

The Shannon entropy of Sudoku matrices

BY PAUL K. NEWTON* AND STEPHEN A. DESALVO

*Department of Aerospace and Mechanical Engineering and Department
of Mathematics, University of Southern California, Los Angeles,
CA 90089-1191, USA*

We study properties of an ensemble of Sudoku matrices (a special type of doubly stochastic matrix when normalized) using their statistically averaged singular values. The determinants are very nearly Cauchy distributed about the origin. The largest singular value is $\sigma_{\max} = 45$, while the others decrease approximately linearly. The normalized singular values (obtained by dividing each singular value by the sum of all nine singular values) are then used to calculate the average Shannon entropy of the ensemble, a measure of the distribution of ‘energy’ among the singular modes and interpreted as a measure of the disorder of a typical matrix. We show the Shannon entropy of the ensemble to be 1.7331 ± 0.0002 , which is slightly lower than an ensemble of 9×9 Latin squares, but higher than a certain collection of 9×9 random matrices used for comparison. Using the notion of *relative entropy* or *Kullback–Leibler divergence*, which gives a measure of how one distribution differs from another, we show that the relative entropy between the ensemble of Sudoku matrices and Latin squares is of the order of 10^{-5} . By contrast, the relative entropy between Sudoku matrices and the collection of random matrices has the much higher value, being of the order of 10^{-3} , with the Shannon entropy of the Sudoku matrices having better distribution among the modes. We finish by ‘reconstituting’ the ‘average’ Sudoku matrix from its averaged singular components.

Keywords: Sudoku matrices; Shannon entropy; singular value decomposition; random matrices; Latin squares; Wishart matrices

1. Introduction

A Sudoku matrix A is a 9×9 real valued matrix with an integer between 1 and 9 in each entry, so long as the following constraints are obeyed:

- each integer can appear only once along any row,
- each integer can appear only once down any column, and
- each integer can appear only once in each of the nine 3×3 sub-blocks.

Without the last ‘regional’ constraint, the matrix is usually called a 9×9 Latin square, whereas, without any of the constraints, if the integer in each entry is chosen with equal probability of $\frac{1}{9}$, we call it a random matrix. It was recently

*Author for correspondence (newton@usc.edu).

2	3	8	5	9	6	4	7	1
6	9	4	1	7	8	3	5	2
5	1	7	3	2	4	9	8	6
1	2	6	9	8	5	7	4	3
7	4	5	2	6	3	8	1	9
9	8	3	7	4	1	2	6	5
4	6	1	8	3	2	5	9	7
3	5	9	4	1	7	6	2	8
8	7	2	6	5	9	1	3	4

Figure 1. An arbitrary Sudoku matrix. Each of the integers 1–9 appears only once down each column, along any row, and in each of the 3×3 sub-blocks. With only the 27 darkened numbers showing, a Sudoku game consists of filling in all the lighter coloured numbers to produce a complete Sudoku matrix.

proven by Felgenhauer & Jarvis (2006) that there are exactly 6, 670, 903, 752, 021, 072, 936, $960 \equiv 9! \times 72^2 \times 2^7 \times 27$, 704, 267, 971 $\sim 6.67 \times 10^{21}$ Sudoku matrices. Their enumeration strategy begins by analysing the permutations of the top 3×3 sub-block used in valid Sudoku grids. Once these sub-block symmetries and equivalence classes for the partial solutions are identified, the other sub-blocks are constructed and counted for each equivalence class. Russell & Jarvis (2006) showed that, if symmetries are taken into account (using Burnside's lemma), there are significantly fewer—only 5, 472, 730, 538 of them. Both of these numbers are small compared with 9^{81} , the total number of random matrices, or the total number of 9×9 Latin squares, which is known to be $O(10^{27})$ (Bammel & Rothstein 1975). In fact, taking the ratio of Felgenhauer and Jarvis' number over 9^{81} gives the minuscule empirical probability of 3.3927×10^{-56} of producing a Sudoku matrix by using a random number generator to select each integer entry independently, without explicitly building in the constraints. An example of a Sudoku matrix is shown in figure 1 and can be found in many major daily newspapers as a popular number game, whose modern version was apparently invented by the American architect Howard Garns in 1979, but whose ancestry dates back at least to Euler. A 'Sudoku game' starts with a few squares filled in (for example, the darkened numbers in figure 1), and the reader must fill in the rest (the lighter numbers in figure 1), obeying the above constraints to create a Sudoku matrix, a classic problem in constrained combinatorial optimization theory (Simonis 2005). If all but one of the squares are filled in initially, it is easy to see that there is a unique value for the remaining square, and hence

the final Sudoku matrix is unique and the problem is said to be well posed. However, if none of the squares are filled in initially, there are many ways to fill them in (ill posed) to arrive at a final Sudoku matrix, namely all of the ones counted by Felgenhauer & Jarvis (2006). The fewest number of filled in squares that renders the solution unique is unknown, although the lowest number yet found is 17. The general problem of solving Sudoku puzzles on $n^2 \times n^2$ boards of $n \times n$ blocks is known to be NP-complete (Yato & Seta 2005). A rapidly evolving discussion of the current state-of-the-art regarding the ‘mathematics of Sudoku’ can be found at the http://en.wikipedia.org/wiki/mathematics_of_sudoku.

In this paper, we focus on the statistical properties of Sudoku matrices, not on strategies for solving Sudoku puzzles. Progress on computer algorithms to solve Sudoku puzzles (using ‘Sinkhorn balancing’) are discussed in Moon *et al.* (2009) and comprehensive strategies are described in Davis (2009). Here, we mostly discuss average properties of ensembles of Sudoku matrices as opposed to specific properties of individual matrices. Of course, we first highlight the main properties that all Sudoku matrices making up an ensemble must share (discussed in the next section). Given the constraints imposed on the entries, we address the question of how random is a Sudoku matrix? We answer this question by creating an ensemble of Sudoku matrices from which we study the statistically averaged singular value distributions and Shannon entropy of the ensemble, which gives us a scalar measure of how evenly distributed are the rank-1 matrices whose linear combination constitutes an ‘average’ matrix, as detailed in the text. In §3, we describe the algorithm we use to generate Sudoku matrices, one of which is shown in figure 1. Section 4 discusses the singular value decomposition of Sudoku matrices, which we then use to calculate their (average) Shannon entropy. For comparison purposes, we calculate the Kullback–Leibler divergence between the ensemble of Sudoku matrices and a corresponding ensemble of Latin squares and random matrices.

2. Basic properties

The following properties hold for any Sudoku matrix:

- Because of constraint 1, every Sudoku matrix A has eigenvalue $\lambda = 45$, with corresponding eigenvector $\eta = (1, 1, 1, \dots, 1)^T$.
- Because of constraint 2, the transpose of every Sudoku matrix A^T has eigenvalue $\lambda = 45$, with corresponding eigenvector $\eta = (1, 1, 1, \dots, 1)^T$.
- Because of constraint 3, we have $A^T \neq A$.
- Because of constraint 3, we have $A^T A \neq A A^T$, and $\lambda = 45^2$ is an eigenvalue of the covariance matrices $A A^T$ and $A^T A$.
- Because of constraint 3, we have $18 \leq \text{trace}(A) \leq 72$. This is because the smallest value for the trace in any 3×3 sub-block is $1 + 2 + 3 = 6$, while the largest value is $7 + 8 + 9 = 24$.

Sudoku matrices can be singular, as the following example shows.

Example 2.1. The following Sudoku matrix has one eigenvalue that is zero, with corresponding eigenvector $\vec{\xi}$:

$$A = \begin{pmatrix} 8 & 3 & 5 & 9 & 4 & 7 & 6 & 2 & 1 \\ 7 & 6 & 1 & 2 & 5 & 8 & 3 & 9 & 4 \\ 2 & 4 & 9 & 6 & 1 & 3 & 5 & 7 & 8 \\ 5 & 1 & 3 & 7 & 8 & 2 & 9 & 4 & 6 \\ 6 & 2 & 4 & 3 & 9 & 1 & 8 & 5 & 7 \\ 9 & 8 & 7 & 4 & 6 & 5 & 1 & 3 & 2 \\ 3 & 7 & 6 & 1 & 2 & 9 & 4 & 8 & 5 \\ 4 & 5 & 2 & 8 & 3 & 6 & 7 & 1 & 9 \\ 1 & 9 & 8 & 5 & 7 & 4 & 2 & 6 & 3 \end{pmatrix} \quad \text{and} \quad \vec{\xi} = \begin{pmatrix} 382 \\ 1723 \\ -554 \\ -122 \\ -1148 \\ -1364 \\ 1669 \\ 355 \\ -941 \end{pmatrix}. \quad (2.1)$$

Sudoku matrices can be highly structured, as in the following example.

Example 2.2. The following Sudoku matrix is produced with an initialized top row containing values $1 \rightarrow 9$. The second row is obtained from the top row by applying a left-shift of three entries (to accommodate the block constraint). The third row is obtained from the second row by applying a left-shift of three entries. The fourth row starts a new 3×3 block; hence, we obtain it from the top row by applying a left-shift of one entry. Rows 5 and 6 are produced from the row above by a left-shift of three entries. Row 7 starts a new block and is obtained from row 4 by applying a left-shift of one entry. The final two rows are obtained from the rows above them by applying the left-shift of three entries. The result is a highly structured yet valid Sudoku matrix,

$$A = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ 4 & 5 & 6 & 7 & 8 & 9 & 1 & 2 & 3 \\ 7 & 8 & 9 & 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 1 \\ 5 & 6 & 7 & 8 & 9 & 1 & 2 & 3 & 4 \\ 8 & 9 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 3 & 4 & 5 & 6 & 7 & 8 & 9 & 1 & 2 \\ 6 & 7 & 8 & 9 & 1 & 2 & 3 & 4 & 5 \\ 9 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{pmatrix}. \quad (2.2)$$

Remark 2.3. Every Sudoku matrix can be made into a doubly stochastic matrix (row and column sums equal to 1) by dividing each entry by 45. This normalized matrix M can then be viewed as a finite Markov chain with nine states, each of which is recurrent and aperiodic (Grimmett & Stirzaker 2005). The stationary distribution, a vector π such that $\pi * M = \pi$, is a left eigenvector of M with eigenvalue 1, and it follows immediately from basic property 2 that the vector of all 1s is a left eigenvector of M ; hence, normalizing to obtain a probability distribution, we have $\pi = (\frac{1}{9}, \dots, \frac{1}{9})$. Note that Latin squares have the same stationary distribution, as they satisfy basic condition 2 as well.

As an indication of the distribution of the eigenvalues (specifically, their products), a histogram of the determinants of an ensemble of 10 000 Sudoku matrices (see §3 for the details of construction) is shown in figure 2 along with

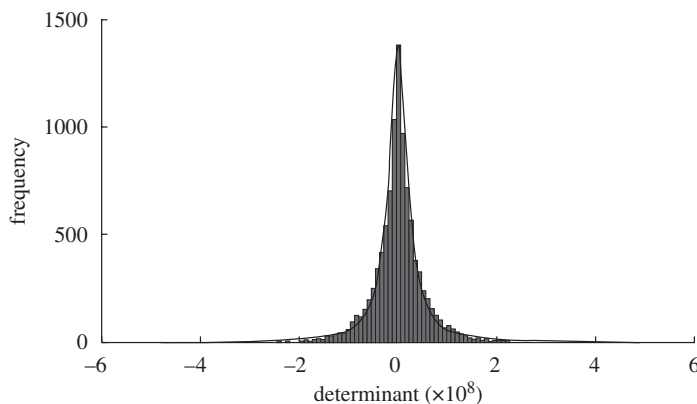


Figure 2. Histogram of determinants of 10 000 Sudoku matrices which appears to closely follow a standard Cauchy distribution, plotted as a solid curve. The sample median is -98820 , and the sample mean is -570352.2660 , with standard deviation $0.56719630860983 \times 10^8$. The smallest determinant from the sample is $-551\,886\,210$ while the largest is $355\,398\,840$.

the sample mean, standard deviation and smallest and largest values from the sample. Plotted together with the histogram is a Cauchy distribution

$$f_{\xi}(x) \equiv C(c, s) = \frac{1}{\pi \cdot s} \frac{1}{1 + ((x - c)/s)^2}, \quad (2.3)$$

which appears to closely model the data (if the entries are not selected uniformly, this would not be the case). Recall that the Cauchy distribution has no defined mean, variance or higher order moments, while its mode and median are c (the peak of the distribution), with s being the scale parameter specifying the half-width at half-maximum. Interestingly, the central limit theorem predicting convergence to a Gaussian distribution does not apply because the variance is not finite. Generically, the Cauchy distribution arises when we take the ratio U/V of two independent Gaussian distributed random variables U , V , with expected values 0 and unit variances. We use the parameter estimation techniques described in Nagy (2006) to determine the parameter values $s = 2.155 \times 10^7$ and $c = -2.902 \times 10^5$. The determinants from the sample range from $\pm 10^8$, with a sample median value $-98\,820$. All of the members of the ensemble had either rank 8 (somewhat special) or rank 9 (more typical) and we believe it is not possible for a Sudoku matrix to have a zero eigenvalue with algebraic multiplicity more than 1, but we have not been able to prove this. The recent article by Dahl (2009) discusses other general properties of Sudoku matrices.

3. The Sudoku algorithm

(a) Algorithm description

An ideal (unbiased) random Sudoku generator would index each of the approximately 10^{21} Sudoku matrices by an integer, and a uniformly distributed random integer between 1 and 10^{21} would be chosen corresponding to one of the matrices. The task of pre-assembling and working with 10^{21} matrices, however, has obvious drawbacks.

Instead, our algorithm for producing Sudoku matrices proceeds one row at a time, starting with the first row and working down to the bottom. First, a *selection array* consisting of the admissible integers 1–9 is created and an index integer j between 1 and 9 is chosen at random, with uniform distribution. The j th number in the selection array is placed in row/column/block = $r/c/b = 1/1/1$ of the Sudoku matrix. This entry is then deleted from the selection array, leaving eight remaining entries. To select the next entry of the Sudoku matrix, a new index integer is picked at random, this time between 1 and 8. This new index is then used to choose the next member of the selection array and placed in $r/c/b = 1/2/1$ of the Sudoku matrix. The process continues until the top row of the Sudoku matrix is filled and all entries of the selection array have been used. The remaining rows are generated in a similar manner, one entry at a time, with the additional preliminary step of reducing the set of admissible values based on the constraints in the corresponding $r/c/b$. In the event of encountering an entry with no possible admissible values, that entire row and the row preceding it are deleted (called the ‘back-stepping’ procedure) and the process continues as before, starting with the first empty row. As an alternative to the back-stepping procedure, we could simply empty the matrix and re-start the algorithm from scratch each time the algorithm runs into a dead end, but it is far faster and more efficient to implement the back-stepping procedure, clearing only two rows. This algorithm seems to be fast and efficient and can easily generate 10 000 Sudoku matrices in a matter of seconds (using C code on a laptop computer).

(b) *Ensemble bias*

The question is whether the algorithm produces an ensemble that is unbiased. Namely, we need the smaller sample produced by the algorithm to have the same statistical properties that a full ensemble of all 10^{21} matrices would have. For this, we describe a *necessary condition* that any unbiased sample ensemble of Sudoku matrices must have. The following simple facts are used. First, note that, given any Sudoku matrix, a new Sudoku matrix can be produced by globally switching any two numbers. For instance, we can switch the numbers 1 and 2 everywhere in the Sudoku matrix (2.2), and the result of the switch will produce a new, less structured, but valid Sudoku matrix. This gives us the following.

Lemma 3.1. *Given a valid Sudoku matrix, we may construct an additional $9! - 1$ distinct Sudoku matrices.*

Proof. Consider the set of all permutations of the numbers 1–9, and create a 1–1 mapping from the set $\{1, \dots, 9\}$ to each of the permutations by matching the n th entry in $\{1, \dots, 9\}$ to the n th entry in the permutation. For each permutation, a distinct Sudoku matrix may be obtained by replacing the number in each corresponding entry of the given Sudoku matrix with the corresponding entry in the permutation. ■

An ensemble of Sudoku matrices produced this way is an *equivalence class*.

Lemma 3.2. *If the values of all Sudoku matrices in an equivalence class are averaged together component-wise, the resulting matrix is the rank-1 matrix with 5 in each entry.*

Proof. The proof is clear by symmetry. Every entry will have each of the numbers 1–9 in it an equal number of times in the ensemble of $9! - 1$ matrices. ■

Thus, as in equation (2.2), we can initialize a given Sudoku matrix by choosing the matrix in the equivalence class that has the first row $\{1, 2, 3, 4, 5, 6, 7, 8, 9\}$. Proceeding this way will give a standardized way of looking at the distribution of matrices generated. Finally, since the matrices in each equivalence class average to the rank-1 matrix with 5 in each entry, we can average these averages and obtain the same matrix. This gives us the following.

Lemma 3.3. *If the values of all Sudoku matrices are averaged together component-wise, the resulting matrix is the rank-1 matrix with 5 in each entry.*

Therefore, we would like any subset of the full set of Sudoku matrices to inherit this property.

Theorem 3.4. *A necessary condition that a set of Sudoku matrices is unbiased is that the component-wise average of the ensemble produce the rank-1 matrix with 5 in each entry.*

The extent to which the sample average deviates from this rank-1 average is a measure of possible bias of the ensemble. Equation (3.1) shows the component-wise average of 10^8 Sudoku matrices generated by the algorithm which is acceptable for our purposes. Of course, more detailed and stringent tests could be developed, as in those discussed in Jacobson & Matthews (1996) for Latin squares, but we regard our relatively weak test sufficient for our purposes.

$$A = \begin{pmatrix} 4.9982 & 4.9975 & 5.0020 & 4.9964 & 4.9995 & 5.0001 & 5.0022 & 5.0019 & 5.0022 \\ 5.0031 & 4.9976 & 5.0001 & 5.0014 & 5.0019 & 4.9999 & 5.0013 & 4.9973 & 4.9973 \\ 5.0017 & 4.9998 & 5.0001 & 5.0006 & 5.0003 & 4.9999 & 4.9985 & 4.9982 & 5.0010 \\ 5.0027 & 4.9997 & 5.0000 & 5.0000 & 5.0000 & 4.9970 & 4.9994 & 4.9998 & 5.0015 \\ 5.0011 & 4.9997 & 4.9969 & 4.9982 & 5.0020 & 5.0017 & 5.0000 & 5.0017 & 4.9986 \\ 5.0002 & 5.0010 & 4.9987 & 5.0031 & 4.9978 & 5.0002 & 4.9997 & 5.0002 & 4.9991 \\ 4.9980 & 5.0002 & 4.9995 & 5.0027 & 4.9974 & 4.9998 & 4.9999 & 5.0015 & 5.0010 \\ 4.9987 & 5.0034 & 5.0017 & 4.9983 & 4.9988 & 4.9995 & 4.9995 & 5.0005 & 4.9996 \\ 4.9962 & 5.0011 & 5.0011 & 4.9993 & 5.0024 & 5.0019 & 4.9995 & 4.9989 & 4.9996 \end{pmatrix}. \quad (3.1)$$

4. Ensemble analysis

Our main tool in the analysis of Sudoku matrices is the singular value decomposition and the resulting calculation of Shannon (1948) entropy for the ensemble of matrices which we now describe.

(a) Singular value decomposition and Shannon entropy

To obtain the singular value decomposition of A , we find the nine eigenvalues λ_i , ($i = 1, \dots, 9$) of the associated covariance matrix $A^T A$

$$A^T A \vec{v}_i = \lambda_i \vec{v}_i, \quad (4.1)$$

where $\lambda_i \geq 0$, $\vec{v}_i \in \mathbb{R}^9$.¹ The singular values σ_i are defined as $\sigma_i = \sqrt{\lambda_i}$. Equivalently, one can define the singular values and singular vectors directly via the system of linear equations

$$A\vec{v}_i = \sigma_i \vec{u}_i, \quad A^T \vec{u}_i = \sigma_i \vec{v}_i, \quad (4.2)$$

where $\sigma_{\max} \equiv \sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_{\min} \equiv \sigma_9 \geq 0$, and the *unit* vectors \vec{v}_i and \vec{u}_i are called the right and left singular vectors, respectively. We use them as columns to construct the 9×9 orthogonal matrices U , V defined as

$$U = (\vec{u}_1 \quad \vec{u}_2 \quad \dots \quad \vec{u}_9), \quad V = (\vec{v}_1 \quad \vec{v}_2 \quad \dots \quad \vec{v}_9), \quad (4.3)$$

which produces the singular value decomposition of A (Kirby 2001; Trefethen & Embree 2005)

$$A = U \Sigma V^T = \sum_{i=1}^9 \sigma_i \vec{u}_i \vec{v}_i^T. \quad (4.4)$$

Σ is the diagonal matrix with singular values down the diagonal, ordered from largest (top left) to smallest (bottom right), with corresponding right and left singular vectors filling in the columns of V and U from left to right,

$$\Sigma = \begin{pmatrix} \sigma_{\max} & 0 & \dots & 0 & 0 \\ 0 & \sigma_2 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & \dots & \sigma_8 & 0 \\ 0 & \dots & \dots & 0 & \sigma_{\min} \end{pmatrix}. \quad (4.5)$$

From equation (4.4), one can see that the singular value decomposition expresses A as the sum of rank-1 matrices, and does so in an optimal way. Namely, if we define the rank- k approximation to A , where $k < 9$, by forming the sum

$$A_k = \sum_{i=1}^k \sigma_i \vec{u}_i \vec{v}_i^T, \quad (4.6)$$

then $\|A - A_k\|_2 \equiv \sigma_{k+1}$, where $\|\cdot\|_2$ represents the 2-norm. It is a standard theorem in linear algebra (Kirby 2001) that any matrix B that is not the rank- k approximation (4.6) has greater error

$$\|A - A_k\|_2 \leq \|A - B\|_2, \quad (4.7)$$

where B is any 9×9 matrix of rank k . Each of the matrices $\vec{u}_i \vec{v}_i^T$ in equation (4.6) are rank 1 and should be thought of as the ‘normal modes’ whose weighted linear combination produces the matrix A_k .

It is useful to normalize the singular values by dividing each by the sum of all nine, so that the normalized values lie in the range between 0 and 1 and can be

¹We note that there are similarities here to the well-studied Wishart matrices, whose distribution arises from the sample covariance matrix obtained from a multivariate normal distribution. These arise frequently in the study of spectral theory of random matrices (Sengupta & Mitra 1999). Also note that sample covariance matrices are well known to be sensitive to statistical outliers.

used to determine the distribution of ‘energy’ over all the singular modes. The normalized values are defined as

$$\hat{\sigma}_i = \sigma_i / \sum_{i=1}^9 \sigma_i, \quad 0 \leq \hat{\sigma}_i \leq 1. \quad (4.8)$$

From these, we define the Shannon entropy to be

$$H = - \sum_{i=1}^9 \hat{\sigma}_i \ln(\hat{\sigma}_i), \quad (4.9)$$

which provides us with a scalar measure of the distribution of ‘energy’ among the modes, which is typically interpreted as the level of disorder, or randomness, associated with the matrices as well as a measure of how rapidly the normalized singular values decay from the peak ($i = 1$). The next two examples give further insight into the relation between the Shannon entropy of a matrix and its singular value distribution.

Example 4.1. All orthogonal matrices have the property $A^T = A^{-1}$, hence $AA^T = I$. Therefore, each eigenvalue of the covariance matrix $\lambda_i = 1$, which when normalized gives equal singular values $\hat{\sigma}_i = \frac{1}{9}$, ($i = 1, \dots, 9$). Therefore, the Shannon entropy for any 9×9 orthogonal matrix is $H = - \sum \hat{\sigma}_i \ln \hat{\sigma}_i = \ln 9 = 2 \ln 3 = 2.1972 = H_{\max}$, which is the maximal value that H can achieve for the 9×9 case. Normalized singular value distributions that are evenly distributed among all the available modes correspond to maximum entropy states.

Example 4.2. The 9×9 matrix with 1s in every entry has the property that its covariance matrix has 9s in each entry. Therefore, the covariance matrix has rank 1 and nullspace dimension 8, which means there are eight normalized singular values that are zero, and one that is exactly 1. The Shannon entropy for this case is 0, which is the minimal value that H can achieve. Singular value distributions that have all the energy clustered into a single mode are minimum entropy states.

From these two examples, one sees that distributions of singular values that drop off rapidly from the peak value correspond to lower entropy matrices than those that are flat. The flattest distribution, where all normalized singular values are equal in height, corresponds to the matrix with maximum entropy.

It is also useful to define the notion of ‘percentage of compression’ a given matrix achieves with respect to the maximal entropy matrix. Hence, given a Sudoku matrix A , define

$$\tilde{H} = \frac{2 \ln 3 - H}{2 \ln 3} * 100. \quad (4.10)$$

From the previous examples, orthogonal matrices of example 4.1 have $\tilde{H} = 0\%$ compression, while minimum entropy matrices of example 4.2 have $\tilde{H} = 100\%$ compression.

Example 4.3. The singular value decomposition of the Sudoku matrix shown in example 2.1 gives rise to matrices U , Σ and V ,

$$U = \begin{pmatrix} -0.3333 & -0.1515 & -0.5041 & 0.4468 & 0.1031 & 0.4343 & -0.3210 & -0.1106 & 0.3124 \\ -0.3333 & 0.2834 & -0.2010 & -0.4892 & 0.4398 & -0.0053 & -0.1160 & 0.5589 & 0.0979 \\ -0.3333 & 0.0258 & 0.6106 & 0.2963 & -0.0829 & 0.4490 & 0.2630 & 0.3862 & 0.0286 \\ -0.3333 & -0.5261 & 0.0066 & -0.1856 & -0.1015 & 0.0614 & -0.3054 & 0.0449 & -0.6843 \\ -0.3333 & -0.3819 & 0.0677 & -0.5251 & -0.2410 & 0.0445 & 0.2604 & -0.2846 & 0.5039 \\ -0.3333 & 0.2816 & -0.4823 & 0.0924 & -0.3406 & -0.0698 & 0.5955 & 0.0101 & -0.3047 \\ -0.3333 & 0.4400 & 0.2478 & -0.0923 & 0.3491 & 0.1334 & -0.0584 & -0.6632 & -0.2057 \\ -0.3333 & -0.3020 & 0.0999 & 0.3728 & 0.4243 & -0.6516 & 0.1906 & 0.0007 & 0.0880 \\ -0.3333 & 0.3306 & 0.1548 & 0.0839 & -0.5503 & -0.3959 & -0.5088 & 0.0576 & 0.1639 \end{pmatrix},$$

$$\Sigma = \begin{pmatrix} 45.0000 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 13.9679 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10.6715 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 9.3370 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 9.0413 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 5.6280 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4.6463 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2.9711 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0000 \end{pmatrix},$$

and $V = \begin{pmatrix} -0.3333 & -0.0802 & -0.6393 & -0.1292 & 0.1009 & 0.2984 & 0.5638 & 0.1610 & -0.1184 \\ -0.3333 & 0.4910 & -0.0342 & 0.1143 & -0.1193 & -0.5735 & 0.0727 & 0.0008 & -0.5340 \\ -0.3333 & 0.2366 & 0.2449 & 0.3496 & -0.5422 & 0.4283 & 0.1939 & -0.3258 & 0.1717 \\ -0.3333 & -0.3341 & -0.1063 & 0.6025 & -0.0547 & -0.0325 & -0.3432 & 0.5289 & 0.0378 \\ -0.3333 & -0.2027 & -0.2590 & -0.4817 & -0.4841 & -0.3246 & -0.2655 & -0.1092 & 0.3558 \\ -0.3333 & 0.3386 & -0.2049 & 0.1510 & 0.5896 & -0.0229 & -0.2530 & -0.3443 & 0.4227 \\ -0.3333 & -0.5105 & 0.1447 & -0.0724 & 0.1777 & 0.1517 & -0.2035 & -0.4884 & -0.5172 \\ -0.3333 & 0.3194 & 0.3173 & -0.4700 & 0.0959 & 0.4021 & -0.2813 & 0.4508 & -0.1100 \\ -0.3333 & -0.2580 & 0.5368 & -0.0641 & 0.2361 & -0.3270 & 0.5161 & 0.1263 & 0.2916 \end{pmatrix}.$

Note that the first term $\sigma_1 \vec{u}_1 \vec{v}_1^T$ in the partial sum representation (4.6) gives rise to the rank-1 matrix (minimal Shannon entropy as in example 4.2)

$$A_1 = \begin{pmatrix} 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 \\ 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 \\ 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 \\ 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 \\ 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 \\ 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 \\ 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 \\ 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 \\ 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 \end{pmatrix}, \quad (4.11)$$

where each entry is the arithmetic average of the allowable digits 1–9 in each column, row, or 3×3 sub-block.² If we normalize the singular values down the diagonal of Σ ,

$$\left. \begin{aligned} \hat{\sigma}_1 &= 0.4444, & \hat{\sigma}_2 &= 0.1379, & \hat{\sigma}_3 &= 0.1054, & \hat{\sigma}_4 &= 0.0922, & \hat{\sigma}_5 &= 0.0893, \\ \hat{\sigma}_6 &= 0.0556, & \hat{\sigma}_7 &= 0.0459, & \hat{\sigma}_8 &= 0.0293, & \hat{\sigma}_9 &= 0.000, \end{aligned} \right\} \quad (4.12)$$

we obtain a value of the Shannon entropy $H = 1.7137$ and a compression factor of 22 per cent.

In the next section, we calculate the Shannon entropy and percentage of compression of an ensemble of Sudoku matrices generated by our algorithm.

(b) Ensemble averages

To generate an ensemble of Sudoku matrices, we run the algorithm described above N times, denoting each realization of a Sudoku matrix $A^{(j)}$. The singular values for the j th realization are denoted $\sigma_{\max}^{(j)} \equiv \sigma_1^{(j)} \geq \sigma_2^{(j)} \geq \dots \geq \sigma_{\min}^{(j)} \equiv \sigma_9^{(j)} \geq 0$ and their corresponding left and right singular vectors are denoted $\vec{u}_i^{(j)}$ and $\vec{v}_i^{(j)}$ ($i = 1, \dots, 9$), respectively. We define the ensemble average of the collection of matrices

$$\langle A \rangle_N = \frac{1}{N} \sum_{j=1}^N A^{(j)} \quad \text{and} \quad \langle A \rangle_\infty = \lim_{N \rightarrow \infty} \langle A \rangle_N \quad (4.13)$$

as well as the ensemble averages of the singular components

$$\langle \sigma_i \rangle_N = \frac{1}{N} \sum_{j=1}^N \sigma_i^{(j)}, \quad \langle \sigma_i \rangle_\infty = \lim_{N \rightarrow \infty} \langle \sigma_i \rangle_N, \quad (4.14)$$

$$\langle \vec{u}_i \rangle_N = \frac{1}{N} \sum_{j=1}^N \vec{u}_i^{(j)}, \quad \langle \vec{u}_i \rangle_\infty = \lim_{N \rightarrow \infty} \langle \vec{u}_i \rangle_N \quad (4.15)$$

and

$$\langle \vec{v}_i \rangle_N = \frac{1}{N} \sum_{j=1}^N \vec{v}_i^{(j)}, \quad \langle \vec{v}_i \rangle_\infty = \lim_{N \rightarrow \infty} \langle \vec{v}_i \rangle_N. \quad (4.16)$$

The standard deviation of each quantity is denoted with double brackets $\langle\langle \cdot \rangle\rangle_N$. If we first normalize the singular values from each member of the ensemble as in equation (4.8), i.e.

$$\hat{\sigma}_i^{(j)} = \frac{\sigma_i^{(j)}}{\sum_{i=1}^9 \sigma_i^{(j)}}, \quad (4.17)$$

²In fact, this will always be the case since the constraints imply that 45 is an eigenvalue of A and A^T , with corresponding eigenvectors $(1, 1, \dots, 1)$ and $(1, 1, \dots, 1)^T$.

and then perform the ensemble averages, we denote the averaged normalized values

$$\langle \hat{\sigma}_i \rangle_N = \frac{1}{N} \sum_{j=1}^N \hat{\sigma}_i^{(j)}, \quad \langle \hat{\sigma}_i \rangle_\infty = \lim_{N \rightarrow \infty} \langle \hat{\sigma}_i \rangle_N, \quad (4.18)$$

with standard deviations $\langle \langle \cdot \rangle \rangle_N$.

The averaged normalized singular values (4.18) are shown in figure 3 for $N = 1, \dots, 10\,000$. Each is seen to converge (the law of large numbers) to a sample mean value denoted by $\langle \hat{\sigma} \rangle_\infty$. The distribution of the nine values is shown in figure 4 (data points), along with error bars (black) showing one standard deviation and the spread of all the data about the mean (grey). The singular values decrease very nearly linearly. Histograms of each of the averaged normalized values are shown in figure 5, along with their Gaussian fits. Table 1 shows the nine sample mean values $\langle \hat{\sigma}_i \rangle_N$, their standard deviations and variances for $N = 10^8$.

We denote the Shannon entropy of the j th member of the ensemble to be

$$H^{(j)} = - \sum_{i=1}^9 \hat{\sigma}_i^{(j)} \ln \hat{\sigma}_i^{(j)} \quad (4.19)$$

with ensemble average

$$\langle H \rangle_N = \frac{1}{N} \sum_{j=1}^N H^{(j)}, \quad \langle H \rangle_\infty = \lim_{N \rightarrow \infty} \langle H \rangle_N, \quad (4.20)$$

and standard deviation $\langle \langle H \rangle \rangle_N$. A histogram of the averaged Shannon entropy is shown in figure 6, along with a Gaussian fit to the data. The ensemble-averaged Shannon entropy has sample mean value $\langle H \rangle_{10^8} = 1.73312$ and standard deviation 0.000173. The smallest entropy achieved in the sample was 1.512975, whereas the largest was 1.881064. A convergence plot of the Shannon entropy is shown in figure 7 for $N = 1, 2, \dots, 10\,000$. If we use this sample mean value in equation (4.10) to calculate the ensemble-averaged percentage of compression of a collection of Sudoku matrices, we arrive at

$$\langle \tilde{H} \rangle = \left(\frac{2 \ln 3 - \langle H \rangle_\infty}{2 \ln 3} \right) * 100 \approx \left(\frac{2 \ln 3 - 1.73312}{2 \ln 3} \right) * 100 = 21\%. \quad (4.21)$$

This value is slightly lower than the corresponding value for a collection of 10^8 9×9 Latin squares, which has Shannon entropy 1.73544 ± 0.0001735 . On the other hand, a collection of random matrices has the much lower entropy value of 1.65128 ± 0.0001656 , giving a 25 per cent compression factor.

Remark 4.4. When comparing two distributions, the *relative entropy* or *Kullback–Leibler divergence* gives a measure of how likely the first distribution will resemble the second (Kullback & Leibler 1951). This value is defined as

$$h(a, p) = \sum_{i=1}^n a_i \ln \frac{a_i}{p_i}, \quad (4.22)$$

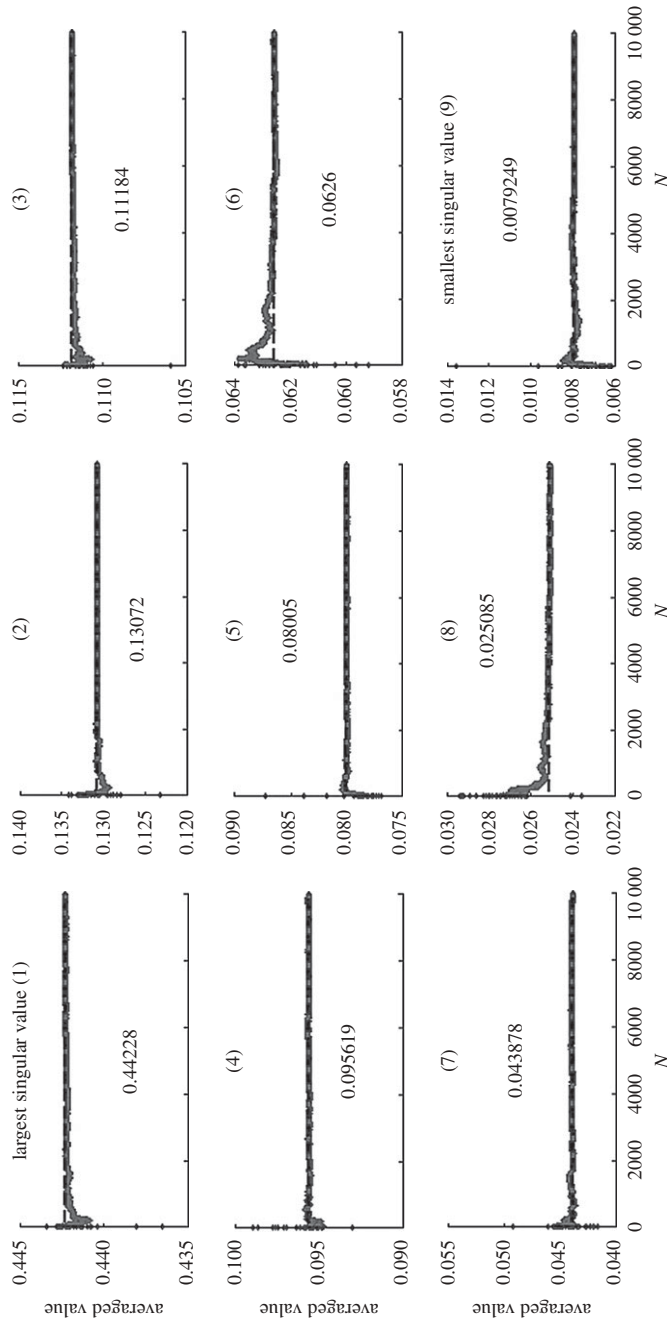


Figure 3. Ensemble average of each of the nine singular values for N Sudoku matrices, where $N = 1, \dots, 10,000$. The horizontal (dashed) line in each plot is $\langle \hat{\sigma}_i \rangle_{10,000}$, with value shown.

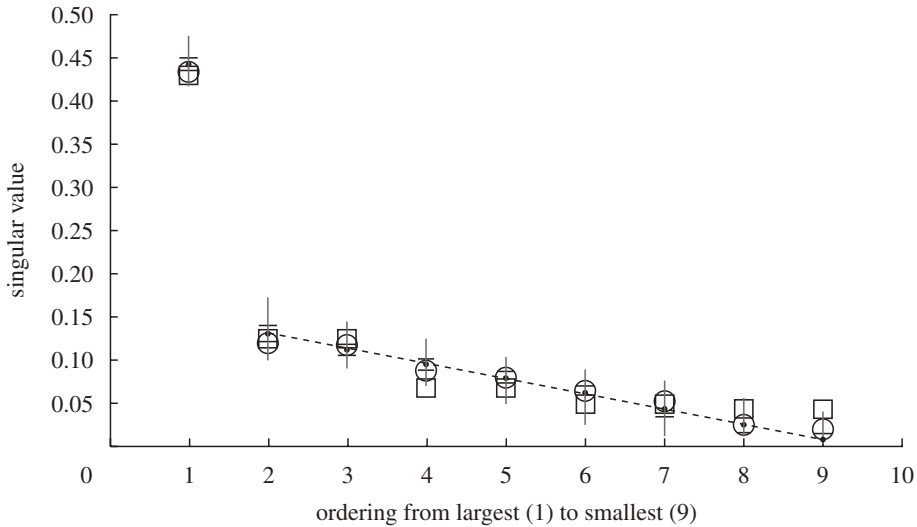


Figure 4. Distribution of the normalized singular values of the ensemble of Sudoku matrices in descending order. The dots denote the sample mean, with error bars showing one standard deviation about the mean and the spread of all the data with a least squares line (dashed) to the first eight singular values (the scale of the x -axis is arbitrary). The computed correlation coefficient is $r^2 = 0.9993$. The unfilled circles denote the singular values of a typical Sudoku matrix selected randomly, while the unfilled squares denote the singular values from equation (2.2).

where $a = (a_1, \dots, a_n)$ and $p = (p_1, \dots, p_n)$ are the two distributions being compared. Note that $h(a, p) \neq h(p, a)$. It has already been established that the entropy for Sudoku matrices and Latin squares is similar, so naturally their relative entropy should be small. Indeed, this is the case, as this value (if a_i corresponds to the Sudoku distributions and p_i corresponds to Latin squares) turns out to be $h(a, p) = 9.74107032 \times 10^{-6} \approx 10^{-5}$. To put this in perspective, the relative entropy between Sudoku matrices and random matrices is $0.004044 \approx 5 \times 10^{-3}$, two orders of magnitude higher, and nearly the same as the relative entropy between Latin squares and random matrices, which is 0.0042812 .

5. Matrix reconstitution

One might ask what the ‘typical’ Sudoku matrix looks like in terms of its decomposition (4.4). For this, we use the ensemble-averaged singular vectors $\langle \vec{u}_i \rangle_N$, $\langle \vec{v}_i \rangle_N$ to construct the matrices $\langle \vec{u}_i \rangle_N \langle \vec{v}_i^T \rangle_N$ along with the averaged singular values $\langle \sigma_i \rangle_N$. The ‘reconstituted’ matrix is then defined to be

$$\langle A_k \rangle_N = \sum_{i=1}^k \langle \sigma_i \rangle_N \langle \vec{u}_i \rangle_N \langle \vec{v}_i^T \rangle_N, \quad (5.1)$$

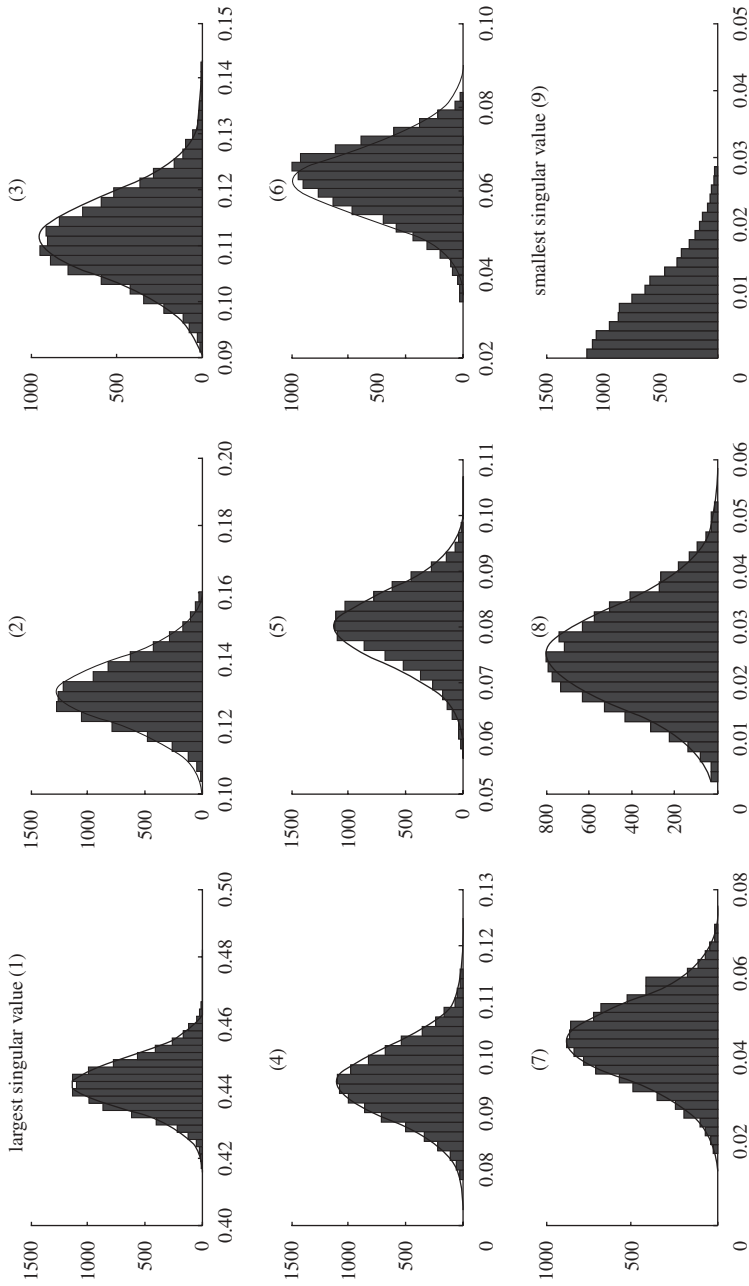


Figure 5. Histograms of the nine singular values for the ensemble of $N = 10,000$ Sudoku matrices. The spread around the mean appears to follow a Gaussian distribution reasonably well. The smallest singular value has an approximate half-Gaussian spread (centred at 0).

Table 1. Minimum value, maximum value, sample mean and standard deviation for the nine ensemble-averaged and normalized singular values based on a sample size of $N = 10^8$.

	$\min_j \hat{\sigma}_i^{(j)}$	$\max_j \hat{\sigma}_i^{(j)}$	$\langle \hat{\sigma}_i \rangle_N$	$\langle \langle \hat{\sigma}_i \rangle \rangle_N$
$\hat{\sigma}_1$	0.41311204510567	0.48811216599382	0.44232319538594	0.00000000195716
$\hat{\sigma}_2$	0.09417944510957	0.19182448301596	0.13072369442204	0.00000000017089
$\hat{\sigma}_3$	0.08503464489795	0.15303595644309	0.11196765935169	0.00000000012601
$\hat{\sigma}_4$	0.06033117834183	0.12811546927484	0.09571062185255	0.00000000009163
$\hat{\sigma}_5$	0.03574065500616	0.10800399513199	0.07987968644928	0.00000000006386
$\hat{\sigma}_6$	0.02072019503703	0.09220340904337	0.06259759902184	0.00000000003946
$\hat{\sigma}_7$	0.00624284569488	0.07760803395316	0.04383380668260	0.00000000001941
$\hat{\sigma}_8$	0.00060920223256	0.06662384524882	0.02506384156452	0.00000000000974
$\hat{\sigma}_9$	0	0.04757243788525	0.00789989526931	0.00000000000113

which yields an optimal rank- k ensemble-averaged Sudoku matrix. The matrices with $N = 10^8$ for $k = 3, 6$ and 9 are as follows:

$$A_3 = \begin{pmatrix} 4.9975 & 4.9996 & 5.0014 & 4.9963 & 5.0011 & 5.0010 & 5.0009 & 5.0012 & 5.0009 \\ 5.0038 & 4.9980 & 4.9988 & 5.0007 & 5.0022 & 4.9997 & 5.0002 & 4.9981 & 4.9985 \\ 5.0013 & 4.9997 & 4.9995 & 5.0009 & 5.0002 & 4.9998 & 4.9999 & 4.9994 & 4.9995 \\ 5.0014 & 4.9997 & 4.9994 & 5.0010 & 5.0002 & 4.9997 & 4.9998 & 4.9993 & 4.9995 \\ 5.0009 & 4.9988 & 5.0000 & 4.9988 & 5.0016 & 5.0003 & 5.0005 & 4.9995 & 4.9996 \\ 5.0005 & 5.0012 & 4.9993 & 5.0029 & 4.9981 & 4.9993 & 4.9991 & 4.9998 & 4.9999 \\ 4.9986 & 5.0018 & 5.0000 & 5.0018 & 4.9975 & 4.9996 & 4.9992 & 5.0008 & 5.0006 \\ 4.9978 & 5.0011 & 5.0007 & 4.9995 & 4.9988 & 5.0002 & 4.9999 & 5.0011 & 5.0009 \\ 4.9983 & 5.0000 & 5.0009 & 4.9981 & 5.0003 & 5.0005 & 5.0004 & 5.0008 & 5.0006 \end{pmatrix}, \quad (5.2)$$

$$A_6 = \begin{pmatrix} 4.9981 & 4.9974 & 5.0018 & 4.9965 & 4.9998 & 4.9999 & 5.0021 & 5.0020 & 5.0024 \\ 5.0029 & 4.9974 & 4.9997 & 5.0016 & 5.0023 & 5.0001 & 5.0006 & 4.9971 & 4.9983 \\ 5.0016 & 5.0001 & 5.0004 & 5.0006 & 5.0003 & 4.9990 & 4.9995 & 4.9986 & 4.9999 \\ 5.0031 & 4.9997 & 5.0003 & 4.9997 & 4.9993 & 4.9975 & 4.9995 & 4.9996 & 5.0012 \\ 5.0011 & 4.9998 & 4.9969 & 4.9982 & 5.0019 & 5.0017 & 5.0001 & 5.0017 & 4.9984 \\ 5.0001 & 5.0009 & 4.9985 & 5.0033 & 4.9982 & 5.0001 & 4.9994 & 5.0003 & 4.9994 \\ 4.9982 & 5.0003 & 4.9996 & 5.0025 & 4.9970 & 5.0000 & 5.0002 & 5.0014 & 5.0008 \\ 4.9984 & 5.0031 & 5.0012 & 4.9985 & 4.9994 & 4.9996 & 4.9986 & 5.0004 & 5.0007 \\ 4.9966 & 5.0012 & 5.0015 & 4.9991 & 5.0018 & 5.0021 & 4.9999 & 4.9989 & 4.9989 \end{pmatrix} \quad (5.3)$$

$$\text{and } A_9 = \begin{pmatrix} 4.9982 & 4.9975 & 5.0020 & 4.9964 & 4.9995 & 5.0001 & 5.0022 & 5.0019 & 5.0022 \\ 5.0031 & 4.9976 & 5.0001 & 5.0014 & 5.0019 & 4.9999 & 5.0013 & 4.9973 & 4.9973 \\ 5.0017 & 4.9998 & 5.0001 & 5.0006 & 5.0003 & 4.9999 & 4.9985 & 4.9982 & 5.0010 \\ 5.0027 & 4.9997 & 5.0000 & 5.0000 & 5.0000 & 4.9970 & 4.9994 & 4.9998 & 5.0015 \\ 5.0011 & 4.9997 & 4.9969 & 4.9982 & 5.0020 & 5.0017 & 5.0000 & 5.0017 & 4.9986 \\ 5.0002 & 5.0010 & 4.9987 & 5.0031 & 4.9978 & 5.0002 & 4.9997 & 5.0002 & 4.9991 \\ 4.9980 & 5.0002 & 4.9995 & 5.0027 & 4.9974 & 4.9998 & 4.9999 & 5.0015 & 5.0010 \\ 4.9987 & 5.0034 & 5.0017 & 4.9983 & 4.9988 & 4.9995 & 4.9995 & 5.0005 & 4.9996 \\ 4.9962 & 5.0011 & 5.0011 & 4.9993 & 5.0024 & 5.0019 & 4.9995 & 4.9989 & 4.9996 \end{pmatrix}. \quad (5.4)$$

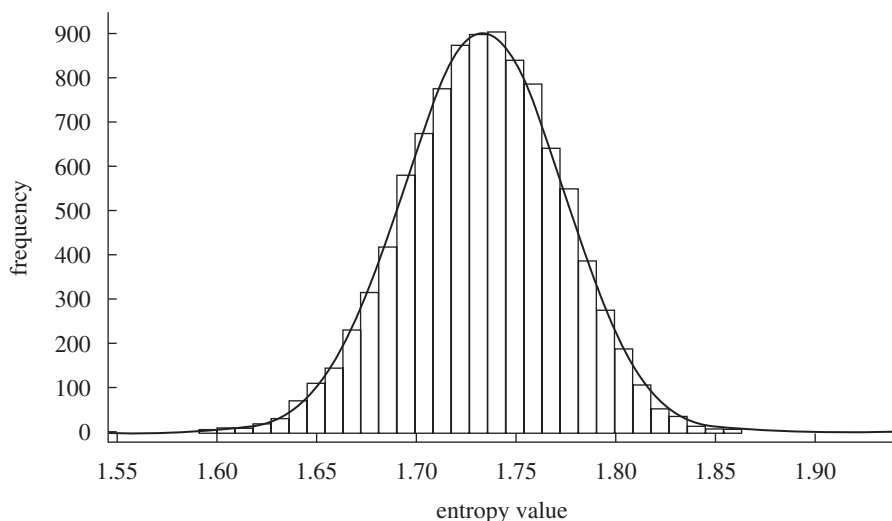


Figure 6. Ensemble-averaged Shannon entropy with $N = 10\,000$, with sample mean value 1.73221822064708 and standard deviation 0.03996073536463. A histogram is shown along with the Gaussian distribution.

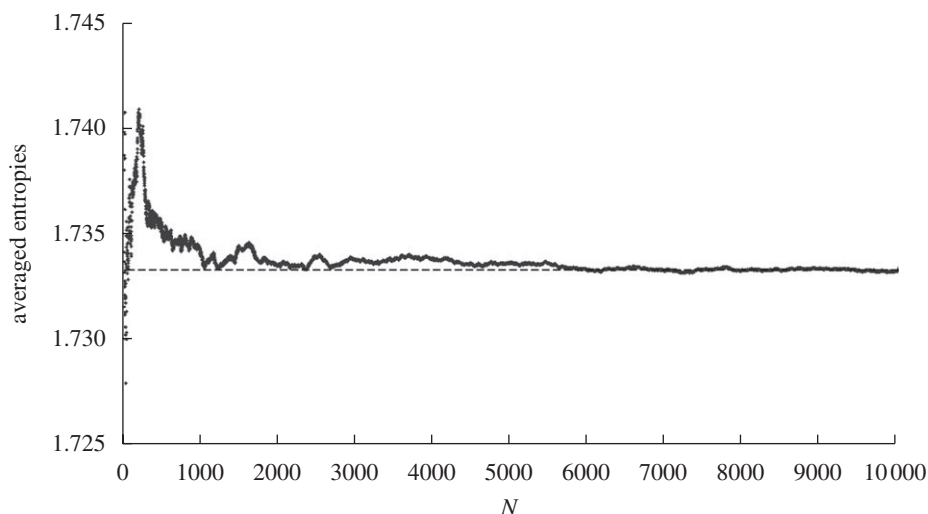


Figure 7. The Shannon entropy of the ensemble of $N = 1, \dots, 10\,000$ Sudoku matrices. The horizontal line (dashed) shows the approximate converged sample average.

Given the discussion at the end of §3 with respect to the extent to which our sample is unbiased, the convergence to the rank-1 matrix (4.11) is expected. By subtracting the matrix (4.11) from A_k , $k = 3, 6, 9$, the singular value spectrum should go to zero. We show this in figure 8. Each shows one dominant non-zero singular value with the eight others close to zero.

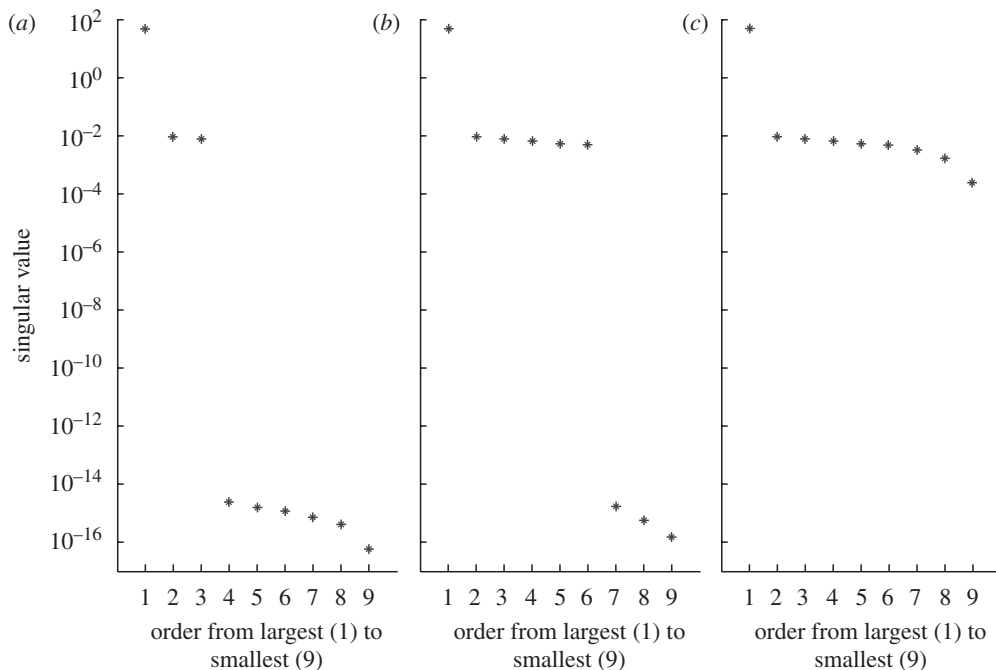


Figure 8. Singular values for the rank- $k=3, 6, 9$ matrices of the ensemble average. Each shows a spectrum similar to the rank-1 matrix of all 5's. (a) Singular values for the rank-3 matrix, (b) singular values for the rank-6 matrix and (c) singular values for the rank-9 matrix.

6. Summary

The normalized distribution of singular values and corresponding Shannon entropy for an ensemble of Sudoku matrices provides a quantitative measure of the level of ‘disorder’ inherent in the collection by characterizing the weighting of the nine rank-1 ‘modes’ $\vec{u}_i \vec{v}_i^T$ used in reconstituting the ‘average’ Sudoku matrix and allows for comparisons with the related ensembles of Latin squares and random matrices obtained by dropping the regional constraint 3, and all three constraints, respectively. Our conclusions based on these comparisons are that the ensemble-averaged Shannon entropy of the collection of Sudoku matrices is slightly lower than a collection of Latin squares, but higher than a collection of appropriately chosen random matrices. Thus, the extra constraints imposed on the Latin squares and Sudoku matrices serve to *increase* their Shannon entropy as compared with what it would be without the constraints imposed (random matrices), forcing a more even distribution among the singular modes. This is consistent with the fact that the set of random matrices includes matrices of all possible rank, including matrices with rank 1. Low-rank matrices have lower Shannon entropy than high-rank ones, pulling the average of the ensemble down. Why the more constrained Sudoku matrices have a lower average Shannon entropy than the less constrained ensemble of Latin squares is not obvious, nor is it clear that the differences in these averages are statistically significant, or that our methods are sufficiently sensitive to distinguish between such subtle differences.

P.K.N. would like to acknowledge financial support through NSF-DMS0504308. S.A.D. would like to thank Chris DeSalvo for help with the development of the Sudoku algorithm.

References

- Bammel, S. F. & Rothstein, J. 1975 The number of 9×9 Latin squares. *Discrete Math.* **11**, 93–95. (doi:10.1016/0012-365X(75)90108-9)
- Dahl, G. 2009 Permutation matrices related to Sudoku. *Linear Alg. Appl.* **430**, 2457–2463. (doi:10.1016/j.laa.2008.12.023)
- Davis, T. 2009 The mathematics of Sudoku. See <http://www.geometer.org/mathcircles/sudoku.pdf>.
- Felgenhauer, B. & Jarvis, F. 2006 Mathematics of Sudoku I. See <http://www.afjarvis.staff.shef.ac.uk/sudoku/>.
- Grimmett, G. R. & Stirzaker, D. R. 2005 *Probability and random processes*. Oxford, UK: Oxford University Press.
- Jacobson, M. T. & Matthews, P. 1996 Generating uniformly distributed random Latin squares. *J. Combin. Des.* **4**, 405–437. (doi:10.1002/(SICI)1520-6610(1996)4:6<;405::AID-JCD3>;3.0.CO;2-J)
- Kirby, M. 2001 *Geometric data analysis: an empirical approach to dimensionality reduction and the study of patterns*. New York, NY: Wiley.
- Kullback, S. & Leibler, R. A. 1951 On information and sufficiency. *Ann. Math. Statist.* **22**, 79–86.
- Moon, T. K., Gunther, J. H. & Kupin, J. J. 2009 Sinkhorn solves Sudoku. *IEEE Trans. Inform. Theory* **55**, 1741–1746. (doi:10.1109/TIT.2009.2013004)
- Nagy, F. 2006 Parameter estimation of the Cauchy distribution in information theoretic approach. *J. Universal Comput. Sci.* **12**, 1332–1344.
- Russell, E. & Jarvis, F. 2006 Mathematics of Sudoku II. See <http://www.afjarvis.staff.shef.ac.uk/sudoku/>.
- Sengupta, A. M. & Mitra, P. P. 1999 Distributions of singular values for some random matrices. *Phys. Rev. E* **60**, 3389–3392. (doi:10.1016/0012-365X(75)90108-9)
- Shannon, C. E. 1948 A mathematical theory of communication. *Bell Syst. Tech. J.* **27**, 379–423.
- Simonis, H. 2005 Sudoku as a constraint problem. In *Proc. 4th Int. Workshop Modelling and Reformulating Constraint Satisfaction Problems, Barcelona, Spain, 1 October 2005* (eds B. Hnich, P. Prosser & B. Smith), pp. 13–27.
- Trefethen, L. N. & Embree, M. 2005 *Spectra and pseudospectra: the behavior of nonnormal matrices and operators*, ch. 8. Princeton, NJ: Princeton University Press.
- Yato, T. & Seta, T. 2005 Complexity and completeness of finding another solution and its application to puzzles. Preprint, University of Tokyo, Japan.