Enhancing Image Classification Using Cellular Automata Inspired by Natural Transformation Processes

Data Science Exam 2022

Emil Trenckner Jessen
201807525@post.au.dk

Jakob Grøhn Damgaard
201808996@post.au.dk

Johan Kresten Horsmans
201810219@post.au.dk

School of Communication and Culture
Aarhus University
Langelandsgade 139, 8000, Aarhus C, Denmark

May 31st, 2022

Abstract

Many recent advances within the field of image classification are coming from the development of various new data augmentation methods which enable ample model training when faced with scarce data resources (Shorten & Khoshgoftaar, 2019). This paper investigates a novel approach to data augmentation in which cellular automata simulations are used as a method to enhance image data distinctiveness. We develop a set of cellular automata rules and apply them to the Fashion-MNIST data set to assess whether it is possible to enhance discriminatory features of the images in a way that will improve the classification performance of a simple multinomial logistic regression model. The rule sets utilized are rooted in natural transformation processes, namely, *the Game of Life*, *pitting corrosion* and *melting*. Our results suggest that the cellular automata algorithms are successful in enhancing unique and distinguishable key-features between the different image categories. The inclusion of the processed images leads to an increase in performance from 0.762 to 0.807 as measured in Matthews Correlation Coefficient. Hereafter, we discuss what might constitute the emergent discriminatory properties in the processed images which lead to improved performance. Lastly, we explore potential future prospects of using CA augmentation in combination with multidimensional convolutional neural networks within the field of computer vision.

Keywords: Cellular automata, Data augmentation, Fashion-MNIST, Image classification.

Regarding initials and code:

ETJ stands for *Emil Trenckner Jessen*.

JGD stands for *Jakob Grøhn Damgaard*.

JKH stands for *Johan Kresten Horsmans*.

Each individual section of this exam has been labeled according to the name of the primary contributor. Nonetheless, it is crucial to stress that the entire exam is regarded as a wholly collaborative effort, since each part has been jointly written and thoroughly edited by all authors. Additionally, all authors have collaborated on the coding and analyses.

The code for our analysis can be found in the following GitHub repository.

Character count: 60,000

Contents

1 Introduction (ETJ)

"Chaos in the technical sense is not random at all. It is completely determined, but it depends hugely, in strangely hard-to-predict ways, on tiny differences in initial conditions", (Dawkins, 2017, p. 23).

Developing viable machine learning methods for classifying images is an integral aspect of computer vision. The field is only becoming increasingly more relevant and well-researched in line with the increased demand for faster and more efficient classification algorithms caused by the rapid expansion in the amount of visual content being generated every day (Meel, 2021; Nath et al., 2014). Automatized image classification is utilized as a key component across a wide array of different applications and fields such as *medicine*, *military* and *engineering* (Bi et al., 2021; Caicedo et al., 2009; Chapelle et al., 1999; Das et al., 2018; Huang et al., 2011). During the past years, there have been sizable leaps in the utilization of complex machine learning approaches for image classification – primarily resulting from the usage of deep learning techniques (Vermeire et al., 2022).

Nonetheless, image classification remains a highly challenging task due to a large amount of inconsistent variation between images belonging to the same class (e.g., differing backgrounds, lighting conditions, varying points of view, obstruction, deformity) making it difficult to build generalizable algorithms (Bi et al., 2021). As such, there is still a need for improvements in the field in order to create highly robust models that can consistently generalize to out-of-sample data. In light of the huge amount of data required to sufficiently train deep-learning architectures without overfitting, a substantial amount of research has been put into improving model classifications through data augmentation rather than model optimization when faced with sparse datasets (Shorten & Khoshgoftaar, 2019).

In this paper, we propose a novel technique for augmenting image data using cellular automata with the goal of enhancing image classification performance in domains where data

scarcity is a persisting problem. To test the proposed framework, we augment Fashion-MNIST images using iterative cellular automata algorithms. During the augmentation process, pixel changes are logged and saved and the last iteration of the augmentation is subsequently used as input in a classification algorithm.

Before delving into the specifics of this scope, however, it is important to understand the concept of cellular automata.

1.1 Cellular automata (JGD)

Cellular automata (CA) are a class of dynamical computational systems consisting of a two-dimensional grid of cells where each cell classically occupy only one of a finite array of possible discrete states. Throughout the simulation, the state of each individual cell synchronously undergoes an iterative step-by-step transformational evolution dictated by the previous state of adjacent cells in the grid and a set of prespecified rules (Rosin, 2006; Wolfram, 1983). Contrary to partial differential equations, space, time and states are discrete rather than continuous in traditional CA (Berec, 2002; da Silva et al., 2015; Wolfram, 1983). For a more elaborate account of the mathematical underpinnings of CA, see the section *Mathematical underpinnings and notational system for cellular automata (JKH).*

CA work at an intriguing intersection between simplicity and complexity which we wish to exploit. CA simulations and their capacity to produce a broad range of oftentimes highly advanced and spatiotemporal patterns from a set of simple rules have been employed in many different modeling frameworks across fields such as physics, biology and data science (da Silva et al., 2015; Rosin, 2006). The power of CA comes from the fact that, although each individual cell only follows a very sparse amount of primitive and local rules, very sophisticated macro-level behavior can emerge from the combination of multiple grids of cells and their plethora of micro-level interactions. In other words, in spite of the fact that every individual cell only has access to the sparse information from its proximate neighbors,

information from local clusters of cells can be propagated at each step in the simulation, which can give rise to more global features in the overarching CA system (Conway, 1970; Rosin, 2006; Wolfram, 1983). Our paper aims to develop CA-rules capable of creating such emergent overarching patterns which we will attempt to use as a way of enriching image data with additional systematic information that can improve the performance of classification models by making images belonging to different classes of objects more differentiable.

1.2 Cellular automata with offset in natural material transformations (JGD)

A noteworthy disadvantage with traditional CA-simulations is rooted in the fact that the rules of the systems have to be meticulously specified and developed by hand. This means that using CA is a very tedious methodology for more complex large-scale problems (Rosin, 2006) and that it is difficult to reason confidently a priori about the outputs. As a way of making the process of defining sensible transition rules more effective and robust, compared to specifying and optimizing arbitrary rules solely through intuition and trial-and-error, we decide to look towards natural phenomena for inspiration. Numerous monumental areas and methods of data science and computer vision have successfully drawn inspiration from naturally occurring processes and phenomena. For instance, the original inspiration for CNN was based on discoveries pertaining to the architecture and constituents of the visual cortex while new findings on the subject also continue to inspire developments within computer vision today (Draelos, 2020; Fu et al., 2016; Hubel & Wiesel, 1959; Kalpande, 2018). Other examples include using neural networks based on *ant colony–* and *particle swarm optimization* to classify tuberculosis sclerosis disease (Ripon et al., 2020) and utilizing genetic algorithms to solve problems pertaining to unit commitment and economic dispatch (Rodríguez del Nozal et al., 2020). Generally, applying iterative, discrete CA over multiple iterations allow for gradual augmentation of the starting data. Hence, we find it reasonable to specify rules that attempt to model and simulate naturally occurring transformation processes. In na-

ture, distinct starting conditions may over gradual modulation given a set of transformation rules evolve into drastically different states. This is true, e.g., in the case of natural growth or decay; an image of a lemon and a golden melon may be hard to visually distinguish, however, a rotten lemon looks markedly different from a rotten melon. Our hope is that by simulating natural transformations, the CA systematically generate emergent patterns that are dependent on the starting conditions innate to the distinct image classes.

Based on the considerations outlined above, we propose and compare three different nature-inspired solutions for utilizing CA in image processing: *pitting corrosion*, *melting* and *the Game of Life*. The first CA – *pitting corrosion* – is borrowed and adapted from da Silva et al. (2015) and loosely models the corrosion of an alloy surface. The second rule is a novel solution devised by the authors and aims at simulating the melting of an object. The latter rule is perhaps the most popularised CA to be developed and aims at simulating an environment in which cells live or die according to the state of their neighbors. All rules are used to gradually transform the original images over several iterations. The augmented result after the last iteration is then extracted and concatenated to the original image and used as input in a classification framework. For a technical unpacking of the proposed CA rules, please refer to section *Applied cellular automata transition rules (JGD)*.

1.3 Scope of the paper (JKH)

We hypothesize that evolving and augmenting the Fashion-MNIST images using cellular automata algorithms may enhance image classification performance by reinforcing the differences between images belonging to the same class whilst simultaneously decreasing confounding overlap between images from distinct classes. A sandal and a shoe may share similar visual properties, however, their gradual corrosion patterns may be highly distinct. Hence, simulating natural transformation processes may enhance or draw out properties that help a classification algorithm distinguish between conventionally similar images. We believe

that our proposed method would be especially applicable in classification problems where the amount of input data is sparse and more predictive variables are needed. We further anticipate that the features derived from the CA data augmentation may be of highest value for edge cases that are prone to being misclassified by ordinary classification frameworks. The main onset for the aim of this assignment is thus to explore whether we can improve model classification through augmenting images with different constellations of cellular automata algorithms. More specifically the research question of the paper is the following:

Can the performance of image classification algorithms, trained and tested on the Fashion-MNIST data set, be improved through the use of data augmented images generated through iterative cellular automata algorithms inspired by natural transformation processes.

1.3.1 Other relevant literature (JGD)

Augmenting images to enhance data quality or data quantity to increase the performance and robustness of image classification algorithms is widely exploited (Krizhevsky et al., 2012; Lin et al., 2021; Nanni et al., 2021; Zhong et al., 2020). Simple image augmentation has been shown to improve classification performance on datasets such as MNIST (Baldominos et al., 2019). Particular methods include geometric transformations and rotations, color space augmentation, image mixing, or deep learning frameworks such as general adversarial networks. Notably, it is also common to augment images using kernels or filters specialised at e.g. blurring or detecting edges (Shorten & Khoshgoftaar, 2019). These kernels are slightly reminiscent of the neighborhood-kernel based transition rules used in cellular automata, however, without the iterative element. Researchers have continuously explored the development of CA rules that may be applied for tasks pertaining to image analysis and processing (Preston Jr & Duff, 2013). Successful applications include image thinning, noise filtering and simple object recognition (Karafyllidis et al., 1997; Rosin, 2006). Recently, a

few attempts have been made at exploiting the capacities of CA to augment image data and through this spawn new features and patterns that may improve classification. Florindo and Metze (2021) and da Silva et al. (2015) both use similar CA-based frameworks to augment texture images and spawn novel or exaggerate key patterns in the pixel values. From the augmented images, they extract time series image descriptors which are used for classifying textures. These studies are - to the authors' knowledge - the only published research to bear close resemblance to the ideas proposed in this paper. Hence, this project is highly explorative in nature and should primarily be regarded as a proof-of-concept. The methods used and decisions made may beneficially be continuously improved in future research.

Another attempt has been made at employing CA for image classification without the use of traditional augmentation (Randazzo et al., 2020). They utilise a model of self-organising neural CA to dynamically classify handwritten digits (Mordvintsev et al., 2020). The model cleverly capitalizes on the inter-cell communicative properties of CA. The model works in a continuous number space which allows the transition rules to be differentiable. The attribute allows the model to optimise the rules using loss functions and back propagation. Conceptually, this method differs substantially from those presented here. It does, however, illustrate the powerful potentials of simple CA rules for data classification.

2 Methods

2.1 Fashion-MNIST Dataset (JKH)

For our project, we utilized the *Fashion-MNIST* dataset developed by Xiao et al. (2017). The Fashion-MNIST dataset consists of 70,000 grayscale images of 10 types of different fashion products evenly distributed with 7.000 images in each category (Baldominos et al., 2019). The Fashion-MNIST dataset is designed to act as a direct alternative to the original MNIST dataset developed by LeCun et al. (1998) and, followingly, it features identical data format and image resolution (28×28).

The original MNIST dataset has become one of the most popular datasets for benchmarking image classification models and is utilized for the introductory *"Hello world"* guides for many machine learning frameworks (e.g. TensorFlow, Pytorch, scikit-learn) (Baldominos et al., 2019; Xiao et al., 2017). This is, in large parts, due to the small size of the dataset and that it affords easy interpretation and, followingly, fast evaluation and subsequent comparison between different classification models. (Xiao et al., 2017). Nonetheless, modern developments in machine learning have made it possible of achieving accuracies exceeding 99.7% as published by Ciregan et al. (2012) and Wan et al. (2013). As a result, there have been multiple calls amongst researchers in the deep-learning field for the development of a more challenging, but equally simple, alternative (Xiao et al., 2017). This has been the motivation behind the development of the Fashion-MNIST dataset (Xiao et al., 2017).

The dataset was made from images retrieved from the online fashion retail platform Zalando. All pieces of clothing on Zalando are professionally photographed with a monochromatic background from a wide array of different angles. However, in the dataset, only the front look thumbnail is used. The images were collected from multiple different types of users: *neutral, kids, male* and *female*. Furthermore, white clothing items were excluded due to the low contrast between product and background. The images were converted to grayscale and underwent a series of standardization steps that can be found in Xiao et al. (2017). For a graphic depiction of 100 samples from the dataset, see *figure 1*.

Figure 1: *Visualization of a small excerpt of the Fashion-MNIST dataset consisting of 100 images of the 10 different classes of objects.*

2.2 Mathematical underpinnings and notational system for cellular automata (JKH)

Due to the complexity of some of the mathematical definitions, the mathematical underpinnings and the notational system for cellular automata rule sets, the explanation and elaboration will be divided up into multiple subsections; 2.2.1 - 2.2.4. The notation used in this paper is based upon da Silva et al. (2015) and Florindo and Metze (2021).

Mathematically 2D CA, C, can be represented the following way:

$$C\langle G, S, s, s_0, N, \Phi\rangle$$

Where,

- G corresponds to a 2D grid containing cells $c_{i,j}$.
- S corresponds to the finite amount of k states, where $S \subset \mathbb{N}$.
- $s(c_{i,j}, t)$ is the output function giving the state of each individual cell $c_{i,j}$ at each individual time step t.
- $s_0(c_{i,j})$ is the initialized state of each individual cell $c_{i,j}$.
- $N(c_{i,j})$ is the neighborhood function which couples each cell $c_{i,j}$ to the predefined neighborhood.
- $\Phi(c_{i,j}, N, t)$ is the transition function which takes the current cell $c_{i,j}$ and its neighborhood at t and computes the state *s($c_{i,j}$, t + 1)* of each individual cell $c_{i,j}$ at the proceeding t-th + 1 time step.

2.2.1 Grid (ETJ)

Throughout this assignment, each two-dimensional 28 x 28 array of pixels which constitutes the images in the Fashion-MNIST dataset are used as the grid of cells, G. Throughout this paper, the cells in the CA will be indexed using the notational system found in *figure 2*, where the coordinates of the cell in the middle of the array in question will always be $c_{i,j}$.

Before applying each iteration of each CA rule, the grid is padded using reflective padding in order to analyse cells in the outer layers of the grid. This is similar to the framework applied in other CA-related image processing articles referenced in this paper (da Silva et al., 2015; Florindo & Metze, 2021).

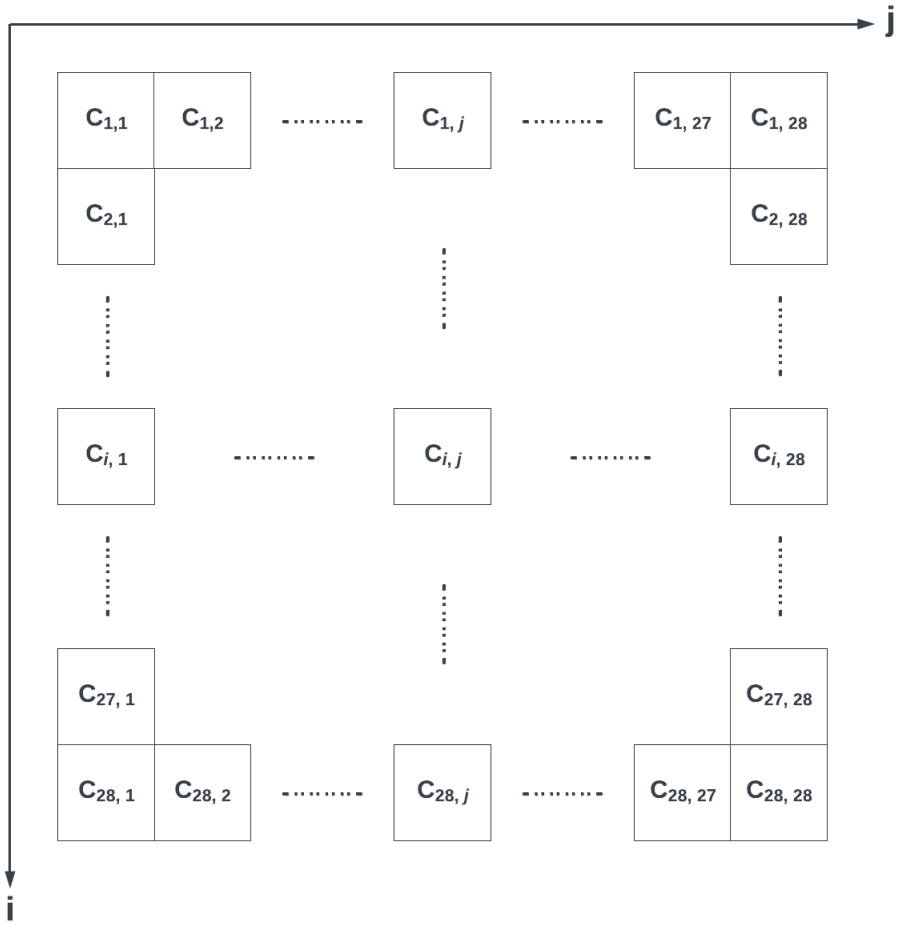

Figure 2: *Figure displaying the notational system for the coordinates of the cell in the image arrays prior to applying reflective padding.*

2.2.2 Neighborhood function N (ETJ)

There are numerous different types of neighborhoods one can utilize for 2D CA – the most prolific being the *Von Neumann*- and *Moore* neighborhoods (da Silva et al., 2015). For this analysis, we have chosen to use the Moore neighborhood which consists of all cells sharing a vertex with cell $c_{i,j}$ – see *figure 3* for graphic depiction.

$C_{i-1,j-1}$	$C_{i-1,j}$	$C_{i-1,j+1}$
$C_{i,j-1}$	$C_{i,j}$	$C_{i,j+1}$
$C_{i+1,j-1}$	$C_{i+1,j}$	$C_{i+1,j+1}$

Figure 3: *Representation of the Moore neighborhood for a given cell* $c_{i,j}$.

2.2.3 Discrete states (JKH)

Each individual cell $c_{i,j}$ is, in our study, occupying one of the possible states k from the set S. The initial condition of G for each cell in G at t=0 - s($c_{i,j}$, 0) is given by the grayscale value for each of the corresponding pixels in the individual Fashion-MNIST-images (binarized for the Game of Life).

2.2.4 Transition functions (JKH)

The purpose of the transition function Φ is to compute the state k of each individual cell $c_{i,j}$ at the proceeding (t+1)-th time step on the basis of the specified rule set and the states of the neighborhood N surrounding the cell. The transition functions (i.e. CA rule sets) and the reasoning behind them are explained in-depth in the following sections.

2.3 Applied cellular automata transition rules (JGD)

The following subsections outline the technical details and elaborate on the intuitions behind the three transition rules inspired by natural transformation processes that are used for augmenting the Fashion-MNIST data. Note that the rules are not intended to minutely simulate the actual physical processes but rather emulate the behavior in a loose, superficial fashion. To get an intuition for the augmentation process, please refer to *figure 4*.

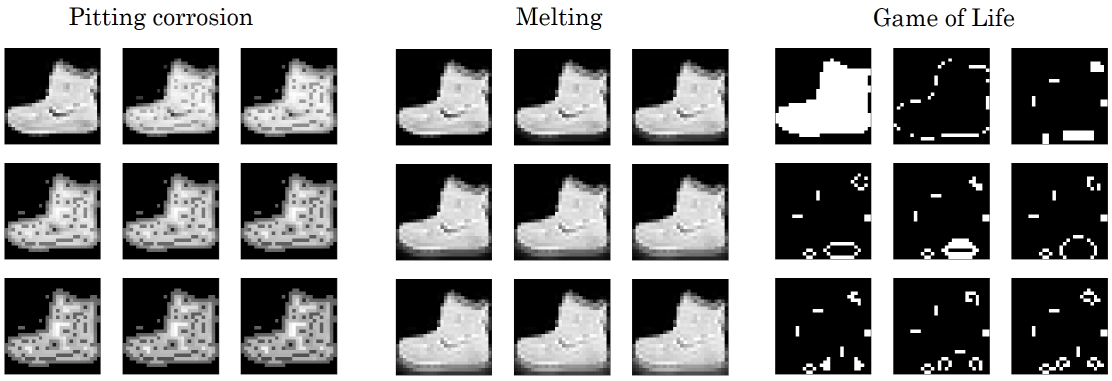

Figure 4: *Visualization of 8 iterations of each of the three CA processes*

2.3.1 Pitting corrosion (JGD)

Our first CA transition rule is designed to mimic the process of pitting corrosion happening on alloy surfaces and is borrowed and adapted from the paper by da Silva et al. (2015). In essence, the rule treats the image to which it is applied as an alloy surface and interprets the discrete pixel colour values as the surface morphology that determines the rate of corrosion; high pixel values signify peaks and low values signify troughs. Over time all alloy surfaces will gradually corrode, a process causing small pits and holes in the surface. Local irregularities in the surface morphology that are neither negligible nor too extreme accelerate the rate of corrosion (Malki & Baroux, 2005). Corrosion - as defined in this framework - is gradually added to the image by increasing the pixel colour values. da Silva et al. (2015) - from whom the rule set has been adapted - apply the rule to texture images from the Brodatz database (Hersey, 1968). They continuously track the cumulative mass of corroded product for each applied iteration of the transition rule and use the resulting time series as an image descriptor for classifying the type of texture of the image.

As stated, we adapt this rule to the Fashion-MNIST images by treating the fashion objects as alloy surfaces that rust over time on the basis of their morphology, although we do not use the cumulative mass of corroded product in the classification. As such, each gray-scale image $I(i, j)$ is regarded as a 2D-grid G and at $t = 0$ cell states $s(c_{i,j}, 0)$ are defined

as the discrete gray-scale value ranging from 0 to 255 for the corresponding pixel. For each iteration of the CA, each pixel in the grid G is synchronously analysed in regards to its Moore neighborhood $N(c_{i,j})$ and its cell state is updated according to the corrosion-inspired transition function Φ. For each cell at each time step t, the difference between the cell state and the minimum state in its neighborhood is found:

$$d_{i,j} = s(c_{i,j}, t) - min(\tilde{s}(N(c_{i,j}), t))$$

This difference is interpreted as the local impurities in the surface of the alloy in the vicinity of the cell. If the difference is below a defined threshold parameter v, the surface is regarded as so smooth that no pitting corrosion will take place. Contrarily, when a difference is large, pit growth rate starts to slow and if $d_{i,j}$ exceeds 255, further corrosion stops. Given $d_{i,j}$, cell-states are for each iteration of the CA updated according to the following transition function:

$$s(c_{i,j}, t+1) = \begin{cases} s(c_{i,j}, t) + Q(d_{i,j}, \gamma) & \text{if } \ 255 \geq d_{i,j} \geq v, \\ s(c_{i,j}, t) & \text{if } \ d_{i,j} < v or d_{i,j} > 255, \end{cases}$$

Note, that for $t = 0$, the difference $d_{i,j}$ cannot possibly exceed 255 as that is the maximum possible gray-scale value. However, as the image is augmented over multiple iterations, some cell states may gradually climb above 255 and, thus, yield differences above 255. Q is a function that determines the rate of corrosion and depends on $d_{i,j}$ and the parameter λ:

$$Q(d_{i,j}, \gamma) = \lfloor (255 - d_{i,j})\gamma \rfloor$$

From this function, it can be seen that a lower $d_{i,j}$ yields a bigger update to the cell. λ is a weight parameter that may adjust the rate of corrosion and mimics the resistance of the alloy surface given environmental conditions. In their study, da Silva et al. (2015) find the

optimal value of v and λ to be 5 and 0.1 using hyperparameter search. Due to our limited computational power we decide to adopt these parameter values for our experiments rather than rerunning an optimization process. As we wish to maintain the cell states and cell state-updates in discrete values, $\lfloor (255 - d_{i,j})\gamma \rfloor$ denotes the floor of $(255 - d_{i,j})\gamma$. For further elaboration on this rule we refer to the original paper (da Silva et al., 2015).

For each iteration of the CA-model, da Silva et al. (2015) record the cumulative mass of eroded product; the cumulative increase in cell states added to all cells in the grid. They use this time series of data points as an image descriptor for classifying the textures. In contrast to their method, we save each iteration of the augmented images and use the final iteration image as input to our classification model. Hence, we do not condense all the emergent patterns that have emanated from the CA augmentation into a single 1D time series but instead retain all the 784 cell values of the last iteration. We do, however, also track the cumulative mass of eroded alloy which we use to determine the optimal number of iterations to iterate the CA over.

Figure 5 shows the mean cumulative mass of corroded product for each class across time t. We decided on the optimal number of iterations to loop the CA over by qualitatively assessing when the development and inter-class divergence stagnated in the cumulative mass time series. For this rule, we deemed that after 30 iterations, further corrosion seems to stagnate for all classes and, hence, settled on 30 iterations.

The first plot in *figure 4* illustrates the simulated corrosion process applied to an image from the training set over 8 iterations of the CA. It is evident how pits start to form on the shoe.

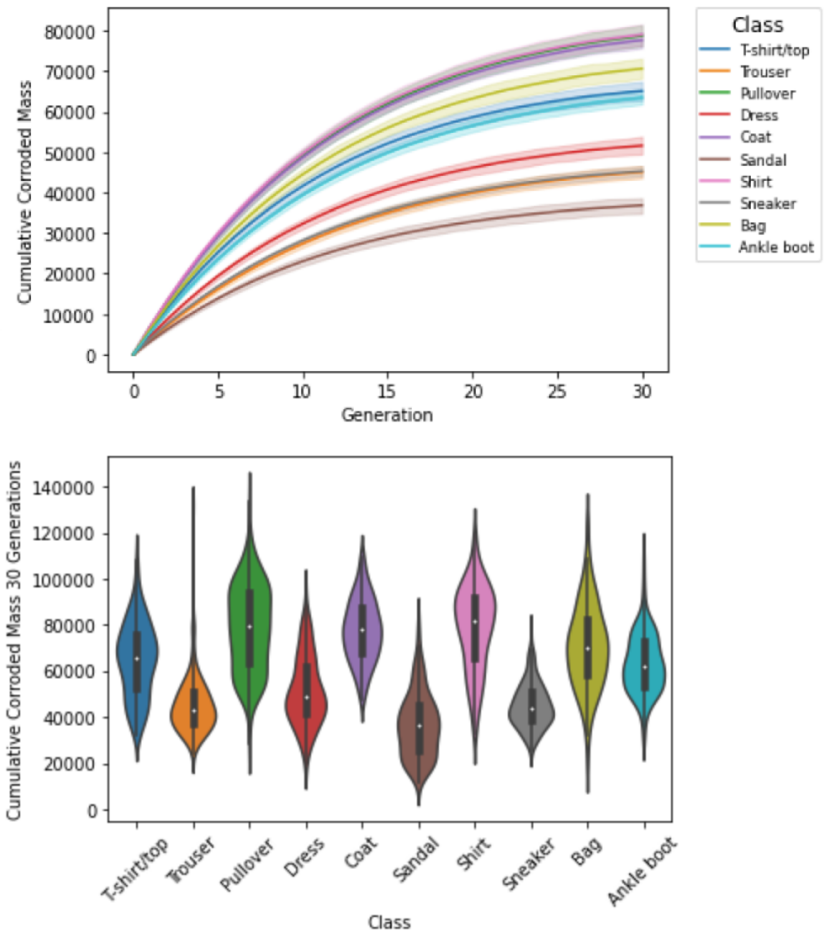

Figure 5: *The top plot displays the gradual augmentation of an example image when applying the cellular automaton designed simulate corrosion of a alloy over 30 time steps. The bottom plot depicts the mean cumulative corroded mass for each image category after 60 iterations*

2.3.2 Melting (JGD)

The second proposed CA rule is a novel algorithm designed to imitate the process of an object melting. For this rule, the fashion items in the data set are viewed as solid 3D objects that gradually transition into liquid form. Similar to the former rule, the original images are regarded as a 2D grid of cells G and at $t = 0$ cell states $s(c_{i,j}, 0)$ are defined to the discrete gray-scale value ranging from 0 to 255 for the corresponding pixel. All cells where $s(c_{i,j}) > 0$ are considered to represent solid material that has the potential to melt. The magnitude of the cell states represents the depth of the solid object at the given cell location

- similar to how these values represent the morphology of an alloy surface in the previous rule. Thus, higher cell-state values indicate a thicker object at that given location. The back of the object is regarded as flat and, consequently, areas with high cell-state values compared to neighboring areas represent ridges, peaks or ledges that rise above their local landscape. When the objects are placed vertically - as when viewed on a computer screen - material from these ridges, peaks and ledges would, given the nature of gravitational forces, melt down over the rest of the image in a downward direction. It is this melting phenomenon this rule aims to simulate.

Overhanging areas on the object that may drip off material are found by calculating the discrete image gradients for the y-direction (down) for every cell in the grid using a Sobel operator (Kanopoulos et al., 1988). The matrix below shows the 3x3 Sobel kernel which is convolved with the images to discretely estimate the vertical derivative of the image intensity function at each cell. Note how this kernel also analyses the Moore neighborhood of each cell in a similar fashion to the transition function for the corrosion CA.

$$G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

All negative gradients represent overhanging material. The absolute value of these gradients represents the amount of overhanging material; the more overhanging material, the larger the overhanging surface area and the more material will melt. *Figure 6* illustrates the discrete image gradients for an example image from the data set. The size of the white arrows on the zoomed-in subsection (image gradients in the y-direction) represents the amount of material that will melt down on the cells below the arrows. Hence, the update of each cell is dependent on the image gradient of the cell directly above. Following this logic, cell-states are for each iteration of the CA updated according to the following transition function:

$$s(c_{i,j}, t+1) = \begin{cases} s(c_{i,j}, t) - \lfloor \nabla f_y(c_{i,j}) \times \alpha \rfloor & \text{if } \nabla f_y(c_{i-1,j}) < 0, \\ s(c_{i,j}, t) & \text{if } \nabla f_y(c_{i-1,j}) \geq 0, \end{cases}$$

where $\nabla f_y(c_{i,j})$ is the image gradient of the cell directly above the cell being updated. α is a weighting parameter used to squeeze the cell update changes into a range that does not explode and is similar in magnitude to the updates applied in the corrosion rule. α is after qualitative investigation set to 500.

Similar to the procedure utilised for the corrosion rule, the cumulative mass of melted material (cumulative change of pixels) is tracked and plotted in order to determine the optimal number of iterations to run the CA. The mean trend line for each image class can be seen in *figure 7*. Interestingly, it seems for this rule that not only does the cumulative mass after n iterations differ across the categories, but the trajectory of the time series also differs substantially. After 60 iterations, the melting progression started to stagnate (objects have presumably melted to the bottom of the image) and the disparity between classes stabilised. Hence, 60 iterations was deemed the optimal number.

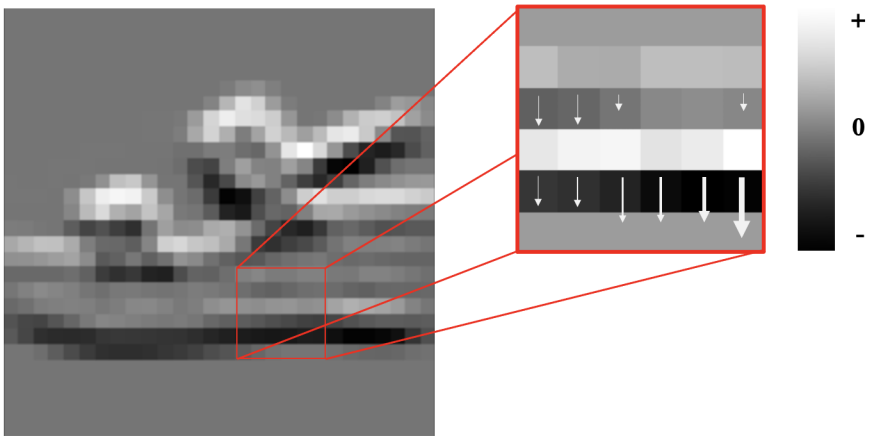

Figure 6: *The discrete image gradient in the y-dimension calculated using a Sobel kernel regulate the rate of melting at that discrete point (arrows represent the same as the pixel colour)*

Note that this rule is overly simplistic. For instance, despite material gradually melting, no material is subtracted from the image.

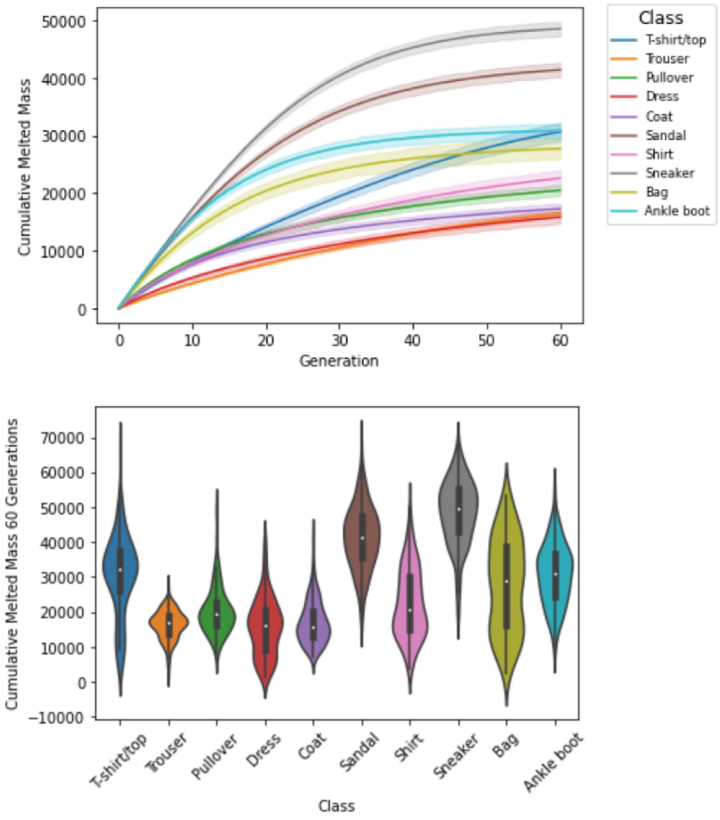

Figure 7: *The top plot displays the gradual augmentation of an example image when applying the cellular automaton designed simulate a melting process over 60 iterations. The bottom plot depicts the mean cumulative melted mass for each image category after 60 iterations*

2.3.3 Game of Life (JKH)

The final proposed CA-rule is an adaptation of what is known as *the Game of Life* (GoL), perhaps the most studied and popularised CA rule set (Conway, 1970). Contrary to the other rules that treat the images as coherent objects upon which to apply a transformation, this rule is more abstract. The game is a computational model inspired by biological processes and was originally devised in order to investigate the evolutionary processes of ecological communities (Caballero et al., 2016; Contreras-Reyes, 2021; Conway, 1970). The original GoL is initialized with an infinitely large two-dimensional grid where each cell can inhabit one of two possible states, namely dead or alive. The seed of the system is traditionally composed of an initial distribution of dead and living cells that is randomly generated. The

rule set consists of the following three rules (Caballero et al., 2016; Conway, 1970; Roberts, 2020):

1. **Death rule:** Any living cell surrounded by less than two live cells in the Moore neighborhood dies.
2. **Survival rule:** Any living cell surrounded by two or three live cells in the Moore neighborhood continues to live.
3. **Birth rule:** Any dead cell with three live cells in the Moore neighborhood comes to life.

Followingly, in each iteration, some cells live while others die leading to an evolving system that changes between each iteration. To create augmented versions of the Fashion-MNIST features using the logic behind GoL, we limited the grid to the 28x28 grid of cells G (30x30 with padding) and accommodated for the binary nature of the GoL process by first binarizing the pixels in each image, $I(i, j)$:

$$s(c_{i,j}, t = 0) = \begin{cases} 0 & \text{if } \ I(j, i) = 0, \\ 1 & \text{if } \ I(j, i) > 0, \end{cases}$$

Pixel values holding a gray-scale value exceeding 0 were thus changed to 1 (to represent live cells), while pixel values of 0 represented dead cells. The binarized images were then used as seed images for $t = 0$ in the GoL CA process with cells being iteratively updated by the following transition (according to the rules described above):

$$s(c_{i,j}, t+1) = \begin{cases} 0 & \text{if } \ s(c_{i,j}, t) = 0 \ \ and \ \ \sum \tilde{s}(N(c_{i,j}), t) \ \neq \ 3, \\ 0 & \text{if } \ s(c_{i,j}, t) = 1 \ \ and \ \ 2 > \ \sum \tilde{s}(N(c_{i,j}), t) \ > \ 3, \\ 1 & \text{if } \ s(c_{i,j}, t) = 0 \ \ and \ \ \sum \tilde{s}(N(c_{i,j}), t) \ = \ 3, \\ 1 & \text{if } \ s(c_{i,j}, t) = 1 \ \ and \ 2 \ > \ \sum \tilde{s}(N(c_{i,j}), t) \ < \ 3, \end{cases}$$

The CA was run for multiple time steps and the cumulative change in the number of living cells compared to the seed states was tracked. The mean time series for each image class for this measure is shown in *figure 8*. The shown time series appear - as expected - much more stochastic than the previous rules. GoL is known for quickly giving rise to complex patterns from only marginally varying starting conditions. After applying the rule for 12 iterations, the CA is able to capture some stochastic developments while maintaining stable variation between the classes. Hence, this was chosen as the number of iterations to run this rule for.

As with the other rules, the resulting 28 x 28 augmented images after 12 iterations of the CA were extracted and utilised in the classification framework.

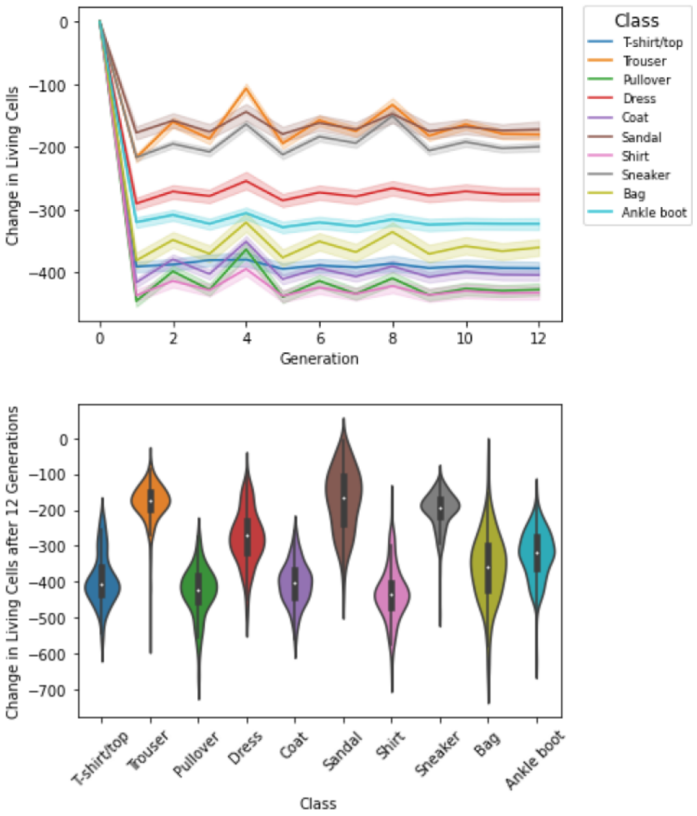

Figure 8: *The top plot displays the gradual augmentation of an example image when summing the number of alive cells when applying the Game of Life over 12 iterations. The bottom plot depicts the distribution of changes in living cells for each image category after 12 iterations.*

2.4 Classification framework (ETJ)

The employment of the CA algorithms resulted in four disparate versions of the Fashion-MNIST images; the original data set and three augmented versions extracted after the last iteration of the three respective rules.

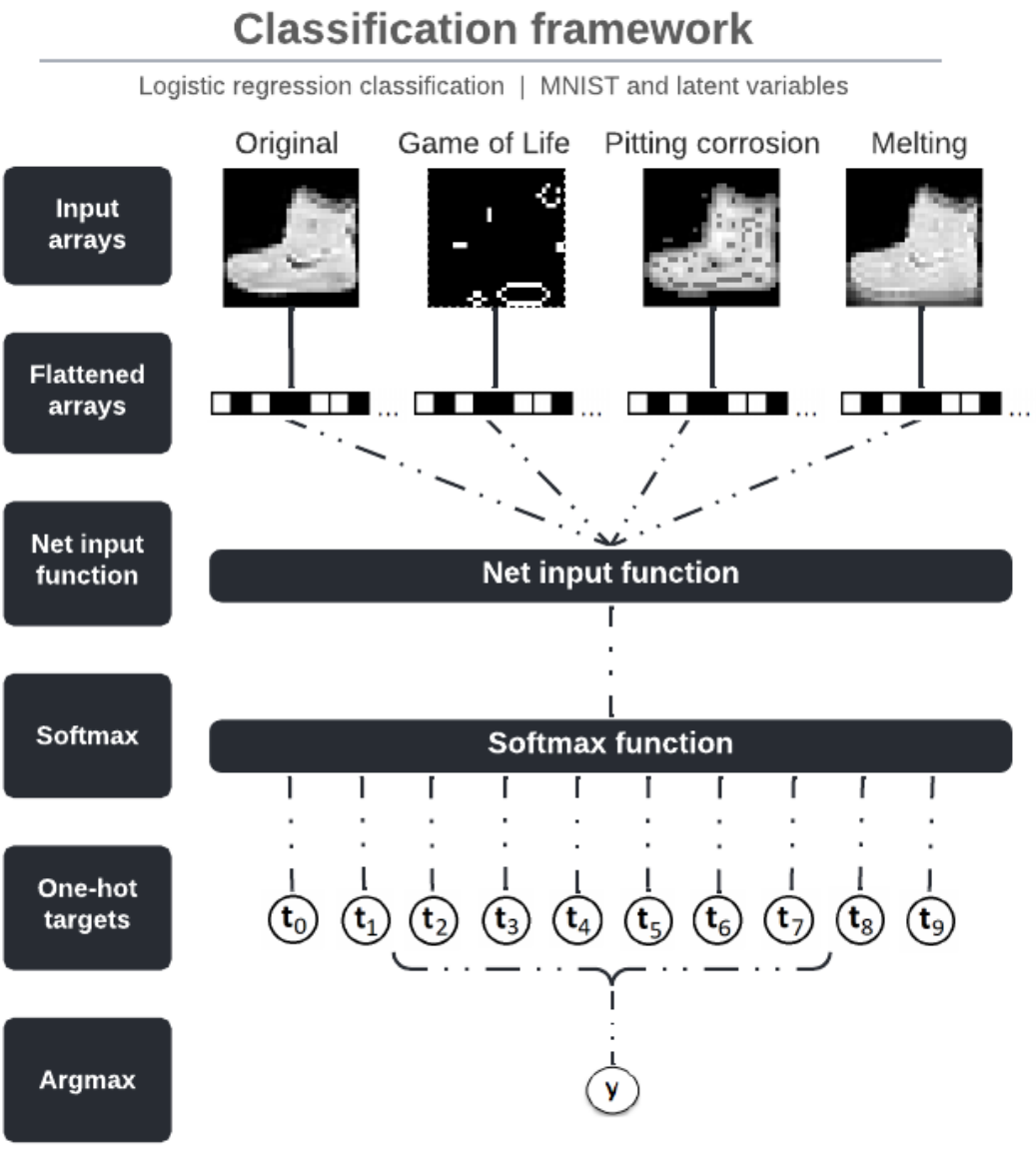

Figure 9: *Simplified representation of the multinomial classification framework. This representation shows model 4 - the full model including all 3 latent feature sets. Please note that the models which only included some feature sets (e.g. model 1, using the original and GoL versions) did not include the remaining input arrays (e.g. pitting corrosion and melting).*

As the aim of the study was to provide a proof-of-concept, investigating whether CA data augmentation could be used to enhance discriminatory features in images, we purposefully ran classification using a relatively simple framework. We trained and evaluated 5 multinomial logistic regression classifiers to classify the images for the analysis. All models were trained using the *sag*-solver and *l2 regularization* with a C-hyperparameter value of 1, using the scikit-learn framework (Pedregosa et al., 2011). All models were trained and tested on the same subset of the Fashion-MNIST data to allow for a lower computational load, using 2.000 images for training and 1.000 unseen images for testing. The number of maximum iterations need for the models to converge was found by evaluating runs using a validation set of 1.000 images. The baseline model used only the flattened Fashion-MNIST images as input. For models 1-3, the flattened original images were concatenated with one set of augmented images from each of the respective CA rule sets. Model 4 also used flattened arrays but combined the original data with the augmented images from all 3 CA rules. The model computed the outcome probabilities for each class by squeezing the output layer through a softmax function. The final predictions were the argmax of the probability vector. For a graphic depiction of the classification framework, see *figure 9*.

The following list provides an overview of the models:

1. **Model 0 (baseline):** Logistic regression model trained and evaluated on flattened Fashion-MNIST data.

2. **Model 1:** Logistic regression model trained and evaluated on flattened Fashion-MNIST images concatenated with corresponding flattened output images processed with the Game of Life CA-rule.

3. **Model 2:** Logistic regression model trained and evaluated on flattened Fashion-MNIST images concatenated with corresponding flattened output images processed with the pitting corrosion CA-rule.

4. **Model 3:** Logistic regression model trained and evaluated on flattened Fashion-MNIST images concatenated with corresponding flattened output images processed with the melting CA-rule.

5. **Model 4:** Logistic regression model trained and evaluated on flattened Fashion-MNIST images concatenated with each of the corresponding flattened output images processed with each rule CA-rule, respectively.

2.5 Performance metrics (JGD)

Whenever researchers in machine learning have to report their results they are faced with the challenge of choosing the best fitting evaluation metric with respect to the particular study at hand. In spite of this being a critical matter in machine learning and data science, there is yet to emerge a unified consensus within the field. The most popular metrics are variations of *accuracy* and *F1-score*, however, these measures are at risk of fallaciously inflating model performances when applied to imbalanced data sets (Chicco & Jurman, 2020; Shmueli, 2020). As the Fashion-MNIST data set is balanced it is not directly problematic to use these measures, but in the pursuit of good practice, we have chosen to use measures that are robust to imbalances.

2.5.1 Matthews correlation coefficient (JKH)

The Matthews Correlation Coefficient (MCC), is a more robust alternative to these aforementioned evaluation metrics since it will only yield a high score if the model classifies convincingly across all dimensions of classification categories (true negatives, false negatives, true positives, and false positives) with respect to potential skew between classes (Chicco & Jurman, 2020). Followingly, the MCC produces a more trustworthy and informative score compared to F1-score and accuracy, which has led the MCC to be proposed as a new Gold-standard evaluation metric within the field (Chicco & Jurman, 2020; Shmueli, 2020).

In cases of multinomial classification such as ours, the MCC can be mathematically defined the following way in terms of a confusion matrix C for a data set consisting of K classes (Pedregosa et al., 2011):

$$MCC = \frac{c \times s - \sum_k^K p_k \times t_k}{\sqrt{(s^2 - \sum_k^K p_k^2) \times (s^2 - \sum_k^K t_k^2)}}$$

Where,

- $t_k = \sum_i^K C_{ik}$ corresponds the amount of times class k actually occurs in the data set.
- $p_k = \sum_i^K C_{ki}$ corresponds the amount of times class k was predicted by the model.
- $c = \sum_k^K C_{kk}$ corresponds the total amount of samples predicted correctly by the model.
- $s = \sum_i^K \sum_j^K C_{ij}$ the total amount of samples in the data set.

For multinomial MCC classification, the output score will range between -1 and 1 where 1 represents perfect discriminatory power (Pedregosa et al., 2011).

2.5.2 Balanced accuracy (ETJ)

To account for the infrequency of reporting performance using the MCC, we also include the balanced accuracy to ease interpretation of the results as accuracy is frequently used within the field.

$$Balanced\ accuracy = \frac{1}{K} \sum_{k=1}^{M} \frac{r_k}{t_k}$$

Where,

- K corresponds to the number of classes

- t_k corresponds to the number of samples belonging to class k
- r_k corresponds to the number of samples correctly predicted to belonging to class k

3 Results (JKH)

Table 1 shows the performances of the five classification models.

Model	Balanced accuracy	Matthews Correlation Coefficient
Model 0 (baseline)	0.788	0.762
Model 1 (orig. & GoL)	0.789	0.768
Model 2 (orig. & corrosion)	0.805	0.783
Model 3 (orig. & melt)	0.803	0.781
Model 4 (orig. & all CA-rules)	**0.826**	**0.807**

Table 1: *Performance of classification models.*

4 Discussion

4.1 General Performance (JKH)

As can be observed in *table 1*, we have, in line with our scope, managed to successfully improve the performance of image classification on the Fashion-MNIST data set through data augmentation using CA-algorithms based on natural transformation processes. *Model 4*, which is a combination of the original data and augmentations derived from all three CA-rules, achieves an MCC-score of 0.807. In contrast, the baseline model only reaches 0.762, corresponding to an increase of 0.045. When evaluating the remaining models individually compared to the baseline, we see that applying the corrosion-, GoL- and melting algorithms all improve performance. Although the results should be regarded with a degree of caution, they indicate that the CA-augmentations have successfully spawned emergent patterns that hold some discriminatory power beyond those of the original visual images.

Furthermore, as a way to further investigate the prospect of using our method in classification tasks facing data scarcity, we carried out a post-hoc exploratory analysis where we compared the performance of model 4 trained on 2,000 samples against a baseline model (0) trained on 40,000 samples. Both models were evaluated on the same subset of the testing data consisting of 1,000 images. For results see *table 2*.

Model	**Train samples**	**Balanced accuracy**	**Matthews Correlation Coefficient**
Model 0 (baseline)	40.000	0.821	0.806
Model 4 (original & all CA)	2.000	**0.826**	**0.807**

Table 2: *Performance table showcasing the performance differences between a model fitted on a small subset with data augmentation and a model trained on a large subset on the original data set.*

Impressively, we observe that even when the raw baseline model is trained on twenty-fold as much data, *model 4* still achieves superior results – albeit, admittedly, only by a small margin. Moreover, it is worth noting that more data yields diminishing returns and that the level of complexity that the architecture of logistic regression holds, means that the model is likely fundamentally incapable of capturing the full range of systematic patterns in the data, no matter the amount of input data; hence, performance is likely to have stagnated before 40.000 training examples. Regardless, we argue that this illustrates the large potential for using CA-processing in fields where data scarcity is a frequent problem as in, e.g., medicine where health data from patients is challenging to collect in large amounts due to patient data-protection laws and other ethical concerns (Bansal et al., 2020; Maier, 2020).

Given the observed performance leap from *model 0* to *model 4*, it is necessary to delve into the outputs of the CA-augmentations to explore some of the reasons behind the apparent gains. *Figure 10* shows an example of an image from the test set which is incorrectly classified by *model 0* but correctly classified by *model 4*. When an image of some footwear is compressed into 28x28 pixels, it may prove highly challenging to establish whether it is

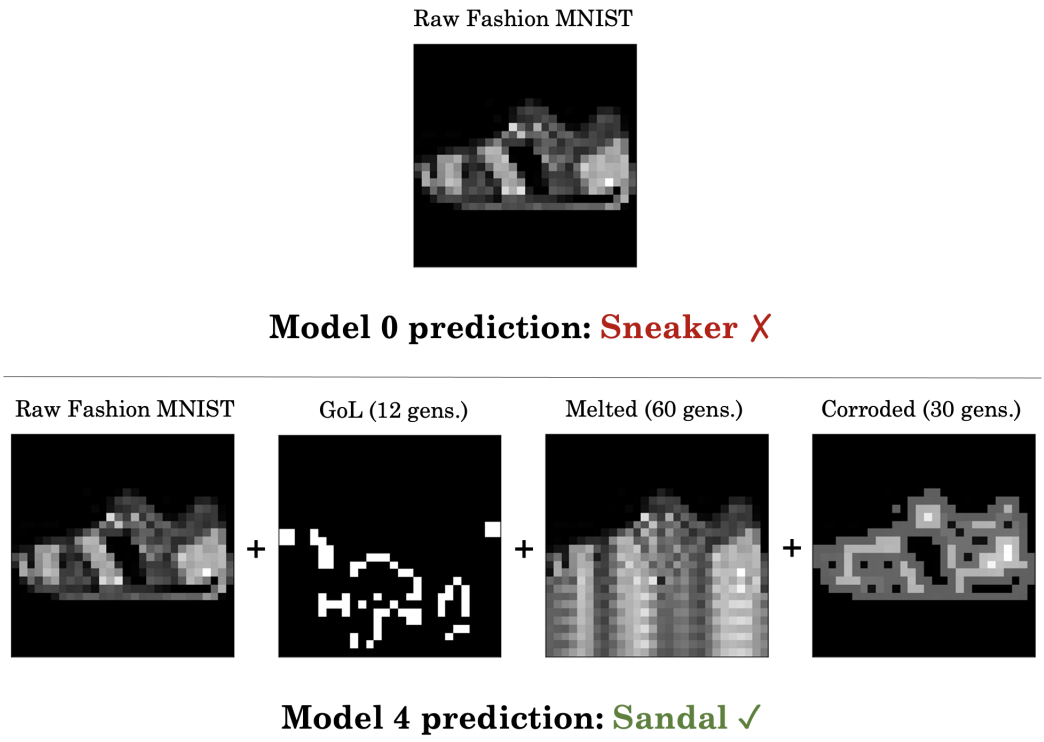

Figure 10: *An example of the enhanced classification performance of Model 4 compared to Model 0. Model 0 misclassifies the shown image as a sneaker. Model 4, using all the augmented images, correctly identifies the image as a sandal.*

a sandal or a sneaker based only on the visual patterns. However, an average sandal with straps presumably has more details and pixel-value disparity than sneakers that are usually more uniform across the surface. Viewed from the perspective of the corrosion CA, the higher degree of pixel variability in the sandals will be interpreted as a more uneven surface. Hence when sneakers and sandals are augmented by this CA, their subtle visual discrepancy may be largely enhanced through gradual build-up of increasingly more rust and pits on the sandals. We can only speculate that this is one of the contributions to the true classification of the sandal by *model 4* in *figure 10*. To take another example, it may be difficult for a computer vision algorithm to differentiate between t-shirts and pullovers; visually it may only depend on the length of the sleeves. When looking at the melted material added by the CA after 60 iterations to the shirt in *figure 11*, it appears that for that given image, the melting augmentation essentially merely represents the length of the sleeve (or, more

precisely, length from the sleeve to the bottom of the image), which, we argue, could be one of the explanations behind why the processing makes it easier for the model to classify the images.

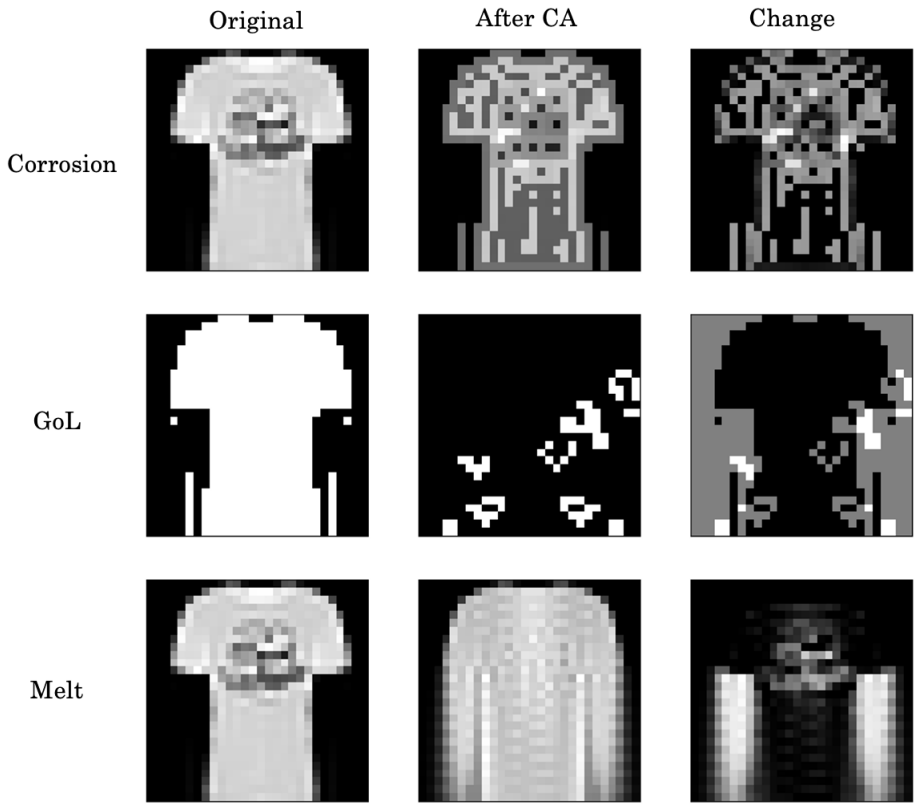

Figure 11: *Augmentations results. Left column: Original images (shows a binarised version of the original image for the GoL row). Middle column: Resulting augmented images after the application of the CA models for the specified number of iterations. Right column: Absolute change of original image (left image - middle image). E.g. top right corner displays the mass and distribution of corroded alloy.*

4.2 Future perspectives and improvements (ETJ)

Despite the promising results, there are numerous reasons to be cautious and tonnes of directions that remain to be explored before the value of the method should be truly celebrated.

Firstly, had more computational power been available, it would have been better practice to conduct a more exhaustive, systematic and intelligent search for the optimal hyperparam-

eters in the transition rules and classification models using the validation set - especially as this project relies on very few related studies. Furthermore, it would have been fruitful to compare the results directly against the impact of other more traditional image augmentation paradigms. Likewise, it would be interesting to investigate how well the models perform on out-of-sample (OOS) images that have been altered by adding noise or rotation. Are the models trained on the augmented images perhaps also more robust to OOS data? Are they also better at classifying e.g. dirty sneakers than the baseline model?

Secondly, as the aim of our study was intended as a proof-of-concept, we implemented relatively simple CA and only tested it on multinomial logistic modeling. We have, thus, not exhaustively captured all the potential the CA data augmentation method may hold. Furthermore, we have most certainly not exhausted the classification model-space to investigate whether the observed performance increases could be either further improved or eaten up by other model frameworks and other ways to handle the generated features. One favorable aspect worth investigating, in regard to this, would be to try and use the entire time series of augmented images generated at each time step of the CA process. Inspecting the temporal evolution of of the images during augmentation seems to provide evidence that the gradual changes also holds information. For instance, in *figure 7*, it can be seen that the progression of the cumulative melted mass over iterations is different across image types; the average time series for t-shirts has very different curve than all other classes. The same goes for t-shirts when looking at cumulative change in number of living cells during GoL-augmentation in *figure 8*. The temporal element is largely lost given an analysis such as the one we conducted. Akin to methods used in medical imaging classification (Singh et al., 2020), using a 3D convolution neural network classifier would have allowed for using the entire time series evolution in a 3D input with dimensions 28 x 28 x number of the iterations. Moreover, using such a classifier would also allow for better modeling of the spatial relationship between pixel values. Although the baseline performance without the use of CA data augmentation would also increase given the use of CNN compared to LR, it could also be possible that the CA

data augmented data performance would increase to an even larger extent as CA evolution holds an extra spatial dimension.

The study also begs the question - why apply corrosion to an image of a shirt? Why melt a shoe? Although one of the implemented CA rules is rooted in previous literature (da Silva et al., 2015), the specific rule sets, generally, appear somewhat arbitrary. This highlights a general problem with CA; specifying rules is a tedious process and must often be carefully designed for each specific domain (Silva & Clarke, 2002; Soares-Filho et al., 2002). However, the results in this paper also to some extent answer the questions of why it makes sense to corrode a shirt and melt a shoe? Because it works. If these rules systematically manage to draw images further apart, how many un-tested, un-designed rules that mimic transformation processes would also work? It is possible that a large plethora of CA could be implemented with various degrees of success and that other CA algorithms would even work considerably better than those tested in this study. Nonetheless, the questions do highlight a deep-rooted issue with the proposed framework. In an era of big data and huge deep learning frameworks, handcrafting and hard coding domain-specific does not exactly seem to be the way forward. A more timely and smarter way to specify CA rules could be to move into the continuous space like Randazzo et al. (2020) and Mordvintsev et al. (2020) do for their self-classifying neural cellular automata models. Continuous CA rules may be optimized more akin to convolutional kernels in CNNs and, thus, seems to be the area towards which most time and effort for future research should be channelled. This approach would better identify the potentials and limits of using CA data augmentation to increase the discriminatory information available in image classification tasks as well as investigate the prospects of uncovering more generalizable rules for image augmentation.

A potential limitation of the method arises, however, when talking about the use of the entire augmentation time series, more complex models and continuous CA; computational load. The increased computational load could potentially be a drawback to using this method

as the data explodes in size by adding an additional dimension to the data, adding more iterations of the CA, and given the resolution of images in real-life tasks that are presumably much larger than 28x28 pixels. Training weights in a convolutional kernel is notoriously cheap and efficient (LeCun et al., 2015). This task would presumably be much harder when training a kernel that is designed to iteratively loop over the data - finding the optimal number of iterations would be a challenge.

Given the prevalence and use of other data augmentation methods, it is worth considering whether future research ought to concern itself with exploring CA data augmentation for image classification further. Since the performance of sophisticated methods such as CNNs may yield extremely high performances with enough quality data, one might argue that other measures, such as increasing the size of data sets or model architectures, ought to take prevalence over data augmentation methods such as the one described in this paper. However, some domains applying ML methods may have a scarcity of data such as medicine or marketing, where it is essential that one maximizes the utility of the data (Bansal et al., 2020). However, even for this purpose, it is important to note that other data augmentation techniques exist and have been used with great success, e.g. *color space transformations*, *random cropping* and *kernel filters*, to mention a few (Krizhevsky et al., 2012; Shorten & Khoshgoftaar, 2019; Zhong et al., 2020).

Although we can only remain speculative about the usefulness of actual practical implementations of CA data augmentation, these preliminary results are exciting. Conducting further research following these lines would be the only way of uncovering the potentials and limitations of the methods.

5 Conclusion (ETJ)

Machine learning modeling on image classification tasks faced with data scarcity has been shown to benefit from the application of data augmentation methods. In an attempt to further advance this field, this paper sought to provide a proof-of-concept analysis to investigate whether a novel approach using discrete CA may be used in the future application as a data augmentation tool that enhances discriminatory features in classification tasks. Thus, we implemented three CA rule sets with offset in naturally occurring transformation processes to iteratively augment the Fashion-MNIST data set. After including the resulting augmented images in a range of multinomial logistic regression models, results showed that the inclusion of the augmented images increases performance. Thus, our study suggests that the application of CA may be successful in enhancing the already distinguishable patterns in the data or spawning new emergent patterns. This is further explored in a qualitative error analysis where we highlight examples of how the corrosion- and melting algorithms might, respectively, be responsible for enhancing key discriminatory features between edge cases in sandals versus sneakers and t-shirts versus pullovers. We recommend that further research be conducted to outline the full potential of the method. The use of multidimensional CNNs may allow for taking advantage of the information that potentially lies in the temporal dimension of the iterative steps in the augmentation process, while the use of algorithms that optimize continuous CA rule sets for classification tasks using backpropagation could further improve the efficiency and generalizability of CA-based augmentation algorithms.

References

Baldominos, A., Saez, Y., & Isasi, P. (2019). A survey of handwritten character recognition with mnist and emnist [Publisher: Multidisciplinary Digital Publishing Institute]. *Applied Sciences*, *9*(15), 3169.

Bansal, M. A., Sharma, D. R., & Kathuria, D. M. (2020). A systematic review on data scarcity problem in deep learning: Solution and applications [Publisher: ACM New York, NY]. *ACM Computing Surveys (CSUR)*.

Berec, L. (2002). Techniques of spatially explicit individual-based models: Construction, simulation, and mean-field analysis [Publisher: Elsevier]. *Ecological modelling*, *150*(1), 55–81.

Bi, Y., Xue, B., & Zhang, M. (2021). *Genetic programming for image classification: An automated approach to feature learning* (Vol. 24). Springer Nature.

Caballero, L., Hodge, B., & Hernandez, S. (2016). Conway's "game of life" and the epigenetic principle. *Frontiers in Cellular and Infection Microbiology*, *6*. Retrieved May 24, 2022, from https://www.frontiersin.org/article/10.3389/fcimb.2016.00057

Caicedo, J. C., Cruz, A., & Gonzalez, F. A. (2009). Histopathology image classification using bag of features and kernel functions. *Conference on Artificial Intelligence in Medicine in Europe*, 126–135.

Chapelle, O., Haffner, P., & Vapnik, V. N. (1999). Support vector machines for histogram-based image classification [Publisher: IEEE]. *IEEE transactions on Neural Networks*, *10*(5), 1055–1064.

Chicco, D., & Jurman, G. (2020). The advantages of the matthews correlation coefficient (MCC) over f1 score and accuracy in binary classification evaluation. *BMC Genomics*, *21*(1), 6. https://doi.org/10.1186/s12864-019-6413-7

Ciregan, D., Meier, U., & Schmidhuber, J. (2012). Multi-column deep neural networks for image classification. *2012 IEEE conference on computer vision and pattern recognition*, 3642–3649.

Contreras-Reyes, J. E. (2021). Lerch distribution based on maximum nonsymmetric entropy principle: Application to conway's game of life cellular automaton. *Chaos, Solitons & Fractals*, *151*, 111272. https://doi.org/10.1016/j.chaos.2021.111272

Conway, J. (1970). The game of life. *Scientific American*, *223*(4), 4.

Das, S., Jain, L., & Das, A. (2018). Deep learning for military image captioning. *2018 21st International Conference on Information Fusion (FUSION)*, 2165–2171.

da Silva, N. R., Van der Weeën, P., De Baets, B., & Bruno, O. M. (2015). Improved texture image classification through the use of a corrosion-inspired cellular automaton [Publisher: Elsevier]. *Neurocomputing*, *149*, 1560–1572.

Dawkins, R. (2017). *Science in the soul: Selected writings of a passionate rationalist*. Random House Trade Paperbacks.

Draelos, R. (2020, January 30). *The history of convolutional neural networks* [Medium]. Retrieved May 23, 2022, from https://towardsdatascience.com/a-short-history-of-convolutional-neural-networks-7032e241c483

Florindo, J. B., & Metze, K. (2021). A cellular automata approach to local patterns for texture recognition. *Expert Systems with Applications*, *179*, 115027. https://doi.org/10.1016/j.eswa.2021.115027

Fu, H., Niu, Z., Zhang, C., Ma, J., & Chen, J. (2016). Visual cortex inspired CNN model for feature construction in text analysis [Publisher: Frontiers]. *Frontiers in computational neuroscience*, *10*, 64.

Hersey, I. (1968). Textures: A photographic album for artists and designers by phil brodatz [Publisher: The MIT Press]. *Leonardo*, *1*(1), 91–92.

Huang, Y., Huang, K., Yu, Y., & Tan, T. (2011). Salient coding for image classification. *CVPR 2011*, 1753–1760.

Hubel, D. H., & Wiesel, T. N. (1959). Receptive fields of single neurones in the cat's striate cortex [Publisher: Wiley-Blackwell]. *The Journal of physiology*, *148*(3), 574.

Kalpande, G. (2018, December 13). *Biological inspiration of convolutional neural network (CNN)* [Medium]. Retrieved May 23, 2022, from https://medium.com/@gopalkalpande/biological-inspiration-of-convolutional-neural-network-cnn-9419668898ac

Kanopoulos, N., Vasanthavada, N., & Baker, R. L. (1988). Design of an image edge detection filter using the sobel operator [Publisher: IEEE]. *IEEE Journal of solid-state circuits*, *23*(2), 358–367.

Karafyllidis, I., Ioannidis, A., Thanailakis, A., & Tsalides, P. (1997). Geometrical shape recognition using a cellular automaton architecture and its VLSI implementation. *Real-Time Imaging*, *3*(4), 243–254. https://doi.org/10.1006/rtim.1997.0076

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, *25*.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning [Publisher: Nature Publishing Group]. *nature*, *521*(7553), 436–444.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition [Publisher: Ieee]. *Proceedings of the IEEE*, *86*(11), 2278–2324.

Lin, L., Zheng, X., Liu, B., Chen, W., & Xiao, Y. (2021). A latent variable augmentation method for image categorization with insufficient training samples [Publisher: ACM New York, NY]. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, *16*(1), 1–35.

Maier, A. (2020, June 11). *Will we ever solve the shortage of data in medical applications?* [Medium]. Retrieved May 29, 2022, from https://towardsdatascience.com/will-we-ever-solve-the-shortage-of-data-in-medical-applications-70da163e2c2d

Malki, B., & Baroux, B. (2005). Computer simulation of the corrosion pit growth [Publisher: Elsevier]. *Corrosion Science*, *47*(1), 171–182.

Meel, V. (2021, September 24). *Image recognition: The basics and use cases (2022 guide)* [Viso.ai]. Retrieved May 30, 2022, from https://viso.ai/computer-vision/image-recognition/

Mordvintsev, A., Randazzo, E., Niklasson, E., & Levin, M. (2020). Growing neural cellular automata. *Distill*, *5*(2), e23.

Nanni, L., Paci, M., Brahnam, S., & Lumini, A. (2021). Comparison of different image data augmentation approaches [Publisher: Multidisciplinary Digital Publishing Institute]. *Journal of Imaging*, *7*(12), 254.

Nath, S. S., Mishra, G., Kar, J., Chakraborty, S., & Dey, N. (2014). A survey of image classification methods and techniques. *2014 International conference on control, instrumentation, communication and computational technologies (ICCICCT)*, 554–557.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Preston Jr, K., & Duff, M. J. (2013). *Modern cellular automata: Theory and applications*. Springer Science & Business Media.

Randazzo, E., Mordvintsev, A., Niklasson, E., Levin, M., & Greydanus, S. (2020). Self-classifying mnist digits. *Distill*, *5*(8), e00027–002.

Ripon, S., Golam Sarowar, M., Qasim, F., & Cynthia, S. T. (2020). An efficient classification of tuberous sclerosis disease using nature inspired PSO and ACO based optimized neural network. In M. Rout, J. K. Rout, & H. Das (Eds.), *Nature inspired computing for data science* (pp. 1–28). Springer International Publishing. https://doi.org/10.1007/978-3-030-33820-6_1

Roberts, S. (2020). The lasting lessons of john conway's game of life. *The New York Times*. Retrieved May 24, 2022, from https://www.nytimes.com/2020/12/28/science/math-conway-game-of-life.html

Rodríguez del Nozal, A., Tapia, A., Alvarado-Barrios, L., & Reina, D. G. (2020). Application of genetic algorithms for unit commitment and economic dispatch problems in microgrids. In M. Rout, J. K. Rout, & H. Das (Eds.), *Nature inspired computing for data science* (pp. 139–167). Springer International Publishing. https://doi.org/10.1007/978-3-030-33820-6_6

Rosin, P. L. (2006). Training cellular automata for image processing [Publisher: IEEE]. *IEEE transactions on image processing*, *15*(7), 2076–2087.

Shmueli, B. (2020, May 20). *Matthews correlation coefficient is the best classification metric you've never heard of* [Medium]. Retrieved May 29, 2022, from https://towardsdatascience.com/the-best-classification-metric-youve-never-heard-of-the-matthews-correlation-coefficient-3bf50a2f3e9a

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*, *6*(1), 60. https://doi.org/10.1186/s40537-019-0197-0

Silva, E. A., & Clarke, K. C. (2002). Calibration of the SLEUTH urban growth model for lisbon and porto, portugal [Publisher: Elsevier]. *Computers, environment and urban systems*, *26*(6), 525–552.

Singh, S. P., Wang, L., Gupta, S., Goli, H., Padmanabhan, P., & Gulyás, B. (2020). 3d deep learning on medical images: A review [Publisher: Multidisciplinary Digital Publishing Institute]. *Sensors*, *20*(18), 5097.

Soares-Filho, B. S., Cerqueira, G. C., & Pennachin, C. L. (2002). DINAMICA—a stochastic cellular automata model designed to simulate the landscape dynamics in an amazonian colonization frontier [Publisher: Elsevier]. *Ecological modelling*, *154*(3), 217–235.

Vermeire, T., Brughmans, D., Goethals, S., de Oliveira, R. M. B., & Martens, D. (2022). Explainable image classification with evidence counterfactual. *Pattern Analysis and Applications*, *25*(2), 315–335. https://doi.org/10.1007/s10044-021-01055-y

Wan, L., Zeiler, M., Zhang, S., Le Cun, Y., & Fergus, R. (2013). Regularization of neural networks using dropconnect. *International conference on machine learning*, 1058–1066.

Wolfram, S. (1983). Statistical mechanics of cellular automata [Publisher: APS]. *Reviews of modern physics*, *55*(3), 601.

Xiao, H., Rasul, K., & Vollgraf, R. (2017). Fashion-mnist: A novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*.

Zhong, Z., Zheng, L., Kang, G., Li, S., & Yang, Y. (2020). Random erasing data augmentation [Issue: 07]. *Proceedings of the AAAI conference on artificial intelligence*, *34*, 13001–13008.