of having dependence between samples is that the estimated variance is lower than the actual variance and a way to overcome this defect is to compensate the variance estimate [7]. In this article, we look at he effect of the degrees of freedom being lower than the theoretically expected number. We can compensate for this by calibrating the degrees of freedom.

In the following section, various well known hypothesis tests and their variations are introduced in detail, followed by tests based on repeated cross validation in Section 3. Section 4 sets out the procedure for calibrating the hypothesis tests by measering the actual degrees of freedom for the tests. In Section 5, we study empirically the behavior of the various tests by varying parameters of the environment in which the tests are applied. We finish with some concluding remarks, recommendations and some pointers to further research.

2. Hypothesis tests

First, we describe hypothesis tests based on k-fold cross validation, resampling and corrected resampling. The 5x2 cross validation test [2] was not considered because of the undesirable low replicability (see Table 5), making it hard to justify the extra work involved in setting up experiments. We make the degrees of freedom explicit in the formulas (usually indicate by df),

Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003), Washington DC, 2003.

so that it is clear what changes when the *df* is fixed to a value that differs from the default one.

2.1. k-fold cross validation

In k-fold cross validation, the data is split in k approximately equal parts and the algorithms are trained on all data but one fold for each fold. The accuracy is estimated by using the data of the fold left out as test set. This gives k accuracy estimates for algorithms A and B, denoted $P_{A,i}$ and $P_{B,i}$ where i $(1 \leq i \leq k)$ is the fold left out. Let x_i be the difference $x_i = P_{A,i} - P_{B,i}$, then the mean of x_i is normally distributed if the algorithms are the same and the folds are sufficiently large (at least containing 30 cases) [6]. The mean $x_.$ is simply estimated using $x_. = \frac{1}{k}\sum_{i=1}^{k} x_i$ and an unbiased estimate of the variance is $\hat{\sigma}^2 = \frac{\sum_{i=1}^{k}(x_i - x_.)^2}{k-1}$. We have a statistic approximating the t distribution with $df = k - 1$ degrees of freedom

$$t = \frac{x_.}{\sqrt{\hat{\sigma}^2}/\sqrt{df+1}}.$$

Dietterich [2] demonstrated empirically that the k-fold cross validation test has slightly elevated Type 1 error when using the default degrees of freedom. In Section 4, we will calibrate the test by using a range of values for *df* and selecting the one that gives the desired Type I error.

2.2. Resampling and corrected resampling

Resampling is repeatedly splitting the training data randomly in two, training on the first fraction and testing on the remaining section and applying a paired t-test. This used to be a popular way to determine algorithm performance before it was demonstrated to have unacceptable high Type I error [2].

Let $P_{A,j}$ and $P_{B,j}$ be the accuracy of algorithm A and algorithm B measured on run j $(1 \leq j \leq n)$ and x_j the difference $x_j = P_{A,j} - P_{B,j}$. The mean m of x_j is estimated by $m = \frac{1}{n}\sum_{j=1}^{n} x_j$, and the variance is first estimated using $\hat{\sigma}^2 = \sum_{j=1}^{n}(x_j - m)^2$. To get an unbiased estimate, this is multiplied with $\frac{1}{n-1}$.

Nadeau and Bengio [7] observe that the high Type I is due to an underestimation of the variance because the samples are not independent. They propose to make a correction to the estimate of the variance, and multiply $\hat{\sigma}^2$ with $\frac{1}{n} + \frac{n_2}{n_1}$ where n_1 is the fraction of the data used for training and n_2 the fraction used for testing. Altogether, this gives a statistic approximating the t distribution with $df = n - 1$ degrees of freedom:

$$t = \frac{\sum_{j=1}^{n} x_j}{\sqrt{(\frac{1}{n-1})\hat{\sigma}^2}/\sqrt{df+1}}$$

for resampling, and

$$t = \frac{\sum_{j=1}^{n} x_j}{\sqrt{(\frac{1}{n} + \frac{n_2}{n_1})\hat{\sigma}^2}/\sqrt{df+1}}$$

for corrected resampling.

3. Multiple run k-fold cross validation

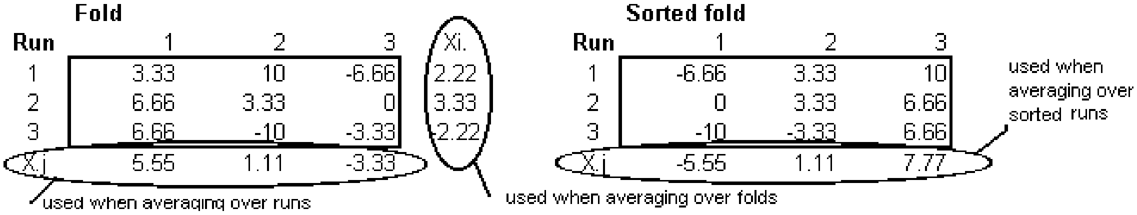
As our experiments will demonstrate, tests from the previous section suffer from low replicability. This means that the result of one researcher may differ from another doing the same experiment with the same data and same hypothesis test but with different random splits of the data. Higher replicability can be expected for hypothesis tests that utilize multiple runs of k-fold cross validation. There are various ways to get the most out of the data, and we will describe them with the help of Figure 1.[1] The figure illustrates the results of a 3 run 3-fold cross validation experiment inside the box at the left half, though in practice a ten run 10-fold cross validation is more appropriate. Each cell contains the difference in accuracy between two algorithms trained on data from all but one folds and measured with the data in the single fold that was left out as test data.

The *use all data* approach is a naive way to use this data. It considers the 100 outcomes as independent samples. The *mean folds* test first averages the cells for a single 10-fold cross validation experiment and considers these averages as samples. In Figure 1 this test would use the numbers in the most right column. These averages are known to be better estimates of the accuracy [5]. The *mean runs* test averages over the cells with the same fold number, that is, the numbers in the last row of the top matrix in Figure 1. A better way seem to be to sort the folds before averaging. Sorting the numbers in each of the folds gives the matrix at the bottom of Figure 1. The *mean sorted runs* test then uses the averages over the runs of the sorted folds.

The *mean folds average var* test uses an alternative way to estimate the variance from the mean folds test. Instead of estimating the variance directly from these numbers, a more accurate estimate of the variance may

[1] A more formal description and elaborated motivation of these tests can be found in [1].

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



Run	Fold 1	2	3	Xi.
1	3.33	10	-6.66	2.22
2	6.66	3.33	0	3.33
3	6.66	-10	-3.33	2.22
X.j	5.55	1.11	-3.33	

Run	Sorted fold 1	2	3
1	-6.66	3.33	10
2	0	3.33	6.66
3	-10	-3.33	6.66
X.j	-5.55	1.11	7.77

used when averaging over runs
used when averaging over folds
used when averaging over sorted runs

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=$r-1$			

be obtained by averaging variances obtained from the data of each of the 10-fold cv experiments. The same idea can be extended to the mean run and mean sorted run test, giving rise to the *mean run averaged var* and *mean sorted run averaged var* tests.

The *mean folds average T* test uses a similar idea as the mean folds averaged var test, but instead of averaging just the variances, it averages over the test statistic that would be obtained from each individual 10 fold experiment. This idea can be extended to the mean run and mean sorted run tests as well, giving the *mean run averaged T* and *mean sorted run averaged T* tests.

More formally, let there be r runs ($r > 1$) and k folds ($k > 1$) and two learning schemes A and B that have accuracy a_{ij} and b_{ij} for fold i ($1 \leq i \leq k$) and run j ($1 \leq j \leq r$). Let $\mathbf{x}$ be the difference between those accuracies, $x_{ij} = a_{ij} - b_{ij}$. We use short notation $x_{.j}$ for $\frac{1}{k}\sum_{i=1}^{k} x_{ij}$ and $x_{i.}$ for $\frac{1}{r}\sum_{j=1}^{r} x_{ij}$. Further, we use the short notation $\hat{\sigma}^2_{.j}$ for $\frac{\sum_{i=1}^{k}(x_{ij}-x_{.j})^2}{k-1}$ and $\hat{\sigma}^2_{i.}$ for $\frac{\sum_{j=1}^{r}(x_{ij}-x_{i.})^2}{r-1}$. For the test that use sorting, let $\theta(i,j)$ be an ordering such that $x_{\theta(ij)} \leq x_{\theta((i+1)j)}$. Then we use $\hat{\sigma}^2_{i.}(\theta)$ for $\frac{\sum_{j=1}^{r}(x_{\theta(ij)}-x_{i.})^2}{r-1}$. Table 1 gives an overview of the various tests and the way they are calculated. For many of the tests, the mean m and variance $\hat{\sigma}^2$ of $\mathbf{x}$ are estimated with which the test statistic Z is calculated. As evident from Table 1, the averaged T tests have a slightly different approach.

4. Calibrating the tests

The tests are calibrated on the Type I error by generating 1000 random data sets with mututal independent binary variables. The class probability is set to 50%, a value that typically generates the highest Type I error [2]. On each of the training sets, two learning algorithms are used, naive Bayes [4] and C4.5 [8] as implemented in Weka [11]. The nature of the data sets ensures that none can be outperformed by the other. So, whenever a test indicates a difference between the two algorithms, this contributes to the Type I error. Each test is run with degrees of freedom ranging from 2 to 100. The degrees of freedom at which the Type I error measured is closest and less than or equal to the significance level is chosen as the calibrated value.

Figure 2. Degrees of freedom (left) and Type I error in percentage (right) that coincide with significance level of $\alpha = 0.05$ and 10 folds for various numbers of runs.

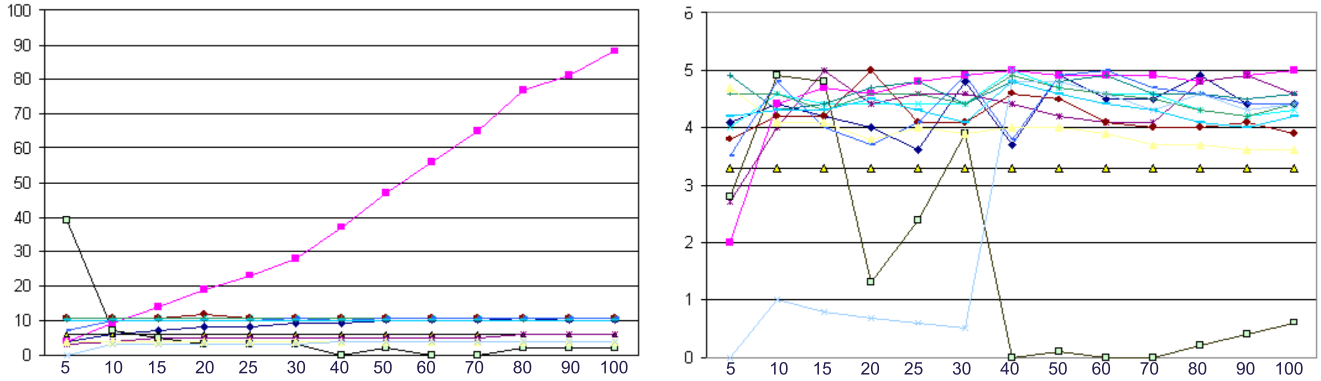
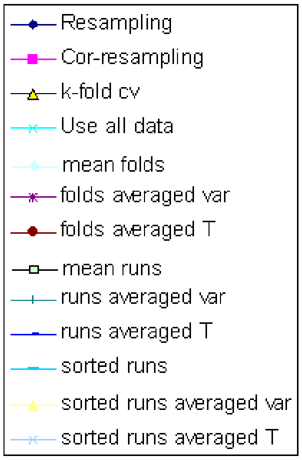


Figure 3. Legend for Figure 2 and 4.



Note that in our experiments we use stratification for the different runs. Stratification ensures that class values are evenly distributed over the various folds, and tends to result in more accurate accuracy estimates [5].

The left half of Figure 2 shows the degrees of freedom thus obtained for 5, 10, 15, 20, 25, 30, 40, 50, 60, 70, 80, 90 and 100 runs. A 5% significance level[2] and 10 folds where chosen to calibrate because these values are fairly often found in the literature. The right half of Figure 2 shows the corresponding Type I error and Figure 3 shows the legend.

The following can be observed from Figure 2:

- The calibrated degrees of freedom for resampling and corrected resampling consistently increases with increasing number of runs r, and they are the only tests doing so.
- The df for resampling is up to 90% lower than the expected $r - 1$ while the df for corrected resampling is circa 10% lower than the expected $r - 1$. This indicates that the variance correction applied to corrected resampling is almost right, but calibration helps a bit more.
- The df for 10 fold cross validation is constant over runs, because it does not vary for different runs. The calibrated df is consistently 33% lower than expected for the various significant levels and the Type I error is close to the mark.
- Comparing results for 1%, 5% and 10% significance levels, the df for all other tests are constant for different significance level (mostly within 1 difference) which can be attributed to sample variations.
- The df for all other tests varies most for 5 and 10 runs but is constant over higher numbers of runs (within 1 difference). This is explained by having a look at the Type I error: with lower numbers of runs there is little choice in values for the calibrated df. Due to variation in the data, no df may be available for which a Type I error close to the desirer value may be available.
- The df for mean folds is very low (at most 7) and is far below the expected $r - 1$. This indicates that the degrees of freedom is not the main issue with this test, but variance underestimation probably is.

[2] Refer to the technical report [1] for full numeric details and results at 1% and 10% significance level as well.

- The mean runs is very low, (except for a glitch at $r = 5$ runs). Further, there are a few dfs found as zeros, indicating that for none of the values of df the Type I error does not exceed the expected value. Again, this all indicates that the value of df is not the issue with this test.

- The df for sorted runs averaged var and average T is very low.

- The df's for "use all data", "folds varT", "folds meanT", "runs varT", "runs meanT", and "sorted runs" are all constantly circa 11. For those tests, the Type I error is mostly on the mark indicating that calibration of df may have fixed invalid independence assumptions.

A number of tests will not be considered further since the low or erratic values for the calibrated df indicate that the df is not the main issue with those tests (examining the results in the following experiments confirm this, however for clarity of presentation they are omitted here). These tests are mean folds, mean runs, sorted runs averaged var, and sorted runs averaged T.

5. Empirical performance

In this section, we study the behavior of the various tests empirically by measuring Type I error for varying class probability, number of runs and significance level. Also Type II error and replicability are measured and some other learning algorithms are considered.

5.1. Vary class probabilities

Table 2 shows the Type I error measured using calibrated tests on 1000 syntetic datases with 10 independent binary variables and a class probability in the range 0.05, 0.1, 0.2, 0.3, 0.4 and 0.5. So, there are a total of 1000 x 6 = 6000 data sets with 300 instances each. Naive Bayes and C4.5 was trained on each of those data sets. The number in the table indicates the number of data sets on which the test indicates that the performance of the algorithms differs. Ten run tenfold cross validation data was used at 0.05 significance level.

Except for the averaged T tests, none of the tests have a Type I error exceeding the expected level. In fact, the Type I error tends to be considerably lower with the lower class probabilities.

5.2. Vary number of runs

In this and the following sections, four randomly generated Bayesian networks were used to generate 1000 data sets with 300 instances each (see [1] for details) resulting in different performances of naive Bayes and C4.5. Table 3 shows the properties of these data sets when learned on the data sets and measured on a single 10.000 instances test set. Set 1 is the set used for calibrating the tests.

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 3. Average accuracies on test data by training on the 1000 data sets (in percentages).

Algorithm	Set 1	Set 2	Set 3	Set 4
Naive Bayes:	50.0	87.84	71.92	81.96
C4.5:	50.0	90.61	77.74	93.23
Difference	0.0	2.77	5.83	11.27

Figure 4 shows the Type I error for various numbers of runs of 10 folds cross validation at 0.05 significance level. Since the tests were calibrated on this data set, it comes as no surprise that the Type I error is close and below the targeted 50.

The power of the tests was measured on data sets where C4.5 outperforms naive Bayes with an increasing margin (see Table 3). Figure 4 also shows the power of the tests for data sets 2, 3, and 4.

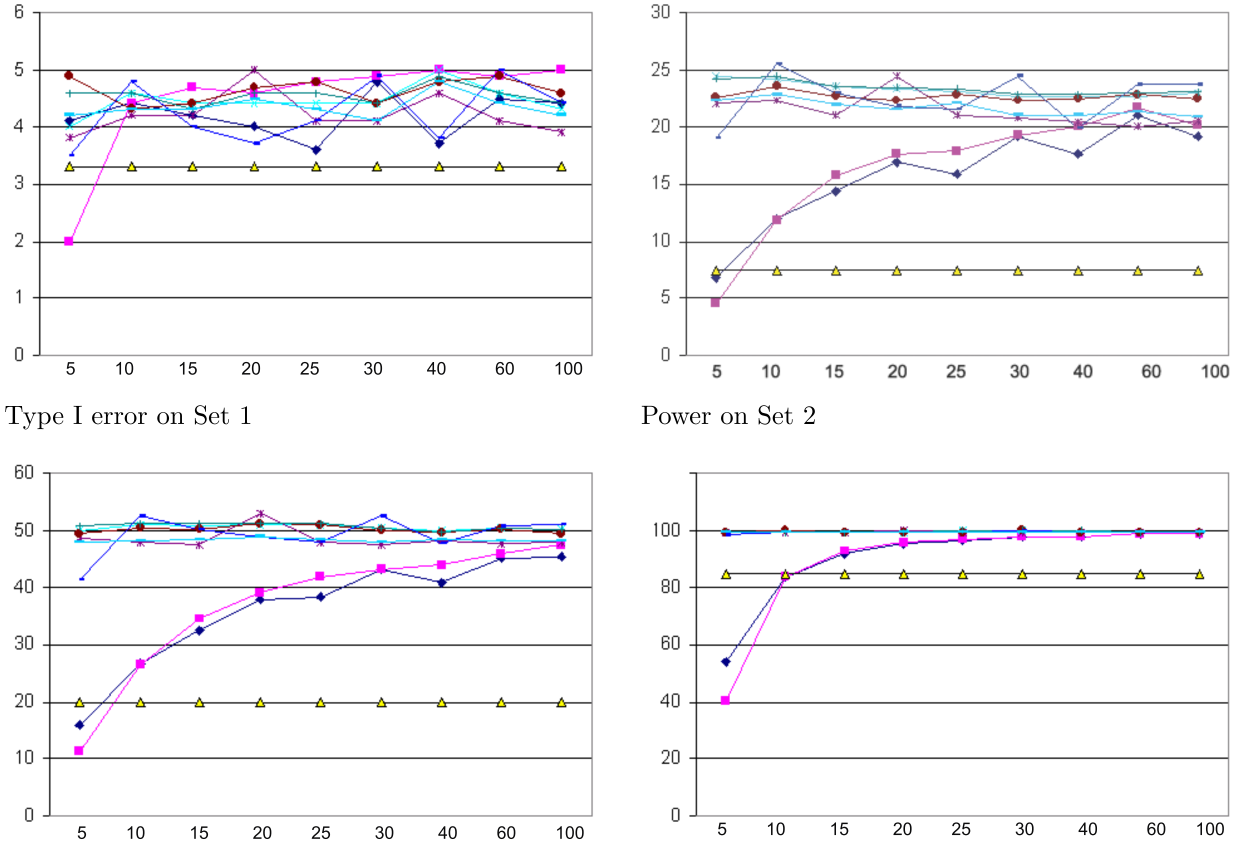
Some observations:

- Note that the k-fold cv test does not vary with increasing numbers of runs since conceptually it only uses the first run and ignores the others.

- The power of resampling and corrected resampling tests increases with increasing the number of runs. With increased runs the amount of datapoints used increases, which results in better power.

- Interestingly, all other calibrated tests that use all data have a power that is rather close to the other tests. This is a good indication that the calibration does its job: compensating for dependency of items used in the test.

Figure 4. Result for multiple runs in percentages (see Figure 3 for legend).



Type I error on Set 1

Power on Set 2

Power on Set 3

Power on Set 4

- Those tests show very little sensitivity to the number of runs, though the power decreases slightly with increasing number of runs. This may be due to the small variability of the calibrated df.

5.3. Vary significance level

Datasets 1 to 4 were used to measure the sensitivity of the tests on the significance level. The tests are for 10 run and 10 folds. Table 4 shows the Type I error for significance levels 1%, 2.5%, 5%, and 10%. Not surprisingly, the Type I error is close and below the expected levels, since the tests were calibrated on this set.

Some observations:

- Increasing the significance level increases the power, but this comes of course at the cost of increased Type I error.

- There is a large gap between the resampled, corrected resampled and k-fold cv test on the one hand and the other tests on the other hand. The tests that use all data of the 10x10 folds have considerably better power

- The runs averaged T test is almost always best and always in the top 3 of the tests listed. However, it also has a slightly elevated Type I error at 10% significance level (10.5%).

- The use all data test is always in the top 3 of the tests listed.

- Overall, all tests (except the first three) have a power[3] that is mutually close, an indication that the calibration compensates appropriately for dependence between samples.

5.4. Measuring replicability

This section shows results for measuring replicability. Each test was run ten times for each data set with dif-

[3] Refer [1] for measurements of the power on Set 2, 3 and 4.

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

ferent random splits. Table 5 shows the overall replicability, defined as the number of times the ten tests all reject or all not reject the null hypothesis. Also, the minimum for the four data sets is listed, which is meaningful because the location where replicability is lowest is typically the area in which decisions need to be made which are not completely obvious.

Table 6 shows the minimum columns as calculated for Table 5 for significance levels 0.01, 0.025, 0.05 and 0.10.

Some observations:

- Again, there is a big difference between the first three tests and the rest. The tests using all data from the 10x10 folds show better replicability overall.
- Judging from the last column of Table 5, use all data has highest replicability overall.
- There is little variance between the replicability of the tests that use all data from the 10x10 folds.

  Again, this indicates that calibrating compensates appropriately for the dependence between samples.
- As Table 6 shows, the significance level does not have a major impact on the replicability of tests that use all data of the 10x10 folds, while the first three tests are considerably affected.

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

Table 6. Replicability defined as fraction of tests (in percentages) having the same outcome 10 times on 10 different partitionings for different significance levels.

test	0.01	0.025	0.05	0.10
Resampling	9.8	20.7	30.1	21.0
Cor-resampling	10.2	22.4	30.3	20.6
k-fold cv	12.8	23.6	42.6	33.6
Use all data	78.3	75.5	76.6	78.0
folds averaged var	77.8	76.0	75.2	78.1
folds averaged T	75.6	73.1	74.2	76.4
runs averaged var	76.8	75.3	76.5	77.0
runs averaged T	74.1	73.3	73.2	74.8
sorted runs	77.6	76.1	75.0	77.4

The 5x2 cv test has a particular low replicability, which appears clearly for set 4. The conservative nature of the test makes set 4 the set on which the outcome is not always clear (unlike the more powerful tests where set 3 has this role). The reason for the low replicability is that the outcome of the test is dominated by an accuracy estimate based on one outcome of a single run of a 2cv experiment, which can vary greatly (as Table 5 shows). Calibration cannot repair this problem.

5.5. Other learning algorithms

Naive Bayes and C4.5 were chosen for calibration because these algorithms are based on completely different principles so that dependence of learning algorithm influences a test minimally. Further, these algorithms are sufficiently fast to perform a large number of experiments. To see how well tests calibrated on those two algorithm perform on other algorithms, we compared nearest neighbor, tree augmented naive Bayes and voted perceptron with default settings as imple-

Table 7. Type 1 error (in percentage) on set 1 for various algorithms with the 10x10 cv use all data test.

Binary	C4.5	nb	nn	tan
naive Bayes (nb)	5.0			
nearest neighbor (nn)	2.0	3.8		
tree augmnt. nb (tan)	4.3	1.7	2.7	
voted perceptron	1.4	0.4	0.1	0.6

mented in Weka [11]. Table 7 shows the Type 1 error at 5% significance level for the 10 run 10 fold use all data test. This table is fairly representative for the other 10x10 fold cv based tests: these tests have a Type 1 error that differs less than 1% in absolute value from that in Table 7. Overall, the Type 1 errors are acceptable, suggesting that the calibrated test can be used on a wider range of algorithms.

6. Conclusions

We studied calibrating the degrees of freedom of hypothesis tests as a way to compensate for the difference between the desired Type I error and the true Type I error. Empirical results show that some tests are not calibratable. However, the ones that are calibratable show surprisingly similar behavior when varying parameters such as the number of runs and significance level, which is an indication that an incorrect numer of degrees of freedom indeed is the cause of a difference in observed and theoretical Type I error.

Furthermore, the calibrated tests show a pleasently high replicability in particular compared to 5x2 cross validation [2], (corrected) resampling [7] and k-fold cross validation.

For choosing between two algorithms, we recommend using the 10 time repeated 10 fold cross validation test where all 100 individual accuracies are used to estimate the mean and variance and with 10 degrees of freedom for binary data. This is conceptually the simplest test and has the same properties as the other calibrated repeated k-fold cross validation tests. Furthermore, it empirically outperforms 5x2 cross validation, (corrected) resampling and k-fold cross validation on power and replicability. Further emperical research is required to confirm whether this test performs well on non-binary data.

There are many ways the techniques presented in this article can be generalized. For example, when there is not sufficient data in the data set to justify the normality assumption used for a t-test, a sign test may be applicable. The benefit of sign tests is that no assumptions need to be made about the distribution of the variable sampled, which is the difference between accuracies in our case. Calibrated tests could be used for this purpose.

The main limitation of the pairwise approach is that a choise between only two algorithms can be made. However, in practice multiple algorithms will be available to choose from. This gives rise to the so called multiple comparison problems [3] in choosing learning algorithms. Suppose that all algorithms perform the same on the domain, then the probability that one of those algorithms seem to outperform the others significantly increases, just like flipping a coin multiple times increases the change of throwing head 5 times in a row. Similar issues arise when comparing two algorithms on multiple data sets, or multiple algorithms on multiple data sets.

Acknowledgements

I want to thank Eibe Frank for his very stimulating discussions on this topic and the other members of the machine learning group at the University of Waikato for their encouragements.

References

[1] R.R. Bouckaert. Choosing between two learning algorithms based on calibrated tests. Working paper, Computer Science Department, University of Waikato, 2003.

[2] T.G. Dietterich. Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. Neural Computation, 10(7) 1895–1924, 1998.

[3] D. Jensen and P.R. Cohen. Multiple comparisons in induction algorithms. Machine Learning 38(3), 309-338, 2000.

[4] G.H. John and P. Langley. Estimating Continuous Distributions in Bayesian Classifiers. Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence. pp. 338–345, Morgan Kaufmann, San Mateo, 1995.

[5] R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence, pages 1137–1143. San Mateo, CA: Morgan Kaufmann, 1995.

[6] T. Mitchell. Machine Learning. McGraw Hil, 1997.

[7] C. Nadeau and Y. Bengio. Inference for the generalization error. Advances in Neural Information Processing Systems 12, MIT Press, 2000.

[8] R. Quinlan. C4.5: Programs for Machine Learning, Morgan Kaufmann Publishers, San Mateo, CA, 1993.

[9] C.J. Wild and G.A.F. Weber. Introduction to probability and statistics. Department of Statistics, University of Auckland, New Zealand, 1995.

[10] S. Salzberg. On Comparing Classifiers: Pitfalls to Avoid and a Recommended Approach. Data Mining and Knowledge Discovery 1:3 (1997), 317-327.

[11] I.H. Witten and E. Frank. Data mining: Practical machine learning tools and techniques with Java implementations. Morgan Kaufmann, San Francisco, 2000.