2022 26th International Conference on Pattern Recognition (ICPR)
August 21-25, 2022, Montréal, Québec, Canada

MeFirst ranking and multiple dichotomies

via linear programming and neural networks

George Nagy
Electrical, Computer, and Systems Engineering Department
Rensselaer Polytechnic Institute
Troy, NY, USA
georgenagy@IEEE.org

Mukkai Krishnamoorthy
Department of Computer Science
Rensselaer Polytechnic Institute
Troy, NY, USA
mskmoorthy@gmail.com

2022 26th International Conference on Pattern Recognition (ICPR) | 978-1-6654-9062-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICPR56361.2022.9956323

***Abstract*— Individuals, institutions and even cities and countries are often ranked according to some linear weighting of their attributes. Under commonly prevailing conditions, it is possible to find weights that give top rank to most arbitrarily designated entries. The number of entries may exceed the number of attributes by orders of magnitude. Necessary and sufficient conditions on the subject-attribute matrix are derived in terms of one-against-all halfplane dichotomies and convex hulls. Pairwise attribute difference vectors are more effective than attribute vectors for one-against-all classification. Comparisons of MeFirst algorithms based on neural networks and on linear programming (LP), on datasets drawn from published rankings of scientists and universities, show that on these tasks, LP is significantly faster.**

Keywords*— *normalization; dimensional augmentation; halfspace dichotomy; unbalanced classes

I. INTRODUCTION

Given a list of M items, each represented by N attributes, it is often possible to find a set of weights for ranking an arbitrarily selected item such that the weighted sum of the attributes places it on top of the list. We define an item as MeFirst Eligible if such a set of weights exists. With these weights, the weighted sum of the selected item must be larger than the weighted sum of any other item. Thus, determining that an item is MeFirst eligible can be reduced to a one-against-all linear separability problem.

Checking linear separability can, in turn, be formulated either as a search for a feasible solution in a linear programming (LP) framework, or as error-free binary classification by a single-layer neural network (NN). Normalization of the attribute vectors to unit length and conversion to halfplane classification (with a hyperplane through the origin) by dimension augmentation are often advocated to simplify and accelerate classification. We will show why such preprocessing may preclude a solution.

Which M items with N attributes are MeFirst eligible? In two dimensions (N=2), any 3 points (M=3) in general positions (i.e., neither coincident nor collinear) are MeFirst eligible. More generally, if the M points are located on distinct vertices of a convex polygon, then they are all eligible. If, however, m points are located inside the convex hull of the remaining M-m points, then these m points are not eligible. Below, we extend this argument to N>>2 and explain the surprising number of MeFirst eligible entries engendered by relatively few attributes in terms of known geometric properties of hyperspace.

In most examples of binary classification – halfplane or other – the numbers of samples in either class are of the same order of magnitude. However, *skew* often arises in biomedical image processing and in anomaly detection. Unbalanced or skewed class membership hampers both statistical and neural classifiers. One-against-all is the extreme case of skew.

A major contribution of this paper is the theoretical and experimental demonstration that the ill effects of skewed classification can be alleviated by half-plane classification of normalized pairwise difference vectors. A further contribution is a formal proof of necessary and sufficient conditions for MeFirst ranking. On the experimental side, we compare two LP and three NN solutions on sizeable published datasets. Although the thin-shell phenomenon of hyperspace geometry has been examined by others in the context of classification, we believe that we are the first to show its singular effect on one-against-all dichotomies.

We were prompted to examine these relationships by the recent compilation of the publication attributes and consequent rank-ordering of over 100,000 scientists. The ranking in the Citation Database is one of many possible rankings based on homogeneous linear weighting of the scientists' attributes. The ranking problem led us to one-against-all halfspace dichotomies of pairwise difference vectors and to conditions for their solution by linear programming and single-layer neural networks. Aside from connecting ranking and classification, our results bear on skewed dichotomies from other sources.

A. Terminology

We will use the following terminology. A *dichotomy* is any separation, linear or not, of samples into two classes. *Halfspace classificat*ion means separation by a hyperplane. *Unitomy* is separation by a hyperplane through the origin. Negating the samples in the "negative" class in a unitomy flips them to the positive side of the hyperplane, thereby locating all the samples in the same halfspace (cf. Fig. 1). Classification via unitomy does not need ground truth for counting errors, because every sample that yields a negative weighted sum is an error.

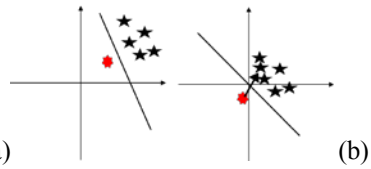

Fig. 1. (a))ne-against-all dichotomy. (b) One-against-all unitomy. A unitomy can be generated from a linear dichotomy by dimensional augmentation.

We define MeFirst ranking of M items according to N attributes as follows. Let A be the M×N *attribute matrix* with rows $\mathbf{a}_k$ of attributes of item s_k. and let W be the N×M *weight matrix* with columns $\mathbf{w}_k = [w_{k,1}, w_{k,2}, \ldots, w_{k,N}]$. Then item s_{k*} is *MeFirst Eligible* if and only if the highest value of the k^{*th} column of A×W is on the diagonal. The solution weights are not unique. In each ranking, we focus only on first-place and let the other items fall where they may. The toy example of Table 1, with M = 7 and N = 3, illustrates this property. *Attributes* can be called *features* or *coordinates*, *weights* are also *coefficients,* and *classes* can be *categories* or *labels*.

B. Outline

In Section II we give examples of MeFirst ranking based on weighted attributes and of other ways of ordering items. In Section III we show that MeFirst Ranking is equivalent to a set of unitomies of pairwise difference vectors. In Section IV we prove a theorem about row normalization and column augmentation of the array of rankable entries and state necessary and sufficient conditions for the existence of a solution. In Section V we propose a geometric explanation of why the number of MeFirst eligible entries vastly exceeds hyperplane capacity. Sections VI describes LP and "perceptron" configurations for MeFirst ranking. Section VII presents experimental support for the above claims and observations. We summarize our results in the concluding section and suggest their applicability to other domains where unbalanced classification is the rule rather than the exception.

Ranking and voting methods are social technologies. Concepts of linear separability, hyperplane capacity, classification by neural networks, and the asymptotic behavior of the convex hull of point sets can be traced back to the early days of pattern recognition, but are rarely discussed in the context of alternative rankings of attribute vectors. Linear separability issues were first raised by threshold functions. The geometric properties of high-dimensional spaces are part of computational geometry. Linear programing and machine learning belong to our own community. Given this diversity, we provide references' to earlier work as we develop the connections instead of a dedicated literature review.

II. Ranking

Just about everything is ranked: Fortune 500 companies according to their valuation, earnings, and number of employees; politicians by their votes or campaign funding; pitchers by ERA, IFIP and WHIP; countries by population, area, GDP and life expectancy; and scientists by various metrics like publication count and h-index. Ranking requires sorting the entries according to the values of an attribute or of some combination of attributes. Reaching the top rank often brings fame, money or other benefits.

A. MeFirst Ranking

We explore when and why weights can be assigned to N attributes in such a way that an arbitrarily chosen entry (scientist or journal or restaurant) ranks first, according to the weighted sum of the attribute values, among M (M >> N) commensurable entries. Only a fraction of all the entries in the Citation Database are MeFirst Eligible with respect to the entire collection. But in many subsets of entries, based, for example, on institutional affiliation, every scientist in the chosen group can claim top rank according to his or her weighting of the published attributes. Even in much larger groups with thousands of candidates, most can claim first rank with a suitable set of weights.

B. Other Types of Ranking

Electoral voting systems also induce rankings, but they don't use linear combinations of attributes. Major systems that may yield different winners include plurality, ranked-choice, approval, and positional voting like the Borda Count [1]. Questions about the outcome usually center on the eligibility of voters and on the validity of their votes [2]. Most political elections rank only a handful of candidates. A trifecta bet on a horserace attempts to rank the top three horses. The PageRank algorithm [3] conspicuously rank-orders billions of websites. The wide-ranging and often harmful effects of ranking human endeavor are critically examined in [4] and [5].

Our interest here is only the connection between ranking by weighted attributes and classification algorithms. More specifically, we will reduce MeFirst ranking to a set of binary classifications and propose solutions that differ from those commonly advocated for classical dichotomies.

TABLE I. Algorithmic solution for the weight matrix W. The highest value in each column of A×W is on its diagonal

Attributes A			Weights W^T			Weighted sum of Attributes A x W						
9	1	1	0.23	-0.11	-0.11	1.85	-0.87	-0.87	1.34	-2.58	1.34	11.00
1	9	1	-0.11	0.23	-0.11	-0.87	1.85	-0.87	1.34	1.34	-2.58	11.00
1	1	9	-0.11	-0.11	0.23	-0.87	-0.87	1.85	-2.58	1.34	1.34	11.00
8	8	1	0.17	0.17	-0.32	0.85	0.85	-1.53	2.34	-1.09	-1.09	17.00
1	8	8	-0.32	0.17	0.17	-1.53	0.85	0.85	-1.09	2.34	-1.09	17.00
8	1	8	0.17	-0.32	0.17	0.85	-1.53	0.85	-1.09	-1.09	2.34	17.00
6	6	6	1.00	1.00	1.00	0.06	0.06	0.06	0.06	0.06	0.06	18.00

Some of our observations and experiments on real data were suggested by results on randomly generated attribute arrays with controlled properties.

III. CLASSIFICATION AND RANKING

Standard text books on pattern recognition and machine learning start with an explanation of two-category and multi-category classification. Most dichotomies discussed in the literature divide M patterns into two sets of the same order of magnitude, but imbalanced (*skewed*) classes arise in applications where samples of either class are rare. Some texts also expand on *single-category classification*, which is the separation of signal from noise, or of normal samples from outliers [6]. The classification of attribute vectors $\mathbf{a}_k$ into two classes can be accomplished with a single-layer network with weights $\mathbf{w}$ and a scalar threshold function.

In N-dimensional space, a weight vector $\mathbf{w}$ is perpendicular to a hyperplane located at distance $\|\mathbf{w}\|$ from the origin. Adding a constant element (say 1) to the attribute vectors and a bias (or *threshold*) weight w_0 raises the dimensionality of the attribute space to N+1 (this is *dimensional* augmentation, not *data* augmentation). In this space, a separating hyperplane passes through the origin, thereby giving rise to a unitomy [7, 8]. More arcane aspects of halfspace classification are discussed in [9, 10, 11, 12].

A. Multi-category to Binary classification

Kessler's Construction replaces a C-category, N-attribute multi-category problem by a single dichotomy with (C-1) times as many C×N-dimensional attribute (or *feature*) vectors. Each new attribute vector contains one of the original attribute vectors, its negative, and zeroes everywhere else [13].

Some classifiers, like Support Vector Machines, reduce multi-class problems to multiple one-against-all tasks. But in the SVM paradigm the number of dichotomies increases only with the number of classes rather than the number of patterns, which is typically orders of magnitude higher. Furthermore, linear separability is not necessarily the most appropriate categorization criterion for garden-variety classification tasks.

B. Multiple Dichotomies via Pairwise Difference Vctors

In contrast to the traditional dichotomy with many patterns on both sides, MeFirst ranking of M items requires solving M distinct dichotomies, each of which separates a selected item from all the others (the extreme case of skewed classification). MeFirst Eligibility of item k* requires that the value of the linear weighting function $\mathbf{a}_{k*}\mathbf{w}_{k*}$ be higher than that of any other $\mathbf{a}_k\mathbf{w}_k$, i.e.,

$$a_{k*,1}\, w_{k*,1} + a_{k*,2}\, w_{k*,2} \ldots, a_{k*,N}\, w_{k*,N} > a_{k,1}\, w_{k*,1} + a_{k,2}\, w_{k*,2}, \ldots, a_{k,N}\, w_{k*,N}, \quad \text{for all } k \neq k^*, \text{ or}$$

$$(a_{k*,1} - a_{k,1})\, w_{k*,1} + (a_{k*,2} - a_{k,2})\, w_{k*,2}, \ldots, (a_{k*,N} - a_{k,N})\, w_{k*,N} > 0, \quad \text{for all } k \neq k^*$$

Therefore MeFirst Eligibility is equivalent to one-against-all Halfspace Dichotomy of the attribute vectors, and to Unitomy of the *pairwise difference vectors* $\mathbf{d}_{k*,k} = \mathbf{a}_{k*} - \mathbf{a}_k$.

One-against-all halfspace dichotomies lead to a new approach because in each dichotomy, we can classify the M-1 pairwise difference vectors. This would be onerous for most two-class problems. If, for example, the patterns are partitioned into equal halves, there would be ~$M^2/4$ difference vectors. Furthermore, finding weights that assign top-m ranking to m arbitrary entries is highly unlikely (cf. Section V).

IV. CONDITIONS FOR MEFIRST ELIGIBILITY

We present some lemmas that lead to a theorem about how augmenting the dimensionality by adding a constant to each attribute and, optionally, normalizing the augmented vectors (typically to unit length), affect MeFirst Eligibility via single-layer neural networks.

Let $\mathbf{a}_k = [a_{k,1}, a_{k,2}, \ldots a_{k,N}]$ be a row vector in array A. The row $\mathbf{a}_{k*}$ is selected to be MeFirst Eligible, and $\mathbf{w} = [w_1, w_2, \ldots, w_N]^T$ is a weight vector.

Lemma 1. (*No preprocessing*)
$\mathbf{a}_{k*}$ is MeFirst Eligible
if $\mathbf{w}$ is a solution of $\mathbf{a}_k\,\mathbf{w} < 0$ for $\forall\, k \neq k^*$ and $-\mathbf{a}_{k*}\,\mathbf{w} < 0$

Proof: $\mathbf{a}_{k*}\,\mathbf{w} > 0 > \mathbf{a}_k\,\mathbf{w} \Rightarrow \mathbf{a}_{k*}\,\mathbf{w} > \mathbf{a}_k\,\mathbf{w} \quad \forall\, k \neq k^*$ **QED**

Now let augmented vector $\mathbf{a}^+_k = [1, a_{k,1}, a_{k,2}, \ldots a_{k,N}]$ and $\mathbf{w}^+ = [w_0, w_1, w_2, \ldots . w_N]^T$.

Lemma 2. (*Augmentation*)
$\mathbf{a}_{k*}$ is MeFirst Eligible
iff $\mathbf{w}^+$ is a solution of $\mathbf{a}^+_k\,\mathbf{w}^+ < 0$ for $\forall\, k \neq k^*$ and $-\mathbf{a}^+_{k*}\,\mathbf{w}^+ < 0$

Proof: $\mathbf{a}^+_{k*}\,\mathbf{w}^+ > 0 > \mathbf{a}^+_k\,\mathbf{w}^+ \Leftrightarrow \mathbf{a}_{k*}\,\mathbf{w} + w_0 > \mathbf{a}_k\,\mathbf{w} + w_0 \Leftrightarrow \mathbf{a}_{k*}\,\mathbf{w} > \mathbf{a}_k\,\mathbf{w}$ **for** $\forall\, k \neq k^*$ **QED**

For normalization, let normalized vector $\mathbf{a}^\times_k = \mathbf{a}_k / \|\mathbf{a}_k\|$.

Lemma 3. (*Normalization*)
$\mathbf{a}_{k*}$ is MeFirst Eligible
if $\mathbf{w}$ is a solution of $\mathbf{a}^\times_k\,\mathbf{w} < 0$ for $\forall\, k \neq k^*$ and $-\mathbf{a}^\times_{k*}\,\mathbf{w} < 0$

Proof: : $\mathbf{a}^\times_{k*}\,\mathbf{w} > 0 > \mathbf{a}^\times_k\,\mathbf{w}$,$\Rightarrow (\mathbf{a}_{k*} / \|\mathbf{a}_{k*}\|)\,\mathbf{w} > (\mathbf{a}_k / \|\mathbf{a}_k\|)\,\mathbf{w}$
$\Rightarrow (\mathbf{a}_{k*}\,\mathbf{w}) / \|\mathbf{a}_{k*}\| > (\mathbf{a}_k\,\mathbf{w}) / \|\mathbf{a}_k\|)$
Since $\|\mathbf{a}_{k*}\| > 0$ and $\|\mathbf{a}_k\| > 0$, $\mathbf{a}_{k*}\,\mathbf{w} > \mathbf{a}_k\,\mathbf{w} \quad \forall\, k \neq k^*$ **QED**

Lemma 4. (*Normalization after augmentation*)
$\mathbf{a}_{k*}$ is MeFirst Eligible if $\mathbf{w}$ is a solution of $\mathbf{a}^+_k\,\mathbf{w}^+ / \|\mathbf{a}^+_k\| < 0$
for $\forall\, k \neq k^*$, and $-(\mathbf{a}^+_{k*}\,\mathbf{w}^+ / \|\mathbf{a}^+_{k*}\|)\,\mathbf{w} < 0$.

Proof: : $(\mathbf{a}^+_k\,\mathbf{w}^+ / \|a^+k\|)\,\mathbf{w}^+ > 0 > \mathbf{a}^+_k\,\mathbf{w}^+ / \|a^+k\|$
$\Rightarrow\ \mathbf{a}^+_{k*}\,\mathbf{w}^+ > 0 > \mathbf{a}^+_k\,\mathbf{w}^+ \Rightarrow$ Lemma 2 **QED**

Theorem
Linear separability of a selected row of attributes from all other rows, determined either directly or via augmentation or normalization of the entire array, or by a combination of augmentation followed by normalization, guarantees its MeFirst Eligibility.

N.B. Separation resulting from augmentation *following* normalization does not imply MeFirst Eligibility.

Necessary Condition for MeFirst Eligibility (from Lemma 2):
$\exists\mathbf{w}$ such that $\mathbf{a}^+_{k*}\,\mathbf{w} > \mathbf{a}^+_k\,\mathbf{w} \quad \forall\, k \neq k^*$

Sufficiency Conditions: the premisses of Lemmas 1-4.

Normalization by scaling every input vector to the same length is recommended by some researchers because it tends to accelerate neural network processing [14]. However, the normalized vectors are not linearly separable in the (admittedly rare) case of two original entries lying on the same ray from the origin (Fig. 2). Difference vectors, in contrast, can be freely normalized. In Fig. 3, the original difference vectors of point D (DA, DB, and DC) span more than π radians, but the difference vectors of the normalized attributes span less than π. In N-dimensional space, any hyperplane bounding the halfspace containing these difference vectors induces a MeFirst ranking of the original attribute vectors.

These figures also suggest that the difference vectors facilitate one-against-all classification because they span a larger solid angle than the row vectors. At the risk of derision, we invite comparison to the usefulness of derivatives in optimization problems. The chasm in the effect of normalization between attribute vector and difference-vector classification is one of the major points of this paper. It also suggests why LP algorithms dominate neural network solutions in this type of problem (cf. Section VIII).

Our experimental results (below, in Table IV) confirm he above observations and provide examples of failures due to augmentation *after* normalization

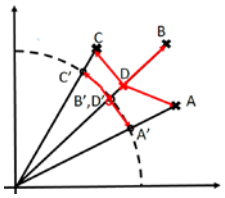

Fig. 2. A disadvantage of attribute length normalization. Attribute D is linearly separable from A, B, and C. After normalization, evwery point lies on a circle of unit radius. D' is n' longer separable from A', B' and C' because it coincides with B'. (M=4, N=2) Such radially collinear vectors are rare in higher-dimensional attribute spaces.

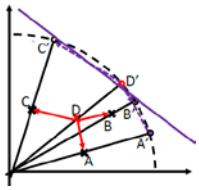

Fig. 3. A benefit of normalization. Attribute D is not linearly separable from A, B, and C. However, after normalization D' is linearly separable from A, B, and C because vectors D'A', D'B', and D'C' span less than π radians. The purple line suggest a possible seprating plane. (M=4, N=2).

V. Linear Separability in High Dimensions

It can be shown that the separable fraction of dichotomies of M points in general positions in N-dimensional space diminishes rapidly as M exceeds 2N+1 (the *capacity* of a hyperplane) [15]. However, point sets generated from uniform multidimensional pseudo-random distributions yield far more one-against-all linearly-separable dichotomies. The Citation Database also yields more separable dichotomies than expected with M >> N > 3. Why? The answer may lie in the geometry of high-dimensional attribute space.

The ratio of the volume of a shell of thickness ε to that of a unit hypersphere is $1-(1-\varepsilon)^N$. The exponential increase with N of the concentration of samples in a thin hyperspherical shell is well known. Bishop relates it to the curse of dimensionality [16] by pointing out that (1) When N>>1, the volume of a thin shell (ε<<1) approaches that of the whole sphere, and (2) Most of the probability mass of a high-dimensional Gaussian is located within a thin shell of specific radius [17]. So let us consider the linear separability of points in the thin shell.

Every vertex on the convex hull of a collection of points is linearly separable from every other point. Therefore every point located on the surface of a hypersphere (the limiting case of a convex polyhedron) can be linearly separated from every other point on the surface by a hyperplane parallel to the tangent plane passing through that point (cf. Fig. 4).

The attribute vectors that lie in a thin shell must be on (or near enough) the surface of a hypersphere. Therefore they form the vertices of a convex polyhedron and are linearly separable and MeFirst Eligible. This can account for the large proportion of MeFirst Eligible items in our data. Here high dimensionality is a blessing rather than a curse! Nevertheless, *two* points are linearly separable from the rest only if they share an edge of the convex hull. (In Fig. 4, there are four eligible pairs of points. The fraction of such pairs decreases rapidly with M.) Quantifying the argument to predict linear separability as a function of M and N will require accurate modeling of the underlying attribute distributions.

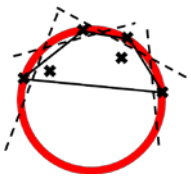

Fig. 4. Convex hull (solid black lines) in thin (red) shell and separating plances (dashed lines) for the four attribure vectors in the shell.

VI. Methodology

This section describes the two linear programming and three single-layer neural network algorithms that we ran on subsets of the Citation and University databases with various combinations of attribute vectors, pairwise difference vectors, normalization, dimensionality augmentation, and maximum number of iterations. After each run, the attribute matrix was multiplied by the new weight matrix, as suggested in Section II. The max-diagonal property of the (M×M) product matrix was checked for independent verification of MeFirst eligibility.

A. Linear Programming

Finding the N weights of each of the M linear ranking functions can be stated as M independent linear programming problems. An LP procedure finds the minimum or maximum of a linear function subject to a set of linear constraints [18, 19]. For ranking, the constraints on the weight vector are imposed by the requirement that the weighted sum of the attributes of a selected item must exceed the sum (with the same weights) for any other item. Weighted vectors that obey the constraints are called a *feasible solution*. Since ranking is not really an

optimization problem, there is considerable flexibility in formulating the objective function. The classical solution was the *Simplex* algorithm. Both the default *Interior Point* algorithm [20] built into MATLAB and into open-source GNU Octave, and the *Dual Simplex* algorithm [21], find a solution, if one exists, in polynomial (here approximately cubic) time.

The LP program is run separately for every row k of the attribute matrix A to find the weight vector $\mathbf{w}_k$. Define B_k as an (M-1)×M *selection matrix* with 1's in the k^{th} column, -1's at (1,1), (2,2), …, (k, k+1), (k+1, k+2), …..(M-1, M), and 0's everywhere else. Then the M-1 inequality constraints for the k^{th} weight vector $\mathbf{w}_k$ are: B_k x A x $\mathbf{w}_k > \mathbf{z}$ where $\mathbf{z}$ is a column vector of M-1 zeros. This yields the (M-1) pairwise difference vectors. The objective function for the k^{th} entry is the sum of its weights. If there is no feasible solution, the algorithm halts.

B. Neural Networks

Besides classification, neural networks are also often used to determine the linear separability of a set of point vectors. The original "fixed-increment perceptron error-correction" procedure [22, 23] for a unitomy (augmented, flipped and cyclically presented patterns $\mathbf{x}$) was (with a slight abuse of notation):

$$\mathbf{w}^{j+1} = \mathbf{w}^j + \mathbf{x}^{jT} \text{ if } \mathbf{x}^j \mathbf{w}^j \leq 0, \text{ and } \mathbf{w}^{j+1} = \mathbf{w}^j \text{ otherwise.}$$

This procedure provably converges to a solution if one exists. Subsequent speed-up efforts introduced corrections proportional to the magnitude of the error and to the fraction of misclassified patterns, a scale factor that decreases the size of the corrections with the number of iterations, randomized order of presentation, and batch correction. Note, however, that the bound on the number of iterations can only be computed from a known solution. After a given number of iterations, 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 the many other algorithms for training a single layer in the MATLAB Deep Learning toolbox [24]. Therefore the maximum number of epochs allowed must be specified.

We compared the count of eligibles and the run time of (a) *Alpha Perceptron,* modeled on the original elementary α-perceptron [25], (b) *MeNet*, a single-layer feed-forward network configured for gradient descent, (c) MATLAB *LegacyPerceptron, and (d) LP*.

The normalization and augmentation experiments were conducted with the hand-coded Alpha Perceptron because the feed-forward net and the legacy perceptron from the Deep Learning toolbox have dozens of hidden functions and parameters, and some deliberately obfuscated *p-files* invisible to external MATLAB users. We set the initial weights to the average attribute vector and added a decreasing correction scale factor. We used this Alpha Perceptron to classify both *attribute vectors* and *pairwise difference vectors,* with and without *dimension augmentation* and *input normalization.*

For *MeNet*, MATLAB's *FeedForwardNet* was configured with zero initial weights, trainFcn *trainlm* (the Levenberg-Marquardt algorithm, which is considered one of the fastest), transferFn *transig* (a soft activation function that requires rounding the outputs to 0 or 1 for classification), and mean-square error criterion *mse*. The MATLAB *Legacy Perceptron* with *hardlim* (step transfer function) was too slow to test on our standard 500-row array of attributes. The best it could do in a nine-hour run was to confirm convergence on 11 of the first 66 entries. All runs were timed on a 2.4GHz Dell Optiplex.

VII. Experiments

A. Data

An article published in October 2020 in *PLOS BIOLOGY* lists publication and citation counts for over 100,000 scientists [26]. We call the updated version of this set of tables [27], from which subsets can be readily extracted according to name, institutional affiliation, or nationality, the *Citation Database*. PLOS (Public Library of Science) was launched in 2001. The PLOS journals are Open Source and charge a publication fee. Many blogs, house organs, and even news media, have already found judicious use for excerpts from the Citation Database. We also drew on a ranking of the 1000 "top" universities of the world according to published attributes [28].

The 12 attributes from the Citation Database that we used for fixed-N ranking are:

1. year of first publication
2. year of most recent publication
3. total cites 1996-2019
4. h-index as of end-2019
5. h_m-index as of end-2019
6. number of single authored papers
7. total cites to single authored papers
8. number of single+first authored papers
9. total cites to single+first authored papers
10. number of single+first+last authored papers
11. total cites to single+first+last authored papers
12. number of distinct citing papers

B. Results

The fraction of MeFirst Eligibility decreases slowly with the number of entries and increases with the number of attributes. The first row of Table II shows that all 79 scientists associated with a small university can be ranked first. According to the last row, even with a hundredfold increase in the number of scientists, 50% remain eligible. Adding columns of publication counts modified only by excluding self-citations barely improves MeFirst Eligibility. Table III shows only a 1% increase from N=11 to N=15 (columns) for 500 rows.

Tables IV and V report comparable results of runs on 500 rows and 12 columns of the Citation Database, starting at randomly chosen Row #6280. Table IV confirms our lemmas. Augmentation (A) and Normalization (N), either separately (N-not-A and A-not-N), or together in that order (A-N), do not hamper determining Eligibility (and one-against-all linear separability) of row vectors. Normalization *followed* by augmentation (N-A) tends to preclude it.

The number of eligible weight vectors found by every algorithm increases monotonically with the maximum number of epochs. Comparing the results of Table V to those of Table IV shows that the perceptron algorithm converges significantly faster on pairwise difference vectors. But it still takes MaxEpochs=1,000,000 to converge on all 496 separable dichotomies. We verified that augmentation and normalization

does not prevent MeFirst Eligibility by computing A×W independently from the output weights of the networks.

Table VI suggests that ranking universities is surprisingly similar to ranking scientists. Since the university data had only 9 attributes, we compared it with the first 9 attributes of the Citation Database. With 9 attributes, 405 entries of 500 from either source are Eligible. For 1000 entries, eligibility differs only by 5%. The averages of 10 runs with integers from a 9-D pseudo-random uniform distribution are even higher.

In the Citation Database, the scientists are already ranked by some criterion devised by its authors. Entries near the top have the most papers and citations. We extracted contiguous groups of 500 entries located in various parts of the array by selecting a different starting row for each group. Table VII shows that the eligibility of groups of 500 scientists consistently favors scientists ranked nearer the bottom (higher 1st row #) in the published list.

Table VIII compares LP with NN. The largest comparison is based on 3000 dichotomies of 3000 12-D attribute vectors. For this task, LP on difference vectors is fastest. (The Karmarkar Interior Point and the Dual Simplex algorithms find the same number of eligibles and barely differ in runtime). The runner-up MeNet, trained with the Levenberg-Marquardt algorithm, is 5-10 times slower. With MaxEpochs=1000, MeNet still misses some solutions.

TABLE II. ELIGIBILITY VS. NUMBER OF ENTRIES M (ROWS) WITH N=12 VIA LINEAR PROGRAMMING

M	Eligible	%	Time(s)
79	79	100	1
500	496	99	6
1000	960	96	20
3000	2254	75	222
5000	3197	64	777
10000	4984	50	5341

TABLE III. ELIGIBILITY VS. NUMBER OF ATTRIBUTES N, WITH M=500 VIA LINEAR PROGRAMMING

N	Eligible	%
9	405	81.0
10	443	88.6
11	495	99.0
12	496	99.2
13	497	99.4
14	499	99.8
15	**500**	**100.0**

TABLE IV. ELIGIBILITY VS. MAX EPOCHS WITH AND WITHOUT AUGMENTATION(A) AND NORMALIZATION (N) ALPHA PERCEPTRON ON ROWS WITH M=500

A/N	Epoch	Eligible	%	Time(s)
~A ~ N	1K	110	22	177
N ~ A	1K	115	23	176
A ~ N	1K	111	22	180
N A	1K	3	1	86
N A	10K	3	1	353
A N	1K	113	23	179
A N	10K	243	49	1359
A N	100K	392	78	10611

TABLE V. ELIGIBILITY VS. MAX EPOCHS ALPHA PERCEPTRON ON ROW DIFFERENVCES WITH M=500

A/N	Epoch	Eligible	%	Time(s)
~N	1K	264	53	117
N	1K	315	63	101
N	10K	409	82	613
N	100K	485	97	2684
N	**1M**	**496**	**99**	**5590**

TABLE VI. ELIGIBILITY VS. SOURCE OF ATTRIBUTES WITH, N=9 VIA LINEAR PROGRAMMING

M	Eligible		
	Universities	Scientists	Random
500	405	405	451
10000	697	663	862

TABLE VII. ELIGIBILITY VS. PUBLISHED RANKING WITH, M=500 VIA LINEAR PROGRAMMING

1st row #	Eligible	%
1	460	92.0
1000	475	95.0
6290	496	99.2
10000	500	100.0
100000	500	100.0

TABLE VIII. COMPARISON OF LP AND NN RUN TIMES WITH N=12

Method	M	MaxEpochs	Eligible	Time(s)
LP linprog	500	N/A	**496**	**6**
LP linprog	1000	N/A	**960**	**20**
LP linprog	3000	N/A	**2254**	**222**
Alpha rows	500	100K	392	10611
Alpha diffs	500	1M	496	5590
MeNet	500	20	496	53
MeNet	1000	100	954	115
MeNet	3000	1000	2243	3013
Legacy	500	500	11/66	43112

VIII. CONCLUSIONS

We do not advocate any particular methods of ranking or classifying scientists and universities. Our sole objective is to shed some light on an obscure and somewhat dusty corner of pattern recognition and machine learning.

A thorough consideration of the relationships between MeFirst ranking, one-against-all halfplane dichotomies, pairwise attribute difference vectors, convex hulls and thin shells in feature space, linear programming, and single-layer neural networks, requires some shifting between algebraic, geometric, and probabilistic perspectives. Our findings can be summarized as follows.

Top-ranking one of M entries with N attributes by a linear weighting function can be restated in terms of linear separability. The relationship between linear separability and linear weighted ranking is governed by necessary and sufficiency conditions on the attribute array. From a geometric perspective, any MeFirst eligible array vector must be located on a vertex of the convex hull of the remaining vectors.

A few attributes suffice to first-rank many entries. Every scientist in most groups of 500 from the Citation Database can be first-ranked with only 12 attributes. Even in a much larger group of 10,000 scientists with the same attributes, about half of the entries are eligible. After a steep initial climb, successful MeFirst ranking increases slowly with the number of attributes. We noticed but did not investigate the effects on MeFirst Eligibility of the statistical dependence among the attributes (e.g. between *with* and *without* self-citation attributes).

The expected concentration of high-dimensional patterns in a thin shell can account for the high fraction of attribute vectors that can be ranked first by a suitable choice of weights. In principle, each of an infinite number of entries with distinct two-dimensional attributes on any convex curve could be ranked first. Top-m ranking is restricted to adjacent points.

For one-against-all dichotomies, verifying the linear separability of pairwise difference vectors is faster than testing the attribute vectors directly. Difference vectors in general positions may be scale normalized without loss of MeFirst eligibility or linear separability. Difference vectors are effective only for highly skewed classifications.

Our experiments suggest that for one-against-many dichotomies, Linear Programming algorithms are much faster than training single-layer ("perceptron") neural networks.

A possible topic for future research is classification based on difference vectors for dichotomies with highly unbalanced class memberships, e.g. novelty detection and one-shot learning.

REFERENCES

1 D.M. Farrell, Electoral systems, 2nd Edition, Red Globe Press, 2011.

2 D. G. Saari, Chaotic elections! A Mathematician looks at voting, American Math Society Publications, 2001.

3 S. Brin and L. Page, The Pagerank citation ranking: Bringing order to the Web http://ilpubs.stanford.edu:8090/422/1/1999-66.pdf accessed Nov. 23, 2021.

4 J.Z. Muller, The tyranny of metrics, Princeton University Press, 2018.

5 M. Biangioli, and A. Lippman, Gaming the metrics, MIT Press, 2020.

6 K. Fukunaga, Statistical Pattern Recognition, 2nd edition, Academic Press, 1990.

7 R.O. Duda, P.E. Hart, D.G. Stork, Pattern Classification, John Wiley and Sons, New Yotk 2001.

8 S. Theodoridis, and K. Koutroumbas, Pattern Recognition, Academic Press, 2009.

9 E. Baum, The Perceptron algorithm is fast for non-malicious distributions. Neural Computation, 2:248–260, 1990.

10 A. Daniely, N. Linial, S. Shalev-Shwartz, The Complexity of learning halfspaces using generalized linear methods, Proceedings of The 27th Conference on Learning Theory, PMLR 35:244-286, 2014.

11 A.T. Kalai, A.R. Klivans, Y.. Mansour, Agnostically learning halfspaces SIAM Journal on Computing, 37(6), 1777-1805, 2008.

12 D. Khashabi. Learning halfspaces: literature review and some recent results, https://danielkhashabi.com/learn/half.pdf , 2016 accessed 11/30/2021.

13 N.J. Nilsson, Learning Machines: The foundations of trainable pattern-classifying systems, McGraw-Hill 1965.

14 P.D. Wasserman, Neural Computing; Theory and Practice, Van Nostrand Reinhold, New York, 1989.

15 G.F. Hughes, On the mean accuracy of statistical pattern recognizers, IEEE Transactions on Information Theory, IT-4 (1), 55-63, 1968.

16 R. Bellman, Adaptive Control Processes: A Guided Tour, Princeton University Press, 1961.

17 C.M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

18 G.B Dantzig: Maximization of a linear function of variables subject to linear inequalities, in Activity Analysis of Production and Allocation, ed. T.C. Koopmans, Wiley & Chapman-Hall, 1951.

19 S.L. Gass, Linear Programming , 5th edition, Dover, 2003.

20 N. Karmarkar, A new polynomial-time algorithm for linear programming, Proceedings of the sixteenth annual ACM Symposium on Theory of Computing STOC '84. p. 302, 1984.

21 H. Nabli, An overview on the simplex algorithm, Applied Mathematics and Computation, Vol. 210, 2, 15 479-489, 2009.

22 F. Rosenblatt, The perceptron: a probabilistic model for information storage and organization in the brain, Psychological Review, 1958.

23 F. Rosenblatt, Principles of Neurodynamics, Spartan Books, New York, 1962.

24 MATLAB, Deep Learning Toolbox. https://www.mathworks.com/products/deep-learning.html, accessed 12/1/2021.

25 T. Barker, A Computer Program for Simulation of Perceptrons and Similar Neural Networks: User's Manual, CSRP Report #8, Cornell University, Ithaca NY 1966.

26 J.P.A. Ioannidis, K.W. Boyack, J. Baas, Updated science-wide author databases of standardized citation indicators, PLoS Biol 18(10): e3000918. doi:10.1371/journal.pbio.3000918, 2020.

27 Mendelay, Data for "Updated science-wide author databases of standardized citation indicators" https://data.mendeley.com/datasets/btchxktzyw/2, accessed 4/10/2021.

28 QS World University Ratings https://group.intesasanpaolo.com/content/dam/portalgroup/nuove-immagini/sociale/2022_QS_World_University_Rankings_1.pdf Accessed 11/30/2021.