A Faster Rankability Measure

SVD Rankability

Paul AndersonAmy N. Langville

This chapter presents a *linear algebraic approach* to solving the rankability problem, contrasting it with the combinatorial approach presented in Chapter 3. The linear algebraic approach is more computationally efficient than the combinatorial approach, but provides a bit less rankability information.

5.1 THE DATA AND VARIOUS ENCODINGS

Let $\boldsymbol{A}$ be an $n \times n$ matrix where a_{ij} represents the *rating differential* of item i over item j. Consider the following small 4-team sports example:

$$\boldsymbol{A} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0 & 1 & 3 & 4 \\ -1 & 0 & 2 & 3 \\ -3 & -2 & 0 & 1 \\ -4 & -3 & -1 & 0 \end{pmatrix} \end{array}.$$

The $(1,3)$-element, $a_{1,3} = 2$, means that team 1 beat team 3 by 2 points. Because i's win over j is also j's loss to i, $\boldsymbol{A}$ is skew-symmetric with this definition of a_{ij}. Other encodings may not create a skew-symmetric matrix and this property is not required for the analysis of this chapter. Row i of the data matrix $\boldsymbol{A}$ contains item i's opinions about the strength of its opponents relative to itself. The diagonal element $(i,i) = 0$ since a team does not play itself, and other teams are judged relative to this. Thus, from team i's perspective, teams that beat i by many points are considered strong and teams that team i beat by many points are considered weak.

This same $\boldsymbol{A}$ matrix could also represent data from another application, such as Amazon products. In this case $a_{1,3} = 2$ might mean that product 1 has two more users who rated it above product 3. Another encoding defines $a_{1,3}$ as product 1's cumulative rating differential over product 3. Yet another encoding defines $a_{1,3}$ as product 1's average rating differential over product 3.

5.2 THE PERFECT INPUT MATRIX

Data can be said to be *perfectly rankable* if $a_{ij} = r_i - r_j$ where r_i is the rating of item i. In the perfect case, there is an underlying rating vector $\boldsymbol{r}$ and all matchups between pairs of items follow this exactly, so that the actual rating differential in a matchup between i and j is the ideal differential $r_i - r_j$. When $a_{ij} = r_i - r_j \ \forall i, j$, then $\boldsymbol{A} = \boldsymbol{e}\boldsymbol{r}^T - \boldsymbol{r}\boldsymbol{e}^T$, where $\boldsymbol{e}$ is the vector of all ones, and $\boldsymbol{A}$ is a rank-two matrix.

An $n = 4$ example of a perfectly rankable dataset with underlying rating vector $\boldsymbol{r} = \begin{pmatrix} 0 & 1 & 2 & 3 \end{pmatrix}^T$ is

$$\boldsymbol{A} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0 & 1 & 2 & 3 \\ -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 \\ -3 & -2 & -1 & 0 \end{pmatrix} \end{array}. \tag{5.1}$$

Team 1 beat team 2 by one point. They beat team 3 by more, and team 4 by even more. Team 2 beat team 3 and team 4 by even more and so on. In other work [**?**], we said that such a matrix was in *hillside form* since a 3-dimensional spy plot of the matrix resembles a hill. Each row (or column because $\boldsymbol{A} = \boldsymbol{e}\boldsymbol{r}^T - \boldsymbol{r}\boldsymbol{e}^T$ is skew-symmetric) gives an item's rating, and hence ranking, of the other items. Notice, for this example and all perfect $\boldsymbol{A}$ matrices, that each item agrees on the ranking of the items, in this case ranking them from first to last as $\begin{pmatrix} 1 & 2 & 3 & 4 \end{pmatrix}$.

An $n \times n$ matrix $\boldsymbol{A}$ is **perfectly rankable** if there exists some rating vector $\boldsymbol{r}$ such that $\boldsymbol{A} = \boldsymbol{e}\boldsymbol{r}^T - \boldsymbol{r}\boldsymbol{e}^T$. As a consequence, a perfectly rankable $\boldsymbol{A}$ is a skew-symmetric rank-2 matrix.

The goal of this chapter is this: given a data matrix, undercover some hidden rating vector $\boldsymbol{r}$ and if one does not exist, give an indication of how far the matrix is from having such an underlying rating.

5.3 THE MATRIX AS A DATA CLOUD

Each row of an $n \times n$ data matrix can be thought of as a point in $\Re^n$. Due to skew-symmetry, the columns represent a cloud of equivalent data points. These points, i.e., the row vectors, are centered about a row mean μ_r^T. When the mean is subtracted from each point, the data cloud is now centered about the origin. See Figure 5.1 for an $n = 3$ depiction. Centering is a geometric property that is helpful for the subsequent analysis.

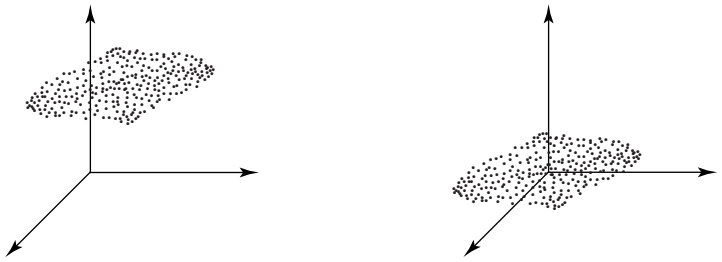

Figure 5.1: **Data Cloud and Centered Cloud.** After centering, the data cloud of $\Re^3$ points on the left shifts to the cloud on the right with mean at the origin.

Centering a Matrix

Centering a matrix by its rows means subtracting the row mean from each row so that the **row-centered matrix** $\boldsymbol{C}_r = \boldsymbol{A} - \boldsymbol{e}\boldsymbol{\mu}_r^T$, where the row mean $\boldsymbol{\mu}_r^T$ of a matrix $\boldsymbol{A}$ is $\boldsymbol{\mu}_r^T = \frac{1}{n}\boldsymbol{e}^T\boldsymbol{A}$. After centering, the $\Re^n$ points defined by the rows of $\boldsymbol{C}$ are centered about the origin.

Column-centering a matrix is defined similarly so that $\boldsymbol{C}_c = \boldsymbol{A} - \boldsymbol{\mu}_c\boldsymbol{e}^T$, where $\boldsymbol{\mu}_c = \frac{1}{n}\boldsymbol{A}\boldsymbol{e}$.

Besides its obvious geometric interpretation, centering also has an algebraic interpretation. Centering can be considered a type of normalization since each data vector is adjusted according to average rating differentials. To understand this, consider a sports application, where a_{ij} represents the number of points by which team i beat team j. Then the j^{th} element of the row mean $\mu_r^T = \frac{1}{n}\boldsymbol{e}^T\boldsymbol{A}$ gives the average column sum, which, in this case, is the average number of points by which team j loses. Thus, the (i,j)-element of the centered matrix c_{ij} is a_{ij} minus the average number of points by which team j loses. In other words, c_{ij} takes the number of points by which i beat j and normalizes by subtracting how many points j is typically beaten by. For example, suppose $a_{ij} = 50$, meaning team i crushed team j beating them by 50 points. Yet suppose $[\mu_r]_j = 45$, which means that team i's dominance over j was not uncommon for j, as they were beaten on average by 45 points. As a result, the normalized $c_{ij} = 5$ gives a fairer representation of the strength of i's win over j than the unnormalized $a_{ij} = 50$.

Lemma 5.1. *After centering, a perfectly rankable (and, hence, rank-2) matrix* $\boldsymbol{A}$ *results in a centered matrix* $\boldsymbol{C}$ *that is rank-1.*

Proof.

$$\begin{aligned}\boldsymbol{C}_r &= \boldsymbol{A} - \boldsymbol{e}\boldsymbol{\mu}_r^T \\ &= \boldsymbol{e}\boldsymbol{r}^T - \boldsymbol{r}\boldsymbol{e}^T - \boldsymbol{e}\boldsymbol{\mu}_r^T \\ &= \boldsymbol{e}\boldsymbol{r}^T - \boldsymbol{r}\boldsymbol{e}^T - \boldsymbol{e}(\boldsymbol{r}^T - \frac{\boldsymbol{e}^T\boldsymbol{r}}{n}\boldsymbol{e}^T) \\ &= (\frac{\boldsymbol{e}^T\boldsymbol{r}}{n}\boldsymbol{e} - \boldsymbol{r})\boldsymbol{e}^T \\ &= -\boldsymbol{\mu}_r\boldsymbol{e}^T.\end{aligned}$$

□

In Section 5.5, we build our rankability measure around the fact that, after centering, a perfectly rankable matrix is rank-1.

5.4 IMPERFECT INPUT MATRICES

Data that does not follow the definition of perfect rankability (i.e., there does not exist an $\boldsymbol{r}$ such that $\boldsymbol{A} = \boldsymbol{e}\boldsymbol{r}^T - \boldsymbol{r}\boldsymbol{e}^T$) is said to be **imperfect** with respect to ranking. It is natural to ask just how imperfect an imperfect data matrix is. Is one matrix more imperfect than another? By how much? We will ultimately build a measure of imperfection in terms of the distance the matrix is from known perfection. In other words, while there is no rating vector $\boldsymbol{r}$ to find for imperfect data, we can find how far the matrix is from having such an $\boldsymbol{r}$.

Consider the *slightly imperfect* matrix

$$\boldsymbol{A} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0 & 1 & 2 & 3 \\ -1 & 0 & 1 & 4 \\ -2 & -1 & 0 & 1 \\ -3 & -4 & -1 & 0 \end{pmatrix} \end{array}. \tag{5.2}$$

Later we will have enough machinery to quantify the level of imperfection. For now, it is enough to notice that intuitively this matrix is not far from the perfectly rankable $\boldsymbol{A}$ of Equation 5.1. When considered as individual ratings of items, the rows of this matrix rank the items rather consistently from first to last as $\begin{pmatrix}1 & 2 & 3 & 4\end{pmatrix}$ with row 4 containing the lone deviation from this ranking, having flip flopped its rankings of items 1 and 2.

On the other hand, the matrix $\boldsymbol{A} = \boldsymbol{e}\boldsymbol{e}^T$, the matrix of all ones, seems *extremely imperfect*. Its rows rank each item the same, making the items indistinguishable from another and hence providing no rankability information. The aim of the next section is to quantify the intuitive observations from these small examples.

5.5 THE LINEAR ORDERING PROBLEM AND THE SVD

Most ranking methods, such as the famous PageRank for ranking webpages [**?**] or Massey for ranking sports teams [**?**], create a rating vector of items that, when *ordered*, gives a ranking of the items. For this reason, ranking is sometimes referred to as the *linear ordering problem* or LOP. A geometric way to think about the LOP is that its goal is to take $\Re^n$ points and represent them as $\Re^1$ points, i.e., scalars. Forcing high-dimensional points onto the $\Re^1$ number *line* creates a linear ordering. In this section, we describe how the singular value decomposition (SVD) relates to linear ordering.

5.5.1 Singular Values

Every matrix has a singular value decomposition. Specifically, each $\boldsymbol{C} \in \Re^{m\times n}$ of rank r has orthogonal matrices $\boldsymbol{U}_{m\times m} = \begin{pmatrix}\boldsymbol{u}_1 & | & \boldsymbol{u}_2 & | & \cdots & | & \boldsymbol{u}_m\end{pmatrix}$ and $\boldsymbol{V}_{n\times n} = \begin{pmatrix}\boldsymbol{v}_1 & | & \boldsymbol{v}_2 & | & \cdots & | & \boldsymbol{v}_m\end{pmatrix}$ and a diagonal matrix $\boldsymbol{D}_{r\times r} = diag(\sigma_1, \sigma_2, \ldots, \sigma_r)$ such that

$$\boldsymbol{C} = \boldsymbol{U}\begin{pmatrix}\boldsymbol{D} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{0}\end{pmatrix}\boldsymbol{V}^T = \sum_{i=1}^{r}\sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T,$$

where $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$. Let σ be the set of singular values including the zero singular values. This SVD can be used to understand the rankability of the square data matrices that we are studying in this chapter. The first measure that we consider, s_1, defined below, gives an indication of how much the first singular triplet contributes to the centered matrix $\boldsymbol{C}$.

Rankability Measure s_1

The contribution of the first singular triplet to the centered matrix $\boldsymbol{C}$ is defined as

$$s_1(\boldsymbol{C}) = \frac{\sigma_1}{\sum_{i=1}^{n}\sigma_i},$$

where σ_i is the i^{th} (sorted from largest to smallest) singular value of $\boldsymbol{C}$.

To understand this new measure, we apply s_1 to several examples. First, the extreme case of a perfectly rankable matrix. Every perfectly rankable matrix has just one nonzero singular value σ_1 and all other $\sigma_i = 0$ so that s_1 achieves its upperbound of 1. This means that the first singular triplet completely describes the centered matrix. For example, below are the set σ of singular values and

the s_1 score for the perfectly rankable matrix $\boldsymbol{A}_1$ shown here.

$$\boldsymbol{A}_1 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0 & 1 & 2 & 3 \\ -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 \\ -3 & -2 & -1 & 0 \end{pmatrix} \end{array}, \quad \sigma = \{4.47, 0, 0, 0\}, \quad s_1 = 1.$$

Slightly modifying the $\boldsymbol{A}_1$ matrix by increasing the $(2,4)$-element to 4 results in the matrix $\boldsymbol{A}_2$ shown below with its singular values and s_1 score. This small modification to the perfectly rankable matrix opens the door to debate the final ranking of these four items. In particular, notice in row 4 that item 4's ranking of the four items deviates a bit from the rankings generated by the three other rows of the matrix. This inconsistency is captured in the lower s_1 score of .72.

$$\boldsymbol{A}_2 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0 & 1 & 2 & 3 \\ -1 & 0 & 1 & 4 \\ -2 & -1 & 0 & 1 \\ -3 & -4 & -1 & 0 \end{pmatrix} \end{array}, \quad \sigma = \{5.61, 1.41, .71, 0\}, \quad s_1 = .72.$$

Suppose we modify the perfectly rankable matrix $\boldsymbol{A}_1$ a bit more, generating the matrix $\boldsymbol{A}_3$ below. Notice in row 2 of $\boldsymbol{A}_3$ that the ranking of the third and fourth items are flip flopped from the ranking in the perfect rankable $\boldsymbol{A}_1$. This flip flopping affects the two subsequent rows in $\boldsymbol{A}_3$. Now row 3 has trouble distinguishing between the ranking of items 1 and 2 and row 4 has trouble distinguishing between the ranking of items 2 and 3. Once again, the s_1 measure has dropped a bit to .64, appearing to capture the reduction in rankability.

$$\boldsymbol{A}_3 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0 & 1 & 2 & 3 \\ -1 & 0 & 2 & 1 \\ -2 & -2 & 0 & 1 \\ -3 & -1 & -1 & 0 \end{pmatrix} \end{array}, \quad \sigma = \{4.32, 1.22, 1.16, 0\}, \quad s_1 = .64.$$

The s_1 scores for the perfectly rankable and two near perfectly rankable matrices $\boldsymbol{A}_1$, $\boldsymbol{A}_2$, $\boldsymbol{A}_3$ follow our intuitive notions about rankability. But how does s_1 do with other matrices? Consider some matrices intuitively thought to be much less rankable, the randomly generated data matrix A_4 and the empty matrix $\boldsymbol{A}_5$ below. The empty matrix has no ranking information and hence gets

the worst possible s_1 score.

$$\boldsymbol{A}_4 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0 & -2 & 3 & -4 \\ 2 & 0 & -1 & 1 \\ -3 & 1 & 0 & 5 \\ 4 & -1 & -5 & 0 \end{pmatrix} \end{array}, \quad \sigma = \{7.38, 7.04, 1.22, 0\}, \quad s_1 = .47.$$

$$\boldsymbol{A}_5 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \end{array}, \quad \sigma = \{0, 0, 0, 0\}, \quad s_1 = \frac{0}{0}.$$

Similarly, the centered version of the completely connected matrix $\boldsymbol{A} = \boldsymbol{e}\boldsymbol{e}^T$ has singular values $\sigma = \{0, 0, \ldots, 0\}$ so that $s_1 = \frac{0}{0}$ is also undefined.

At this point, it appears that the s_1 score provides a good start for a rankability measure. Yet the examples of the next section show that, by itself, s_1 may not give a complete rankability picture. Fortunately, more information from the singular value decomposition, namely the singular vectors, adds some missing details.

5.5.2 The First Singular Vectors

Visualize once more the data cloud of the row-centered points in $\Re^n$. It is known from [**?**] and [**?**] that the first right singular vector $\boldsymbol{v}_1$ of $\boldsymbol{C}$ gives the direction of principal trend of a row-centered dataset $\boldsymbol{C}$.[1] In other words, of all vectors passing through the mean of the data cloud, $\boldsymbol{v}_1$ is the one that best captures the trend of the data. The vector $\boldsymbol{v}_1$ is also the direction of minimal deviation in the sense that the total sum of squares of orthogonal deviations of the data points from the trend line (the line extended through the trend vector $\boldsymbol{v}_1$) is minimal. See Figure 5.2.

Let $\boldsymbol{c}_j$ be the j^{th} $\Re^n$ data point, i.e., the j^{th} row of $\boldsymbol{C}$. Orthogonally projecting $\boldsymbol{c}_j$ onto the the trend line creates the projected point $\hat{\boldsymbol{c}_j}$, which is given by $\hat{\boldsymbol{c}_j} = \sigma_1[\boldsymbol{u}_1]_j\boldsymbol{v}_1$ where $[\boldsymbol{u}_1]_j$ is the j^{th} component of the left singular vector $\boldsymbol{u}_1$. Because $\sigma_1 > 0$, the orthogonally projected points are ordered along the trend line according to the values in $[\boldsymbol{u}_1]_j$. In other words, the sorted $\boldsymbol{u}_1$ provides a linear ordering of the data. See Figure 5.3 for two examples. Because $n = 3$, the centered points $\boldsymbol{c}_j$ and their orthogonal projections $\hat{\boldsymbol{c}_j}$ onto the trend line can be shown graphically. In the example on the left, the projected points are evenly spaced along the trend line and it is clear that the ranking of these three items is [1 3 2]. In the example on the right, the projections of the points $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ onto the trend line are near each other, while the projection of $\boldsymbol{c}_3$ is much father away. This means that the data reveals that the third item is clearly ranked below the first two items, yet it is harder to make a conclusion

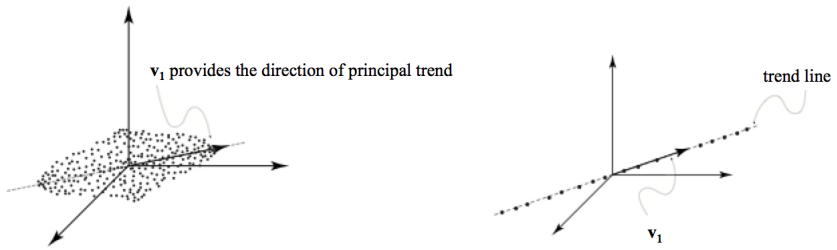

Figure 5.2: **Principal Trend and Points Projected onto Trend Line.** The left figure shows the direction of principal trend given by the first right singular vector $\boldsymbol{v}_1$. The right figure shows the projections of the $\Re^n$ points onto the trend line defined by extending the vector $\boldsymbol{v}_1$.

about the dominance relation between items 1 and 2.

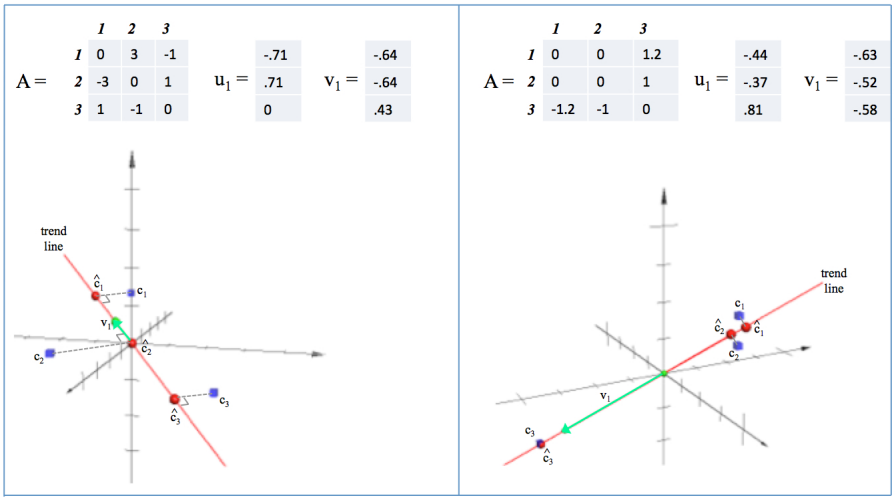

Figure 5.3: **Centered Points c_j, Principal Trend v_1, and Projected Points $\hat{c}_j$.**

Even though we no longer have such visual pictures for the n=4 example matrices $\boldsymbol{A}_1$ through $\boldsymbol{A}_5$ of the previous section, we can still study their singular vectors. In doing so, we discover that the spacing of the elements in $\boldsymbol{u}_1$ provides rankability clues.

The perfectly rankable matrix $\boldsymbol{A}_1$ has the left and right first singular vectors shown below. Notice that the projections in $\boldsymbol{u}_1$ onto the trend line are evenly spaced. Sorting $\boldsymbol{u}_1$ from smallest to largest produces the ranking of the four items from first to last as [1 2 3 4], which matches our intuition regarding the

data in the matrix $\boldsymbol{A}_1$.

$$\boldsymbol{A}_1 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0 & 1 & 2 & 3 \\ -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 \\ -3 & -2 & -1 & 0 \end{pmatrix} \end{array}, \quad \boldsymbol{v}_1 = \begin{pmatrix} -.5 \\ -.5 \\ -.5 \\ -.5 \end{pmatrix}, \quad \boldsymbol{u}_1 = \begin{pmatrix} -.67 \\ -.22 \\ .22 \\ .67 \end{pmatrix}.$$

In fact, perfectly rankable data must adhere to Lemma 5.2.

Lemma 5.2. *A row-centered perfectly rankable matrix $\boldsymbol{C}_r$ has a left singular vector $\boldsymbol{u}_1$ that is a scalar multiple of the row mean $\boldsymbol{\mu}_r^T$ and a right singular vector $\boldsymbol{v}_1$ that is a scalar multiple of $\boldsymbol{e}$, the vector of all ones.*

Proof. By Lemma 5.1, we know that $\boldsymbol{C}_r$ is rank-one and $\boldsymbol{C}_r = -\boldsymbol{\mu}_r \boldsymbol{e}^T$. We also know, because $\boldsymbol{C}_r$ is rank-one, that the first singular triplet represents the matrix exactly so that $\boldsymbol{C}_r = \sigma_1 \boldsymbol{u}_1 \boldsymbol{v}_1$. Thus, $-\boldsymbol{\mu}_r \boldsymbol{e}^T = \sigma_1 \boldsymbol{u}_1 \boldsymbol{v}_1$. Therefore, $\boldsymbol{u}_1$ is a scalar multiple of the row mean $\boldsymbol{\mu}_r^T$ and $\boldsymbol{v}_1$ is a scalar multiple of $\boldsymbol{e}$. □

The next matrix $\boldsymbol{A}_2$ was a slight perturbation of $\boldsymbol{A}_1$, and this is reflected in its singular vectors. Recall that in $\boldsymbol{A}_2$, the first three rows showed agreement ranking the four items from first to last as [1 2 3 4], while the fourth row contained the lone dissenting opinion ranking them as [2 1 3 4]. Notice that $\boldsymbol{u}_1$ reveals this because the projections of items 1 and 2 are much closer in magnitude now.

$$\boldsymbol{A}_2 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0 & 1 & 2 & 3 \\ -1 & 0 & 1 & 4 \\ -2 & -1 & 0 & 1 \\ -3 & -4 & -1 & 0 \end{pmatrix} \end{array}, \quad \boldsymbol{v}_1 = \begin{pmatrix} -.39 \\ -.65 \\ -.39 \\ -.52 \end{pmatrix}, \quad \boldsymbol{u}_1 = \begin{pmatrix} -.53 \\ -.37 \\ .16 \\ .74 \end{pmatrix}.$$

The matrix $\boldsymbol{A}_3$ was also a small perturbation of $\boldsymbol{A}_1$. Its direction of principal trend points in nearly the same direction as that of the perfectly rankable $\boldsymbol{A}_1$.

$$\boldsymbol{A}_3 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0 & 1 & 2 & 3 \\ -1 & 0 & 2 & 1 \\ -2 & -2 & 0 & 1 \\ -3 & -1 & -1 & 0 \end{pmatrix} \end{array}, \quad \boldsymbol{v}_1 = \begin{pmatrix} -.51 \\ -.45 \\ -.57 \\ -.45 \end{pmatrix}, \quad \boldsymbol{u}_1 = \begin{pmatrix} -.68 \\ -.25 \\ .34 \\ .59 \end{pmatrix}.$$

On the other hand, the matrix $\boldsymbol{A}_4$, which was randomly generated, has a direction of principal trend quite different from that of the perfectly rankable $\boldsymbol{A}_1$. The $\boldsymbol{u}_1$ vector ranks these items from first to last as [3 2 4 1], which is

reasonable for this data.

$$\boldsymbol{A}_4 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0 & -2 & 3 & -4 \\ 2 & 0 & -1 & 1 \\ -3 & 1 & 0 & 5 \\ 4 & -1 & -5 & 0 \end{pmatrix} \end{array}, \quad \boldsymbol{v}_1 = \begin{pmatrix} .44 \\ -.29 \\ .08 \\ -.85 \end{pmatrix}, \quad \boldsymbol{u}_1 = \begin{pmatrix} .57 \\ -.04 \\ -.79 \\ .23 \end{pmatrix}.$$

Recall that the singular values of the empty matrix $\boldsymbol{A}_5$ is $\sigma = \{0, 0, 0, 0\}$, which means there is a 4-way tie for the largest singular value σ_1 and hence a 4-way tie for the largest left and right singular vectors $\boldsymbol{u}_1$ and $\boldsymbol{v}_1$. The singular vectors for $\boldsymbol{A}_5$ are the columns of $\boldsymbol{U}$ and $\boldsymbol{V}$ below. Thus, we see that the direction of principal trend varies significantly depending on which column of $\boldsymbol{V}$ is selected. And the columns of $\boldsymbol{U}$ also disagree vastly in the ranking of the items with many ties. This confirms our opinion that the empty matrix $\boldsymbol{A}_5$ is extremely unrankable.

$$\boldsymbol{A}_5 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \end{array}, \quad \boldsymbol{V} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \boldsymbol{U} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

We conclude this section by noting that there are many possible lines on which to force the $\Re^n$ points defined by the rows (or columns) of $\boldsymbol{A}$. We choose to use the SVD because it forces the points onto the *line of maximal trend*, which also happens to be the *line of minimal deviation*, and thus, our use of the SVD is optimal in these two senses.

5.6 TIES IN THE PROJECTED RANKING

The example matrices $\boldsymbol{A}_6$, $\boldsymbol{A}_7$, and $\boldsymbol{A}_8$ of this section examine cases where the data demonstrate greater indecision in the ranking of items. Consider the matrix $\boldsymbol{A}_6$ below. While the singular value score $s_1 = 1$ for this matrix, its pairwise comparisons make it unclear as to whether item 3 is above item 4 or vice versa. This indecision is revealed in the projection vector $\boldsymbol{u}_1$, which projects items 3 and 4 onto the same point on the trend line defined by the direction vector $\boldsymbol{v}_1$. As a result, the sorted $\boldsymbol{u}_1$ produces the ranking of the items from first to last as [1 2 3/4], where the slash symbol (/) represents the tie between items 3 and

4.

$$\boldsymbol{A}_6 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0 & 1 & 2 & 2 \\ -1 & 0 & 1 & 1 \\ -2 & -1 & 0 & 1 \\ -2 & -1 & 0 & 0 \end{pmatrix} \end{array}, \quad s_1 = 1, \quad \boldsymbol{v}_1 = \begin{pmatrix} -.5 \\ -.5 \\ -.5 \\ -.5 \end{pmatrix}, \quad \boldsymbol{u}_1 = \begin{pmatrix} -.75 \\ -.15 \\ .45 \\ .45 \end{pmatrix}.$$

The next example matrix, $\boldsymbol{A}_7$, contains more indecision. It is clear that item 1 outranks the others, yet there is no information to distinguish between items 2, 3, and 4. Once again, the $s_1 = 1$ makes the data appear perfectly rankable because the centered matrix is rank-1. Yet the indecision in the parts of the ranking associated with items 2, 3, and 4 is reflected in $\boldsymbol{u}_1$, which once sorted ranks the items from first to last as $[1 \quad 2/3/4]$ with a 3-way tie at the bottom of the ranking. For comparison, the method of Chapter 3 would produce these same $p = 6$ alternate optimal rankings.

$$\boldsymbol{A}_7 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0 & 1 & 1 & 1 \\ -1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{pmatrix} \end{array}, \quad s_1 = 1, \quad \boldsymbol{v}_1 = \begin{pmatrix} -.5 \\ -.5 \\ -.5 \\ -.5 \end{pmatrix}, \quad \boldsymbol{u}_1 = \begin{pmatrix} -.87 \\ .29 \\ .29 \\ .29 \end{pmatrix}.$$

The matrix $\boldsymbol{A}_8$ presents a final example with ties. The sorted $\boldsymbol{u}_1$ ranks the items from first to last as $[1/2 \quad 3/4]$ with a two 2-way ties at both the top and bottom of the ranking. For comparison, the method of Chapter 3 would produce these same $p = 4$ alternate optimal rankings.

$$\boldsymbol{A}_8 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ -1 & -1 & 2 & 2 \\ -1 & -1 & 2 & 2 \end{pmatrix} \end{array}, \quad s_1 = 1, \quad \boldsymbol{v}_1 = \begin{pmatrix} -.5 \\ -.5 \\ -.5 \\ -.5 \end{pmatrix}, \quad \boldsymbol{u}_1 = \begin{pmatrix} -.5 \\ -.5 \\ .5 \\ .5 \end{pmatrix}.$$

At this point, the question becomes: is s_1 is an adequate rankability measure or should it be modified to make it more compatible with the rankability measure of Chapter 3? On the one hand, perhaps s_1 is enough. The examples $\boldsymbol{A}_6$, $\boldsymbol{A}_7$, $\boldsymbol{A}_8$ all received perfect s_1 scores because one is confident that the set of optimal rankings with ties is complete. In other words, while there may be ties, there is no indecision in the ranking of these items given by the projected ranking vector $\boldsymbol{u}_1$. Said yet another way, if ties are not considered as indicators of indecision, then these data are very rankable. On the other hand, should $\boldsymbol{A}_6$, $\boldsymbol{A}_7$, $\boldsymbol{A}_8$, which have multiple optimal rankings, receive the same rankability score as $\boldsymbol{A}_1$, which has one lone unquestionable ranking of its items? If you answer "no" to this question, likely because you feel ties indicate indecision, then the following

rankability score presents one way to reconcile the SVD rankability measure of this chapter with the combinatorial rankability measure of the previous chapter.

Rankability Measure r_{SVD}

The rankability measure r_{SVD} is defined as

$$r_{SVD} = s_1 \times (1 - \frac{nties}{n-1}),$$

where $nties$ is the number of ties in the projection vector $\boldsymbol{u}_1$.

APPLY r_{SVD} TO ALL EXAMPLES and PUT IN A TABLE

5.7 NEAR TIES IN THE PROJECTED RANKING

Unlike the examples $\boldsymbol{A}_6$, $\boldsymbol{A}_7$, and $\boldsymbol{A}_8$ of the previous section, the matrix $\boldsymbol{A}_9$ below does not have exact ties in the projected ranking $\boldsymbol{u}_1$. Yet it does have a near tie between items 1 and 2. Thus, it is possible for the user to set a "near tie tolerance" $\gamma > 0$. If the difference between two values in the projected ranking $\boldsymbol{u}_1$ is less than γ, then these two items are declared a tie.

$$\boldsymbol{A}_9 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0 & 1 & 2 & 3 \\ -1 & 0 & 3 & 1 \\ -2 & -3 & 0 & 4 \\ -3 & -1 & -4 & 0 \end{pmatrix} \end{array}, \quad s_1 = .51, \quad \boldsymbol{v}_1 = \begin{pmatrix} -.35 \\ -.24 \\ -.87 \\ -.24 \end{pmatrix}, \quad \boldsymbol{u}_1 = \begin{pmatrix} -.45 \\ -.41 \\ .07 \\ .79 \end{pmatrix}.$$

5.8 RANK AGGREGATION

There is a large body of work on the **rank aggregation problem**, which given a set of rankings of n items attempts to form an aggregate or summary ranking that best represents the consensus of the individual rankings []. There are various methods for creating the aggregate ranking. For instance, some methods minimize the average Spearman distance between the aggregate ranking and all individual rankings. Others minimize the total Spearman distance. Others use the Kendall τ distance, p-norm, or AP distance.

The $\boldsymbol{u}_1$ vector of the SVD method of this chapter can be considered an aggregate *rating* in the sense that it minimizes the distance of the individual ratings (i.e., the row vectors in $\boldsymbol{C}$ and hence points in $\Re^n$) to the trend line defined by

the trend vector $\boldsymbol{v}_1$. Thus, we provide a solution to the **rating aggregation problem**. But, of course, every rating, including the SVD aggregate rating, induces a ranking, and hence this SVD method can be viewed as another solution to the rank aggregation problem. In other words, if the rows of the input matrix $\boldsymbol{C}$ are each item's ratings (or rankings) of all items, then the SVD method outputs an aggregate rating, an aggregate ranking, and rankability information.

5.9 LIMITATIONS OF THE SVD METHOD

Discuss scaling issues when the elements in $\boldsymbol{A}$ are of wildly different scales. Possible remedy = damping/skewing functions (see Keener chapter in "Who's #1" Ranking book)

5.10 SUMMARY OF THE SVD RANKABILITY METHOD

The Notation

$\boldsymbol{A}$ item-by-item data matrix containing pairwise relationships such as point differentials
$\boldsymbol{C}$ row-centered data matrix; $\boldsymbol{C} = \boldsymbol{A} - \boldsymbol{e}\boldsymbol{\mu}_r^T$
$\boldsymbol{\mu}_r^T$ row mean of $\boldsymbol{A}$; $\boldsymbol{\mu}_r^T = \frac{1}{n}\boldsymbol{e}^T\boldsymbol{A}$
$\boldsymbol{e}$ vector of all ones;
$\boldsymbol{u}_1$ first left singular vector of $\boldsymbol{C}$
$\boldsymbol{v}_1$ first right singular vector of $\boldsymbol{C}$

The Algorithm

1. Row center the original data $\boldsymbol{A}$, creating the centered data $\boldsymbol{C} = \boldsymbol{A} - \boldsymbol{e}\boldsymbol{\mu}_r^T$.

2. Compute the singular values σ_1 through σ_n of $\boldsymbol{C}$ and the first singular vectors $\boldsymbol{u}_1$ and $\boldsymbol{v}_1$.

3. Calculate the SVD rankability measure r_{SVD}

$$r_{SVD} = \left(\frac{\sigma_1}{\sum_{i=1}^{n}\sigma_i}\right) \times \left(1 - \frac{nties}{n-1}\right),$$

where *nties* is the number of ties in the projection vector $\boldsymbol{u}_1$.

When $\boldsymbol{A}$ is sparse, as it often is for large matrices, Step 1 of the algorithm destroys this property by creating a dense matrix $\boldsymbol{C}$. Fortunately, sparsity can still be exploited by algorithms that compute the SVD of a *rank-one updated matrix*. The centered matrix $\boldsymbol{C} = \boldsymbol{A} - \boldsymbol{e}\boldsymbol{\mu}_r^T$ is a rank-one update of the sparse $\boldsymbol{A}$. In effect, the dense matrix $\boldsymbol{C}$ is never explicitly formed and all operations are done on $\boldsymbol{A}$ and $\boldsymbol{e}\boldsymbol{\mu}_r^T$.

In Step 2, if computing all singular values is too expensive for a large problem, then only the first k singular values may be computed and $s_1 = \frac{\sigma_1}{\sum_{i=1}^{n} \sigma_i}$ can be approximated by $\hat{s}_1 = \frac{\sigma_1}{\sum_{i=1}^{k} \sigma_i}$.

In Step 3, the user can decide if *nties* counts only exact ties or includes near ties as well. In which case, the user must set a tolerance on what defines a near tie.

5.11 COMPARING THE TWO TYPES OF RANKABILITY