# Understanding Behaviors in Different Domains: The Role of Machine Learning Techniques and Network Science

Grace Teo[1(✉)], Lauren Reinerman-Jones[1], Joseph McDonnell[2], Hayden J. Trainor[3], Rainier A. Porras[3], and Jacob G. Feuerman[3]

[1] Institute for Simulation and Training, University of Central Florida, Orlando, FL, USA
{gteo, lreinerm}@ist.ucf.edu
[2] Dynamic Animation Systems, Fairfax, VA, USA
joe.mcdonnell@d-a-s.com
[3] United States Military Academy, West Point, NY, USA
{hayden.trainor, rainier.porras, jacob.feuerman}@usma.edu

**Abstract.** Recent developments in the Internet of Things (IoT), social media, and the data sciences have resulted in larger volumes of data than ever before, offering more opportunity for observing and understanding behaviors. Advances in data analytic and machine learning techniques have also enabled assessments to be more multi-faceted, incorporating data from more sources. Machine learning algorithms such as Decision Trees and Random Forests, K-nearest neighbors, and Artificial Neural Networks have been used to uncover hidden patterns in data and derive predictions and recommendations from a wide range of data types and sources. However, these do not necessarily yield insights into behaviors in complex systems/domains. Methods from mathematics such as Set Theory, Graph Theory, and Network Science may be useful in shedding light on the interactions and relationships within and across domains. This paper provides a description of the applications, strengths, and limitations of some of these techniques and methods.

**Keywords:** Machine learning techniques · Decision tree · Random forest
K-nearest neighbor · Artificial Neural Network · Network science

## 1 Introduction

Most of the data in the world today has only been created within the last few years [1], and the amount of data generated daily is projected to only increase, especially with the rise of the Internet of Things (IoT) and social media. One report projected that by 2020, each individual would create about 1.7 megabytes of new information every second [2]. All this offers unprecedented opportunities for assessments to understand various behavioral phenomena. With this growth of big data, there has also been a surge in the number of data analytic techniques. Some of these techniques employ machine learning, which has been applied to analytic problems such as prediction, classification,


D. D. Schmorrow and C. M. Fidopiastis (Eds.): AC 2018, LNAI 10915, pp. 329–340, 2018.
https://doi.org/10.1007/978-3-319-91470-1_27

clustering, and revealing associations. These can be helpful in addressing research questions such as:

- What task or person characteristics predict trust in automation? (Prediction)
- What indicators tend to cluster and do they suggest a new construct, e.g., the construct of *fitness for duty* from hours of sleep, blood alcohol level, cardiovascular functioning [3, 4]? (Clustering)
- How to classify someone as being in high vs. low workload? (Classification)

## 2 Machine Learning

Machine learning techniques can be categorized into those where the machine is trained with data consisting of inputs with the corresponding behavioral outcomes (supervised learning), and those where the behavioral outcome is unknown and the machine is simply tasked to uncover hidden patterns or structure in the data (unsupervised learning). Unlike unsupervised techniques which have limited applications, supervised learning techniques are more commonplace [5]. They can be used to identify precedents of certain behaviors. Examples of supervised learning techniques include decision trees and random forests, k-nearest neighbor, and artificial neural networks. The data that the machine uses to learn or is trained on, is called *training data*. The new data to which the predictive, machine-developed algorithm or model is applied, is *test data*.

## 3 Decision Trees and Random Forests

Decision trees are a supervised, predictive modeling technique where input variables are expressed as decision rules that are applied in succession (i.e., recursive partitioning of data) with the goal of classifying observations into their outcomes classes. In doing so, the most influential inputs that relate to the outcome are identified. This result can then be used to predict the outcome class for new observations. Decision trees can be constructed by numerous software programs, both basic and sophisticated such as Microsoft Excel, Weka, etc. Here we will examine how simple decision trees and random forests are constructed, how they can be applied to data sets and what their strengths and limitations are.

### 3.1 Use of Decision Trees

A decision tree comprises a set of decision rules that are used to classify observations into their outcome classes by the values of their inputs. They can also be used to predict the outcome class of a new observation. Following the creation of a decision tree, the new observation data is run through the tree, which functions like a flowchart, where decision rules "fork" observations into different branches. In the decision tree, the decision rules are the "nodes" that determine the branch that the observations falls into. This branching at the nodes occurs in succession until the observation is eventually

classified into an outcome class at the leaf level. The inputs that specify the decision rules comprise the algorithm of the decision tree.

Figure 1 depicts a decision tree showing the algorithm in predicting a new customer's decision to purchase a cell phone with the newest technology. In this hypothetical decision tree, the observations can be fully classified (i.e., each end node is homogeneous) by the inputs of age, condition of existing phone, and presence of current sales promotions. This means that if all these three pieces of information are known about a new observation, the behavioral outcome of the new customer can be predicted. That is, a new customer will purchase the phone with the latest technology if (i) s/he is below 50 years old and has a phone in poor condition, or (ii) if s/he is below 50 years old and is offered a sale promotion when his/her phone is in good condition.

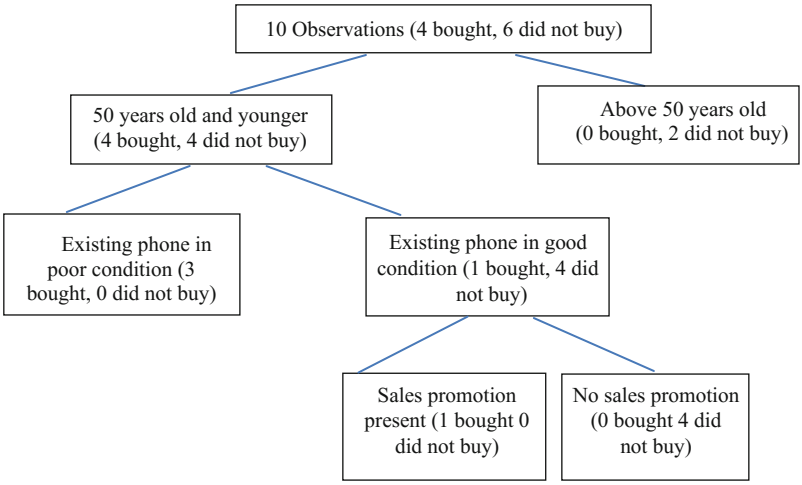


**Fig. 1.** Example of a decision tree

### 3.2 Constructing a Decision Tree

In constructing a decision tree, a greedy splitting approach is usually adopted. This process examines the input variables and chooses a tree path that minimizes a particular cost function. Classification utilizes the Gini cost function, while regression (i.e., regression trees) utilizes the sum of squared errors (SSE) cost function [5]. Once a basic decision tree is constructed, it is typically pruned in order to prevent overfitting of the data. Overfitting occurs when the model accurately assesses the training data but fails to accurately assess test data. Pruning is a means to combat this problem by removing each leaf node one by one, evaluating the effect on the cost function after each removal. There are many means to prune a decision tree, but a typical rule of thumb is that the smaller the decision tree, the less likely the overfit, and the more likely it is to be successful with the test data.

### 3.3 Random Forests

Random forests, or random decision trees, are ensemble methods involving multiple decision trees. Random forests essentially combine multiple decision trees to create multiple classifiers or predictors. The data will then be classified by the mode of these decision trees. Random forests are able to overcome error associated with using only one decision tree by establishing randomness across multiple trees. In creating a random forest, each decision tree uses a random selection of data and a random selection of variables [6]. This allows the trees in a random forest to examine a subset of the data while not focusing on all of the training data. The random forest examines all of the training data by utilizing numerous random decision trees. This method permits overlap among the trees but also prevents the classification or prediction to be made solely by one decision tree.

### 3.4 Data Required

Decision trees and random forests can be built from various types of data. Large data sets can be easily classified by a larger tree that has numerous nodes or splitting points [5]. Not all decision trees have binary splits; any number of splits may occur. For instance, one variable labeled "age" may split data into "50 years old or younger" and "Above 50 years old," or be characterized as having four age classes from "0–20 years old," "21–40 years old," "41–60 years old," and "Above 60 years old." Decision trees do not require the dataset to be complete and can also be constructed if there are missing data for some observations [7]. Due to the random selection of data and variables of random forests, if a variable in a data set is omitted, some decision trees within the random forest may not execute while the random forest as a whole will still produce a reasonable classification or prediction.

### 3.5 Strengths

One of the most beneficial attributes of decision trees is that they can be easily interpreted through graphics. Although this may become difficult with larger trees, smaller trees can be easily described through a simple diagram and explanation [5, 8, 9]. Curram [9] even explains that this may contribute to insight into factor relationships. Another positive attribute of decision trees is that data does not need to be transformed in any way, as there are no assumptions about the normality of the data or the underlying distribution of the data. Nodes within a decision tree can evaluate both quantitative and qualitative measurements without transforming one into the other. Due to the randomness of a random forest, any inaccuracy of one decision tree can be overcome by numerous other decision trees. This overlap causes each individual decision tree to be less robust while permitting the whole forest of decision trees to be fairly robust. Lastly, decision trees mimic the actual decision-making process of humans [9].

### 3.6 Limitations

The largest limitation of decision trees is their potential overfitting of the training data. This becomes a problem when new test data is applied to the decision tree. The tree is not broad enough to encompass the new test data even though it corresponds to the training data very well. Another limitation of this model is the technique utilized in pruning the decision tree. Mingers' [10] experiment analyzed the effects of five different pruning techniques on a decision tree. The experiment found that there were significant differences between the methods. In pruning a decision tree, the resulting output may be altered depending upon which method is utilized. As for random forests, the necessity to randomize what data subset and variables are used requires some configuration and prior programming.

## 4 K-Nearest Neighbor

Not to be confused with K-means clustering, an unsupervised clustering technique, the K-nearest neighbor is a supervised, classification technique that is one of the simplest machine learning techniques. This model is typically used to characterize a new observation based on data that it most closely resembles or relates to. This machine learning technique can answer how we should classify data that does not have binary characteristics. There are numerous nearest neighbor algorithms to examine but specifically we will focus on the K-nearest neighbor algorithm. This model of machine learning can be developed with numerous software programs. Here we will examine how K-nearest neighbor works and what strengths and limitations are present in such models.

### 4.1 Use of K-Nearest Neighbor

The easiest way to visualize a nearest neighbor problem is through a two-dimensional visualization. Note however, that a nearest neighbor model can be applied to data sets with any number of variables. Figure 2 depicts a visualization of a K-nearest neighbor model with two dimension variables $X_1$ and $X_2$. There are two classes present: orange circles and green squares. Both of these classes have already been populated through training data. The question here is should the test data, the yellow triangle near the center, be classified as a green square or an orange circle? Since this is a K-nearest neighbor model, the value of K can be manipulated. Should the value of K equal three, the yellow triangle would be classified as an orange circle since the majority of its three closest neighbors are orange circles. However, if K were to equal six, the majority of the six closest neighbors to the triangle are now green squares. Therefore, the value of K can drastically alter the results of the test data.

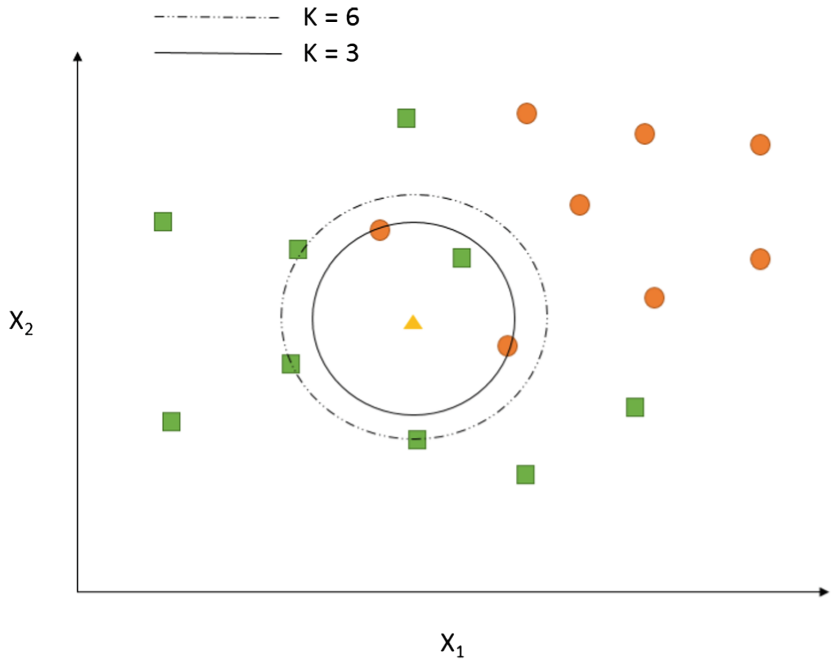


**Fig. 2.** Simple K-nearest neighbor Visualization

### 4.2 Constructing K-Nearest Neighbor (K-NN)

One of the most difficult questions to answer when conducting a K-nearest neighbor algorithm is what the value of K should be. Typically K is a smaller number due to the fact that as K increases towards the number of training points, the test point will more likely be classified as the class with the most training points. Also, a smaller K-value will result in overfitting while a larger K-value may result in oversimplification [10]. The other primary area of concern when constructing a K-nearest neighbor algorithm is determining what distance measure to use [11]. The most common distance measure is the Euclidean distance measure. However, if distance to the K-NN is large, there is an increased likelihood that an outlier would be included in the computation. Lastly, the variable ranges will need to be normalized, typically between the values of 0 and 1. This ensures that one variable does not affect the distance from the test data to the nearest neighbors more than another variable due to its range of values.

### 4.3 Data Required

Like decision trees, the K-nearest neighbor algorithm is non-parametric and does not rely on any assumptions on the underlying distribution of the data. This makes it a valuable algorithm especially when there is little or no prior knowledge of the data's distribution. When examining applicable data to run in a K-nearest neighbor algorithm, it is important to first look at the number of dimensions in a particular data set.

The experiment by Beyer and her colleagues [12] experiment suggests that as the number of dimensions per data set increases, the change in distance between the nearest neighbor and farthest neighbor approaches 0. This limits the data being tested to only a certain number of variables. However, data that represents more variables than computationally applicable may be able to omit some variables. The data will also need to be transformed into a set of vector inputs [12].

### 4.4 Strengths

Even though Nearest Neighbor is one of the simplest machine learning techniques, it is still a very strong model used to classify data. One of best qualities about Nearest Neighbor is that there are numerous enhancement techniques to classify each data set better. One example that Weinberger and colleagues [13] examined is that of a margin that takes into account other classes that are trying to invade the distance area. Another commonly used technique to weigh the distance from the test point to the K-nearest neighbors is by assigning those neighbors with a value of (1/d). Also, by increasing the K-value, the model becomes more stable at the cost of potential oversimplification. Other advantages of the KNN algorithm include its ability to handle "noisy" and large training data, and the ease it which it "learns" [12].

### 4.5 Limitations

As depicted in Fig. 2, the greatest challenge with K-nearest neighbor is determining what the value of K should be. Too small of a value results in overfitting the data while too large of a value creates an oversimplified model. Although Euclidean distance might be the obvious choice for most K-nearest neighbor models, the distance metric may manipulate the results of different nearest neighbor models. Furthermore, K-nearest neighbor tends to be computationally expensive for many applications and result in large data sets with numerous dimensions to take a long time to compute [14]. The KNN algorithm may also be easily influenced by irrelevant attributes and tends to run slowly due to its computation complexity [12].

## 5 Artificial Neural Networks

Artificial Neural Network (ANN) is another machine learning tool that researchers use to solve problems. Artificial neural networks are so named because they purportedly mimic the way the human brain processes inputs and responds with an output. They comprise networks of neurons/nodes organized into layers, and synapses which are the connection in between layers [15, 17]. Then neurons are where the data are, and the synapses are the connections within the data. The result of processing with the ANN is an output or prediction, such as the "watch next" option on YouTube or the ads that appear on your web [16]. Other applications of ANNs include Amazon's product recommendations.

### 5.1 How Artificial Neural Networks Work

There are many kinds of ANN architectures, one of which is the Multilayer perceptron (MLP). An MLP ANN contains at least three layers of neurons/nodes (i.e., there can be more than one hidden layer): (i) the input layer which are the predictive variables, (ii) a hidden layer (so called because they are not "visible" since they are neither the predictors/inputs nor the outputs/outcomes) which works on the data from the previous layers, and (iii) an output layer which is the outcome of the prediction. Between each adjacent layer are synapses/connections which accept the data from the multiple neurons activated from the preceding layer. The data, combined with weights, passes through an activation function, and the result determines which neurons in the subsequent layer get activated. The prediction is the result of neurons that are activated at the final output layer. During training, which can either be only in a feed-forward activation flow (uni-directional) or also include the backward propagation of errors (bi-directional), the ANN will adjust the weights and activation function of the hidden and output layers (processing layers) such that the output would most closely match the values of the outcome/target variable [17]. There are many activation functions; these include linear, step, sigmoid, linear threshold between bounds, etc. Besides the MLP, other ANN architectures include the Radial Basis Function Network, Recurrent Neural Network, the Hopfield Network, the Long/Short Term Memory Network, etc. [18].

### 5.2 Data Required

There is almost no limit to the type of data that can be used with ANNs, so they are used to solve a variety of problems. Data can range from being demographical information in the prediction of political affiliation, to being parts of an image for an image recognition task, to inputs from an audio file for a speech and language recognition task. However, because ANN requires numeric data, this often requires some types of data to be coded. For instance, categorical data such as "male" and "female" can be coded as "0" and "1" respectively, or the image may be coded as saturation levels in different locations on a matrix. Often, the way the data is coded impacts the quality of the prediction by the ANN. In addition to encoding, data preparation also involves standardization, which is especially necessary when nonlinear activating functions are applied. Standardization involves coding categorical or nominal data, and normalizing data.

### 5.3 Usability of Artificial Neural Networks

The artificial neural network can solve optimization problems, estimation problems, and cost functions just to name a few [15]. However, experts have leaned towards the idea that artificial neural networks "learn from the observed data" and act accordingly —similar to how a human brain functions [15]. Today, government agencies, small and large businesses use artificial neural networks in the most sophisticated ways possible. For example, credit card companies rely on this tool to detect any unusual activities to protect their clients and avoid fraud [16], and shipping companies use artificial neural networks to determine the fastest and most efficient route to deliver a product [16]. In

this case, shipping companies tell the artificial neural network where the product is headed and this machine learning tool takes this information, analyzes potential routes, and suggests the most cost-efficient and direct route. Other applications of ANNs are tasks in medical diagnosis, machine translation, etc.

### 5.4 Strengths and Limitations

Artificial neural networks' biggest strength is its versatility. Artificial neural networks can be applied to almost any scenario or problem. They implicitly address the problem of feature selection, which is one of the most challenging problems in machine learning and any prediction.

Nevertheless, just like any other machine learning tool, artificial neural networks have their limitations as well. To function at a high level, an artificial neural network must gather a large data set to operate [15]. This shortcoming can be problematic in domains where very little research has been conducted, such as extremely specific fields of study. Among the machine learning algorithms, ANNs have the weakest theoretical foundation which impedes their explanatory value since it is virtually impossible to work out the typology of the neural network – we rarely know what goes on in the hidden layers.

## 6 Challenges to Understanding Behavior in New Research Domains

These machine learning techniques are useful for understanding underlying patterns and can help with the prediction and classification of behaviors even when there is limited knowledge about relationships and phenomena in a domain[1]. For instance, an ANN may be able to predict task-induced workload from a host of predictors that include data on demographics, personality, task characteristics, medical history, etc., but it is less useful in extracting any new constructs or measures that have no theoretical foundation. For such newer domains, where there are fewer established theories and research findings, these techniques are less able to shed light on the domain itself. To help increase knowledge and understanding in a newer domain, researchers often draw upon ideas and concepts from related domains. For instance, research in the domain of human-robot teaming has included constructs such as trust and teaming, both of which are found in social and organizational psychology, and have yielded fruitful research in human-robot teaming. This process of identifying related domains, importing constructs and research ideas from these more established and related domains is useful in spurring research in a relatively new domain and may benefit from the application of set theory, graph theory, and network analysis.

[1] In this paper, we are loosely defining "Domain" as a system comprising the increasing aggregation of units, parts, and subsystems that are interconnected and interrelated [19]. Examples of domains include the mining, nuclear plant, space missions, marine transport, power grids, manufacturing, assembly-line production domains [19].

### 6.1 Set Theory

Set Theory provides a way to think about elements and how they may be organized (sets). In the context of assessments in the behavioral sciences, Set Theory can be used to describe constructs (sets) in terms of their operationalization or how they are measured (elements) [20]. Since constructs in new domains tend to be less clearly defined, and their operationalization less standardized, the ability of Set Theory to deal with such ambiguity or "fuzziness" would enable even such constructs to be analyzed. Applying Set Theory may contribute to the understanding of construct/set similarity, and the degree of abstraction and generalization of constructs [20, 21].

### 6.2 Graph Theory

A graph is a diagram representing a system of connections or interrelations among two or more things by a number of distinctive dots, lines, bars, etc. [22]. In the assessment context, the node in the graph may be a representation of a construct set containing its measures, while a link of the graph denote the relationships among constructs. Alternatively, the nodes could be measures with the links representing the relationship among measures (see Fig. 3). Hence, Graph Theory would allow analyses of different graphs that may comprise constructs or measures, or both.

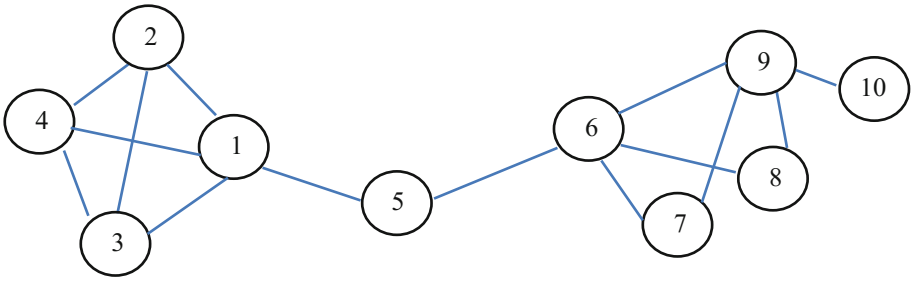


**Fig. 3.** A graph showing links and nodes

### 6.3 Network Analysis

Network science has been used in research to understand social behavior and networks. According to Lubell [23], these include:

- Identifying central individuals who can help spread ideas and behaviors (top influencers)
- Identifying disconnected individuals who need to be brought into social communities
- Identifying key social relationships that should be cultivated in order to integrate and bring together diverse communities

Network analysis and network science enable analysis of the relationships among multiple interconnected nodes and/or clusters of nodes. The nodes can be

representations of sets of various constructs and their measures, or even sets of relationships between constructs. Taking a simple example of the nodes in Fig. 3 representing constructs, with nodes 1 through 4 being constructs in a domain, and nodes 6 through 10 being constructs in another domain. Although construct/node 5 is not linked to as many construct as most of the other constructs, it enables constructs in different domains to be connected. In this case, construct/node 5 may be considered an "influential or central construct" and in network science, its *node centrality index* would be highest of all the nodes. Conversely, construct/node 10, being only linked to one other construct, may be construed as a "disconnected construct."

## 7 Summary and Conclusions

This paper presented a few machine learning techniques available for data analytics. These strategies allow machines to classify, prescribe, suggest, and predict behavioral outcomes. Dependent on the situation, one technique might be more advantageous to use over another considering each technique has its own strengths and limitations. For scientists seeking to understand behaviors within and across domains, especially new domains with few established constructs and theories, methods from mathematics and network science such as set and graph theories can be beneficial. These techniques can contribute to a "bottom-up" approach, complementing the traditional "top-down" approach, to assessments and research that is more theory-driven.

**Acknowledgements.** This research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-15-2-0100. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied of the Army Research Laboratory of or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

## References

1. Jacobson, R.: 2.5 quintillion bytes of data created every day. How does CPG & Retail manage it? https://www.ibm.com/blogs/insights-on-business/consumer-products/2-5-quintillion-bytes-of-data-created-every-day-how-does-cpg-retail-manage-it/
2. Marr, B.: https://www.forbes.com/sites/bernardmarr/2015/09/30/big-data-20-mind-boggling-facts-everyone-must-read/#631ddd5017b1
3. Burford, C., Reinerman-Jones, L., Teo, G., Matthews, G., McDonnell, J., Orvis, K., Riecken, M., Hancock, P., Metevier, C.: Unified Multimodal Measurement for Performance Indication Research, Evaluation, and Effectiveness (2018)
4. Edward, J., Bagozzi, R.: On the nature and direction of relationship constructs and measurement. Psychol. Methods **5**, 155–174 (2000)
5. Marr, B.: Supervised V Unsupervised Machine Learning - What's The Difference? vol. 6. https://www.forbes.com/sites/bernardmarr/2017/03/16/supervised-v-unsupervised-machine-learning-whats-the-difference/#5ae5a61b485d

6. Brownlee, J.: Classification and Regression Trees for Machine Learning. http://www.machinelearningmastery.com
7. Malakar, G.: What is Random Forest Algorithm? A Graphical Tutorial on How Random Forest Algorithm Works? https://www.youtube.com
8. Mitchell, T.: Decision tree learning. Machine learning, pp. 52–80. WCB/McGraw-Hill, Boston (1997)
9. Curram, S.P., Mingers, J.: Neural networks, decision tree induction and discriminant analysis: an empirical comparison. J. Oper. Res. Soc. **45**, 440–450 (1994)
10. Mingers, J.: An empirical comparison of pruning methods for decision tree induction. Mach. Learn. **4**, 227–243 (1989)
11. Wikipedia: K-nearest neighbors Algorithm. https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm
12. Beyer, K., Goldstein, J., Ramakrishnan, R., Shaft, U.: When is "nearest neighbor" meaningful? In: Beeri, C., Buneman, P. (eds.) ICDT 1999. LNCS, vol. 1540, pp. 217–235. Springer, Heidelberg (1999). https://doi.org/10.1007/3-540-49257-7_15
13. Weinberger, K.Q., Saul, L.K.: Distance metric learning for large margin nearest neighbor classification. J. Mach. Learn. Res. **10**, 207–244 (2009)
14. Bhatia, N.: Survey of nearest neighbor techniques. Int. J. Comput. Sci. Inf. Secur. **8**, 302–305 (2010)
15. Wordpress: The Shape of Data: K-nearest Neighbors. https://shapeofdata.wordpress.com/2013/05/07/k-nearest-neighbors/
16. Wikipedia: Artificial Neural Network. https://en.wikipedia.org/wiki/Artificial_neural_network
17. Templeton, G.: Artificial neural networks are changing the world. What are they? https://extremetech.com/extreme/215170-artificial-neural-networks-are-changing-the-world-what-are-they
18. Narula, G.: Machine learning algorithms for business applications- complete guide. https://www.techemergence.com/machine-learning-algorithms-for-business-applications-complete-guide/
19. Perrow, C.: Normal Accidents: Living With High Risk Systems. Basic Books, New York (1984)
20. Nelson, E.: Internal set theory: a new approach to nonstandard analysis. Bull. Am. Math. Soc. **83**, 1165–1198 (1977)
21. Stoll, R.R., Enderton, H.: Set Theory. https://www.britannica.com/topic/set-theory
22. Zweig, Katharina A.: Graph theory, social network analysis, and network science. Network Analysis Literacy. LNSN, pp. 23–55. Springer, Vienna (2016). https://doi.org/10.1007/978-3-7091-0741-6_2
23. Lubell, M.: Three hard questions about network science. http://environmentalpolicy.ucdavis.edu/node/292