KINGS, SAGES, HATS, AND CODES

STEVEN DOUGHERTY
UNIVERSITY OF SCRANTON
SCRANTON, PA, USA
AND
YUSRA NAQVI
AMHERST COLLEGE
AMHERST, MA, USA

In the Kingdom of Noether, a capricious and mathematically inclined king ruled the land. He was known for posing difficult mathematical challenges to those over whom he ruled. He used these challenges to help make royal decisions. We shall describe two such occurrences together with the clever solutions of those who were able to defeat the king in his challenge. We then show how these challenging problems are related to modern information theory.

Hiring Royal Assistants

The king wishes to hire two sages to assist him in his royal duties and intends to pay them generously for their services. However, to ensure that they are not frauds, he wants to test their abilities. If they prove themselves worthy, they will be given a place in court. If not, they shall be considered impostors and thrown in jail. The test that the king proposes is as follows.

Question 0.1. *The sages will be put in two separate waiting rooms. The king will lay out an 8 by 8 chess board in the throne room and place some number of stones on different squares of the chessboard. There can be any number of stones from 0 to 64. Then the first sage will be brought in, and the king will point to a totally arbitrary square on the board (the square can either have a stone or not have a stone). The first sage then has the choice of one of the following three options:*

1. *place one stone on an unoccupied square on the board,*
2. *remove one stone that is already on the board,*
3. *do nothing.*

The first sage will then be taken back to the first waiting room, and the second sage will be brought in to the throne room. This second sage must now look at the board and determine to which square the king has pointed. Of course, the two sages are informed of this test when they apply for the job, and they can discuss a strategy before they are taken to their respective waiting rooms, but they may not communicate at all after that. What strategy do they employ to determine to which square the king has pointed?

Prisoners

It should not be surprising that such a temperamental and demanding king has had several occasions to imprison people on various insignificant charges. However, continuing his tradition of enjoyment of mathematical challenges, he would often allow them a challenge to free themselves. We shall describe one such challenge.

Question 0.2. *A group of 7 prisoners will be taken into separate cells. They will each be given either a red hat or a yellow hat. The hats are chosen from a sack containing thousands, and it is safe to assume that each prisoner has equal likelihood of getting either color of hat. They will then all be brought into the same room. No one can see the color of their own hat but they can, of course, see the color of the other hats. When prompted, they must immediately choose from one of the following two options:*

(1) *guess the color of their own hat,*
(2) *pass on the question.*

Notice that all of the prisoners give their answer at the same time, either by guessing a color or by signaling that they pass. If someone guesses wrongly or all of them pass, they will all have their prison sentences extended, but if any one prisoner guesses correctly, they will all be freed. They are allowed to communicate ahead of time but they are not allowed to communicate in any way after the hats are handed out, nor are they allowed to delay in responding. What should their strategy be so that there answer has a better chance of succeeding than simply guessing the color of a random hat with probability $\frac{1}{2}$.

1. Hints

Before providing solutions to these problems, we would like to make a few observations about these two problems.

Observation 1. Both challenges involve a number of objects (chessboards squares or hats) that take on exactly two modes. The chessboard squares can either be *covered* or *uncovered.* The hats can be either *red* or *yellow.* Therefore, all possible configurations of these objects can be represented in binary, as a sequence of '1's and '0's.

Observation 2. The second sage in Question 0.1 does not need to determine which stone, if any, was changed by first sage, only to which square the king pointed. The second sage answers based entirely on the configuration of stones at hand.

Observation 3. Since the prisoners in Question 0.2 do not know in which order they will be called, they do not need to use previous passes, if any, as part of their strategy. They each determine how to answer based on the configuration of hats that they see.

The King of Noether would give those who were questioned one hour to think about a solution before being put to the test. The interested reader may wish to pause at this point to come to a solution on their own!

2. The Chessboard Question

At first glance the question posed to the sages seems to be quite difficult. If they were allowed to move more than one stone it would seem that they could arrange them in a manner to indicate the row and the column, but being able to change only one location seems to make the problem very difficult.

Let's examine a small case first and then examine the general solution. Consider a 2×2 chessboard. Label each square with a different number from 0 to 3.

0	1
2	3

The king places stones on the board. Construct a vector $\mathbf{v}$ of length 4 with coordinates labeled 0, 1, 2, 3 as follows. If he has placed a stone on a square, then the coordinate corresponding to that square has a 1 in it, and all other coordinates are 0. For convenience, we think of this vector as a column vector, that is as

$$\mathbf{v} = \begin{pmatrix} v_0 \\ v_1 \\ v_2 \\ v_3 \end{pmatrix}.$$

Thus, if the king places stones on squares 0 and 3:

*	
	*

then our vector $\mathbf{v}$ is

$$\mathbf{v} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}.$$

Next, construct a matrix M_4 by writing each possible length 2 vector consisting of zeroes and ones as a column. Thus:

$$M_4 = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{pmatrix}$$

Then $\mathbf{w} = M_4\mathbf{v}$ is a 2 coordinate vector. (Note all computations are done over the finite field $\mathbb{F}_2$, that is everything is done modulo 2. In other words we have only two elements, namely 0 and 1 and arithmetic works as usual for multiplication and addition except that $1 + 1 = 0$.) In our example, this vector would be:

$$\mathbf{w} = M_4\mathbf{v} = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

This vector $\mathbf{w}$ is called the syndrome. One might notice that many different vectors $\mathbf{v}$ give the same syndrome including the vector that has a 0 in every coordinate except the coordinate that corresponds to the column in M_4 that is the syndrome. In this example that would be the vector

$$\begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}.$$

Each number from 0 to 3 can be represented as a two digit binary number $a_1 a_0$, so given a vector $\left(\begin{smallmatrix} a_1 \\ a_0 \end{smallmatrix}\right)$, it corresponds to the number $2a_1 + a_0$. Therefore, in our example, $\mathbf{w} = \left(\begin{smallmatrix} 1 \\ 1 \end{smallmatrix}\right)$ would correspond to the number 3. Notice that the number 3 is in the coordinate of the array corresponding to where we put the 1 in the vector corresponding to the syndrome.

Now, the king may point to any one of the squares on this board. Thus, our strategy is to alter the vector $\mathbf{v}$ in one coordinate to form a new vector $\mathbf{v}'$ so that $\mathbf{w}' = M_4\mathbf{v}'$ gives the two coordinate vector that represents the number of the square to which the king pointed.

If $\mathbf{w}$ is the two coordinate vector given by the original configuration of stones, and $\mathbf{w}'$ is the actual vector we wish to convey to the second sage, we need to consider the vector $\mathbf{w} + \mathbf{w}'$. (Again, this computation is done modulo 2.) The result is a length 2 vector which must appear as one of the 4 columns in the matrix M_4. Construct a length 4 column vector $\mathbf{u}$ which only has a 1 in the coordinate corresponding to that column of M_4 (and zeroes everywhere else). Then

$$M_4(\mathbf{v} + \mathbf{u}) = \mathbf{w} + (\mathbf{w} + \mathbf{w}') = \mathbf{w}'.$$

Hence, by changing the square corresponding to this nonzero coordinate of $\mathbf{u}$, we effect the desired change so that the second sage can determine the answer. More specifically, if there is a stone already on the square indicated by $\mathbf{u}$, the sage removes it, and if there is no stone there, then the sage adds one. In other words, the sage adds a 1 (modulo 2) to that coordinate. If $\mathbf{w} = \mathbf{w}'$ then the sage does nothing.

Let's return to our example, and suppose that the king points to square 2. Then $\mathbf{w}' = \left(\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right)$. Since $\mathbf{w}$ in our example is $\left(\begin{smallmatrix} 1 \\ 1 \end{smallmatrix}\right)$, the vector $\mathbf{w}+\mathbf{w}'$ (modulo 2) is $\left(\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}\right)$. This vector is the second column of of M_4, and therefore

$$\mathbf{u} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix},$$

and so the first sage must add a new stone to the square labelled 1 since this square was empty before.

*	*
	*

The second sage, upon entering the room, will see this new configuration

$$\mathbf{v}' = \mathbf{v} + \mathbf{u} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \end{pmatrix},$$

and compute $M_4\mathbf{v}' = \left(\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right)$, and will correctly declare that the king pointed to the square labelled 2.

The question however, was for an 8×8 chessboard. Let's see to which boards we can extend this solution. We need to construct a matrix with all possible binary columns of length k. Of course there are 2^k such vectors. So for an $n \times n$ chessboard we need $n^2 = 2^k$, which requires n to also be a power of 2.

The general algorithm for solving this problem can be described as follows.

(1) Label the squares from 0 to $2^k - 1$.
(2) Construct M_{2^k} consisting of all possible binary columns of length k.
(3) Construct $\mathbf{v}$ where $\mathbf{v}_i = 1$ if the square corresponding to i has a stone on it and $\mathbf{v}_i = 0$ if it does not.
(4) Compute $\mathbf{w} = M_{2^k}\mathbf{v}$ and $\mathbf{w}'$ as the vector corresponding to the point to which the king pointed.
(5) Change the value in the square corresponding to the column $\mathbf{w} + \mathbf{w}'$ by adding or removing a stone.
(6) Compute $M_{2^k}(\mathbf{w} + \mathbf{w}')$.
(7) Receive employment, and accompanying riches, from the king of Noether.

3. The Hat Question

For this question, we once again start by examining a smaller case. Consider the same problem, but with only 3 prisoners. We shall denote a red hat as R and a yellow hat as Y.

The possibilities for 3 people wearing these hats are as follows:

RRR
RRY
RYR
RYY

YYY
YYR
YRY
YRR

The standard solution is as follows. The probability that all three hats have the same color is $\frac{2}{8} = \frac{1}{4}$. Hence, it is much more probable that they are not all the same color. So here is how you proceed. If you look at the other two people and they both have the same color, then you say that you have the opposite color. If you see the other two people have different colors, then you pass.

With this strategy you will win with probability $\frac{3}{4}$, which is better than just guessing, in which case you win with probability $\frac{1}{2}$.

In order to generalize this solution, we must first examine this in a more mathematical setting. Let R be associated with the number 0 and Y be associated with the number 1 and look at the following matrix:

$$H_3 = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}.$$

(Note that H_3 is composed of all possible nonzero binary vectors of length 2 as its columns.) The set C_3 of vectors $\mathbf{v}$ such that $H_3\mathbf{v}^T = \mathbf{0}$ is $\{(000), (111)\}$. These vectors precisely correspond to the case of all red hats or all yellow hats. However, it is much more likely that the correct configuration lies outside the set C_3, rather than inside it, and so our strategy was based on this assumption.

Now consider the case for 7 people. In this case, we must look at the matrix

$$H_7 = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}.$$

Note that H_7 consists of all nonzero binary columns of length 3 (of which there are precisely 7).

Let C_7 be the set of all binary vectors $\mathbf{v}$ such that $H_7\mathbf{v}^T = \mathbf{0}$. Since H_7 consists of 3 independent rows, we know that C_7 must have dimension 4, and therefore C_7 contains the following 2^4 vectors:

0000000
1110000
1001100
0101010
0010110
1000011
0100101
0011001

1111111
0001111
0110011
1010101
1101001
0111100
1011010
1100110

Therefore, the probability that the configuation of hats is in C_7 is $\frac{2^4}{2^7} = \frac{1}{8}$, and so once again, it is much more probable that the actual configuration lies outside this set. This assumption allows us to design a strategy that will lead to freedom $\frac{7}{8}$ of the time.

In order to proceed with this strategy, we must first order the prisoners by assigning each one a unique number from 1 to 7. Then, we construct $\mathbf{v}$ by setting its i^{th} coordinate to 0 if prisoner i has a red hat and 1 if yellow. For example, if the assignment of hats is:

Number:	1	2	3	4	5	6	7
Hat :	Y	R	Y	R	R	R	Y

then $\mathbf{v} = (1, 0, 1, 0, 0, 0, 1)$.

Now as long as $\mathbf{v}$ is not in C_7, the vector $\mathbf{w} = H_7\mathbf{v}^T$ (computed again in the binary field) must be nonzero, and so in fact, it must appear as one of the columns of H_7. We notice, that this vector $\mathbf{w}$ is the syndrome just like in the previous problem. In the case of our example,

$$H_7\mathbf{v}^T = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}$$

which is the same as column 5 of H_7. This column also corresponds to the binary representation of 5, which is 101.

If $\mathbf{w}$ matches column i of H_7, construct a length 7 vector $\mathbf{u}$ which only has a 1 in the coordinate i, and zeroes everywhere else. Then $H_7\mathbf{u}^T$ also equals $\mathbf{w}$, and so, modulo 2, we have:

$$H_7(\mathbf{v}+\mathbf{u})^T = \mathbf{w}+\mathbf{w} = \mathbf{0}.$$

This implies that if prisoner i was to have the opposite hat to the one currently assigned, the vector $\mathbf{v}$ would be transformed to a new vector $\mathbf{v}' = \mathbf{v}+\mathbf{u}$ that lies inside C_3. In our example, if prisoner 5 had a yellow hat instead of a red one, the resulting vector $\mathbf{v}'$ would be $(1,0,1,0,1,0,1)$, which is indeed in C_7.

Therefore, the prisoners use the following strategy. Each prisoner guesses $\mathbf{v}$ by looking at the hats of the other 6 prisoners and assuming that their own hat is red (i.e., they each use 0 for their own coordinate). They all compute $\mathbf{w} = H_7\mathbf{v}^T$ based on their personal guess for $\mathbf{v}$. Then, as long as the actual $\mathbf{v}$ is not in C_7, there is a unique prisoner who will get either $\mathbf{0}$ or the column in H_7 corresponding to their own number. If this prisoner gets $\mathbf{0}$, then the correct guess is "yellow"; otherwise it is "red". Everyone else will get a column of H_7 that is different from the one corresponding to their number, and should pass on their turn.

In our example, suppose that prisoner number 3 is asked to guess first. Prisoner 3 would guess $\mathbf{v}$ to be $(1,0,0,0,0,0,1)$, using a 0 for coordinate 3 based on the assumption of having received a red hat. Then $H_7\mathbf{v}^T$ would be

$$\begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix},$$

which is nonzero, and also not column 3 of H_7. So prisoner 3 passes. Similar computations would show that all other prisoners besides prisoner 5 would also pass. However, prisoner 5 computes $\mathbf{v} = (1,0,1,0,0,0,1)$ and sees that $H_7\mathbf{v}$ is in fact column 5 of H_7 in this case. Therefore, this prisoner would correctly guess "red."

We notice at this point that the matrices H_3 and H_7 are the same as M_4 and M_8 of the last section, with the all-zero columns removed. These solutions are thus easily generalized to H_{2^k-1}, which deals with the case of $2^k - 1$ prisoners.

The general algorithm for solving this problem can be described as follows.

(1) Label the prisoners from 1 to $2^k - 1$.

(2) Construct H_{2^k-1} consisting of all possible non-zero binary columns of length k.

(3) Compute the set of all vectors $\mathbf{u}$ such that $H_{2^k-1}\mathbf{u} = \mathbf{0}$ and call this set C_{2^k-1}. The matrix H_{2^k-1} has k linearly independent rows and so by the Rank-Nullity Theorem, the dimension of C is $2^k - 1 - k$.

(4) Construct $\mathbf{v}$ where $\mathbf{v}_i = 1$ if the person corresponding to i has a yellow hat and $\mathbf{v}_i = 0$ if that person has a red hat, and using a 0 for your own coordinate.

(5) If $H_{2^k-1}\mathbf{v}$ is $\mathbf{0}$, guess "yellow." If $H_{2^k-1}\mathbf{v}$ is the same as the column of H_{2^k-1} corresponding to your prisoner number, guess "red." If it is neither of these, pass.

(6) Enjoy your freedom with probability $\frac{2^k-1}{2^k}$.

4. Codes

It may be clear to some readers at this point that the King of Noether has an intimate knowledge of coding theory, that branch of mathematics developed in the second half of the twentieth century to correct errors that occur in electronic communication.

The king proposed two seeming different problems, which really have solutions that come from the same exact structure: specific matrices and the set of vectors that multiply by those matrices to get the zero vector. This set of vectors is actually a well known code called the Hamming code.

We can show the coding theory which the king obviously knew when proposing such problems. Given a fixed length r, there are $\ell = 2^r - 1$ non-zero binary vectors of length r. Let H_r be the $r \times \ell$ matrix such that the columns of H_r consist of all possible non-zero binary vectors. This matrix H_r is known as the parity check matrix. The set of binary vectors $\mathbf{v}$ such that $H_r\mathbf{v}^T = \mathbf{0}$ forms a binary vector space C_r in $\mathbb{F}_2^\ell$. This set of vectors is known as a code. The matrix H_r has r linearly independent rows, and so by the well-known Rank-Nullity Theorem, the dimension of C_r over $\mathbb{F}_2$ is $\ell - r$. This vector space C_r is known as the *Hamming code.*

The *Hamming distance* between two binary vectors is the number of coordinates by which they differ. For instance, the vectors 1001100 and 0101010 in $\mathbb{F}_2^7$ have a distance of 4, since they differ in the 4 positions indicated below:

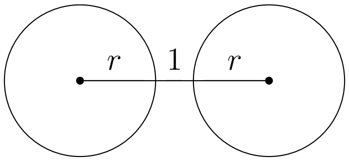
$$\boxed{1}\boxed{0}\ 0\ 1\ \boxed{1}\boxed{0}\ 0$$

$$\boxed{0}\boxed{1}\ 0\ 1\ \boxed{0}\boxed{1}\ 0$$

If the vector has length n, then there are $\binom{n}{s}$ vectors in $\mathbb{F}_2^n$ that are distance s from a given vector, each obtained by flipping s of the n coordinates in the vector. For example, since the vector 010 has length 3, there must be $\binom{3}{2} = 3$ vectors at a distance of 2 from it, obtained by changing 2 of the digits in each case.

$$\begin{aligned} \boxed{0}\boxed{1}\,0 &\longrightarrow \boxed{1}\boxed{0}\,0 \\ \boxed{0}\,1\,\boxed{0} &\longrightarrow \boxed{1}\,1\,\boxed{1} \\ 0\,\boxed{1}\boxed{0} &\longrightarrow 0\,\boxed{0}\boxed{1} \end{aligned}$$

Around any vector, we define a sphere of a given radius r to be the set of all vectors that are distance r or less from the vector. So in a sphere of radius r, there are $\left(\binom{n}{0} + \binom{n}{1} + \binom{n}{2} + \cdots + \binom{n}{r}\right)$ vectors. If the minimum distance between vectors in a code (a set of vectors) is $2r + 1$, then the spheres around the vectors in the code are disjoint.



The minimum distance between any two vectors in C_r is 3, since no two columns of H_r are linearly dependent but some three are. This tells us that any sphere of radius 1 must be disjoint from any other.

As we have seen above, each vector in the code C_r has length $\ell = 2^r - 1$, and C_r has dimension $\ell - r$ over $\mathbb{F}_2$. It follows that there are $2^{\ell - r}$ different vectors in C_r. This means that there must be $2^{\ell - r}$ disjoint spheres of radius 1 in $\mathbb{F}_2^n$, one for each vector in C_r. Since each sphere of radius 1 contains $\ell + 1$ vectors (i.e., the center vector that lies in C_r along with the ℓ additional vectors that are obtained by changing exactly one coordinate), we see that in total, these spheres contain

$$2^{\ell - r} \cdot (\ell + 1) = 2^{\ell - r} \cdot 2^r = 2^\ell$$

vectors, which is exactly the total number of vectors in $\mathbb{F}_2^\ell$. Codes for which this occurs are known as *perfect codes.*

The construction of this perfect code is precisely the amazing fact which allows us to solve both puzzles! It implies that in the ambient space, any vector is either in the code (the set of vectors) or distance 1 from a unique vector in the code. So to solve the first puzzle you are simply changing one position on the grid to get to the vector that represents the spot where the king points. In the second puzzle, each scenario either represents an element in the code or something distance one from the code. We assume then with ever greater probability that the element is not in the code and we have our solution to the problem.

In terms of electronic communication, a similar technique helps us detect errors that occur during transmission. Suppose we receive a vector $\mathbf{v}$ of length $2^r - 1$ as a possible message. Then we can compute the *syndrome* $\mathbf{w} = H_r\mathbf{v}$. If the syndrome is the zero vector, then we assume that the vector $\mathbf{v}$ was correctly transmitted. If the syndrome is not the zero vector, then $\mathbf{w}$ is the same syndrome as for a vector $\mathbf{u}$ with only a single 1 in it, by the above argument. We assume that the error in transmission occurred in the coordinate where the 1 is in **u**, and this coordinate is corrected by adding **u** to **v**. The resulting vector $\mathbf{v} + \mathbf{u}$ has a syndrome of $\vec{0}$, and so we take this to be the intended message.

Notice that in both problems posed by the King of Noether, the key to the solution is simply finding the coordinate where the *error* is. The same type of technique also allows for nearly error free transmission of information in diverse areas like space probes, cable television, phones, compact disc players, or internet communication. So whether you want to free yourself from the dungeon of the King of Noether, win a position in his court, or send a picture from space back to earth, the strategy is really the same!

5. Further Reading

For those interested in reading Hamming's original paper, you can read his early work from 1950 in [2]. For an undergraduate text describing algebraic coding theory see Hill's classic text [3] and for more advanced treatment of the subject see the text by Huffman and Pless which has become the standard in classical coding theory [4]. For a very broad approach to coding theory as pure mathematics, see the recent text by one of the authors of this work [1].

References

[1] S.T. Dougherty, Algebraic coding theory over finite commutative rings. Springer-Briefs in Mathematics. Springer, Cham, 2017.

[2] R.W. Hamming, Error detecting and error correcting codes, Bell Syst. Tech. J., 29, 147 - 160, (1950).

[3] R. Hill, R., A first course in coding theory, Oxford University Press, 1990.

[4] W.C. Huffman, V.S. Pless, Fundamentals of error-correcting codes, Cambridge: Cambridge University Press, (2003).