One-Against-All Halfplane Dichotomies

George Nagy(✉) and Mukkai Krishnamoorthy

Rensselaer Polytechnic Institute, Troy, N.Y. 12180, USA
nagy@ecse.rpi.edu, moorthy@cs.rpi.edu

Abstract. Given M vectors in N-dimensional attribute space, it is much easier to find M hyperplanes that separate each of the vectors from all the others than to solve M arbitrary linear dichotomies with approximately equal class memberships. An explanation of the rapid growth with M and N of the number of separable one-against-all linear halfplane dichotomies is proposed in terms of convex polyhedra in a hyperspherical shell. The counterintuitive surge is illustrated by averaged results on pseudo-random integer arrays obtained by Linear Programming and Neural Networks. Although the initial motivation arose from seemingly arbitrary rankings of scientists and universities, this project is not directed at any application.

Keywords: Linear separability · Halfplane dichotomy · Ranking

1 Introduction

We consider datasets of M homogenous patterns, i.e., sets of observations on objects that do not naturally fall into two or more categories. Each pattern is, as usual, represented by a feature vector of N elements. Examples of such datasets include census records, fact sheets for countries, cities, schools, universities or hospitals, and collections of publication data for scientists. In the small collections we have in mind, M may range from 10 to 10^6, and N from 1 to 100.

Our objectives are to explore, theoretically and experimentally, (1) why so many of the M patterns can be linearly separated from all the others when $M >> N > 3$; (2) equivalently, why most patterns can be ranked first among all the patterns by a linear weighting function; (3) whether feature difference vectors facilitate finding the weight vectors for such one-against-all dichotomies; and (4) how the required weight vectors can be obtained with either linear programming or a perceptron-type neural network. We introduced some of these issues in an ICPR 2022 submission [1].

There are, of course, other ways to analyze pattern matrices beside linear separation or ranking, including descriptive statistics, higher moments of the empirical distributions, and classification with various degrees and types of supervision. Data may be analyzed to reveal groups or clusters that are more like each other in some respect than like other groups. The expectation in such exercises is that each class is grouped somewhat compactly in feature space, with every class bounded by non-intersecting linear or nonlinear surfaces. We are, however, interested only in how many patterns in a given

A. Krzyzak et al. (Eds.): S+SSPR 2022, LNCS 13813, pp. 183–192, 2022.
https://doi.org/10.1007/978-3-031-23028-8_19

set can be linearly separated from all the others. We believe that the answer is counter-intuitive because of the difference of volume-surface relations in hyperspace from our experience in the 3-D world.

In the remainder of this paper, we present a brief literature review (Sect. 2); examine the prevalence of one-against-all halfplane dichotomies in pseudo-random arrays (Sect. 3); relate the number of such dichotomies to surface-to-volume ratios in hyperspace (Sect. 4), comment on the merits of difference vectors (Sect. 5), compare Linear Programming and Neural Network solutions (Sect. 6); return briefly to MeFirst ranking (Sect. 7); and summarize our putative contributions (Sect. 8).

2 Prior Work

Most of this work is based on theory established at least 50 years ago. Our terminology and notation mirrors those of venerable textbooks on pattern recognition like DHS, Fukunaga, K&K, and Bishop [2–5]. These include material about weight vectors and separating planes in hyperspace, feature normalization and column augmentation, and some aspects of dynamic and linear programming and of single-layer neural networks for which we feel compelled to cite primary sources.

One-class classification (OCC), typically encountered in anomaly detection or separation of signal from noise, is really a two-class discriminations where one class (normal, signal, inlier) is well defined by one or more clusters of samples, and the other class (abnormal, anomalous, noise, outlier), usually with far fewer samples, does not exhibit any compactness characteristic in feature space [3]. Any pattern far enough from the normal class (according to some metric) falls into this abnormal class. One-shot learning is different: here it is desired to improve the classifier after seeing each new labeled sample [6].

Multi-category classification can be reduced to either a large binary discrimination (via Kessler's Construction [7]), or to a set of dichotomies with the same number of features, as is customary for Support Vector Machines. We mention these paradigms because they are easy to confuse with our main subject of one-against-all dichotomies.

The ease of one-against-all linear separation may be contrasted with Hughes' demonstration that the fraction of linearly separable pairs of sample sets (i.e., *halfplane dichotomies*) of M points in general positions in N-dimensional space diminishes rapidly as M exceeds $2N + 1$ (the *capacity* of a hyperplane) [8].

Hilbert's and Coxeter's illustrations of 4-dimensional polyhedra offer a gentle introduction to hyperspace [9, 10]. Formulas for the volume and surface area of an n-sphere are listed in [11]. Bishop provides a good explanation of why with rising dimensionality samples are increasingly concentrated in a thin shell [5]. Kernel methods map nonlinear boundaries into hyperplanes in higher dimensions [12].

Although Linear Programming (LP) is designed to optimize a linear function subject to a set of linear constraints [13, 14], we used it only to obtain weights for linear separation. Both the default Interior Point algorithm [15] built into MATLAB and into open-source GNU Octave, and the Dual Simplex algorithm [16], find a solution, if one exists, in polynomial time, and halt if the constraints cannot be satisfied.

Neural networks (NNs) have also been used to determine linear separability. The original "fixed-increment perceptron error-correction" procedure provably converges to

a solution if one exists [17, 18]. Replacing the Heaviside step activation by a differentiable function led to gradient descent methods that accelerate convergence. However, the bound on the number of iterations can still be computed only from a known solution. It is impossible to tell whether the weight vector is still approaching the solution cone or the input data is linearly inseparable. This holds also for other algorithms for training a single layer. Therefore the maximum allowed number of epochs or training cycles must be specified. Unlike LP, a single-layer neural network can never confirm linear inseparability. We programmed an elementary α-perceptron according to the 1966 perceptron software manual [19], and ran a single-layer feed-forward network configured for gradient descent from the MATLAB Deep Learning toolbox [20].

The wide-ranging and often harmful effects of ranking human endeavor are critically examined in [21] and [22]. OpenAI forbids such use of its software [23].

3 One-Against-All Halfplane Dichotomies

We confine our attention to classifying each pattern in the dataset against every other pattern. For M patterns, we "train" M classifiers to perform M binary classifications. Each classifier is trained on all M patterns, but with a different pattern singled out for the positive class. Since each of the M classifiers knows the identity of the distinguished pattern, it could achieve 100% correct performance via table look-up. However, we seek only to isolate each pattern by a linear weighting of its features.

Since we don't have any test set (which dispenses us from the vexatious concern for generalization), how do we measure performance? Our performance metric for the M classifiers is the fraction of linearly separable dichotomies. Suppose, for example, that $M = 200$. If an algorithms finds 180 weight vectors such that each separates one pattern from the remaining 199, then the metric is 0.9.

Let us look at a two-dimensional example where we can plot both the patterns and the weight vectors. If one had to find seven one-against all linearly separable vectors in two-space, they would have to lie on the vertices of a convex polygon, like the four different sets of 7 points in Fig. 1. (Only one of the polygons – for the + set – is traced explicitly in the Figure.) Any points inside the polygon would not be separable from all the other points. Each point set gives rise to 7 linear dichotomies. Only the line separating one of the +'s from the other six is shown. Note that most of the points fall in an annulus (or *shell* in higher N). The next section suggests that the prevalence of one-against all linear separability is due to the increasing concentration of points in the shell.

For a fixed N, the number Fsep(M, N) of one-against-all halfplane dichotomies is a rising function of the number of patterns M that flattens out when M is large enough. We denote its asymptotic vale as M^. The function depends critically on the number of attributes N (i.e., the dimensionality of the feature space). The values of the function change by orders of magnitude, but its general behavior remains the same.

We believe that our experimental exploration of the parameter space for N up to 12 and M up to 10,000 reveals most of the interesting behavior of this function. We conducted our experiments on $M \times N$ arrays of pseudo-random integers (randi(R, M, N)) with $R = 1000$. (To keep coincidences low enough to avoid affecting the overall behavior, the range R must satisfy $R^N >> M$.) Depending on the variability

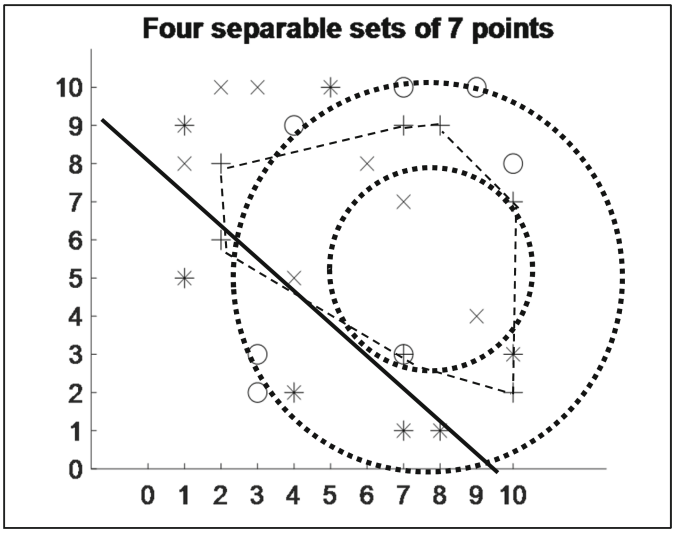

Fig. 1. Four sets of one-against-all linearly separable points at the vertices of convex polygons. One of the polygons is shown with dashed lines, The solid black line separates one of the +s from the other six +s These four (○, +, *, ×) were the only separable sets of 7 points generated by 200 pseudo-random trials with a 1–10 range of integers. The dotted circles bound an annulus that contains most of the points.

of the observations, we ran each setting with 10 or 100 trials and recorded the average Fsep and its standard deviation. For exploring the behavior of Fsep for larger values of M, we sampled only every 5th, 10th, 100th or 1000th M, as shown in the figures. Even so, several of the experiments ran for over ten hours on our vintage 2.4 GHz Dell Optiplex. The run-time is roughly proportional to $T \times M^3$, where T is the number of trials. In Sect. 6, we tabulate some results, including timing, on small and large attribute arrays.

Initially, Fsep(M, N) = M for any N, because in N-space, any N + 1 points in general positions are linearly separable. In 3-D, these points form the vertices of a tetrahedron, but in higher dimensions they are difficult to visualize. The linear growth of Fsep with M extends rapidly with N. With N = 2, the increase moderates as soon as M = 5 and the curve is almost flat by M = 100 (Fig. 2). With N = 5, the slope is still about 0.75 at M = 100 (Fig. 3). But with N = 12, the fraction of linearly separable dichotomies is 90% even at M = 6400 (Fig. 4). We lack the computer resources to explore it further. For a fixed M as a function of N, Fsep necessarily plateaus at Fsep(M, N) = M after a linear rise (Fig. 5).

4 A Geometric Perspective

Although an infinite number of distinct points located on the surface of an N-dimensional sphere are one-against-all separable, their number reaches a limiting value in any realistic scenario where the points are subject to perturbation. Both this asymptotic magnitude

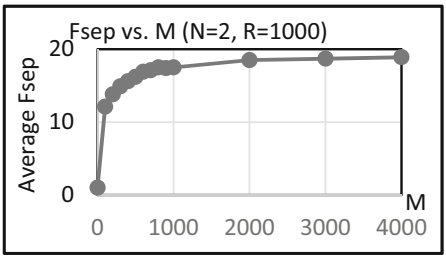

Fig. 2. Fsep, the number of separable patterns, vs. M, with N = 2

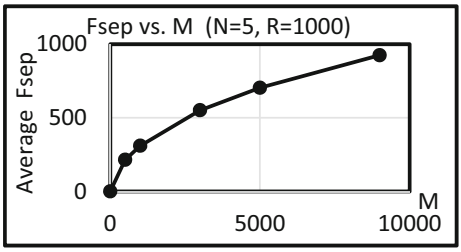

Fig. 3. Fsep, the number of separable patterns, vs. M, with N = 5

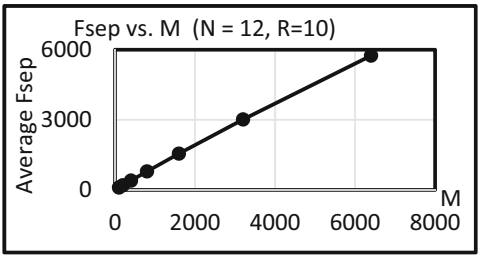

Fig. 4. Fsep, the number of separable patterns, vs. M, with N = 12

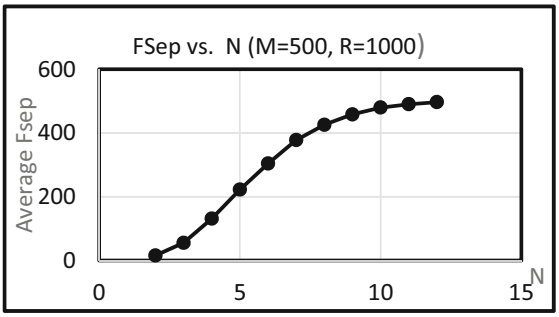

Fig. 5. Increase in the number of one-against-all separable dichotomies (FSep) with N

M^ and the value of M where it prevails increase rapidly with N. A possible cause is the following. The ratio of the volume of a shell of thickness ε to that of a unit hypersphere is $1\text{-}(1\text{-}\varepsilon)^N$. If, for example, N = 5 and ε = 0.2 (a shell of thickness equal to 20% of the radius), then the volume of the shell is 67% of the volume of the sphere. Therefore most of the randomly generated points would fall in the shell, as already suggested by Fig. 1 (where N is only 2). They are all separable only if they constitute the vertices of a convex polyhedron.

For an additional point to be separable from the rest without altering the convexity of any existing vertex, it would have to fall near the center of one of the faces of the polyhedron, as suggested by the 2-D example in Fig. 6. Since every new separable point decreases the area of the facets, the space available for new separable pints approaches zero at a rate decreasing with M. For any M, with large-enough N it is possible to generate sets of M points in general positions that are one-against-all linearly separable. We believe that these notions generalize to N dimensions and look forward to suggestions from the Workshop participants on turning arguments into a proofs.

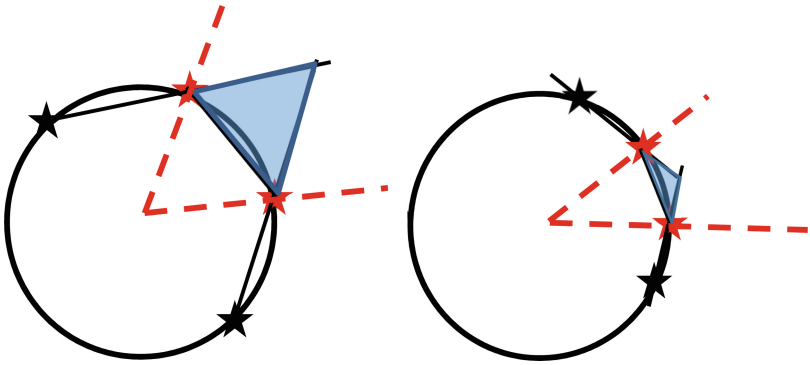

Fig. 6. How near is near enough? Suppose that there are already 4 point located on a circle (hypersphere). Where in the red sector can we add a new point? The space available for a new point shrinks as the points in the shell get closer to each other. Space available for new separable points in the red sector of the convex hull is shown in blue. (Color figure online)

5 Pairwise Attribute Difference Vectors

We must at this point introduce some notation. Our data vector is the M × N array A. Let $\mathbf{a}_k$, k = 1 to M, be the row vectors of A. Let $\mathbf{w}_k$ be the M column vectors of the N × M weight matrix W. The task is to find M weight vectors $\mathbf{w}_k$ such that $\mathbf{a}_{k*}\mathbf{w}_k > \mathbf{a}_{k*}\mathbf{w}_k$ for all for all $k \neq k^*$. In other words, we must solve M one-against all dichotomies.

Stated in terms of *pairwise difference vectors* $\mathbf{d}_{k^*,k} = \mathbf{a}_{k^*} - \mathbf{a}_k$, $\mathbf{d}_{k^*,k} > 0$ for all $k \neq k^*$. The difference vectors define homogenous halfspace dichotomies as illustrated in Fig. 7. The constraints for a linear programming solution are naturally framed in terms of difference vectors. Our experiments indicate that difference vectors also lead to faster convergence of single-layer neural networks. A possible reason is that in the direct solution, the distinguished vector $\mathbf{a}_{k^*}$ can change the weight vector only once per epoch. In the proposed alternative, every vector plays an equal role in training.

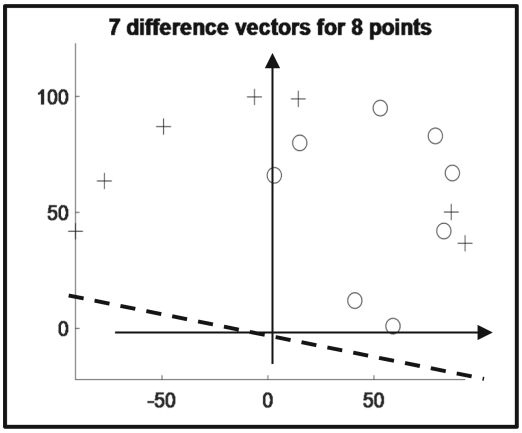

Fig. 7. Normalized pairwise difference vectors (+) for 8 one-against-all separable attribute vectors (o) form a homogenous halfplane dichotomy, bounded by the dashed line. For better visibility, the unit-length difference vectors were scaled (multiplied) by 100.

6 Linear Programming and Neural Networks

To find the weights of the separating vector, we used the difference vectors as the required constraints and minimized the sum of the weights as the objective function. We found equivalent, but not identical, solutions with the MATLAB default Interior Point algorithm and the Dual Simplex algorithm. The LP program was run for each of the M one-against-all dichotomies of the pseudo-random M $\times$ N integer arrays folded into a loop for the specified number of trials (10 or 100).

Because the networks in the MATLAB Deep Learning toolbox have so many invisible built-in functions (including their *Legacy Perceptron*), we programmed a classical α-perceptron in m-code. As shown below, it required far fewer iterations with the input preprocessed into pairwise difference vectors than directly on the rows of the attribute array. We report only the fastest configuration of the Alpha Perceptron, with dimensionality augmentation after normalization. As proved in [1], the weights obtained with such preprocessing also separate the original array.

We also experimented with a single layer Levenberg–Marquardt feedforward network. Since the feedforward net has a smooth (*hyperbolic tangent sigmoid*) activation function, we rounded its output to 0 or 1. We set the performance criterion to *least mean squares*. After missing a few dichotomies with MaxEpochs = 10, this net reached the correct value of Fsep = 496 on the 500 $\times$ 12 arrays.

The general trend of the performance comparisons are shown in Table 1. In every experiment, we checked the output weights directly on the original array. The runtimes reported were obtained with *tic-toc*, which is not immune to Windows background activity. Our objectives for the experiments were only to show that one-against-all dichotomies are easier for both LP and NN than arbitrary divisions of M patterns into approximately equal classes, and that such dichotomies benefit from pairwise difference input.

Table 1. Performance of LP and NN on 8 × 2 and 500 × 12 pseudo-random attribute arrays.

Method	M	N	MaxEpochs	Trials	Fsep	Time (s)
LP	8	2	N/A	100	5.5	7
Alpha Perceptron (diffs)	8	2	100	100	5.5	0.3
Alpha Perceptron (rows)	8	2	100,000	100	.4.9	239
Levenberg–Marquardt	8	2	10	100	5.4	7
LP	500	12	N/A	10	496	63
Alpha Perceptron (diffs)	500	12	100	10	496	9
Levenberg–Marquardt	500	12	10	10	496	564

7 Ranking

What originally led us to one-against-all dichotomies was MeFirst Ranking. A ranking of 100,000 scientists with their citation attributes was published in October 2020 in PLoS BIOLOGY [24, 25]. We wanted to demonstrate the existence of linear weighting functions that would allow most of these scientists to claim first rank in a large subgroup (defined, for example, by institution or nationality). In [1], we explored that dataset, and another of the 1000 "top" universities [26]. Rather than repeating those results here, we observe only that while the relationship of the number of separable entries to M and N, Fsep(M, N), was much the same as what we found on pseudo-random integer arrays, separating the real data took more work. Although for M = 500 and N = 12 both sets yield Fsep = 496, it takes the Alpha Perceptron 1,000,000 instead of 100 epochs to complete the task. The LP, however, shows no increase in runtime.

Table 2 is a small example with small integer coefficients that shows clearly the equivalence of linear separability and ranking. It also illustrates the method we used in all our experiments to verify the weights of the separating vector. We note that if ties for first place area not considered admissible, an additional check is required. With the range of integer arrays used in our simulations (1000), coincidences among the 1000^2 (N = 2) or 1000^{12} (for N = 12) possible entries are rare.

Because attribute vector normalization (usually to unity) and column augmentation (via the addition of a constant attribute and a threshold weight) are often used in statistical and neural network classification, in [1] we gave sufficient and necessary conditions for MeFirst ranking and one-against-all separability in terms of normalized or augmented attribute vectors. We close with an informal statement of sufficient and necessary conditions in geometric terms.

Proposition: *Each of a set of M points in N-space are one-against-all separable from the remaining M-1 points if and only if each point constitutes a distinct convex vertex of the convex hull of all M points.*

Note that points on the edges or faces, which are by convention considered part of the convex hull, are excluded, as are coincident points.

Table 2. The highest value of each weighted attribute vector is on the diagonal of the product matrix. For example, the product of (1, 8, 8) × (0, 1, 1) is the highest value (16) in the fifth column.

Attributes A			Weights W^T			Weighted sum of Attributes A × W						
9	1	1	1	0	0	9	1	1	10	2	10	11
1	9	1	0	1	0	1	9	1	10	10	2	11
1	1	9	0	0	1	1	1	9	2	10	10	11
8	8	1	1	1	0	8	8	1	16	9	9	17
1	8	8	0	1	1	1	8	8	9	**16**	9	17
8	1	8	1	0	1	8	1	8	9	9	16	17
6	6	6	1	1	1	6	6	6	12	12	12	18

8 Summary

This paper presents a problem that we believe has received little attention. We investigated the behavior of one-against-all halfplane dichotomies of pseudo-random integer arrays and found that it is very different from that of similar dichotomies with approximately balanced populations. We showed that for a limited range of M patterns in N-dimensional feature space, linear separability can be determined with either Linear Programming or single-layer Neural Networks. In contrast to balanced populations, the preferred input for both methods is an array of pairwise attribute difference vectors.

We found that one-against-all linear separability in any dimensionality N increases rapidly with M, with the slope changing gradually from unity to zero. With fixed M, the number of linearly separable halfplane dichotomies increases monotonically with N from N + 1 to its maximum of M.

We suggested that this surprising behavior is due to the concentration of high-dimensional patterns in a hyperspherical shell where they form the vertices of a convex polyhedron. Convex vertices are linearly separable from all other points regardless of whether these are on the surface or inside the entire point set.

References

1. Nagy, G., Krishnamoorthy, M.: MeFirst ranking and multiple dichotomies via linear programming and neural networks. In: ICPR 2022 (2022). Accepted
2. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. Wiley, New York (2001)
3. Fukunaga, K.: Statistical Pattern Recognition, 2nd edn. Academic Press, Cambridge (1990)
4. Theodoridis, S., Koutroumbas, K.: Pattern Recognition. Academic Press, Cambridge (2009)
5. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer, New York (2006). ISBN 978-0387-31073-2
6. Li, F.-F., Fergus, R., Perona, P.: One-shot learning of object categories. IEEE Trans. Pattern Anal. Mach. Intell. **28**(4), 594–611 (2006)
7. Nilsson, N.J.: Learning Machines: The Foundations of Trainable Pattern-Classifying Systems. McGraw-Hill, New York (1965)

8. Hughes, G.F.: On the mean accuracy of statistical pattern recognizers. IEEE Trans. Inf. Theory IT **4**(1), 55–63 (1968)
9. Hilbert, D., Cohn-Vossen, S.: Geometry and the Imagination, Chap. III. Chelsea Publishing, Hartford (1952)
10. Coxeter, H.S.M.: Introduction to Geometry, Chap. 22. Wiley, Hoboken (1989)
11. Woldfram MathWorld, Hypersphere. https://mathworld.wolfram.com/Hypersphere.html
12. Shawe-Taylor, J., Cristianini, N.: Kernel Methods for Pattern Analysis. Cambridge University Press, Cambridge (2004)
13. Dantzig, G.B.: Maximization of a linear function of variables subject to linear inequalities. In: Koopmans, T.C. (ed.) Activity Analysis of Production and Allocation. Wiley & Chapman-Hall (1951)
14. Gass, S.L.: Linear Programming, 5th edn. Dover, Downers Grove (2003)
15. Karmarkar, N.: A new polynomial-time algorithm for linear programming. In: Proceedings of the Sixteenth Annual ACM Symposium on Theory of Computing, STOC 1984, p. 302 (1984)
16. Nabli, H.: An overview on the simplex algorithm. Appl. Math. Comput. **210**(2), 479–489 (2009)
17. Rosenblatt, F.: The perceptron: a probabilistic model for information storage and organization in the brain. Psychol. Rev. **65**, 386 (1958)
18. Rosenblatt, F.: Principles of Neurodynamics. Spartan Books, New York (1962)
19. Barker, T.: A Computer Program for Simulation of Perceptrons and Similar Neural Networks: User's Manual. CSRP Report #8. Cornell University, Ithaca, NY (1966)
20. MATLAB: Deep Learning Toolbox. https://www.mathworks.com/products/deep-learning.html. Accessed 12 Jan 2021
21. Muller, J.Z.: The Tyranny of Metrics. Princeton University Press, Princeton (2018)
22. Biangioli, M., Lippman, A.: Gaming the Metrics. MIT Press, Cambridge (2020)
23. Johnson, S.: The Writing on the Wall, The New York Times Magazine, 17 April 2022 (2022)
24. Ioannidis, J.P.A., Boyack, K.W., Baas, J.: Updated science-wide author databases of standardized citation indicators. PLoS Biol. **18**(10), e3000918 (2020). https://doi.org/10.1371/journal.pbio.3000918
25. Mendelay: Data for "Updated science-wide author databases of standardized citation indicators". https://data.mendeley.com/datasets/btchxktzyw/2. Accessed 10 Apr 2021
26. QS World University Ratings. https://group.intesasanpaolo.com/content/dam/portalgroup/nuove-immagini/sociale/2022_QS_World_University_Rankings_1.pdf. Accessed 30 Nov 2021