Nonograms: Multi-Color Grids Impact on Occurrence and Predictions on Uniqueness

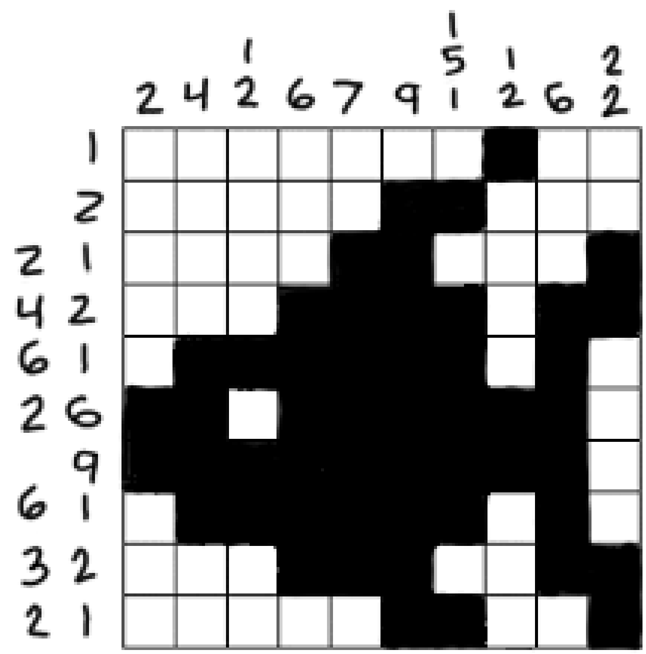
Emily Lasecki

September 2, 2024

Background

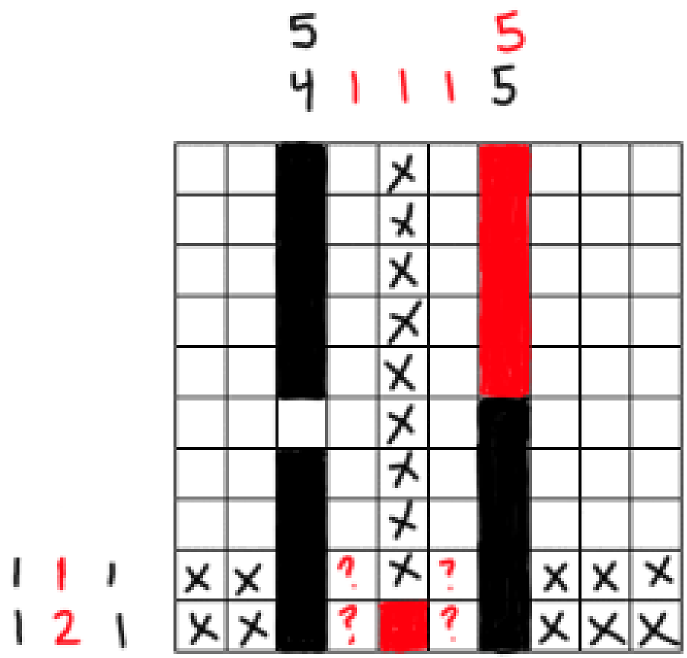
Nonograms, also known as Picross, Paint by Numbers, Japanese Puzzles, among others are an application of discrete tomography in the form of a logic puzzle, where based on numbers on the top of columns and to the left of rows, users fill in squares of a grid with one or more colors to reconstruct an image [1]. Finding forcible lines or parts of lines allows the player to be certain of the status of a certain square [2]. As more squares are determined to be filled or left blank, players can use both information from the newfound squares and clues of different rows/columns to find the solution.

Figure 1: Nonogram of a fish



These clues indicate how many “chunks” of color appear in a row. For example, a 10x10 grid may have the numbers 5 and 4 appear in a row. The user then can tell that five squares must be filled in together, as well as another four in that order starting from either the top or left depending on if the clue is for a column or row. Since the colors are the same, there must be at least one blank square to separate them. In the case of a 10x10 board, there must be strictly one blank square between them, but for a 11x11 grid game or larger more than one square may separate them. If a nonogram clue contains two different colors next to each other, a blank square does not need to separate them.

Figure 2: Forcible and non-forcible sequences



Not all nonogram grids are created equal. Some combinations of clues result in a game where players must make a guess as to where to fill a square to complete the game. A well designed nonogram will only bring rise to one solution, but this requires checking by the developer to ensure it does so. This begs the question of how creation of nonogram games could be automated. Exactly how many nonogram grids are possible, and how many are unique? Calculation of the former involves elementary combinatorics – counting how many different combinations of grids there are. Uniqueness is determined by how many different combinations of clues there are, and how many of these bring rise to exactly one grid. The ways in which rows and columns interact to narrow down possible boards is difficult to calculate, and combinatorics alone cannot solve, hence the difficulty with measuring uniqueness [2].

Project Objectives

This program aimed to calculate the number of nonogram combinations of each possible color. Then this data was exported into Excel to create a graphic that shows how colors play a role in the number of combinations. The result allows a baseline for future projects to use the number of possible grids as a comparison point for unique grids.

Method of Calculation

To calculate the number of nonogram grids, I utilized some basic combinatorics. The Principle of Inclusion Exclusion can be applied perfectly to this scenario. In the instance of each color, calculate color ^ size. This method doesn't dictate that all of colors must be used however, so to correctly calculate we must subtract the instances where not all the colors are used. Binomial coefficients can be used to calculate how many instances each color ^ size must be subtracted. Since each alternating case overcounts those of the following case, we can apply this principle to accurately count the number of grids.

One way to validate this counting method is to count in two ways. For demonstration, take the case of a 2x2 grid with 3 colors.

Proof:

We count the number of ways to fill in a 2x2 grid with 3 colors, red, blue, and black.

Part 1:

First, we count the number of ways to fill a 2x2 grid without restrictions that all colors must be present at least once. We choose 1 of 4 options (red, blue, black, or blank) for each square in the grid, or 4^4 = 256. But we must subtract the cases where all 3 colors aren't present. There are 3 combinations of colors that create a nonogram grid of 2 colors, red and blue, red and black, and blue and black. (This can also be calculated by (3 choose 2)). A grid of 2 colors can be created with 3^4 = 81. But this subtracts grids of a single color too many times, so we must add them back. There are (3 choose 1) = 3 ways to pick a color, so we add back 2^4 = 16, three times. But this overcounts the grid containing no colors, or the blank board, so we must add the blank board back. This gives 256 – 3(81) + 3(16) -1 = 60 ways for a board of 3 colors.

Part 2:

Alternatively, we count two different scenarios for this board. Assume there are no blank squares in the 2x2 grid. First choose which of the 3 colors to duplicate (3 ways). Next, place those 2 colors. (4 choose 2) = 6 ways. Now, place the next color (2 ways). And place the final color (1 way). 6 * 2 *3 = 36 ways to create a board without blank spaces.

In the second scenario, assume there is one blank square. First, place the blank square (4 ways). Next, place the black square (3 ways). Now, place the red square (2 ways). Finally, place the blue square (1 way). 4 * 3 * 2 = 24 ways for a grid to contain a blank square.

By rule of sum, noting these sets are disjoint, we have 36 + 24 = 60 ways to make a board of 3 colors.

Conclusion:

Since both procedures count a 2x2 nonogram grid of 3 colors, we know that the two counting methods are equal. This provides evidence that the principle of inclusion exclusion is viable to this scenario. The principle demonstrated in part 1 of the proof is the basis of the program.

To create a more comprehensible graph, I calculated the percentage of total colors and percentage of possible graphs rather than just displaying the number of colors and number of possible grids. Scaling this data allows us to see the trend change across each size board. For each color c and number or squares in the grid x, (c/x) * 100 is the percentage of possible colors. Similarly for each size, we use our program to find the color with the maximum number of grids, call it x. We then take our current number of grids at the given iteration, call it n and calculate (n/x) * 100 for the percentage of possible grids.

Caveats and Future Improvements

- I couldn't manipulate arbitrarily large floats, so I had to round the percentage of possible grids. Large integers don't transfer to Excel in the current format, so they were also converted to scientific notation. The margin of error is trivial for the purpose of visualization, but for future calculations a more precise result will be necessary.
- Python limits printing more than 4300 characters with a warning message users can override. This becomes an issue with larger grid sizes. Anything above 20x20 has not been tested and may not be possible without manually expanding this limit.
- Under the current version of the code, 1x1 through 20x20 grids are automatically tested without any user input. Testing only specific, larger, or rectangular sizes is

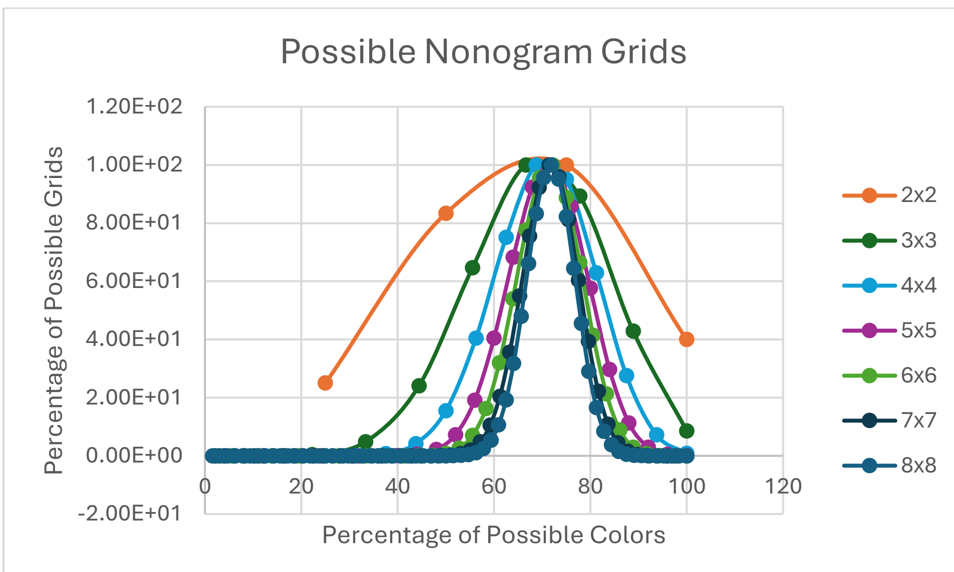
not possible under the current version. Testing various sizes provides another variable that would impact accuracy and should be isolated and tested in a separate study.

- Matplotlib or similar Python libraries could create better visualizations of the data. At the time of execution, I was not aware of these extensions.
- Using a default scaling feature designed for machine learning may improve the run-time of this program.

Results

Upon graphing, the following trend appears.

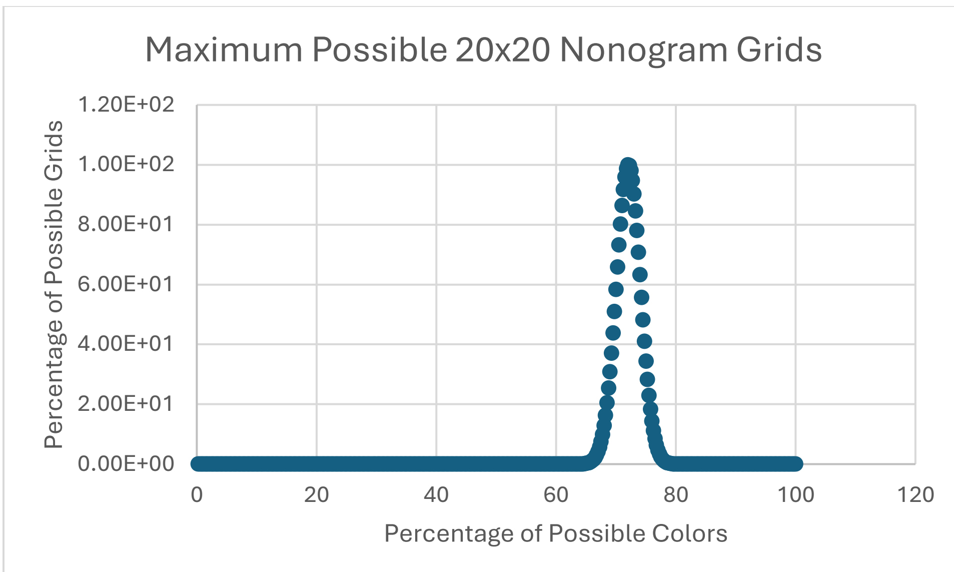
Figure 3: Nonogram possibilities for 2x2 through 8x8 grids



The graph for the 2x2 grid provides little insight, but at the 3x3 graph the trend starts to emerge. Somewhere around the 75% mark, the number of colors provides the largest number of possible grids. This closes in to about 72% as the size of the boards increase. Before 72% the number of possible grids appears to grow in an exponential manner, and after 72% the number begins to decline.

This implies that the colors provide more possibilities until the sheer number of them begin to cause restrictions. This is as expected. We know that a board from zero colors to one should increase because for any size grid with no colors there exists one, the blank board. Since grids must always be larger than 2x2, (2 ^ size) -1 where size is at minimum 4, is always greater than 1 hence the increase. We know that for greater numbers at some point possibilities decrease as well. For any grid we know that having the maximum number of colors makes the size and number of colors the same, with possibilities being size! (size factorial). After this point, there is no possible grid that meets color requirements as trying to fill a n size board with n+1 colors is not possible. Proving consistent increases between other numbers requires a more sophisticated proof.

Figure 4: 20x20 nonogram possibilities



Next Steps

Although this data shows that the number of possible grids of multiple colors is shaped like a bell curve, it remains unknown how putting uniqueness restrictions affects the growth of the number of grids. Combinatorics alone cannot determine uniqueness. To test for this, I plan to create a nonogram game that can take a key as input dictating what colors go in what squares, decipher the corresponding clues, then "solve" the puzzle and mark each case as unique or not unique. Since Python can process very large strings, up to a 20x20 board will be possible. If the data isn't distinct or if it would provide better results to calculate larger grids, I may expand this limit. I then will compare the ratio of possible boards with unique ones to establish how this trend changes with the grids being scaled up and/or number of colors being altered.

Areas for Further Study

Nonograms can be analyzed for uniqueness and occurrence by isolating one or more variables whether it be size, shape, or color. From a combinatorics perspective, the ways in which the rows and columns restrict grids is currently unknown [2]. The more frequent exploration of nonograms have been attempts at creating the most efficient algorithm to solve arbitrarily large boards. Jan Wolter's forum post on webpbn.com has extensive documentation on different algorithms' performances [4].

Several variations of the nonogram game exist. Slanted and curved nonograms break from the traditional rectangular grid, where lines or curves form the edges of the board. Ways to ensure clue readability have been studied in-depth [5]. Nintendo's Picross franchise introduced Mega Picross, where the game operates as a standard single color nonogram except two or more sets of lines are merged and have shared clues for how large chunks should be between those two lines. Uniqueness, solvability, and occurrence of Mega Picross boards has not been studied. As of now, colored Mega Picross does not exist, but begs questions of at what threshold of colors does the game cease to produce Mega boards. Another Nintendo Trademark Micross, also known as mosaic nonograms, prompts the user to solve multiple nonograms that linked together create a bigger image. An answer to mosaic nonogram uniqueness and solvability can be expanded on from the uniqueness of a standard board.

Beyond the restrictions of 2D plains and strictly two axes of information, similar discrete tomography problems offer applications to X-ray imaging in the form of CT (Computed Tomography) scans [6]. Advanced research on nonograms may realize a connection between the two.

References

[1] Batenburg, K., & Kosters, W. (2008, August 1). Solving nonograms by combining relaxations. *Science Direct*. https://www.sciencedirect.com/science/article/abs/pii/S0031320308005153

[2] Mullen, R. (2009). On Determining Paint by Numbers Puzzles with Nonunique Solutions. *Digital Commons*. https://digitalcommons.sacredheart.edu/math_fac/78/

[3] Salcedo-Sanz, S., Portilla-Figueras, A., Ortiz-Garcia, E., Perez-Bellido, A., Yao, X. (2007, June). Teaching Advanced Features of Evolutionary Algorithms Using Japanese Puzzles. *Research Gate*. https://www.researchgate.net/publication/3051139_Teaching_Advanced_Features_of_Evolutionary_Algorithms_Using_Japanese_Puzzles

[4] Wolter, J. (n.d.). Survey of Paint-by-Number Puzzle Solvers. *Webpbn.com*. Retrieved September 2, 2024, from https://webpbn.com/survey/index.html

[5] Klute, F., Löffler, M., Nöllenburg, M. (2021, October). Labeling nonograms: Boundary labeling for curve arrangements. *Science Direct*. https://www.sciencedirect.com/science/article/pii/S092577212100047X [7] Batenburg, K., & Walter, K. (2005, January). A discrete tomography approach to Japanese puzzles. *Research Gate*. https://www.researchgate.net/publication/30469970_A_Discrete_Tomography_Approach_to_Japanese_Puzzles

[6] Richard, G., & Peter, G. (1997, June). Discrete tomography: determination of finite sets by x-rays. *Western Washington University*. https://cedar.wwu.edu/math_facpubs/26/?utm_source=cedar.wwu.edu%2Fmath_facpubs%2F26&utm_medium=PDF&utm_campaign=PDFCoverPages