# Low Replicability of Machine Learning Experiments is not a Small Data Set Phenomenon

Remco R. Bouckaert[1,2]
1. Computer Science Department, University of Waikato
2. Xtal Mountain Information Technology
New Zealand
remco@cs.waikato.ac.nz, rrb@xm.co.nz

## Abstract

This paper investigates the relation between replicability of experiments for deciding which of two algorithms performs better on a given data set. We prove that lack of replicability is not just a small data phenomenon (as was shown before [1, 2, 3]), but is present in experiments on medium and large data sets as well. Counter intuitively, under particular circumstances replicability can slightly decrease when more data is available for an experiment.

In this paper, we establish intuition in the relation between data set size, power and replicability. The main factors for improving replicability appear to be increasing the number of samples and carefully selecting a sampling method. Unfortunately, for large data sets and/or inefficient learning algorithms, this implies that experiments may take a long time to completion.

## 1. Introduction

Comparison of classifiers through emprical measures forms one of the foundations of meta learning and in fact machine learning in general. Almost any machine learning paper contains presentations of empirical data. Designing a machine learning experiment is a problem with a lot of pitfalls [4, 6, 7, 8]. One such pitfall is whether other researcher can replicate the data presented in such works is an issue that is often overlooked. It was shown [1, 2, 3], that many popular experiments for deciding which of two algorithms performs better for a given data set lack replicability when the data set is small. For example, for a paired t-test on a 10 fold cross validation experiment the probability that two experiments produce the same outcome can be as low as 66%, which is only slightly better than the 50% obtained by randomly selecting one of the two algorithms.

Research so far has concentrated on small data sets [1, 2, 3]. In this paper, we study the effect of data set size on replicability of experiments.

The problem we investigate is given a data set $D$ and two learning algorithms A and B, decide which one we should select for the given data set based on expected accuracy. In order to make the decision, we perform an experiment where we split $D$ in a training set $D_t \subset D$ and test set containing the remaining instances $D \backslash D_t$. Then we train A and B on $D_t$ and measure performance on $D \backslash D_t$. In some experimental designs, this process is repeated several times which provides us with a larger sample. We perform a hypothesis test on the sample with null hypothesis that the two algorithms perform the same on $D$.

This paper is organized as follows. The following section summarizes terms, definitions and various experimental designs. In Section 3, we perform a theoretical analysis of McNemar's test, which can be done thanks to the simplicity of the test. In Section 4, we look at the influence of the difference between algorithms A and B on replicability and establish a relation between power and replicability, in particular worst case replicability. We look at various experimental designs and perform an empirical analysis of them. We end with conclusions and directions for further research.

Appearing in *Proceedings of the ICML-2005 Workshop on Meta-learning*, Bonn, Germany, 2005. 

## 2. Experimental design

We summarize four popular and/or effective experimental designs suitable for selecting a learning algorithm for a given dataset. For more details see [4] for McNemar's test, [7] for (corrected) resampling and [1, 2, 3] for the remaining tests.

### 2.1. McNemar's test

McNemar's test [4] is based on a single train/test split of $D$. This can be considered a sign test [9] in disguise, so the only assumption is that the test instances are independently and identically distributed. We count the number of cases where algorithm A is correct but B is not, denoted by $n_{10}$, and the number of cases where B is correct but A is not, denoted by $n_{01}$. If A and B perform the same, we would expect that $n_{10}$ and $n_{01}$ are approximately the same. The statistic

$$T = \frac{(|n_{10} - n_{01}| - 1)^2}{n_{10} + n_{01}} \tag{1}$$

exploits this idea and $T$ is distributed approximately according to the $\chi^2$ distribution with 1 degree of freedom. So, at the 5% significance level we can reject the null hypothesis if $T > \chi^2_{1,0.95} = 3.841459$. Alternatively, one can use $T = \frac{|n_{10}-n_{01}|-1}{\sqrt{n_{10}+n_{01}}}$, which is normally distributed (for $n_{10} + n_{01} \geq 25$) [5] and easier to analyze.

### 2.2. Resampling test

Let $a_j$ and $b_j$ be the accuracy of algorithms $A$ and $B$ respectively, measured on run $j$ ($1 \leq j \leq r$). Assume that in each run $r_1$ instances are used for training, and the remaining $r_2$ instances for testing. Let $x_j$ be the difference $x_j = a_j - b_j$, and $\hat{\mu}$ and $\hat{\sigma}^2$ the estimates of the mean and variance of the $r$ differences. The statistic of the "corrected resampled $t$-test" is:

$$t = \frac{\hat{\mu}}{\sqrt{(\frac{1}{r} + \frac{r_2}{r_1})\hat{\sigma}^2}} \tag{2}$$

This statistic is approximately distributed according to Student's $t$ distribution with $r - 1$ degrees of freedom. The difference to the standard $t$-test (which shows unacceptable high Type I error [4]) is that the factor $\frac{1}{r}$ in the denominator has been replaced by the factor $\frac{1}{r} + \frac{r_2}{r_1}$. The result is a test which tends to have an acceptable Type I error [7].

### 2.3. Repeated Cross Validation test

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 $x_i = P_{A,i} - P_{B,i}$.

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 $x_i = \hat{x}_{i \mod r, \lceil i/r \rceil}$).

Let $\hat{\mu}$ and $\hat{\sigma}^2$ the estimates of the mean and variance of the accuracy differences $x_{ij}$, then the statistic (2) can be used with $r_1/r_2 = 1/(k-1)$.

### 2.4. Sorted Cross Validation test

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 [1, 2]. 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 $x_i = \sum_{a=1}^{r} \hat{x}_{\theta(a,i)}/r$. Let $\hat{\mu}$ and $\hat{\sigma}$ be the mean and variance of $x_i$, then $t = \frac{\hat{\mu}}{\sqrt{(\frac{1}{r})\hat{\sigma}^2}}$ (that is, like Equation (2), but without correction factor) is approximately distributed according to Student's $t$ distribution with $k - 1$ degrees of freedom.

### 2.5. Evaluating Experiments

Experiments can be judges on a number of criteria:

- 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.
- 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.
- *Replicability* of an experiment is a measure of how well the outcome of an experiment can be reproduced. It is defined as the probability that when performing

the same experiment twice, we get the same outcome in both experiments.

• *Computational cost* of an experiment is the amount of calculations required for performing the experiment. Since the computational complexity of training and testing of most learning algorithms grows proportional to the data set size, this is especially a concern with large data sets.

## 3. Analysis of McNemar's test

In this section, we analyze the Type I and worst case replicability of McNemar's test. We define *Type I Replicability* as the replicability of an experiment when the null hypothesis holds (so when the two learning algorithms have the same performance). An experiment can be expected to reject the null hypothesis while the null hypothesis holds with a probability equal to the significance level $\alpha$. In this situation, if two experiments with only different randomizations of the data would produce independent outcomes the Type I replicability would be equal to the probability both experiments reject the null hypothesis ($\alpha \cdot \alpha$) plus the probability both experiments accept the null hypothesis ($(1-\alpha)\cdot(\alpha-1)$). Together, this is $1-2\alpha+2\alpha^2$, which is the lower bound on Type I replicability. For example with $\alpha = 5\%$ minimal Type I replicability is 90.5%. Note that this lower bound is independent of data set size.

In reality, the outcomes of two experiments will not be the same since there is overlap in training and test sets. We analyze the Type I error for McNemar's test.

First, we slightly reformulate McNemar's test in order to facilitate analysis. One interpretation of McNemar's test is that it is a sign test [9] with $p = \frac{1}{2}$. For every instance in the test set, either algorithm A wins, contributing one to the number of pluses (#+) or algorithm B wins, contributing to the number of minuses (#−), or both perform the same and the instance is ignored. Now, $Z = \frac{p(+)-0.5}{\sqrt{\frac{1}{2}\cdot\frac{1}{2}/n}}$ with $p(+) = \frac{\#+}{n}$ and $n = (\#+) + (\#-)$ is approximately normally distributed. Note that we can rewrite $Z$ as $Z = \frac{p(+)-0.5}{\sqrt{\frac{1}{2}\cdot\frac{1}{2}/n}} = \frac{2(\frac{\#+}{n}-1)}{\sqrt{1/n}} = \frac{1/n(2(\#+)-(\#+)-(\#-))}{\sqrt{1/n}} = \frac{(\#+)-(\#-)}{\sqrt{n}} = (n_{01}-n_{10})/\sqrt{n_{01}+n_{10}}$. Comparing with Equation (1) shows that $T$ is just the square of $Z$ when we ignore the continuity correction term $'-1'$. This is allowed since its influence vanishes with large data sets, so we ignore it in the following analysis. So, we can reject the null hypothesis in a two sided test if $Z > Z_{\alpha/2} = N_{\alpha/2}(0,1) = 1.96$ for $\alpha = 5\%$.

### 3.1. Type I replicability

Assume algorithm A and B perform the same on a domain from which data set $D$ is drawn. Consider two experiments using McNemar's test with $100(1-t)\%/100t\%$ test/train split of the data $D$ at significance level $\alpha$ with corresponding threshold point $Z_{\alpha/2}$.

With probability $1-\alpha$, the first experiment accepts the null hypothesis (note that this is regardless of the size of $D$). Let $D_{t,1}$ and $D_{t,2}$ be the test set in experiment 1 and 2 respectively. In the optimal situation, algorithm A and B produce the same outcomes on instances in $D_{t,1} \cap D_{t,2}$ in both experiments. (This is a reasonable assumption for large data sets and stable learning algorithms.) The expected sum of $'+'$s and $'-'$s (denoted by $x$) in $D_{t,1}$ is $\int_{-Z_{\alpha/2}}^{Z_{\alpha/2}} x \cdot N(0,\frac{1}{2})dx$ which is 0 by symmetry of the normal distribution $N(0,\frac{1}{2})$. So, the distribution of $'+'$s and $'-'$s in $D_{t,2} \cap D_{t,1}$ is equal to that in $D_{t,2} \backslash D_{t,1}$. Therefore, the probability that the second experiment accepts the null hypothesis given that the first experiment did is $1-\alpha$. In total, contributing $(1-\alpha)\cdot(1-\alpha)$ to replicability.

With probability $\alpha$, the first experiment will reject the null hypothesis. Assume that there are no draws in the experiment, hence $n = |D_{t,1}| = |D_{t,2}|$ (later we will relax this assumption). The expected sum of $'+'$s and $'-'$s (denoted by $x$) in $D_{t,1}$ is $E(x|x > Z_{\alpha/2}) = \int_{Z_{\alpha/2}}^{\infty} x \cdot P(x|x > Z_{\alpha/2})dx = \int_{Z_{\alpha/2}}^{\infty} x \cdot P(x, x > Z_{\alpha/2})/P(x > Z_{\alpha/2})dx = \int_{Z_{\alpha/2}}^{\infty} x \cdot N(0,1)/(\alpha/2)dx = \int_{Z_{\alpha/2}}^{\infty} x \cdot \frac{1}{\sqrt{2\pi}}e^{-x^2/2}dx/(\alpha/2) = -\frac{1}{\sqrt{2\pi}}e^{-x^2/2}|_{Z_{\alpha/2}}^{\infty}/(\alpha/2) = \frac{1}{\sqrt{2\pi}}e^{-Z_{\alpha/2}^2/2}/(\alpha/2)$ which is approximately $Z_{\alpha/2}$ (actually, slightly higher).[1]

So, the expected number of '+'s in $D_{t,2} \cap D_{t,1}$ is $Z_{\alpha/2}$, while in $D_{t,2} \backslash D_{t,1}$ it still is 0. So, the probability of selecting a '+' inside $D_{t,2} \cap D_{t,1}$ is $\frac{1}{2} + Z_{\alpha/2}/n$ where $n = n_{01} + n_{10}$. The probability of selecting an instance in $D_{t,2} \cap D_{t,1}$ is proportional to the test set size $t$. Therefore, the probability of selecting a '+' in $D_{t,2}$ is $t \cdot (\frac{1}{2} + Z_{\alpha/2}/n)$ plus $(1-t)\cdot\frac{1}{2}$ equalling $p' = \frac{1}{2} + t \cdot Z_{\alpha/2}/n$.

So instead of drawing from $D$ with $p = \frac{1}{2}$, we draw with $p'$. This implies that $m' = E\{n_{01} - n_{10}\} = np' - n(1-p') = 2 \cdot t \cdot Z_{\alpha/2}$ and variance $\sigma' = \sqrt{np'(1-p')}$. Therefore, the probability that experiment 2 rejects the null hypothesis given experiment 1 does is the probability $N(m', \sigma') = N(2tZ_{\alpha/2}, \sqrt{np'(1-p')}) \approx N(2tZ_{\alpha/2}, \frac{1}{2}\sqrt{n}) > Z_{\alpha/2}$, since $p'$ is approximately $\frac{1}{2}$

[1] Using `plot [x=0:8] 1.0/(sqrt(2.0*pi))*exp(-(x*x)/2.0)/(1-norm(x)),x,0.97*x+0.33` in gnuplot shows this.

for large $n$. So, the probability that experiment 2 rejects the null hypothesis given that experiment 1 rejects goes to $\alpha + \epsilon$ where $\epsilon$ increases with increased $t$ and $\epsilon \to 0$ with increasing $n$. In total, this case is contributing $\alpha \cdot (\alpha + \epsilon)$ to replicability.

In case there are draws in the two experiments, ...

Total replicability is $(1-\alpha)\cdot(1-\alpha) + \alpha \cdot (\alpha + \epsilon) = 1 - (2-\epsilon)\alpha + \alpha^2$ where $\epsilon$ increases with increasing test set size $t$ and decreases with increasing data set size. Surprisingly, Type I replicability decreases with increasing data set size.

### 3.2. Worst case replicability

In the appendix, we perform a formal analysis of the worst case replicability of McNemar's test. In summary, under some mild assumptions, worst case replicability does not depend on the size of the data set and increasing the data set does not increase replicability.

## 4. Power and Replicability

In this section, we consider the behavior of various experimental designs empirically. We use an artificial learning task, which allows us to control various parameters in the experiment.

**Task I**: Consider a data source over a binary class variable $y \in \{0, 1\}$ and an independent binary attribute $x \in \{0, 1\}$, each value occurring with equal probability. Learning algorithm A always predicts 1 while algorithm B assigns the value of $x$ to $y$, so both algorithms ignore the training set. Note that in this case the null hypothesis that A and B perform the same is true. The task is to decide whether A or B is better or whether both perform the same by performing an experiment on the data set.

We can measure power by controlling the expected difference in accuracy between algorithm A and B for Task I simply by controlling the probability that the data source produces combinations of $x$ and $y$. First, fix the probability that $y = 0$ to $\frac{1}{2}$ (so $P(y = 0) = P(y = 1) = \frac{1}{2}$). This way, algorithm A, which always classifies $y$ as 1, will have expected accuracy of $\frac{1}{2}$. (No reasonable algorithm will produce a worse accuracy on binary classes.) Further, fix the probability $x = 0$ and $y = 0$ to the probability $x = 1$ and $y = 1$ to $q$ (i.e. $P(x = 0, y = 0) = P(x = 1, y = 1) = q$) and likewise $P(x = 0, y = 1) = P(x = 1, y = 0) = \frac{1}{2} - q$. Note $q$ has to be less than $\frac{1}{2}$ to make the distribution sum to 1. The accuracy of Algorithm B (which predicts for $y$ the values of $x$) is the probability that $x = 0$ and $y = 0$ plus the probability $x = 1$ and $y = 1$, which is $P(x = 0, y = 0) + P(x = 1, y = 1) = q + q = 2q$. Further, if $q$ less than $\frac{1}{4}$, the accuracy of Algorithm B is less than $\frac{1}{2}$, which is unreasonable since inverting its prediction would increase the accuracy. So, it is reasonable to vary $q$ in the interval $[\frac{1}{4}, \ldots, \frac{1}{2}]$. The expected difference in accuracy between A and B is $2q - \frac{1}{2}$ (since the expected accuracy of B is $2q$ minus expected acc. of A is $\frac{1}{2}$), so this is completely controlled by parameter $q$.

We draw 1000 data sets and perform experiments 10 times on each of the data sets drawn and increase the difference in accuracy of A and B in 100 steps. So, a total of 1000x10x100 is one million experiments is performed for each data set size.

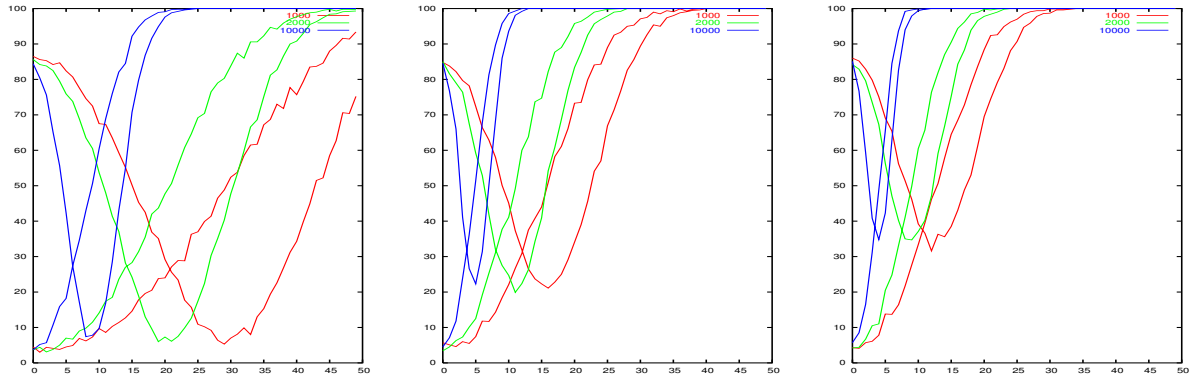
### 4.1. McNemar's test

Figure 1 shows power and replicability of a McNemar experiment with 10%/90% test train split.
In general, power increases with increasing data set sizes, a well known property of hypothesis tests. Replicability decreases where power increases up to 50% and increases after that point. Worst case replicability occurs where power is circa 50%. Replicability goes to 100% where the power goes to 100%. Type I replicability slightly decreases with increasing data set size. This trend continues; for 90%/10/is 86.5%, 85.6%, 84.4%, 83.0%, 80.5% and 82.0% for 1, 2, 10, 20, 100 and 200 thousand instances in the data set respectively. This confirms our analysis in the section on independent data.

Also remarkable is that worst case replicability does not change with increasing data set size, which we expected from out theoretical analysis. Indeed, the trend continues for larger data set sizes. For $n = 20.000$, we found worst case replicability of 6.4% at difference in A and B of $\Delta = 6\%$ and 46.2% power, For $n = 100.000$, replicability is 7.0% ($\Delta = 3\%$, 54.4% power), and for $n = 200.000$, replicability is 6.7% ($\Delta = 2\%$, 51.8% power). So, it appears that worst case replicability is indeed a property inherent to an experiment.

Figure 1 also shows results for 33%/66% and 50%/50% test/train splits. Increasing the test set size increases power. As expected from our analysis, increasing the test set size increases worst case replicability. This is due to increasing overlap in test sets, which causes performance between experiments to be more similar, hence increase replicability of the experiments. However, increasing test set size does not increase Type 1 replicability. Worst case replicability is very low for each of the train/test split values. When an experiment is performed in the unacceptable replicability area (replicability less than 90%) another experimen-

*Figure 1.* McNemar test for various data set sizes. 90%/10% train/test split left, 66%/33% train/test split middle, and 50%/50% split right. X-axis shows the difference in accuracy of algorithm A and B. Y-axis shows power (monotonically increasing curves) and replicability (curves with a local minimum) in percentage.




tal design should be chosen. The results from Figure 1 suggest that increasing the test set size is one way of changing the experiment and increasing replicability. Alternatively, a resampling experiment should be performed.

### 4.2. Resampling

Figure 2 shows power and replicability of a 10x resampling (graph on the left) experiment for data set sizes of 1, 2 and 10 thousand instances. It shows the effect of increasing data set sizes on power and replicability. The trends in power and replicability are similar to that of 10x resampling; power increases with increasing data set size; replicability has a minimum where power is 50%; worst case replicability does not change with data set size.
Comparing with 90%/10% McNemar test (Figure 1) we see that power and worst case replicability increases. However, Type I replicability decreases. Comparing with 50%/50% McNemar test, we see that both power and replicability are worse over all. This comes at about a two times larger cost (processing 50% of data once for McNemar vs. processing 10% of data 10 times for 10x resampling).

Some interesting observations comparing 100x and 10x resampling; The Type I error remains about the same and power increases by taking more samples (about twice as good). Likewise, replicability increases by taking more samples. The worst case replicability is considerably better when taking more samples. Unlike McNemar's test, the type I replicability remains about the same when taking more samples. Worst case replicability also remains the same.

Figure 2 shows the results using 500x resampling (graph on the right). Again, trends observed for the McNemar and 10x resampling experiment are present. Some observations comparing 500x resampling with 10x and 100x resampling; Power does not increase considerably more than for 100x resampling. So, increasing the number of samples has its limits in increasing power. From previous experience [1], we observed no further increase in power after 100 samples. However, replicability remains increasing with increasing number of samples. Though replicability increases, the number of samples required increases more than linearly.

### 4.3. k-fold Cross Validation

Figure 3 shows results for repeated cross validation. Same trends as for McMenar test holds. Note 10x resampling takes the same computational effort as 1x10 fold cross validation (likewise 100x and 500x resampling take the same effort as 10x10 and 50x10 fold cross validation respectively). Comparing the corresponding graphs between resampling (Figure 3) and repeated cross validation (Figure 2) we observe that repeated cross validation has slightly better power for the same computational effort. Repeated cross validation has considerably better Type I replicability and worst case replicability. This can be explained by the larger guaranteed overlap of test sets and train sets when comparing two separate experiments. Like for resampling, increasing the number of repeats does not increase power after 100 samples. However, replicability does increase when increasing samples.

Figure 4 shows results for sorted folds [2] sampling scheme. Note that the profiles look very similar to those of the repeated cross validation scheme. The main difference is in the 1 time 10 cross validation

*Figure 2.* Resampling results (axis same as Figure 1, but x-axis is scaled). 10x resampling in the left graph, 100x in the middle, and 500x resampling in the right graph.

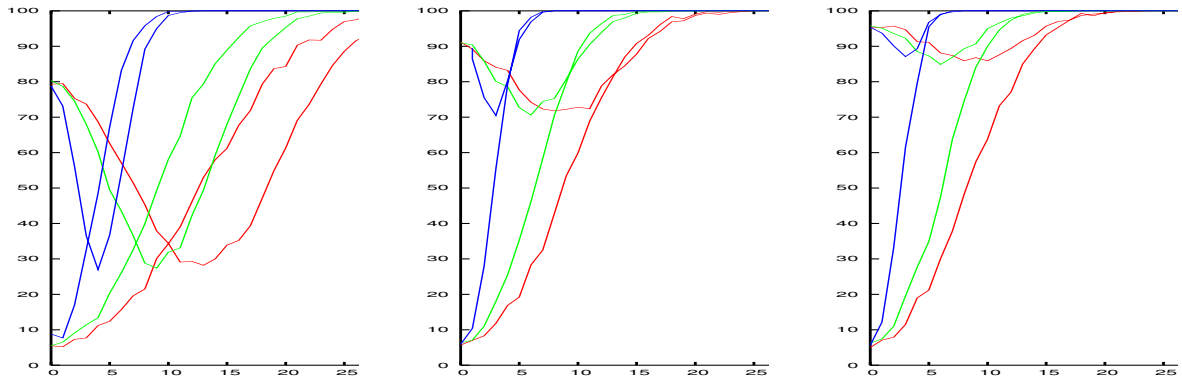
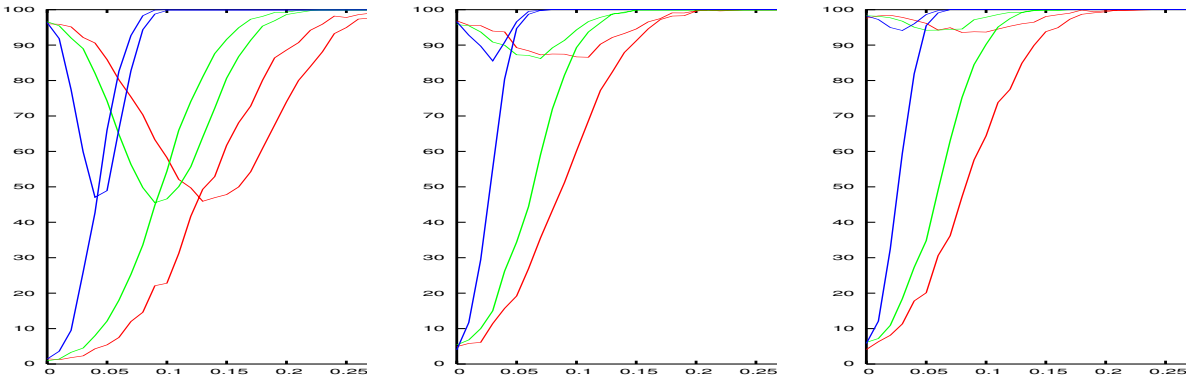



*Figure 3.* Repeated 10 fold cross validation results (axis same as Figure 2). 1x left, 10x middle, 50x right graph.



scheme due to the cross validation scheme using the variance correction factor, while the sorted scheme is not. Consequently, the Type I error for 1x10 cross validation is (too) low, hence the Type I replicability higher. Note that the worst case replicability is higher for the sorted than 1x10 cross validation, which indicates that the correction factor really is not appropriate for 1x10 cross validation.

We generated profiles for other sampling schemes in comparison, namely mean over folds, mean over runs and sorted folds (see [2] for details). All of these show the same trends as shown the figures here.

### 4.4. Summary

In all experiments, worst case replicability remains the same independent of data set size. Further, there is an area where the difference between algorithm is either large enough or small enough to guarantee high replicability. Replicability can be increased by taking more samples and increasing test set size. The former has computational costs, the latter may result in an incorrect decision because the decision is made based on algorithms trained on considerably smaller data sets than the algorithm is going to be used with. So, it is possible that the learning curves cross after the point where the test is performed (anecdotic evidence suggests this has happened in practice). Increasing the number of samples remains the only feasible option in such instance.

## 5. Conclusions

We showed in this paper that lack of replicability of certain common machine learning experiments is not only an issue for small data sets, but is a phenomenon that is present with large data sets as well. We gained insight in the relation between power and replicability and showed how worst case replicability coincided with the point where the power of a test is around 50%. We showed how replicability decreases with increasing difference between two algorithms from Type I replicability to the worst case replicability. Then, replicability rises again to 100% where power reached 100%.

Experimental design is one of the main factors that impact replicability. Repeated cross validation performs well and sorted repeated cross validation slightly better. However, such experiments may require a lot of computational effort. One could think of an adaptive scheme where first a simple experiment is performed at relatively low computational cost and depending on the outcome more samples may be obtained depending on the outcome of the experiment. Indeed, some preliminary exploration of this idea with the sorted cross validation sampling scheme has showed some promissing results

In this paper, we studied the case of two algorithms and one data set. Since in practice many more algorithms are available to select from, future efforts should be directed at the more complex situation where more than two algorithms are considered. For machine learning researchers, usually more than one data set is considered, which makes it important to develop experiments that can take multiple algorithms and multiple data sets in account. One has to be aware that appart from Type I error, Type II error and replicability issues multiple comparison problems [6] will appear as well.

## Appendix: Analysis of worst case replicability of McNemar's test

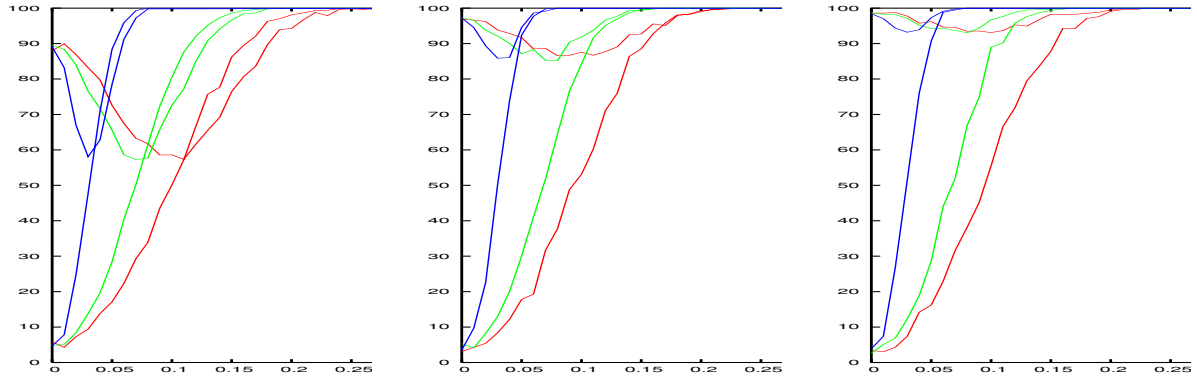
Suppose that algorithm B outperforms A on a domain from which data set $D$ is drawn such that the probability of a $'+'$ is $q$. Let $q$ be such that the expected value of $Z$ equals the threshhold, $E\{Z\} = Z_{\alpha/2}$. Then, $(n.q - n.\frac{1}{2})/\sqrt{n.\frac{1}{4}} = Z_{\alpha/2}$, or $q = \frac{1}{2} + \frac{1}{2\sqrt{n}} Z_{\alpha/2}$.

Note that since $Z$ follows approximately a normal distribution with mean $Z_{\alpha/2}$, 50% of the time the null hypothesis will be rejected, and 50% of the time it won't. So, replicability is $\frac{1}{2}X + \frac{1}{2}Y$ where $X$ (and $Y$) is the probability that the second experiment rejects (accepts) given the first experiment rejects (accepts). As before, let us assume that the classifiers produce the same outcome on the same instances and that there are no draws.

If in the first experiment the null hypothesis is accepted, the expected number of '+'s in $D_{t,1}$ is $E(x|T > Z_{\alpha/2}) = \sum_{n \cdot q}^{n} x \cdot P(x|T > Z_{\alpha/2})dx \approx \int_{Z_{\alpha/2}}^{\infty} x \cdot P(x|T > Z_{\alpha/2})dx = \int_{Z_{\alpha/2}}^{\infty} x \cdot P(x, T > Z_{\alpha/2})/P(T > Z_{\alpha/2})dx = \int_{Z_{\alpha/2}}^{\infty} x \cdot N(n \cdot q, \sigma)/(1/2)dx = \int_{0}^{\infty} n \cdot q \cdot N(0, \sigma)/(1/2)dx + \int_{0}^{\infty} x \cdot N(0, \sigma)/(1/2)dx = q \cdot n + \int_{0}^{\infty} x \cdot \frac{1}{\sqrt{2\pi\sigma^2}} e^{-x^2/2\sigma^2} dx/(1/2) = q \cdot n - \frac{\sigma}{\sqrt{2\pi}} e^{-x^2/2\sigma^2}|_{0}^{\infty}/(1/2) = q \cdot n + \frac{\sigma}{\sqrt{2\pi}}/(1/2) = q \cdot n + \frac{2\sigma}{\sqrt{2\pi}}$ with $\sigma^2 = n \cdot q(1-q)$ gives $E(x|T > Z_{\alpha/2}) = q \cdot n + \frac{2\sqrt{n \cdot q(1-q)}}{\sqrt{2\pi}}$.

If in the first experiment the null hypothesis is accepted, the expected number of '+'s in $D_{t,1}$ is $E(x|T > Z_{\alpha/2}) = \sum_{n \cdot q}^{n} x \cdot P(x|T > Z_{\alpha/2})dx \approx \int_{Z_{\alpha/2}}^{\infty} x \cdot P(x|T > Z_{\alpha/2})dx = \int_{Z_{\alpha/2}}^{\infty} x \cdot P(x, T >$

*Figure 4.* Repeated sorted 10 fold cross validation (axis same as Figure 1 with rescaled axis). 1x left, 10x middle, 50x right graph.



$Z_{\alpha/2})/P(T > Z_{\alpha/2})dx = \int_{Z_{\alpha/2}}^{\infty} x \cdot N(n \cdot q, \sigma)/(1/2)dx = \int_0^{\infty} n \cdot q \cdot N(0, \sigma)/(1/2)dx + \int_0^{\infty} x \cdot N(0, \sigma)/(1/2)dx = q \cdot n + \int_0^{\infty} x \cdot \frac{1}{\sqrt{2\pi\sigma^2}} e^{-x^2/2\sigma^2} dx/(1/2) = q \cdot n - \frac{\sigma}{\sqrt{2\pi}} e^{-x^2/2\sigma^2} |_0^{\infty}/(1/2) = q \cdot n + \frac{\sigma}{\sqrt{2\pi}}/(1/2) = q \cdot n + \frac{2\sigma}{\sqrt{2\pi}}$ with $\sigma^2 = n \cdot q(1-q)$ gives $E(x|T > Z_{\alpha/2}) = q \cdot n + \frac{2\sqrt{n \cdot q(1-q)}}{\sqrt{2\pi}}$.

So, in $D_{t,1}$ the probability of drawing a '+' is $\frac{q \cdot n}{n} + \frac{2\sqrt{n \cdot q(1-q)}}{n\sqrt{2\pi}}$) The probability of drawing a '+'s in $D_{t,2}$ is $t(\frac{qn}{n} + \frac{2\sqrt{n \cdot q(1-q)}}{n\sqrt{2\pi}})$ for $D_{t,2} \cap D_{t,1}$, and $(1-t) \cdot q$ for $D_{t,2} \backslash D_{t,1}$. Together, this is $p' = (1-t)q + t(q + \frac{2\sqrt{q(1-q)}}{\sqrt{2\pi n}}) = q + t\frac{2\sqrt{q(1-q)}}{\sqrt{2\pi n}}$ So, let $T'$ be the statistic for the second experiment, then $P(T' > Z_{\alpha/2}) = P(\frac{np' - n/2}{\sqrt{\frac{1}{2}\frac{1}{2}n}} > Z_{\alpha/2}) \approx \int_{n \cdot q}^{\infty} N(n \cdot p', \frac{1}{2}\sqrt{n})dx = \int_{n \cdot q}^{n \cdot p'} N(n \cdot p', \frac{1}{2}\sqrt{n})dx + \int_{n \cdot p'}^{\infty} N(n \cdot p', \frac{1}{2}\sqrt{n})dx = \int_0^{n \cdot (p'-q)} N(0, \frac{1}{2}\sqrt{n})dx + \frac{1}{2} = \int_0^{n \cdot t\frac{2\sqrt{q(1-q)}}{\sqrt{2\pi n}}} N(0, \frac{1}{2}\sqrt{n})dx + \frac{1}{2}$. Now, $n \cdot t\frac{2\sqrt{q(1-q)}}{\sqrt{2\pi n}} = t\frac{\sqrt{2n \cdot q(1-q)}}{\sqrt{\pi}} = t\frac{\sqrt{2n \cdot (\frac{1}{2} + \frac{1}{2\sqrt{n}} Z_{\alpha/2})(\frac{1}{2} - \frac{1}{2\sqrt{n}} Z_{\alpha/2})}}{\sqrt{\pi}} = t\frac{\sqrt{2n \cdot (\frac{1}{4} - \frac{1}{4n} Z_{\alpha/2}^2)}}{\sqrt{\pi}} \approx t\frac{\sqrt{n \cdot \frac{1}{2}}}{\sqrt{\pi}} = t\frac{\sqrt{n}}{\sqrt{2\pi}}$ Therefore, $P(T' > Z_{\alpha/2}) \approx \int_0^{t\frac{\sqrt{n}}{\sqrt{2\pi}}} N(0, \frac{1}{2}\sqrt{n})dx + \frac{1}{2} = \int_0^{t\frac{1}{\sqrt{2\pi}}} N(0, \frac{1}{2})dx + \frac{1}{2}$, which surprisingly is an expression that does not contain the data set size $n$. The situation where the first experiment rejected the null hypothesis follows a similar line of reasoning, which shows that the probability that the second experiment rejects $H_0$ given that the first experiment does is an expression that is not dependent on the size of the sample $n$.

## References


[1] R.R. Bouckaert. Choosing between two learning algorithms based on calibrated tests. Proceedings of the 20t Internation Conference on Machine Learning, Morgan Kaufmann, 2003.

[2] R.R. Bouckaert. Estimating Replicability of Classifier Learning Experiments. Proceedings of the 21t Internation Conference on Machine Learning, ACM, 2004.

[3] R.R. Bouckaert and E. Frank. Evaluating the Replicability of Significance Tests for Comparing Learning Algorithms. Proc 8th Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer-Verlag, 2004.

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

[5] IFA Services: Statistics, McNemar's Test. http://www.fon.hum.uva.nl/Service/Statistics/McNemars_test.html. Last vistited 31 March 2005.

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

[7] C. Nadeau and Yoshua Bengio. Inference for the generalization error. Advances in Neural Information Processing Systems 12, pp. 307–313, 2000.

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

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