Proceedings of the 2019 Winter Simulation Conference
N. Mustafee, K.-H.G. Bae, S. Lazarova-Molnar, M. Rabe, C. Szabo, P. Haas, and Y.-J. Son, eds.

SOLVING CHALLENGES AT THE INTERFACE OF SIMULATION AND BIG DATA USING MACHINE LEARNING

Philippe J. Giabbanelli

Department of Computer Science and Software Engineering
Miami University
510 E. High St.
Oxford, OH 45056, USA

ABSTRACT

Modeling & Simulation (M&S) and Machine Learning (ML) have been used separately for decades. They can also straightforwardly be employed in the same study by contrasting the results of a theory-driven M&S model with the most accurate data-driven ML model. In this paper, we propose a paradigm shift from seeing ML and M&S as two independent activities to identifying how their integration can solve challenges that emerge in a big data context. Since several works have already examined this interaction for conceptual modeling or model building (e.g., creating components with ML and embedding them in the M&S model), our analysis is devoted on three relatively under-studied stages: calibrating a simulation model using ML, dealing with the issues of large search space by employing ML for experimentation, and identifying the right visualizations of model output by applying ML to characteristics of the output or actions of the users.

1 INTRODUCTION

Machine Learning (ML) and Modeling & Simulation (M&S) have been around for several decades. Given the explosion of research in data science, a recurring question in the recent years is to (re-)assess the role that ML and M&S can play in the big data ecosystem. At a high-level, these approaches proceed in very similar ways: they derive a model from some of the evidence (the 'training set' of ML or the 'calibration step' of M&S) and use the remainder to evaluate the model (the 'testing set' of ML or the 'validation step' of M&S). A complementary use occurs when a project builds a model using each method, and assesses whether results are consistent regardless of the model building method. Although Voinov *et al.* have reported that there are relatively few projects "where more than one method has been tried for the same problem within the same project" (Voinov et al. 2018), a complementary use is an easy starting point as each method can be followed independently without calling for technical innovations.

Alternatively, M&S and ML may be used for different *questions* within the same *project*. For instance, a purely data-driven ML model is implicitly built on the specific context in which the data was collected. Such a model can thus provide a baseline by accurately predicting what will happen if the context remains unchanged. In contrast, many M&S models employ a mix of assumptions and data which allows stakeholders to deal with nonstationary distributions and/or test new scenarios (i.e. 'what-if questions') that predict what will happen after changing the context. Consider eating behaviors as a sample application: we can use machine learning to accurately determine what individuals currently eat (Rosso and Giabbanelli 2018), and we can make projections over 10 years by accounting for aging and secular trends in nutrition. We can also use modeling and simulation (e.g., agent-based models) to determine what individuals will eat in 10 years *if* we implement policies such as increasing taxes on unhealthy food items, subsidizing healthier options, or influencing social norms (Khademi et al. 2018). In this situation, the effect of the policies on key outcomes (e.g., prevalence of hypertension and diabetes) may be evaluated by contrasting the simulations with the most accurate projection from machine learning (e.g., through a 'difference in differences' approach). Once again, this approach does not require technical innovations as the M&S and ML model-building processes can take place independently.

Using the lens of big data, this paper promotes a paradigm shift from seeing M&S and ML as two independent model-building activities to identifying how a closer integration can help tackle specific challenges. Such integration requires a detailed understanding of both M&S and ML, for instance to use the right ML solution *given* the characteristics of data generated by M&S, or to study how a simulation model can be partially re-designed through ML. This paradigm shift is applicable to address numerous challenges. The challenge may stem from the large datasets *collected* for a project: we may have tracked thousands of individual-level factors over a large population, and we would need to extract the most relevant factors for a behavior of interest when creating an agent-based model of this population (Giabbanelli and Crutzen 2017). Challenges may also stem from the data *generated* during a project: complex M&S models may produce a massive amount of data from which we need to identify patterns. Note that using ML to assist with such M&S tasks results in a *hybrid* M&S study (Mustafee and Powell 2018), defined as the use of machine learning methods (or generally non-M&S methods) in one or more stages of an M&S study (e.g., implementation / model development, experimentation, conceptual modeling).

In this paper, we explain which big data challenges can arise at several stages of a M&S study, and how machine learning can be used in technically innovative ways to address some of these challenges. While a M&S study goes through several stages, several of them have already received attention when it comes to using ML and M&S. At the conceptual modeling stage, ML can help to define the structure of a model by articulating/hypothesizing relevant factors and their interactions (Sandhu et al. 2019). When designing the model, some components may be trained by machine learning and embedded into the simulation model (Wallis and Paich 2017), such as when agents derive models from their own observations. Consequently, we focus on three stages that have received relatively less attention: model calibration, experimentation, and the visualization of results. Since a systematic review of ML at each one of these stages would be better served by three independent studies, this paper provides a curated set of references to assist the reader in exploring the use of ML within each stage. Several of these references focus on the techniques, while others build on the experience of our team and our collaborators to illustrate challenges through real-world projects.

The remainder of this paper is structured around each of the three stages of M&S, organized in the order in which they are typically performed. In section 2, we focus on calibrating a simulation model using machine learning, that is, how ML can automatically adjust parts of a simulation model by creating, deleting, or tuning rules. A calibrated and validated computational model can then be used to run simulations, and section 5 accordingly targets this experimentation stage. Finally, experiments may need to be visualized either by modellers or by end-users, which is addressed in section 6. We conclude this paper by outlining the next steps to promote a tighter integration of ML and M&S.

2 MACHINE LEARNING FOR MODEL CALIBRATION

Models often have to account for phenomena that are difficult to directly observe, and hence to measure. One way to handle this situation is through *parametric uncertainty* (Briggs et al. 2012). For instance, our model of social influence on eating and physical activity behaviors included two parameters with unknown values: the level at which social norms become strong enough to trigger a change in behavior (threshold), and the extent to which the behavior is then changed (impact) (Giabbanelli et al. 2012). We had quantitative evidence on the expected model output and the value of all other parameters. We thus varied the two parameters until we identified a range of values in which the model's output matched expectations. This simple calibration process did not need machine learning solutions. However, parametric uncertainty is only one of many forms of uncertainty in a model. As observed by Jackson and others, "it is then important to characterize uncertainty not only regarding the values of model parameters within each assumption but also between different assumptions, as they may lead to different conclusions" (Jackson et al. 2011). Said otherwise, there is uncertainty when we are forced to make structural choices or 'assumptions' that affect the model's dynamics. Such assumptions are particularly common when modeling the decision-making processes of individuals, which includes several unobservable factors such as the threshold and impact aforementioned.

Within the context of Agent Based Modeling (ABM), there are at least two approaches to reduce structural uncertainty in the agents' rules via machine learning. One approach posits that, since humans have a brain, their virtual counterpart can be equipped with a neural network as well. This solution was implemented and presented at the 2017 Winter Simulation Conference, showing that a deep neural network can be trained and embedded within each agent (Negahban 2017). This novel data-driven approach treats cognition as a blackbox with a focus on accuracy. However, if the goal is to preserve the data- and theory-driven nature of a model, then the agents' decision-making processes can only be adjusted rather than entirely derived using machine learning. To this end, we discuss a second approach in this section.

Fuzzy Cognitive Maps (FCMs) are a modeling approach rooted in soft computing. They externalize the perspectives or 'mental models' held by participants into a computational model that articulates key factors and the rules governing their dynamics. For instance, our study at the 2018 Spring Simulation showed how mental models were externalized from 264 participants as FCMs, and compared them to understand differences between their rules (Lavin et al. 2018). Creating computational models in the form of FCMs may be as simple for participants as completing a questionnaire which is then automatically transformed into an FCM (Giabbanelli and Crutzen 2014). As an FCM contains the mental model of a person, it can be embedded within an agent. Hybrid ABM/FCM simulation models thus equip each agent with an FCM (Giabbanelli et al. 2019). However, the simplicity of building an FCM starts to become a problem at this stage because individuals may not be aware of all the factors that truly shape their decision-making processes, or they may struggle to assess the causal strengths of rules connecting factors. To reduce this uncertainty when data is also available, Papageorgiou has pioneered the use of machine learning for FCMs (Papageorgiou 2012). For example, unsupervised learning techniques can fine-tune the strengths of the rules (Papageorgiou et al. 2006), which may include removing a rule entirely (through a strength of 0) or adding in a rule (Figure 1). Note that machine learning algorithms *can*, but do not have to, use big data to adjust the model's rules. For instance, a study trained the model using a survey of 10 questions administered to 2,903 respondents (Dikopoulou et al. 2017), which is less than a 1 Mb file. Paradoxically, the absence of such training data would create the big data problem: without data to guide the search for the true model structure, we may have to generate a massive number of plausible model structures (i.e. large search space), assess each one, and navigate the results to select a suitable candidate. Said otherwise, the ability to employ machine learning to automatically tune a model avoids the big data problem of generating a massive number of possible rules.

In summary, modeling techniques such as FCMs have already been tightly coupled with machine learning for over a decade. This may not have reached the broader M&S community yet, where approaches such as ABM are more commonplace, but hybrid ABM/FCM models could leverage machine learning to automatically refine the agents' rules. A limiting factor to adopt this approach is a lack of familiarity with these tools. We need not only proficiency with several modelling approaches to avoid falling into the pitfalls of hybridization (Giabbanelli et al. 2017), but also a deep understanding of how machine learning is automatically re-engineering some of the model's structure. This may prompt the M&S community to tighten the relation between modeling and machine learning in the curriculum of new modelers, going even further than "teaching computational modeling in the data science era" as we previously discussed (Giabbanelli and Mago 2016). As some M&S curriculum may already be part of degrees (e.g., systems engineering) that have limited room for an additional specialization, a practical alternative is to promote collaborations between experts in M&S and experts in machine learning. While our interdisciplinary practice of modeling often rests on the trio of modeller-experts-stakeholders, this need for technical collaborations may lead to adding a machine learning expert as a fourth role.

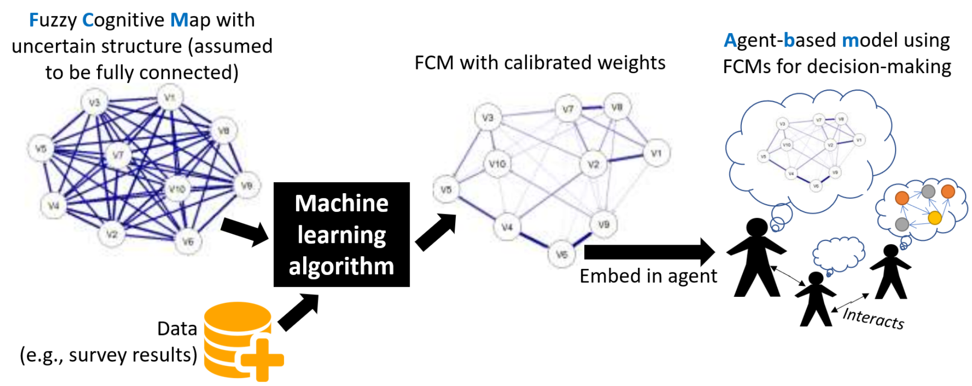

Figure 1: Machine learning can calibrate a model, such as a Fuzzy Cognitive Map, which can then perform complex functions such as governing the decision-making activities of each agent in an Agent-Based Model. *Adapted from (Dikopoulou et al. 2017) and (Giabbanelli et al. 2017).*

2.1 Opportunity #1

Uncertainty is common in simulation models. Parametric uncertainty can be solved in small models by calibration rather than machine learning, e.g. by identifying a narrow range of parameter values that replicate the target model behavior. When models have structural uncertainty and data is available, machine learning can navigate the larger space of possible structures, thus avoiding the big data problem of generating many model structures and comparing their simulation outcomes. In particular, machine learning has been used for over a decade to fine-tune the structure of Fuzzy Cognitive Maps, which can now be used in Agent-Based Models to provide the agents' mental model. Given the multiplicity of case studies and the maturity of methods, the main barrier to using machine learning for model calibration is the expertise needed. Indeed, when a machine learning algorithm adjusts a model's structure, we must understand both the algorithm and the model. This can impact practices in M&S either by having experts on modeling *and* on machine learning in the same team, or by changing the curriculum to equip the next generation of modelers with machine learning skills.

3 MACHINE LEARNING AT THE EXPERIMENTATION STAGE

After calibrating and validating the model within its intended context of use, we can now perform *experiments*. In public policy-making, experiments may consist of establishing a baseline and comparing it with a series of potential interventions. For instance, Firmansyah and colleagues first computed the expected pollution level of a city in the absence of new interventions (Firmansyah et al. 2019). This baseline outcome was then compared to alternative (or 'future') scenarios such as an increase in green space or a change in the energy mix. This type of experiment is relatively low on computations compared to the next two types that we will discuss. Indeed, the number of experiments is essentially a function of the number of scenarios and replications if the model is stochastic. Consider 1 baseline and 2 alternative scenarios, where each must be repeated 10 times: we will need $3 \times 10 = 30$ runs. However, in many cases, the goal is not only to test *what* we should intervene on (e.g., do we change green space or the energy mix?) but also to accurately characterize to *which extent* (e.g., how much more green space do we need to make a difference?). Each scenario would then have to be run at different levels. In the model of Khademi *et al.*, the authors explored the five year prevalence of hypertension with respect to three different levels of change in social norms on food behaviors (Khademi et al. 2018). The number of experiments is now growing as a function of scenarios, levels, and replications. This number grows even larger when the scenarios need to be identified from the model's behavior instead of being pre-determined. For example, the pollution level in Firmansyah's model is driven directly by several parameters (e.g., green space, waste infrastructure, energy mix) and indirectly by many others. Any of these parameters is a potential policy

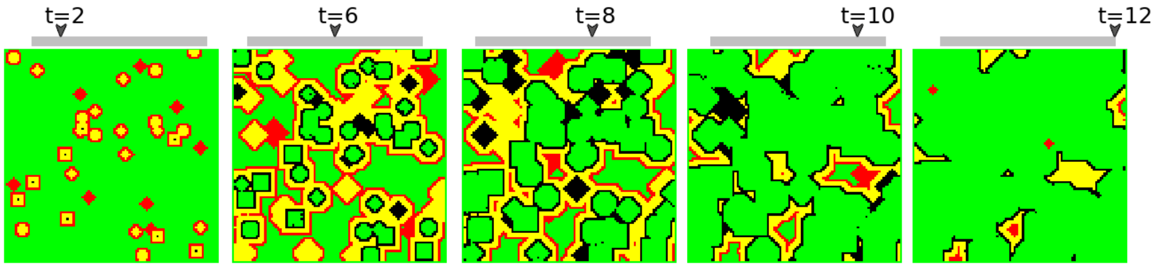

Figure 2: A cellular automaton model of HIV represents the state of each biological cell as a color. Visualizing the state of the whole model consists of visualizing a color grid. Here, the user is visually five out of 12 time ticks of the model. As it is a stochastic model, another run can look different. *Adapted from (Giabbanelli et al. 2019).*

lever, thus we may start by having *at least* as many scenarios as there are parameters. In practice, we have significantly more policy scenarios than parameters since a scenario can be defined by a synergistic action on *several* parameters jointly (Giabbanelli and Crutzen 2017). As a result, the number of experiments can grow exponentially in the number of parameters and level. For instance, if ten parameters have only two levels, then we have $\underbrace{2 \times 2 \ldots \times 2}_{2^{10}}$ combinations, each of which needs to be repeated in stochastic models.

The exponential growth is not a challenge if two conditions are met: (*i*) the number of parameters and their levels is small, and (*ii*) each simulation run can be computed quickly. Design of Experiments (DoE) techniques such as factorial analyses are particularly effective in this situation (Giabbanelli and Crutzen 2013; Giabbanelli et al. 2012). Our analysis presented at the 2017 Winter Simulation Conference showed that all combinations could be explored for models with low computational costs having up to 25 binary parameters (Lavin and Giabbanelli 2017). However, there are also many cases of practical relevance where at least one of these two conditions is violated. If only (*i*) is violated, then we have a big data explosion (similarly to the previous section) but the problem may still be tractable. If both are violated, then it becomes prohibitive to even generate the big dataset of simulation results.

When using causal connections in a Fuzzy Cognitive Map as policy levers, Firmansyah's model includes 98 parameters (Firmansyah et al. 2019) and the Provincial Health Services Authority model of obesity has as many as 269 parameters (Drasic and Giabbanelli 2015). Even *if* each parameter only had two levels and computations were fast, generating 2^{269} combinations would still be beyond the scope of big data. In contrast, detailed biological models for the spread of the Human Immunodeficiency Virus (HIV) may have few parameters but a single run can be extremely costly (Figure 2) given the number of cells in a human body, the need to track the different genome of each infected cell, a temporal resolution in hours but a duration spanning years. Such cases call for innovative and efficient approaches to run experiments.

A well-chosen Design of Experiments allows us to select a few combinations of parameter values to approximate a model's behavior (Sanchez et al. 2018; Law 2017), but two observations suggest that machine learning can go further. First, the *computational costs of experiments* are not always identical, even when using the same model. For instance, one combination of parameter values in the HIV model can produce a low viral load (so we must simulate until the end to guarantee that the patient remains healthy), but another may quickly lead to the onset of AIDS (so the simulation can stop). Similarly, a simulation of an epidemic can stop if the virtual population has died. Second, the outcome for each combination of parameter values is not necessarily unique: a similar combination may yield an indistinguishable outcome within the application context. Taken together, these observations suggest that a machine learning model can be used to predict the cost of an experiment and, particularly for the costly ones, predict their outcomes from the experiments that we have already performed. The type of machine learning model to build depends on whether we seek a fine-grained numerical prediction or a coarser categorical prediction. For a given experiment, a binary classification can tell us whether computations are too costly, or predict very different outcomes (e.g., the disease either prevails or disappears). A multi-class classification could categorize the computations (e.g., very/moderately/not expensive) or the outcome (e.g., most/few/no people are infected). Finally, a regression can output the specific cost (e.g., in minutes) or outcome (e.g., prevalence of a disease).

Some of this potential has been shown in the last few years. Noting that Agent-Based Models "are quite heavy computationally" but that we need a "large number of simulations", van der Hoog concluded that the aforementioned issue of "computational intractability is therefore looming and ubiquitous". His suggested solution is "to use machine learning algorithms in order to reduce the computer simulation to a lighter form, by emulating the models", that is, through surrogate modelling. He then proposed to use deep neural networks either at a micro-level, to predict each agent's behavior, or at a macro-level, to emulate the entire Agent-Based Model (van der Hoog 2019). This proposition was realized by Lamperti *et al.*, who successfully created a machine learning surrogate of an ABM using XGBoost and decision trees (Lamperti et al. 2018). The authors also noted how this machine learning approach differs from *kriging*, which is hard to apply to models with many parameters and can lead to very smooth response surfaces unlike those produced by an ABM. As machine learning surrogates *can* be built effectively, the next task is to select the right type of surrogate (e.g., support vector machine, neural networks). Gorissen developed an automatic approach for this model type selection problem (Gorissen et al. 2009). Finally, as we need to perform *some* experiments to create training data for the machine learning model, we should identify which experiments are likely to capture the rugged response surface of a simulation model. Edali and Yucel showed that sequential sampling can perform this task accurately (Edali and Yücel 2019).

3.1 Opportunity #2

Finding the best policy levers in a model may require a tremendous amount of experiments. This big data generation can become intractable when there is also a high computational cost associated with performing each experiment. Machine learning can address this issue by creating a computationally cheaper model, that is, via surrogate modeling. A surrogate can leverage previous experiments to *predict the outcome* instead of performing new simulations, and/or identify identify which simulations are really needed. Future work may also use machine learning to *predict the cost* of an experiment before deciding whether to conduct it.

4 VISUALIZING MODEL RESULTS WITH MACHINE LEARNING

4.1 How Can Visualizations of Big Data Offer Insight into a Model's Behavior?

Visualization can take place at many stages of M&S for different reasons. For instance, in the first phase of conceptual modeling, we may visualize a dataset to identify important elements that would need to be included in the model. In a later phase of conceptual model, we can visually compare the structure of a model with the available evidence (Giabbanelli and Jackson 2015). During verification, we can visually explore the outputs vis-a-vis the expected model behavior, that is, we already have a notion of what would be wrong visually. For example, noticing that agents get stuck or witnessing the unlikely disappearance of a disease would then suggest errors in the implementation. Visualizing results of a verified and validated model is a different task: instead of looking for implementation errors by knowing what we should see, we try to *learn* from the output. We are thus looking for a variety of patterns across space (e.g., can regions be clustered in terms of model output?), time (e.g., are the entities' states stabilizing or oscillating?), and runs of the model (e.g., is there a lot of variability in the outputs?). Identifying patterns through this big, multi-faceted data is arduous. Consider that we are looking for temporal patterns through as cycling through states (Giabbanelli and Baniukiewicz 2019). A simple visualization would be to display the state of all agents at a given time tick. To go to another time tick, we can click and refresh the entire view (Figure 2). In this situation, we would be unlikely to notice whether some of the agents are starting to cycle through the same sequence of states as they had fifteen steps ago. While we could visualize the data as time series, we may not be able to look at the time series of states for every single agent either. The big data nature of the output entails that there is too much information to track or even display at once.

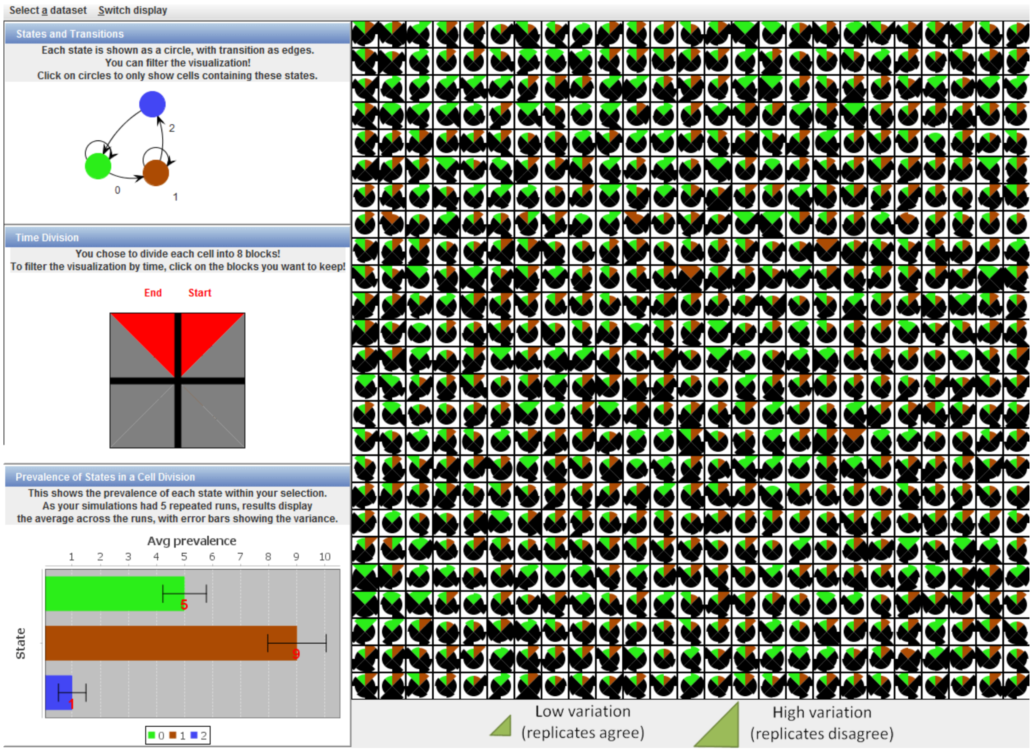

Figure 3: This advanced environment provides multiple visualizations at the same time, such as a state-transition graph (top-left) and a bar chart (bottom-left). The main pane uses clock glyphs to wrap the successive states of a cell over time within a single display: the value at $t = 0$ is positioned at 'noon', then the next value continues clock-wise. *Adapted from (Giabbanelli and Baniukiewicz 2019).*

There are at least two ways in which machine learning can assist modelers with finding patterns in large multi-faceted data. First, machine learning can filter the data: instead of directly looking at the simulation output, we visualize the results of a machine learning model applied to this output. For example, hierarchical clustering can reveal how the output may be decomposed into spatial regions that are similar in states. Alternatively, we can visualize the simulation output with help from machine learning techniques, for instance by highlighting regions based on the cluster to which they belong. Similar situations were named 'visually enhanced mining' and 'computationally enhanced visualizations' respectively (Bertini and Lalanne 2010). Note that both situations differ from 'interactive model analysis' in which an interactive visualization serves to improve a machine learning model (Liu et al. 2017).

While the broad principles of applying machine learning to visualizations of simulation outputs are easy to enunciate, the specifics still need to be defined by the M&S research community. Consider the case of visually enhanced mining, in which machine learning acts as filter. To create a software environment in which modelers could provide simulation data and automatically visualize results from machine learning algorithms, we would need three broad steps: (*i*) *select* suitable machine learning algorithms based on characteristics of the outputs and/or questions from modellers, (*ii*) run these algorithms on the outputs, and (*iii*) present the results through visualizations. Step (*ii*) may require an efficient implementation given the big data nature of the output, but it does not require innovation as it boils down to applying a suitable algorithm onto data. Research on step (*iii*) has already provided many possible solutions to visualize certain types of machine learning approaches, such as decision tree classifiers (van den Elzen and van Wijk 2011), random forest classifiers (Welling et al. 2016), or association rules (Hahsler and Karpienko 2017). Even visualizations of highly complex machine learning models such as deep neural networks have emerged in the recent years (Kahng et al. 2018).

The crux of the issue in developing the whole 'visually enhanced mining' environment is thus step (*i*): how to automatically *select* which machine learning algorithms are worth running on a set of simulation outputs? We cannot run a large collection of machine learning and tell modellers to somehow sift through the results, because that would be translating a big data problem (simulation output) into yet another big data problem (visualizing many machine learning models). The challenge may even be greater when developing 'computationally enhanced visualizations': given features of the data, questions from modellers, *and the visualizations used*, which machine learning methods should be selected to enhance these visualizations? While practices in machine learning are well-established to select *features of a dataset given a method*, here we face the inverse problem of *selecting machine learning models given the features of a dataset*. Some reviews have been devoted to listing properties of the dataset and suitable classifiers, such as the highly cited work of Lotte *et al.* within the context of ElectroEncephaloGraphy (Lotte et al. 2007). Undertaking a similar effort within M&S would already be significant, but most benefits will be unlocked once this selection can be automatized.

4.2 Opportunity #3

The big data nature of a simulation output makes it challenging to create visualizations that are conducive to generating new insights into the model's behavior. These visualizations may be augmented or filtered using machine learning. However, either approach needs to *select* a suitable set of machine learning algorithms given the properties of the simulation output and/or questions from modellers. Selection itself is a machine learning task. A leap forward for the visualization of simulation outputs would thus be to use machine learning *twice*: first to select the right algorithms, and then applying them to create or enhance a visualization.

4.3 How Can we Adapt Visualizations to the Needs of the End-Users?

For modellers, a visualization may only be *one* of various ways to assess a model's behavior. We can rely on a variety of statistics displayed in table form. We may also employ software engineering techniques to guide the assessment when there are formal descriptions, such as a model description in the Systems Biology Markup Language (Hucka et al. 2018), or descriptions of experiments with NEDL or SESSL (Peng et al. 2014). The situation is different for the end-users who may not be able or interested to look into the code, statistics, and formal descriptions. This audience may thus rely more heavily on interactive visualizations. While the practice of accessible design provides detailed recommendations in general, two challenges are of particular interest here. First, seasoned modellers may be used to visualizations that turn out to be unfamiliar for the end-users. This can apply to more advanced visualizations such as parallel coordinates or using glyphs (Figure 3), but it also holds for forms that modellers may assume to be more 'universally' understood. For instance, we developed a software that lets policymakers interact with a weighted, directed network (Giabbanelli and Baniukiewicz 2018). Our usability test showed that policymakers occasionally struggled in using such large node-links diagrams (Giabbanelli et al. 2016). The struggle may be subtle as end-users may be *able* to correctly accomplish a task using a visualization of outputs, yet it takes them too long to be feasible in practice, or they may not be confident that they did it correctly (Giabbanelli and Baniukiewicz 2019). The second challenge is that models that are truly useful for end-users may kept on being employed much beyond the duration of the initial engagement with modellers. That is, users will eventually be on their own using the models and examining visualizations of outputs. In this context, they may realize that they have new questions, which may be cumbersome or even impossible to answer using the visualizations provided. Revising the software every time creates a barrier (e.g., human, financial) and may not always be an option.

Machine learning presents opportunities to address both challenges for end-users as it can *adapt* the visualization to their needs. While the previous subsection focused on adapting to the data's characteristics, the emphasis here is on learning the right visualization *from the trace of user interactions*. In short, we

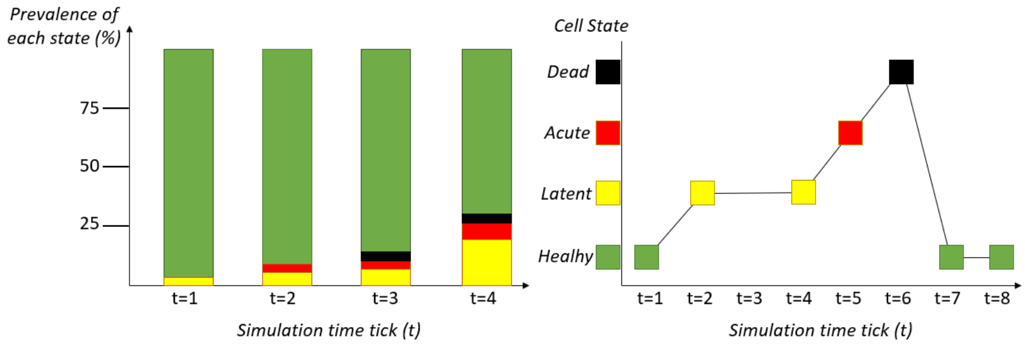

Figure 4: Two possible visualizations from the same simulation outputs: (a) a stack bar chart, aggregating the state of all entities; and (b) a time series, showing the state of one entity.

can know what a user did by logging mouse motions, clicks (e.g. to select or undo), and pauses (Dou et al. 2009; Brown et al. 2014). For some category of end-users, devices that perform eye tracking may also be available to capture exactly what they are looking at. From these traces, we may infer what a user wishes to do, and propose a visualization that is more supportive of this goal. Consider the example in the previous section where we show the state of all entities at one time tick of the simulation. A user wishing to see which entities changed state from t to $t+1$ may be repeatedly going back-and-forth between these two time ticks. The software may realize that the user wishes to perform a comparison, and thus adapt the visualization by highlighting entities whose state changes while dimming the other ones.

There can be a problematic temptation to believe that machine learning can perfectly reveal what users truly desire based on what they do. The reality is more nuanced, so an operational definition is needed to appreciate practical opportunities and limitations. Let us denote by $\mathbb{V}$ the set of all visualizations that the application can provide. At a given time t, a user interacts with a subset $V_t \subseteq \mathbb{V}$ of the visualizations, such as a bar chart and state-transition graph (Figure 3). The actions performed by users (e.g., clicking on a button, selecting an entity) are logged as a time-ordered sequence $a_1, \ldots, a_i, \ldots a_n$ where each action tells us what the user did within an environment V_t at time t. The machine learning task is to predict what V_{t+1} should be given the actions $a_1, \ldots, a_t$. This is similar to the use of machine learning on-the-fly to adapt data structures, where the sequence of actions (e.g., adding, deleting, searching) serves to predict the structure into which we swap (Eastep et al. 2011). One essential limitation is that the prediction depends on what the user *was able* to do with the previous visualizations. For instance, if all the user can do is interact with a pie chart showing *aggregate* simulation results (e.g., half of the agents are infected), we cannot tell that the goal was actually to track the state of *a specific agent* over time. The other limitation is that several candidates for V_{t+1} can be equally likely given the log of actions. Consider that a user visualizes the state of each entity at the beginning of the simulation, then moves onto the next time tick, and so on until the end of the simulation (Figure 2). The user may have wanted to get a sense for the prevalence of different states throughout the simulation, which can be done through an aggregate such as a stacked bar chart (Figure 4-a). Alternatively, the user may have been tracking the state of *one* specific entity, and a time series for this entity would be useful (Figure 4-b). The user may even have wanted to see the spread of an infection from a given point, in which case a contour plot may have worked. A machine learning model may thus *suggest* likely visualizations to the user, but should not automatically switch to one. As automatically guiding a user through visualizations is an open field of research, several prototypes have been proposed but mature solutions are still in development (Ceneda et al. 2017). In the “Small Multiples, Large Singles” approach, users start with a large visualization and are shown several possible next steps as ‘small visualizations’ to succeed in their task (van den Elzen and van Wijk 2013). More recently, `Voyager` (and its successor `Voyager 2`) supported users in “browsing of recommended charts chosen according to statistical and perceptual measures” (Wongsuphasawat et al. 2016).

4.4 Opportunity #4

Visualizations support end-users in generating insight from simulation results. However, the needs and abilities of end-users may be difficult to precisely capture for modellers when creating visualizations, or they may change beyond the initial engagement with modellers. Machine learning can use the sequence of interactions that end-users have with current visualizations to suggest which visualizations should be next to support their task. There are limitations since the same interactions may be associated with several tasks, and these interactions are constrained by visualizations provided to end-users. Automatically guiding end-users through visualizations of simulation outputs using machine learning is an important area of future work to support the use and impact of models.

5 CONCLUSION

Neither Modeling & Simulation nor Machine Learning are novel research fields. This article explored the many benefits that can emerge from using existing methods in machine learning to support several key steps performed in the simulation community, such as investigating the model's parameter space or visualizing the results of experiments. As the lack of big data methods in Modeling & Simulation might partly stem from a lack of expertise, this article also contributes to drawing the attention of our community to the new field of possibilities that machine learning offers with respect to current and future simulation challenges.

When a new approach with high potential is identified, there can be a temptation for end-users to aim at becoming experts in the approach. This is certainly attractive from a logistical standpoint, as end-users can then take care of their own needs, at their pace. However, acquiring another expertise can be challenging for simulationists. As getting students to graduate later by taking additional courses is an undesirable institutional outcome, simulation expertise is more likely to be traded for machine learning expertise by replacing M&S courses with ML courses. Time for professional training is also a finite bucket, as professionals may hardly be able to attend machine learning workshops and conferences unless they reduce exposure to simulation conferences. Adding roles to team members can thus backfire compared to considering, as we propose here, the addition of an expert in machine learning on suitable simulation projects. To enable this addition and make it fruitful, we form two recommendations. First, simulationists should be exposed to what machine learning can do for them, and what it needs (e.g., data requirements, computational costs). This would position simulationists as informed end-users who can identify reasonable questions in a given context. Second, bridges should be made between communities such that simulationists can identify the machine learning experts they need for their specific questions. Tracks such as 'Big Data in Simulation' (WinterSim'19) or 'AI and Simulation' (SpringSim'20) contribute to creating such bridges. Failure to build an interface between ML and M&S by ignoring either of these two points can not only bring disappointment but, if it happens repeatedly, set the field back instead of moving it forward.

REFERENCES

Bertini, E., and D. Lalanne. 2010. "Investigating and Reflecting on the Integration of Automatic Data Analysis and Visualization in Knowledge Discovery". *ACM SIGKDD Explorations Newsletter* 11(2):9–18.

Briggs, A. H., M. C. Weinstein, E. A. Fenwick, J. Karnon, M. J. Sculpher, A. D. Paltiel, ISPOR-SMDM Modeling Good Research Practices Task Force et al. 2012. "Model Parameter Estimation and Uncertainty: a Report of the ISPOR-SMDM Modeling Good Research Practices Task Force-6". *Value in Health* 15(6):835–842.

Brown, E. T., A. Ottley, H. Zhao, Q. Lin, R. Souvenir, A. Endert, and R. Chang. 2014. "Finding Waldo: Learning About Users From Their Interactions". *IEEE Transactions on Visualization and Computer Graphics* 20(12):1663–1672.

Ceneda, D., T. Gschwandtner, T. May, S. Miksch, H.-J. Schulz, M. Streit, and C. Tominski. 2017. "Characterizing Guidance in Visual Analytics". *IEEE Transactions on Visualization and Computer Graphics* 23(1):111–120.

Dikopoulou, Z., E. Papageorgiou, V. Mago, and K. Vanhoof. 2017. "A New Approach Using Mixed Graphical Model for Automatic Design of Fuzzy Cognitive Maps From Ordinal Data". In *2017 IEEE International Conference on Fuzzy Systems*. July 9^{th}-12^{th}, Naples, Italy, 1–6.

Dou, W., D. H. Jeong, F. Stukes, W. Ribarsky, H. R. Lipford, and R. Chang. 2009. "Recovering Reasoning Processes From user interactions". *IEEE Computer Graphics and Applications* 29(3):52–61.

Drasic, L., and P. J. Giabbanelli. 2015. "Exploring the Interactions Between Physical Well-being and Obesity". *Canadian Journal of Diabetes* 39:S12–S13.

Eastep, J., D. Wingate, and A. Agarwal. 2011. "Smart Data Structures: an Online Machine Learning Approach to Multicore Data Structures". In *Proceedings of the 8th ACM International Conference on Autonomic Computing*. June 14th-18th, Karlsruhe, Germany, 11–20.

Edali, M., and G. Yücel. 2019. "Exploring the Behavior Space of Agent-based Simulation Models Using Random Forest Metamodels and Sequential Sampling". *Simulation Modelling Practice and Theory* 92:62–81.

Firmansyah, H. S., S. H. Supangkat, A. A. Arman, and P. J. Giabbanelli. 2019. "Identifying the Components and Interrelationships of Smart Cities in Indonesia: Supporting Policymaking via Fuzzy Cognitive Systems". *IEEE Access* 7:46136–46151.

Giabbanelli, P., R. Flarsheim, C. Vesuvala, and L. Drasic. 2016. "Developing Technology to Support Policymakers in Taking a Systems Science Approach to Obesity and Well-being". *Obesity Reviews* 17:194–195.

Giabbanelli, P. J., A. Alimadad, V. Dabbaghian, and D. T. Finegood. 2012. "Modeling the Influence of Social Networks and Environment on Energy Balance and Obesity". *Journal of Computational Science* 3(1-2):17–27.

Giabbanelli, P. J., and M. Baniukiewicz. 2018. "Navigating Complex Systems for Policymaking Using Simple Software Tools". In *Advanced Data Analytics in Health*, 21–40. Cham, Switzerland: Springer.

Giabbanelli, P. J., and M. Baniukiewicz. 2019. "Visual Analytics to Identify Temporal Patterns and Variability in Simulations from Cellular Automata". *ACM Transactions on Modeling and Computer Simulation (TOMACS)* 29(1):5.

Giabbanelli, P. J., and R. Crutzen. 2013. "An Agent-based Social Network Model of Binge Drinking Among Dutch Adults". *Journal of Artificial Societies and Social Simulation* 16(2):10.

Giabbanelli, P. J., and R. Crutzen. 2014. "Creating Groups with Similar Expected Behavioural Response in Randomized Controlled Trials: a Fuzzy Cognitive Map Approach". *BMC Medical Research Methodology* 14(1):130.

Giabbanelli, P. J., and R. Crutzen. 2017. "Using Agent-based Models to Develop Public Policy About Food Behaviours: Future Directions and Recommendations". *Computational and Mathematical Methods in Medicine* 2017.

Giabbanelli, P. J., M. Fattoruso, and M. L. Norman. 2019. "Cofluences: Simulating the Spread of Social Influences via a Hybrid Agent-based/fuzzy Cognitive Maps Architecture". In *Proceedings of the 2019 ACM SIGSIM Conference on Principles of Advanced Discrete Simulation*. June 3rd-5th, Chicago, Illinois, 71–82.

Giabbanelli, P. J., C. Freeman, J. A. Devita, N. Rosso, and Z. L. Brumme. 2019. "Mechanisms for Cell-to-cell and Cell-free Spread of HIV-1 in Cellular Automata Models". In *Proceedings of the 2019 ACM SIGSIM Conference on Principles of Advanced Discrete Simulation*. June 3rd-5th, Chicago, Illinois, 103–114.

Giabbanelli, P. J., S. A. Gray, and P. Aminpour. 2017. "Combining Fuzzy Cognitive Maps With Agent-based Modeling: Frameworks and Pitfalls of a Powerful Hybrid Modeling Approach to Understand Human-environment Interactions". *Environmental Modelling & Software* 95:320–325.

Giabbanelli, P. J., and P. J. Jackson. 2015. "Using Visual Analytics to Support the Integration of Expert Knowledge in the Design of Medical Models and Simulations". *Procedia Computer Science* 51:755–764.

Giabbanelli, P. J., and V. K. Mago. 2016. "Teaching Computational Modeling in the Data Science Era". *Procedia Computer Science* 80:1968–1977.

Gorissen, D., T. Dhaene, and F. D. Turck. 2009. "Evolutionary Model Type Selection for Global Surrogate Modeling". *Journal of Machine Learning Research* 10:2039–2078.

Hahsler, M., and R. Karpienko. 2017. "Visualizing Association Rules in Hierarchical Groups". *Journal of Business Economics* 87(3):317–335.

Hucka, M., F. T. Bergmann, A. Dräger, S. Hoops, S. M. Keating, N. Le Novère, C. J. Myers, B. G. Olivier, S. Sahle, J. C. Schaff, L. P. Smith, and D. J. Wilkinson. 2018. "The Systems Biology Markup Language (SBML): Language Specification for Level 3 Version 2 Core". *Journal of Integrative Bioinformatics* 15(1).

Jackson, C. H., L. Bojke, S. G. Thompson, K. Claxton, and L. D. Sharples. 2011. "A Framework for Addressing Structural Uncertainty in Decision Models". *Medical Decision Making* 31(4):662–674.

Kahng, M., P. Y. Andrews, A. Kalro, and D. H. P. Chau. 2018. "ActiVis: Visual Exploration of Industry-scale Deep Neural Network Models". *IEEE Transactions on Visualization and Computer Graphics* 24(1):88–97.

Khademi, A., D. Zhang, P. J. Giabbanelli, S. Timmons, C. Luo, and L. Shi. 2018. "An Agent-Based Model of Healthy Eating with Applications to Hypertension". In *Advanced Data Analytics in Health*, 43–58. Cham, Switzerland: Springer.

Lamperti, F., A. Roventini, and A. Sani. 2018. "Agent-based Model Calibration Using Machine Learning Surrogates". *Journal of Economic Dynamics and Control* 90:366–389.

Lavin, E. A., and P. J. Giabbanelli. 2017. "Analyzing and Simplifying Model Uncertainty in Fuzzy Cognitive Maps". In *2017 Winter Simulation Conference*, edited by W. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, 1868–1879. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Lavin, E. A., P. J. Giabbanelli, A. T. Stefanik, S. A. Gray, and R. Arlinghaus. 2018. "Should We Simulate Mental Models to Assess Whether They Agree?". In *Proceedings of the Annual Simulation Symposium*, 6. Society for Computer Simulation International.

Law, A. M. 2017. "A Tutorial on Design of Experiments for Simulation Modeling". In *2017 Winter Simulation Conference*, edited by W. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, 550–564. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Liu, S., X. Wang, M. Liu, and J. Zhu. 2017. "Towards Better Analysis of Machine Learning Models: A Visual Analytics Perspective". *Visual Informatics* 1(1):48–56.

Lotte, F., M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi. 2007. "A Review of Classification Algorithms for EEG-based Brain–computer Interfaces". *Journal of Neural Engineering* 4(2):R1.

Mustafee, N., and J. H. Powell. 2018. "From Hybrid Simulation to Hybrid Systems Modelling". In *Proceedings of the 2018 Winter Simulation Conference*, edited by M. Rabe, A. Juan, N. Mustafee, A. Skoogh, S. Sain, and B. Johansson, 1430–1439. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Negahban, A. 2017. "Neural Networks and Agent-based Diffusion Models". In *Proceedings of the 2017 Winter Simulation Conference*, edited by W. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, 1407–1418. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Papageorgiou, E. I. 2012. "Learning Algorithms for Fuzzy Cognitive Maps–a Review Study". *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 42(2):150–163.

Papageorgiou, E. I., C. Stylios, and P. P. Groumpos. 2006. "Unsupervised Learning Techniques for Fine-tuning Fuzzy Cognitive Map Causal Links". *International Journal of Human-Computer Studies* 64(8):727–743.

Peng, D., R. Ewald, and A. M. Uhrmacher. 2014. "Towards Semantic Model Composition via Experiments". In *Proceedings of the 2nd ACM SIGSIM Conference on Principles of Advanced Discrete Simulation*. May 18th-21th, Denver, Colorado, 151–162.

Rosso, N., and P. Giabbanelli. 2018. "Accurately Inferring Compliance to Five Major Food Guidelines Through Simplified Surveys: Applying Data Mining to the UK National Diet and Nutrition Survey". *JMIR Public Health and Surveillance* 4(2):e56.

Sanchez, S. M., P. J. Sánchez, and H. Wan. 2018. "Work Smarter, not Harder: a Tutorial on Designing and Conducting Simulation Experiments". In *2018 Winter Simulation Conference*, edited by M. Rabe, A. Juan, N. Mustafee, A. Skoogh, S. Sain, and B. Johansson, 237–251. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Sandhu, M., P. J. Giabbanelli, and V. K. Mago. 2019. "From Social Media to Expert Reports: the Impact of Source Selection on Automatically Validating Complex Conceptual Models of Obesity". In *Proceedings of the 11th International Conference on Social Computing and Social Media*. July 26th-31st, Orlando, Florida, 434–452.

van den Elzen, S., and J. J. van Wijk. 2011. "Baobabview: Interactive Construction and Analysis of Decision Trees". In *2011 IEEE Conference on Visual Analytics Science and Technology (VAST)*, 151–160. Institute of Electrical and Electronics Engineers.

van den Elzen, S., and J. J. van Wijk. 2013. "Small Multiples, Large Singles: A New Approach for Visual Data Exploration". In *Computer Graphics Forum*, Volume 32, 191–200. Oxford, UK: Wiley Online Library.

van der Hoog, S. 2019. "Surrogate Modelling in (and of) Agent-based Models: A Prospectus". *Computational Economics* 53(3):1245–1263.

Voinov, A., K. Jenni, S. Gray, N. Kolagani, P. D. Glynn, P. Bommel, C. Prell, M. Zellner, M. Paolisso, R. Jordan, E. Sterling, L. Schmitt Olabisi, P. J. Giabbanelli, Z. Sun, C. Le Page, S. Elsawah, T. K. BenDor, K. Hubacek, B. K. Laursen, A. Jetter, L. Basco-Carrera, L. Singer, Alison andYoung, J. Brunacini, and A. Smajgl. 2018. "Tools and Methods in Participatory Modeling: Selecting the Right Tool for the Job". *Environmental Modelling & Software* 109:232–255.

Wallis, L., and M. Paich. 2017. "Integrating Artifical Intelligence with Anylogic Simulation". In *2017 Winter Simulation Conference*, edited by W. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, 4449–4449. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Welling, S. H., H. H. Refsgaard, P. B. Brockhoff, and L. H. Clemmensen. 2016. "Forest Floor Visualizations of Random Forests". *arXiv preprint arXiv:1605.09196*.

Wongsuphasawat, K., D. Moritz, A. Anand, J. Mackinlay, B. Howe, and J. Heer. 2016. "Voyager: Exploratory Analysis via Faceted Browsing of Visualization Recommendations". *IEEE Transactions on Visualization and Computer Graphics* 22(1):649–658.

AUTHOR BIOGRAPHIES

PHILIPPE J. GIABBANELLI, Ph.D., is an Associate Professor in the Department of Computer Science & Software Engineering at Miami University (USA). He received his Ph.D. from Simon Fraser University (Canada), worked as a researcher at the University of Cambridge (UK), and taught Modeling & Simulation as well as Machine Learning at several nationally ranked American universities. His research interests include machine learning and simulation models. He is an Associate Editor for BMC Medical Informatics & Decision Making (MIDM) as well as Social Network Analysis and Mining (SNAM). He was also the lead editor for the volume *Advanced Data Analytics in Health* in 2018. He has held leadership positions at international simulation conferences, most recently as chair of the Ph.D. Colloquium for the ACM SIGSIM PADS'19, and track chair at the 2020 Spring Simulation conference. His email address is giabbapj@miamioh.edu. His website is https://www.dachb.com.