SIAM J. MATRIX ANAL. APPL.
Vol. 33, No. 4, pp. 1214–1236

STOCHASTIC DATA CLUSTERING*

CARL D. MEYER† AND CHARLES D. WESSELL‡

Abstract. In 1961 Herbert Simon and Albert Ando [*Econometrika*, 29 (1961), pp. 111–138] published the theory behind the long-term behavior of a dynamical system that can be described by a nearly uncoupled matrix. Over the past fifty years this theory has been used in a variety of contexts, including queueing theory, brain organization, and ecology. In all of these applications, the structure of the system is known and the point of interest is the various stages the system passes through on its way to some long-term equilibrium. This paper looks at this problem from the other direction. That is, we develop a technique for using the evolution of the system to tell us about its initial structure, and then use this technique to develop an algorithm that takes the varied solutions from multiple data clustering algorithms to arrive at a single data clustering solution.

Key words. cluster analysis, Markov chains, Simon–Ando theory

AMS subject classifications. 60J20, 62H30, 91C20

DOI. 10.1137/100804395

1. Introduction.

There is no shortage of data clustering algorithms. Indeed, many individual algorithms provide one or more parameters that can be set to a variety of values, effectively turning that single algorithm into many. Even if we restrict ourselves to a single algorithm with fixed starting parameters, we can still get varied results since methods like k-means and nonnegative matrix factorization (NMF) use random initializations that can lead to different final results.

In order to avoid the confusion of multiple algorithms and differing solutions, a researcher might decide on one clustering method with one set of parameters and then accept the result as *the* clustering for that data set. While such an approach is simple, it can lead to the acceptance of a poor clustering result. An alternative approach used by some clustering researchers is to gather many clustering solutions and to use all of them to arrive at a single clustering superior to any individual solution.

The purpose of this paper is to motivate and develop a new method for merging multiple clustering results using theory on the behavior of nearly uncoupled matrices developed by Nobel laureate Herbert Simon and his student Albert Ando.

When a collection of clustering methods is used, the collection is called an ensemble, and so this process is sometimes referred to as *ensemble clustering*. Others use the term *cluster aggregation* [20]. Since the goal is for these varied methods to come to some agreement, it is also sometimes known as *consensus clustering*, which will be the term used throughout this paper.

The starting point for any clustering method is an m-dimensional data set of n elements. The data set can thus be stored as an $m \times n$ matrix A where each column represents an element of the data set and each row contains the value of a particular attribute for each of the elements. If the assignment of clusters from a single run of

*Received by the editors August 4, 2010; accepted for publication (in revised form) by D. B. Szyld August 14, 2012; published electronically November 20, 2012.
http://www.siam.org/journals/simax/33-4/80439.html
†Department of Mathematics and Institute of Advanced Analytics, North Carolina State University, Raleigh, NC 27695 (meyer@ncsu.edu).
‡Department of Mathematics, Gettysburg College, Gettysburg, PA 17325 (cwessell@gettysburg.edu).

a clustering algorithm is denoted by $\mathcal{C}_k$, then the input to any consensus method will be $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_r\}$.

One approach for solving this problem is attempting to find a clustering $\mathcal{C}^*$ that is as close as possible to all the $\mathcal{C}_k$'s. This is an optimization problem known as *median partition*, and is known to be NP-complete. A number of heuristics for the median partition problem exist. Discussion of these heuristics with comparisons and results on real-world data sets can be found in [14, 15, 21].

Other researchers have brought statistical techniques to bear on this problem, using bootstrapping or other more general resampling techniques to cluster subsets of the original data set, and then examining the results using some measure of consistency to settle on the final clustering [18, 35].

Additional approaches include a consensus framework built on a variational Bayes mixture of Gaussians model [23] and using algorithms originally intended for rank aggregation problems [2].

Other approaches to this problem begin by storing the information from each $\mathcal{C}_k$ in an $n \times n$ adjacency matrix $A^{(k)}$ such that if data set elements i and j are in the same cluster according to $\mathcal{C}_k$, then $a_{ij}^{(k)} = 1$, and $a_{ij}^{(k)} = 0$ if they are not (in this paper we will define $a_{ii}^{(k)} = 1$ for $i = 1, 2, \ldots, n$). The collection of these r adjacency matrices can be used to define a hypergraph which can then be partitioned (i.e., clustered) using known hypergraph partitioning algorithms [47].

Alternatively, this collection of adjacency matrices can be summed to form the consensus matrix S. Each entry s_{ij} of S denotes how many times elements i and j clustered together. For those who would prefer that all entries of S lie in the interval $[0, 1]$, S can be defined as the sum of the adjacency matrices times $\frac{1}{r}$, resulting in a symmetric similarity matrix whose similarity measure is the fraction of the time that two elements were clustered together. In this paper, S will always be used to refer to the sum of the adjacency matrices.

Once S is constructed, its columns can be clustered and thus the original data is clustered [38]. This method using single-link hierarchical clustering on S, after elements below a threshold have been zeroed out, has proven effective [17].

A new methodology developed to cluster different conformations of a single drug molecule comes the closest to the approach developed in this paper. For this application, a Markov chain transition matrix can be created where the ijth entry gives the probability the molecule changes from conformation i to conformation j. The goal is to then find sets of conformations such that if the molecule is currently in a particular set, it will remain in that set for a relatively long time. Approaches to this clustering problem have included examination of the first few eigenvectors of the transition matrix ([11] and then improved in [12]), clustering the data based on the second singular vector [19, 49], and spectral analysis of a family of Hermitian matrices that is a function of the transition matrix [25].

2. A new approach.

The consensus clustering method introduced in this paper is based on the 1950's variable aggregation work of the Nobel prize winning economist Herbert Simon and his graduate student Albert Ando [41]. Their theory will be reviewed in section 2.1, and further theoretical work will be developed in sections 2.2–2.4 before the algorithm is introduced in section 3.

2.1. Theoretical background.

Simon–Ando theory was originally designed as a way of understanding the short and long term behavior of an economy with a

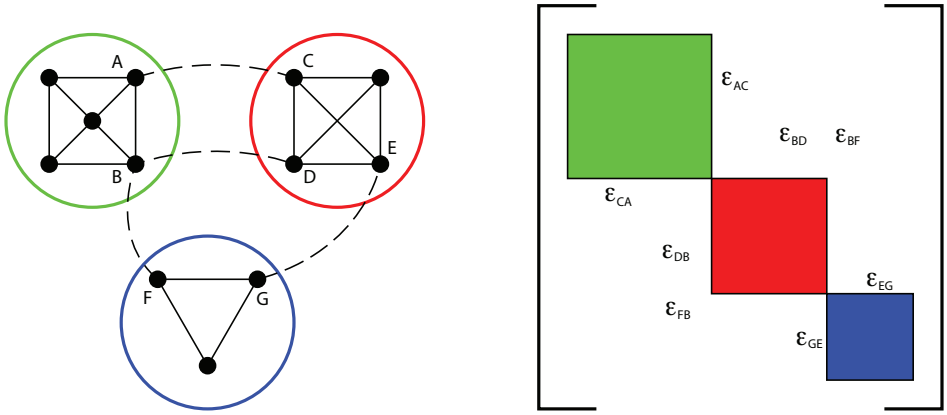

FIG. 2.1. *This figure illustrates a simple Simon–Ando system and how it would be represented in matrix form. Let the circles on the left represent three small countries. The graphs within each circle represent companies in those countries and the solid lines between them represent a large amount of capital exchange between the companies. The dashed lines represent a small amount of cross-border exchange. A matrix whose entries represented the amount of economic activity between any two companies in this system would look like the one on the right with the shaded areas being dense with relatively large values and the epsilons being relatively small.*

certain structure. Figure 2.1 illustrates a simple system where Simon–Ando theory would apply.

Such a closed economic system, without any outside influences, is known to eventually reach a state of equilibrium, that is, after some initial fluctuations, the flow of goods and capital between any two industries will remain more or less constant. Rather than waiting for this economic equilibrium to occur, Simon and Ando tried to predict the long-term equilibrium by making only short-term observations. They proved that what happens in the short run completely determines the long-term equilibrium.

Over the years scholars in a variety of disciplines have realized the usefulness of a framework that represents a number of tightly knit groups that have some loose association with each other, and Simon–Ando theory has been applied in areas as diverse as ecology [28], computer queueing systems [9], brain organization [45], and urban design [40]. Simon himself went on to apply the theory to the evolution of multicellular organisms [42].

The $n \times n$ matrix S is called *uncoupled* if it has the form

$$S = \begin{pmatrix} S_{11} & 0 & \cdots & 0 \\ 0 & S_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & S_{kk} \end{pmatrix},$$

where the diagonal blocks S_{ii} are square. If S is not uncoupled for any value of $k \geq 2$ and if entries in the off-diagonal blocks are small relative to those in the diagonal blocks, then we say that S is *nearly uncoupled*. The matrix in Figure 2.1 is an example of a nearly uncoupled matrix. A more formal measure of uncoupledness will be introduced in Definition 2.12.

If the consensus matrix S described in the introduction is nearly uncoupled, we will show that Simon–Ando theory can be used to cluster the data it describes. Notice that S is symmetric and this combined with it not being uncoupled means S is also

irreducible. For reasons that will soon become apparent, the new clustering method will require that S be converted to doubly stochastic form. This new matrix will be called P and the data clustering method will depend on P having a unique stationary distribution vector (which is guaranteed by irreducibility) with a known structure (which is guaranteed by double stochasticity).

Before we can use P to cluster data we need to introduce the concept of stochastic complementation.

If P is stochastic, then each diagonal block P_{ii} has a stochastic complement defined by

$$C_{ii} = P_{ii} + P_{i\star}\left(I - P_i\right)^{-1} P_{\star i}, \tag{2.1}$$

where P_i is the matrix obtained by deleting the ith row and ith column of blocks from P, $P_{i\star}$ is the ith row of blocks of P with P_{ii} removed, and $P_{\star i}$ is the ith column of blocks of P with P_{ii} removed. Since every principal submatrix of $I - P$ of order $n-1$ or smaller is a nonsingular M-matrix, the matrix $(I - P_i)^{-1}$ found in (2.1) is defined and $(I - P_i)^{-1} \geq 0$. Furthermore, if P is stochastic and irreducible, then each C_{ii} is itself a stochastic, irreducible matrix with stationary distribution vector c_i^T [4, 32].

Let x_0^T be a probability row vector and consider the evolution equation

$$x_t^T = x_{t-1}^T P \tag{2.2}$$

or its equivalent formulation

$$x_t^T = x_0^T P^t. \tag{2.3}$$

Simon–Ando theory asserts that x_t^T will pass through distinct stages as t grows to infinity. Meyer [32] describes how these stages can be interpreted in terms of the individual stationary distribution vectors c_i^T. The following lemma and theorem will aid in extending that explanation to the case where P is doubly stochastic. The proof of the lemma is a direct application of principles of permutation matrices and is omitted.

LEMMA 2.1. *Let P be an $n \times n$ irreducible doubly stochastic matrix in which the diagonal blocks are square. Let Q be the permutation matrix associated with an interchange of the first and ith block rows (or block columns) and let $\tilde{P}$ be defined as*

$$\tilde{P} = QPQ.$$

If $\tilde{P}$ is partitioned into a 2×2 block matrix

$$\tilde{P} = \begin{pmatrix} \tilde{P}_{11} & \tilde{P}_{12} \\ \tilde{P}_{21} & \tilde{P}_{22} \end{pmatrix} \text{ where } \tilde{P}_{11} = P_{ii}, \tag{2.4}$$

then the stochastic complement of P_{ii} is

$$C_{ii} = \tilde{C}_{11} = \tilde{P}_{11} + \tilde{P}_{12}\left(I - \tilde{P}_{22}\right)^{-1} \tilde{P}_{21}. \tag{2.5}$$

THEOREM 2.2. *If*

$$P = \begin{pmatrix} P_{11} & P_{12} & \dots & P_{1k} \\ P_{21} & P_{22} & \dots & P_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ P_{k1} & P_{k2} & \dots & P_{kk} \end{pmatrix}$$

is an irreducible doubly stochastic matrix, then each stochastic complement is also an irreducible, doubly stochastic matrix.

Proof. As stated earlier, if the stochastic matrix P is irreducible, then so are each of its stochastic complements. Therefore, we need only prove that each S_{ii} is doubly stochastic. For a given i, suppose diagonal block P_{ii} has been repositioned such that $\tilde{P}_{11} = P_{ii}$ as in (2.4) of Lemma 2.1.

Let e represent a column vector of all ones. Both the row and column sums of P are one, so allowing the size of e to be whatever is appropriate for the context, the following four equations are true:

$$\tilde{P}_{11}e + \tilde{P}_{12}e = e, \tag{2.6}$$

$$\tilde{P}_{21}e + \tilde{P}_{22}e = e, \tag{2.7}$$

$$e^T\tilde{P}_{11} + e^T\tilde{P}_{21} = e^T, \tag{2.8}$$

$$e^T\tilde{P}_{12} + e^T\tilde{P}_{22} = e^T. \tag{2.9}$$

Equations (2.7) and (2.9) can be rewritten to yield

$$e = \left(I - \tilde{P}_{22}\right)^{-1}\tilde{P}_{21}e \quad \text{and} \quad e^T = e^T\tilde{P}_{12}\left(I - \tilde{P}_{22}\right)^{-1}.$$

As noted earlier, $(I - \tilde{P}_{22})^{-1} \geq 0$, and hence

$$\tilde{C}_{11} = \tilde{P}_{11} + \tilde{P}_{12}\left(I - \tilde{P}_{22}\right)^{-1}\tilde{P}_{21} \geq 0.$$

Multiplying $\tilde{C}_{11}$ on the right by e and on the left by e^T yields

$$\tilde{C}_{11}e = \tilde{P}_{11}e + \tilde{P}_{12}\left(I - \tilde{P}_{22}\right)^{-1}\tilde{P}_{21}e = \tilde{P}_{11}e + \tilde{P}_{12}e = e$$

and

$$e^T\tilde{C}_{11} = e^T\tilde{P}_{11} + e^T\tilde{P}_{12}\left(I - \tilde{P}_{22}\right)^{-1}\tilde{P}_{21} = e^T\tilde{P}_{11} + e^T\tilde{P}_{21} = e^T.$$

Therefore, since $C_{ii} = \tilde{C}_{11}$, each stochastic complement is doubly stochastic. ∎

Markov chain theory tells us that as $t \to \infty$, x_t^T will approach the uniform distribution vector $(1/n \;\; 1/n \;\; \ldots \;\; 1/n)$. If the size of each P_{ii} is $n_i \times n_i$, we also know that $c_i^T = (1/n_i \;\; 1/n_i \;\; \ldots \;\; 1/n_i)$.

As t increases from zero, x_t^T initially goes through changes driven by the comparatively large values in each P_{ii}. Once these changes have run their course, the system settles into a period of short-term stabilization characterized by

$$\begin{aligned} x_t^T &\approx (\alpha_1 c_1 \;\; \alpha_2 c_2 \;\; \ldots \;\; \alpha_k c_k) \\ &= \left(\frac{\alpha_1}{n_1}\;\frac{\alpha_1}{n_1}\;\cdots\;\frac{\alpha_1}{n_1}\middle|\frac{\alpha_2}{n_2}\;\frac{\alpha_2}{n_2}\;\cdots\;\frac{\alpha_2}{n_2}\middle|\;\cdots\;\middle|\frac{\alpha_k}{n_k}\;\frac{\alpha_k}{n_k}\;\cdots\;\frac{\alpha_k}{n_k}\right), \end{aligned}$$

where each α_i is a constant dependent on x_0^T.

After this equilibrium period, the elements of x_t^T begin to change again through a period called middle-run evolution, this time being affected by the small values in the off-diagonal blocks, but the change is predictable and can be described by

$$x_t^T \approx (\beta_1 c_1 \;\; \beta_2 c_2 \;\; \dots \;\; \beta_k c_k)$$
$$= \left(\frac{\beta_1}{n_1} \frac{\beta_1}{n_1} \dots \frac{\beta_1}{n_1} \middle| \frac{\beta_2}{n_2} \frac{\beta_2}{n_2} \dots \frac{\beta_2}{n_2} \middle| \;\; \dots \;\; \middle| \frac{\beta_k}{n_k} \frac{\beta_k}{n_k} \dots \frac{\beta_k}{n_k} \right),$$

where each β_i is dependent on t.

Simon and Ando were not interested in clustering data. For them, the importance of stages like short-term stabilization and middle-run evolution lie in the fact that even for small values of t, the structure of x_t^T reflected the stationary probability vectors of the smaller C_{ii} matrices. From there, examination of the x_t^T vector during the relatively stable periods would allow for determination of these smaller stationary probability vectors and facilitate the calculation of the stationary probability vector for P.

For cluster analysis, however, the focus is turned around. Since we will be using doubly stochastic P matrices, we already know that the stationary probability vector is the uniform probability vector. We also know that each diagonal block P_{ii} is associated with a uniform probability vector related to its stochastic complement. Identification of the clusters then comes down to examining the entries of x_t^T. The key is to look for elements of x_t^T that are approximately equal. The only difference between short-run and middle-run is whether the elements of x_t^T stay at approximately the same value for a number of iterations or move together towards the uniform probability distribution.

All of the development in this section assumed a doubly stochastic matrix. We will now consider how to convert a matrix into doubly stochastic form, and show that the process does not destroy any of the desirable characteristics of our matrix.

2.2. Sinkhorn–Knopp. The process of converting a matrix into doubly stochastic form has drawn considerable attention, and in 1964 Sinkhorn showed that any positive square matrix can be scaled to a unique doubly stochastic matrix [43]. This result can be extended to nonnegative matrices as long as the zero entries are in just the right places. An understanding of this zero structure will require some definitions.

DEFINITION 2.3 (see Sinkhorn and Knopp [44]). *A nonnegative $n \times n$ matrix S is said to have* total support *if $S \neq 0$ and if every positive element of S lies on a positive diagonal, where a* diagonal *is defined as a sequence of elements $s_{1\sigma(1)}, s_{2\sigma(2)}, \dots, s_{n\sigma(n)}$, where σ is a permutation of $\{1, 2, \dots, n\}$.*[1]

DEFINITION 2.4 (see Minc [34, p. 82]). *An $n \times n$ matrix S is* partly indecomposable *if there exist permutation matrices P and Q such that*

$$PSQ = \begin{bmatrix} X & Z \\ 0 & Y \end{bmatrix},$$

where X and Y square. If no such P and Q exist, then S is fully indecomposable.

DEFINITION 2.5 (see Minc [34, p. 82]). *Two matrices A and B are* permutation equivalent*, or* p-equivalent*, if there exist permutation matrices Q and $\hat{Q}$ such that $A = QB\hat{Q}$.*

[1] Notice that by this definition of *diagonal*, the main diagonal of a matrix is the one associated with the permutation $\sigma = (1\ 2\ 3 \dots n)$.

This new terminology will help in understanding the following, nearly identical theorems that were independently proven and then published within a year of each other, the first in 1966 and the second in 1967.

THEOREM 2.6 (see Brualdi, Parter, and Schneider [6]). *If the $n \times n$ matrix A is nonnegative and fully indecomposable, then there exist diagonal matrices D_1 and D_2 with positive diagonal entries such that D_1AD_2 is doubly stochastic. Moreover, D_1 and D_2 are uniquely determined up to scalar multiples.*

THEOREM 2.7 (see Sinkhorn and Knopp [44]). *If the $n \times n$ matrix A is nonnegative, then a necessary and sufficient condition that there exists a doubly stochastic matrix of the form D_1AD_2 where D_1 and D_2 are diagonal matrices with positive diagonal entries is that A has total support. If D_1AD_2 exists, then it is unique. Also D_1 and D_2 are unique up to a scalar multiple if and only if A is fully indecomposable.*

The uniqueness up to a scalar multiple of D_1 and D_2 mentioned in both theorems means that if E_1 and E_2 are also diagonal matrices such that E_1AE_2 is doubly stochastic, then $E_1 = \alpha D_1$ and $E_2 = \beta D_2$, where $\alpha\beta = 1$.

The way that the consensus similarity matrix S is constructed guarantees its nonnegativity, so the only thing standing in the way of knowing that the scaling matrices D_1 and D_2 exist is showing that S either has total support or is fully indecomposable. Reviewing the definitions of these terms, neither of these tasks seems inviting. Fortunately, there is a theorem that will simplify the matter.

THEOREM 2.8 (see Minc [34, p. 86]). *A nonnegative matrix is fully indecomposable if and only if it is p-equivalent to an irreducible matrix with a positive main diagonal.*

S is trivially p-equivalent to itself since $S = ISI$ and S is an irreducible matrix with a positive main diagonal. Now that we know S is fully indecomposable, its symmetry is going to guarantee another useful result. The proof of the following lemma is included since there was a typographical error in the original paper.

LEMMA 2.9 (see Csima and Datta [10]). *Let S be a fully indecomposable symmetric matrix. Then there exists a diagonal matrix D such that DSD is doubly stochastic.*

Proof. Let D_1 and D_2 be nonnegative diagonal matrices such that D_1SD_2 is doubly stochastic. Then $(D_1SD_2)^T = D_2SD_1$ is also doubly stochastic. By the uniqueness up to a scalar multiple from Theorems 2.6 and 2.7, we know $D_2 = \alpha D_1$ and $D_1 = \beta D_2$. Using the first of these facts

$$\begin{aligned} D_1SD_2 &= D_1S\alpha D_1 \\ &= \sqrt{\alpha}D_1S\sqrt{\alpha}D_1 \\ &= DSD \end{aligned}$$

shows us that $D = \sqrt{\alpha}\, D_1$. ∎

2.3. The structure of *DSD*. We will use P as the symbol for the doubly stochastic matrix derived from S, that is, $P = DSD$. For simplicity of notation, the ith diagonal entry of D will be denoted d_i. We will show that P has the same desirable properties that S has.

LEMMA 2.10. *If S is an $n \times n$ fully indecomposable irreducible matrix and $P = DSD$ is doubly stochastic, then P is irreducible.*

Proof. Since S is irreducible, there is no permutation matrix Q such that

$$QSQ^T = \begin{bmatrix} X & Z \\ 0 & Y \end{bmatrix},$$

where both X and Y are square.

Thus the only way that $P = DSD$ could be reducible is if the zero structure of S is changed by the multiplication. But notice that since $p_{ij} = d_i d_j s_{ij}$ and both d_i and d_j are positive, $p_{ij} = 0$ only when $s_{ij} = 0$. So the zero structure does not change, and P is irreducible. □

Since the number of times elements i and j cluster with one another is necessarily equal to the number of times elements j and i cluster with one another, the symmetry of the consensus similarity matrix S reflects a real-world property of the consensus clustering problem and so it is important that symmetry is not lost when S is converted into P.

LEMMA 2.11. *If S is an $n \times n$ fully indecomposable symmetric matrix and $P = DSD$ is doubly stochastic, then P is symmetric.*

Proof. The proof is that

$$P^T = (DSD)^T = DS^T D = DSD = P. \qquad \square \tag{2.10}$$

We wish to prove that if S is nearly uncoupled, then so is P. To do so we first need a formal definition of near uncoupledness. Then we will show how this uncoupling measure for P is related to the uncoupling measure of S.

DEFINITION 2.12. *Let n_1 and n_2 be fixed positive integers such that $n_1 + n_2 = n$, and let S be an $n \times n$ symmetric, irreducible matrix whose respective rows and columns have been rearranged to the form*

$$S = \begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix},$$

where S_{11} is $n_1 \times n_1$ and S_{22} is $n_2 \times n_2$ so that the ratio

$$\sigma(S, n_1) = \frac{e^T S_{12} e + e^T S_{21} e}{e^T S e} = \frac{2 e^T S_{12} e}{e^T S e}$$

is minimized over all symmetric permutations of S. The quantity $\sigma(S, n_1)$ is called the uncoupling measure *of S with respect to parameter n_1. In other words $\sigma(S, n_1)$ is the ratio of the sum of the elements in the off-diagonal blocks to the sum of all the matrix entries.*

Before moving on, two points should be made clear. First, there is no arbitrary uncoupling measure value below which a matrix is deemed to be nearly uncoupled. Rather, $\sigma(S, n_1)$ is a relative value whose meaning is dependent on the uncoupling measures of S using other choices of n_1 or on comparisons with other similarity matrices a researcher has experience with. Second, exact calculation of the uncoupling measure for all but very small problems is not feasible, but its theoretical value is important since it allows us to compare matrices S and P as the the following theorem shows.

THEOREM 2.13. *If S is the $n \times n$ consensus matrix created from r clustering results, then for the doubly stochastic matrix $P = DSD$, $\sigma(P, n_1) \leq \frac{\Sigma}{nr} \sigma(S, n_1)$, where $\Sigma = e^T S e$.*

Proof. By the way we constructed S, $s_{ii} = r$ for $i = 1, 2, \ldots, n$. Since $p_{ii} = d_i d_i s_{ii}$ and $p_{ii} \leq 1$, it follows that $d_i^2 r$ implies $d_i \leq \frac{1}{\sqrt{r}}$.

If we impose the same block structure on D that exists for S, that is

$$D = \begin{bmatrix} D_1 & 0 \\ 0 & D_2 \end{bmatrix},$$

and recall that P is doubly stochastic, then

$$\sigma(P, n_1) = \frac{2e^T D_1 S_{12} D_2 e}{n}.$$

Since each element of D_1 and D_2 is less than $\frac{1}{\sqrt{r}}$,

$$\sigma(P, n_1) \leq \frac{\left(\frac{1}{\sqrt{r}}\right)^2 (2e^T S_{12} e)}{n} = \frac{\Sigma}{nr}\sigma(S, n_1),$$

and the bound is established. □

2.4. The spectrum of P. Consider the following facts about the eigenvalues of P:

1. Since P is stochastic, all of its eigenvalues lie on or inside the unit circle of the complex plane.
2. Since P is real-symmetric, all of its eigenvalues are real. Combined with the last fact, this means all eigenvalues of P reside in the interval $[-1, 1]$.
3. The largest eigenvalue of P is one, and since P is irreducible, that eigenvalue is simple (i.e., it appears only once).
4. $\lambda_i(P) \neq -1$ for all i because P is a primitive matrix. P is primitive because it is irreducible and has at least one positive diagonal element [33, p. 678].

Unlike Markov chain researchers who desire a small second eigenvalue since it leads to faster convergence when calculating the chain's stationary distribution vector, we want a second eigenvalue near one. Slow convergence is a good thing for us since it allows time to examine the elements of x_t as it passes through short-term stabilization and middle-run evolution. Also, $\lambda_2(P) \approx 1$ *may* indicate that the matrix is nearly uncoupled [46].

We will now show that $\lambda_2(P) \approx 1$ along with other properties of P *guarantees* that P is nearly uncoupled. First, observe the following lemma whose proof is self-evident.

LEMMA 2.14. *Let $\{P_k\}$ be a sequence of matrices with limit P_0. Then we have the following:*

1. *If each matrix in $\{P_k\}$ is symmetric, P_0 is symmetric.*
2. *If each matrix in $\{P_k\}$ is stochastic, P_0 is stochastic.*

THEOREM 2.15. *For a fixed integer $n > 0$, consider the $n \times n$ irreducible, symmetric, doubly stochastic matrix P. Given $\epsilon > 0$, there exists a $\delta > 0$ such that if $\sigma(P, n_1) < \delta$, then $|\lambda_2(P) - 1| < \epsilon$. In other words, if P is sufficiently close to being uncoupled, then $\lambda_2(P) \approx 1$.*

Proof. Two proofs will be presented. The first relies on a continuity argument, while the second gives an explicit bound on $|\lambda_2(P) - 1|$.

Proof (1): Let $\epsilon > 0$. Consider a sequence of irreducible, symmetric, doubly stochastic matrices

$$P_k = \begin{bmatrix} P_{11}^{(k)} & P_{12}^{(k)} \\ P_{21}^{(k)} & P_{22}^{(k)} \end{bmatrix}$$

defined so that $\lim_{k\to\infty} \sigma(P_k, n_1) = 0$. The Bolzano–Weierstrass theorem [3, p. 155], guarantees that this bounded sequence has a convergent subsequence $P_{k_1}, P_{k_2}, \ldots$, which converges to a stochastic matrix T whose structure is

$$T = \begin{bmatrix} T_{11} & 0 \\ 0 & T_{22} \end{bmatrix}, \quad T_{11} \neq 0, T_{22} \neq 0,$$

where each T_{ii} is stochastic. By the continuity of eigenvalues, there exists a positive integer M such that for $k_i > M$,

$$|\lambda_2(P_{k_i}) - \lambda_2(T)| < \epsilon \quad \Rightarrow \quad |\lambda_2(P_{k_i}) - 1| < \epsilon,$$

and the theorem is proven.

Proof (2): Suppose that the rows and respective columns have been permuted so that

$$P = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix},$$

where P is nearly uncoupled, and define C to be the $n \times n$ block diagonal matrix with the stochastic complements of P_{11} and P_{22} on the diagonals, that is

$$C = \begin{bmatrix} C_{11} & 0 \\ 0 & C_{22} \end{bmatrix}.$$

If E is defined to make the equation $C = P + E$ true, then a consequence of the Courant–Fisher theorem can be used [33, pp. 550–552] to show that for any matrix norm[2]

$$\lambda_2(P) - ||E|| \leq 1 \leq \lambda_2(P) + ||E|| \rightarrow |1 - \lambda_2(P)| \leq ||E||. \qquad \square$$

THEOREM 2.16. *For a fixed integer $n > 0$, consider the $n \times n$ irreducible, symmetric, doubly stochastic matrix P. Given $\epsilon > 0$, there exists a $\delta > 0$ such that if $|\lambda_2(P) - 1| < \delta$, then $\sigma(P, n_1) < \epsilon$ for some positive integer $n_1 < n$. In other words, if $\lambda_2(P)$ is sufficiently close to 1, then P is nearly uncoupled.*

Proof. The argument is by contradiction and similar to one used in [24]. Suppose there is an $\epsilon > 0$ such that for any $\delta > 0$ there is an $n \times n$ irreducible, symmetric, doubly stochastic matrix P with $|\lambda_2(P) - 1| < \delta$ and $\sigma(P, n_1) > \epsilon$ for all for positive integers $n_1 < n$. For $\delta = \frac{1}{k}$ let P_k be such a matrix. There must be a subsequence $P_{i_1}, P_{i_2}, \ldots$ which converges, say to P_0. Then P_0 must have $\lambda_2(P_0) = 1$ and thus $\sigma(P_0, n_1) = 0$. Yet, $\sigma(P_0, n_1) = \lim_{k \to \infty} \sigma(P_k, n_1) \geq \epsilon$, a contradiction. $\square$

Although we previously defined an uncoupling measure for a general matrix in section 2.3, for doubly stochastic matrices this theorem allows us to use λ_2 as an uncoupling indicator with a value near one signifying almost complete uncoupling.

There may be additional eigenvalues of P that are close to one. This group of eigenvalues is called the *Perron cluster* [11, 12], and in the case where all eigenvalues are real, the Perron cluster can be defined as follows.

DEFINITION 2.17. *Let P be an $n \times n$ symmetric, stochastic matrix with eigenvalues, including multiplicities, of $1 = \lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \cdots \geq \lambda_n$. If the largest difference between consecutive eigenvalues occurs between λ_k and λ_{k+1}, the set $\{1, \ldots, \lambda_k\}$ is called the* Perron cluster *of P. If two or more pairs of eigenvalues each have differences equal to the largest gap, use the smallest value of k to choose λ_k. The larger the gap, the more well defined the cluster.*

[2] If the 2-norm is used, then the bound is $|1 - \lambda_2(P)| \leq 2\sqrt{n}\sigma(P, n_1)$. We thank Ilse Ipsen for this observation.

Some researchers use the number of eigenvalues in the Perron cluster as the number of clusters they search for [11, 19]. This inference is a natural extension of Theorems 2.15 and 2.16, that is, if P had k eigenvalues sufficiently close to 1, then P is nearly uncoupled with k dominant diagonal blocks emerging after an appropriate permutation QPQ^T. This is also the approach we will take with the stochastic consensus clustering algorithm. Unlike with the vast majority of clustering methods, the user will not have to tell the algorithm the number of clusters in the data set unless they explicitly want to override the algorithm's choice. Instead, the stochastic consensus clustering algorithm will set k equal to the size of the Perron cluster.

3. Putting the concept into practice. Now that the theoretical underpinnings are in place, it is time to formally describe the stochastic consensus clustering algorithm.

The algorithm takes as input the consensus similarity matrix S which the user has created from whatever combination of clustering methods and/or parameter settings they choose. S is then converted into the doubly stochastic matrix P using the Sinkhorn–Knopp algorithm. All eigenvalues are computed, and the Perron cluster of P is identified. Eigenvalues of symmetric matrices can be efficiently computed [37], but if finding all eigenvalues is too costly, the user, with knowledge of the underlying data set, can direct the program to find only the $\hat{k}$ largest eigenvalues ($\hat{k} > k$). The size, k, of the Perron cluster of these $\hat{k}$ eigenvalues is then used by the stochastic consensus clustering algorithm to separate the data into k clusters.

Starting with a randomly generated x_0^T, $x_t^T = x_{t-1}^T P$ is evaluated for $t = 1, 2, \ldots$. After each calculation, the entries of x_t^T are sorted, the $k-1$ largest gaps in the sorted list identified and used to divide the entries into k clusters. When the k clusters have been identical for n iterations, where n is a user-chosen parameter, the program stops and the clusters returned as output. Figure 3.1 summarizes the algorithm.

3.1. A small example. Consider the following small data matrix which includes the career totals in nine statistics for six famous baseball players (the row labels stand for Games, Runs, Hits, Doubles, Triples, Home Runs, Runs Batted In, Stolen Bases, and Bases on Balls).

Stochastic consensus clustering algorithm (SCCA)

1. Create the consensus similarity matrix S using a clustering ensemble of user's choice.
2. Use matrix balancing to convert S into a doubly stochastic symmetric matrix P.
3. Calculate the eigenvalues of P. The number of clusters, k, is the number of eigenvalues in the Perron cluster.
4. Create a random x_0^T.
5. Track the evolution $x_t^T = x_{t-1}^T P$. After each multiplication, sort the elements of x_t^T and then separate the elements into k clusters by dividing the sorted list at the $k-1$ largest gaps. Alternatively, the elements of x_t can be clustered using k-means or any other widely available clustering method. When this clustering has remained the same for a user-defined number of iterations, the final clusters have been determined.

FIG. 3.1. *The stochastic consensus clustering algorithm.*

$$A = \begin{array}{c} \\ \text{G} \\ \text{R} \\ \text{H} \\ \text{2B} \\ \text{3B} \\ \text{HR} \\ \text{RBI} \\ \text{SB} \\ \text{BB} \end{array} \begin{array}{c} \begin{array}{cccccc} \text{Rose} & \text{Cobb} & \text{Fisk} & \text{Ott} & \text{Ruth} & \text{Mays} \end{array} \\ \begin{pmatrix} 3562 & 3034 & 2499 & 2730 & 2503 & 2992 \\ 2165 & 2246 & 1276 & 1859 & 2174 & 2062 \\ 4256 & 4189 & 2356 & 2876 & 2873 & 3283 \\ 746 & 724 & 421 & 488 & 506 & 523 \\ 135 & 295 & 47 & 72 & 136 & 140 \\ 160 & 117 & 376 & 511 & 714 & 660 \\ 1314 & 1938 & 1330 & 1860 & 2213 & 1903 \\ 198 & 897 & 128 & 89 & 123 & 338 \\ 1566 & 1249 & 849 & 1708 & 2062 & 1464 \end{pmatrix} \end{array}.$$

Those familiar with baseball history would mentally cluster these players into singles hitters (Rose and Cobb), power hitters (Mays, Ott, and Ruth) and, a great catcher who doesn't have enough home runs and runs batted in to fit with the power hitters nor the long career and large number of hits to fit with the singles hitters (Fisk).

The consensus similarity matrix was built using the multiplicative update version of the nonnegative matrix factorization algorithm [27]. Since it isn't clear whether two or three clusters would be most appropriate, S was created by running the NMF algorithm 50 times with $k = 2$ and 50 times with $k = 3$. The resulting similarity matrix is

$$S = \begin{array}{c} \\ \text{Rose} \\ \text{Cobb} \\ \text{Fisk} \\ \text{Ott} \\ \text{Ruth} \\ \text{Mays} \end{array} \begin{array}{c} \begin{array}{cccccc} \text{Rose} & \text{Cobb} & \text{Fisk} & \text{Ott} & \text{Ruth} & \text{Mays} \end{array} \\ \begin{pmatrix} 100 & 67 & 73 & 2 & 0 & 2 \\ 67 & 100 & 50 & 1 & 2 & 7 \\ 73 & 50 & 100 & 15 & 9 & 24 \\ 2 & 1 & 15 & 100 & 92 & 82 \\ 0 & 2 & 9 & 92 & 100 & 77 \\ 2 & 7 & 24 & 82 & 77 & 100 \end{pmatrix} \end{array}.$$

With a small example like this, especially one where the players that will cluster together have been purposely placed in adjacent columns, it would be simple enough to cluster the players through a quick scan of S. However, following the SCCA algorithm to the letter we use the Sinkhorn–Knopp method to convert S to doubly stochastic form. The resulting matrix (rounded to four places) is

$$P = \begin{array}{c} \\ \text{Rose} \\ \text{Cobb} \\ \text{Fisk} \\ \text{Ott} \\ \text{Ruth} \\ \text{Mays} \end{array} \begin{array}{c} \begin{array}{cccccc} \text{Rose} & \text{Cobb} & \text{Fisk} & \text{Ott} & \text{Ruth} & \text{Mays} \end{array} \\ \begin{pmatrix} 0.4131 & 0.2935 & 0.2786 & 0.0075 & 0.0000 & 0.0075 \\ 0.2935 & 0.4644 & 0.2023 & 0.0040 & 0.0082 & 0.0277 \\ 0.2786 & 0.2023 & 0.3525 & 0.0517 & 0.0323 & 0.0826 \\ 0.0075 & 0.0040 & 0.0517 & 0.3374 & 0.3233 & 0.2761 \\ 0.0000 & 0.01082 & 0.0323 & 0.3233 & 0.3660 & 0.2701 \\ 0.0075 & 0.0277 & 0.0826 & 0.2761 & 0.2701 & 0.3361 \end{pmatrix} \end{array}.$$

The eigenvalues of P are 1.0000, 0.8670, 0.2078, 0.1095, 0.0598, and 0.0254 suggesting that there are two clusters in this data.

Table 3.1 shows the results from a sample run of the consensus clustering method. The initial probability vector x_0^T was chosen randomly, and the table shows the value of x_t^T and the corresponding clusters for the next seven steps of the algorithm. Since

TABLE 3.1
Following the stochastic consensus clustering algorithm for the small example.

t	x_t^T	Clusters
0	(0.2334 0.2595 0.0364 0.2617 0.1812 0.0279)	{Rose, Cobb, Ott, Ruth} {Fisk, Mays}
1	(0.1848 0.1997 0.1520 0.1592 0.1618 0.1425)	{Rose, Cobb} {Fisk, Ott, Ruth, Mays}
2	(0.1795 0.1836 0.1707 0.1554 0.1557 0.1550)	{Rose, Cobb, Fisk} {Ott, Ruth, Mays}
3	(0.1779 0.1787 0.1732 0.1565 0.1561 0.1576)	{Rose, Cobb, Fisk} {Ott, Ruth, Mays}
4	(0.1765 0.1765 0.1729 0.1578 0.1574 0.1589)	{Rose, Cobb, Fisk} {Ott, Ruth, Mays}
5	(0.1752 0.1751 0.1722 0.1590 0.1586 0.1600)	{Rose, Cobb, Fisk} {Ott, Ruth, Mays}
6	(0.1741 0.1739 0.1715 0.1600 0.1597 0.1609)	{Rose, Cobb, Fisk} {Ott, Ruth, Mays}
7	(0.1731 0.1729 0.1709 0.1609 0.1606 0.1616)	{Rose, Cobb, Fisk} {Ott, Ruth, Mays}

$k = 2$, the clusters are determined by ordering the entries of x_t^T, finding the largest gap in this list, and clustering the elements on either side of this gap. For example, at the $t = 6$ step shown in the table, the largest gap in the sorted list is between 0.1609 and 0.1715. This leads to the numerical clustering of $\{0.1597, 0.1600, 0.1609\}$ and $\{0.1715, 0.1739, 0.1741\}$ which translates to the clustering {Rose, Cobb, Fisk} and {Ott, Ruth, Mays}.

From $t = 2$ the clusters remain the same. The SCCA defines the stopping condition as a user-defined number of consecutive identical clusterings. If that number is six, then the final clustering of {Rose, Cobb, Fisk} and {Ott, Ruth, Mays} is determined when $t = 7$. For the reader wondering if the clustering changes at some later point, the algorithm was run through $t = 1000$ and the same clustering was found at each step.

4. Implementation. As is to be expected with a new algorithm, actual implementation of ideas that looked fine on paper can still be problematic. Even before implementation, there may be concerns about perceived weak links in the algorithm. In this section we will address some of these concerns. Since this section and the results section involve many of the same issues, it will be hard to talk about them without highlighting some of the results to come. Hopefully, no great surprises are spoiled, and the turning of pages back and forth is kept to a minimum.

4.1. Impact of initial probability vectors. The fact that the stochastic consensus clustering algorithm depends on a random initial probability vector (IPV) raises the question of whether all random probability vectors will lead to the same clustering. Since P is irreducible, we are guaranteed that the matrix has a unique stationary distribution vector that is independent of the IPV. But, for clustering purposes, that is not the issue. Instead we would like to have confidence that for a certain IPV, x_t^T will remain in short-term stabilization and middle-run evolution long enough for us to identify the clusters. Second, as we will see soon in section 5, different IPVs can lead to different cluster results.

We will consider the IPV question in two parts. First we address the rare occurrence of an IPV that does not lead to a clustering at all, and then we address the fact that different IPVs can lead to different clusterings.

4.2. IPVs leading to no solution. Clearly not every initial probability vector will help us in data clustering. Suppose, for example, that

$$x_0^T = \begin{pmatrix} \frac{1}{n} & \frac{1}{n} & \frac{1}{n} & \dots & \frac{1}{n} \end{pmatrix}_{1\times n}.$$

Since $P_{n\times n}$ is doubly stochastic, x_0^T is its stationary distribution vector. With such a choice for the IPV, x_t^T never changes and we have no ability to group the probabilities in x_t^T in order to cluster the original data.

It is simple enough to make sure that x_0^T is not the uniform distribution vector, but it is equally important that there are enough iterations for the algorithm to recognize either short-term stabilization or middle-run evolution before x_t^T reaches the uniform vector. Since each new x_t^T is the result of the continuous operation of matrix multiplication, x_t^T being close to the uniform distribution vector, ensures that x_{t+1}^T cannot be significantly further away for it. Therefore, even though the algorithm generates x_0^T randomly, the cautious user may want to set a tolerance ϵ and if

$$||x_0^T - (1/n \;\; 1/n \;\; \dots \;\; 1/n)|| < \epsilon,$$

generate another x_0^T. It should be noted that in the preparation of this paper the stochastic consensus clustering algorithm was run hundreds, if not thousands, of times and never was a failure due to an IPV being too close to the uniform distribution.

4.3. IPVs leading to different solutions. The fact that cluster analysis is an exploratory tool means that getting different solutions depending on the IPV is not the end of the road, but rather an opportunity to examine these solutions in the hope of gaining additional insight into the data set's structure.

That said, it would still be instructive to know as much as possible about the characteristics shared by IPVs that lead to the same solution, how many different solutions are possible, and how often each of them is likely to appear. Probabilistic analysis of random starting vectors has been done in the context of iterative methods for finding eigenvalues and eigenvectors [13, 26], and is a natural area for further research on the stochastic consensus clustering method.

4.4. Using a single measure. The workload in consensus clustering is concentrated at the beginning of the process when the large number of clustering results are computed. Even if a user has access to a multiprocessor environment where this work can be shared, it would be advantageous to find a single similarity measure which is compatible with the stochastic consensus clustering algorithm.

Since the SCCA is inspired by the Simon–Ando theory, the underlying matrix must be nearly uncoupled. For a given data set, the problem with most traditional similarity (or dissimilarity) measures is that their values tend to the middle of their range. To illustrate, consider two common similarity measures: Euclidean distance and the cosine measure

$$c(x_1, x_2) = \frac{x_1^T x_2}{||x_1||_2 \, ||x_2||_2}.$$

The former has the advantage of being familiar to almost everyone, while the latter has been found to be particularly useful in text-mining [5]. However, as Figures 4.1 and 4.2 show for the leukemia DNA microarray data set that will be introduced in section 5.2, the distribution of values returned by these two common measures is not the kind of distribution needed to form a nearly uncoupled matrix.

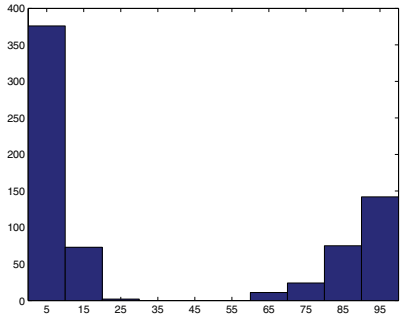

FIG. 4.1. *This is the histogram of the* 703 *similarity values used to build a consensus matrix for the* 38*-element leukemia DNA microarray data set that will be introduced in section* 5.2*. The horizontal axis measures the number of times out of* 100 *that two elements clustered together. The histogram shows that pairs of data points clustered together either a small or large number of times.*

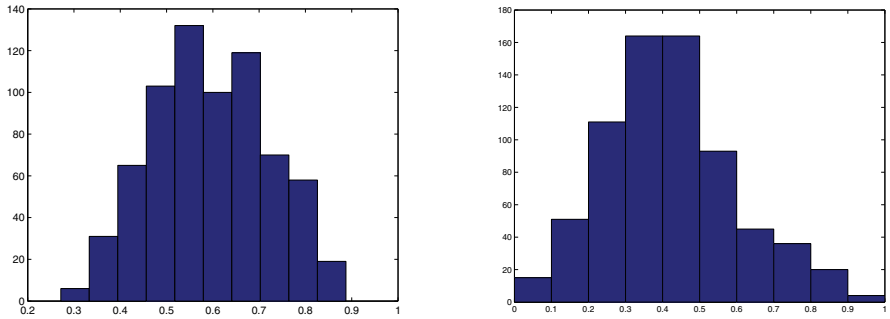

FIG. 4.2. *The histogram on the left shows the distribution of cosine similarity measures between the same elements used for Figure* 4.1*, while the histogram on the right does the same for Euclidean norm values scaled to the interval* $[0, 1]$*. Contrast these distributions with the one shown in Figure* 4.1*.*

In the case of the cosine measure whose range is $[0, 1]$, there have been attempts to "massage" distributions so that they contain more values near the extremes. Such methods often involve changing small values to zero and then performing some arithmetic operation that gives the remaining data a larger variance (for example, squaring each value) [50]. These methods, however, are far from subtle, and in experiments for use with the SCCA, the matrix P went from dense to too sparse for clustering in one iteration of attempting to adjust its values.

Working with the Euclidean norm brings with it the additional requirement large distances need to be mapped to small similarity values while small distances are mapped to large similarity values. A typical function used in making this translation

is a Gaussian of the form

$$f(x_1, x_2) = e^{\frac{-||x_1 - x_2||^2}{2\sigma^2}},$$

where σ is a parameter that typically has to be adjusted for each similarity matrix [36]. This is certainly an area for future study in implementing the SCCA, but so far a reliable way to build a matrix of Gaussians with the distribution required by the SCCA has not been found.

It should be noted that *power iteration clustering* introduced by Lin and Cohen has succeeded in using a single measure to cluster data using an algorithm similar in philosophy to the SCCA. This method uses a row-stochastic Laplacian-like matrix derived from a similarity matrix constructed using the cosine similarity measure [29, 30, 31]. Like the SCCA, clusters are determined by examining intermediate iterates of the power method. It is interesting to note despite mentioning a Gaussian approach to the Euclidean norm in [29], all results in the paper were obtained using either a 0/1 or a cosine measure.

A single measure that has been used with some success involves the idea of nearest neighbors, those data points closest to a given data point using a specific distance measure. For each element g in the data set, let the set $\mathcal{N}_g$ consist of the κ nearest neighbors of g, where the user chooses both the positive integer κ and the distance measure used. The s_{ij} element of the consensus matrix is equal to the number of elements in $\mathcal{N}_i \cup \mathcal{N}_j$ [1].

Work with consensus matrices built in this fashion is still in its initial stages. It has become obvious that the choice of κ and the distance measure greatly affect the results as can be seen in Table 4.1.

TABLE 4.1

Building a consensus matrix based on the number of shared nearest neighbors can work well or poorly depending on the value of κ, the number of nearest neighbors calculated for each data point. The results in this table are from clustering the rather simple, four-cluster Ruspini data set [39]. *When $\kappa = 15$ the stochastic consensus clustering algorithm detects five clusters. This fifth cluster only has one member, while the rest of the solution is correct.*

κ	Clusters	Errors
15	5	1
20	4	0
25	4	18

4.5. No SCCA solution. It is possible for the stochastic consensus clustering algorithm to return no solution. Though rare, there is a way to use the SCCA to produce a clustering in such a case.

1. Run the SCCA, but override its choice for the number of clusters with $k = 2$.
2. If more than two clusters are desired, examine the clusters and use some user-defined criteria to choose a cluster to be split into two by the SCCA.
3. Continue the process from the last step until the target number of clusters is reached.

5. Results. In building test cases for our proposed algorithm, one complication is determining the ensemble used to build the initial similarity matrix S. In the results that follow the ensembles will typically consist of multiple runs of the multiplicative update version of NMF [27] or k-means or a combination of both.[3] In each case,

[3]For an example of how the factors found by NMF are used to cluster data, see [8].

the value or values of k used when calling these algorithms will be noted, though as explained above the new stochastic consensus clustering algorithm will use the number of eigenvalues in the Perron cluster of P to determine k.

5.1. Iris data set. The Fisher iris data set [16] consists of four measurements (petal length, petal width, sepal length, and sepal width) for 150 iris flowers, fifty each from three iris species (*Iris setosa*, *Iris virginica*, and *Iris versicolor*). It is well documented that the *setosa* cluster is easily separable from the other two, but separating the *virginica* and *versicolor* species is more difficult [17].

When building S using NMF the choice of k is limited to two or three since NMF requires k to be less than both the dimension of the data and the number of samples. Running the multiplicative update version of NMF 100 times with $k = 2$ never results in a perfect splitting of *setosa* from the other two species, though there are three or fewer clustering errors 67 times. However, there are six instances of more than 15 errors including a worst case of 26. Despite these problems, the SCCA, using a consensus similarity matrix built from these rather poor results, gets the clustering correct for all but three irises. Although NMF does quite poorly in trying to separate the irises into three clusters, the S derived from these results leads to a perfect two-cluster separation of *setosa* irises from *virginica* and *versicolor* ones.

On the whole, individual clustering results on the iris data set using k-means clustering with $k = 2$ or $k = 3$ are better than those returned by NMF. However, building S using the results from k-means clustering, we get very similar results to what we saw with NMF.

If we decide to build S using $k = 4$ just to see if it will give us any insight into the data set, SCCA recognizes that there are three clusters in the data set, but 16 flowers are misclustered. Though that result may not seem encouraging, notice that this is an improvement over the the range of errors (21–38) when using k-means with $k = 3$.

Finally, the consensus matrices found using NMF and k-means were summed to see if a more robust clustering than the one found by SCCA using S from just one of these methods could be found. Notice that this approach proved fruitful as there is at most one error regardless of the value of k used.

The results from using all of these different consensus matrices are summarized in Table 5.1.

TABLE 5.1

Clustering the iris data set, S created using NMF (first two lines), k-means (next three lines), and a combination of the two (last two lines).

Method and k used to create S	Range of # of errors in single clusterings	k found by SCCA	# of errors in SCCA result
NMF (2)	1–26	2	3
NMF (3)	19–72	2	0
k-means (2)	3	2	3
k-means (3)	21–38	2	0
k-means (4)	n/a	3	16
Combined (2)	1–26	2	1
Combined (3)	19–72	2	0

5.2. Leukemia DNA microarray data set. In 1999 a paper was published analyzing a DNA microarray data set containing the gene expression values for 6817 genes from 38 bone marrow samples [22]. Five years later, the same 38 samples were examined, though this time only 5000 genes were used [7]. The samples came

from leukemia patients who had all been diagnosed with either acute lymphoblastic leukemia (ALL) or acute myeloid leukemia (AML). Additionally, the ALL patients had either the B-cell or T-cell subtype of the disease (ALL-B or ALL-T). This data set is well known in the academic community (Google Scholar reports that the 1999 paper has been cited over 6000 times) and is an excellent test for new clustering algorithms since it can be divided into either two (ALL/AML) or three (ALL-B/ALL-T/AML) clusters. The actual clustering for the leukemia data set is known (see Table 5.2), though the 2004 paper noted that the data "contains two ALL samples that are consistently misclassified or classified with low confidence with most methods. There are a number of possible explanations for this, including incorrect diagnosis of the samples [7]."

TABLE 5.2
The correct clustering of the leukemia DNA microarray data set.

Diagnosis	Patients
ALL-B	1 – 19
ALL-T	20 – 27
AML	28 – 38

Since the 2004 paper was published to demonstrate the effectiveness of nonnegative matrix factorization in clustering this data set, this seems to be an appropriate test for the stochastic consensus clustering algorithm, using NMF with different k values to build the ensemble. The data set was clustered using NMF 100 times each for $k = 2$ and $k = 3$. Additionally, to explore the data set further, the data were clustered an additional 100 times for $k = 4, 5$, and 6.

Figure 5.1a shows the number of errors for each clustering used in building S_2, the $k = 2$ consensus similarity matrix. NMF is clearly quite good at clustering this data set into two clusters, which was the point of [7]. Each time the stochastic consensus clustering algorithm is used to cluster the patients based on S_2, it makes exactly two errors—misclustering Patients 6 and 29.

Similar comparisons were done using S_3, the $k = 3$ consensus similarity matrix, and again the stochastic consensus clustering method could not improve on the already excellent results of NMF. NMF made an average of 3.18 errors per clustering compared to 4.76 for the SCCA. Even the hope that the SCCA would provide a narrower band of errors than NMF is not realized (see Table 5.1b of Figure 5.1). Perhaps the lesson is that if the original method does a good job of clustering, then SCCA is not likely to improve on it, though it is also not likely to worsen it.

Since cluster analysis is an exploratory tool, consensus matrices S_4, S_5, and S_6 were constructed to see if either the stochastic consensus clustering algorithm or nonnegative matrix factorization could discover some hidden structure in the data set that would indicate one or more undiscovered clusters. If a group of elements all break away from an existing cluster or clusters, there is reason for further investigation regarding a new cluster. Interestingly, when $k = 4$, the results from both NMF and the SCCA agree. As Table 5.1c of Figure 5.1 summarizes, they both have identified a fourth cluster made up of four ALL-B patients and two AML patients.

Neither of the methods give any indication of further clusters. When $k = 5$ or $k = 6$, both methods begin to build two or three large clusters with the remaining clusters containing only two or three members.

Before we move on to the next data set, there is one other interesting result to report. If the stochastic consensus clustering algorithm is run using the sum of

# of errors	1	2	3	4
# of instances (NMF)	30	65	3	2
# of instances (SCCA)	0	100	0	0

(a) The leukemia DNA microarray data set was clustered 100 times using NMF with $k = 2$. The number of errors ranged between one and four. When the SCCA was used on the consensus matrix created from those 100 NMF clusterings, it misclustered Patients 6 and 29 each time.

# of errors	1	2	3	4	5	6	7	8	9	10+
# of instances (NMF)	0	71	3	9	3	3	1	2	0	8
# of instances (SCCA)	0	67	0	0	0	0	0	0	0	33

(b) Neither the SCCA nor NMF shows an advantage over the other when clustering the consensus matrix S_3.

Diagnosis	Patients	Patients
ALL-B	1 – 19	1, 3, 5, 7 – 9, 11 – 14, 16 – 18
ALL-T	20 – 27	10, 20 – 27
AML	28 – 38	28, 30 – 35, 37, 38
New cluster		4, 6, 19, 29, 36

(c) Both NMF and SCCA agree that there may be a new cluster. The third column shows the membership of this new cluster and the patients remaining in the other three.

FIG. 5.1. *This is a collection of tables that compare the results of clustering consensus matrices constructed using different k-values. The consensus matrices were clustered by both the SCCA and NMF. Table* 5.1*a compares the results for* $k = 2$. *Table* 5.1*b shows very little difference between the two methods when* $k = 3$. *Table* 5.1*c shows a possible fourth cluster suggested by both NMF and SCCA.*

S_2 and S_3, it identifies two clusters and makes only one clustering mistake, namely Patient 29.[4]

5.3. Custom clustering. As we first mentioned in section 4.1, the fact that the stochastic consensus clustering algorithm uses a random initial probability vector means that it can arrive at different solutions, and when clustering the leukemia data set we found this to be so. While this might be viewed as a weakness of the algorithm, it does give the researcher the ability to answer a very specific question by creating a specific initial probability vector.

In section 5.2, we noticed that the SCCA did not cluster the leukemia data set consensus matrix any better than nonnegative matrix factorization. But what if our primary interest was not in clustering the entire data set, but instead in finding the membership of the cluster of a particular data point. For example, if you are the physician for Patient number 2 you have limited interest in a global view of the leukemia data set. Indeed, rather than knowing which of the three clusters Patient 2 belonged to, it would be of greater use to you to know a small number of other patients that are most like Patient 2 in the hope that knowledge would help you tailor the best treatment plan possible.

[4]Throughout the research period for this paper, the Patient 29 sample was misclustered nearly 100 percent of the time. One of the authors of the 2004 paper verifies that in their work, the Patient 29 sample was also often placed in the wrong cluster [48].

Custom clustering algorithm (CCA)

1. Create the consensus similarity matrix S and the doubly stochastic symmetric matrix P just as in the stochastic consensus clustering algorithm.
2. Construct x_0^T to contain all zeros except for a one in the place of the element we are interested in creating a custom cluster for.
3. Pass the algorithm values for the minimum and maximum size cluster you desire and the maximum number of iterations the CCA should take trying to find that cluster.
4. After each $x_t^T = x_t^T P$ multiplication, cluster the elements of x_t^T as in the SCCA. If the cluster containing the target element is within the size parameters, output the cluster and end the program.

FIG. 5.2. *The custom clustering algorithm.*

To create such a custom clustering, we construct an IPV containing all zeros except for a one in the place corresponding to our data point of interest. We then ask the stochastic consensus clustering algorithm to find the cluster containing our specific data point. Since we may be interested in a collection much smaller than that cluster, the stochastic consensus clustering algorithm can be modified to ask for a small number data points whose x_t entries are closest to our target point (see Figure 5.2 for a summary of the custom clustering algorithm).

Here again we find hope in a feature of the SCCA that seemed to disappoint us in section 5.2. In that section, the clustering of consensus matrices built from methods using $k = 5$ and $k = 6$ seemed to supply new information. In fact, the small clusters found then are indicative of an especially close relationship between the cluster members.

Incorporating these ideas using the consensus matrix S_6 from section 5.2 and an initial probability vector of all zeros except for a one in the second position gives us the custom cluster of $\{2, 4, 6, 15, 19, 29, 36\}$, a cluster with four other AML-B patients and two AML patients (although one of them, Patient 29, consistently clusters with the AML-B patients in our experience). These results are presented in Table 5.3 along with the six nearest neighbors of Patient 2 using Euclidean distance and cosine measure. The SCCA's custom cluster for Patient 2 features three patients not found in these nearest neighbor sets and suggests that physicians could learn a great deal by examining these hidden connections between Patient 2 and Patients 15, 29, and 36.

TABLE 5.3

Custom cluster for leukemia Patient 2. *This table shows the six other patients most similar to Patient* 2. *The patients are listed in similarity order, that is, the first one is the one most similar to Patient* 2. *The cluster returned by the SCCA differs by three patients with both lists derived from two traditional distance measures.*

Method	Other patients
SCCA	29, 19, 4, 15, 36, 6
2-norm	19, 16, 9, 3, 6, 18
cosine	16, 19, 9, 3, 18, 4

6. Discussion. These initial tests prove that the SCCA can be an effective clustering tool. As with any new method, this initial promise raises multiple questions for further study, some of which are listed here.

- Use probabilistic analysis of initial probability vectors to see what we can

learn about the number of possible solutions the SCCA can return and whether there is any connection between $\sigma(P, n_1)$ and the tendency of P to produce multiple solutions.

- Devise a fuzzy clustering for a data set based on the multiple results returned when using different initial probability vectors.
- Investigate whether in situations where the stochastic consensus clustering algorithm returns multiple answers, if building a consensus matrix from these results, and applying the SCCA again will eventually yield a unique solution.
- Examine whether the Sinkhorn–Knopp balancing step can be replaced by a simple scaling to make all row sums equal. Though we lose the results from Markov chain theory, perhaps they are unneeded since all we are looking for is x_t^T values that are approximately equal. The work of Lin and Cohen mentioned in section 4.4 would seem to indicate that this is a possibility.
- Continue the search for a single similarity measure whose values are distributed in a way that can be exploited by the stochastic consensus clustering method.
- Improve the bounds for values of d_i. Numerical results indicate that the upper bound found for Theorem 2.13 can be greatly improved.
- Explore the structure of the spectrum of symmetric, irreducible, nearly uncoupled, doubly stochastic matrices. For this paper, we were only concerned with the eigenvalues near one, but from examining eigenvalues during the course of this research, there appears to be some structure to the spectrum, especially a large number of eigenvalues near zero.
- Work to find a tighter bound on the numeric connection between $\lambda_2(P)$ and $\sigma(P, n_1)$ that Theorems 2.15 and 2.16 establishes.

Acknowledgments. The authors would like to thank the referees for helpful comments that made this a much better paper. In particular, two referees brought notable earlier results to our attention for which we are very grateful. Thanks also to Ilse Ipsen for her suggestions and the 2-norm bound in the proof of Theorem 2.15.

REFERENCES

[1] R. Abbey, *Personal communication*, April 28, 2011.
[2] N. Ailon, M. Charikar, and A. Newman, *Aggregating inconsistent information: Ranking and clustering*, J. ACM, 55 (2008), article 23.
[3] R. G. Bartle, *The Elements of Real Analysis*, John Wiley, New York, 1964.
[4] A. Berman and R. J. Plemmons, *Nonnegative Matrices in the Mathematical Sciences*, Academic Press, New York, 1979.
[5] M. W. Berry and M. Browne, *Understanding Search Engines: Mathematical Modeling and Text Retrieval*, 2nd ed., SIAM Book Series: Software, Environments, and Tools, Philadelphia, 2005.
[6] R. A. Brualdi, S. V. Parter, and H. Schneider, *The diagonal equivalence of a nonnegative matrix to a stochastic matrix*, J. Math. Anal. Appl., 16 (1966), pp. 31–50.
[7] J.-P. Brunet, P. Tamayo, T. R. Golub. J. P. Mesirov, and E. S. Lander, *Metagenes and molecular pattern discovery using matrix factorization*, Proc. Natl. Acad. Sci. USA, 101 (2004), pp. 4164–4169.
[8] T. Chartier and C. Wessell, *A nonnegative analysis of politics*, Math Horizons, 18 (2011), pp. 10–13.
[9] P. J. Courtois, *Decomposability: Queueing and Computer System Applications*, Academic Press, New York, 1977.
[10] J. Csima and B. N. Datta, *The DAD theorem for symmetric non-negative matrices*, J. Combinatorial Theory Ser. A, 12 (1972), pp. 147–152.
[11] P. Deuflhard, W. Huisinga, A. Fischer, and C. Schütte, *Identification of almost invariant*

aggregates in reversible nearly uncoupled Markov chains, Linear Algebra Appl., 315 (2000), pp. 39–59.

[12] P. Deuflhard and M. Weber, *Robust Perron cluster analysis in conformation dynamics*, Linear Algebra Appl., 398 (2005), pp. 161–184.

[13] J. D. Dixon, *Estimating extremal eigenvalues and condition numbers of matrices*, SIAM J. Numer. Anal., 20 (1983), pp. 812–814.

[14] V. Filkov and S. Skiena, *Integrating microarray data by consensus clustering*, International Journal on Artificial Intelligence Tools, 13 (2004), pp. 863–880.

[15] V. Filkov and S. Skiena, *Heterogeneous data integration with the consensus clustering formalism*, Proceedings of Data Integration in the Life Sciences, Springer, Berlin, 2004, pp. 110–123.

[16] R. A. Fisher, *The use of multiple measurements in taxonomic problems*, Annals of Eugenics, 7 (1936), pp. 179–188.

[17] A. Fred and A. K. Jain, *Data clustering using evidence accumulation*, in Proceedings of the 16th International Conference on Pattern Recognition, Quebec, Canada, IEEE Computer Society, Washington, DC, 2002, pp. 276–280.

[18] A. Fred and A. K. Jain, *Robust data clustering*, in Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Madison, WI, Vol. 2, IEEE Computer Society, Los Alamitos, CA, 2003, pp. 128–133.

[19] D. Fritzsche, V. Mehrmann, D. B. Szyld, and E. Virnik, *An SVD approach to identifying metastable states of Markov chains*, Electron. Trans. Numer. Anal., 29 (2008), pp. 46–69.

[20] A. Gionis, H. Mannila, and P. Tsaparas, *Clustering aggregation*, ACM Trans. Knowl. Discov. Data, 1 (2007), article 4.

[21] A. Goder and V. Filkov, *Consensus clustering algorithms: Comparison and refinement*, in Proceedings of the Tenth Workshop on Algorithm Engineering and Experiments, SIAM, Philadelphia, 2008, pp. 109–117.

[22] T. R. Golub, D. K. Slonim, P. Tomayo, C. Huard, M. Guassenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander, *Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring*, Science, 286 (1999), pp. 531–537.

[23] T. Grotkjaer, O. Winther, B. Regenberg, J. Nielsen, and L. K. Hansen, *Robust multi-scale clustering of large DNA microarray datasets with the consensus algorithm*, Bioinformatics, 22 (2006), pp. 58–67.

[24] D. J. Hartfiel and C. D. Meyer, *On the structure of stochastic matrices with a subdominant eigenvalue near* 1, Linear Algebra Appl., 272 (1998), pp. 193–203.

[25] M. N. Jacobi, *A robust spectral method for finding lumpings and meta stable states of non-reversible Markov chains*, Electron. Trans. Numer. Anal., 37 (2010), pp. 296–306.

[26] E. R. Jessup and I. C. F. Ipsen, *Improving the accuracy of inverse iteration*, SIAM J. Sci. Statist. Comput., 13 (1992), pp. 550–572.

[27] D. D. Lee and H. S. Seung, *Learning the parts of objects by non-negative matrix factorization*, Nature, 401 (1999), pp. 788–791.

[28] S. A. Levin, *The problem of pattern and scale in ecology: The Robert H. MacArthur award lecture*, Ecology, 73 (1992), pp. 1943–1967.

[29] F. Lin and W. W. Cohen, *Power iteration clustering*, Proceedings of the 27th International Conference on Machine Learning, Omnipress, Madison, WI, 2010, pp. 655–662.

[30] F. Lin and W. W. Cohen, *A very fast method for clustering big text datasets*, Proceedings of the 2010 conference on ECAI 2010: 19th European Conference on Artificial Intelligence, IOS Press, Amsterdam, 2010, pp. 303–308.

[31] M. Meila and J. Shi, *A Random Walks View of Spectral Segmentation*, in Proceedings of the 8th International Workshop on Artificial Intelligence and Statistics, AISTATS, 2001.

[32] C. D. Meyer, *Stochastic complementation, uncoupling Markov chains, and the theory of nearly reducible systems*, SIAM Rev., 31 (1989), pp. 240–272.

[33] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*, SIAM, Philadelphia, 2000.

[34] H. Minc, *Nonegative Matrices*, John Wiley & Sons, New York, 1988.

[35] S. Monti, P. Tamayo, J. Mesirov, and T. Golub, *Consensus clustering: A resampling-based method for class discovery and visualization of gene expression microarray data*, Machine Learning, 52 (2003), pp. 91–118.

[36] S. Mouysset, J. Noailles, and D. Ruiz, *Using a global parameter for Gaussian affinity matrices in spectral clustering*, in High Performance Computing for Computational Science (VECPAR 2008), J. M. Palma, P. R. Amestoy, M. Daydé, M. Mattoso, and J. C. Lopes, eds., Springer-Verlag, Berlin, 2008, pp. 378–390.

[37] B. N. Parlett, *The Symmetric Eigenvalue Problem*, SIAM, Philadelphia, PA, 1998.

[38] S. Race, *Data Clustering via Dimension Reduction and Algorithm Aggregation*, Master's thesis, North Carolina State University, Raleigh, NC, 2008.
[39] E. H. Ruspini, *Numerical methods for fuzzy clustering*, Inform. Sci., 2 (1970), pp. 319–350.
[40] N. A. Salingaros, *Complexity and urban coherence*, J. Urban Design, 5 (2000), pp. 291–316.
[41] H. A. Simon and A. Ando, *Aggregation of variables in dynamic systems*, Econometrica, 29 (1961), pp. 111–138.
[42] H. A. Simon, *Near decomposability and the speed of evolution*, Industrial and Corporate Change, 11 (2002), pp. 587–599.
[43] R. Sinkhorn, *A relationship between arbitrary positive matrices and doubly stochastic matrices*, Ann. Math. Statist., 35 (1964), pp. 876–879.
[44] R. Sinkhorn and P. Knopp, *Concerning nonnegative matrices and doubly stochastic matrices*, Pacific J. Math., 21 (1967), pp. 343–348.
[45] O. Sporns, G. Tononi, and G. M. Edelman, *Connectivity and complexity: The relationship between neuroanatomy and brain dynamics*, Neural Networks, 13 (2000), pp. 909–922.
[46] W. J. Stewart, *An Introduction to the Numerical Solution of Markov Chains*, Princeton University Press, Princeton, NJ, 1994.
[47] A. Strehl and J. Ghosh, *Cluster ensembles—A knowledge reuse framework for combining multiple partitions*, J. Mach. Learn. Res., 3 (2002), pp. 583–617.
[48] P. Tamayo, *Personal communication*, May 9, 2011.
[49] R. M. Tifenbach, *On an SVD-based algorithm for identifying meta-stable states of Markov chains*, Electron. Trans. Numer. Anal., 38 (2011), pp. 17–33.
[50] S. van Dongen, *Graph Clustering by Flow Simulation*, Ph.D. thesis, University of Utrecht, Utrecht, The Netherlands, 2000.