Braverman's Spectrum and Matrix Diagonalization versus iK-Means: A unified framework for clustering

Boris Mirkin[1],[2]

[1] Department of Data Analysis and Machine Intelligence, National Research University Higher School of Economics, Moscow, Russian Federation
[2] Department of Computer Science, Birkbeck University of London, UK, (bmirkin@hse.ru)

Abstract. In this paper, I discuss current developments in cluster analysis to bring forth earlier developments by E. Braverman and his team. Specifically, I begin by recalling their Spectrum clustering method and Matrix diagonalization criterion. These two include a number of user-specified parameters such as the number of clusters and similarity threshold, which corresponds to the state of affairs as it was at early stages of data science developments; it remains so currently, too. Meanwhile, a data-recovery view of the Principal Component Analysis method admits a natural extension to clustering which embraces two of the most popular clustering methods, K-Means partitioning and Ward agglomerative clustering. To see that, one needs just adjusting the point of view and recognising an equivaent complementary criterion demanding the cluster to be simultaneously "large-sized" and "anomalous". Moreover, this paradigm shows that the complementary criterion can be reformulated in terms of object-to-object similarities. This criterion appears to be equivalent to the heuristic Matrix diagonalization criterion by Dorofeyuk-Braverman. Moreover, a greedy one-by-one cluster extraction algorithm for this criterion appears to be a version of the Braverman's Spectrum algorithm – but with automated adjustment of parameters. An illustrative example with mixed scale data completes the presentation.

1 Two early approaches by Braverman and his team

1.1 Braverman's algorithm Spectrum

The problem of clustering has been formulated by Misha Braverman as related to a set of objects $I = \{i_1, i_2, ..., i_N\}$ in the so-called potential field which is specified by a similarity function between objects $A(i, j)$, $i, j = 1, 2, ..., N$ [5], [4]. A preferred potential function is defined by equation

$$\phi(d) = \frac{1}{1 + \alpha d^2} \tag{1}$$

where d is Euclidean distance between feature vectors (see, [4]). As is currently well recognised, this is what is referred to as a kernel, one of the most important concepts in machine learning theory [12]. That is a similarity function which forms a positive semidefinite function at every finite set of objects. Moreover, when depending on the Euclidean distance between objects as elements of a Euclidean space of a finite dimension, any kernel function admits a finite set of "eigen-functions" such that $\phi(d(i,j))$ can be expressed as a linear combination of products of values of the eigen-functioms on objects x_i, x_j, $i, j \in I$.

To define his early batch, or parallel, clustering heuristic, Braverman introduces the concept of average similarity between a point x_i and subset of points S referred to as the potential of x_i inflicted by S,

$$A(x_i, S) = \sum_{x_j \in S} A(i,j)/|S| \tag{2}$$

where $|S|$ is the cardunality of S, that is, the number of elements in S. His algorithm *Spectrum* begins at arbitrary point x_1 to build a sequence $x_1, x_2, ..., x_N$ over I so that each next point x_{k+1} maximizes the similarity $A(x_{k+1}, S_k)$ (2) where $S_k = \{x_1, x_2, ..., x_k\}$, $k = 1, 3, ..., N-1$. The sequence of points is accompanied by the sequence of the average similarity values $A(x_2, S_1), A(x_3, S_2), ...,$ $A(x_N, S_{N-1})$. These two sequences form a spectrum, in Braverman's opinion, which can be illustrated with Figure 1.1 replicating an image from Arkadiev and Braverman's book [4], p. 107. On this Figure, x-axis represents the sequence of objects, hand-written images of digits 1, 2, 3, 4, 5, and y-axis shows the levels of the average similarity $A(x_{k+1}, S_k)$. Normally, when a set of homogeneous clusters is present in the data, the graph of the spectrum should look like that on Figure 1.1.

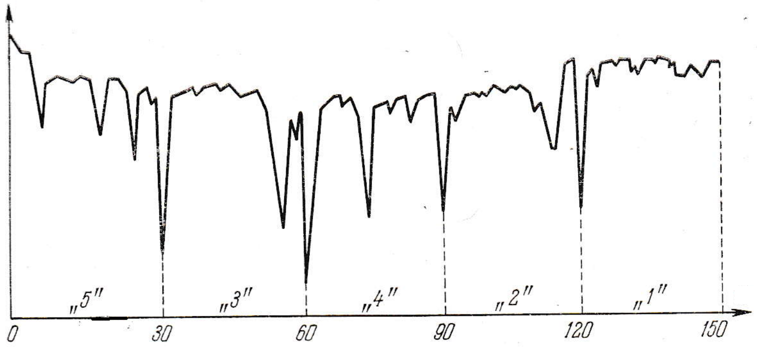

Fig. 1. A spectrum of hand-written images of digits 1, 2, 3, 4, 5, 30 copies of each. The x-axis represents the sequence of objects after application of algorithm Spectrum, and y-axis shows the levels of the average similarity $A(x_{k+1}, S_k)$.

In the current author's view, the graph on Figure 1.1 looks somewhat unlikely. Consider, say, the deep drop between images for "5" and "3". Indeed, the image number 31, of “3”, would look much different from the 30 images of “5”. But the image number 32 is also much different from the previous 30 images, which makes the return of the curve to high levels immediately highly unlikely. A simple modification, though, can save the picture. Assume that the spectrum sequence breaks immediately just before the drop, and a new ordering procedure starts again. Then the averaging of the similarity with the previous cluster discontinues, and new averages are computed starting from the object 31. This is, I think, how a practical version of Spectrum algorithm was working. A threshold value for the drop of similarity has to be pre-chosen, so that a drop of the similarity value below that level would stop the algorithm’s run at a found cluster. After this, the found cluster is removed, and another run of the algorithm is performed at the remaining objects, possibly with a different threshold value. This goes on till the set of remaining objects gets empty. In the follow-up this version of Spectrum will be referred to as Spectrum-B.

1.2 Diagonalization of similarity matrices

This is another idea of Braverman’s team, probably generated by the work over the PhD thesis by Alex Dorofeyuk [8], [6]. Given a similarity matrix $A = (a_{ij})$, consider criterion of goodness of a partition $S = \{S_1, S_2, ..., S_K\}$ with a prespecified number of clusters K by scoring it according to formula

$$f(S) = \sum_{k=1}^{K} \frac{1}{N_k} \sum_{i,j \in S_k} a_{ij} \qquad (3)$$

where N_k is the cardinality of cluster S_k $(k = 1, 2, ..., K)$.

Criterion (3) has been selected by the authors as the best performer out of a family of criteria

$$f(S) = \sum_{k=1}^{K} \frac{1}{\phi(N_k)} \sum_{i,j \in S_k} a_{ij},$$

where $\phi(S)$ is either $\phi(S) = 1$ or $\phi(S) = 1/|S|$ or $\phi(S) = 1/|S|(|S|-1)$. Of course, the experimental base was much limited at the time. And there are obvious drawbacks of the other two criteria in the family. Indeed, at $\phi(S) = 1$ the criterion is just the sum of within-cluster similarities. At non-negative similarity matrix $A = (a_{ij})$ this criterion would lead to a trivial optimal partition S at which all elements gather into the same “big” cluster, whereas all other clusters would be singletons consisting of the weakest links. In contrast, at $\phi(S) = 1/|S|(|S| - 1)$, the criterion would be proportional to the average within cluster similarity. Maximization of this criterion normally would prohibit large-sized clusters because the average similarity may only decrease when a cluster size grows. This leaves criterion (3) as the only option remaining for getting normal-size clusters. In the follow-up we will see that this is not just a lucky occurrence but rather a model-based property.

2 K-Means clustering as a data recovery method

2.1 K-Means algorithm and criterion

K-Means is arguably the most popular clustering algorithm. For an empirical proof of this statement, one may wish to consult [3] and references therein. Specifically, the following Table 1 from [3] clearly demonstrates the prevalence of K-means over other clustering techniques.

Table 1. Numbers of relevant web pages returned by the most popular search engines with respect to queries of the named methods at a computer in Birkbeck University of London (15 November 2015).

Search engine	Google	Bing	Yahoo
K-means	2,070,000	481,000	537,000
Hierarchical clustering	677,000	251,000	268,000
Neighbor-joining	591,000	146,000	148,000
Spectral clustering	202,000	71,500	78,100
Single linkage	140,000	30,900	32,800
Agglomerative clustering	130,000	33,100	33,000

Another, less controversial, statement would be that K-means has nothing to do with Braverman's team developments described above. Here is a conventional formulation of K-Means as a method for the analysis of an object-to-feature dataset.

Batch K-Means

0. *Data pre-processing.* Transform data into a standardized quantitative $N \times V$ matrix Y where N is the number of objects and V, the number of quantified features.

1. *Initial setting.* Choose the number of clusters, K, and tentative centers $c_1, c_2, ..., c_K$, frequently referred to as seeds. Assume initial cluster lists S_k empty.

2. *Clusters update.* Given K centers, determine clusters S'_k $(k = 1, ..., K)$ with the Minimum distance rule assigning any object to its nearest center.

3. *Stop-condition.* Check whether $S' = S$. If yes, end with clustering $S = \{S_k\}$, $c = \{c_k\}$. Otherwise, change S for S'.

4. *Centers update.* Given clusters S_k, calculate within cluster means c_k $(k = 1, ..., K)$ and go to Step 2.

This algorithm usually converges fast, depending on the initial setting. Location of the initial seeds may affect not only the speed of convergence but, more importantly, the final results as well.

As is well known, there is a scoring function, which is minimized by K-Means. To formulate the function, let us define the within cluster error. For a cluster S_k with centroid $c_k = (c_{kv})$, $v \in V$, its square error is defined as the summary distance from its elements to c_k:

$$W(S_k, c_k) = \sum_{i \in S_k} d(y_i, c_k) = \sum_{i \in S_k} \sum_{v \in V} (y_{iv} - c_{kv})^2. \tag{4}$$

The square error criterion is the sum of these values over all clusters:

$$W(S, c) = \sum_{k=1}^{K} W(S_k, c_k) = \sum_{k=1}^{K} \sum_{i \in S_k} d(y_i, c_k) \tag{5}$$

Criterion $W(S, c)$ (5) depends on two groups of arguments: cluster lists S_k and centroids c_k. An alternating minimization algorithm for this criterion would proceed in a series of iterations. At each of the iterations, $W(S, c)$ is, first, minimized over S, given c, and, second, minimized over c, given the resulting S. It is not difficult to see that the batch K-Means above is such an alternating minimization algorithm This warrants that K-Means converges in a finite number of steps because the set of all partitions S over a finite I is finite and $W(S, c)$ is decreased at each change of c or S. Moreover, as experiments show, K-Means typically does not move far away from the initial setting of c. Considered from the perspective of minimization of criterion (5), this leads to the conventional strategy of repeatedly applying the algorithm starting from various randomly generated sets of prototypes to reach as deep a minimum of (5) as possible. This strategy may fail especially if the feature set is large because in this case random settings cannot cover the space of solutions in a reasonable time.

Yet, there is a different perspective, of typology making, in which the criterion is considered not as something that must be minimized at any cost but rather a beacon for direction. In this perspective, the algorithm is a model for developing a typology represented by the centers. The centers should come from an external source such as advice by experts, leaving to data analysis only their adjustment to real data. In this perspective, the property that the final centers are not far away from the original ones, is more of an advantage than not. What is important in this perspective, though, is defining an appropriate, rather than random, initial setting.

2.2 Data recovery equation: encoder and decoder

According to conventional wisdom, the data recovery approach is a cornerstone of contemporary thinking in statistics and data analysis. It is based on the assumption that the observed data reflect a regular structure in the phenomenon of which they inform. The regular structure A, if known, would produce data $F(A)$ that should coincide with the observed data Y up to small residuals which are due to possible flaws in any or all of the following three aspects: (a) sampling entities, (b) selecting features and tools for their measurements, and (c) modeling the phenomenon in question. Each of these can drastically affect results. However, so far only the simplest of the aspects, (a), has been addressed by introduction of probabilities to study the reliability of statistical inference in data analysis. In this text we are not concerned with these issues. We are concerned with the underlying equation :

$$\textbf{Observed data } Y = \textbf{Recovered data } F(A) + \textbf{Residuals } E \qquad (*)$$

In this equation, the following terminology applied in the context of unsupervised learning is getting popular [15]. Data model A such as, for example, partition, is referred to as "encoded data" produced with an encoder, whereas the recovered data, $F(A)$, are those decoded with a decoder. The quality of the encoded data A is assessed according to the level of residuals E: the smaller the residuals, the better the model. Since both encoder and coder methods involve unknown coefficients and parameters, this naturally leads to the idea of fitting these parameters to data in such a way that the residuals become as small as possible, which can be captured by the least squares criterion.

Data analysis involves two major activities: summarization and correlation [15]. In machine learning, their couterparts are unsupervised learning and supervised learning, respectively. In a correlation problem, there is a target feature or a set of target features that are to be related to other features in such a way that the target feature can be predicted from values of the other features. Such is the linear regression problem considered above. In a summarization problem, such as the Principal component analysis or clustering, all the features available are considered target features so that those to be constructed as a summary can be considered as "hidden input features" (see Figure 2).

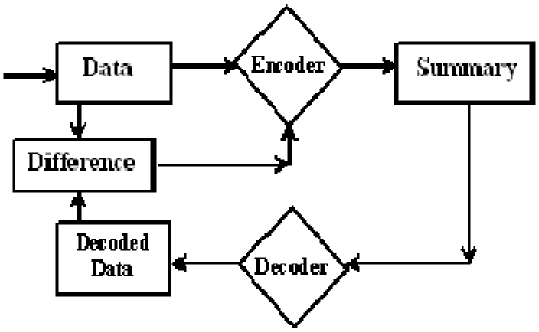

Fig. 2. A diagram for data recovery summarization. Rectangles are for data, both observed and computed, rhombs are for computational constructions. A double feedback shows the ways for adjustment of the encoder according to both the data and decoder.

Considering the structure of a summarization problem as a data recovery problem, one should rely on existence of a rule, decoder, providing a feedback from the summary back to the data. This makes it possible to use the criterion of minimization of the difference between the original data and the output so that the less the difference, the better. In supervised learning, this criterion work only for the feature (s) which is being predicted. Here all the data is to be approximated by the encoded model. This leads to a number of difficulties such as, for example, the issue of data standardization, which, in the supervised context, can be solved by using mainly the feature being predicted.

The least squares perspective gives us a framework in which K-Means, Spectrum and Matrix diagonalization become comparable, as will be shown further on.

2.3 Principal Component Analysis extended to clustering

Principal Component Analysis is a major tool for approximating observed data with model data formed by a few 'hidden' factors. Observed data such as marks of students $i \in I$ at subjects labeled by $v = 1, ..., V$ constitute a data matrix $X = (x_{iv})$. Assume that each mark x_{iv} reflects student i abilities over a set of K hidden talent factors z_ik up to coefficients c_vk, $(i \in I,\ k = 1, 2, ..., K)$. The principal component analysis model [15], suggests that the student i's marks v reflect the inner product of the hidden talent factor scores of the student i, $z_i = (z_{ik})$ and subject v loadings, $c_v = (c_{vk})$. Then equation $(*)$ can be examplified as

$$x_{iv} =< c_v, z_i > +e_{iv}, \tag{6}$$

where the inner product $< c_v, z_i >$ is a specific decoder leading to a rather remarkable, spectral, solution. The least squares criterion is $L^2 = \sum_{i \in I} \sum_{v \in V} (x_{iv} - \sum_{k=1}^{K} c_{vk} z_{ik})^2$.

In matrix terms equation 6 can be rewritten as

$$X = ZC^T + E, \tag{7}$$

where Z is $N \times K$ matrix of hidden factor scores $Z = (z_{ik})$, C is $V \times K$ matrix of subject loadings $C = (c_{vk})$, and $E = (e_{iv})$ is the matrix of residuals. The least squares criterion can be put as $L^2 = ||E||^2 = Tr(E^T E)$ to minimize. Of course, the solution Z, C to this problem is specified up to any unitary $K \times K$ matrix U so that pair ZU, CU leads to the same product $ZUU^T C^T = ZC^T$ since UU^T is an identity matrix. That means that the solution to the problem is not unique but rather specifies a subspace of rank K. Let us denote non-trivial singular triplets of matrix X as (μ_k, z_k, c_k), $k = 1, ..., r$, where r is the rank of X, $\mu_k > 0$ is a singular value, z_k is a normed N-dimensional singular vector $(z_k = (z_{ik})$, c_k is V-dimensional normed vector $c_k = (c_{vk})$ such that $Xc_k = \mu_k z_k$ and $X^T z_k = \mu_k c_k$.

It is not difficult to prove that (μ_k, z_k, c_k) is a singular triplet for matrix X if and only if μ_k^2 is eigen-value of the square matrix $X^T X$ corresponding to eigenvector c, and $z_k = Xc_k/\mu$. This implies that the singular vectors $z_1, z_2, ..., z_K$ are mutually orthogonal as well as vectors $z_1, z_2, ..., z_K$.

Provided that $K < r$, the first-order optimality conditions for the least-squares criterion lead to the first K singular triplets forming its minimizer as the matrix $Z_K M_K C_k^T$ where Z_K and C_K are matrices whose columns are the first K singular vectors, z_k and c_k, respectively, and M_K is the diagonal matrix of the first K singular values μ_k, $k = 1, 2, ..., K$. This implies that $Z = Z_K(M_K)^{1/2}$ and $C = C_K(M_K)^{1/2}$ form solution to 7. Moreover, this solution provides for a Pythagorean decomposition of the data scatter:

$$||X||^2 = Tr(X^T X) = \mu_1^2 + \mu_2^2 + ... + \mu_K^2 + L^2 \tag{8}$$

This implies a one-by-one method for low-rank approximation of the data. You want a one-dimensional approximation? Take the first, the maximum, singular value. You want a visualization of the data on a plane? Take two. The descending order of the singular values implies the order of extraction of the principal components, one-by-one.

This one-by-one extraction approach of PCA was extended by the author to cluster analysis [14]. Specifically, equation (7) with criterion (8) was extended to the constraint that the unknown Z must be zero-one binary N-dimensional vectors to represent K clusters to be found. More precisely, any binary z_k one-to-one corresponds to subset of I, $S_k = \{i : z_{ik} = 1\}$.

The requirements that clusters do not overlap is translated as the constraint that binary vectors z_k are to be mutually orthogonal. Because of this, the square error criterion can be reformulated by putting the sum over k in the beginning while using the binary membership values to limit summation over i by the cluster S_k only:

$$L^2 = \sum_{i \in I} \sum_{v \in V} (x_{iv} - \sum_{k=1}^{K} c_{vk} z_{ik})^2 = \sum_{k=1}^{K} \sum_{i \in S_k} \sum_{v \in V} (x_{iv} - c_{vk})^2$$

.

This proves that the least-squares criterion for the model (7) in the clustering context, that is, under the orthogonal binarity of Z constraint, is but the square-error K-Means clustering criterion.

Thus, a proven fact is that K-Means is a clustering analogue to the Principal Component Analysis. A few conclusions from that:

i A data scatter decomposition should hold.
ii One-by-one strategy for extracting clusters should be valid.
iii Reformulation in terms of feature-to-feature covariance and object-to-object similarity should be tried.

We do not mention some other analogies such as, for instance, a possibility of trying the spectral relaxation of the clustering model for obtaining clusters. These three will be covered in brief in the follow-up sections 2.4, 2.5 and 2.6.

2.4 Data scatter clustering decomposition

Although the data scatter decomposition involving the square error clustering criterion (4) can be derived by using matrix algebra [14], we, however, do this here from the criterion itself:

$$W(S,c)=\sum_{k=1}^{K}\sum_{i\in S_k}\sum_{v=1}^{V}(y_{iv}-c_{kv})^2=$$

$$\sum_{k=1}^{K}\sum_{i\in S_k}\sum_{v=1}^{V}(y_{ik}^2-2y_{iv}c_{kv}+c_{kv}^2)=$$

$$\sum_{k=1}^{K}\sum_{i\in S_k}\sum_{v=1}^{V}y_{iv}^2-\sum_{k=1}^{K}N_k<c_k,c_k>$$

where N_k is the number of elements in cluster S_k. The last equation is derived from the definition that $\sum_{k=1}^{K}\sum_{i\in S_k}\sum_{v=1}^{V}y_{iv}c_{kv}=\sum_{k=1}^{K}N_k\sum_{v=1}^{V}c_{kv}c_{kv}$ because $c_k=\sum_{i\in S_k}y_i/N_k$.

As $\sum_{k=1}^{K}\sum_{i\in S_k}\sum_{v=1}^{V}y_{iv}^2$ is but the data scatter by $T(Y)=\sum_{i=1}^{N}\sum_{v=1}^{V}y_{iv}^2$, denoting the right-hand term in the equation above, by

$$F(S,c)=\sum_{k=1}^{K}N_k<c_k,c_k>, \tag{9}$$

the equation above can be expressed as

$$T(Y)=F(S,c)+W(S,c) \tag{10}$$

which is the Pythagorean decomposition. It should be mentioned that this decomposition is well known in the analysis of variance, a classical part of mathematical statistics: at a centered matrix Y, the data scatter $T(Y)$ is proportional to the summary feature variance, whereas $F(S,c)$ and $W(S,c)$, respectively, to the inter-group and within-group summary variances.

In clustering, however, it is important on it own, because of the complementary criterion in (9) which is to be maximized to make $W(S,c)$ minimum. The value $F(S,c)$ is the part of the data scatter taken into account, that is, the contribution of clustering (S,c) to the data scatter.

The complementary criterion in (9) is the sum of contributions by individual clusters, $f(S_k,c_k)=N_k<c_k,c_k>$; each is the product of the cluster's cardinality and the squared distance from the cluster's center to the origin, 0. Provided that the origin preliminarily has been shifted into the point of 'norm' such as the gravity center, the problem of maximization of $F(S,c)$ is of finding as large-sized and as anomalous clusters as possible, to maximize the sum of cluster contributions $f(S_k,c_k)=N_k<c_k,c_k>$, $k=1,2,...,K$.

2.5 One-by-one strategy for extracting clusters

A procedure proposed by Mirkin [13] is an extension of the one-by-one principal component analysis method to the case at which the scoring components are constrained to represent clusters by having only 1/0 values. This procedure was later renamed as the method of anomalous clusters because of the criterion minimized by the algorithm. Denoting an anomalous cluster sought by S and its center by c, the criterion can be put as

$$D(S, c) = \sum_{i \in S} d(y_i, c) + \sum_{i \notin S} d(y_i, 0) \tag{11}$$

where 0 is the space origin.

This criterion is akin to that of k-means. Two points of departure from the batch K-Means formulation are: (i) the number of clusters K=2 and (ii) one of the centers is at 0 and never changes.

Anomalous cluster alhorithm
Input: $N \times V$ data matrix X.
Output: A single cluster S and its center c as far away from 0, as possible.
Step 1. **Initialization.** Choose an object as far away from 0 as possible and put c at this location.
Step 2. **Cluster update.** Assign to S all those i for which $d(y_i, c) < d(y_i, 0)$.
Step 3. **Center update.** Recompute c as the mean of the newly found S.
Step 4. **Test.** If the new c coincides with the previous one, halt and output the current S and c. Otherwise, take the newly computed c and go to Step 2.

After S is found, its elements can be removed from the entity set, so that next anomalous cluster could be found with the very same process, while the origin remains unchanged. Continuing the process of one-by-one extraction of anomalous clusters, one arrives at the situation when no non-clustered objects remain. Then all the small anomalous clusters are to be removed and centers of the remaining clusters are to be taken to initialise the K-means clustering process – this whole procedure is referred to as the intelligent K-Means (iK-Means) in [7], [2], [16]. The “small” clusters are defined as those containing a predefined number t or less elements.

By default, t is usually taken as unity, $t = 1$. At synthetic data with Gaussian clusters generated, the iK-Means at $t = 1$ tends to produce about twice the number K^* of generated clusters. Indeed, in our computations, it never ever led to a smaller than K^* number of clusters [7]. This is why, in a recent paper, Amorim et al. [2] proposed further agglomeration of iK-Means clusters with the same criterion. It appears, such a hybrid method is much faster than the classic Ward agglomeration method while maintaining similar cluster recovery capabilities [2].

Another comment that should be made is that the criterion in (11) is not exactly equivalent to the criterion of maximization of cluster’s contribution $N_k < c_k, c_k >$, see [18] where a different algorithm is proposed. Here is an example of a dataset at which the two criteria lead to different solutions from [18]. Consider a set of eight two-dimensional observations A to H in Table 2. Assume that the "norm" here is specified as the origin, point 0=(0,0), and no normalization is required. Then criterion would lead to two non-trivial clusters S1=A,B,C and S2=D,E,F, leaving G and H singletons. Anomalous Cluster method outputs only one nontrivial cluster, S1=A,B,C here.

Table 2. Illustrative example of the difference between two single cluster criteria.

Object	x	y
A	-1	3
B	0	3
C	2	2
D	1	1
E	-1	1
F	0	1
G	-2	-1
H	0	-1

2.6 Reformulation of the complementary criterion in terms of object-to-object similarity

Consider the complementary clustering criterion

$$F(S, c) = \sum_{k=1}^{K} \sum_{v \in V} c_{kv}^2 N_k = \sum_{k=1}^{K} N_k d(0, c_k) \tag{12}$$

To maximize this criterion, the clusters should be as far away from 0 as possible. This idea is partly implemented in the Anomalous clustering algorithm above. A batch version is developed in [3] by using an accordingly modified version of the Affinity Propagation algorithm developed by Frey and Dueck [10] (see also URL http://scikit-learn.org/stable/modules/clustering.html#affinity-propagation).

However, in this account, I am not going to concentrate on this, but rather on reformulation of criterion by using row-to-row inner products. Indeed, let us substitute one within cluster average c_k by its definition in (12): $< c_k, c_k >=< c_k, \sum_{i \in S_k} y_{kv}/N_k >= \sum_{i \in S_k} < c_k, y_i > /N_k$. This implies that

$$F(S, c) = \sum_{k=1}^{K} \sum_{i \in S_k} < y_i, c_k > \tag{13}$$

This expression shows that the K-Means criterion that minimizes the within-cluster distances to centroids is equivalent to the criterion (13) for maximization of the within-cluster inner products with centroids. By further substituting the same formula into (13), we arrive at equation

$$F(S,c)=\sum_{k=1}^{K}\sum_{i,j\in S_k}<y_i,y_j>/N_k \tag{14}$$

expressing the criterion in terms of entity-to-entity similarities $a_{ij}=<y_i,y_j>$, with centroids c_k present implicitly. Criterion (14) is the total within-cluster semi-averaged similarity that should be maximized to minimize the least-squares clustering criterion.

2.7 Returning to Matrix diagonalization

Obviously, by denoting $a_{ij}=\sum_v y_{iv}y_{jv}$, criterion (14) gets the form of Braverman-Dorofeyuk criterion in (3), that is, the semi-average within cluster similarity to maximize. Indeed, $a_{ij}=\sum_v a_{v,ij}$ is the sum of $a_{v,ij}=y_{iv}y_{jv}$, that are scores of the level of similarity between entities due to single features:

$$F(S,c)=\sum_{k=1}^{K}\frac{1}{N_k}\sum_{i,j\in S_k}a_{ij} \tag{15}$$

Assuming that y_{iv} has been pre-processed by subtracting a reference value such as the grand mean $m_v=\sum_i^N y_{iv}/N$, one can see that $a_{v,ij}$ is negative if i and j differ over m_v so that y_{iv} and y_{jv} lie on v-axis at the different sides of zero. In contrast, $a_{v,ij}$ is positive if y_{iv} and y_{jv} lie on v-axis at the same side, either positive or negative. The greater the distance from 0 to y_{iv} and y_{jv}, the greater the value of $a_{v,ij}$. If the distribution is similar to a Gaussian one, most entities fall near grand mean, which is 0 after the normalization, so that most similarities are quite small.

Moreover, there is a claim in the literature that the inner product is much beneficial at larger sizes of the feature set if the data is pre-normalized in such a way that the rows, corresponding to entities, are normed so that each has its Euclidean norm equal to unity, so that the inner product becomes the cosine of the angle between the vectors.

This criterion has something to do with the spectral approach. Indeed, matrix $A=(a_{ij})$ can be expressed through the data matrix Y as $A=YY^T$ so that criterion (14), in a matrix form, is:

$$F(S,c)=\sum_{k=1}^{K}\frac{s_k^TAs_k}{s_k^Ts_k} \tag{16}$$

where $s_k=(s_{ik})$ is a binary membership vector for cluster S_k so that $s_{ik}=1$ if $i\in S_k$ and $s_{ik}=0$, otherwise. In mathematics, s_k is referred to as indicator of S_k. Proof of (16) follows from the fact that $s_k^Ts_k=N_k$ and $\sum_{i,j\in S_k}a_{ij}=s_k^TAs_k$. This is the sum of Rayleigh quotients whose extremal values are eigenvalues of

A. Of course, this is similar to Rayley quotients in the Proincipal Component Analysis, except that solutions here must be binary zero-one vectors. Therefore, the contributions of individual clusters, $\sum_v c_{kv}^2 N_k$, are akin to the contributions of individual principal components, μ_k^2, both expressing the maximized values of the corresponding Rayleigh quotients, albeit over different domains.

One more comment should be about potentially using kernels, that is, functions $K(y_i, y_j)$ forming positive semi-definit matrices, to imitate the inner products $a_{ij} =< y_i, y_j >$. These functions were introduced by E. Braverman and his team to be used as a computationally feasible possibility of non-linearly transforming the feature space. Kernel "trick" is quite popular in clustering. However, it applies to the K-Means criterion itself leading to a nuch more complex and less interpretable formula than the expression (15) – see, for example, review in [9].

There can be several approaches to optimization of the criterion (15). Of course one of them is the agglomerative approach by Braverman and Dorofeyuk. However, I concentrate on an extension of the Anomalous clustering one-by-one approach to the case of similarity matrices.

2.8 Returning to Spectrum

Consider any item in the criterion (15) as a maximized criterion for finding an individual cluster. One may drop the index k in this section, since one cluster S is sought here. (Note that S here is not a partition but just a subset.)

Given a similarity matrix $A = (a_{ij})$, $i, j \in I$, let us define within-S average similarity as $a(S) = \sum_{i,j \in S} a_{ij}/N_S^2$ where N_S is the cardinality of S. Then the semi-average clustering criterion in (15) can be converted to

$$g(S) = \sum_{i,j \in S} a_{ij}/N_S = N_S a(S). \tag{17}$$

This criterion combines two:

(i) maximize the cluster's tightness as measured by the within-cluster mean $a(S)$,
(ii) maximize the cluster size, measured by N_S.

These two goals are not quite compatible: the greater the cluster, the weaker the average within-cluster similarity. The product in (17), thus, balances them and, in this wald lead to a relatively tight cluster of a reasonable size.

Indeed, the tightness of S can be mathematically described by using the following concept [15]. For any entity $i \in I$, define its attraction to subset S as the difference between its average similarity to S, $a(i, S) = \sum_{j \in S} a_{ij}/N_S$, and half the average within-cluster similarity:

$$\alpha(i, S) = a(i, S) - a(S)/2. \tag{18}$$

The fact that $a(S)$ is equal to the average $a(i, S)$ over all $i \in S$, leads us to expect that normally $a(i, S) \geq a(S)/2$ for the majority of elements $i \in S$, that is, normally $\beta(i, S) \geq 0$ for $i \in S$. It appears, a cluster S maximizing the criterion $A(S)$ is much more than that: S is cohesive internally and separated externally because all its members are positively attracted to S whereas non-members are negatively attracted to S [15].

Maximizing criterion $g(S)$ in (17) is akin to a combinatorially feasible problem of finding a subgraph of maximum density when the similarity matrix is non-negative [14]. Otherwise, it is non-polynomial. A local search algorithm would use a pre-defined neighborhood system to locally maximize the criterion at the neighborhood of a pre-specified subset S. In [15], the neighborhood system was considered to include $S \pm k$ for any k. Here, we take on a simpler neighbourhood system to include only those subsets $S + k$ obtained from S by adding to S any $k \notin S$.

Specifically, a version of algorithm ADDI [14] can be defined as follows.

Start from a singleton $S = \{i\}$ consisting of any object $i = 1, 2, ..., N$. Then proceed iteratively as follows. Given S, take an object $k \notin S$ maximizing the difference $\Delta(S, k) = g(S + k) - g(S)$. Check whether $\Delta(S, k) > 0$. If Yes, make $S = S + k$ and go to the beginning of the iteration. If not, stop and output S and its contribution to the data scatter $g(S)$ as well as the within-cluster average similarity $a(S)$.

This algorithm is a model-based version of Spectrum algorithm. Indeed, it is not difficult to prove that

$$\Delta(S, k) = \frac{N_S}{N_S + 1}(a(k, S) - \frac{a(S)}{2} + \frac{a_{kk}}{2N_S}) \tag{19}$$

Consider a simplifying assumption that $a_{kk} = 0$, so that clusters are defined by similarities between different objects only. Computationally, this assumption is easy to maintain by zeroing the main diagonal immediately before the start of a run of the ADDI algorithm. Then the maximum of $\Delta(S, k)$ corresponds to the maximum of the average similarity $a(k, S)$ between k and S, exactly as in Spectrum algorithm. But in ADDI, there is a natural stopping condition according to the sign of (19). Adding of elements stops whenever condition $a(k, S) < \frac{a(S)}{2}$ holds.

In this way, ADDI algorithm may be considered a theory based version of Spectrum. Moreover, ADDI not only defines a cluster-specific stopping condition, but also, the starting object: that must be that i maximizing a_{kk}.

2.9 Illustrative example

Consider an illustrative dataset in Table 3 after [16]. It relates to eight fictitious colleges in the UK described by five features.

Two features are of college sizes:

1) Stud - The number of full time students;

2) Acad - The number of full time teaching staff.

Three features of teaching environment:

3) NS - The number of different schools in a college;

4) DL - Yes or No depending on whether the college provides distant e-learning courses or not;

5) Course type - The course type is a categorical feature with three categories: (a) Certificate, (b) Master, (c) Bachelor, depending on the mainstream degree provided by the college.

Table 3. Colleges: Eight colleges: the first three mostly in Science, the next three in Engineering, and the last two in Arts. These categories are not supposed to be part of the data. They can be seen via first letters of the names, S, E and A, respectively.

College	Stud	Acad	NS	DL	Course type
Soli	3800	437	2	No	MSc
Semb	5880	360	3	No	MSc
Sixpe	4780	380	3	No	BSc
Etom	3680	279	2	Yes	MSc
Efin	5140	223	3	Yes	BSc
Enkee	2420	169	2	Yes	BSc
Ayw	4780	302	4	Yes	Certif.
Ann	5440	580	5	Yes	Certif.

The data in Table 3 can be utilized to cluster the set of colleges and describe clusters in terms of the features. One would ask whether the clusters are in line with the three main areas: Science, Engineering, Arts.

To analyze the data, one should first quantify the table by enveloping all the qualitative categories into binary zero-one features. Therefore, the Type of course will be represented by three yes/no features: is it MSc; is it Bsc; is it Certificate. Then Yes answer is coded by 1 and No answer by 0 (see Table 4).

Table 4. Quantitative representation of the Colleges data as an 8×7 entity-to-attribute matrix.

Entity	Stud	Acad	NS	DL	MSc	BSc	Certif.
1	3800	437	2	0	1	0	0
2	5880	360	3	0	1	0	0
3	4780	380	3	0	0	1	0
4	3680	279	2	1	1	0	0
5	5140	223	3	1	0	1	0
6	2420	169	2	1	0	1	0
7	4780	302	4	1	0	0	1
8	5440	580	5	1	0	0	1

Now this table can be standardized by subtracting from each column its average and dividing it by its range. To further balance the total contribution of the three categorical features on the right so that it is equal to the contribution of one feature represented by them, we divide them by the square root of 3. Then their part in the data scatter will be divided by 3 and the effect of tripling the original feature, Type of course, will be reversed [15]. The resulting data table is in Table 5.

Table 5. Range standardized Colleges matrix with the additionally rescaled nominal feature attributes; Mean is grand mean, Range the range and Cntr the relative contribution of a feature to the data scatter.

Item	Stud	Acad	NS	DL	MSc	BSc	Cer.
1	-0.20	0.23	-0.33	-0.63	0.36	-0.22	-0.14
2	0.40	0.05	0.00	-0.63	0.36	-0.22	-0.14
3	0.08	0.09	0.00	-0.63	-0.22	0.36	-0.14
4	-0.23	-0.15	-0.33	0.38	0.36	-0.22	-0.14
5	0.19	-0.29	0.00	0.38	-0.22	0.36	-0.14
6	-0.60	-0.42	-0.33	0.38	-0.22	0.36	-0.14
7	0.08	-0.10	0.33	0.38	-0.22	-0.22	0.43
8	0.27	0.58	0.67	0.38	-0.22	-0.22	0.43
Mean	4490	341.3	3.0	0.6	0.4	0.4	0.3
Range	3460	411	3.00	1.00	1.73	1.73	1.73
Cntr, %	12.42	11.66	14.95	31.54	10.51	10.51	8.41

Example 1. **Centers of subject clusters in Colleges data**

Let us consider the subject-based clusters in the Colleges data. The cluster structure is presented in Table 6 in such a way that the centers are calculated twice, once for the raw data in Table 4 and the second time, with the standardized data in Table 5.

Table 6. Means of the variables in Table 5 within K=3 subject-based clusters, real (upper row) and standardized (lower row).

		Mean						
Cl.	List	St (f1)	Ac (f2)	NS (f3)	DL (f4)	B (f5)	M (f6)	C (f7)
1	1, 2, 3	4820	392	2.67	0	0.67	0.33	0
		0.095	0.124	-0.111	-0.625	0.168	-0.024	-0.144
2	4, 5, 6	3740	224	2.33	1	0.33	0.67	0
		-0.215	-0.286	-0.222	0.375	-0.024	0.168	-0.144
3	7, 8	5110	441	4.50	1	0.00	0.00	1
		0.179	0.243	0.500	0.375	-0.216	-0.216	0.433

Example 2. **Minimum distance rule at subject cluster centroids in Colleges data**

Let us apply the Minimum distance rule to entities in Table 5, given the standardized centroids c_k in Table 6. The matrix of distances between the standardized eight row points in Table 5 and three centroids from Table 6 is in Table 7.

Table 7. Distances between the eight standardized College entities and centroids; within column minima are highlighted.

	Entity, row point from Table 5							
Centers	1	2	3	4	5	6	7	8
c_1	**0.22**	**0.19**	**0.31**	1.31	1.49	2.12	1.76	2.36
c_2	1.58	1.84	1.36	**0.33**	**0.29**	**0.25**	0.95	2.30
c_3	2.50	2.01	1.95	1.69	1.20	2.40	**0.15**	**0.15**

The table, as expected, shows that points 1,2,3 are nearest to centroid c_1, 4,5,6 to c_2, and 7, 8 to c_3. This means that the rule does not change clusters. These clusters will have the same centroids. Thus, no further calculations can change the clusters: the subject-based partition is to be accepted as the result.

But of course the algorithm may bring wrong results if started with a wrong set of centers, even if the initial setting fits well into clustering by subject.

Table 8. Distances between the standardized Colleges entities and entities 1, 4, 7 as tentative centroids.

	Row-point							
Centers	1	2	3	4	5	6	7	8
1	**0.00**	**0.51**	**0.88**	1.15	2.20	2.25	2.30	3.01
4	1.15	1.55	1.94	**0.00**	0.97	**0.87**	1.22	2.46
7	2.30	1.90	1.81	1.22	**0.83**	1.68	**0.00**	**0.61**

Example 3. **Unsuccessful K-Means run with subject-based initial seeds**

With the initial centroids at rows 1, 4, and 7, all of different subjects, the entity-to-centroid matrix in Table 8 leads to cluster lists $S_1 = \{1, 2, 3\}, S_2 = \{4, 6\}$ and $S_3 = \{5, 7, 8\}$ which do not change in the follow-up operations. These results put an Engineering college among the Arts colleges. Not a good outcome.

Example 4. **Explained part of the data scatter**

The explained part of the data scatter, $F(S, c)$, is equal to 43.7% of the data scatter $T(Y)$ for partition $\{\{1, 4, 6\}, \{2\}, \{3, 5, 7, 8\}\}$, found with entities 1,2,3 as initial centroids. The score is 58.9% for partition $\{\{1,2,3\},\{4,6\},\{5,7,8\}\}$, found with entities 1,4,7 as initial centroids. The score is maximum, 64.0%, for the subject based partition $\{\{1,2,3\},\{4,5,6\},\{7,8\}\}$, which is thus superior.

Example 5. **Similarity matrix and clusters using ADDI algorithm**

Table 9. The matrix of similarity between objects obtained as the result of multiplication of the standardized matrix in Table 5 by its transpose.

College	1	2	3	4	5	6	7	8
Soli	0.794	0.519	0.260	0.086	-0.474	-0.237	-0.478	-0.470
Semb	0.519	0.752	0.293	-0.137	-0.307	-0.629	-0.299	-0.191
Sixpe	0.260	0.293	0.604	-0.404	-0.048	-0.126	-0.330	-0.250
Etom	0.086	-0.137	-0.404	0.527	0.005	0.320	-0.069	-0.328
Efin	-0.474	-0.307	-0.048	0.005	0.457	0.347	0.090	-0.069
Enkee	-0.237	-0.629	-0.126	0.320	0.347	0.983	-0.074	-0.583
Aiw	-0.478	-0.299	-0.330	-0.069	0.090	-0.074	0.549	0.612
Ann	-0.470	-0.191	-0.250	-0.328	-0.069	-0.583	0.612	1.279

The similarity matrix $A = YY^T$ in Table 9 is obtained from the standardized matrix Y in Table 5. Its structure pretty much corresponds to that underlying the Spectrum algorithm: there are three groups, S, E, and A, so that all the within-group similarities are positive, whereas almost all the inter-group similarities are negative, or quite small when positive, see a(Soli,Etom)=0.086 and a(Efin,Aiw)=0.09.

Let us apply algorithm ADDI to it starting from the most anomalous object, which is Ann with its squared Euclidean distance to 0 equal to 1.279. Then Aiw joins in, with a(Ann,Aiw)=0.612. All the other objects have negative similarities with this cluster (assuming zeroing of all the diagonal elements while computing or not), so that the computation stops here: cluster A is complete.

After removal of the two A-colleges, college Enkee becomes the most anomalous, with the diagonal element 0.983. Its nearest is Efin, a(Efin, Enkee)=0.347. Merge them into one cluster. The only positive average similarity to this is frim Etom, (0.05+0.320)/2=0.162. Now the results split. Assuming the diagonals zeroed, this value is less than half of the within-cluster dimilarity, 0.347/2=0.174, so that Etom should not be added to the cluster. By sticking to the inequality (19), one can see that $a(k,S) - a(S)/2 + a_{kk}/(2N_S) = 0.162 - 0.368/2 + 0.527/4 = 0.1098 > 0$, that is, adding Etom to the cluster will increase the criterion $g(S)$ and, thus, must be done. This would complete E-cluster.

This leaves three unclustered S-colleges. Of them, Soli is the most anomalous; it goes into the cluster first. Its nearest is Semb with a(Soli, Semb)=0.519. The average similarity of Sixpe to the current cluster {Soli, Semi } is (0.260+0.293)/2=0.276. This is greater than half the within-ckuster similarity (with the diagonal zeroed) 0.519/2=0.260; the more so at the diagonal taken into account.

Therefore, ADDI leads to the course based clusters only, unlike K-Means itself, which may fail on this.

3 Conclusion

This paper gives a review of author's contribution to K-Means clustering, which include the two seemingly unrelated E. Braverman's approaches described in the beginning of the presentation. Main concepts and results reported are:

1. K-Means can be considered as a procedure emerging within the data-recovery approach. Specifically, this is a method for fitting an extension of the SVD-like Principal Component Analysis data model towards binary hidden factor scores.
2. This opens up a bunch of equivalent reformulations of K-Means criterion including:
   (a) Maximum of partition's contribution to the data scatter, that is the sum of squared Euclidean distances between centers and the origin weighted by cluster cardinalities;
   (b) Maximum of the summary inner product between object points and corresponding centers;
   (c) Spectral reformulation;
   (d) Maximum of Dorofeyuk-Braverman's semi-average within-cluster similarity.
3. One-by-one Principal Component Analysis strategy applies to clustering. This leads to:
   (a) One-by-one extraction of anomalous clusters leading to a natural initialization of K-Means. The initialization has proven competitive experimentally in application to such issues as speeding-up agglomerative Ward clustering and determining the number of clusters.
   (b) One-by-one extraction of clusters over a similarity matrix. This approach appears to be much similar to Braverman's Spectrum algorithm, leading additionally to automation of starting and stopping conditions.

Other extensions emerging within the data-recovery approach, such as clustering over mixed data, feature weighting, Minkowski metric clustering, consensus clustering, and hierarchical clustering can be found in the author's monograph [16].

References

1. M.A. Aiserman, E.M. Braverman, L.I. Rosonoer (1970) Method of Potential Functions in the Theory of Machine Learning, Nauka Publishers: Main Editorial for Physics and Mathematics, Moscow (In Russian).
2. R. de Amorim, V. Makarenkov, B. Mirkin (2016). A-Ward$_{p\beta}$: Effective hierarchical clustering using the Minkowski metric and a fast k-means initialisation. Information Sciences, 370, 343-354.

3. de Amorim, R. C., Shestakov, A., Mirkin, B., and Makarenkov, V. (2017). The Minkowski central partition as a pointer to a suitable distance exponent and consensus partitioning, *Pattern Recognition*, 67, 62-72.
4. A.G. Arkadiev, E.M. Braverman (1971) Machine learning for classification of objects, Nauka Publishers: Main Editorial for Physics and Mathematics, Moscow (In Russian).
5. O.A. Bashkirov, E.M. Braverman, I.B. Muchnik (1964) Algorithms for machine learning of visual patterns using potential functions, Automation and Remote Control, no. 5, 25 (In Russian).
6. E. Braverman, A. Dorofeyuk, V. Lumelsky, I. Muchnik (1971) Diagonalization of similarity matrices and measuring of hidden factors, In "Issues of extension of capabilities of automata", Institute of Control Problems Press,Moscow, 42-79 (In Russian).
7. M. Chiang, B. Mirkin (2010) Intelligent choice of the number of clusters in k-means clustering: an experimental study with different cluster spreads. Journal of Classification, 27(1), 3-40.
8. A. A. Dorofeyuk (1966) Machine learning algorithm for unsupervised pattern recognition based on the method of potential functions, Automation and Remote Control (USSR), vol. 27, 1728-1737.
9. M. Filippone, F. Camastra, F. Masulli, and S. Rovetta (2008). A survey of kernel and spectral methods for clustering. Pattern Recognition, 41(1), 176-190.
10. B. Frey and D. Dueck (2007)Clustering by passing messages between data points, *Science*, v. 315, no. 5814, 972–976.
11. K.J. Holzinger and H.H. Harman (1941) *Factor Analysis*, Chicago: University of Chicago Press.
12. Kung, S. Y. (2014) Kernel methods and machine learning. Cambridge University Press.
13. B.G. Mirkin (1987) The method of principal clusters, *Automation and Remote Control*, 48(10), 1379-1388.
14. B. Mirkin (1990) Sequential fitting procedures for linear data aggregation model, *Journal of Classification*, 7, 167-195.
15. B. Mirkin (2011). Core concepts in data analysis: summarization, correlation, visualization. Springer, London.
16. B. Mirkin (2012) Clustering: A data recovery approach, Chapman and Hall/CRC Press.
17. B. Mirkin, M. Tokmakov, R. de Amorim, and V. Makarenkov (2017) Capturing the number of clusters with K-Means using a complementary criterion, affinity propagation, and Ward agglomeration (submitted)
18. Z. Taran, B. Mirkin (2017) Exploring patterns of corporate social responsibility using a complementary k-means clustering criterion (submitted).