Latin Squares

Maciej Stec

July 2023

Contents

1 Introduction

Matrices are a well-established part of mathematics with plenty of research being done on them as well as their properties. The aim of this project is to focus on a certain type of matrix known as a Latin Square. Over the course of this summer project, I investigated the concept of Latin Squares, some of their properties such as orthogonality and connection to finite fields as well as some of their real-world applications.

A lot of extensive research has already been done on Latin Squares, and while an interesting topic in my opinion, it is easy to go down many rabbit holes, looking for certain methods or patterns that may or may not necessarily been there. Even famous mathematical minds such as Euler spent a lot of time studying Latin Squares and still coming to the wrong conclusions. Due to the exhaustive nature of studying this topic, the goal for this project wasn't to come up with some new breakthroughs, but rather to explore what is already known and see what kind patterns and connections can be found.

2 What is a Latin Square?

2.1 Definition

Let's take a step back and take a look at what defines a Latin Square. A Latin Square of order n is a $n \times n$ matrix, where $n \in \mathbb{N}$ such that no symbol appears more than once in each row or column. That is, each row and column of a Latin Square has distinct entries. Normally, Latin Squares will either consist of numbers from 1 to n, or numbers from 0 to $n-1$, but letters are also often used, especially in LS design. A normalised Latin Square is one where the first entries of each row and column increase in steps of 1 and are arranged in increasing order.

Here are some valid examples of Latin Squares of order $n = 4$:

$$(a)\ \begin{pmatrix} 1 & 2 & 3 & 4 \\ 4 & 3 & 2 & 1 \\ 2 & 4 & 1 & 3 \\ 3 & 1 & 4 & 2 \end{pmatrix} (b)\ \begin{pmatrix} 0 & 1 & 2 & 3 \\ 2 & 0 & 3 & 1 \\ 3 & 2 & 1 & 0 \\ 1 & 3 & 0 & 2 \end{pmatrix} (c)\ \begin{pmatrix} A & B & C & D \\ B & C & D & A \\ D & A & B & C \\ C & D & A & B \end{pmatrix}$$

2.2 Generation

A common method of generating a Latin Square of order n is by means of a cyclic shift. If row i has distinct entries in each of its columns, then row $i+1$ is row i with its entries shifted either to the right or the the left by 1 position by adding or subtracting 1 *mod* n to each entry. A Latin Square generated as such is a mod n addition table with the first row and column as its indices.

Theorem 2.1. *Let L be a $n \times n$ matrix with entries $a_{i,j} \in S$ where $i, j \in \{1, 2, ..., n\}$ designate the row and column of L respectively, and $S = \{0, 1, 2, ..., n-1\}$. Let the entries of the first row of L, $a_{1,j}$, be distinct such that $a_{1,1} \neq a_{1,2} \neq ... \neq a_{1,n}$. This ensures that every element of the set S appears exactly once in $a_{1,j}$. If we apply the row permutations $a_{i,j} = a_{i+1,j+1}$ mod n, each row will have distinct entries in each of its columns.*

Proof. Suppose two elements are identical in column j, i.e. $a_{r,j} = a_{s,j}$ where $r \neq s$ and $r, s \in \{1, 2, ..., n\}$. Let $t = s - r$. Then, $a_{r,j} = a_{r+t,j+t}$. Since $a_{i,j} = a_{i+1,j+1}$ due to the cyclic shift, then $a_{r,j} = a_{s,j+t} = a_{s,j}$. Hence, we find that $j = j + t$ and thus, $t = 0$ and $s = r$. Therefore, the row entries of each column are distinct. Since the entries of each row and column of L are distinct, L is a Latin Square. □

Example: Latin Square of order 5 generated by cyclic shift

$$\begin{pmatrix} 0 & 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 & 0 \\ 2 & 3 & 4 & 0 & 1 \\ 3 & 4 & 0 & 1 & 2 \\ 4 & 0 & 1 & 2 & 3 \end{pmatrix}$$

3 Real World Applications

3.1 Sudoku

Although it may not seem like it at first, there are many real-world uses for Latin Squares. The first one that usually comes to mind is Sudoku. The common Sudoku puzzle is indeed a Latin Square of order 9, as each entry in every row and column is distinct and contains every number from 1 to 9, though it is not necessary for a Latin Square to have the 3×3 subdivisions that are present in a Sudoku puzzle.

3.2 Tyre Example

Another motivating example is provided in the book "Design Theory Volume 2" [1], which involves the testing of 4 different tyre types on 4 different brands of car in such a way that each tyre model appears in each of the 4 wheel positions on each car. Below is a similar recreation of that example:

Car Tyres				
-	Front Left	Front Right	Rear Left	Rear Right
Audi	Type 1	Type 2	Type 3	Type 4
BMW	Type 2	Type 3	Type 4	Type 1
Honda	Type 3	Type 4	Type 1	Type 2
Mazda	Type 4	Type 1	Type 2	Type 3

3.3 Sport Example

Another common example would be that of sport tryouts. There are 5 available positions on a sports team, and if we want to find out which of the 5 players is most effective in which positions, we can design a series of tryouts in a Latin Square so that none of the players are in a position they already were in during a previous tryout. However, the correlation between positions being adjacent to each other numerous times is not immediately clear and would require further statistical analysis.

Team Tryouts					
-	Position 1	Position 2	Position 3	Position 4	Position 5
Tryout 1	Andy	Brian	Cian	Dan	Ethan
Tryout 2	Cian	Andy	Dan	Ethan	Brian
Tryout 3	Ethan	Cian	Andy	Brian	Dan
Tryout 4	Dan	Ethan	Brian	Cian	Andy
Tryout 5	Brian	Dan	Ethan	Andy	Cian

3.4 Statistical Experiments

The use of Latin Square designs in statistical experiments allows for two blocking factors. This means that these designs can be used to simultaneously control or eliminate two sources of nuisance variability (random variables fundamental to the model but aren't themselves of particular interest) via the rows and columns of the Latin Square. For example, suppose the mean yield of n types of grain is to be compared on a certain type of soil, with a rectangular field subdivided into n^2 plots. If we were to plant all of one type of grain in the same row or column, we wouldn't be certain whether the yield would be the result of the fertility of the soil where it was plated or of the type of grain itself. This is why it would be more beneficial to plant only one of each type of grain in every row and column in a Latin Square design as it would remove the nuisance variables and the mean yield would be more accurate.

Furthermore, one may also want to simultaneously test other factors influencing the yield such as fertilisers or treatments. Much like above, each treatment would be unique to each row and column resulting in a Latin Square. In addition, each treatment would also be applied once to each grain type. This resulting Latin Square is orthogonal (see next chapter) to the Latin Square above. Further research into the statistical models of Latin Square designs has been conducted by the PennState Eberly College of Science from which this example was taken [2].

4 Properties

4.1 Orthogonal Array Representation

Each Latin Square can be represented in a $n^2 \times 3$ array known as an orthogonal array. The 3 columns (or rows if you make a row orthogonal array) are denoted by r, c and s which represent the row, column and symbol in that row and column of the Latin Square. For example, take the following Latin Square of order 3:

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix}$$

Its orthogonal array is:

r	c	s
1	1	1
1	2	2
1	3	3
2	1	2
2	2	3
2	3	1
3	1	3
3	2	1
3	3	2

4.2 Orthogonal Latin Squares

In the previous chapter, we briefly mentioned the concept of Latin Squares being orthogonal to each other. A pair of Latin Squares of order n is said to be orthogonal to each other if, and only if, when superimposed, their ordered pairs of entries are distinct. In other words, let the Latin Square L_1 have entries $a_{i,j}$ and the Latin Square of the same order L_2 have entries $b_{i,j}$ where i, j denote the row and column respectively. L_1 and L_2 are orthogonal to each other iff the ordered pairs $(a_{i,j}, b_{i,j})$ are distinct for all i, j. By this definition, it is clear that a Latin Square is never orthogonal to itself as no pair of entries will be distinct.

Example: Orthogonal Latin Squares of order $n = 3$

$$(L_1) \quad \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix} (L_2) \quad \begin{pmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 2 & 3 & 1 \end{pmatrix}$$

$$\begin{pmatrix} (1,1) & (2,2) & (3,3) \\ (2,3) & (3,1) & (1,2) \\ (3,2) & (1,3) & (2,1) \end{pmatrix}$$
$$(Superimposed)$$

As each entry of the superimposed array is distinct, L_1 and L_2 are orthogonal to each other. Alternatively, L_2 is said to be L_1's orthogonal mate and vice versa.

Example: Non-Orthogonal Latin Squares of order $n = 3$

$$(L_1) \quad \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix} (L_2) \quad \begin{pmatrix} 2 & 3 & 1 \\ 3 & 1 & 2 \\ 1 & 2 & 3 \end{pmatrix}$$

$$\begin{pmatrix} (1,2) & (2,3) & (3,1) \\ (2,3) & (3,1) & (1,2) \\ (3,1) & (1,2) & (2,3) \end{pmatrix}$$
$$(Superimposed)$$

Here, L_1 and L_2 are not orthogonal to each other as pairs of entries in the superimposed array repeat themselves. Since not every pair is distinct, these are not orthogonal.

Not every order of Latin Square has an orthogonal mate. For example, as noted by R.N.Mohan, Moon Ho Lee, and Subash Shree Pokhrel in their paper on the orthogonality of Latin Squares [3], Tarry proved by brute force that no Latin Square of order 6 has an orthogonal mate. That is, no pair of Latin Squares of order 6 exist such that they are orthogonal to each other. Meanwhile some orders can have many orthogonal mates.

4.3 Equivalence Classes

An equivalence class is a subset of some set S in which its components are equivalent to each other, i.e. there exists an equivalence relation between them, meaning these three properties are satisfied: reflexive ($a \sim a$), symmetric (if $a \sim b$, then $b \sim a$) and transitive (if $a \sim b$ and $b \sim c$, then $a \sim c$).

Isotopy is an example of an equivalence relation, with the isotopy classes being the equivalence classes. Two Latin Squares are said to be isotopic to each other if there exist three bijections of rows, columns and symbols from one Latin Square to the other. In other words, we can permute the rows, columns and symbols of one Latin Square to obtain the other Latin Square.

Example:

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 1 \\ 3 & 4 & 1 & 2 \\ 4 & 1 & 2 & 3 \end{pmatrix} \quad isotopic \quad to \quad \begin{pmatrix} 4 & 2 & 1 & 3 \\ 2 & 1 & 3 & 4 \\ 1 & 3 & 4 & 2 \\ 3 & 4 & 2 & 1 \end{pmatrix}$$

By permuting the 1st and 3rd rows, the 2nd and 4th columns and the symbols 3 and 4 in the Latin Square on the left, the Latin Square on the right will be obtained.

Example:

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 1 \\ 3 & 4 & 1 & 2 \\ 4 & 1 & 2 & 3 \end{pmatrix} \quad not \quad isotopic \quad to \quad \begin{pmatrix} 1 & 3 & 2 & 4 \\ 4 & 1 & 3 & 2 \\ 3 & 2 & 4 & 1 \\ 2 & 4 & 1 & 3 \end{pmatrix}$$

These are not isotopic as there is no way of permuting the rows, columns and symbols of the first Latin Square to obtain the second one.

Isomorphism is another equivalence relation. A pair of isotopic Latin Squares is said to be isomorphic if their three bijections are equal. This can be shown by permuting the rows, columns and symbols of the orthogonal array of one Latin Square into the orthogonal array of the other. [4]

Example:

Below are given two Latin Squares of order 3 along with their corresponding orthogonal arrays.

$$(L_1) \begin{pmatrix} 2 & 1 & 3 \\ 3 & 2 & 1 \\ 1 & 3 & 2 \end{pmatrix}$$

r	c	s
1	1	2
1	2	1
1	3	3
2	1	3
2	2	2
2	3	1
3	1	1
3	2	3
3	3	2

$$(L_2) \begin{pmatrix} 2 & 3 & 1 \\ 1 & 2 & 3 \\ 3 & 1 & 2 \end{pmatrix}$$

r	c	s
1	1	2
1	2	3
1	3	1
2	1	1
2	2	2
2	3	3
3	1	3
3	2	1
3	3	2

We can re-order the orthogonal array of L_1 so that it matches the orthogonal array of L_2 without permuting anything within L_1 itself. If we make the first column of the orthogonal array of L_1 represent the symbol, the second column represent the row and the third column represent the column of L_1, the orthogonal array of L_1 will be identical to the orthogonal array of L_2, but still represent L_1.

$$(L_1) \quad \begin{pmatrix} 2 & 1 & 3 \\ 3 & 2 & 1 \\ 1 & 3 & 2 \end{pmatrix}$$

s	r	c
1	1	2
1	2	3
1	3	1
2	1	1
2	2	2
2	3	3
3	1	3
3	2	1
3	3	2

5 Mutually Orthogonal Latin Squares

5.1 Definition

Mutually Orthogonal Latin Squares (MOLS) of order n is a set of Latin Squares of order n such that they are all orthogonal to each other. That is, each Latin Square in a set of MOLS is orthogonal to all other Latin Squares in that set (other than itself).

Example 5.1: MOLS of order $n = 5$

$$\begin{pmatrix} 0 & 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 & 0 \\ 2 & 3 & 4 & 0 & 1 \\ 3 & 4 & 0 & 1 & 2 \\ 4 & 0 & 1 & 2 & 3 \end{pmatrix} \begin{pmatrix} 0 & 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 0 & 1 \\ 4 & 0 & 1 & 2 & 3 \\ 1 & 2 & 3 & 4 & 0 \\ 3 & 4 & 0 & 1 & 2 \end{pmatrix} \begin{pmatrix} 0 & 1 & 2 & 3 & 4 \\ 3 & 4 & 0 & 1 & 2 \\ 1 & 2 & 3 & 4 & 0 \\ 4 & 0 & 1 & 2 & 3 \\ 2 & 3 & 4 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 & 2 & 3 & 4 \\ 4 & 0 & 1 & 2 & 3 \\ 3 & 4 & 0 & 1 & 2 \\ 2 & 3 & 4 & 0 & 1 \\ 1 & 2 & 3 & 4 & 0 \end{pmatrix}$$

There is continuous, exhaustive research being done into finding or generating sets of MOLS of different orders. If a Latin Square is of order q generated from a finite field of that order, where q is either a prime number, or a power of a prime number, then there are $q-1$ MOLS of that order due to the cyclical nature of such finite fields. Where the study gets extensive is in regards to other orders of Latin Squares. The famous mathematician Leonhard Euler spent many years studying orthogonality of Latin Squares and even he made conjectures that ended up being disproven. For example, in the previously mentioned ”Design Theory” book [1], it is noted that Euler conjectured that no orthogonal Latin Squares existed of an order that is twice an odd number. This was disproven in 1959 by Bose, Shrikhande, and Parker when mutually orthogonal Latin Squares of order 22 were generated.

5.2 MOLS of Prime Order p

A set of MOLS of order p where p is a prime number is the simplest case to study as it is the one that has the most clear and easy to notice patterns for each Latin Square in the set. Let's take a closer look at the $p = 5$ example 5.1. By looking at the first Latin Square in that set, we can see that it was generated by a cyclic shift, much like in Chapter 2. By looking at the entire set, we can see that the first row of every Latin Square remains fixed. Furthermore, we can see that all other rows are just permutations of the rows of the first Latin Square. In other words, each individual row remains the same across each Latin Square and it's just the position of each row (except the first row) that changes. In fact, all rows except the first one permute in such a way that none of them are in a position they were in a previous Latin Square.

Upon further inspection, it can be noticed that the first column of each of the MOLS is a modulo 5 multiple of the column of the first Latin Square, i.e. the first column of the second Latin Square in the set of MOLS is 2 times the first column of the first Latin Square.

$$2 \times \begin{pmatrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{pmatrix} = \begin{pmatrix} 0 \\ 2 \\ 4 \\ 1 \\ 3 \end{pmatrix} mod\, 5$$

Similarly for the rest, with the first column of the third and fourth Latin Square in the set of MOLS being 3 and 4 times the first column of the first Latin Square modulo 5 respectively.

Finally, if the first column of the first Latin Square is added to each of its columns modulo 5, it generates the second Latin Square in the set of MOLS, adding twice the first column to each column modulo 5 generated the third Latin Square, and adding 3 times the first column to each column modulo 5 generated the fourth and final mutually orthogonal Latin Square.

With this algorithm, we generated 4 MOLS of order 5 by generating the first Latin Square by means of a cyclic shift, and then adding

$$k \times \begin{pmatrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{pmatrix} mod\, 5,$$

with scalar $k \in \{1, 2, 3\}$. The question now is: does this generate $p-1$ MOLS of any prime order p? The following section goes through the general case for any prime order p, showing it will always work.

5.3 The Algorithm

Let L_1 be a Latin Square of order p generated by a cyclic shift, where p is a prime number and the entries $L_{1,i,j} \in 0, 1, 2, ..., p-1$ with i, j denoting the rows and columns respectively. Let v denote the first column of L_1 and $k = 1, 2, ..., p-2$. Adding $k \times v$ to each individual column of L_1 mod p generates the remaining MOLS of order p, one for each value of k, leaving us with a set of $p-1$ mutually orthogonal Latin Squares. Below is the theorem that proves this algorithm will work only for prime orders p.

Theorem 5.1. *For any vector $v \ni \{0, 1, 2, ..., p-1\}$ in any order, and scalar $k = 1, 2, ..., p-1$, the product $k * v$ mod p has no repeated entries for any prime number p.*

Proof. Assume $k * v$ has repeated entries $(k * v)_i = (k * v)_j$ mod p, where $i \neq j$. We can rewrite this as $k * v_i = k * v_j$ mod p. Because p is prime and $k < p$, p and k will always be co-prime. The cancellation law of modular arithmetic states that if $k * a = k * b$ mod n and $k \neq 0$, then $a = b$ mod n. By applying this law, we get $v_i = v_j$ mod p. This is a contradiction as by definition, v has no repeated entries. Thus $k * v$ mod p has no repeated entries for any prime number p.

Alternatively, assume p has some factors r, s such that $p = r * s$ with $1 < r, s < p-1$. Let $k = r$ and let the entry $v_i = 0$ Naturally, $k * v_i = r * 0 = 0$ mod p. Now let a different entry $v_j = s$. We find that $k * v_j = r * s = p = 0$ mod p. This results in a repetition in $k * v$ as $k * v_i = k * v_j = 0$ mod p. Therefore the theorem does not hold for non-prime numbers. □

5.4 Code

Below is a bit of MATLAB code I wrote that generates $p-1$ MOLS of prime order p by means of the above algorithm. By setting ”order” to any prime number, we can generate the MOLS for that value.

This first function is used to generate the first Latin Square of order n by method of cyclic shifts.

```
function A = LatinSquare(order)

    % Checks if a valid order for a Latin Square is entered.
    if(order < 1 || mod(order, 1) ~= 0)
        fprintf("Latin square of order %g does not exist!", order);

    else
        A = zeros(order);
        numbers = 0:order-1;
        for i = 1:order
            A(i, :) = circshift(numbers, 1-i);
        end
    end
end
```

.

This next function checks for orthogonality between two Latin Squares.

```
function result = Orthogonality(A, B)
    % Checks if the input matrices are square and of the same size
    [n1, m1] = size(A);
    [n2, m2] = size(B);

    if n1 ~= m1 || n2 ~= m2 || n1 ~= n2
        error('Input matrices must be square and of the same size.');
    end

    n = n1; % Size of the Latin squares

    % Creates the array of vectors

    vectors = [A(:),B(:)];

    % Checks if all vectors are distinct from eachother
    distinct = unique(vectors, 'rows');

    % Outputs the result
    result = size(distinct, 1) == n^2;
end
```

.

With these functions defined, we can now start generating our set of MOLS. (Note: If writing in MATLAB Live Script, these functions go at the end.)

```
clear;

% Set the order of the Latin Squares
order = 7;

% Defines vector x based on the order
x = (0:order-1)'

% Creates first Latin Square
L{1} = LatinSquare(order);

% Generates n-1 Latin Squares by adding different
% multiples of x mod n to the first Latin Square
for i = 2:order-1

    L{i} = mod(L{1} + (i-1)*x, order);

    % You can uncomment the line below if you wish for the numbers to be
    % in range 1 to p instead of 0 to p-1.

    %L{i}(L{i} == 0) = order;
```

```

end

% Displays the Latin Squares
for i = 1:numel(L)
    disp(L{i});
end
```

.

This part checks for orthogonality between all generated Latin Squares in the list and displays result as a matrix. Each row / column represents a corresponding Latin Square. 1 means orthogonal, 0 means not orthogonal.

```
for i = 1:numel(L)
    for j = i:numel(L)

        O(i,j) = Orthogonality(L{i}, L{j});
    end
end

fprintf("Orthogonality matrix = ");
disp(O);
```

.

This final part confirms whether the generated Latin Squares are mutually orthogonal based on the above matrix. They're mutually orthogonal if the orthogonality matrix is upper triangular with 1s in the top right corner and 0s everywhere else.

```
T = triu(ones(order-1),1);
if (sum(T,"all") == sum(double(O),"all"))
    disp("We have a set of MOLS!");
else
    disp("Not mutually orthogonal!");
end
```

6 Powers of Primes

6.1 Order 4

In the previous chapter, we looked for patterns that would hint at some algorithm that could generate the $p-1$ MOLS of prime order p. But could we do the same for some order q that is a power of a prime? Let's take a look at MOLS of order 4 [5].

$$\begin{pmatrix} 0 & 1 & 2 & 3 \\ 1 & 0 & 3 & 2 \\ 2 & 3 & 0 & 1 \\ 3 & 2 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 2 & 3 \\ 2 & 3 & 0 & 1 \\ 3 & 2 & 1 & 0 \\ 1 & 0 & 3 & 2 \end{pmatrix} \begin{pmatrix} 0 & 1 & 2 & 3 \\ 3 & 2 & 1 & 0 \\ 1 & 0 & 3 & 2 \\ 2 & 3 & 0 & 1 \end{pmatrix}$$

By looking at this set of MOLS, are there any obvious patterns that will help us generate a set of MOLS of order 8? Much like for prime numbers, the first row remains fixed across all MOLS with all other rows being permuted. In fact, all rows except the first one seem to be cyclically shifted across each Latin Square. However, this may be a coincidence due to the small size of the MOLS.

Unfortunately, this is where the similarities to the prime example ends as the initial Latin Square is not generated by a cyclic shift. Furthermore, we can't generate MOLS by adding multiples of the first column due to the theorem in the previous chapter. But we could try find an orthogonal mate for order 8 by trying to discover any patterns involved in the generation of the first mutually orthogonal Latin Square of order 4.

6.2 Orthogonal Mate for Order 8

The first of the MOLS 6.1 of order 4 has 0s along its descending diagonal and 3s along its ascending diagonal. Also, the entries in the first row and column increase in steps of 1 from 0 to 3, while the entries of the last row and column decrease from 3 to 0 in steps of 1. My first approach was to try something similar for order 8 but with 0s and 7s along the diagonal and the ”edge” rows and columns increasing and decreasing in steps of 1 in a similar fashion. The remaining entries were filled out by extending the ”alternating” pattern of the existing Latin Square of order 4. Doing this generated the following Latin Square:

$$\begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 1 & 0 & 3 & 2 & 5 & 4 & 7 & 6 \\ 2 & 3 & 0 & 1 & 6 & 7 & 4 & 5 \\ 3 & 2 & 1 & 0 & 7 & 6 & 5 & 4 \\ 4 & 5 & 6 & 7 & 0 & 1 & 2 & 3 \\ 5 & 4 & 7 & 6 & 1 & 0 & 3 & 2 \\ 6 & 7 & 4 & 5 & 2 & 3 & 0 & 1 \\ 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 \end{pmatrix}$$

My first approach was to do a cyclic permutation of all rows except the first one by 1 step, resulting in this Latin Square.

$$\begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 \\ 1 & 0 & 3 & 2 & 5 & 4 & 7 & 6 \\ 2 & 3 & 0 & 1 & 6 & 7 & 4 & 5 \\ 3 & 2 & 1 & 0 & 7 & 6 & 5 & 4 \\ 4 & 5 & 6 & 7 & 0 & 1 & 2 & 3 \\ 5 & 4 & 7 & 6 & 1 & 0 & 3 & 2 \\ 6 & 7 & 4 & 5 & 2 & 3 & 0 & 1 \end{pmatrix}$$

However, I quickly saw that it wasn't orthogonal as in the original Latin Square 0 appears along the diagonal, meanwhile 1 appears three times along the diagonal of the new Latin Square, meaning that upon superposition, the pair (0,1) would appear three times along the diagonal. In fact, for a Latin Square of order 8 to be orthogonal to our first one, each number from 0 to 7 has to appear exactly once along both diagonals. Unfortunately, there is no simple pattern to this case that can be found by brute force and a deeper understanding of finite fields is required.

7 Connection to Finite Fields

7.1 What are Finite Fields?

I have mentioned the concept of finite fields numerous times throughout this paper. It isn't a subject I have studied yet, but I got an introduction to it while working on this project.

A field is a set on which addition, multiplication and some "0" is defined and an inverse exists for every element $\neq 0$. A finite field is a field with a finite number of elements. For example, $\mathbb{F}_5 = \{0, 1, 2, 3, 4\}$ is a finite field as each element is closed under addition and multiplication modulo 5. That is, adding and multiplying any elements of $\mathbb{F}_5$ mod 5 always results in an element of $\mathbb{F}_5$. Furthermore, each element has an inverse mod 5. $\mathbb{F}_6 = \{0, 1, 2, 3, 4, 5\}$ is not a finite field as 2, 3 and 4 do not have an inverse modulo 6.

7.2 Connection to Latin Squares

The field $\mathbb{F}_q$ is a finite field only if q is a prime number or a power of a prime. This is why, for Latin Squares of order q, there are $q-1$ mutually orthogonal Latin Squares, as their construction arises from finite fields. This is also why there aren't $n-1$ MOLS of any order $n \neq q \in \mathbb{N}$, because $\mathbb{F}_n$ is not a finite field and thus it is more challenging to find an orthogonal mate for such Latin Squares.

Although it is established we can generate $q-1$ MOLS from the finite field $\mathbb{F}_q$, there is still the open question of: is the converse true? That is, given a set of $q-1$ MOLS, can one construct a field or field-like structure?

8 Conclusion

When I first started looking into the topic of Latin Squares, I thought it was a rather simple concept - a square matrix with no repeating entries in any row or column. However, I quickly realised how much more nuance there is to this subject. I learned how Latin Squares have many properties that mathematicians spend years of their lives researching. One with little experience can spend a lot of time searching for patterns that may not even be there. I learned more about finite fields and how their properties connect to Latin Squares and how extensive knowledge of the topic is required to investigate more complicated properties of Latin Squares.

References

[1] Thomas Beth, Dieter Jungnickel, and Hanfried Lenz. *Design Theory*. Vol. 2. Cambridge University Press, 2000.

[2] *The Latin Square Design*. URL: https://online.stat.psu.edu/stat503/lesson/4/4.3.

[3] R.N. Mohan, Moon Ho Lee, and Subash Shree Pokhrel. "On Orthogonality of Latin Squares". In: *Information & Communication* (2006). URL: https://arxiv.org/ftp/cs/papers/0604/0604041.pdf.

[4] *Small Latin Squares and Quasigroups*. URL: https://en.wikipedia.org/wiki/Small_Latin_squares_and_quasigroups.

[5] *Mutually Orthogonal Latin Squares*. URL: https://en.wikipedia.org/wiki/Mutually_orthogonal_Latin_squares.