

# VARIANTS OF THE BORDA COUNT METHOD FOR COMBINING RANKED CLASSIFIER HYPOTHESES

MERIJN VAN ERP AND LAMBERT SCHOMAKER

*NICI, P.O.Box 9104, 6500 HE Nijmegen, the Netherlands*  
*{M.vanErp, schomaker}@nici.kun.nl*

The Borda count is a simple yet effective method of combining rankings. In pattern recognition, classifiers are often able to return a ranked set of results. Several experiments have been conducted to test the ability of the Borda count and two variant methods to combine these ranked classifier results. By using artificial data, domain-specific results were avoided. The results show the strength of the Borda count when many errors occur in the results, but also show its weakness in case of a limited number of large ranking errors.

## 1 Introduction

In all fields of pattern recognition there exist multiple, different techniques to classify instances of patterns, each approach being characterized by its own virtues and shortcomings. The idea of combining the output of multiple classifiers has been studied for several years<sup>1,2,3,4</sup> but it is still difficult to choose a suitable combination algorithm. Using a combination of classifiers enables one to use all available knowledge and the extra computing time becomes less of a problem with the current developments in computer processing power.

Instead of defining the integration of classifier opinions as a meta-classification problem, we will focus on less cumbersome techniques. This avoids the undesirable consequences of meta-classification<sup>5</sup>, i.e., (1) an extra, large amount of training data is needed and (2) for every classifier that is added, the complete meta-classifier needs to be trained again.

The most straightforward form of opinion integration is to let the classifiers cast a vote by forwarding the class they prefer best. The class with the most votes wins. This is called plurality voting and while it is simple and quite effective, it lacks depth. With depth we mean that classifiers often have a ranking of classes to indicate which are more likely candidates than others. Plurality voting only uses the absolute top of those rankings. In this article we will discuss a method for combining the rankings of different classifiers, the Borda count.

The Borda count is an easy, intuitively appealing, and powerful method of combining different rankings. Moreover, it has some variants that may perform better on specific classification problems (see section 2). However, the theoretical foundation of the approach is less well developed than in the

case of plurality voting. Experiments have been conducted on the original Borda count and two of its variants, to measure the ability of the methods to combine ranked classifier outputs. In order to produce unbiased results, no actual classifier data from a specific pattern recognition domain was used. As a first step, artificial data was generated, simulating the results of classifiers. In doing this, the following assumptions were made:

1. Classifiers are able to provide a ranked result for all classes.
2. The combined ranking of the Borda count is assumed to be an estimate of an existing optimal ranking of all classes.

Assumption 1 excludes rule-based or all-or-nothing classifiers. In the experiments discussed here it is assumed that the ranked result contains all possible classes. Although we intend to study partial rankings in further experiments, this assumption restricts the generalization of the results. Assumption 2 can be explained with an example from handwriting recognition. If human readers are presented with a handwritten word and a list of word candidates, there should be a ranking of candidate words that would be chosen most often. This assumption may not hold true for all classification areas. In the next section, the Borda count and its variants will be fully explained. Kendall's Concordance, a statistic for measuring agreement between rankings, will be described in section 3. In section 4, the experiments will be introduced and in section 5 the results will be discussed.

## 2 Standard Borda count and two variants

The Borda count is originally a voting method in which each voter gives a complete ranking of all possible alternatives <sup>1,6</sup>. The highest ranked alternative (in for example an n-way vote) gets n votes and each subsequent alternative gets one vote less (so the number two gets n-1 votes and the number three n-2 and so on). Then, for each alternative, all the votes are added up and the alternative with the highest number of votes wins the election. Ties in the accumulated votes are not resolved in the original Borda count<sup>a</sup>. This method, introduced in 1770 by Jean-Charles de Borda <sup>7</sup>, is easily adapted to classification problems. Each classifier is a voter and the classes are the candidates. The method has depth as it uses the entire ranking information to come to a decision, not just the best guess of each classifier. It also returns a complete ranking of the possible classes instead of its best guess, offering more flexibility for further uses. Consider, e.g., classifications with a large number of possible classes, where the top-ranked candidate may be wrong. If

---

<sup>a</sup>In the experiments a fair random choice will be made between all tied classes.

the application context allows it (i.e. a collection of classes can be the answer instead of just one class), one could choose to accept a group of the best possible classifications instead of just the top guess, increasing the probability of including the correct class. The ranked result of the Borda count gives suggestions concerning the alternatives just below the top rank.

What the Borda count lacks, is a way to differentiate between several classifiers based on their general performance or expertise. In fact, the assumption is that the top-ranked candidates of all classifiers are of comparable quality, thus all classifiers (voters) are treated equal, while this may not be desirable. A solution for this problem is given in <sup>1</sup>. Another way of calculating the Borda count is *averaging* the rank given by each voter to a class, instead of adding up the votes. The new ranking is then calculated by ranking the averaged votes, highest one on top. Note that effectively, this does not change the results of the combination process, however, the concept of an average rank has interesting implications: Assuming a probability distribution of rank numbers for a given class, there exist other measures than the mean to describe central values of that distribution. An example is the median, i.e. the rank value that splits the number of given rank numbers in half. The Borda count using the median (Borda variant 1) instead of the mean, will be less susceptible to extreme voting behavior of a few classifiers with respect to some classes.

The second Borda variant is Nanson's Borda-elimination procedure<sup>8,7</sup>. This is a multi-step procedure that repeatedly performs a Borda count and in each iteration deletes the lowest Borda ranked alternative from each classifiers ranking. This allows the top-ranked classes to recover from extreme low votes (see section 5.1).

### 3 Kendall's concordance

The use of the Borda count to combine classifier results has as main goal the improvement of the correctness of the total classification. That is, the correct class should be as high in the ranking as possible, preferably first. However, the Borda count uses rankings and delivers a ranking as a result. We find it interesting to know what happens to the rest of the ranking, apart from the top. It might, for example, indicate that the Borda count or one of its variants is only reliable in the top ranks. In order to recreate the "optimal ranking" mentioned in the second assumption (section 1), a reliable performance on the entire ranking is needed.

To measure this we will also test the Borda count's ability to reconstruct rankings. We do this in our experiments by starting with a ranking, which

is considered the true ranking  $R_0$  (for more information see section 4). This ranking is compared to the ranking returned by the Borda count. For this comparison we use Kendall's coefficient of concordance <sup>9</sup>. This is a measure that indicates the agreement between several rankings. Normally, a high degree of agreement would not automatically imply correctness, as the rankings could agree on a faulty answer. However,  $R_0$  is defined as correct, so the concordance measure can be used to indicate correctness of the Borda count ranking. In the experiments we want to compare the resulting coefficient to the amount of noise. The amount of noise can be measured by calculating the agreement between all classifier rankings using Kendall's concordance. The Kendall concordance measure is calculated as follows:

1. Let  $C$  be the number of classifiers and  $n$  be the number of classes.
2. Calculate for each class the total number of votes  $V_i$  ( $n - rank + 1$  for each classifier).
3. Take the mean of the  $V_i$  and calculate:

$$s = \sum_i (V_i - V_{mean})^2 \quad (1)$$

4. Now calculate Kendall's concordance

$$W = \frac{s}{\frac{1}{12}n^2(C^3 - C)} \quad (2)$$

## 4 Method

In the experiments to measure the abilities of the Borda count and its two variants, the classes and classifiers are all artificially generated. The idea is to establish an original ranking  $R_0$  and then simulate the classifiers by introducing errors into that ranking. Each classifier  $i$  has an independent set of errors  $P_i$ , which simulates the classification errors the classifier makes. Subsequently, the Borda count is used to combine the results  $R_i$  and its ranking  $R_b$  is compared to  $R_0$ . Of special interest is the place of the correct class in the Borda count ranking.

The method of introducing the errors  $P_i$  is to take the original ranking  $R_0$  and to permute it enough to simulate the result of a classifier with some obvious mistakes, but not so severely that the result becomes a random ranking, so result  $R_i = P_i(R_0)$ . The following two methods were used:

1. Permute the ranking by randomly swapping the ranks of two classes. This is repeated a random number of times (up to a maximum) for each classifier to simulate differences in performance. These errors can be said to simulate the decisions some classifiers need to make between two classes or groups of classes based on a thresholded value. If the threshold is trained slightly wrong

or there is some noise on the value, the wrong decision may be taken with the kind of radical results that this *swap error* accomplishes.

2. Use noise on artificial “confidence” values. When a classifier produces a ranking, it is based on some sort of preference value. This value can for example be a distance value, a confidence measure or the likelihood of being the intended class. We will call all these values confidence values for convenience. The original ranking in our experiments is established by generating confidences for each class and ranking them accordingly. Small variations in confidence values, e.g. the noise from input devices or a lack of knowledge of a classifier for a particular input, are simulated by adding a small random value to all confidence values, causing a reranking of the classes. This is done for each classifier. The result is less radical than the swap errors.

Each experiment performs the following set of operations:

1. Assign each class  $i$  a number  $C_i$  generated by a Gaussian random source with a standard deviation of 1.
2. Rank the classes according to the  $C_i$  assigned. This forms the original ranking  $R_0$ .
3. Copy  $R_0$  for each classifier and apply noise to each classifier's ranking  $R_i$  according to either the swap or the confidence noise permutation.
4. Calculate one of the three Borda count variants (original and median Borda and Nanson's procedure), using the classifiers' rankings. Do this for all numbers of independent classifiers from 2 to 25.
5. Calculate Kendall's concordance  $W$  between the results and the original.
6. Let  $N$  be the number of possible classes. For all  $n < N$ , check if the correct class (the top class of the original ranking) is one of the top  $n$  ranked classes of the results. The answer is either 0 or 1 ('no' and 'yes') for each value of  $n$ . These are called the  $top_n$  values.

This is repeated 10000 times for each experiment, after which the average is taken of all the Kendall's concordances and of all  $top_n$  values for each different number of classifiers. These averages will be called *scores* in the sequel.

#### 4.1 Swap errors

The experiments using swap errors use two parameters to adjust the amount of errors in the artificial classifier rankings. The maximum number of swaps parameter,  $n_{swap}$ , and the maximum swap distance parameter,  $e_{swap}$ , determine the upper limits of the swap errors. When adding the noise to classifier rankings  $R_i$  (using swap errors), the following steps are taken:

1. Determine the number of swaps that are made in a ranking. This differs per classifier ranking to simulate classifiers with different performance rates. A random number between 0 and  $n_{swap}$  is generated by a uniform random generator and that number of swaps is made.

2. For each swap:
  - Choose a starting class uniformly from all classes.
  - Choose a second class by picking a distance (uniformly) from the range  $-e_{swap}$  to  $e_{swap}$  (distances outside the range of the available classes are not allowed).
  - The two classes are then swapped in rank.

Each of the three Borda count variants had two swap error experiments, one with smaller errors ( $e_{swap} = 5$ ) and one with larger errors ( $e_{swap} = 15$ ). In both cases  $n_{swap}$  was chosen to be 20. The parameters were chosen with some consideration. The large error value ( $e_{swap} = 15$ ) was chosen so that the individual classifier had a  $top_1$  score of the order of 60%, which is appropriate for large errors. The smaller error ( $e_{swap} = 5$ ) was set at a third of the large error value.

#### 4.2 Confidence errors

Experiments with both uniform and normal distributions of the noise value were conducted. As the results were not significantly different, only the uniform variant will be discussed. In the experiments the class values that were used to generate the original ranking (the  $C_i$  values in section 4) are used to create a separate ranking for each classifier. This is performed as described below:

1. A noise parameter  $e_{uni}$  is chosen between 0 and 1.
2. For each classifier a list of so-called confidence values is made from the class values. For each class  $i$  its confidence value  $v_i$  is calculated from its class value  $C_i$ :
  - a random number  $r_i$  is generated between -1 and 1 from a uniform random generator.
  - $v_i = (r_i \times e_{uni}) + C_i$ .
3. For each classifier, the classes are reranked according to their new confidence values.

For each of the three Borda count variants two experiments have been conducted. They consisted of 25 classes and had noise parameters of 0.3 and 0.8. The error values were chosen in a similar fashion as the error values in the swap experiment (see section 4.1). A third experiment had a noise parameter of 0.8, but 100 classes were used. This was performed for the original Borda count and the median variant. All parameter values were determined on the basis of pilot experiments with actual character and word classifiers.

## 5 Results

In this section the results of the experiments will be presented. The results consist of the so-called  $top_n$  and Kendall scores. The  $top_n$  scores (only  $top_1$  and  $top_3$  are shown) represent the percentage of the runs of the experiment that the correct class was one of the  $n$  best ranked classes of the resulting ranking. So, for example, a  $top_2$  score of 70% would mean that in 70% of the runs of an experiment, the correct class was on the top rank or on the second rank of the result and in 30% of the runs this was not the case. The  $top_n$  scores give an indication of the ability of the Borda count (and its variants) to reproduce the correct class out of the results of several classifiers.

The Kendall score is the average of Kendall's concordance  $W$  over all the runs of an experiment.

In order to show the differences more clearly, the results of the experiments with the highest error rate ( $e_{swap} = 15$  and  $e_{uni} = 0.8$ ) are shown. Both experiments shown had 25 possible classes and up to 25 independent classifiers. The swap error experiment  $top_1$  and  $top_3$  results are in figure 1, while those of the confidence error experiment are in figure 2. The Kendall scores are shown in figure 3 with the swap results on the left and the confidence results on the right. The straight horizontal lines in the graphs denote the average scores of the single classifiers before any Borda count method was applied. These lines show the amount of error that was present in the classifiers' ranking.

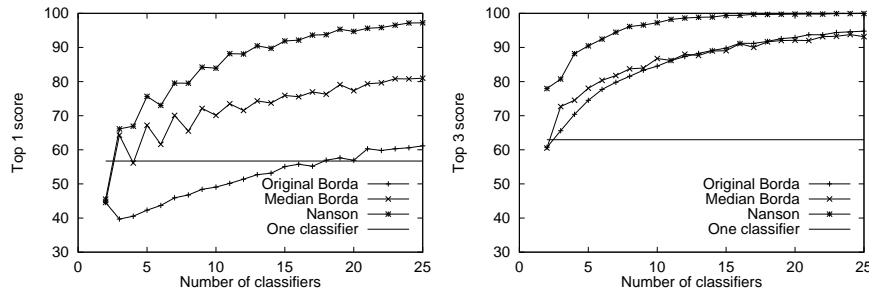


Figure 1.  $Top_1$  scores (left) and the  $top_3$  scores (right) (both as % correctly classified) of the swap error experiment with  $e_{swap} = 15$  and  $n_{swap} = 20$

The seesaw appearance of some of the graphs (most notably the scores of the median Borda count and also *not* those of the original Borda count)

are due to the difference in an even or odd number of classifiers<sup>b</sup>. This was not corrected to accentuate the difference between the swap errors and the confidence errors. In the case of the swap errors the results of the classifications with an odd number of classifiers are better, whereas the results of the confidence error experiment show the opposite. The difference is caused by the difference in performance of the mean and the median method. The even median calculations use an averaging (the two middle values) and thus they are more like the mean method than the odd cases.

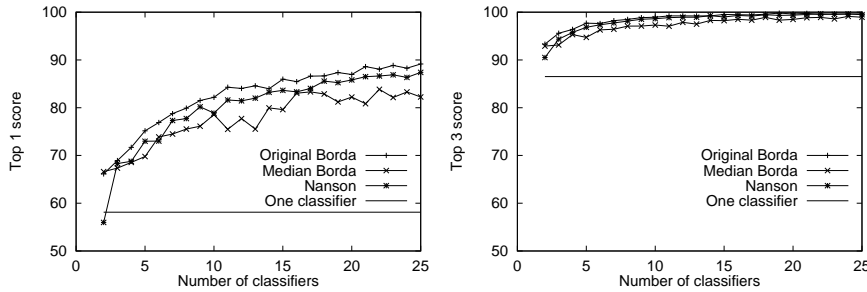


Figure 2.  $Top_1$  scores (left) and the  $top_3$  scores (right) (both as % correctly classified) of the confidence error experiment with  $e_{uni} = 0.8$

The  $top_1$  scores (figure 1) and the Kendall scores (figure 3) of the swap error experiment are low compared to the original score of one classifier, especially when the combining methods are used with a low number of classifiers and even with any number of classifiers in case of the Kendall scores.

### 5.1 Discussion on the results

When considering the swap error experiment, the three combining methods have rather different results. The original Borda count is sensitive to the sporadic, but sometimes extreme results since it takes into account the severity of the mistake. Nanson's procedure has good  $top_n$  results as it steadily throws away the worst classes and as a result the downward (in rank) swaps of the better classes gradually diminish in the rank difference they swapped over, reducing their effect on the average. Its Kendall score is still low, as the procedure is almost equal to the original Borda count in the lower part of the ranking. The median Borda deals with extreme mistakes by the nature of the

<sup>b</sup>A similar phenomenon occurs in simple plurality voting on binary decisions

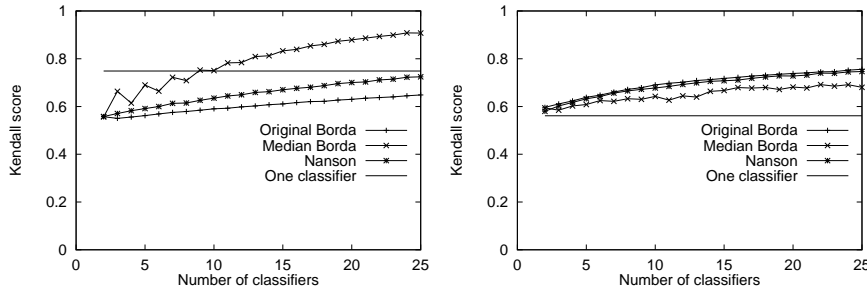


Figure 3. Kendall scores of the swap error experiment with  $e_{swap} = 15$  and  $n_{swap} = 20$  (on the left) and of the confidence error experiment with  $e_{uni} = 0.8$  (on the right)

median calculation. This does not have the same power as Nanson’s procedure in the top ranks (although it is still a lot better than the original Borda count), but it does give a much better overall result, as its Kendall scores show. Looking at the results for the confidence errors, we see the opposite happening. The confidence errors produce more, but smaller mistakes than the swap errors. The single classifier lines of the Kendall scores and the  $top_3$  scores show this effect. This kind of mistakes suits the mean better than the median, which is sensitive to a large number of mistakes, as that increases the chance that the middle value is wrong. Nanson’s procedure performs only slightly worse than the Borda count. However, it needs a new Borda count each time an alternative is thrown away, making it much slower than the simple Borda count.

## 6 Conclusion

What conclusions can be drawn based on these experiments? Completely artificial data and simulated classifiers were used (and they are simple simulations as well). What the results show is that the Borda count is (in one variation or the other) good in reconstructing a permuted rank ordering. Furthermore, it has become apparent that a sophisticated model of classifier errors is necessary. In a real classifier, both the error on confidence errors and the “swap” type error may play a role in its internal decision process. The degree to which both error types are present clearly determines the applicability of the Borda counts and its variants.

Depending on what kind of error representation comes closest to the real

errors in a certain case, we can also give the following advice on which Borda count variant to use. When swapping seems to fit best (that is, when the errors tend to be low in number but extreme in result), Nanson's procedure is preferred as  $top_n$  values are more important than a balanced performance on the whole in most cases. The only drawback is the amount of computation this procedure needs, so if the process takes too much time to compute in a certain application, the median Borda count may be better. And if the number of classifiers available is low, it may be better to refrain from using any Borda count variant as the result may very well be worse than before. For the confidence errors (many smaller errors) the Borda count should be used, as it simply performs best and is easiest to calculate as well.

The drawback of the Borda count is that it needs ranked classifier results, which not all classifiers can produce. Also, the Borda count needs the ranking to be complete, which is often not feasible due to large amounts of classes. However, in future studies we intend to test the Borda count on partial rankings (early test trials gave promising results).

## References

1. Tin Kam Ho. *A Theory of Multiple Classifier Systems And Its Application to Visual Word Recognition*. PhD thesis, Graduate School of State University of New York, Buffalo, May 1992.
2. O.G. Selfridge. Pandemonium: a paradigm for learning in mechanisation of thought processes. In *Proceedings of a Symposium Held at the National Physical Laboratory*, pages 513–526, London, November 1958. HMSO.
3. Lei Xu, Adam Krzyzak, and Ching Y. Suen. Methods of combining multiple classifiers and their applications to handwriting recognition. *IEEE Transactions on Systems, Man and Cybernetics*, 22(3):418–435, May/June 1992.
4. Robert K. Powalka, Nasser Sherkat, and Robert J. Whitrow. Multiple recognizer combination topologies. In Marvin L. Simner, editor, *Basic and Applied Issues in Handwriting and Drawing Research*, pages 128–129, 1995.
5. L.G. Vuurpijl and L.R.B. Schomaker. Multiple-agent architectures for the classification of handwritten text. In *IWFHR6, International Workshop on Frontiers of Handwriting Recognition*, pages 335–346, August 1998.
6. Jean-Charles de Borda. *Memoire sur les Elections au Scrutin*. Histoire de l'Academie Royale des Sciences, Paris, 1781.
7. Duncan Black. *The Theory of Committees and Elections*. Cambridge University Press, 1968.
8. E.J. Nanson. Methods of election. In *Transactions and Proceedings of the Royal Society of Victoria*, volume 18, pages 197–240, 1882.
9. Maurice G. Kendall. *Rank Correlation Methods*. Charles Griffin and Company, 1962.