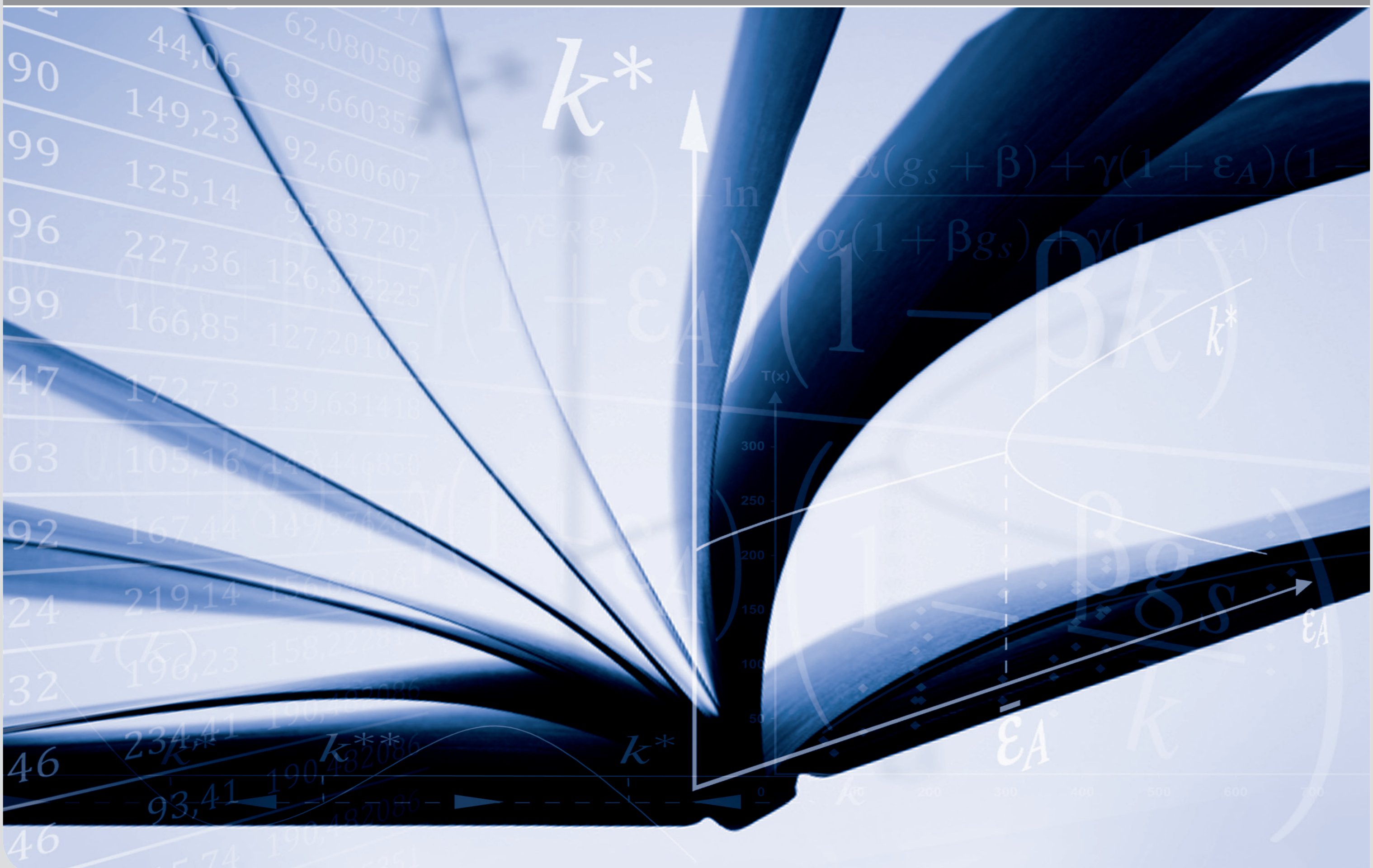
Karlsruher Institut für Technologie

Voting: a machine learning approach

by Dávid Burka, Clemens Puppe, László Szepesváry and Attila Tasnádi

No. 145 | NOVEMBER 2020

WORKING PAPER SERIES IN ECONOMICS



Impressum

Karlsruher Institut für Technologie (KIT)
Fakultät für Wirtschaftswissenschaften
Institut für Volkswirtschaftslehre (ECON)

Kaiserstraße 12
76131 Karlsruhe

KIT – Die Forschungsuniversität in der Helmholtz-Gemeinschaft

Working Paper Series in Economics
No. 145, November 2020

ISSN 2190-9806

econpapers.wiwi.kit.edu

Voting: a machine learning approach

Dávid Burka,[1] Clemens Puppe,[2] László Szepesváry[3] and Attila Tasnádi[4] *

(1) Department of Computer Science, Corvinus University of Budapest, Fővám tér 8, H – 1093 Budapest, Hungary, *david.burka@uni-corvinus.hu*

(2) Department of Economics and Management, Karlsruhe Institute of Technology, D – 76131 Karlsruhe, Germany, and Higher School of Economics, Russian Federation, *clemens.puppe@kit.edu*

(3) Department of Operations Research and Actuarial Sciences, Corvinus University of Budapest, Fővám tér 8, H – 1093 Budapest, Hungary, *szepesvaryl@gmail.com*

(4) Department of Mathematics, Corvinus University of Budapest, H – 1093 Budapest, Fővám tér 8, Hungary, *attila.tasnadi@uni-corvinus.hu*

November 2020

Abstract *Voting rules can be assessed from quite different perspectives: the axiomatic, the pragmatic, in terms of computational or conceptual simplicity, susceptibility to manipulation, and many others aspects. In this paper, we take the machine learning perspective and ask how 'well' a few prominent voting rules can be learned by a neural network. To address this question, we train the neural network to choosing Condorcet, Borda, and plurality winners, respectively. Remarkably, our statistical results show that, when trained on a limited (but still reasonably large) sample, the neural network mimics most closely the Borda rule, no matter on which rule it was previously trained. The main overall conclusion is that the necessary training sample size for a neural network varies significantly with the voting rule, and we rank a number of popular voting rules in terms of the sample size required.*

Keywords: voting, social choice, neural networks, machine learning, Borda count.

*Versions of this work have been presented at the Workshop on Game Theory and Social Choice (Budapest, December 2015), at the Workshop on Voting Theory and Social Choice (Berlin, June 2016), at the COMSOC Conference (Toulouse, June 2016), and at the 13th Meeting of the Society for Social Choice and Welfare (Lund, June 2016). We are grateful to the audiences for comments and suggestions, in particular to Ulle Endriss, Ilan Nehama, Miklós Pintér, Balázs Sziklai and William Zwicker. Earlier results using a different neural network are reported in Burka et al. (2016) which the present work supersedes.

1 Introduction

Some Background on Voting Theory

Is there an optimal voting rule? This question has occupied a central role in political and social theory for a long time, its origins can be traced back (at least) to the writings of Ramón Llull and Nikolaus of Kues.[1] The issue at hand found a particularly clear expression in the debate between the Marquis de Condorcet and Jean-Charles de Borda about the appropriate method to elect new members to the French Academy of Sciences in the late 18th century. The Chevalier de Borda recognized the serious shortcomings of the simple plurality rule used at that time by the Academy and suggested an alternative method based on the aggregation of scores received by each candidate from the voters – the method nowadays known as the *Borda rule.* Nicolas de Condorcet, then secretary of the Academy, criticized Borda's method by noticing that it sometimes fails to elect a candidate that would receive majority support in a pairwise comparison against *all* other candidates, a so-called *Condorcet winner.*[2] However, an evident disadvantage of pairwise majority comparisons of candidates is that they sometimes result in cyclic collective preferences, a phenomenon already noticed by Condorcet himself. In particular, in some voting constellations, a Condorcet winner does not exist. On the other hand, the (perhaps less obvious) disadvantage of Borda's rule is that the social evaluation of two candidates not only depends on their relative position in the voters' rankings but on their cardinal scores, i.e. on their evaluation vis-á-vis *other* candidates. Borda's method thus violates a condition known as 'independence of irrelevant alternatives,' henceforth simply, *binary independence.*

The controversy about the 'best' voting rule culminated in Arrow's famous impossibility theorem (1951/63) which states that the only aggregation methods that always produce consistent (i.e. transitive) social evaluations, respect unanimous consent in pairwise comparisons of candidates and satisfy binary independence are the *dictatorial* ones. Arrow's theorem thus shows that *every* democratic (i.e. non-dictatorial) election method suffers from some shortcomings, or even 'paradoxes.' But this insight has, of course, not ended the search for the optimal election method. By contrast, it has made the underlying problem even more urgent.

The predominant method of arguing for, or against, a particular voting method is *axiomatic.* In this spirit, axiomatic characterizations have been put forward for the Borda rule (Smith, 1973; Young, 1974; Saari, 2000), for general scoring rules (Young, 1975) and for voting methods that always choose the Condorcet winner if it exists, for instance, the Copeland method (Henriet, 1985) and the Kemény-Young method (Young and Levenglick, 1978).[3] These and many other contributions in the same spirit have

[1] An introduction to the history of social choice theory with reprints of classic contributions can be found in the volume edited by McLean and Urken (1995). For an illuminating account especially of the role of Llull and Nikolaus (Cusanus) in this context, see also the Web edition of Llull's writings on electoral systems (Drton et. al., 2004) and the article Hägele and Pukelsheim (2008) on the relevant parts in Nikolaus' work *De concordantia catholica.*

[2] The election procedure that Llull describes in his *De arte eleccionis* (1299) is indeed based on pairwise majority comparisons in the spirit of Condorcet, while the method suggested by Nikolaus of Kues in the year 1433 for the election of the emperor of the Holy Roman Empire is the scoring method suggested more than three centuries later by Borda (cf. McLean and Urken, 1995; Pukelsheim, 2003).

[3] Axiomatizations of other voting rules and related aggregation procedures include approval voting (Fishburn, 1978), plurality rule (Goodin and List, 2006) and majority judgement (Balinski and Laraki, 2016).

certainly deepened our understanding of the structure of the voting problem. However, by lifting the controversy about different methods to an analogous discussion of their respective properties ('axioms'), the axiomatic approach has not been able to settle the issue. And indeed, a consensus on the original question seems just as far as ever (as argued, for instance, by Risse, 2005).

As a possible route, an 'operations research approach' has been proposed that tries to single out particular election methods as solutions to appropriately defined distance minimization problems, see Elkind et al. (2015) for a recent contribution.[4] However, a very large class of voting rules can be obtained in this way, and the problem is then lifted to the issue of selecting the appropriate distance metric.[5]

Another approach is motivated by the empirical method so successful in many other branches of science. Couldn't one simply argue that the election methods that are predominant in *real life* reveal their superiority due to the very fact that they are widely used for deciding real issues? Doubts about the validity of this claim are in order. Indeed, on the count of empirical success, plurality rule (i.e. the election of the candidate who receives the greatest number of first votes) would fare particularly well. But, if there is one thing on which the experts in voting theory agree, it is the ineptness of that particular voting method in many contexts (see the article 'And the looser is ... plurality voting,' Laslier, 2011).[6]

Finally, starting with the seminal work of Bartholdi et al. (1989), there is now a sizable literature that assesses voting rules in terms of their computational complexity, see Brandt et al. (2016). While it has been argued that computational complexity may serve as a 'shield' against manipulations, high complexity may also have adverse effects on the perceived legitimacy of the outcome of a voting process.

Our Contribution

In this paper, we take a 'quasi-empirical' approach to assessing the complexity of a voting rule by investigating which election method best describes the behavior of a sophisticated machine learning method that operates in a voting environment. More specifically, we ask which voting rule corresponds to the *implicit* selection mechanism employed by a trained neural network. By answering this question we hope to shed light on the 'conceptual' complexity (in a non-technical sense), or the *salience* of different voting rules. Concretely, we trained a Multi-Layer Perceptron, henceforth MLP, to output the Condorcet winner, the Borda winners, and plurality winners, respectively, and statistically compare the chosen outcomes by the trained MLP. It is well-known that MLPs are universal function approximators (Hecht-Nielsen, 1987; Funahashi, 1989), in particular an MLP will learn any voting rule on which it is trained with arbitrary accuracy *provided that the size of the training sample is sufficiently large.* This is where our work comes into the play. Specifically, we investigate the relation between the

[4]Bednay et al. (2017) offer a 'dual' approach based on distance *maximization*.

[5]A noteworthy alternative approach is taken by Nehring and Pivato (2018) who argue for a generalization of the Kemény-Young method on the ground of its superior properties in the general 'judgement aggregation' framework in which the preference aggregation problem occurs only as one particular special case among many others.

[6]There are also experimental studies with non-expert subjects on the question of the public opinion about the 'best' voting method, see, e.g., Giritligil Kara and Sertel (2005). However, the problem of these studies is that it is not clear how to incentivize subjects to give meaningful answers. Moreover, the underlying motives of subjects seem to be particularly hard to identify in this context.

sample size and the accuracy by which the MLP learns different voting rules. Besides the Borda count and plurality voting, we considered two Condorcet consistent voting methods, the Copeland rule and the Kemény-Young method. As a further point of comparison, we also looked at 2-approval voting.

Our empirical results are clear-cut: for limited, but still reasonably large sample sizes, the implicit voting rule employed by the MLP is most similar to the Borda rule and differs significantly from plurality rule; the Condorcet consistent methods such as Copeland and Kemény-Young lie in between. Indeed, the choices made by the MLP are closer to those of the Borda rule than to those by any of the other rules *no matter* whether the MLP was trained on the choice of the Condorcet, Borda, or plurality rule. In view of its popularity and simplicity, the poor performance of plurality rule is remarkable but confirms its bad reputation among social choice theorists. Finally, we also find that 2-approval voting does not perform well in our analysis.[7]

Relation to the Literature

MLPs has been very successfully employed in pattern recognition and a great number of related problems (Haykin, 1999).[8] More generally, neural networks have been used by econometricians for forecasting and classification tasks (McNelis, 2005); in economic theory, they have been applied to bidding behavior (Dorsey et al., 1994), market entry (Leshno et al., 2002), boundedly rational behavior in games (Sgroi and Zizzo, 2009) and for financial market predictions (Fischer and Krauss, 2018; Kim et al., 2020). To the best of our knowledge, the present application to the assessment of voting rules is novel. The papers closest in the literature to ours are Procaccia et al. (2009) and Kubacka et al. (2020). The goal of Procaccia et al. (2009) is to demonstrate the (PAC-)learnability of specific classes of voting rules and to apply this to the automated design of voting rules. Kubacka et al. (2020) investigate the learning rates for the Borda count, the Kemény-Young method and the Dodgson method with a dozen of machine learning algorithms. Their main motivation is to find an effective computation method for the Kemény-Young and Dodgson methods which are known to be computationally complex (NP-hard in the number of alternatives, see Bartholdi et al., 1989).[9]

It is also worth mentioning that Richards et al. (2006), in a somewhat 'converse' approach, employed specific voting rules in the construction of new learning algorithms for 'winner-takes-all' neural networks.

The remainder of the paper is organized as follows. Section 2 introduces our framework, formally defines a number of prominent voting rules and provides a brief overview of the structure of the employed MLP. Section 3 describes the data generation process. Section 4 gives an overview of the learning rates of our voting rules for the two-layer perceptron and a fixed sample size of 1000 profiles in each application. Section 5 looks

[7]We also trained the MLP on 2-approval voting and did not find much improvement; the results can be found at `http://www.uni-corvinus.hu/~tasnadi/results.xlsx`.

[8]Recently, a combination of neural networks has been successfully employed by Silver et al. (2016) to defeat one of the world leading human Go champions.

[9]The NP-hardness results highlight that – despite the universal approximation theorem for neural networks – even MLPs will not be able to forecast the Kemény-Young or Dodgson winners with appropriate precision in a realistic time frame. Kubacka et al. (2020) use the Borda count as a benchmark because of its simplicity. The Borda count could be learned by some methods with a prediction accuracy of 100%, while the highest accuracy found for the Kemény-Young and Dodgson methods is 85% and 87%, respectively. This confirms the results of our study below.

in much more detail at the two-layer and three-layer perceptrons for a large range of sample sizes, and largely confirms the results for the fixed sample. Section 6 concludes.

2 Framework

2.1 Voting rules

Let X be a finite set of alternatives with cardinality q. By $\mathcal{P}$, we denote the set of all linear orderings (irreflexive, transitive and total binary relations) on X. Let $rk[x, \succ]$ denote the *rank* of alternative x in the ordering $\succ \in \mathcal{P}$ (i.e. $rk[x, \succ] = 1$ if x is the top alternative in $\succ$, $rk[x, \succ] = 2$ if x is second-best in $\succ$, and so on). The set of voters is denoted by $N = \{1, \ldots, n\}$. In all what follows, we will assume that n is odd. A vector $(\succ_1, ..., \succ_n) \in \mathcal{P}^n$ is referred to as a *profile of ballots*.

Definition 1. A mapping $F : \mathcal{P}^n \to 2^X \setminus \{\emptyset\}$ that selects a non-empty set of winning alternatives for all profiles of ballots is called a *voting rule*.

Note that this definition allows for ties among the winners. The following voting rules are among the most studied in the literature and will be the subject of our subsequent investigation. Denote the *Borda score* of $x \in X$ in the ordering $\succ$ by $bs\,[x, \succ] := q - rk[x, \succ]$.

Definition 2. The *Borda count* is defined by

$$Borda\,(\succ_1, ..., \succ_n) := \arg\max_{x \in X} \sum_{i=1}^{n} bs[x, \succ_i].$$

For a given profile $(\succ_1, ... \succ_n) \in \mathcal{P}^n$, denote by $v(x, y, (\succ_i)_{i=1}^n)$ the number of voters who prefer x to y, and say that alternative $x \in X$ *beats* alternative $y \in X$ if $v(x, y, (\succ_i)_{i=1}^n) > v(y, x, (\succ_i)_{i=1}^n)$, i.e. if x wins against y in pairwise comparison. Moreover, denote by $l[x, \succ_i]$ the number of alternatives beaten by $x \in X$ for a given profile $(\succ_1, ..., \succ_n)$.

Definition 3. The *Copeland method* is defined by

$$Cop\,(\succ_1, ..., \succ_n) := \arg\max_{x \in X}\ l[x, (\succ_i)_{i=1}^n].$$

In order to define the next voting rule, let

$$\mathcal{D}_{KY}\,(\succ_1, ..., \succ_n) := \arg\max_{\succ \in \mathcal{P}} \sum_{\{x,y \in X,\ x \succ y\}} v(x, y, (\succ_i)_{i=1}^n)).$$

Definition 4. The *Keményi-Young method* chooses the top ranked alternative(s) from the set of linear orderings in $\mathcal{D}_{KY}$, i.e.,

$$x \in Kem{-}You\,(\succ_1, ..., \succ_n) :\Longleftrightarrow \{rk[x, \succ] = 1 \text{ for some } \succ \in \mathcal{D}_{KY}\,(\succ_1, ..., \succ_n)\}\,.$$

Definition 5. The *plurality rule* is defined by

$$Plu\,(\succ_1, ..., \succ_n) := \arg\max_{x \in X}\ \#\{i \in N \mid rk[x, \succ_i] = 1\}\,.$$

Definition 6. The *k-approval voting rule* is defined by

$$k{-}AV\left(\succ_1, ..., \succ_n\right) := \arg\max_{x \in X} \ \#\left\{i \in N \mid rk[x, \succ_i] \leq k\right\}.$$

Definition 7. A *Condorcet winner* is an alternative that beats every other alternative in a pairwise majority comparison.

Note that, if a Condorcet winner exists given a profile of ballots, it must necessarily be unique. It is well-known (and easy to verify) that both the Copeland and Kemény-Young methods are *Condorcet consistent* in the sense that they select the Condorcet winner whenever it exists. None of the other methods listed above is Condorcet consistent.

2.2 The Multi-Layer Perceptron

The fundamental idea behind artificial neural networks, henceforth ANNs, is that a simplified model of the human brain could be the starting point for developing computational models. ANNs mimic the behavior of the brain by following a simple model of connected neurons. ANNs are an appropriate tool for supervised learning. Specifically, we provide the ANN with a set of 'examples' and the respective 'correct responses.' A part of this data constitutes the training set from which the ANN learns the functional relationship between the inputs (i.e. examples) and the outputs (i.e. correct responses). Another part of the data is used for validation, that is for providing a stopping condition of the learning process. A third and final part of the data is used as a test set to evaluate the quality of the learning process. Exposed to a training set, ANNs develop a 'memory' by setting the weights between its neurons appropriately.

The original McCulloch and Pitts (1943) perceptron has a given number of inputs, a single neuron with an activation function determining a single output. The learning process determines the weights of the inputs based on the data provided for supervised learning as described in the previous paragraph. Not surprisingly, a single McCulloch and Pitts perceptron has a limited learning capability. Therefore, in a modern multi-layer perceptron (MLP), a number of perceptrons all facing the same inputs are organized into a 'layer.' The outputs of the perceptrons of the first layer are the inputs of the perceptrons of the next layer and so on, until one arrives at the final layer which yields the outcome results.

While we considered *a priori* the more general case, it turned out that for our purposes an MLP with two layers is sufficient. In fact it is known that two layers are already enough to learn a large class of functions, specifically, one can show that the two-layer perceptron is a universal function approximator, see Hecht-Nielsen (1987) and Funahashi (1989). We checked the robustness of our results by verifying that adding further layers only minimally improved our learning rates. Therefore, we restrict our following description to the case of two-layers (the three- and multi-layered perceptrons work in a completely analogous way).

To define a two-layered perceptron formally, we denote by m, p and r the number of inputs, hidden neurons and output neurons, respectively. Figure 1 illustrates the general structure of a two-layered perceptron. The weight matrices $V \in \mathbb{R}^{(m+1)\times p}$ and $W \in \mathbb{R}^{(p+1)\times r}$ are determined by the backpropagation algorithm of Rumelhart et al. (1986). The trained two-layered perceptron gathers its knowledge in $\mathbf{V}$ and $\mathbf{W}$ from the training set, which in our case are various sets of profiles of ballots with prespecified

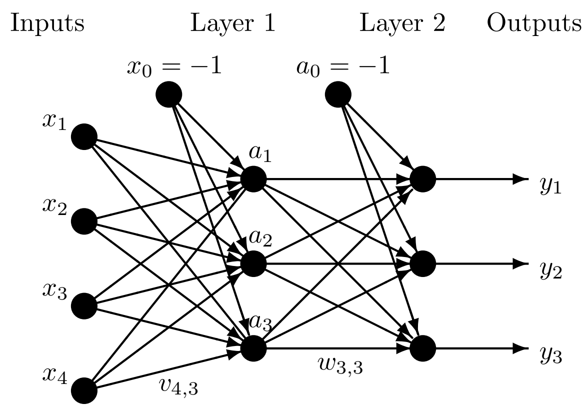


Figure 1: Structure of the MLP

winners. For profiles without specification of a winner we then obtain the 'choice' of the trained neural network by determining first the activation level

$$h_j := \sum_{i=0}^{m} v_{ij} x_i \ , \quad a_j := g(h_j) = \frac{1}{1 + e^{-\beta h_j}} \ , \tag{2.1}$$

for each hidden neuron, and subsequently the activation level

$$o_k := \sum_{j=0}^{p} w_{jk} a_j \ , \quad y_k := g(o_k) = \frac{1}{1 + e^{-\beta o_k}} \ , \tag{2.2}$$

for each output neuron, where g is the so-called activation or threshold function. For more detailed description and analysis of neural networks, see, e.g., Haykin (1999) or Marshland (2009).

3 Data Generation

In order to investigate the speed and accuracy with which the MLP learns different voting rules, we randomly generated a set of profiles using the impartial culture (IC) assumption according to which each preference relation is assigned independently and with equal probability to each voters.[10]

We considered cases with 7, 9 or 11 voters and 3, 4 or 5 alternatives. We encoded preference orderings in the following way. Let $X = \{x_1, x_2, \ldots, x_q\}$. If $x_{i_1} \succ x_{i_2} \succ \cdots \succ x_{i_q}$, where $(i_1, \ldots, i_q)$ is a permutation of $(1, \ldots, q)$, we store the respective pairwise comparisons in a vector corresponding to the upper triangular matrix $(a_{jk})_{j=1,k=j+1}^{q,q}$ with $a_{jk} = 1$ if $x_j \succ x_k$ and $a_{jk} = 0$ otherwise. For example, the ordering $x_1 \succ x_4 \succ x_2 \succ x_3$ is coded by $(1,1,1,1,0,0)$ corresponding to the binary comparisons x_1 vs. x_2, x_1 vs. x_3, x_1 vs. x_4, x_2 vs. x_3, x_2 vs. x_4, x_3 vs. x_4. A profile

[10]In the working paper Burka et al. (2016), we also report the results for the anonymous impartial culture (AIC) assumption and other underlying distributions. A careful inspection of the results shows that the influence of the underlying distribution is small; if at all, our main findings are even more pointed under the AIC assumption.

is then given by a row vector with $n \cdot q(q-1)/2$ entries. We did not include results for encoding a preference relation by the simple ordering of alternatives, since this encoding favored the Borda count for obvious reasons.

To complete an input of a training set we also specified a target alternative, the 'winner(s)' of the respective voting rule. For this we used the so-called '1-of-N encoding,' e.g. the indicator vector $(0,\ldots,0,1,0,\ldots,0)$, in which the ith coordinate equals 1 if and only if the respective alternative is a winner in that profile for the investigated voting rule.

When training for a set of winners, we considered three scenarios. First, we trained on the subset of profiles with a (necessarily unique) Condorcet winner from the randomly generated profiles; secondly, we trained the Borda count as a set-valued function on the set of randomly generated profiles; and thirdly, we trained the plurality rule as a set-valued function again on the set of randomly generated profiles. In the first scenario, we decided to train on Condorcet winners (and used only profiles for which a Condorcet winner exists) because we did not want to commit to a particular Condorcet consistent extension.

For all scenarios, we generated random training sets ranging from 100 to 3000 profiles. Section 4 gives an overview of the results for sample size 1000 and the two-layered perceptron, while Section 5 uses the entire range of sample sizes and also presents the results for the three-layered perceptron in the case of 11 voters (to keep the number of figures manageable, we omitted the other cases for the three-layered perceptron; they can be found at `http://www.uni-corvinus.hu/~tasnadi/results.xlsx`).

The generation of profiles, training sets, the training of the neural networks and, subsequently, the 'predicting' of the winning alternative(s) without specified target values was done in MatLab. The statistical evaluation was carried out in Excel. All program codes are available from the authors upon request.[11]

To train an MLP we generated five random sample training seeds and took five random network seeds for the training procedure of the MLP. Finally, we selected one random testing seed pair for each training seed to generate test samples as well. For a given random training seed the five trained MLPs were each tested using the sample based on the respective testing seed pair. An alternative was selected as a winner on a test sample if it was selected by the majority of the five MLPs (i.e. selected by at least three out of the MLPs corresponding to the five possible network seeds). In order to determine the prediction accuracy, we used the five test seeds and took the average of the five hitting ratios to stabilize our results. Altogether, we evaluated and aggregated 25 results for each profile and for each testing sample to obtain a prediction accuracy for a given number of voters and alternatives.

4 Overview of Results for Fixed Sample Size

To give a first overview of the results as a function of the number of alternatives and the number of voters, we consider in this section the results from the two-layered perceptron with a fixed sample size of 1000 profiles. First, consider the case in which we took profiles with a Condorcet winner as training sample and the Condorcet winner

[11] For all results of the present study, we used the MatLab MLP; by contrast, the results of Burka et al. (2016) were obtained with Marshland's (2009) Python code.

as target value. Table 1 shows the corresponding results for three, four, and five alternatives and 7, 9, and 11 voters, respectively. The table entries give the average percentages of those cases in which a trained MLP selects a winner of the method appearing in the respective column heading. As can be seen, the Borda count performs best for the cases with five alternatives and for the case with four alternatives and 11 voters. It is particularly remarkable that the Borda count outperforms in these cases both the Copeland and the Kemény-Young method even though these are Condorcet consistent while the Borda is not.

Method	Cop	Kem-You	Borda	Plu	2-AV
q=3, n=7	96.12%	96.12%	94.36%	86.82%	80.94%
q=3, n=9	97.88%	97.88%	95.60%	88.28%	79.00%
q=3, n=11	97.58%	97.56%	94.50%	87.12%	77.80%
q=4, n=7	96.68%	93.58%	92.90%	80.80%	84.00%
q=4, n=9	92.54%	89.40%	92.30%	76.98%	82.62%
q=4, n=11	89.58%	86.38%	90.72%	75.76%	81.40%
q=5, n=7	84.30%	80.92%	87.12%	71.26%	78.86%
q=5, n=9	78.14%	74.48%	81.98%	66.22%	74.66%
q=5, n=11	77.02%	72.76%	80.16%	63.44%	72.78%

Table 1: Trained on Condorcet winners

While the two Condorcet consistent methods are also not far from MLPs choices (with a slight advantage of the Copeland method as compared to the Kemény-Young method), the other methods differ significantly, in particular for more alternatives. Interestingly, and in contrast to the plurality rule, coincidence of MLPs choice with the 2-approval voting winner is larger for four alternatives than for three and five alternatives.[12]

We obtain essentially the same ordering of methods in terms of coincidence with MLP's choices if we train the neural network either on Borda winners (see Table 2), or on plurality winners (see Table 3). In these cases the Borda count performs unanimously best among all voting methods. Not surprisingly, MLP's choice behavior comes even closer to the Borda count if trained to choose the Borda winner. On the other hand, it is remarkable that, except in the case of three alternatives and 7 voters, plurality rule does not seem to perform better even when the MLP is trained to choose the plurality winner (compare the entries in the column for plurality rule across Tables 1-3).

It is worth mentioning that the great majority of percentages in Tables 1-3 is decreasing both in the number of alternatives and in the number of voters for all investigated voting rules. However, this does not necessarily mean that the MLP learns these rules with lower accuracy for higher number of alternatives and voters. The reason is that an increase in the number of alternatives and voters over-proportionally increases the size and dimension of the input data; for instance, for $q = 3$ and $n = 7$ the dimension of the input (under binary encoding) is $n \cdot q(q-1)/2 = 21$ while it grows to 66 as we move to the case $q = 4$ and $n = 11$.

[12] In evaluating the differences in percentages one must keep in mind that different methods agree on many profiles, so that even small differences in percentage points may hint at significant underlying differences in learning performance.

Method	Cop	Kem-You	Borda	Plu	2-AV
$q = 3, n = 7$	93.60%	93.60%	96.50%	86.08%	83.24%
$q = 3, n = 9$	93.36%	93.34%	98.20%	87.08%	81.82%
$q = 3, n = 11$	92.40%	92.32%	97.86%	86.48%	80.84%
$q = 4, n = 7$	88.84%	86.14%	95.38%	78.82%	85.62%
$q = 4, n = 9$	87.40%	84.30%	94.08%	75.86%	83.94%
$q = 4, n = 11$	87.32%	84.16%	92.78%	75.34%	82.86%
$q = 5, n = 7$	82.00%	78.70%	88.26%	69.88%	78.60%
$q = 5, n = 9$	78.42%	74.94%	84.20%	65.36%	76.26%
$q = 5, n = 11$	76.56%	72.60%	80.96%	62.88%	73.10%

Table 2: Trained on Borda winners

Method	Cop	Kem-You	Borda	Plu	2-AV
$q = 3, n = 7$	94.44%	94.42%	95.64%	86.24%	82.54%
$q = 3, n = 9$	92.00%	91.82%	94.46%	86.16%	80.14%
$q = 3, n = 11$	89.76%	89.52%	92.62%	84.86%	78.48%
$q = 4, n = 7$	80.52%	78.10%	83.74%	73.16%	80.38%
$q = 4, n = 9$	76.48%	73.82%	79.56%	69.72%	75.54%
$q = 4, n = 11$	77.42%	74.32%	80.52%	69.58%	75.46%
$q = 5, n = 7$	66.62%	63.68%	70.12%	59.90%	67.04%
$q = 5, n = 9$	65.56%	62.72%	67.32%	57.02%	64.34%
$q = 5, n = 11$	60.78%	57.10%	61.74%	53.02%	58.80%

Table 3: Trained on plurality winners

5 Detailed Results

In this section, we investigate the robustness of our results both with respect to the sample size and the depth of the MLP. In particular, we examine if an increase in the depth of the network improves the learning accuracy for identical sample sizes. We do not find much improvement both with respect to the speed and the accuracy of learning; moreover, the absence of a positive effect of increasing the depth of the MLP by adding a layer holds essentially at all sample sizes.

Since the results are qualitatively similar for all number of voters, we present here only the results with 11 voters, all other results are downloadable at `http://www.uni-corvinus.hu/~tasnadi/results.xlsx`.[13] The results generally confirm the picture of the previous section. For 'small' sample sizes the choices of the MLP resemble most closely the Borda rule no matter if it was trained on Condorcet winners, the Borda count or the plurality rule. But, of course, the size of the training sample has to be compared to the complexity of the problem. For five alternatives, the Borda rule unanimously outperformed all other rules for all investigated sample sizes and all training treatments. The Borda rule also outperformed all other rules when the MLP was

[13]The downloadable files also contain the prediction accurateness of the two-layered perceptron with 5 and 10 hidden neurons, and for the three-layered perceptron with 5 and 10 hidden neurons in the first hidden layer and 5 and 10 hidden neurons in the second hidden layer.

trained either on plurality rule or the Borda rule itself. The picture becomes different only when the MLP was trained to choose the Condorcet winner; in that case, MLP's choices are closer to the Borda count for sample sizes up to slightly more than 1000 in the case of four alternatives, up to a few hundreds in the case of three alternatives.

5.1 Trained on Condorcet winners

The following six figures (Figures 2 – 7) present the results for the two-layered (2LP) and three-layered perceptrons (3LP) when trained on the set of profiles with Condorcet winners.

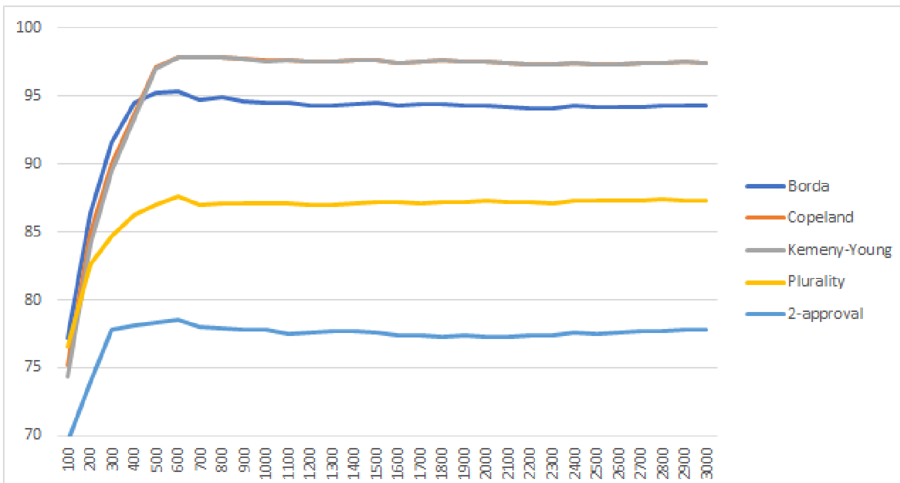


Figure 2: 2LP trained on Condorcet winners for $q = 3$

In Figure 2, we see that for three alternatives and sample sizes of more than 500 the 2LP trained on the set of profiles with Condorcet winners makes choices most similar to the Copeland and the Kemény-Young method with a slight (and barely visible) advantage for the Copeland method. At a sample size of 3000, the 2LP chooses the same alternative as the Kemény-Young and the Copeland method in 97.43% of the cases (for either rule). On the other hand, for small sample size of up to 400 the 2LP behaves more like the Borda count.

Figure 3 shows the case of four alternatives. One can see that when trained on the set of profiles with Condorcet winners the 2LP behaves now more similar to the Borda count than to the two Condorcet consistent methods even for sample sizes up to 1100. For sample sizes larger than 1100 it comes closest to the Copeland method. For a sample size of 3000 the trained 2LP makes choices according to the Copeland, the Kemény-Young, and the Borda methods in 97.19%, 93.85% and 91.29% of the cases, respectively.

Figure 4 shows the case of five alternatives. Now the 2LP even though trained on the set of profiles with Condorcet winners behaves most similar to the Borda count for all considered sample sizes. The trained 2LP behaves still far more like the Copeland than the Kemény-Young method. For a sample size of 3000 the 2LP makes the same

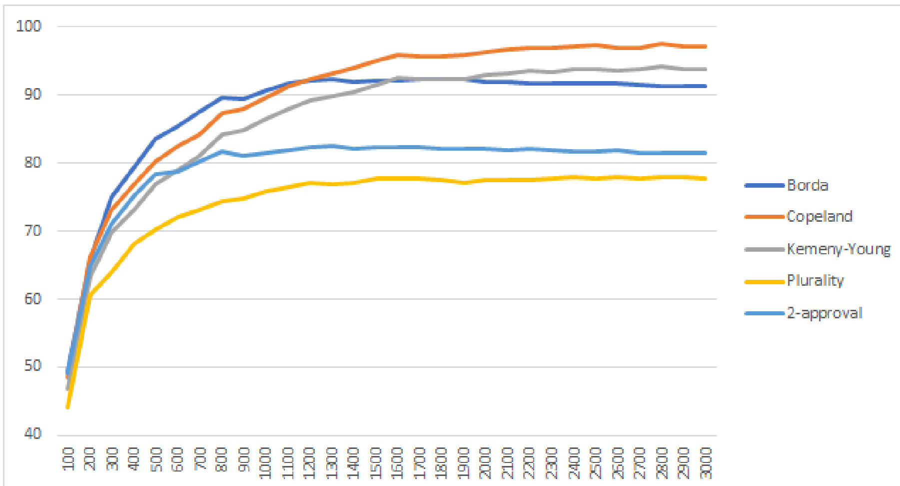


Figure 3: 2LP trained on Condorcet winners for $q = 4$

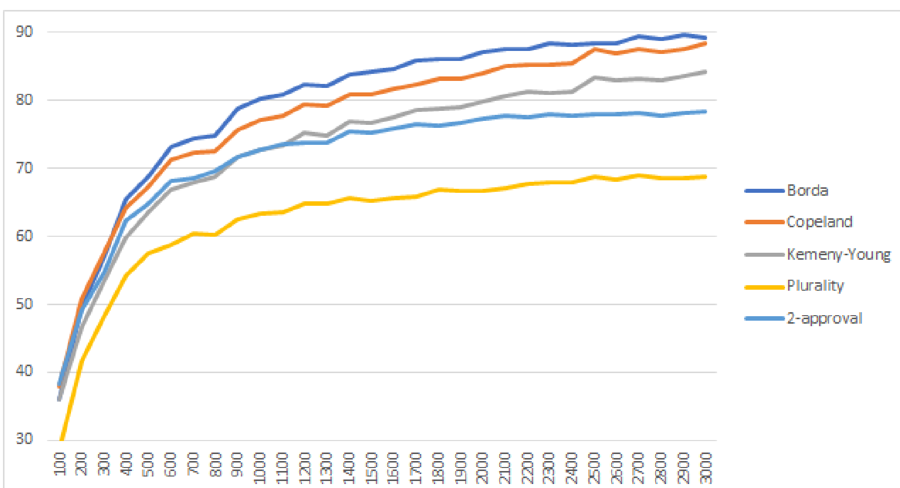


Figure 4: 2LP trained on Condorcet winners for $q = 5$

choice as the Copeland, Kemény-Young and Borda methods in 88.35%, 84.2% and 89.17% of the cases, respectively.

Figures 5 – 7 contain the respective results for the three-layered perceptron; the differences between the 2LP and 3LP are minimal.[14]

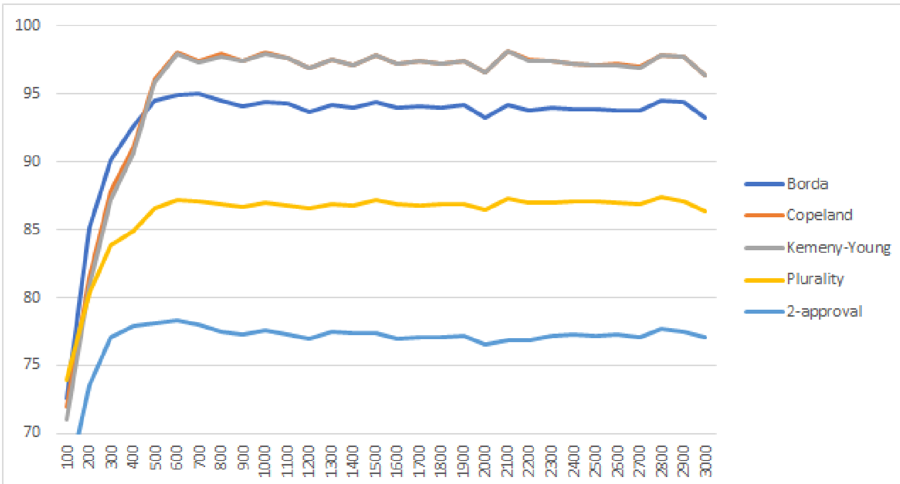


Figure 5: 3LP trained on Condorcet winners for $q = 3$

5.2 Trained on the Borda count

Figure 8 shows the learning rates of the 2LP when trained on the Borda winners. The rates for the Borda count itself are close to 100%. We also see almost identical percentages for the Copeland and Kemény-Young methods (again with a slight advantage for the Copeland method). Interestingly, the 2LP behaves more like the two Condorcet consistent methods than like the other two scoring methods, i.e. plurality rule and 2-approval voting. The learning accuracy for the Borda count at a sample size of 3000 equals 97.35%. The rates for the Copeland and Kemény-Young methods are 92.21% and 92.19%, respectively. To interpret these figures, one has to keep in mind that for three alternatives the choices of different rules are more similar than for a larger number of alternatives.

The picture for four and five alternatives (Figures 9 and 10) is very similar. Now, the advantage for the Copeland over the Kemény-Young method is more pointed. The learning accuracy for the Borda count at a sample size of 3000 equals 97.15% for $q = 4$ and 93.31% for $q = 5$. As a side observation, we note that in contrast to the case of three alternatives 2-approval voting fares better than plurality rule.

Figures 11 – 13 show the results from the three-layer perceptron. One can see that adding a further layer does not change the graphs much; indeed, the improvement in predicting the Borda winner is minimal at larger sample sizes. The learning accuracy for the Borda count at a sample size of 3000 equals 97.57% for $q = 3$ which is just slightly

[14]Inspecting the precise numerical values one finds (surprisingly, perhaps) that the 3LP differs in slightly more cases from the voting rule on which it was trained than the 2LP.

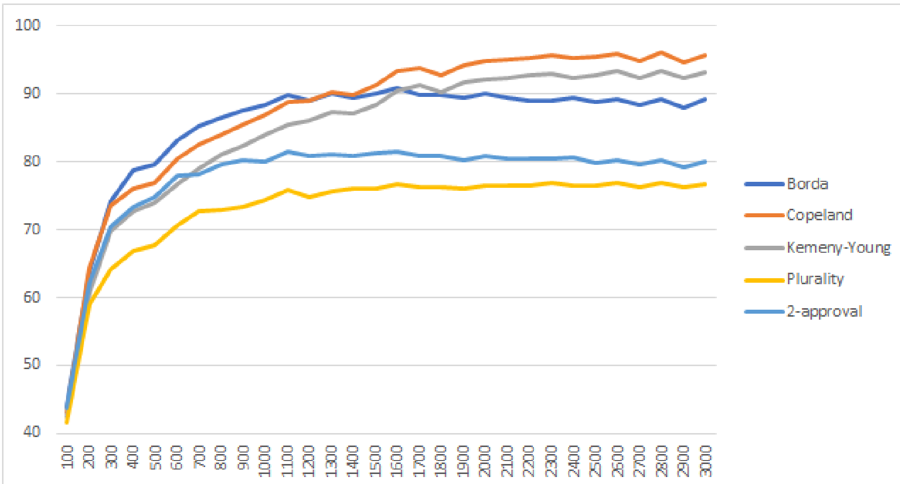


Figure 6: 3LP trained on Condorcet winners for $q = 4$

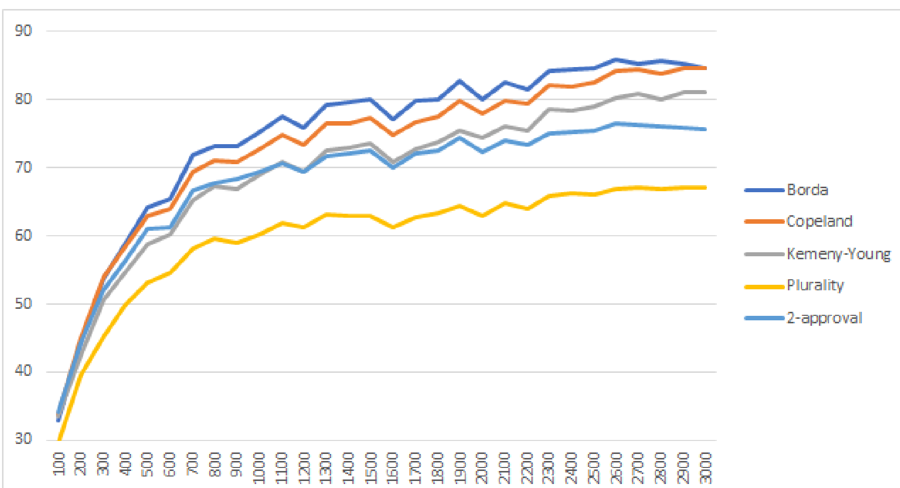


Figure 7: 3LP trained on Condorcet winners for $q = 5$

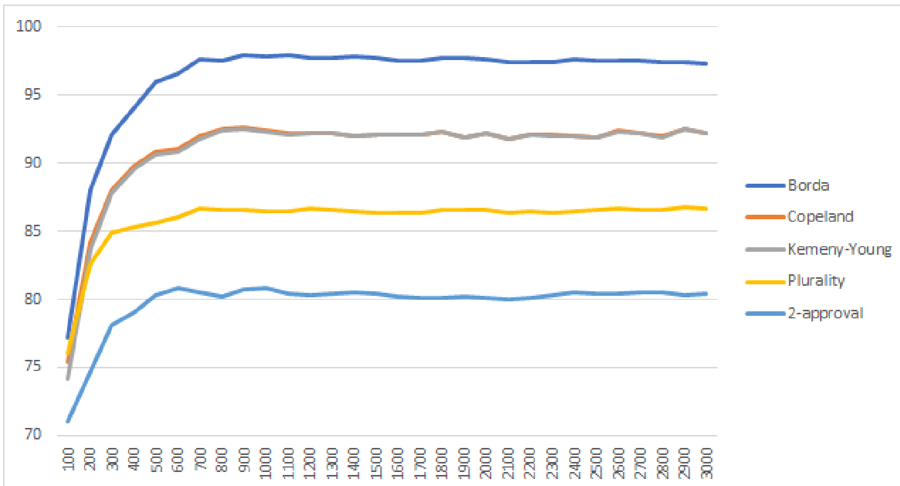


Figure 8: 2LP trained on the Borda count for $q = 3$

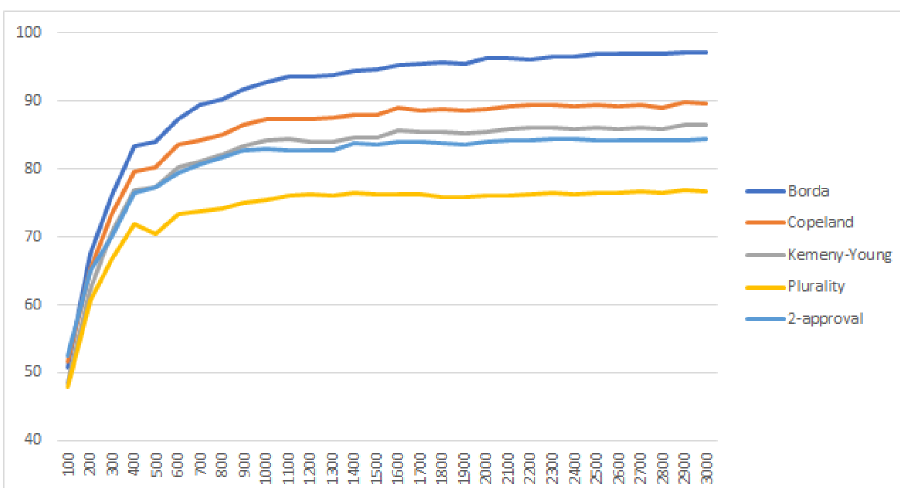


Figure 9: 2LP trained on the Borda count for $q = 4$

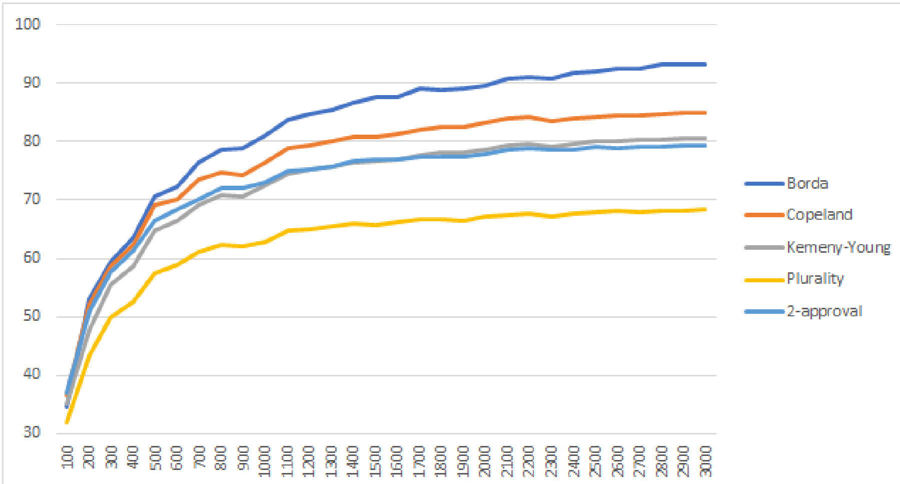


Figure 10: 2LP trained on the Borda count for $q = 5$

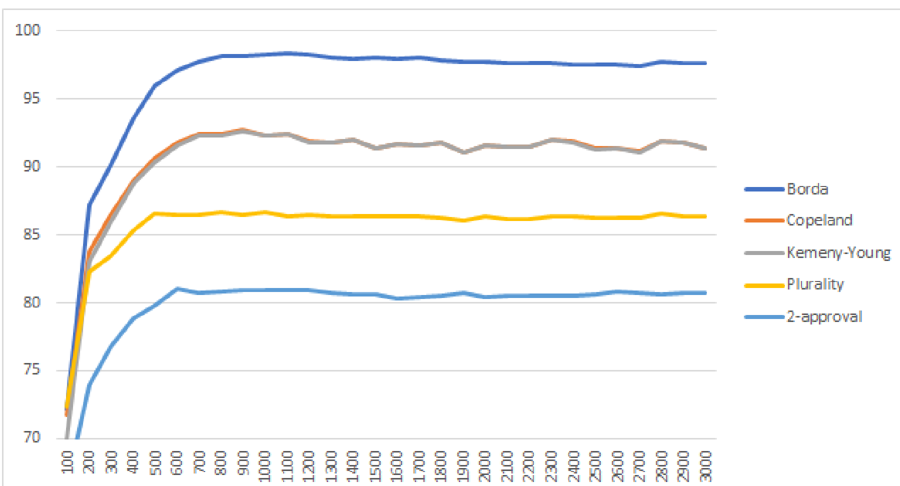


Figure 11: 3LP trained on the Borda count for $q = 3$

higher than in case of the 2LP. We conclude that adding a further layer improves our results just minimally. Interestingly, for small sample sizes the 2LP performs even better than the 3LP. For $q = 3$, the cutting sample size is 500 for which both the two-layer and the three-layer perceptron's prediction accuracy for the Borda count equals 95.96%.

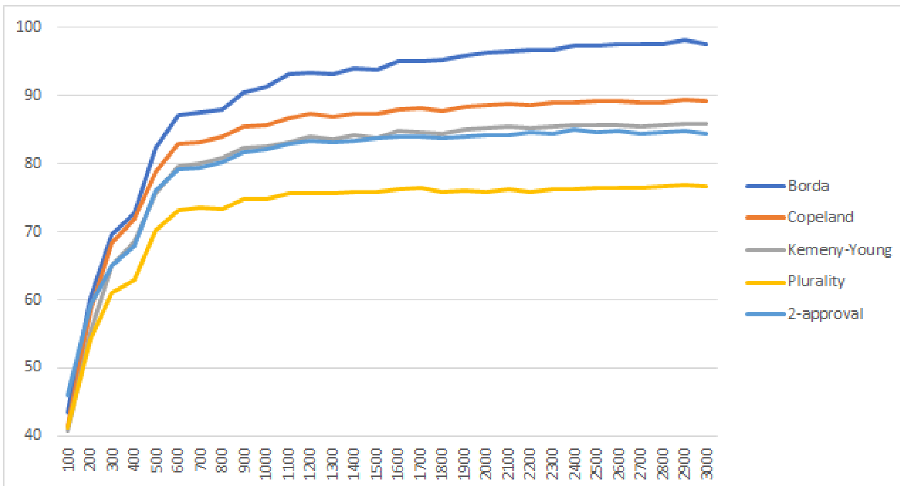


Figure 12: 3LP trained on the Borda count for $q = 4$

For $q = 4$, the learning accuracy for the Borda count at a sample size of 3000 equals 97.58%. Again, for lower and medium ranges of the sample sizes the 2LP performs better than the 3LP; for instance, at a sample size of 1000 the 2LP's prediction accuracy for the Borda count equals 92.78%, while the same value for the 3LP is only at 91.22%.

Finally, for $q = 5$ the learning accuracy of the 3LP for the Borda count at a sample size of 3000 is 91.84%. Although the rate is somewhat lower than for three and four alternatives, the trained 3LP still behaves much more like the Borda count than any of the other methods. Now the Copeland method performs significantly better than the Kemény-Young method.

5.3 Trained on plurality rule

In line with the results of Section 4, the 2LP hardly learned plurality rule for any number of alternatives (see Figures 14 – 16), and adding a further layer does not help the situation. Since plurality rule is one of the simplest voting rules it is surprising to see the MLP performing so poorly. One possible reason is the fact the input (individual preferences) contains a lot of superfluous information since only the top alternatives count.

For all numbers of alternatives, even though the MLP was trained on plurality rule, the Borda count performed best followed by the Copeland and Kemény-Young methods. Plurality rule itself is on the fourth rank beating only 2-approval voting.

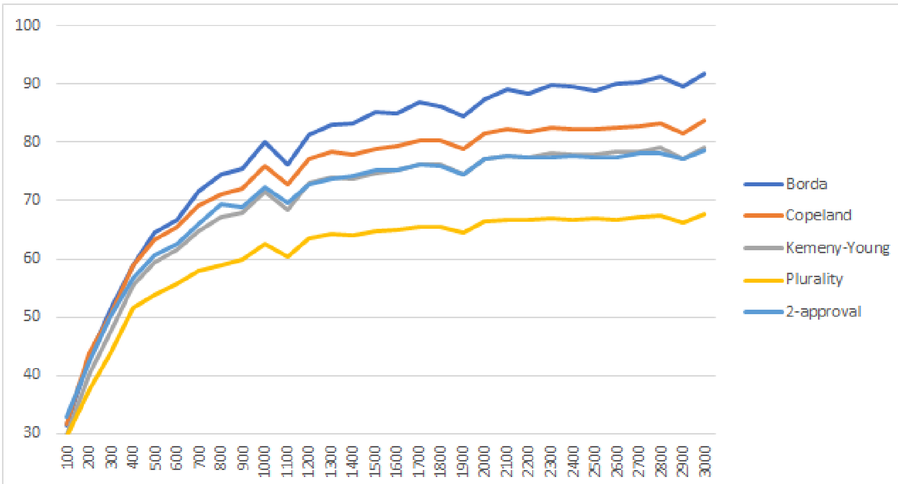


Figure 13: 3LP trained on the Borda count for $q = 5$

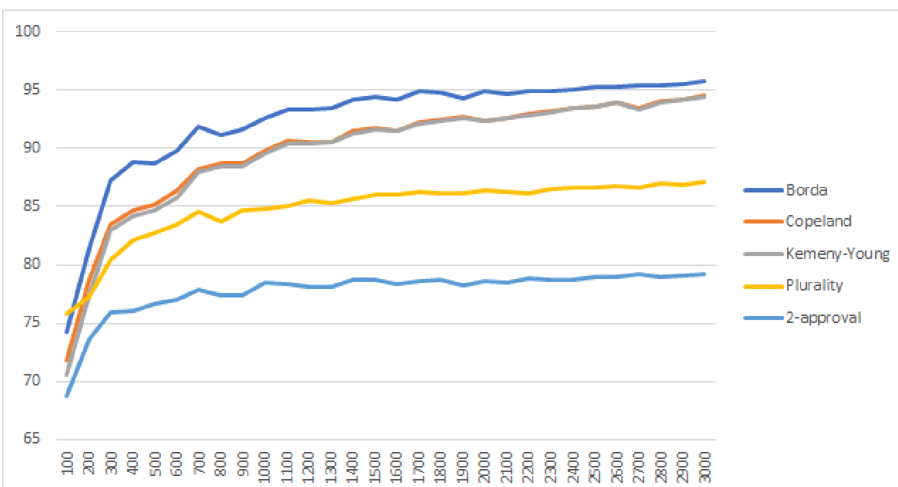


Figure 14: 2LP trained on the plurality rule for $q = 3$

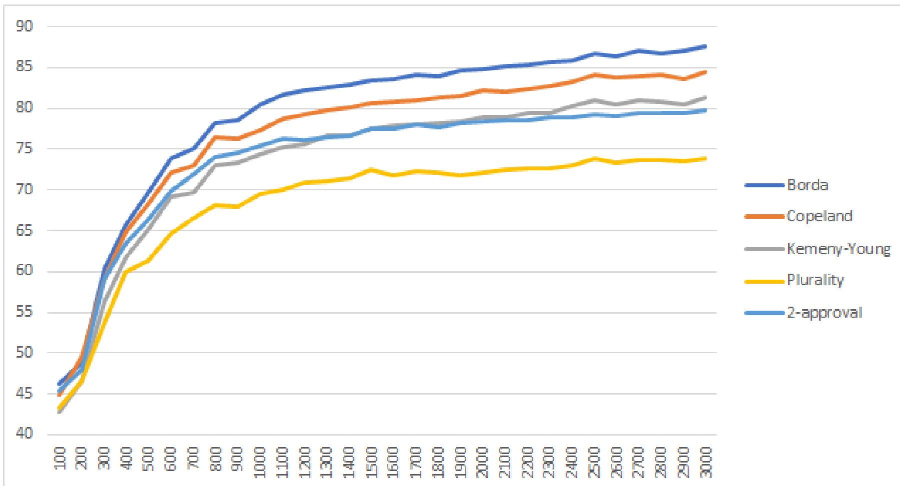


Figure 15: 2LP trained on the plurality rule for $q = 4$

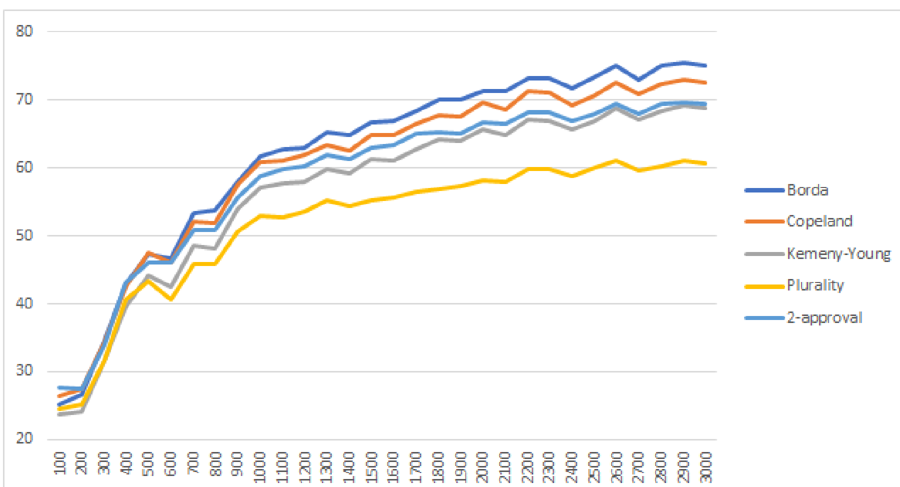


Figure 16: 2LP trained on the plurality rule for $q = 5$

6 Concluding Remarks

Our results demonstrate that among a number of popular voting rules, the Borda count enjoys a special status from the viewpoint of machine learning. Indeed, among a number of popular voting rules, it seems to be the one that best represents the overall behavior of our trained MLP.

Due to the theoretical properties of MLPs, *every* voting rule can be learned by the trained network. In particular, if trained on some rule different from the Borda count the MLP's choices must necessarily converge to the choices of that rule. However, as we have shown, the required size of the training sample can become very large for some rules. For the Condorcet consistent rules, the sample size had to be beyond 1000 in the case of three and four alternatives, and beyond 3000 in the case of five alternatives. For smaller sample sizes, the Borda count outperformed both the Copeland and the Kemény-Young method. The Borda count also outperformed plurality rule in all cases and for all sample sizes even when the MLP was trained to choose the plurality winner.

Thus, from a machine learning perspective, the Borda count can be viewed as the most *salient* of a number of popular voting methods: it is the voting rule that best describes the behavior of a trained neural network in a voting environment for limited sample sizes. One should be careful, however, in using this finding as an argument for the general superiority of the Borda count vis-à-vis other voting rules, even the ones tested here. Indeed, our results may 'only' show that the internal topology of the employed MLP is best adapted to the 'linear' mathematical structure underlying the Borda rule. But then again, if this common underlying structure is successful in a number of different application areas, the Borda count must at least be considered as a serious contender in the competition for 'optimal' voting rule.

One may interpret learning by neural networks also as a device to select a 'suitable' degree of internal complexity. On such an account, plurality rule and its variant 2-approval turn out to be too simple while the two investigated Condorcet consistent methods seem to be too sophisticated. When choosing a winner, the MLP obviously uses more information than only the top ranked alternatives in each ballot. On the other hand, it does also not seem to make the pairwise comparisons necessary in order to determine the Copeland or Kemény-Young winners. The comparison of the learning rates of the Copeland method vis-a-vis the Kemény-Young method is well in line with this interpretation: the computationally more complex of these two methods, the Kemény-Young rule, performs consistently worse.

Based on our analysis one might conjecture that the intuitive choices of humans not trained in social choice theory would also be more in line with the Borda count than with other voting methods. However, this would have to be examined by carefully designed experiments with human subjects.

References

[1] Arrow, K. J. 1951/63, *Social Choice and Individual Values* (First edition: 1951, second edition: 1963), Wiley, New York.

[2] Balinski, M. and R. Laraki (2016), Majority judgement vs. majority rule, *preprint*, https://hal.archives-ouvertes.fr/hal-01304043.

[3] Bartholdi, J., C.A. Tovey and M.A. Trick (1989), Voting schemes for which it can be difficult to tell who won the election, *Social Choice and Welfare* **6**, 157-165.

[4] Bednay, D., A. Moskalenko and A. Tasnádi (2017), Does avoiding bad voting rules result in good ones?, *Operations Research Letters* **45**, 448-451.

[5] Brandt, F., V. Conitzer, U. Endriss, J. Lang and A.D. Procaccia (2016), *The Handbook of Computational Social Choice*, Cambridge University Press, Cambridge.

[6] Burka, D., C. Puppe, L. Szepesváry and Tasnádi, A. (2016), Neural networks would 'vote' according to Borda's rule. Working Paper Series in Economics # 96, Karlsruher Institut für Technologie (KIT), `https://doi.org/10.5445/IR/1000062014` .

[7] Dorsey, R.E., J.D. Johnson and M.V. van Boening (1994), The use of artificial neural networks for estimation of decision surfaces in first price sealed bid auctions, in W.W. Cooper and A.B. Whinston (eds.), *New decisions in computational economics*, 19-40, Kluwer Academic Publishing, Dordrecht.

[8] Drton, M., G. Hägele, D. Haneberg, F. Pukelsheim and W. Reif (2004), A rediscovered Llull tract and the Augsburg Web Edition of Llull's electoral writings, *Le Médiéviste et l'ordinateur*, **43**, http://lemo.irht.cnrs.fr/43/43-06.htm.

[9] Elkind, E., P. Faliszewski and A. Slinko (2015), Distance rationalization of voting rules, *Social Choice and Welfare* **45**, 345-377.

[10] Fishburn, P.C. (1978), Axioms for approval voting: direct proof, *Journal of Economic Theory* **19**, 180-185, corrigendum **45** (1988) 212.

[11] Fischer, T. and C. Krauss (2018), Deep learning with long short-term memory networks for financial market predictions, *European Journal of Operational Research* **270**, 654-669.

[12] Funahashi, K.I. (1989), On the approximate realization of continuous mappings by neural networks, *Neural Networks* **2**, 183-192.

[13] Giritligil Kara, A.E. and M.R. Sertel (2005), Does majoritarian approval matter in selecting a social choice rule? An exploratory panel study, *Social Choice and Welfare* **25**, 43-73.

[14] Goodin, R.E. and C. List (2006), A conditional defense of plurality rule: Generalizing May's theorem in a restricted informational environment, *American Journal of Political Science* **50**, 940-949.

[15] Hägele, G. and F. Pukelsheim (2008), The electoral systems of Nicholas of Cusa in the *Catholic Concordance* and beyond, in: G. Christianson, T.M. Izbicki and C.M. Bellitto (eds.), *The church, the councils, and reform – The legacy of the Fifteenth Century*, Catholic University of America Press, Washington D.C.

[16] Haykin, S. (1999), *Neural Networks: A Comprehensive Foundation*, 2nd Edition, Prentice-Hall.

[17] Hecht-Nielsen, R. (1987), Kolmogorov's mapping neural network existence theorem, *Proceedings of the International Conference on Neural Networks*, vol. III, p. 11-13, IEEE Press, New York.

[18] Henriet, D. (1985), The Copeland choice function: an axiomatic characterization, *Social Choice and Welfare* **2**, 49-63.

[19] Kim, A., Y. Yang, S. Lessmann, T. Ma, M.-C. Sung and J. E. V. Johnson (2020) Can deep learning predict risky retail investors? A case study in financial risk behavior forecasting, *European Journal of Operational Research* **283**, 217-234.

[20] Kubacka, H., M. Slavkovik and J-J. Rückman (2020), Predicting the winners of Borda, Kemeny and Dodgson elections with supervised machine learning, *Proceedings of the 17th European Workshop on Multiagent systems*, forthcoming, `http://slavkovik.com/eumasCR.pdfaccessed:08/15/2020`.

[21] Laslier, J.F. (2011), And the loser is ... plurality voting, in: D.S. Felsenthal and M. Machover (eds.), *Electoral systems – Paradoxes, assumptions, and procedures*, Springer, Heidelberg.

[22] Leshno, M., D. Moller and P. Ein-Dor (2002), Neural nets in a group decision process, *International Journal of Game Theory* **31**, 447-467.

[23] Marshland, S. (2009), *Machine Learning: An Algorithmic Perspective*, Chapman & Hall, CRC Press, Boca Raton, Florida, USA.

[24] McCulloch, W.S. and W. Pitts (1943), A logical calculus of ideas imminent in nervous activity, *Bulletin of Mathematical Biophysics* **5**, 115-133.

[25] McLean, I. and A. Urken (1995), *Classics of social choice*, University of Michigan Press, Ann Arbor.

[26] McNelis, P.D. (2005), *Neural Networks in Finance*, Academic Press, Boston.

[27] Nehring, K. and M. Pivato (2018), The Median Rule in Judgement Aggregation, *preprint*, http://www.parisschoolofeconomics.eu/docs/ydepot/semin/texte 1112/KLA2012MAJ.pdf.

[28] Procaccia, A.D., A. Zohar, Y. Peleg and J.S. Rosenschein (2009), The learnability of voting rules, *Artificial Intelligence* **173**, 1133-1149.

[29] Pukelsheim, F. (2003), Social choice: The historical record, in: S. Garfunkel, Consortium for Mathematics and its applications (eds.), *For all practical purposes – Mathematical literacy in today's world (sixth edition)*, Freeman, New York.

[30] Richards, W., H.S. Seung and G. Pickard (2006), Neural voting machine, *Neural Networks* **19**, 1161-1167.

[31] Risse, M. (2005), Why the count de Borda cannot beat the Marquis de Condorcet, *Social Choice and Welfare* **25**, 95-113.

[32] Rumelhart, D.E., G.E. Hinton and R.J. Williams (1986), Learning Internal Representations by Error Propagation, in: D.E. Rumelhart, J.L. McClelland, and the PDP research group (eds.), *Parallel distributed processing: Explorations in the microstructure of cognition*, Volume 1: Foundations. MIT Press, Cambridge MA.

[33] Sgroi, D. and Zizzo, D.J. (2009): Learning to play 3×3 games: Neural networks as bounded-rational players, *Journal of Economic Behavior & Organization* **69**, 27-38.

[34] Silver, D., A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis (2016), Mastering the game of Go with deep neural networks and tree search, *Nature* **529**, 484-489.

[35] Saari, D.G. (2000), Mathematical structure of voting paradoxes II: Positional voting, *Economic Theory* **15**, 55-101.

[36] Smith, J.H. (1973), Aggregation of preferences with variable electorate, *Econometrica* **41**, 1027-1041.

[37] Young, H.P. (1974), An axiomatization of Borda's rule, *Journal of Economic Theory* **9**, 43-52.

[38] Young, H.P. (1975), Social choice scarring rules, *SIAM Journal of Applied Mathematics* **28**, 824-838.

[39] Young, H.P. and A. Levenglick (1978), A consistent extension of Condorcet's election principle, *SIAM Journal on Applied Mathematics* **35**, 285-300.

Working Paper Series in Economics

recent issues

No. 145 *Dávid Burka, Clemens Puppe, László Szepesváry and Atilla Tasnádi:* Voting: a machine learning approach, November 2020

No. 144 *Guanhao Li, Clemens Puppe and Arkadii Slinko:* Towards a classification of maximal peak-pit Condorcet domains, September 2020

No. 143 *Andranik S. Tangian:* Using composite indicators in econometric decision models with application to occupational health, September 2020

No. 142 *Ingrid Ott and Susanne Soretz:* Institutional design and spatial (in)equality – the Janus face of economic integration, August 2020

No. 141 *Laura Reh, Fabian Krüger and Roman Liesenfeld:* Predicting the global minimum variance portfolio, July 2020

No. 140 *Marta Serra Garcia and Nora Szech:* Understanding demand for COVID-19 antibody testing, May 2020

No. 139 *Fabian Krüger and Lora Pavlova:* Quantifying subjective uncertainty in survey expectations, March 2020

No. 138 *Michael Müller and Clemens Puppe:* Strategy-proofness and responsiveness imply minimal participation, January 2020

No. 137 *Andranik S. Tangian:* Tackling the Bundestag growth by introducing fraction-valued votes, October 2019

No. 136 *Susanne Fuchs-Seliger:* Structures of rational behavior in economics, September 2019

No. 135 *Cornelia Gremm, David Bälz, Chris Corbo and Kay Mitusch:* Intermodal competition between intercity buses and trains – A theoretical model, September 2019