S. Krajči (ed.): ITAT 2018 Proceedings, pp. 59–63
CEUR Workshop Proceedings Vol. 2203, ISSN 1613-0073, © 2018 Jan Motl and Pavel Kordík

Violation of Independence of Irrelevant Alternatives in Friedman's test

Jan Motl, Pavel Kordík

Czech Technical University in Prague,
Thákurova 9, 160 00 Praha 6, Czech Republic,
jan.motl@fit.cvut.cz, pavel.kordik@fit.cvut.cz

Abstract: One of the most common methods for classifier comparison is Friedman's test. However, Friedman's test has a known flaw — ranking of classifiers A and B does not depend only on the properties of classifiers A and B, but also on the properties of all other evaluated classifiers. We illustrate the issue on a question: "What is better, bagging or boosting?". With Friedman's test, the answer depends on the presence/absence of irrelevant classifiers in the experiment. Based on the application of Friedman's test on an experiment with 179 classifiers and 121 datasets we conclude that it is very easy to game the ranking of two insignificantly different classifiers. But once the difference becomes significant, it is unlikely that by removing irrelevant classifiers we obtain a significantly different classifiers but with reversed conclusion.

1 Introduction

Friedman's test is the recommended way how to compare algorithms in machine learning (ML) [9]. It is a nonparametric test, which calculates scores not on the raw performance measures (e.g. classification accuracy or correlation coefficient) but on the ranks calculated from the raw measures. Nonparametric tests are favored over parametric tests in ML, because the standard formulation of the central limit theorem (CLT) does not apply on bounded measures [8], many measures used in ML are bounded, and commonly used parametric tests rely on CLT. Nevertheless, even nonparametric tests, which are based on ranking, have flaws as demonstrated by Arrow's impossibility theorem [1]. In this article, we discuss one such flaw of Friedman's test: violation of *Independence of Irrelevant Alternatives* (IIA).

2 Problem Description

Friedman's test Friedman's test, if applied on algorithm ranking, is defined as:

> Given data $\{x_{ij}\}_{n\times k}$, that is, a matrix with n rows (the *datasets*), k columns (the *algorithms*) and a single performance observation at the intersection of each dataset and algorithm, calculate the ranks *within* each dataset. If there are tied values, assign to each tied value the average of the ranks that would have been assigned without ties. Replace the data with a new matrix $\{r_{ij}\}_{n\times k}$ where the entry r_{ij} is the rank of x_{ij} within dataset i. Calculate then rank sums of algorithms as: $r_j = \sum_{i=1}^{n} r_{ij}$.

We can rank the algorithms based on *rank sums* r_j [10]. Friedman's test then continues with the evaluation of the null hypothesis that there are no differences between the classifiers. Since this article is not about hypothesis testing but rather about algorithm ranking based on r_j, we reference a keen reader to read [9] for a detailed description of Friedman's test hypothesis testing.

IIA Independence of Irrelevant Alternatives [14] condition is defined as:

> If algorithm A is preferred to algorithm B out of the choice set $\{A,B\}$, introducing a third option X, expanding the choice set to $\{A,B,X\}$, must not make B preferable to A.

In other words, preferences for algorithm A or algorithm B, as determined by rank sums r_j, should not be changed by the inclusion of algorithm X, i.e., X is irrelevant to the choice between A and B.

Illustration Boosting tends to outperform base algorithms (e.g. decision trees) by a large margin. But sometimes, boosting fails [2, Chapter 8.2] while bagging reliably outperforms base algorithms on all datasets, even if only by a small margin [2, Chapter 7.1]. This is illustrated in the left part of Figure 1. If we compare boosting only to bagging, then by the rank sums r_j, boosting wins, because boosting is better than bagging on the majority of datasets. However, if we add irrelevant algorithms that are always worse than bagging, the conclusion may change. Bagging will be always the first or the second. But boosting will be either the first or (in the provided illustration) the last. And a few extreme values in the rank sum r_j can result into the change of the conclusion.

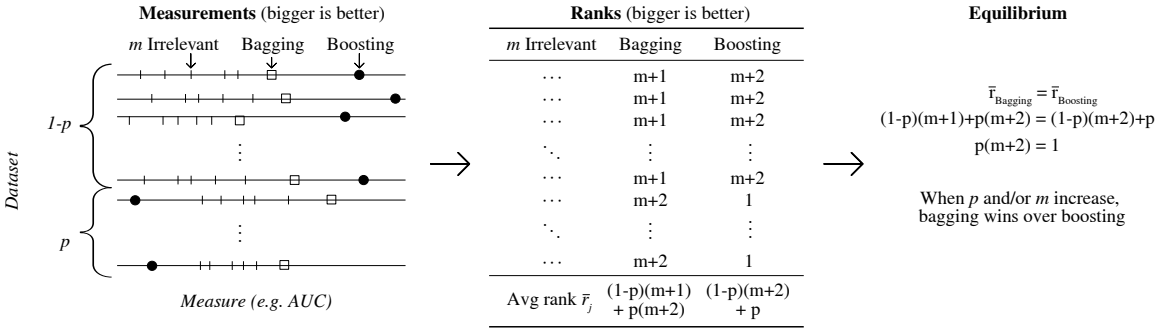

Figure 1: What is better, bagging or boosting? Boosting is better than bagging by a large margin on majority of datasets (e.g. in 95%). But with a constant probability p, boosting fails critically. Interestingly, the decision which of the algorithms is better does not depend only on p but also on the count of irrelevant algorithms (m).

3 Impact

3.1 Bias of Authors

Hypothesis Authors of highly cited papers, knowingly or not, frame their algorithms in the best possible light. Based on the equilibrium equation in Figure 1, we would expect that proponents of boosting algorithms use *fewer algorithms* than proponents of bagging in the design of experiments (DOE).

Evaluation Breiman, the author of bagging [6], compared bagging against 22 other algorithms while Freund and Schapire, authors of AdaBoost [11], compared their boosting algorithm against only 2 other algorithms. Our expectations were fulfilled.

Threats to validity due to omitted variables Since both articles were published in the same year, the availability of algorithms for comparison should be comparable and cannot be used to explain the observed differences in the DOE.

Conclusion Since this is just a single observation, which can be just by chance, following section analyses the impact of DOE numerically.

3.2 Effect on Large Studies

A nice comparison of 179 classifiers on 121 datasets is provided by Fernández-Delgado et al. [10]. The authors follow Demšar's [9] recommendation to use Friedman's test to rank the algorithms.

If we directly compare just two classifiers, *AdaBoostM1_J48_weka* and *Bagging_J48_weka*, and calculate rank sums r_j, then we get that *boosting beats bagging* 64 to 47. But if we calculate rank sums over all algorithms, then we get that *bagging beats boosting* 13136 to 12898. A completely reversed conclusion!

How frequently does the ordering flip? If we perform above analysis over all unique pairs of classifiers ($\frac{1}{2}179(179-1) = 15753$), then we get that the ordering flips in 5% of cases (831/15753).

Are the changes in the ranks significant? We repeated the experiment once again, but considered only classifier pairs that are based on Friedman's test:

1. *pairwise* significantly different
2. and significantly different in the presence of *all the remaining classifiers*.

If we do not apply multiple testing correction, the classifiers flip the order in mere 0.05% cases (8/15753) at a shared significance level $\alpha = 0.05$. Once we add multiple testing correction, *the count of significant flips drops to zero*. In our case, the exact type of multiple testing correction does not make a difference. Bonferroni, Nemeneyi, Finner & Li and Bergmann & Hommel [17] corrections all give the same result as the lowest p-value before the correction in that 8 cases is 0.023, which is already extremely close to the significance level of 0.05.

4 Related Work and Discussion

We are not the first one to notice that Friedman's test does not fulfill the IIA property. The oldest mention of the issue in the literature, that we were able to track down, dates back to 1966 [4]. Following paragraphs discuss possible solutions to the issue.

4.1 Pairwise Ranking

Benevoli et al. [4] recommend replacing Friedman's test with pairwise *Wilcoxon signed-rank test* followed with *multiple testing correction*. The application of a pairwise-test solves the issue with IIA if we want to show how well "a new algorithm A fares in comparison to a set of old algorithms $\mathbb{B}$" because each pairwise comparison between A and some other algorithm $B \in \mathbb{B}$ is by definition independent on $C \in \mathbb{B}, C \neq B$. However, if we aim to "rank the current algorithms together", pairwise tests may not deliver a *total ordering* while a method comparing all algorithms at once can, as illustrated in Figure 2.

	A	B	C
α	1	2	3
β	1	2	3
γ	3	1	2
δ	3	1	2
ε	2	3	1
Rank sum	10	9	11

Figure 2: A minimal example of 3 algorithms and 5 datasets where rank sums deliver total ordering while pairwise methods end up with a cycle: $A \prec B, B \prec C, C \prec A$.

One important implementation detail, which is not discussed by Benevoli et al., is that Wilcoxon signed-rank test does not by default take into account ties (while Friedmans's test does). Ties may appear in an experimental study as a result of rounding or as a consequence of using a dataset with a small sample size. And if the percentage of ties is high, the negligence of the ties can result in misleading results, as discussed by Pratt [13]. In R, we can use `wilcoxsign_test` function from `coin` package, which implements Pratt's tie treatment.

4.2 Vote Theory

During the French revolution in the 18th century, the need to come with a fair vote method arose. Two competitors, Condorcet and Borda[1], came with two different methods. And they did not manage to agree, which of the methods is better. This disagreement spurred an interest into vote theory. One of the possible alternatives to Friedman test that vote theory offers is *Ranked pairs* method [16]:

1. Get the count of wins for each pair of algorithms.
2. Sort the pairs by the difference of the win counts in the pair.
3. "Lock in" the pairs beginning with the strongest difference of the win counts.

While Ranked pairs method also fails IIA criterium, it at least fulfills a weaker criterium called *Local Independence from Irrelevant Alternatives* (LIIA). LIIA requires that the following conditions hold:

> If the best algorithm is removed, the order of the remaining algorithms must not change. If the worst algorithm is removed, the order of the remaining algorithms must not change.

An example where Friedman's test fails LIIA criterium is given in Figure 3 [14].

	A	B	C
α	1	3	2
β	1	3	2
γ	2	1	3
δ	2	1	3
ε	2	1	3
Rank sum	8	9	13

$\longrightarrow$

	A	B
α	1	2
β	1	2
γ	2	1
δ	2	1
ε	2	1
Rank sum	8	7

Figure 3: A minimal example of 3 algorithms and 5 datasets demonstrating that Friedman's test does not fulfill LIIA criterium — when we remove the best algorithm C, algorithm A becomes better than algorithm B.

Friedman's test also violates *Independence of Clones criterion* [15]:

> The winner must not change due to the addition of a non-winning candidate who is similar to a candidate already present.

An example, when can this can become a problem is given by [4]:

> Assume that a researcher presents a new algorithm A_0 and some of its weaker variations $A_1, A_2, \ldots, A_k$ and compares the new algorithms with an existing algorithm B. When B is better, the rank is $B \succ A_0 \succ \ldots \succ$

[1] Borda count is equivalent to rank sums r_j. Hence, whatever vote theory has to say about Borda count also applies to Friedman's test.

> A_k. When A_0 is better, the rank is $A_0 \succ A_1 \succ \ldots \succ A_k \succ B$. Therefore, the presence of $A_1, A_2, \ldots, A_k$ artificially increases the difference between A_0 and B.

But Ranked pairs method fulfills this criterion.

Finally, Friedman's test violates *Majority criterion* [3]:

> If one algorithm is the best on more than 50% of datasets, then that algorithm must win.

We have already observed a violation of this criterion in the though example with bagging versus boosting — even thought boosting was the best algorithm on the majority of the datasets, this fact alone did not guarantee boosting's victory. The violation of Majority criterion also implies a violation of *Condorcet criterion* [5]:

> If there is an algorithm which wins pairwise to each other algorithm, then that algorithm must win.

Which is, nevertheless, once again fulfilled with Ranked pairs method. However, just like in machine learning we have no free lunch theorem, Arrow's impossibility theorem [1] states that there is not a ranked vote method without a flaw. The flaw of all ranked vote methods, but dictatorship[2], that fulfill Condorcet criterium is that they fail *Consistency criterium* [18, Theorem 2]:

> If based on a set of datasets $\mathbb{A}$ an algorithm A is the best. And based on another set of datasets $\mathbb{B}$ an algorithm A is, again, the best. Then based on $\mathbb{A} \cup \mathbb{B}$ the algorithm A must be the best.

Notably, Friedman's test fulfills this criterium while Ranked pairs method fails this criterium. For convenience, a summary table with the list of discussed criteria is given in Table 1.

Table 1: Method compliance with criteria.

Criterium	Friedman's	Ranked pairs
IIA	✗	✗
LIIA	✗	✓
Independence of Clones	✗	✓
Majority	✗	✓
Condorcet	✗	✓
Consistency	✓	✗

4.3 Bayesian

Another option is to go parametric and replace the common assumption of normal distributions with Beta distributions, which are more appropriate for modeling of upper and bottom bounded measures [7, 12].

[2] In a dictatorship, the quality of an algorithm is determined on a single dataset

4.4 Continue Using Friedman's Test

Finally, there is the option of continuing using Friedman's test as before. In the numerical analysis in Section 3.2, we did not observe any *significant flip* in the ordering of the algorithms.

5 Conclusion

Contrary to our expectations, based on analysis of 179 classifiers on 121 datasets, Friedman's test appears to be fairly resistant to manipulation, where we add or remove irrelevant classifiers from the analysis. Therefore, we *cannot* recommend avoiding Friedman's test only because it violates Independence of Irrelevant Alternatives (IIA) criterium.

6 Acknowledgment

I would like to thank Adéla Chodounská for her help. We furthermore thank the anonymous reviewers, their comments helped to improve this paper. The reported research has been supported by the Grant Agency of the Czech Technical University in Prague (SGS17/210/OHK3/3T/18) and the Czech Science Foundation (GAČR 18-18080S).

References

[1] Kenneth J. Arrow. A Difficulty in the Concept of Social Welfare. *J. Polit. Econ.*, 58(4):328–346, 1950.

[2] Eric Bauer and Ron Kohavi. An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants. *Mach. Learn.*, 36(1-2):105–139, 1999.

[3] Harry Beatty. Voting Rules and Coordination Problems. In *Methodol. Unity Sci.*, pages 155–189. Springer Netherlands, Dordrecht, 1973.

[4] Alessio Benavoli, Giorgio Corani, and Francesca Mangili. Should we really use post-hoc tests based on mean-ranks? *J. Mach. Learn. Res.*, 17:1–10, 2016.

[5] Duncan Black. On the Rationale of Group Decision-making. *J. Polit. Econ.*, 56(1):23–34, feb 1948.

[6] Leo Breiman. Bagging predictors. *Mach. Learn.*, 24(2):123–140, 1996.

[7] Giorgio Corani, Alessio Benavoli, Janez Demšar, Francesca Mangili, and Marco Zaffalon. Statistical comparison of classifiers through Bayesian hierarchical modelling. *Mach. Learn.*, 106(11):1817–1837, 2017.

[8] Nicholas J. Cox. Correlation with confidence, or Fisher's z revisited. *Stata J.*, 8(3):413–439, 2008.

[9] Janez Demšar. Statistical Comparisons of Classifiers over Multiple Data Sets. *J. Mach. Learn. Res.*, 7:1–30, 2006.

[10] Manuel Fernández-Delgado, Eva Cernadas, Senén Barro, and Dinani Amorim. Do we Need Hundreds of Classifiers to Solve Real World Classification Problems? *J. Mach. Learn. Res.*, 15:3133–3181, 2014.

[11] Yoav Freund and Robert E. Schapire. Experiments with a New Boosting Algorithm. *Int. Conf. Mach. Learn.*, pages 148–156, 1996.

[12] John K. Kruschke. Bayesian data analysis. *Wiley Interdiscip. Rev. Cogn. Sci.*, 1(5):658–676, 2010.

[13] John W. Pratt. Remarks on Zeros and Ties in the Wilcoxon Signed Rank Procedures. *Am. Stat. Assoc.*, 54(287):655–667, 1959.

[14] Paramesh Ray. Independence of Irrelevant Alternatives. *Econometrica*, 41(5):987, sep 1973.

[15] Markus Schulze. A new monotonic and clone-independent single-winner election method. *Voting Matters*, (17):9–19, 2003.

[16] Thorwald N. Tideman. Independence of clones as a criterion for voting rules. *Soc. Choice Welfare*, 4(3):185–206, 1987.

[17] Bogdan Trawiński, Magdalena Smetek, Zbigniew Telec, and Tadeusz Lasota. Nonparametric statistical analysis for multiple comparison of machine learning regression algorithms. *Int. J. Appl. Math. Comput. Sci.*, 22(4):867–881, jan 2012.

[18] H. P. Young and A. Levenglick. A Consistent Extension of Condorcet's Election Principle. *SIAM J. Appl. Math.*, 35(2):285–300, sep 1978.

Appendix: Arrow's theorem

Arrow's theorem, once applied on algorithms and datasets, states that once we have 3 or more algorithms, a ranking method cannot fulfill all following "reasonable" properties:

Unrestricted domain The ranking method should be complete in that given a choice between algorithms A and B it should say whether A is preferred to B, or B is preferred to A or that there is indifference between A and B.

Transitivity The preferences should be transitive; i.e., if A is preferred to B and B is preferred to C then A is also preferred to C.

Non-dictatorship The outcome should not depend only upon a single dataset.

Weak Pareto Efficiency If algorithm A is better than algorithm B on all datasets, then A must rank above B.

Independence of Irrelevant Alternatives (IIA) If algorithm A is preferred to algorithm B out of the choice set $\{A,B\}$, introducing a third option X, expanding the choice set to $\{A,B,X\}$, must not make B preferable to A.

Appendix: Criteria

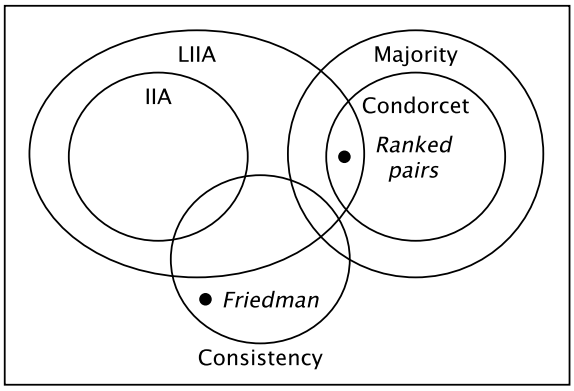

Figure 4: Venn diagram with the dependencies between the discussed vote theory criteria. IIA and consistency are inconceivable with the Condorcet criterion. Independence of clones is independent of other criteria and is not depicted. The diagram holds when following conditions from Arrow's theorem are satisfied: unrestricted domain, transitivity, non-dictatorship, 3 or more competing algorithms.