# Estimating Replicability of Classifier Learning Experiments

**Remco R. Bouckaert**[1,2] REMCO@CS.WAIKATO.AC.NZ, RRB@XM.CO.NZ
1. Computer Science Department, University of Waikato, Hamilton, New Zealand
2. Xtal Mountain Information Technology, Auckland, New Zealand

## Abstract

Replicability of machine learning experiments measures how likely it is that the outcome of one experiment is repeated when performed with a different randomization of the data. In this paper, we present an estimator of replicability of an experiment that is efficient. More precisely, the estimator is unbiased and has lowest variance in the class of estimators formed by a linear combination of outcomes of experiments on a given data set.

We gathered empirical data for comparing experiments consisting of different sampling schemes and hypothesis tests. Both factors are shown to have an impact on replicability of experiments. The data suggests that sign tests should not be used due to low replicability. Ranked sum tests show better performance, but the combination of a sorted runs sampling scheme with a t-test gives the most desirable performance judged on Type I and II error and replicability.

## 1. Introduction

Machine learning research on classifiers relies to a large extent on experimental observations. It is widely recognized that there are many pitfalls in performing experiments [3, 6, 8]. But, so far, most research in this area concentrates on undesirable high levels of Type I error, the situation where the experiment indicates that one classifier outperforms another, while in reality it does not. An often overlooked issue with experimental research is that the particular randomizations used in the experiment can have a major impact on the outcome of the experiment. This effect can be so large that for some experimental designs only in 2 out of 3 cases repetition of the experiment produces the same outcome [2].

Appearing in *Proceedings of the 21$^{st}$ International Conference on Machine Learning*, Banff, Canada, 2004. 

In this paper, we try to get a better insight in this issue of replicability and which factors in an experiment influence replicability. In order to do so, we need a practical definition of replicability and a way to measure replicability of an experiment. Once this is established we can actually perform experiments on various set-ups. In the following section, we consider a number of experimental designs. We continue in Section 3 with ways to estimate replicability and perform a theoretical analysis of their performance. Section 4 presents empirical results where we measure replicability for the various experimental set-ups. We finish with some concluding remarks.

## 2. Machine learning experiments

The problem we want to address is, given two learning algorithms $A$ and $B$ that generate classifiers and a small data set $D$, how to make a decision which of the two algorithms performs best based on classification accuracy for the given data set. A general method to make such a decision is to split $D$ into a training set $D_t$ and a test set $D \backslash D_t$. Then, train algorithm $A$ and $B$ on $D_t$ and register the classification accuracy on the $D \backslash D_t$. This way, we obtain two classification accuracies $P_A$ and $P_B$ and the difference $x = P_A - P_B$ gives an indication which algorithm performs better.

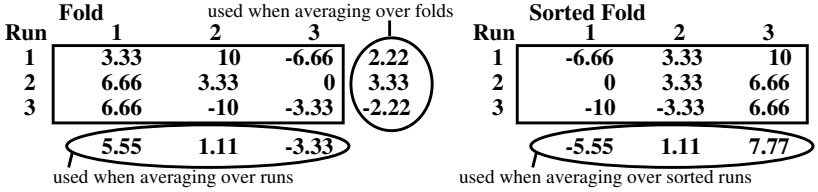
A formal way to make such a decision is to apply a hypothesis test. However, such hypothesis test typically requires more than a single outcome $x$. Unfortunately, for small datasets $D$, we have to split $D$ repeatedly in training and test sets to obtain multiple outcomes $P_{A,i}$ and $P_{B,i}$ with associated differences $x_i = P_{A,i} - P_{B,i}$, $1 \leq i \leq n$ obtaining a sample of size $n$.

So, an experiment has two components. Firstly, a sampling scheme for obtaining a sample $x_1, \ldots, x_n$, and secondly, a hypothesis test to make a decision based on the sample. There are various ways to obtain samples and to perform hypothesis tests.

### 2.1. Sampling methods

We consider six different sampling schemes.

*Figure 1.* Example illustrating the data used for the various sampling schemes.




**Resampling:** Resampling consist of splitting the data $n$ times in a randomly selected test set $D_{t,i}$ containing a fraction of the data (typically 10% to 33%) and a training set $D \backslash D_{t,i}$. The algorithms $A$ and $B$ learn on the training set and accuracies $P_{A,i}$ and $P_{B,i}$, $1 \leq i \leq n$ are obtained by classifying instances on the accompanying test set giving $n$ accuracy differences $x_i = P_{A,i} - P_{B,i}$ for the sample. Resampling used to be an accepted way for applying the t-test on the sample $x_1, \ldots, x_n$ till it was discredited by Dietterich [3] due to its extremely high Type I error. Nadeau and Bengio [6] showed how this problem can be solved by correcting the variance.

**K-fold cross validation:** Cross validation splits the data $D$ into $k$ approximately equal parts $D_1, \ldots, D_k$, and learns on the data $D \backslash D_i$, $1 \leq i \leq k$ with one part left out. The part $D_i$ left out is used as test set, giving $n = k$ accuracy differences $x_i = P_{A,i} - P_{B,i}$. Dietterich [3] observed a slightly elevated Type I error for cross validation with a t-test and its replicability is rather low [2].

**Use all data:** To obtain more samples, we can repeat $k$-fold cross validation $r$ times with different random splits into folds for each of the runs. This gives us $r \times k$ accuracy differences. Let $\hat{x}_{i,j}$, $1 \leq i \leq r$, $1 \leq j \leq k$ denote the difference in accuracy of algorithms $A$ and $B$ in the $i$th run on the $j$th fold. Here $A$ and $B$ are trained on the $k-1$ remaining folds in the $i$th run. We obtain a sample of size $n = r \times k$ by using all of the accuracy differences $x_{i,j}$ (formally by setting $x_i = \hat{x}_{i \mod r, \lceil i/r \rceil}$).

**Average over folds:** In averaging over folds, the recommended method for Weka [9], we take the result in a repeated cross validation experiment. We obtain one sample value per run by taking the average difference over all results for a single run, $x_i = \sum_{j=1}^{k} \hat{x}_{i,j}/k$ (where $\hat{x}_{i,j}$ as for the use all data scheme).

**Average over runs:** Averaging over folds can be interpreted as an improved way of doing resampling. The natural extension is performing an improved way of k-fold cross validation, and instead of averaging over folds, average over runs. We obtain one sample value per fold defined as the average difference $x_i = \sum_{a=1}^{r} \hat{x}_{a,i}/r$. Both averaging over folds and over runs show a very high Type I error when applying a t-test [2].

**Average over sorted runs:** Averaging over runs combines results from different runs rather arbitrarily. One gets better estimates of a k-fold cross validation experiment by first sorting the results for the individual k-fold cross validation experiments and then taking the average. This way, the estimate for the minimum value is calculated from the minimum values in all folds, the one but lowest from the one but lowest results in all folds, etc. Let $\hat{x}_{\theta(i,j)}$ be the $j$th highest value of accuracy difference $\hat{x}_{i'j'}$ of run $i$. Then, the sample consisting of $k$ values is defined by $x_i = \sum_{a=1}^{r} \hat{x}_{\theta(a,i)}/r$.

Figure 1 illustrates the difference between the data used for the sampling schemes. The figure shows an example of 3x3 fold cross validation outcomes in the box at the left half (though in practice a 10x10 fold cross validation is more appropriate). All the data in the box in Figure 1 is used for the "use all data" scheme. For resampling, essentially only the first column is required when performing a 2/3-1/3 split of training and test data. Cross validation uses only the first run, that is, the first row of a 3x3 fold cross validation outcome. Averaging over folds and runs is essentially summing over columns and rows respectively. For getting sorted means, first the results have to be sorted over folds, giving the table at the right of Figure 1. Then the means are obtained by summing over rows.

### 2.2. Hypothesis tests

In our experiment, we want to test the null hypothesis $H_0$ that $A$ and $B$ perform the same. More formally, we want to test whether the sample $x_1, \ldots, x_n$ has zero mean. There are different methods to test such hypothesis, all of which are based on slightly different assumptions. We consider the popular t-test, the sign test and the rank sum test, also known as Wilcoxon's test. All these tests assume that the outcomes $x_i$ in the sample are mutually independent, an assumption

that is obviously violated.

These hypothesis tests follow a similar procedure. First, we calculate a statistic $Z$ from the sample. Different tests have different methods of calculating $Z$ (see below). Then, we calculate the probability $p(Z)$ that the value $Z$ *or less* is observed assuming $H_0$ is true. We choose a significance level $\alpha$ and accept $H_0$ if $p(Z)$ is higher than $\alpha/2$ but less than $1-\alpha/2$. If $p(Z) < \alpha/2$, the test indicates $B$ outperforms $A$ and if $p(Z) > 1-\alpha/2$, the test indicates $A$ outperforms $B$.

**Paired t-test:** The assumption underlying the paired t-test is that the outcomes $x_i$ are normally distributed. If this is true, then the mean can be estimated using $\hat{m} = \frac{1}{n}\sum_{i=1}^{n} x_i$, the variance using $\hat{\sigma}^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \hat{m})^2$. With $n-1$ degrees of freedom ($df = n-1$) we have a statistic $Z = \frac{\hat{m}}{\sqrt{\hat{\sigma}^2}/\sqrt{df+1}}$, which is distributed according to Students t-distribution $P_t$ with $df$ degrees of freedom. The probability that the data $x_1, \ldots, x_n$ is observed assuming the null hypothesis is true is obtained by finding $P_t(T, df)$.

**Sign test:** The attractiveness of the sign test is that it is simple and makes no assumptions about the underlying distribution of the sample. Instead, it only looks at the signs of $x_1, \ldots, x_n$ and statistic $Z$ is the number of pluses. When accuracies $P_{A,i}$ and $P_{B,i}$ are the same, which occurs quite often when two algorithms perform very similarly, $x_i = 0$ and we count this as half a plus. If the null hypothesis is true, the probability of generating a plus or a minus is 0.5, in other words $H_0 : p = 0.5$. The probability of observing $Z$ pluses in $n$ comparisons is $P(Z) = \sum_{i=0}^{Z}\binom{n}{i}p^i(1-p)^{n-i}$, which with $p = 0.5$ is $P(Z) = \sum_{i=0}^{Z}\binom{n}{i}\frac{1}{2}^n$.

**Rank sum test:** Like the sign test, the rank sum test makes no assumption about the underlying distribution of outcomes $x_i$. However, the rank sum test does exploit the size of the values of $x_i$, which contains potentially valuable information. The rank sum test sorts the outcomes $x_i$ on its *absolute* value, giving a set of outcomes $y_1, \ldots, y_n$, $|y_i| \leq |y_{i+1}|$ $(1 \leq i < n)$. When accuracies are the same (i.e. outcomes for which $x_i = 0$) they are removed from the sample, leaving $n'$ items. Now, we add the ranks of outcomes that are positive, $r = \sum_{i=1, y_i>0}^{n'} i$. This statistic has mean $m = \frac{n'(n'+1)}{4}$ and variance $\sigma^2 = \frac{n'(n'+1)(n'+2)}{24}$ and is approximately normally distributed. So, we use $Z = \frac{r-m}{\sigma}$, which is normally distributed with mean 0 and variance 1.

### 2.3. Quality of experiments

There are essentially three methods to judge the quality of an experiment:

- The *Type I error* is the probability that the conclusion of an experiment is there is a difference between algorithms, while in reality there is not. In theory, the Type I error equals the significance level chosen for the hypothesis test if none of the assumptions of the test are violated. In practice, the independence assumption is often violated resulting in an elevated Type I error.
- The *Type II error* is the probability the conclusion of an experiment is there is no difference between algorithms, while in reality there is. The *power* is defined as 1 minus the Type II error. The power is not directly controllable like the Type I error is. However, there is a trade-off between power and Type I error and a higher power can be obtained at the cost of a higher Type I error. The exact relation between the two depends on the experimental design.
- *Replicability* of an experiment is a measure of how well the outcome of an experiment can be reproduced.

The most desirable experiment has a low Type I error, a high power an high replicability. In the following section we will have a closer look at replicability.

## 3. Replicability

In [2], an ad hoc definition for replicability was proposed as follows. When an experiment is repeated ten times with different randomizations of a given data set, the experiment is deemed replicable if its outcome is the same for all ten experiments. If one or more outcomes differ, it is not replicable. An impression of the replicability of an experiment can be obtained by averaging over a large number (say 1000) of data sets. This definition is useful in highlighting that replicability of experiments is indeed an issue in machine learning. However, the disadvantage is that replicability measured this way cannot be compared with results for doing the experiment another number than ten times. Also, replicability defined this way would not distinguish between having 1 out of 10 outcomes being different and 5 out of 10 outcomes being different. Further, increasing the number of experiments to say 100 increases the likelihood that one of the experiments differ and thus decreases replicability according to the definition of [2]. A definition of replicability that does not suffer from these issues is the following.

**Definition:** *Replicability* of an experiment is the probability two runs of the experiment on the same data set, with the same pair of algorithms and the same

method of sampling the data produces the same outcome.

This definition applies both in the situation where the algorithms perform the same and when one outperforms another. Note the difference between Type I error and replicability. When the algorithms perform the same, the Type I error expresses the probability *over all data sets* that a difference is found. Replicability only expresses that error for *one data set*.

By defining replicability in terms of probabilities, one can compare replicability of different experiments with different experimental set-ups and number of runs. Furthermore, an experiment that produces 9 same outcomes out of 10 has a higher replicability this way than when it only produces 5 same outcomes out of 10.

Note that replicability always lies between 50% and 100%. *Normalized replicability* is replicability linearly scaled to the range 0% to 100%. So, if replicability is $r$, normalized replicability is $2(r - \frac{1}{2})$.

### 3.1. A simple estimator

The only way to determine the replicability of an experiment is to measure it empirically. So, we need an estimator of replicability. A simple approach is to obtain pairs of runs of an experiment on a data set $D$ and just interpret those as the outcome of Bernoulli trial with probability $r$ that the outcomes are the same.

The outcome $e$ of an experiment on data set $D$ is 'accept' or 'reject'. When the outcome is 'accept' the null hypothesis that the two learning algorithms perform the same on $D$ is accepted, otherwise they are not.

**Definition** Let $\mathbf{e} = e_1, \ldots, e_n$ ($n > 0$ and $n$ even) be the outcomes of $n$ experiments with different randomizations on data set $D$. The estimator $\hat{R}_1$ of replicability $r$ is

$$\hat{R}_1(\mathbf{e}) = \frac{\sum_{i=1}^{n/2} I(e_{2i} = e_{2i-1})}{n/2}$$

where $I$ is the indicator function, which is 1 if its argument is true, and 0 otherwise.

We write $\hat{R}_1$ if it is clear from the context what the argument $\mathbf{e}$ of $\hat{R}_1(\mathbf{e})$ is.

LEMMA 3.1 *$\hat{R}_1$ is an unbiased estimator of replicability $r$ with variance $\frac{r-r^2}{n/2}$.*

**Proof:** The bias of $\hat{R}_1$ is $E(\hat{R}_1) - r$. Now, $E(\hat{R}_1) = E(\frac{\sum_{i=1}^{n/2} I(e_{2i}=e_{2i-1})}{n/2})$. Taking the constant $\frac{1}{n/2}$ outside the expectation gives $E(\hat{R}_1) = \frac{1}{n/2} E(\sum_{i=1}^{n/2} I(e_{2i} = e_{2i-1}))$. Distributing the sum results in $E(\hat{R}_1) = \frac{1}{n/2} \sum_{i=1}^{n/2} E(I(e_{2i} = e_{2i-1}))$. Now, $E(I(e_{2i} = e_{2i-1})) = P(I(e_{2i} = e_{2i-1}))I(e_{2i} = e_{2i-1}) + P(I(e_{2i} = e_{2i-1}))I(e_{2i} \neq e_{2i-1})$. Note that $P(I(e_{2i} = e_{2i-1})) = r$ and $P(I(e_{2i} \neq e_{2i-1})) = 1 - r$ so we get $E(I(e_{2i} = e_{2i-1})) = r \cdot 1 + (1 - r) \cdot 0 = r$. Substituting in $E(\hat{R}_1)$ above gives $E(\hat{R}_1) = \frac{1}{n/2} \sum_{i=1}^{n/2} r = \frac{n/2}{n/2} r = r$. So, the bias of $\hat{R}_1 = E(\hat{R}_1) - r = r - r = 0$, which shows that $\hat{R}_1$ is an unbiased estimator of $r$.

The variance of $\hat{R}_1$ is $var(\hat{R}_1) = E(\hat{R}_1^2) - E(\hat{R}_1)^2 = \sum_{i=0}^{n/2} P(i \text{ same pairs out of } n/2)(\frac{i}{n/2})^2 - E(\hat{R}_1)^2$ where $\hat{R}_1 = \frac{1}{n/2}$. From the derivation above, we have $E(\hat{R}_1)^2 = r^2$. Further, observe that $P(i \text{ same pairs out of } n/2)$ follows the binomial distribution with probability $r$. So, we have $var(\hat{R}_1) = \sum_{i=0}^{n/2} r^i (1 - r)^{n/2-i} \binom{n/2}{i} (\frac{i}{n/2})^2 - r^2 = \frac{1}{(n/2)^2} \sum_{i=0}^{n/2} r^i (1 - r)^{n/2-i} \binom{n/2}{i} i^2 - r^2$. Using Lemma A.1 (see Appendix), $\sum_{i=0}^{n/2} r^i (1 - r)^{n/2-i} \binom{n/2}{i} i^2 = (n/2)^2 r^2 - r^2 n/2 + rn/2$ giving $var(\hat{R}_1) = \frac{1}{(n/2)^2}((n/2)^2 r^2 - r^2 n/2 + rn/2) - r^2 = \frac{r-r^2}{n/2}$. □

### 3.2. An advanced estimator

The simple estimator $\hat{R}_1$ uses experiment $e_1$ only to compare with $e_2$. Since $e_3$ is independent of $e_1$, one could compare $e_1$ with $e_3$ as well. Likewise, the pair $(e_1, e_k)$ for any $k > 1$ could be compared and used in the estimate of replicability. In fact, we can use all pairs of outcomes and estimate replicability as the fraction of pairs with the same outcome. This defines a new estimator $\hat{R}_2$.

**Definition** Let $\mathbf{e} = e_1, \ldots, e_n$ and $n$ as before, then we define estimator $\hat{R}_2(\mathbf{e})$ of $r$ as

$$\hat{R}_2(\mathbf{e}) = \sum_{1 \leq i < j \leq n} \frac{I(e_i = e_j)}{n \cdot (n-1)/2} \qquad (1)$$

According to the following lemma, we can actually calculate $\hat{R}_2$ directly from counting the number of accepted tests out of the $n$ experiments. So, $\hat{R}_2$ can be calculated efficiently in linear time of the number of experiments.

LEMMA 3.2 *Let $\mathbf{e} = e_1, \ldots, e_n$ and $n$ as before and $i$ out of $n$ tests be accepting the null hypothesis, then*

$$\hat{R}_2(\mathbf{e}) = \hat{R}_2(i, n) = \frac{i \cdot (i-1) + (n-i) \cdot (n-i-1)}{n \cdot (n-1)}$$

**Proof:** The numerator of $\hat{R}_2$ in (1) is the number of pairs with equal outcomes. If $i$ ($0 \leq i \leq n$) tests accept the null hypothesis and the remaining $n-i$ do not, then $\binom{i}{2}$ pairs of rejecting pairs and $\binom{n-i}{2}$ pairs of non rejecting pairs can be formed. This gives an estimate of replicability as $\hat{R}_2(i,n) = (\binom{i}{2} + \binom{n-i}{2})/(n(n-1)/2) = \frac{i(i-1)+(n-i)(n-i-1)}{n(n-1)}$. □

Now, we will examine the bias and variance of $\hat{R}_2$. It turns out that $\hat{R}_2$ is an unbiased estimator of replicability and its variance can be expressed in closed form.

THEOREM 3.1 *$\hat{R}_2$ is an unbiased estimator of replicability $r$ with variance $\frac{1}{n\cdot(n-1)} \cdot (2(n-2)(n-3)F(p,4) + (4-2(n-3))(n-2)F(p,3) + (n-2)(n-3)+2)F(p,2) - r^2$ where $F(p,x) = p^x + (1-p)^x$ and $p = 1/2 + 1/2\sqrt{2r-1}$.*

The proof that $\hat{R}_2$ is unbiased closely follows that of Lemma 3.1. The proof establishing the variance of $\hat{R}_2$ is rather technical and is omitted here. A full proof is available in the report version of this paper.

Unfortunately, the closed form expression for the variance of $\hat{R}_2$ is hard to interpret and compare with that of $\hat{R}_1$. Figure 2 shows the variance of $\hat{R}_1$ and $\hat{R}_2$ for various values of replicability $r$ and number of experiments $n$. It shows that the variance of $\hat{R}_2$ is equal to that of $\hat{R}_1$ when $r = 1$. This is when there is full replicability and in this case the variance is zero. However, for other values of $r$, the variance of $\hat{R}_2$ is always below that of $\hat{R}_1$, indicating that $\hat{R}_2$ is a more efficient estimator of replicability than $\hat{R}_1$.

### 3.3. Is there a better estimator?

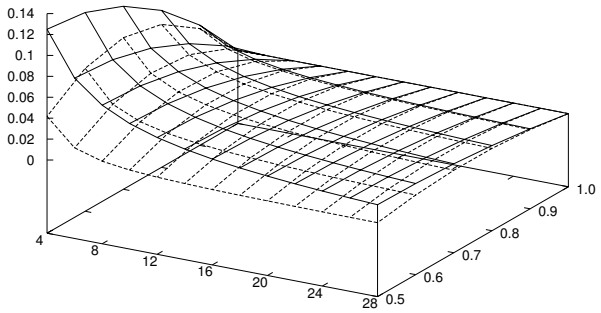
Is there an unbiased estimator of replicability with lower variance than $\hat{R}_2$ based on experiments $\mathbf{e} = e_1, \ldots, e_n$ on a single database? We will consider the class of estimators based on linear functions of $I(e_i = e_j)$.

**Definition:** Let $\mathbf{e} = e_1, \ldots, e_n$ ($n > 0$ and $n$ even) be the outcomes of $n$ experiments with different randomizations on data set $D$. Then estimator $\hat{R}_\mathbf{k}$ of $r$ is

$$\hat{R}_\mathbf{k}(\mathbf{e}) = \sum_{1 \leq i < j \leq n} k_{i,j} I(e_i = e_j) \qquad (2)$$

Note that $\hat{R}_1$ is in this class with $k_{i,i+1} = \frac{1}{n/2}$ for odd $i$ and $k_{i,j} = 0$ otherwise. Likewise, $\hat{R}_2$ is in this class with $k_{i,j} = \frac{1}{n(n-1)/2}$ for all $1 \leq i < j \leq n$. If we demand that $\hat{R}_\mathbf{k}$ is unbiased, we put a restriction on the coefficients $k_{i,j}$ expressed in the following lemma.

*Figure 2.* Variance of $\hat{R}_1$ (upper surface) and $\hat{R}_2$ (lower surface) as function of replicability $r \in (0.5 \ldots 1.0)$ and number of tests $n \in (4 \ldots 28)$




LEMMA 3.3 *$\hat{R}_\mathbf{k}$ is an unbiased estimator of replicability $r$ iff*

$$\sum_{1 \leq i < j \leq n} k_{i,j} = 1 \qquad (3)$$

In the proof, we use the property that if $r$ is the replicability of an experiment for a given data set, then, by definition, $r$ is the probability two experiments produce the same outcome. Now, two experiments use different independent randomizations. So, if $p$ is the probability that the outcome of a single experiment is accept, then the replicability is the probability that two outcomes are accept ($p \cdot p$) plus the probability that two outcomes are reject ($(1-p) \cdot (1-p)$). So, $r = p \cdot p + (1-p) \cdot (1-p)$, which can be solved for $p$ giving $p = \frac{1}{2} \pm \frac{1}{2}\sqrt{2r-1}$.

**Proof:** For $\hat{R}_\mathbf{k}$ to be an unbiased estimator of replicability $r$, we must have $E(\hat{R}_\mathbf{k}) = r$. Now, $E(\hat{R}_\mathbf{k})$ by definition of expectation is $\sum_\mathbf{e} P(\mathbf{e}) \hat{R}_\mathbf{k}(\mathbf{e})$. Using (2), this equals $\sum_\mathbf{e} P(\mathbf{e}) \sum_{1 \leq i < j \leq n} k_{i,j} I(e_i = e_j)$. Changing the order of sums, we get $\sum_{1 \leq i < j \leq n} \sum_\mathbf{e} P(\mathbf{e}) k_{i,j} I(e_i = e_j)$. Note that $\sum_\mathbf{e} P(\mathbf{e})$ has equal outcomes for $e_i$ and $e_j$ only with probability $p^2$ (both accept) and $(1-p)^2$ (both rejects). So, $\sum_\mathbf{e} P(\mathbf{e}) k_{i,j} I(e_i = e_j) = (p^2 + (1-p)^2) k_{i,j} = r k_{i,j}$. Summing over $i$ and $j$ gives $\sum_{1 \leq i < j \leq n} r k_{i,j} = r \sum_{1 \leq i < j \leq n} k_{i,j} = r$ where the last equality follows from the condition that the estimator is unbiased. Consequently, $\sum_{1 \leq i < j \leq n} k_{i,j} = 1$. □

So, $\hat{R}_1$ and $\hat{R}_2$ being unbiased (Lemma 3.1 and Theorem 3.1) can be proven observing $\hat{R}_1$ and $\hat{R}_2$ are instances of $\hat{R}_\mathbf{k}$ and noting that the coefficients $k_{i,j}$ add to 1.

THEOREM 3.2 *$var(\hat{R}_\mathbf{k}) \geq var(\hat{R}_2)$ for any unbiased*

*Table 1.* Type I error on Set 1, power on Set 2, 3 and 4 and replicability (in percentages) for various sampling methods (95% confidence interval in brackets).

| Test | Sampling scheme | Source 1 Type I | Source 2 Power | Source 3 Power | Source 4 Power | Minimum average norm.replicability |
|---|---|---|---|---|---|---|
| Rank sum test | Resampling | 14.8 (±0.4) | 27.9 (±0.6) | 48.0 (±0.3) | 95.7 (±0.3) | 34.0 (±0.5) |
| | k-fold cv | 11.0 (±0.2) | 23.2 (±0.2) | 45.8 (±0.7) | 97.5 (±0.2) | 46.0 (±0.5) |
| | Use all data | 55.2 (±0.3) | 71.5 (±0.2) | 88.0 (±0.1) | 100.0 (±0.0) | 61.8 (±0.6) |
| | Average over folds | 59.8 (±0.5) | 78.9 (±0.4) | 90.2 (±0.2) | 100.0 (±0.0) | 56.4 (±0.7) |
| | Average over runs | 50.5 (±0.6) | 68.3 (±0.1) | 86.1 (±0.1) | 100.0 (±0.0) | 57.0 (±0.9) |
| | Average sorted runs | 4.1 (±0.1) | 20.2 (±0.3) | 46.8 (±0.5) | 99.3 (±0.1) | 80.6 (±0.2) |
| Sign test | Average sorted runs | 5.0 (±0.5) | 21.2 (±0.4) | 48.6 (±0.3) | 99.1 (±0.1) | 75.2 (±0.7) |
| T-test | Average sorted runs | 4.5 (±0.1) | 21.1 (±0.2) | 51.7 (±0.5) | 99.6 (±0.1) | 81.6 (±0.6) |

*estimator* $\hat{R}_{\mathbf{k}}$.

**Proof:** We determine the minimum of $var(\hat{R}_{\mathbf{k}})$ and show that $\hat{R}_2$ realizes the minimum. By definition, $var(\hat{R}_{\mathbf{k}})$ equals $E(\hat{R}_{\mathbf{k}}^2) - E(\hat{R}_{\mathbf{k}})^2$. Since $\hat{R}_{\mathbf{k}}$ is unbiased, $E(\hat{R}_{\mathbf{k}}) = r$ so $var(\hat{R}_{\mathbf{k}}) = E(\hat{R}_{\mathbf{k}}^2) - r^2 = \sum_{\mathbf{e}} P(\mathbf{e})\hat{R}_{\mathbf{k}}^2(\mathbf{e}) - r^2$.

At the minimum, $dvar(\hat{R}_{\mathbf{k}})/dk_{i,j} = 0$ for all $1 \leq i < j \leq n$. Taking derivatives w.r.t. $k_{a,b}$ for any $a, b$ such that $(a,b) \neq (1,2)$ gives $dvar(\hat{R}_{\mathbf{k}})/dk_{a,b} = d\sum_{\mathbf{e}} P(\mathbf{e})\hat{R}_{\mathbf{k}}^2(\mathbf{e}) - r^2/dk_{a,b}$ which computes as $\sum_{\mathbf{e}} P(\mathbf{e})2\hat{R}_{\mathbf{k}}(\mathbf{e})(d\hat{R}_{\mathbf{k}}(\mathbf{e})/dk_{a,b})$.

We can write $\hat{R}_{\mathbf{k}} = \sum_{1\leq i<j\leq n, j>2} k_{i,j} I(e_i = e_j) + k_{1,2} I(e_1 = e_2)$ and use (3) to write $k_{1,2} = 1 - \sum_{1\leq i<j\leq n, j>2} k_{i,j}$, giving $\hat{R}_{\mathbf{k}} = \sum_{1\leq i<j\leq n, j>2} k_{i,j} I(e_i = e_j) + (1 - \sum_{1\leq i<j\leq n, j>2} k_{i,j}) I(e_1 = e_2)$. So, the term $d\hat{R}_{\mathbf{k}}(\mathbf{e})/dk_{a,b}$ can be written as $d\sum_{1\leq i<j\leq n, j>2} k_{i,j} I(e_i = e_j) + (1 - \sum_{1\leq i<j\leq n, j>2} k_{i,j}) I(e_1 = e_2)/dk_{a,b}$ which equals $I(e_a = e_b) - I(e_1 = e_2)$. So $dvar(\hat{R}_{\mathbf{k}})/dk_{a,b}$ is $\sum_{\mathbf{e}} P(\mathbf{e})2\hat{R}_{\mathbf{k}}(\mathbf{e})(I(e_a = e_b) - I(e_1 = e_2))$.

We need to distinguish two cases, namely $a \leq 2$ and $a > 2$. If $a > 2$, $dvar(\hat{R}_{\mathbf{k}})/dk_{a,b}$ is $\sum_{\mathbf{e}} P(\mathbf{e})2\sum_{1\leq i<j\leq n} k_{i,j} I(e_i = e_j)(I(e_a = e_b) - I(e_1 = e_2))$ reduces to $2k_{a,b}(p(1-p)^3 + p^3(1-p)) - 2k_{1,2}(p(1-p)^3 + p^3(1-p))$ where $p = \frac{1}{2} + \frac{1}{2}\sqrt{2r-1}$ as before. For this to equal zero, we have $p = 0$ or $p = 1$ coinciding with replicability of $r = 1$, or $k_{a,b} = k_{1,2}$. Likewise, if $a \leq 2$ $dvar(\hat{R}_{\mathbf{k}})/dk_{a,b}$ reduces to $2k_{a,b}(p(1-p)^2 + p^2(1-p)) - 2k_{1,2}(p(1-p)^2 + p^2(1-p))$. And again, we have $r = 1$ or $k_{a,b} = k_{1,2}$.

So, $var(\hat{R}_{\mathbf{k}})$ reaches an optimum at $k_{a,b} = k_{1,2}$ for all $a, b$, which means all coefficients are equal. And since they sum to 1, we have $k_{a,b} = \frac{1}{n(n-1)/2}$ since there are $n(n-1)/2$ coefficients.

The optimum is a minimum, as Figure 2 shows. □

In summary, Theorem 3.2 states that $\hat{R}_2$ is indeed an efficient (i.e. unbiased with lowest variance) estimator in the class of unbiased estimators $\hat{R}_{\mathbf{k}}$.

## 4. Empirical results

First, we establish which sampling scheme results in acceptable experiments based on Type I error and power. Then, we look at factors that impact replicability. To measure Type I error and power, algorithm $A$ (naive Bayes [5] as implemented in Weka 3.3 [9]) and algorithm $B$ (C4.5 [7] as implemented in Weka with default parameters) were compared on synthetic data and UCI data sets. The synthetic data sets was generated using four data sources based on four randomly generated Bayesian networks ([2] for more details). The data sets contained 10 binary variables and 50% class probability. Each of the data sources were used to generate 1000 data sets with 300 instances. Data source 1 has mutually independent variables, so there is no performance difference between naive Bayes and C4.5, which allows us to measure the Type I error. For sources 2, 3 and 4, C4.5 outperforms naive Bayes with increasing margin (on average 2.77%, 5.83% and 11.27% respectively as measured on 10.000 instance test sets), which allows us to measure the power of tests. The sampling methods mentioned in Section 2.1 were performed 10 times with 10 folds and 10 runs at 5% significance level.

Table 1 shows the results on the synthetic data with numbers in brackets indicating a 95% confidence interval. The first six rows are for the rank sum test. Note that the use all data, average over folds and over runs sampling schemes have a Type I error over 50%, while a 5% Type I error is desired. The resampling scheme has an elevated Type I error as has the 10 fold cross validation scheme. Only the sorted runs scheme shows an appropriate level of Type I error. This comes at the

*Table 2.* UCI data sets. Nr of draws of sorted 10 x 10 fold cv ($\alpha = 5\%$, 95% intervals for $\hat{R}_2$ within $\pm 3\%$)

| Data set | 123456789 | 10+ 01234567890 | 20+ 01234567 | Mean norm. $\hat{R}_2$ |
|---|---|---|---|---|
| Sign test | | | | 84.4 |
| NB vs C45 | ..9.....2 | .4..3..27. | ......5. | 78.2 |
| NB vs NN | .....96.9 | ....19.9.. | ....6... | 84.6 |
| C45 vs NN | ...1.4... | .........5 | ........ | 90.4 |
| Rank test | | | | 90.2 |
| NB vs C45 | ......... | .4..3...9. | ......7. | 87.6 |
| NB vs NN | ......8.. | .......9.. | ....8... | 93.2 |
| C45 vs NN | ...2.6... | .........7 | ........ | 90.0 |
| T test | | | | 90.8 |
| NB vs C45 | ......... | .1..2...9. | ......7. | 91.0 |
| NB vs NN | ......9.. | .......9.. | ....7... | 93.6 |
| C45 vs NN | ...2.6... | .........6 | .......9 | 88.0 |

cost of decreased power compared to most of the other schemes.

Table 1 also shows the minimum of the average replicability over Set 1 to 4. It shows that resampling and 10-fold cv has a level of replicability which is not acceptable (below 50%). The schemes based on repeated cross validation do show acceptable replicability. In particular, the sorted runs scheme has a replicability of over 80%. Results for the sign test and t-test are similar to the results for the rank sum test.

Table 1 also shows Type I error and power for sorted runs with sign test and t test. Those figures are very close to the ones for the rank sum test, taking in account that a slightly higher Type I error should lead to slightly better power. The replicability for sorted runs with the sign test is 75.2% and with the t-test is 81.6%. Compared to the 80.6% for the rank sum test, the sign test performs considerably worse. This can be explained by the lack of exploiting sizes of differences in the sample by the sign test. The replicability of the t-test is only slightly better.

Further experiments were performed using the sorted runs sampling scheme while varying various parameters of the experiment, namely
• significance level (1%, 2.5%, 5% and 10%),
• number of runs (10, 20, 30, 40, 50, 60, 70, 80, 90 and 100),
• class probability for binary data (0.1, 0.2, 0.3, 0.4 and 0.5),
• class cardinality (2, 3 and 4),
• different pairs of algorithms (out of Naive Bayes, C4.5, nearest neighbor, tree augmented naive Bayes, decision stump and support vector).

Though space limitations prevent us from presenting the complete set of outcomes here, we can report that the experiments resulted in a Type I error not exceeding the significance level by more than 1% with the sorted runs sampling scheme for all three tests considered. Decreasing the class probability increased replicability. The explanation for this behavior can be found in realizing that learners tend to predict the majority class the more this class dominates the data. Increasing the number of runs consistently increased replicability. It appears that the sorted runs sampling scheme results in a sample for which the independence assumption is not heavily violated, so that no correction in variance [6] or degrees of freedom [2] is required.

Table 2 shows results for 27 data sets from the UCI repository [1][1] using the sorted runs sampling scheme with the three different types of tests. We compared naive Bayes, C4.5 and nearest neighbor (NB, C4.5 and NN respectively in Table 2). Each algorithm was run ten times. The middle three columns show the number of times that the experiment decides that the null hypothesis is acceptable (so algorithms perform equal on a data set) as numbered in the footnote[1]. When the null hypothesis is 0 or 10 times accepted only a dot is shown, since both situations indicate perfect replicability. The first observation is that replicability is an issue for non-synthetic data sets, and thus affects many machine learning researchers. Further, the sign test performs much worse than the other two tests, while the t-test shows marginally higher replicability than the rank sum test. So, not only the sampling method, but also the hypothesis test has an impact on the replicability of the experiment.

[1] 1: anneal, 2: arrhythmia, 3: audiology, 4: autos, 5: balance-scale, 6: breast-cancer, 7: credit-rating, 8: ecoli, 9: German credit, 10: glass, 11: heart-statlog, 12: hepatitis, 13: horse-colic, 14: Hungarian heart disease, 15: ionosphere, 16: iris, 17: labor, 18: lymphography, 19: pima-diabetes, 20: primary-tumor, 21: sonar, 22: soybean, 23: vehicle, 24: vote, 25: vowel, 26: Wisconsin breast cancer, and 27: zoo.

## 5. Conclusions

We defined replicability of machine learning experiments in terms of probability. This has the benefit that it allows for comparison over different experimental designs, unlike a previous rather ad hoc definition [2]. For example, replicability measured on $n$ repeats of an experiment can be compared with replicability measure on $2n$ repeats. Furthermore, threshold effects present in the ad hoc definition are not present in our definition.

The main theoretical result of this paper is the presentation of an estimator for replicability that was shown to be unbiased and which has the lowest variance in its class. Using this estimator, we gathered empirical data to gain new insights in how experimental designs influence replicability and found that the hypothesis test, the sampling scheme, and the class probability impact replicability. In our experiments, replicability consistently increased with sampling methods that draw more samples from the same data set. Replicability appears to be an issue both with synthetic data sets as well as with UCI data sets. This indicates that machine learning researcher and data analysts should be wary when interpreting experimental results.

The main practical outcome of the experiments is that judged on replicability the sorted runs sampling scheme with the widely used t-test showed superior properties compared to the sign test and performed marginally better than the rank sum test. The sorted runs scheme is based on combining accuracy estimates in a way that produces a representative sample of accuracy differences of learning algorithms. Surprisingly, the sorted runs sampling schemes is the only scheme out of a set of popular schemes we considered that also showed acceptable Type I errors and reasonable power for a wide range of parameters using the three hypothesis tests considered. Consequently, experiments based on sorted runs sampling schemes do not require variance corrections [6] or calibration of degrees of freedom [2]. In summary, based on replicability, Type I error, power and theoretical considerations, we recommend using the sorted runs sampling scheme with a t-test for comparing classifiers on a small data set.

One would expect that replicability ceases to be an issue with larger data sets. In the future, we would like to perform larger scale experiments to get a better insight in the relation between replicability, the number of samples taken in an experiment and data set size. This should also give a better insight in the relation between replicability and Type I and II error.

In this paper, we considered machine learning experiments in which we choose the best of two classifiers for a given data set. In practice, more than two classifiers are available. Also, machine learning researchers routinely compare algorithms over a large number of data sets. This leads to new replicability issues and multiple comparison problems, issues that require further research.

**Acknowledgements**

I would like to thank the Machine Learning Group of Waikato University for stimulating discussions and the anonymous reviewers for their helpful comments.

## A. Appendix

LEMMA A.1 *For $0 \leq p \leq 1$, and $n \geq 2$ a positive integer,*

$$\begin{aligned} \sum_{i=0}^{n} p^i (1-p)^{n-i} \binom{n}{i} &= 1 \\ \sum_{i=0}^{n} p^i (1-p)^{n-i} \binom{n}{i} i &= np \\ \sum_{i=0}^{n} p^i (1-p)^{n-i} \binom{n}{i} i^2 &= n^2p^2 - np^2 + np \end{aligned}$$

**Proof:** (sketch) The first equation is the binomial theorem [4]. The second follows from the observation that the term in sum is zero for $i = 0$, so the range of the sum can be changed to $1 \leq i \leq n$. Using $\binom{n}{i} = \binom{n-1}{i-1}\frac{n}{i}$ for $i > 0$ we can absorb the $i$ at the end of the sum, and taking $p$ outside the term in the summation, we can apply the binomial theorem. The third equation follows from a similar line of reasoning. □

## References

[1] C.L. Blake and C.J. Merz. UCI Repository of machine learning databases. Irvine, CA: University of California, 1998.

[2] R.R. Bouckaert. Choosing between two learning algorithms based on calibrated tests. ICML, 51–58, 2003.

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

[4] R.L. Graham, D.E. Knuth and O. Patashnik Concrete mathematics. Addison-Wesley, 1994.

[5] G.H. John and Pat Langley. Estimating Continuous Distributions in Bayesian Classifiers. UAI, 338–345, 1995.

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

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

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

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