Summary and Semi-Average Similarity Criteria for Individual Clusters

Boris Mirkin

1 Graph theoretic cluster relevant concepts

Currently, the most popular format for similarity data is of square matrix $A = (a_{ij})$ of pair-wise indices a_{ij} expressing similarity between entities $i, j \in I$. The greater the value of a_{ij}, the greater is the similarity between i and j. Intuitively, cluster is a set of highly similar entities that are dissimilar from entities outside of the cluster. Some examples of similarity data are (i) individual judgements of similarity expressed using a fixed range, (2) correlation coefficients between variables or time series, (3) graphs represented by 1/0-similarity matrices, (4) weighted graphs, or networks, (5) probabilities of common ancestry, especially in proteomics, (6) affinnity data obtained by transformation of distances using a Gaussian or another kernel function. Here are two data sets of this type:

Similarity between Elementary Functions

There are common knowledge control devices, oral questioning or written test papers: the mark depending on thematch between the answers and corresponding knowledge fragments. A few decades ago, an education research team in Russia proposed a different method for knowledge control based on the respondent scoring similarity between the base concepts of the subject tested. The premise is that there exists a structure of semantic relationship among the concepts, which must be acquired by learning; the discrepancy between a student's personal structure and that one to be acquired may be used as a measure of the student's knowledge extent (see [35, 24]). Therefore, the student score is determined from the differences between two concept-to-concept similarity matrices: first, that considered to be right, and second, that produced by the respondent.

Boris MIrkin
Division of Applied Mathematics, Higher School of Economics, Moscow, RF, and Department of Computer Science and Information Systems, Birkbeck, University of London, UK, e-mail: bmirkin@hse.ru

The following Table 1 of similarities between 9 elementary algebraic functions has been produced by an expert high-school mathematics teacher. How the dif-

Function	e^x	lnx	$1/x$	$1/x^2$	x^2	x^3	$\sqrt{x}$	$\sqrt[3]{x}$	
lnx	7								
$1/x$	1	1							
$1/x^2$	1	1	7						
x^2	2	2	2	2					
x^3	3	2	1	1	6				
$\sqrt{x}$	2	4	1	2	5	4			
$\sqrt[3]{x}$	2	4	1	1	5	3	5		
$\|x\|$	2	3	1	1	5	2	3	2	

Table 1 Functions: Similarity scores between nine elementary functions as rated by a high-school mathematics teacher.

ferences should be evaluated? Well, over their semantic structures. The semantic structure conventionally is determined with a multidimensional scaling method. In this case, however, the semantic space dimensions obviously corresponded to entity clusters, which brought forward an idea of using clusters only. This brings forward the problem of finding a cluster structure by the similarity matrix.

Eurovision song contest scoring

Eurovision song contest is an all-European television show at which each participating country presents a song performed by its singer. The performances are evaluated by each of the participating countries separately, so that each country selects eleven best performances and gives them scores in the order of preference. The winner is that country whose song received the maximum score combined. The participants frequently suspect that voting might be led by ethnic and cultural relations rather than by the quality of performance.

Table 2 presents a part of the whole picture using data of only 19 European countries, out of 46, in its left column. The table, including the sample of countries, has been compiled by the author using scoring data publicly available at http://www.escstats.com/ (visited 28/2/2013). Its entries are the average scores given by each country to her 10 top choices at all the Eurovision song contests (up to and including year 2011). I tried to make a representative sample of less than 20 entities so that the reader could see the data with a naked eye. Each row of the table corresponds to one of the nineteen countries, and assigns a non-zero score to those of the other eighteen that have been among the 10 best choices. For the sake of convenience, each score has been multipled by 10 so that all scores are expressed with whole integers.

The cluster structure of the table should quantify to what extent the gossip of the effects of cultural and ethnical links on voting is justified, because song and performance "quality" may be considered random from year to year, so that in the ideal case when no cultural preferences are involved at evaluations, the similarity matrix should be of a random structure too.

Table 2 Eurovision scoring: Each row contains the average score given by the row country to the column country in Eurovision song contests (multiplied by 10).

Country	Az	Be	Bu	Es	Fr	Ge	Gr	Is	It	Ne	Pol	Por	Ro	Ru	Se	Sp	Sw	Ukr	UK
1 Azerbajan	0	0	0	0	0	0	61	48	0	0	0	0	50	65	0	0	0	90	0
2 Belgium	38	0	0	0	0	39	40	0	0	47	0	0	0	0	0	34	0	0	42
3 Bulgaria	67	0	0	0	0	0	93	0	0	0	0	0	0	48	60	0	0	44	0
4 Estonia	41	0	0	0	0	0	0	0	43	0	0	0	0	88	0	0	0	43	0
5 France	0	37	43	0	0	0	0	56	47	0	0	54	0	0	80	0	0	0	41
6 Germany	0	0	0	0	34	0	37	35	0	0	55	0	0	0	70	0	0	0	42
7 Greece	54	0	80	0	41	0	0	0	0	0	0	0	40	0	80	44	0	38	0
8 Israel	50	0	0	0	0	0	0	0	0	43	0	0	66	74	50	0	0	62	43
9 Italy	0	0	100	0	54	0	0	0	0	0	0	0	120	0	0	0	0	65	52
10 Netherlands	39	46	0	0	0	38	0	45	0	0	0	0	0	0	70	0	0	0	0
11 Poland	84	43	0	39	0	0	0	0	90	0	0	0	0	0	0	0	0	82	0
12 Portugal	0	35	0	0	0	45	0	41	81	0	0	0	52	0	57	42	0	74	43
13 Romania	52	0	0	0	0	0	82	0	60	0	0	0	0	49	80	0	0	35	0
14 Russia	99	0	0	0	0	0	37	36	0	0	0	0	0	0	80	0	0	77	0
15 Serbia	0	0	53	0	0	0	73	0	0	0	0	0	0	44	0	0	0	44	0
16 Spain	0	0	78	0	0	51	45	0	74	0	0	43	79	0	47	0	0	46	0
17 Switzerland	0	0	0	0	44	0	0	42	47	0	0	0	0	0	106	41	0	0	41
18 Ukraine	111	0	0	0	0	0	0	0	0	0	60	0	0	98	90	0	0	0	0
19 UK	0	0	0	36	0	39	38	0	0	0	0	0	0	0	37	0	0	0	0

As one can see both similarity matrices are non-symmetric, which is not that important with the clustering criteria used in this paper.

Formal thinking have been applied to similarity clusters rather early, in graph theory, much before the need in clustering has been recognised as a general problem. A graph may be thought of as a flat 1/0 similarity matrix, its nodes corresponding to row/columns and edges/arcs to entities with ones corresponding to edges. Cluster related graph-theoretic concepts include: (a) *connected component* (a maximal subset of nodes in which there is a path connecting each pair of nodes), (b) *bicomponent* (a maximal subset of nodes in which each pair of nodes belongs to a cycle), and (c) *clique* (a maximal subset of nodes in which each pair of nodes is connected by an edge). These remain much relevant in various contexts. For example, define $S \subset I$ to be a strong cluster if for any $i, j \in S$ there exists $k \in I - S$ such that $a_{ij} > a_{ik}$. It is not difficult to prove that S is a strong cluster if and only if S is a clique for some threshold graph. A threshold graph is defined by a real t so that (i, j) is its arc if and only if $a_{ij} > t$. Even more relevant is a more recent concept of the maximum density subgraph [9]. The density $g(S)$ of a subgraph $S \subset I$ is the ratio of the number of edges in S to the number of elements $|S|$. For an edge weighted graph with weights specified by the matrix $A = (a_{ij})$, the density of a subgraph on $S \subseteq I$ $g(S)$ is defined by the *Rayleigh quotient* $s^T As/s^T s$, where $s = (s_i)$ is the characteristic vector of S, viz. $s_i = 1$ if $i \in S$ and $s_i = 0$ otherwise. A subgraph of maximum density represents a cluster. After removing such a cluster from the graph, a maximum density subgraph of the remaining graph can be found. This may be repeated until no "significant" clusters remain. Such an incomplete clustering procedure is natural for many types of data, including protein interaction networks. However, to our knowledge, this method has never been applied to such problems, probably be-

cause it involves rather extensive computations. A heuristic analogue can be found in [2]. The maximum density subgraph problem is of interest because it is a reasonable relaxation of the maximum clique problem and fits well into data recovery clustering (see section 4.2). The maximum value of the Raleigh quotient of a symmetric matrix over any real vector s is equal to the maximum eigenvalue and is attained at an eigenvector corresponding to this eigenvalue. This gives rise to the so-called *spectral clustering*, a method of clustering based on first finding a maximum eigenvector s^* and then defining the spectral cluster by $s_i = 1$ if $s_i^* > t$ and $s_i = 0$ otherwise, for some threshold t. This method may have computational advantages when A is sparse. Unfortunately, it does not necessarily produce an optimal cluster, but in practice it works well.

2 Within cluster summary criterion

2.1 Uniformly shifting simiarities

For any subset $S \subseteq I$, maximizing the sum of within-S similarities

$$f(S) = \sum_{i,j \in S} a_{ij} \tag{1}$$

seems a perfect criterion for clustering. It is simple and intuitive. The greater the total within cluster similarity $f(S)$, the tighter the cluster.

There is an issue though. If all the similarities are non-negative, as it usually happens (see data in Tables 1 and 2), the function $f(S)$ can only increase if any other entities are added to S so that the maximum $f(S)$ is reached at $S = I$, the universal cluster. This is why the criterion has been applied only to the situations at which the size of the cluster is pre-specified or, in the case of partitioning, the distribution of entities over clusters is pre-specified, say, by restricting admissible parttions to those consisting of equal-sized clusters only (as in Kupershtoch, Mirkin [19]).

Yet the criterion can be saved if applied to unrestricted similarity values. That is, the criterion does work if the similarity data is pre-processed to admit negative values. An intuitively reasonable way for doing that is by subtracting some "background noise" from the similarity data.

Specifically, Kupershtoch, Trofimov and Mirkin [20] proposed subtraction of a constant value π from all the similarity values so that criterion (1) becomes

$$f(S, \pi) = \sum_{i,j \in S} (a_{ij} - \pi) = \sum_{i,j \in S} a_{ij} - \pi |S|^{sq} \tag{2}$$

where $|S|^{sq}$ denotes either $|S|(|S|-1)$ if the diagonal entries a_{ii} are absent from the criterion so that only $i \neq j$ are taken in the sum, or $|S|^2$, otherwise. The last expression in (2) is easily obtained with little arithmetic.

The value of similarity shift π can be considered a "soft threshold" so that i and j should be put in the same cluster if $a_{ij} > \pi$ and rather not, otherwise. Usually the user can specify such a threshold value depending on the nature of data and clustering goal. In fact, this also reflects the extent of desired granulation of clusters, that is usually introduced with a less intuitive parameter, the cluster size. This is warranted by the following property.

Statement 1 *The optimal cluster size according to criterion (2) can only decrease when π grows.*

Proof. Let us assume that, on the contrary, the optimal S_1 at $\pi = \pi_1$ and optimal S_2 at $\pi = \pi_2$ are such that $|S_1| > |S_2|$ and $\pi_1 > \pi_2$. Because of the optimality, two obvious inequalities: $f(S_1, \pi_1) - f(S_2, \pi_1) \geq 0$ and $f(S_2, \pi_2) - f(S_1, \pi_2) \geq 0$. Summing these two inequalities together and using equations (2), we obtain, after little arithmetic, that $\pi_1|S_2|^{sq} - \pi_1|S_1|^{sq} + \pi_2|S_1|^{sq} - \pi_2|S_2|^{sq} = (\pi_1 - \pi_2)(|S_2|^{sq} - |S_1|^{sq}) < 0$, because of the assumption. This clearly contradicts the condition that the sum is not negative, and proves the statement, q.e.d..

A useful property is that the similarity matrix, which may be not symmetric originally, can be equivalently transformed to a symmetric matrix by summing it with its transpose A'.

Statement 2 *A cluster S optimizes criterion (1) over similarity matrix A if and only if S optimizes it over symmetric similarity matrix $A + A'$.*

Proof. Let us take indicator vector s for subset S so that $s_i = 1$ if $i \in S$ and $s_i = 0$, otherwise. Then criterion (1) can be equivalently rewritten as $f(S) = \sum_{i \in S} \sum_{j \in S} a_{ij} = \sum_i \sum_j a_{ij} s_i s_j$. The proof follows from the fact that $s_i s_j = s_j s_i$, q.e.d.

The statement allows us to symmetrize similarities beforehand by by putting $a_{ij} + a_{ji}$ for every a_{ij} . Therefore, from now on, let us assume that A is symmetric.

One more property of the criterion is that it leads to provably tight clusters. Let us refer to cluster S as suboptimal if, for any entity i, the value of criterion (2) can only decrease if i changes its state in respect to S. Entity i changes its state in respect to S if it is added to S, in the case that $i \notin S$, or removed from S if $i \in S$.

Statement 3 *If S is a suboptimal cluster, then the average similarity $a(i,S)$ of i with other entities in S is greater than π if $i \in S$, or less than π if $i \notin S$.*

Proof. Let us assume that $k \in S$ for a suboptimal S, but $a(k,S) < \pi$. Let us consider difference $f(S,\pi) - f(S-k,\pi)$ where $S-k$ is what remains of set S after k is removed. For the sake of simplicity, assume that diagonal entries (i,i) are absent from the sum in (2). Then it is easy to see that $f(S,\pi) - f(S-k,\pi) = -2\sum_{i \in S}(a_{ik} - \pi)$ assuming that A is symmetric. Since this is non-negative according to suboptimality of S, the following inequality holds: $\sum_{i \in S} a_{ik} \geq (|S| - 1)\pi)$. Dividing this over $(|S| - 1)$, we obtain $a(k,S) \geq \pi$, which proves one part of the statement. The other part, for $k \notin S$, can be proven similarly, q.e.d.

The proof involves a simple formula relating the values of criterion $f(S)$ in (1) (criterion (2) coincides with (1) if the similarity matrix is pre-processed by subtracting π from all its entries) at two sets differing by just one entity rhat is present in one

of the sets and absent in the other. To make a universal expression for the formula, let us use vector $z = 2s - 1$ one-to-one relating to $S \subset I$ so that $z_i = 1$ if $i \in S$ and $z_i = -1$ if $i \notin S$. Then change of state of entity k with repect to S is expressed as change of the sign of z_k. Therefore, the change of the criterion value is equal to

$$\Delta(S,k) = f(S \pm k) - f(S) = -2z_k \sum_{i \in S} a_{ik}, \tag{3}$$

under the assumption that the diagonal similarities a_{ij} are not considered and z_k in (3) corresponds to S, that is, taken before the change of sign.

Of course, the problem of finding an optimal S over a matrix A with possibly negative entries is NP-hard. A locally-optimal improvement algorithm starting from any $S \subseteq I$ can be formulated as follows:

Summary Criterion Add-and-Remove(S)

Input: matrix $A = (aij)$ pre-normalized, subset S. Output: suboptimal cluster T_S and value of the summary criterion $f(T_S)$.

1.**Initialization.** Set N-dimensional z so that $z_i = 1$ if $i \in S$ and $z_i = -1$, otherwise. Set the summary similarity equal to $f(S)$.

2.**General step.** For each entity $k \in I$, compute the value $c_k = \Delta(S,k)$ according to (3) and find $k*$ maximizing it. 3. **Test.** If $c_{k*} > 0$, change the sign of z_{k*} in vector z, after which recalculate the summary criterion by adding c_{k*} to it, and go to 2. Otherwise, go to 4. (In the case of massive data, computing the differences in (3) can be costly. Therefore, a vector of these values $c = (c_k)$ should be maintained and dynamically changed after each addition/removal step.)

4.**Output.** Define $T_S = \{i | z_i = 1\}$ and output it along with the summary criterion value.

The suboptimaly of T_S and, therefore its tightness in the sense of the statement above is warranted by the step 2 of the agorithm.

Algorithm SC AddRem(S) utilizes no ad hoc parameters, except for the π of course, so the cluster sizes are determined by the preprocessing steps. Three obvious choices for the starting S are: (a) $S = I$, (b) $S = \{i, j\}$ such that a_{ij} is the maximum in A, and (c) $S = \{i\}$ for any $i \in I$. In the case (c), running a loop over all $i \in I$ will produce us many different clusters T_i; the structure of their overlaps gives a portrayal of the cluster structure in matrix A. One more strategy would be multiple runs of the algorithm starting from random S.

The algorithm CAST [3], popular in bioinformatics, is a version of this algorithm, in which $\Delta(S,k)$ is reformulated as $\sum_{j \in S} a_{ij} - \pi|S|$ and $\sum_{j \in S} a_{ij}$ is referred to as the affinity of i to S.

Let us apply the algorithm to the Eurovision data made symmetric by summing the matrix with its transpose, then subtracting the average non-diagonal value $\bar{a} = 35.7193$ and, afterwards, zeroing all the diagonal entries. Let us take the maximum of the final matrix A, that is, $a_{1,18} = 165.2807$. Threfore, let us start with the corresponding cluster $\{1,18\}$= { Azerbajan, Ukraine}. Then the following entities are added one-by-one because of positive summary similarities: 14. Russia, 267.6; 8. Israel, 162.8; 15. Serbia, 165.1; 7. Greece, 164.4; 13. Romania, 239.7; 3. Bulgaria, 195.0 (the summary similarity to the cluster follows the country name). So

far, the maximum of the summary similarity of S with remaining entities has been the maximum of the entire similarity matrix. From now on, this is not so. The next entity to join in is 9. Italy with the summary similarity 59.2, which is somewhat smaller than the maximum of similarities 65.3, between Italy and France.

This makes a diference. If a partitional algorithm is run, making clusters in parallel, with the summary similarity criterion, then it would build another cluster simultaneously. The other cluster(s) would compete with S in attracting entities so that further steps could have been impossible. This is exactly the case being reported. Instead of joining S, Italy would have started a different cluster altogether. Indeed, when run in parallel, say, with agglomeration steps, the summary similarity criterion leads to one more meaningful cluster T=$\{$France, Germany, Italy, Portugal, UK$\}$.

As we build a single suboptimal cluster for the summary similarity criterion, Italy goes in, after which one more entity with a positive summary similarity to S, 32.8, Portugal is added to S. Now the computation halts, because each of the yet unclustered entities has a negsative summary similarity to S. It should be pointed out that the cluster S is not exactly homogeneous: it contains East-European members somewhat attenuated by a few mediterranean countries.

2.2 Subtraction of random interactions

Let us assign each entity $i \in I$ with a probability of interaction with other entities, equal to the proportion of the summary similarity in i-th row in the total sum of the similarity values. Then random interactions between two entities will occur with the probability equal to the product of their respective probabilities. Under this interpretation, the random interactions should be subtracted from the similarity values to clear up the "nonrandom part of the total similarity structure".

The summary criterion $f(S)$ at the similarities pre-processed in this way is but what is referred to as the modularity of S. The concept of modularity as a clustering criterion was introduced in Newman and Girvan [33], Newman [32], who use it for partitioning, not for individual cluster finding.

All the contents of the previous section holds for the modularity criterion except Statements 1 and 2. The former is irrelevant because there is nothing variant in the modularity transformation; one may explore, though, putting a changeable factor to the random interactions. The latter should be reformulated by using the so-called modularity attraction (see detail in [25], p. 327).

Multiple runs of Add-and-Remove(i) at different starting points $i \in I$ allow to (a) find a better cluster S maximizing the summary similarity criterion over the runs, and (b) explore the cluster structure of the dataset by analyzing both differing and overlapping clusters.

Let us apply this approach to the Eurovision matrix. To do that, we compute the sums of the rows, the sum of the columns, and the total. Then we subtract from every entry a_{ij} the product of the sums of the corresponding row and column related to the total, $a_{i+}a_{+j}/a_{++}$. In the resulting matrix, all the diagonal entries are made

zero, after which the matrix is summed with its transpose to make it symmetric. The summary similarity criterion of subset S for this matrix is what is referred to as the modularity of S (doubled, because of the symmetrical summation) [32].

Let us build a cluster starting from $S = \{1\}$. Once again the maximum positive similarity to 1 is 18, at 127.7. Therefore, we have now S= {Azerbajan, Ukraine}. Russia, number 14, has the maximum positive summary similarity to this, 215.7, which makes S= {Azerbajan, Ukraine, Russia}. But then the order of nearest entities changes (from the case of uniform similarity criterion). Next joining is 11. Poland at 107.5; 4. Estonia at 127.5; 8. Israel at 66.4; 10. The Netherlands at 9.0; and 2. Belgium at 22.8. Here the process stops – all the other entities have negative summary similarity to the final S. At no step was any need to remove an entity; all the summary within cluster similaritiers have been positive. This, probably, is even more controversial agglomeration than that found at the uniformly pre-processed data. This is probably because the row and column totals here highly affect the result of data preprocessing so that even relatively small similarity values can remain positive if the totatls are small enough.

There exists even more powerful transformation of the similarity data involving the products of the row and column totatls, the so-called pseudo-inverse Laplace transformation (see, for example, [26]), which frequently somewhat sharpens the structure hidden in data, but not always, unfortunately.

3 Semi-average criterion

Another seemingly natural individual clustering criterion is the average similarity

$$a(S) = \frac{\sum_{i,j \in S} a_{ij}}{|S|(|S|-1)}. \tag{4}$$

This definition relates to the case at which no diagonal elements of A are involved. Unfortunately, this criterion cannot work for clustering because its maximum is reached at a pair $\{i, j\}$ at which a_{ij} is the maximum value in A – this is the average similarity in this set. Indeed, further addition of other entities can only decrease the average similarity.

As a remedy, $a(S)$ can be multiplied by $|S|-1$ so that the decrease in $a(S)$ can be compensated by the increase in the cardinality $|S|$:

$$b(S) = \frac{\sum_{i,j \in S} a_{ij}}{|S|} = (|S|-1)a(S) \tag{5}$$

We refer to this as the semi-average criterion. In spite of a rather long history of using this criterion (see, for instance, [25, 26]), this case has never been explicitly analyzed; all the derivations referred to the situation of the diagonal entries present so that the denominator is $|S|^2$, leading to simpler formulas.

Consider the change of (5) when entity k changes its state over S (again, entries a_{ii} are considered non-existent. Once again, vector $z = (z_i)$ is invoked, with $z_i = 1$, if $i \in S$, and $z_i = -1$, otherwise.

Statement 4 *The change of $b(S)$ when entity k changes its state with respect to non-singleton S is*

$$b(S \pm k) - b(S) = z_k[(|S| + z_k)a(S) - 2(|S| + (1 + z_k)/2)a(k,S)]/(|S| + 1) \quad (6)$$

where $a(k,S)$ is the average similarity between entity k and entities in S, and $a(S)$, the average within-S similarity (4).

Proof. Let $k \in S$ so that $z_k = 1$. Then the difference is

$$b(S-k) - b(S) = [\sum_{i,j \in S} a_{ij} - 2\sum_{i \in S} a_{ik}]/(|S| - 1) - \sum_{i,j \in S} a_{ij}/|S| =$$

$$[|S| \sum_{i,j \in S} a_{ij} - 2|S| \sum_{i \in S} a_{ik} - (|S| - 1) \sum_{i,j \in S} a_{ij}]/(|S|(|S| - 1)) =$$

$$[\sum_{i,j \in S} a_{ij} - 2|S| \sum_{i \in S} a_{ik}]/(|S|(|S| - 1)) = a(S) - 2a(k,S)$$

The only place in these rather dull derivations that probably deserves a comment is that the average $a(k,S)$ is computed over $|S| - 1$ entities, not $|S|$ because $k \in S$ and a_{kk} is not taken into account. The final expression is exactly (6) at $z_k = 1$.

Assume now that $k \notin S$ and take the difference

$$b(S+k) - b(S) = [\sum_{i,j \in S} a_{ij} + 2\sum_{i \in S} a_{ik}]/(|S| + 1) - \sum_{i,j \in S} a_{ij}/|S| =$$

$$[|S| \sum_{i,j \in S} a_{ij} + 2|S| \sum_{i \in S} a_{ik} - (|S| + 1) \sum_{i,j \in S} a_{ij}]/(|S|(|S| + 1)) =$$

$$[2|S| \sum_{i \in S} a_{ik} - \sum_{i,j \in S} a_{ij}]/(|S|(|S| + 1)) = [2|S|a(k,S) - (|S| - 1)a(S)]/(|S| + 1)$$

It is not difficult to see that the final expression corresponds to (6) at $z_k = -1$. This completes the proof, q.e.d.

Let us refer to expression

$$\alpha(k,S) = a(k,S) - a(S)/2 \quad (7)$$

as the attraction of entity k to subset S. A tight cluster S should have all the attractions of its elements positive and attractions of external entities negative.

This is exactly the case for clusters that are suboptimal over the semi-average criterion.

Statement 5 *If a subset $S \subset I$ is subpotmal over criterion $b(S)$ in (5), then for each entity $k \in I$ its attraction to S is not negative if $k \in S$ and not positive if $k \notin S$.*

The proof easily follows from the fact that all the differences (6) must be not positive for a suboptimal S. If, for example, $k \in S$, that is, $z_k = 1$, then $b(S-k) - b(S) = a(S) - 2a(k,S) \leq 0$ and, therefore, $-2\alpha(k,S) \leq 0$.

Let us note that the statement much resembles that for the summary similarity criterion at threshold $\pi = a(S)/2$. The difference is that π is an expert-given value, whereas half the inner average similarity $a(S)/2$ is determined by cluster itself. This is not just a chance co-occurrence. Next section will show more general criterion of which these two are just part.

Of course, algorithm Semi-average criterion Add-and-Remove(S) works exactly as defined before except that this time the change of the criterion is computed using formula (6).

As an example, let us apply the Add-and-Remove algorithm for finding a suboptimal cluster of Eurovision data for the Semi-average criterion, starting again from entity 1. Azerbajan. Matrix A is pre-processed by summing it with its transpose, subtracting the average of its non-diagonal entries, and by zeroing the diagonal. The algorithm produces cluster $S = \{1,3,7,14,15,18\}$, that consists of countries: Azerbajan, Bulgaria, Greece, Russia, Serbia, Ukraine. This looks a culturally cohesive cluster: all four Slav Christian Orthodox countries plus Greece, which is Christian Orthodox, and Azerbajan, a former Soviet republic. Take a look at the process: first comes 18. Ukraine, then 14. Russia, then 8. Israel. Then, one by one, 15. Serbia, 7. Greece, 13. Romania, 3. Bulgaria are added. After this the attractions of 8. Israel and 12. Romania become negative, and they are removed one by one. Both the process and result show a relative versatility of the semi-average criterion.

4 Approximation models for summary and semi-average criteria

4.1 Median in Hamming distance space

Consider the set of all binary relations on I that are subsets of cartesian product $I \times I$, that is set of all ordered pairs (a,b), $a,b \in I$. Among them those one-to-one corresponding to subsets $S \subseteq I$ are of format $\tau_S = S \times S$. Converted to the format of $|I| \times |I|$ binary matrices, a binary relation $\rho \in I \times I$ is represented by matrix $r = (r_{ij})$ in which $r_{ij} = 1$ if $(i,j) \in \rho$ and $r_{ij} = 0$, otherwise. A binary matrix t_S corresponding to the "square" binary relation τ_S corresponding to subset S has a clear-cut block structure: it is all zero, except for $t_{ij} = 1$ if and only if both $i, j \in S$. Consider Hamming distance between such binary matrices, $d(r,s) = \sum_{i,j \in I} |r_{ij} - s_{ij}|$, which is in fact what is referred to as the city-block distance or, in the binary case, the squared Euclidean distance between r,s as $|I| \times |I|$-dimensional vectors [21]. Given n binary relations on I, $r^1, r^2, ..., r^n$, a median subset S is defined as corresponding to a "clear-cut block" matrix $s = (s_{ij})$ minimizing the summary distance $\sum_{m=1}^{n} d(s, r^k)$. Define

concensus matrix $A = \sum_k^n r^k$ so that a_{ij} is the number of those among the given relations that relate to (i, j), that is, $r_{ij}^k = 1$. Then the problem of finding a median is of finding a cluster according to the uniform summary similarity criterion.

Statement 6 *Subset S is a median if and only if it maximizes the uniform summary similarity criterion (2) at threshold value* $\pi = n/2$.

The proof rather obviously follows from the definition and the fact that $|u - v| = (u - v)^2$ for 1/0 variables u, v.

In a specific case of $n = 1$ the statement can be interpreted as this. Consider this problem: given an ordinary graph, transform it into a complete subgraph (with all outside nodes being isolated) by adding or removing edges so that the number of changes is minimum. This problem is equivalent to the following. Consider a 1/-1 $|I| \times |I|$ matrix with its entries being one, if the corrsponding edge is in the graph, and minus one, if not. Then finding a cluster S maximizing $f(S)$ in (1) over this matrix is the same problem. Of course, this is a hard problem, equivalent to that of finding a complete subgraph with the maximum number of nodes.

4.2 Least-squares approximation

The idea is to find such a subset $S \subseteq I$ that its binary matrix $s = (s_{ij})$ approximates a given symmetric similarity matrix A as well as possible. To accommodate for the difference in the unit of measurement of the similarity as well as for its zero point, matrix s should be also supplied with (adjustable) scale shift and rescaling coefficients, say λ and μ. That would mean that the approaximation is sought in the set of all binary $\lambda + \mu$ / μ matrices $\lambda s + \mu$ with $\lambda > 0$. Unfortunately, such an approximation, at least when follows the least squares approach, would have little value as a tool for producing a cluster, because the optimal values for λ and μ would not separate the optimal S from the rest. This is why from about 1995, this author uses only one parameter λ, change of the unit of measurement, in formulating approximation problems in clustering. The issue of adjustment of similarity zero point, in such a setting, is moved out of the modeling stage to the data pre-processing stage. Basically, this amounts to the need in substraction of a similarity shift value before doing data analysis. Choice of the similarity shift value may affect the clustering results, which can be of advantage in contrasting within- and between- cluster similarities. Figure 1 demonstrates the effect of changing a positive similarity a_{ij} to $a'_{ij} = a_{ij} - \lambda_0$ for $\lambda_0 > 0$; small similarities $a_{ij} < \lambda_0$ are transformed into negative similarities a'_{ij}.

Therefore, in the remainder of this section, it is assumed that a similarity shift value has been subtracted from all the similarity entries. Another assumption, for the sake of simplicity, is that the diagonal entries a_{ii} are all zero (after the pre-processing step). One more simplifying assumption is that the subset $S \subseteq I$ now is presented with a binary vector rather than binary matrix. From now on, S is represented by a vector $s = (s_i)$ such that $s_i = 1$ if $i \in S$ and $s_i = 0$, otherwise.

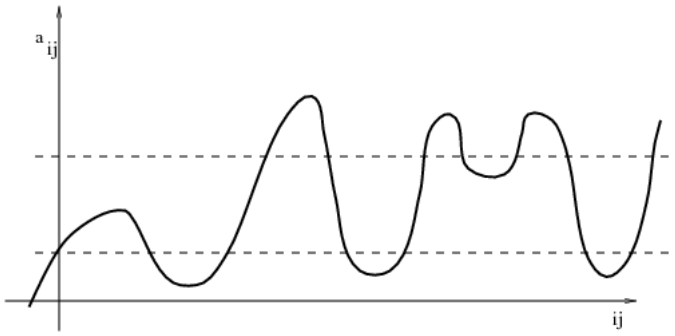

Fig. 1 A pattern of clustering depending on the subtracted similarity shift λ_0.

Therefore, our approximation model is

$$a_{ij} = \lambda s_i s_j + e_{ij} \tag{8}$$

where a_{ij} are the preprocessed similarity values, $s = (s_i)$ is the unknown cluster belongingness vector and λ, the rescaling value, also referred to as the cluster intensity value. To fit the model (8), various criteria can be utilized. Only the least squares criterion $L^2 = \sum_{i,j \in I} e_{ij}^2$ is considered here.

4.2.1 Pre-specified Intensity

We first consider the case in which the intensity λ of the cluster to be found is pre-specified. Remembering that $s_i^2 = s_i$ for any 0/1 variable s_i, the least squares criterion can be expressed as

$$L^2(S,\lambda) = \sum_{i,j \in I} (a_{ij} - \lambda s_i s_j)^2 = \sum_{i,j \in I} a_{ij}^2 - 2\lambda \sum_{i,j \in I} (a_{ij} - \lambda/2) s_i s_j \tag{9}$$

Since $\sum_{i,j} a_{ij}^2$ is constant, for $\lambda > 0$, minimizing (9) is equivalent to maximizing the summary within-cluster similarity after subtracting the threshold value $\pi = \lambda/2$, i.e.,

$$f(S,\pi) = \sum_{i,j \in I} (a_{ij} - \pi) s_i s_j = \sum_{i,j \in S} (a_{ij} - \pi). \tag{10}$$

This is exactly the summary similarity criterion considered above in section 2.1.

4.2.2 Optimal Intensity

When λ in (9) is not fixed but can chosen to further minimize the criterion, it is easy to prove that the optimal λ is equal to the average within-cluster similarity:

$$\lambda = a(S) = s^T A s / [s^T s]^2, \tag{11}$$

The second equation follows from the fact that $s^T s = |S|$.

By putting this equation in the least-squares criterion (9) it is not difficult to derive a Pythagorean decomposition:

$$L^2(S) = (A,A) - [s^T As/s^T s]^2, \tag{12}$$

where A plays the role of hypothenuse. This decomposition implies that the optimal cluster S is a maximizer of a criterion depending on S only:

$$g^2(S) = [s^T As/s^T s]^2 = a^2(S)|S|^2 \tag{13}$$

According to (13), the maximum of $g^2(S)$ may correspond to either positive or negative value of $a(S)$. The latter case may emerge when the preliminarily subtracted similarity shift is large and corresponds to S being the so-called *anti-cluster* [24]. An anti-cluster consists of entities that all are mutually far away from each other. Therefore, we skip this latter case but rather focus on maximizing (13) only for positive $a(S)$. This is equivalent to maximizing its square root, that is the Rayleigh quotient,

$$g(S) = s^T As/s^T s = a(S)|S| \tag{14}$$

This criterion is a form of the semi-average clustering criterion considered in section 3.

Another form would be obtained for the case at which similarities of entities to themselves, a_{ii}, are not defined. In this case, the criterion (13) would be $a^2(S)|S|(|S|-1)$ so that $g(S)$ would involve the square root $\sqrt{|S|(|S|-1)}$, thus leading to a bit more complicated formulas in the corresponding local optimization algorithm Add-and-Remove.

Therefore, one can see that both heuristic criteria considered in the beginning are related to the least-squares approximation criterion (9). At a pre-specified λ, the summary similarity clustering criterion with the uniformly subtracted $\pi = \lambda/2$ is obtained. At the optimal λ, the semi-average similarity clustering criterion emerges. But in fact it is more than just the criterion. Because of the data scatter decomposition in (12), its square, $g^2(S)$ expresses the share of the similarity data scatter $(A,A) = \sum_{i,j} a_{ij}^2$ "explained", or better to say, "taken into account" by the cluster S and its intensity $\lambda = a(S)$ found by (locally) optimizing $g(S)$. This allows one to judge how well the cluster reflects the structure of A.

The least-squares criterion $L^2(S,\lambda)$ itself can be considered as justifying alternating application of Add-and-Remove to the summary similarity criterion. Each iteration of this starts at the within-cluster average value of λ and corresponding threshold $\pi = \lambda/2$ and finishes with finding a suboptimal S. The computation stops at the average λ coinciding with its previous value. In the beginning, λ can be taken as the average similarity over I if no expert-driven hypothetical value is available.

4.3 Partitional, additive and incjunctive clusters: iterative extraction

A similar approximational model can be considered as underlying a set of (not necessarily disjoint) similarity clusters $S_1, S_2, ..., S_K$. Specifically, let a similarity cluster is specified by its binary belongingness vector $s = (s_i)$ and a positive real λ, the cluster intensity, to make the binary scale compatible with that of similarity A. Then the model is defined by equations

$$a_{ij} = \biguplus_{k=1}^{K} \lambda_k s_i^k s_j^k + e_{ij}, \quad \text{for } i, j \in I, \tag{15}$$

where $s^k = (s_i^k)$ and λ_k are k-th cluster belongingness vector and its intensity. The symbol $\biguplus$ denotes an operation of integration of the binary values together with their intensities. We consider three versions of the operation: (a) additive clusters: $\biguplus$ is just summation; (b) partitional clusters: $\biguplus$ denotes the fact that clusters are disjunct, no overlapping; (c) incjunctive clusters: $\biguplus$ is maximum over $k = 1, 2, ..., K$, that is, operation of inclusive disjunction.

The goal is to minimize the residuals e_{ij} with respect to the unknown relations R^k and intensities λ_k.

Additive cluster model was introduced, in the English language literature, by Shepard and Arabie in [37], and independently in a more general form embracing other cluster structures as well, by the author in mid-seventies in Russian ([21], see references in [22]). Incjunctive clusters have not been considered in the literature, to our knowledge.

We maintain that cluster structures frequently are similar to that of the Solar system so that clusters hidden in data much differ with respect to their "contributions". Then the iterative extraction method ([21, 26]) can be applied to find clusters one by one. Depending on the setting, that is, meaning of $\biguplus$ in (15), one may use the following options:

i **Additive clusters.** The iterative extraction works as this:

a. Initialization. Given a preprocessed similarity matrix A, compute the data scatter $T = (A, A)$. Put $k = 0$.
b. General step. Add 1 to k. Find cluster S (locally) maximizing criterion $g(S)$ in (14). Output that as S_k, the intensity of this cluster, the within-cluster average $a(S)$ as λ_k, and its contribution to the data scatter, $w_k = a(S)^2|S|^2$.
c. Test. Check a stopping condition (see below). If it does hold, assign $K = k$ and halt. Otherwise, compute the residual similarity matrix as $A - \lambda_k s_k s_k^T$ and go back to General step with the residual matrix as A.

The stopping condition can be either reaching a prespecified number of clusters or contribution of the individual cluster has become too small or the total contribution of the so far found clusters has become too large. The individual cluster

contributiona are additive in this process. Moreover, the residual matrix in this process tends to 0 when k increases [24].

ii **Partitional clusters** This method works almost like the iterative extraction at the additive clustering model, except that here no residual matrix is considered, but rather the found clusters are removed from the set of entities.

   a. Initialization. Given a preprocessed similarity matrix A, compute the data dcatter $T = (A,A)$. Put $k = 0$.
   b. General step. Add 1 to k. find cluster S (locally) maximizing criterion $g(S)$ in (14). Output that as S_k, the intensity of this cluster, the within-cluster average $a(S)$ as λ_k, and its contribution to the data scatter, $w_k = a(S)^2|S|^2$.
   c. Test. Check a stopping condition (see below). If it does hold, assign $K = k$ and halt. Otherwise, compute the residual entity set $I = I - S_k$ and remove S_k from the similarity matrix. Go back to General step.

   The stopping condition can be either reaching equation $I = S_k$ or the situation at which the matrix A has no positive entries on the set of yet unclustered entities. The individual cluster contributiona are additive in this process.

iii **Incjunctive clusters.** Make a loop over $i \in I$. Run semi-average criterion Add-and-Remove(S) algorithm at $S = \{i\}$ for each i. Remove those of the found clusters that overlap with others too much. This can be done by applying the same algorithm to the cluster-to-cluster similarity matrix; entries in this matrix are defined as proportional to the overlap values. The individual cluster over this matrix contains those clusters that overlap too much - only one of them shoild be left.

For an example, let us apply each of these three strategies to the Eurovision matrix, preliminarily made symmetric with zeroed diagonal entries.

a Additive clusters one by one: With the condition to stop when the contribution of an individual cluster becomes less than 1.5% of the total data scatter, the algorithm found, in addition to the universal cluster I with the intensity equal to the similarity average, six more clusters (see Table 3). We can see that, say, pair

Table 3 Additive clusters found at the Eurovision song contest dataset.

n.	Cluster	Intensity	Contribution, %
1	Azerbajan, Bulgaria, Greece, Russia, Serbia, Ukraine	70.0	21.43
2	Azerbajan, Israel, Romania, Russia, Ukraine	49.5	7.13
3	Bulgaria, Greece, Italy, Romania, Spain	46.8	6.38
4	Azerbajan, Poland, Ukraine	66.8	3.90
5	Italy, Portugal, Romania	53.0	2.46
6	Greece, Romania, Serbia	43.7	1.67

Azerbajan and Ukraine belong to three of the clusters and contribute, therefore, the summary intensity value 70.0+49.5+66.8=186.3 as the "model" similarity between them. This is, by the way, is greater than the observed similarity between them, 165.3.

b Partitional clusters one by one. Here the algorithm is run on the entities remain unclustered after the previous step (see Table 4).

Table 4 Partitional clusters found one-by-one at the Eurovision song contest dataset.

n.	Cluster	Intensity	Contribution, %
1	Azerbajan, Bulgaria, Greece, Russia, Serbia, Ukraine	70.0	21.43
2	Italy, Portugal, Romania, Spain	56.1	5.50
3	Belgium, Netherlands	57.3	0.96
4	Germany, UK	45.3	0.60
5	France, Israel, Switzerland	11.6	0.12
6	Estonia, Poland	3.3	0.00

Here, in fact, there are only two meaningful clusters, East European and Latin South European; the other four contribute so little that should not be considered at all as possibly being "noise-generated". The first of the clusters is just a replica of that in the additive clustering computation. Yet the second cluster is a different phenomenon, a combination of clusters 3 and 5 in the additive clusters results Table 3 cleaned of the Balkans.

c Incjunctive clusters from every entity. The semi-average Add-And-Remove(S) algorithm has been applied starting from $S = \{i\}$ for every $i \in I$. Most of the final clusters coincide with each other, so that there are very few different clusters. A surprising feature of the computation is that the algorithm sometimes moved out of the starting entity to finish with a cluster from which the entity was absent. Table 5 presents different clusters only.

Table 5 All four different incjunctive clusters found at the Eurovision song contest dataset starting from every entity.

	Cluster	Intensity	Contribution, %
1.	Azerbajan, Bulgaria, Greece, Russia, Serbia, Ukraine	70.0	21.43
2.	Belgium, Netherlands	57.3	0.96
3.	Bulgaria, Greece, Serbia	110.6	10.7
4.	Italy, Portugal, Romania, Spain	56.1	5.50

According to the data recovery model, these clusters lead a recovered similarity matrix as follows: first of all, the subtracted average value, 35.72, should be put at every entry. Then the two entries of Belgium/Netherlands link are to be increased by the intensity of cluster 2, 57.3. Similarly, the intensities of clusters 1 and 4 are to be added for any pair of entities within each. Then entries for pairs from cluster 3 are to be changed for 35.7+110.6=146.3.

It should be reminded that of the four clusters in Table 5, only one is globally optimal, the first cluster since the criterion is co-monotone with the contribution to data scatter, which is maximum at cluster one. Therefore, the other clusters have been found only because of the local nature of Add-and-Remove algorithm.

This is an example at which the local nature of the algorithm is of an advantage rather than a drawback. Because of the local nature, one can explore the local clustering structure. One can see that in this case, the structure is rather sharp, no intermediate clusters at all! Many entities remain out of the cluster structure. One of the clusters is of special interest, cluster 3 consisting of Bulgaria, Greece and Serbia - hard-core Balkan countries. This comes on top of a less intensive cluster 1, containing all of the cluster 3. One can notice that this structure reflects more than 10 years of voting, and thus probably has nothing to do with the "quality" of songs presented to the contest. If so, the clusters reflect cultural interrelations rather than anything else.

On a more personal note, I can recall that a few years ago the media in the United Kingdom made an issue of the humble history of failure of the UK songs in the contest and decided to go for a win. They attracted best composers, best performers and best relation managers - all this to arrive at one more disappointing failure. Looking at the clusters above, one cannot help but joining the scores of pessimists: the UK has no chances at the Eurovision song contest, unless the rules are changed to accommodate the cluster structure.

5 Applications

5.1 Semantics of domain-specific nouns

The idea that semantics of domain-specific nouns lies in their relation to specific situations, functions, etc., a few decades back was not that obvious in cognitive sciences as it is now. In the absence of Internet, the researchers used the so-called sorting experiments to shed light on semantics of domain specific nouns [34, 8]. In a sorting experiment, a set of domain-specific words is specified and written down, each on a small card; a respondent is asked then to partition cards into any number of groups according to their perceived similarity among the nouns. Then. a similarity matrix between the words can be drawn so that the similarity score between two words is defined as the number of respondents who put them together in the same cluster. A cognitive scientist may think that behind the similarity matrix can be some "additive" elementary meanings. Say, for a group of kitchenware terms, such as, say, "glass", "cup", "casserole" and "saucepan", there is a common elementary meaning that they all are water reservoirs plus a meaning - cookware relating the two latter items. Depending on the social and cultural background, word clusters expressing the same meaning may have this or that intensity weight. The weights of different meanings relating two terms, are added together in the mind when considering them as a whole: this sounds a plausible hypothesis. Therefore, the additive cluster model can be adequate for the analysis of sorting experiment results

In the analysis of similarities between 72 kitchenware terms, the iterative one-by-one extraction with Semi-average similarity Add-and-Remove algorithm found that

none of the clusters reflected logical or structural similarities between the kitchenware items; all the clusters related to the usage only. Specifically, three types of term communality were manifested in the clusters: (i) a cooking process, such as frying or boiling; (ii) a common consumption use, such as drinking or eating, and (iii) a common situation such as a banket [8].

Let us show in a bit more detail, how the individual clusters can be found at the matrix of similarity scores between elementary functions in Table 1. The average of non-diagonal entries in it is $\bar{a} = 2.56$. After subtracting this from the data the resulting similarity matrix upper triangle is:

	2	3	4	5	6	7	8	9
1	3.44	−1.56	−1.56	−0.56	0.44	−0.56	−0.56	−0.56
2		−1.56	−1.56	−0.56	−0.56	1.44	1.44	0.44
3			3.44	−1.56	−1.56	−1.56	−1.56	−1.56
4				−1.56	−1.56	−1.56	−1.56	−1.56
5					3.44	2.44	2.44	2.44
6						1.44	0.44	−0.56
7							2.44	0.44
8								−0.56

Since we are going to apply the iterative extraction method, the subtraction of 2.56 from the original data can be considered as transition to the residual matrix after the universal cluster, consisteing of all the entities, is extracted. The contribution of the universal cluster to the original similarity data scatter is 207.8/676=30.4%.

Let us find an individual cluster, starting from $S = \{5\}$. Entity 6 has the maximum similarity with S, $a_{56} = 3.44$. After it is added, $S = \{5,6\}$. Let us find an entity that has the largest positive summary similarity to S. Entities 1, 2, 3 and 4 all have negative summary similarity to S. The summary similarity to S is equal to 2.44+1.44=3.88 for entity 7, 2.44+0.44=2.88 for 8, and 2.44-0.56=1.88 for 9. This leads to entity 7 being the candidate to be added to S. The average similarity from 7 to S, 3.88/2=1.94 is greater than half the within-S similarity, 3.44/2=1.72, so that 7 joins S. Now $S = \{5,6,7\}$ and the positive summary similarities with S are 2.44+0.44+2.44=5.32, for 8, and 2.44 - 0.56+0.44=2.32, for 9. The candidate 8 has its average similarity $a(8,S) = 5.32/3 = 1.77$. The average within-S similarity is (3.44+2.44+1.44)/3=7.32/3=2.44, so that its half, 2.44/2=1.22, is less than 1.77. Therefore, 8 joins in S. The remaining entity 9 summary similarity to $S = \{5,6,7,8\}$ is 2.44-0.56+0.44-0.56=1.76. The average similarity $a(9,S) = 1.76/4 = 0.44$ is less than the half of the within-S average similarity, $a(S) = 2.11$, so that 9 cannot join S. Moreover, no entity should be removed from S, because each its element has a positive attraction to S. This leads to halting the process and outputting $S = \{5,6,7,8\}$, together with its intensity $a(S) = 2.11$ and the contribution $w(S) = a(S)^2 * 16/676 = 10.55\%$. It is not difficult to see that this S is the best among all the clusters found at different j.

Table 6 presents 6 additive clusters found with the one-by-one process of additive clustering. The total contribution to data scatter is about 60%. It should be noted

that optimally adjusting weights of the six clusters with the orthogonal projection of matrix $A - \bar{a}$ over binary matrices of the clusters would drastically increase that, up to about 94% of the original data scatter. But in that case the additive property of individual contributions would be lost.

Table 6 Additive clusters found at Elementary functions dataset.

Cluster	Intensity	Contribution, %	Interpretation
1. 1 – 9	2.56	30.44	Functions
2. 5,6,7,8	2.11	10.55	Power: growing
3. 1,2	3.44	7.02	Natural
4. 3,4	3.44	7.02	Power: decreasing
5. 5,9	2.44	3.54	Growing even
6. 2,7,8	1.07	1.53	Slow growing

Unfortunately, the contribution weights do not lead to any sound advice for halting the process. Here, the process stops at six clusters as having reached the contribution of an individual cluster of 1.5%. It is nice that all the found clusters have more or less reasonable, and in fact exclusive, interpretations.

The same process, under the constraint that the next clusters are not to overlap those previously found, would lead to just three clusters in Table 6, nn. 2, 3, and 4, leaving function 9 a singleton.

5.2 Determining similarity threshold by combining knowledge

In Mirkin et al. (2010) partitional clusters of protein families in herpes viruses are found. The similarity between them is derived from alignments of protein amino acid sequences. Yet in viruses, the amino acid contents is highly changeable even in related genomes. This is why the similarity in [27] is measured over their neighborhoods [40, 17], that are subsets of proteins. The similarity between sets S and T is measured by the ratios $|S \cap T|/|S|$ and $|S \cap T|/|T|$ as well as by their average $mbi(S,T) = (|S \cap T|/|S| + |S \cap T|/|T|)/2$, which goes in line with the symmetric reformulation of the raw similarity data in Statement 2. This index spans interval from 0, no overlap, to 1, full coincidence.

At different similarity shifts, different numbers of clusters can be obtained, from 99 non-singleton clusters (of 740 entities) at the zero similarity shift to only 29 non-singleton clusters at the shift equal to 0.97 [27]. To choose a proper value of the shift, external information can be used – of functional activities of the proteins under consideration in [27]. Although function of most proteins under consideration was unknown, the set of pairs of functionally annotated proteins can be used to shed light onto potentially admissible values of the similarity shift. In each pair, the proteins can be synonymous (sharing the same function) or not. Because of a high simplicity of virus genomes, the synonymous proteins should belong in the

same aggregate protein family, whereas proteins of different functions should belong in different protein families. The similarity shift value should be taken as that between the sets of similarity values for synonymous and nonsynonymous proteins. Then, after subtraction of this value, similarities between not synonymous HPFs get negative while those between synonymous HPFs remain positive. In [27] no non-synonymous pair has a greater *mbc* similarity than 0.66, which should imply that the shift value 0.67 confers specificity for the production of aggregate protein families Unfortunately, the situation is less clear cut for synonymous proteins: although the similarities between them indeed are somewhat higher, 24% pairs is less than 0.67. To choose a similarity shift that minimizes the error in assigning negative and positive similarity values, one needs to compare the distribution of similarity values in the set of synonymous pairs with that in the set of non-synonymous pairs (Figure 2) and derive the intersection point similarity value as 0.42.

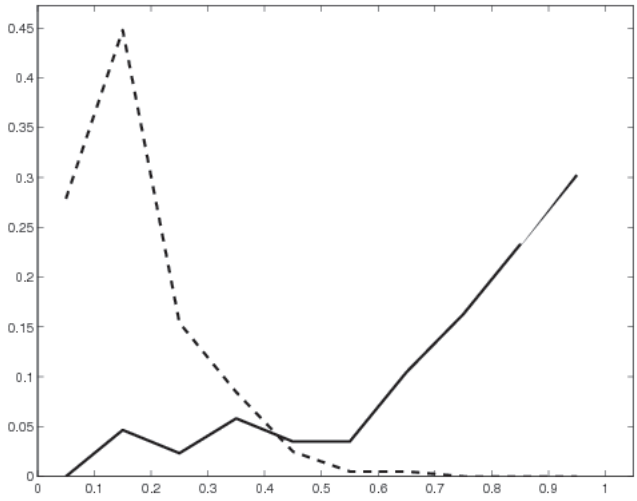

Fig. 2 Empirical percentage frequency functions (y-values) for the sets of synonymous pairs (solid line) and non-synonymous pairs (dashed line). The x-values represent the *mbc* similarity.

Thus the external knowledge of functional synonymity/non-synonymity reduces the set of candidate similarity shift values to two:

(a) $\lambda_0 = 0.67$ to guarantee specificity in that non-synonymous proteuns are not be clustered together, and
(b) $\lambda_0 = 0.42$ to ensure the minimum misclassification error rate.

The final choice, $\lambda_0 = 0.42$, has been made over yet another type of external information: compatibility between the evolutionary histories of protein families derived in the framework of the maximum parsimony principle [**?**, 27] and the structure of gene arrangement in the genomes.

5.3 Consensus clustering

Consensus clustering is an activity of summarizing a set of clusterings into a single clustering. This has become popular recently because after applying different clustering algorithms, or the same algorithm at different parameter settings, on a data set, one gets a number of different solutions. Consensus clustering seeks a unified cluster structure behind the solutions found (see, for example, [42, 26]). This author has contributed to the problem a few dozen decades ago, in Russian. Here some results of applying an approach from Mirkin and Muchnik [30] in the current setting will be reported. Some supplementary aspects of the least-squares consensus clustering are described in [26].

Consider a partition $S = \{S_1, ..., S_K\}$ on I and corresponding binary membership $N \times K$ matrix $Z = (z_{ik})$ where $z_{ik} = 1$ if $i \in S_k$ and $z_{ik} = 0$, otherwise ($i = 1, ..., N, k = 1, ..., K$). Obviously, $Z^T Z$ is a diagonal $K \times K$ matrix in which (k,k)-th entry is equal to the cardinality of S_k, $N_k = |S_k|$. On the other hand, $ZZ^T = (s_{ij})$ is a binary $N \times N$ matrix in which $s_{ij} = 1$ if i and j belong to the same class of S, and $s_{ij} = 0$, otherwise. Therefore, $(Z^T Z)^{-1}$ is a diagonal matrix of the reciprocals $1/N_k$ and $P_Z = Z(Z^T Z)^{-1} Z^T = (p_{ij})$ is an $N \times N$ matrix in which $p_{ij} = 1/N_k$ if both i and j belong to the same class S_k, and $p_{ij} = 0$, otherwise. Matrix P_Z represents the operation of orthogonal projection of any N-dimensional vector x onto the linear subspace $L(Z)$ spanning the columns of matrix Z.

A set of partitions R^u, $u = 1, 2, ..., U$, along with the corresponding binary membership $N \times L_t$ matrices X^u, found with various clustering procedures, can be thought of as proxies for a hidden partition S, along with its binary membership matrix Z. Each of the partitions can be considered as related to the hidden partition S by equations

$$x^u_{il} = \sum_{k=1}^{K} c^u_{kl} z_{ik} + e^u_{ik} \tag{16}$$

where coefficients c^u_{kl} and matrix z_{ik} are to be chosen to minimize the residuals e^u_{ik}.

By accepting the sum of squared errors $E^2 = \sum_{i,k,u} (e^u_{ik})^2$ as the criterion to minimize, one immediately arrives at the optimal coefficients being orthogonal projections of the columns of matrices X^u onto the linear subspace spanning the hidden matrix Z. More precisely, at a given Z, the optimal $K \times L_u$ matrices $C^u = (c^u_{kl})$ are determined by equations $C^u = Z(Z^T Z)^{-1} X^u$. By substituting these in equations (16), the square error criterion can be reformulated as:

$$E^2 = \sum_{u=1}^{U} ||X^u - P_Z X^u||^2 \tag{17}$$

where $||.||^2$ denotes the sum of squares of the matrix elements.

The problem is to find a partition with the membership matrix Z minimizing crterion (17). This criterion seems rather original, never mentioned in the current literature, to this author's knowledge. It is not difficult to show that the criterion can be reformulated in terms of the so-called consensus similarity matrix. To this end,

let us form $N \times L$ matrix $X = (X^1 X^2 ... X^U)$ where $L = \sum_{u=1}^{U} L_u$. The columns of this matrix correspond to clusters R_l that are present in partitions $R^1, ..., R^U$. Then the least squares criterion can be expressed as $E^2 = ||X - P_Z X||^2$, or equivalently, as $E^2 = Tr((X - P_Z X)(X - P_Z X)^T)$ where Tr denotes the trace of $N \times N$ matrix, that is, the sum of its diagonal elements, and ^{T}, the transpose. By opening the parentheses in the latter expression, one can derive that $E^2 = Tr(XX^T - P_Z XX^T)$. Let us denote $A = XX^T$ and take a look at (i, j)-th element of this matrix $a_{ij} = \sum_l x_{il} x_{jl}$ where summation goes over all clusters R_l of all partitions $R^1, R^2, ..., R^U$. Obviously, a_{ij} equals the number of those partitions $R^1, R^2, ..., R^U$ at which i and j are in the same class. This matrix is referred to in the literature as the consensus matrix. The latter expression can be reformulated thus as

$$E^2 = NU - \sum_{k=1}^{K} \sum_{i,j \in S_k} a_{ij}/N_k.$$

This leads us to the following statement. This leads us to the following statement.

Statement 7 *A partition $S = \{S_1, ..., S_K\}$ is an ensemble consensus clustering if and only if it maximizes criterion*

$$g(S) = \sum_{k=1}^{K} \sum_{i,j \in S_k} a_{ij}/N_k \tag{18}$$

where $A = (a_{ij})$ is the consensus matrix between entities for the given set of partitions.

Criterion (18) is but the sum of semi-average criteria for the clusters S_1, ..., S_K. Therefore, the iterative extraction algorithm in its partitional clusters format is applicable here. We compared the performances of this algorithm and a number of up-to-date algorithms of consensus clustering (see Table 7) [38].

Table 7 Consensus clustering methods involved in the experiments.

n.	Method	Author(s)	Reference
1	Bayes	Wang et al.	[44]
2	Vote	Dimitriadi et al.	[6]
3	CVote	Ayad, Kamel	[1]
4	Borda	Sevillano et al.	[**?**]
5	Fusion	Guenoche	[11]
6	CSPA	Strehl, Ghosh	[42]
7	MCLA	Strehl, Ghosh	[42]

These algoritms have been compared with two versions of the iterative extraction partitional clusters method above differing by the condition whether the option of zeroing all the diagonal entries of the similarity matrix has been utilized or not (Lsc1 and Lsc2). These build a cluster by applying Add-and-Remove(i) for every i avail-

able and choosing that of the clusters found with the maximum contribution. Three types of datasets have been used: (a) datasets from the Irvine Data Repository, (b) generated synthetic datasets, and (c) specially drawn artificial 2D shapes. The results are more or less similar to each other. Here we present only results of applying the algorithms to the Wisconsin Diagnostic Breast Cancer (WDBC) dataset from UCI Data Repository (569 entities, 30 features, two classes). The settng was as follows. The number of clusters is considered unknown. Therefore, first, the number of clusters k is generated randomly in the interval from 2 to 15 inclusive. Then k-means runs using a random initialization. The resulting partition is stored. This is repeated 20 times, after which each of the nine algorithms applies to find a corresponding consensus partition. This partition is compared with that 2-class partition of WDBC dataset that is supplied with it, one class is of malignant cases, the other, benign. The scoring function reported is the accuracy, a measure of the quality of prediction. Given a consensus partition, overlaps of each of its clusters with the benign and malignant classes are computed, and that the larger of the two is taken as the "prediction". Then the union of all the "predictions" is taken and its proportion in the entire dataset is the accuracy. Figure 3 presents the box-plot of all the accuracy values after a hundred of repetitions of the above.

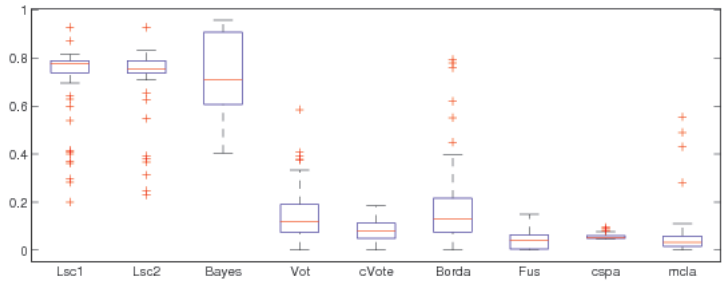

Fig. 3 Comparison of the accuracy of consensus clustering algorithms at WDBC dataset.

This shows that Semi-average Criterion Add-and-Remove algorithm, in spite of its "greedyness", can be successfully applied in some specific situations, such as that reported, when each of the ensemble of clusterings to be aggregated is found as a result of a k-means run.

Conclusion

The paper describes a method for finding individual similarity clusters, that can be presented in several perspectives - summary similarity criterion, semi-average criterion, spectral clustering criterion and approximation criterion. The clustering criterion involves, in different forms, the concept of similarity threshold, or similarity

shift - a value subtracted from all the similarity matrix entries. The summary similarity criterion involves a constant similarity threshold, whereas the semi-average criterion can be thought of as a summary criterion with a changeable similarity threshold that keeps to the half mean of the within cluster similarities. The threshold can be used for bridging different aspects of the phenomenon under study together. This is demonstrated in section 5.2, in which the final choice of clustering involves the protein function and gene arrangement in the genomic circle, in addition to the original similarity derived from protein sequences.

The criterion leads to nice properties of the clusters. First of all, the similarity data always can be made symmetric by just redistributing similarity symmetrically. Second, clusters are quite tight over average similarities of individual entities with them: these within clusters are always greater than outside clusters; the half average within cluster similarity being a threshold physically separating the two in the case of the semi-average criterion. Third, the recommended locally optimizing the criterion method, Add-and-Remove, is much intuitive: each step is based on choosing an entity according to its average similarity to the fragment of cluster that has been built already in the process. Fourth, unlike methods for finding global optima, this one leads to recovery of the local cluster structure of the data, probably a single most important innovation proposed in this paper.

The Semi-average criterion, in fact, does not hang out as a sole object. It is closely related to the least-squares criterion utilized for clustering data in the conventional setting of entity-to feature data tables [26]. It also closely follows the concept of consensus clustering. An original method for fuzzy clustering by exploiting the same principles of one-by-one extraction of clusters, has shown rather good results in experiments [31].

In the end, I would like to express my gratitude for the partial support of this work to the International Laboratory of Decision Choice and Analysis at NRU HSE (headed by F. Aleskerov) and the Laboratory of Algorithms and Technologies for Network Analysis NRU HSE Nizhny Novgorod by means of RF goverment grant ag. 11.G34.31.0057 (headed by V. Kalyagin).

References

1. H. Ayad, M. Kamel (2010) On voting-based consensus of cluster ensembles, *Pattern Recognition*, 1943-1953.
2. Bader, G.D. and Hogue, C.W.V. (2003) An automated method for finding molecular complexes in large protein interaction networks, *BMC Bioinformatics*, 4:2.
3. A. Ben-Dor, R. Shamir, Z. Yakhini, Clustering gene expression patterns, *Journal of Computational Biology*, **6**, 281-297, 1999.
4. S. Brohée and J. van Helden (2006) Evaluation of clustering algorithms for protein-protein interaction networks, *BMC Bioinformatics*, 7:488 (http://www/biomedcentral.com/1471-2105/7/488.
5. A.J. Davison, D.J. Dargan and N.D. Stow (2002) Fundamental and accessory systems in herpesvirus: Review, *Antiviral Research*, 56, 1-11.
6. E. Dimitriadou, A. Weingessel and K. Hornik (2002) A Combination Scheme for Fuzzy Clustering, *Journal of Pattern Recognition and Artificial Intelligence*, 332-338.

7. K. Florek, J. Lukaszewicz, H. Perkal, H. Steinhaus, and S. Zubrzycki (1951) Sur la liason et la division des points d'un ensemble fini, *Colloquium Mathematicum, 2*, 282-285.
8. R. Frumkina, B. Mirkin (1986) Sematics of domain-specific nouns: a psycho-linguistic approach, Notices of Russian Academy of Science: Language and Literature, 45(1), 12-22 (in Russian).
9. G. Gallo, M.D. Grigoriadis, and R.E. Tarjan (1989) A fast parametric maximum flow algorithm and applications, *SIAM Journal on Computing, 18*, 30-55.
10. N. Garg, V. V. Vazirani, M. Yannakakis (1996) Approximate Max-Flow Min-(Multi)Cut theorems and their applications, *SIAM Journal on Computing, 25*, n.2,235-251.
11. A. Guenoche (2011) Consensus of partitions : a constructive approach, *Adv. Data Analysis and Classification*, 5, pp. 215-229.
12. J.C. Gower and G.J.S. Ross (1969) Minimum spanning trees and single linkage cluster analysis *Applied Statistics, 18*, 54-64.
13. J.A. Hartigan (1967) Representation of similarity matrices by trees, *J. Amer. Stat. Assoc., 62*, 1140-1158.
14. K.J. Holzinger and H.H. Harman (1941) *Factor Analysis*, University of Chicago Press, Chicago.
15. U.M. Fayyad, G. Piatetsky-Shapiro and P. Smyth (1996) From data mining to knowledge discovery: An overview. In U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy (Eds.), *Advances in Knowledge Discovery and Data Mining*, Menlo Park, Ca: AAAI Press/The MIT Press, 1-37.
16. A.K. Jain and R.C. Dubes (1988) *Algorithms for Clustering Data*, Englewood Cliffs, NJ: Prentice Hall.
17. R. A. Jarvis and E.A. Patrick (1973) Clustering using a similarity measure based on shared nearest neighbors, *IEEE Trans. Comput., 22*, 1025–1034.
18. Hideya Kawaji, Yoichi Takenaka, Hideo Matsuda (2004) Graph-based clustering for finding distant relationships in a large set of protein sequences, *Bioinformatics*, 20(2), 243-252.
19. V. Kupershtoh, B. Mirkin (1968) A problem for automatic classification, In: K. Bagrinowski (Ed.) *Mathematical Methods for Economics*, Siberian Branch of Nauka Publisher, Novosibirsk, 39-49 (in Russian).
20. V. Kupershtoh, B. Mirkin, and V. Trofimov (1976) Sum of within partition similarities as a clustering criterion, Automation and Remote Control, 37, n.2, 548-553.
21. B. Mirkin (1976) *Analysis of Categorical Features*, Finansy i Statistika Publishers, Moscow, 166 p. (In Russian)
22. B. Mirkin (1987) Additive clustering and qualitative factor analysis methods for similarity matrices, *Journal of Classification, 4*, 7-31; Erratum (1989), *6*, 271-272.
23. B. Mirkin (1990) A sequential fitting procedure for linear data analysis models, *Journal of Classification, 7*, 167-195.
24. B. Mirkin (1996) *Mathematical Classification and Clustering*, Dordrecht: Kluwer Academic Press.
25. B. Mirkin (2011) *Core Concepts in Data Analysis: Summarization, Correlation, Visualization*, Springer, London.
26. B. Mirkin (2012) *Clustering: A Data Recovery Approach*, 2nd Edition, Chapman and Hall, Boca Raton.
27. B.G. Mirkin, R. Camargo, T. Fenner, G. Loizou and P. Kellam (2010) Similarity clustering of proteins using substantive knowledge and reconstruction of evolutionary gene histories in herpesvirus, *Theoretical Chemistry Accounts: Theory, Computation, and Modeling*, 125(3-6), 569-581.
28. B. Mirkin, T. Fenner, M. Galperin and E. Koonin (2003) Algorithms for computing parsimonious evolutionary scenarios for genome evolution, the last universal common ancestor and dominance of horizontal gene transfer in the evolution of prokaryotes, *BMC Evolutionary Biology*, 3:2 (www.biomedcentral.com/1471-2148/3/2/).
29. B. Mirkin and E. Koonin (2003) A top-down method for building genome classification trees with linear binary hierarchies, In M. Janowitz, J.-F. Lapointe, F. McMorris, B. Mirkin, and F. Roberts (Eds.) *Bioconsensus*, DIMACS Series, V. 61, Providence: AMS, 97-112.

30. B. Mirkin and I. Muchnik (1981) Geometric interpretation of clustering criteria, in B. Mirkin (Ed.) Methods for Analysis of Multidimensional Economics Data, Nauka Publishers (Siberian Branch), Novosibirsk, 3-11 (in Russian).
31. B. Mirkin and S. Nascimento (2012) Additive spectral method for fuzzy cluster analysis of similarity data including community structure and affinity matrices, *Information Sciences*, 183, 16-34.
32. M.E.J. Newman (2006) Modularity and community structure in networks, *PNAS*, 103(23), 8577-8582.
33. M. Newman and M. Girvan (2004) Finding and evaluating community structure in networks, *Physical Review E*, 69, 026113.
34. S Rosenberg, MP Kim (1975) The method of sorting as a data-gathering procedure in multivariate research, *Multivariate Behavioral Research*, 10, 489-502.
35. G.A. Satarov (1981) A non-intrusive knowledge evaluation method (Personal communication).
36. X. Sevillano Dominguez, J. C. Socoro Carrie and F. Alias Pujol (2009) Fuzzy clusterers combination by positional voting for robust document clustering, *Procesamiento del lenguaje natural*, 43, pp. 245-253.
37. R.N. Shepard and P. Arabie (1979) Additive clustering: representation of similarities as combinations of overlapping properties, *Psychological Review, 86*, 87-123.
38. A. Shestakov, B. Mirkin (2013) Least squares consensus clustering applied to k-means results (in progress).
39. J. Shi and J. Malik (2000) Normalized cuts and image segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence, 22*, n. 8, 888-905.
40. H. Small (1973) Co-citation in the scientific literature: A new measure of the relationship between two documents, *Journal of the American Society for Information Science*, 24, 265-269.
41. M. Smid, L.C.J. Dorssers and G. Jenster (2003) Venn Mapping: clustering of heterologous microarray data based on the number of co-occurring differentially expressed genes, *Bioinformatics*, 19, no. 16, 2065-2071.
42. A. Strehl, J. Ghosh (2002) Cluster ensembles - a knowledge reuse framework for combining multiple partitions, *Journal on Machine Learning Research*, 583-617.
43. S. Swift, A. Tucker, V. Vinciotti, N. Martin, C. Orengo, X. Liu and P. Kellam (2004) Consensus clustering and functional interpretation of gene expression data, *Genome Biology*, 5:R94.
44. H. Wang, H. Shan, A. Banerjee (2009) Bayesian cluster ensembles. In: *Proceedings of the Ninth SIAM International Conference on Data Mining*, 211-222.