Unclassified **DAF/COMP/WD(2017)50**

Organisation for Economic Co-operation and Development

7 June 2017

English - Or. English

DIRECTORATE FOR FINANCIAL AND ENTERPRISE AFFAIRS
COMPETITION COMMITTEE

It's a Feature, not a Bug: On Learning Algorithms and what they teach us - Note by Avigdor Gal

Roundtable on Algorithms and Collusion

21-23 June 2017

This paper by Avigdor Gal was submitted as background material for Item 10 at the 127th meeting of OECD Competition Committee on 21-23 June 2017.

The opinions expressed and arguments employed herein do not necessarily reflect the official views of the Organisation or of the governments of its member countries.

More documents related to this discussion can be found at

www.oecd.org/daf/competition/algorithms-and-collusion.htm

JT03415453

This document, as well as any data and map included herein, are without prejudice to the status of or sovereignty over any territory, to the delimitation of international frontiers and boundaries and to the name of any territory, city or area.

It's a Feature, not a Bug: On Learning Algorithms and what they teach us

Note by Avigdor Gal*

1. Introduction

1. Algorithms, the recipe of any computational program, are considered to be an essential component of computer science[1]. Algorithms may be simple, such as those for sorting integers, or highly complex. What they all share is an exact sequential set of commands that allow a computer system to perform computational tasks. The efficiency of an algorithm is affected, inter alia, by the availability of data structures, which allow storing and retrieving data efficiently, and by the level of its complexity, which determines the time it may take an algorithm to perform its task in a worst-case scenario (which may be forever).

2. Algorithms come in various shapes and forms. Some algorithms are monolithic, encompassing every aspect of the computational process. Others may perform only part thereof, leaving other tasks to other algorithms. Some are designed for sequential execution while others are meant to be run distributed and in parallel. Yet, their main essence remains the same: given an appropriately designed input, a sequence of commands is performed over this input to generate an output in a clearly defined format[2].

3. Recent years have seen an increased interest in a specific class of algorithms, Machine Learning (ML) algorithms[3]. Such algorithms are at the backend of what many believed to be part of human beings realm alone. IBM introduced Watson, a winner of the American popular show Jeopardy[4]. Google introduced AlphaGo, a Go winner[5]. Autonomous driving vehicles[6] are no longer science fiction, language translators have shown an immense improvement, and we all have our personal assistants listening and reacting to our vocal commands over the phone. The impact ML algorithms have on the new digital economy stretches far beyond what many sceptical researchers thought possible. So much so that ML algorithms have invoked many primeval concerns and fears brought forward in well-known movies such as the Matrix[7].

* Professor at Faculty of Industrial Engineering & Management, Technion – Israel Institute of Technology. The author would like to thank Arik Senderovich for useful comments.

[1] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. Introduction to Algorithms. The MIT Press, 2nd edition, 2001.

[2] R. A. Wilson and F. C. Keil. The MIT Encyclopedia of the Cognitive Sciences. The MIT Press, 1999.

[3] M. J. Kearns and U. V. Vazirani. An Introduction to Computational Learning Theory. The MIT Press, 1994.

[4] https://www.youtube.com/watch?v=WFR3lOm_xhE

[5] https://en.wikipedia.org/wiki/AlphaGo

[6] https://en.wikipedia.org/wiki/Autonomous_car

[7] http://www.imdb.com/title/tt0133093/

4. The purpose of this short note is to enhance the understanding of what it means for an algorithm to learn, who teaches the algorithms, what they learn, and to what extent they can share with us what they have learned. This understanding can serve as a starting point for regulators to better understand what can and cannot be demanded from market players using algorithms. The discussion focuses in particular on machine learning algorithms. The reason is three-fold. First, the use of such algorithms is becoming more commonplace. Second, they often obscure the reasons for their outcomes, making it easier for the user to claim he was not aware of what the outcome would be. Finally, such algorithms can be employed by regulatory authorities to study other algorithms or to determine what has driven the market outcome.

2. Algorithms Learn Whatever Data Teach Them

5. Machine learning algorithms are first and foremost, algorithms. Accordingly, they re- quire a well-structured input, they follow a clearly defined sequence of commands and they produce output. So what exactly sets ML algorithms apart from other algorithms? Have the advancements in ML that took place in the past decade created a paradigm shift? The answer to these questions may come as a surprise. It is not the algorithmic practice that has significantly changed, although new and more efficient algorithms are invented all the time. It is the data that made the major impact here. To be more specific, it is the information explosion (a.k.a Big Data) of recent years, which includes technological advancements such as the Internet of Things (accumulation), cloud computing (management), and data mining (analytics)[8], that has made the main difference for using algorithms in our marketplace.

6. Big data is commonly characterized via a set of "V"s, out of which three became mostly prominent. Big data is characterized by volumes of data that are gathered, managed, and analysed. Velocity refers to the speed in which data accumulates and the amount of change to the data. Big data variety refers to the availability of heterogeneous data sources.

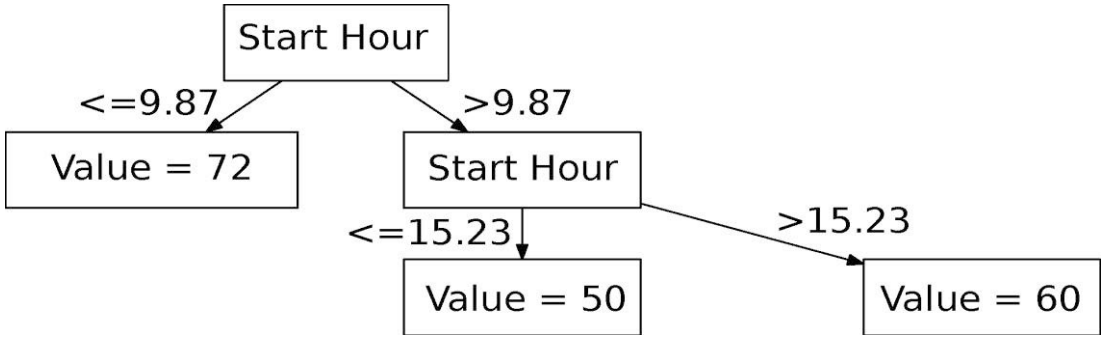
7. How has this information explosion affected the performance of algorithms? Given sufficient amounts of data (which we can collect or purchase) and sufficient amounts of computing resources (which we can purchase), algorithms can learn from data. Generally, the larger the volume, velocity, and variety from which the algorithm can learn, the better it becomes at its tasks. Learning from data is generally separated into supervised and unsupervised learning. In supervised learning a part of the data, called the training set, is annotated with the "correct" answer to the learned task. This training set is used to create a model for a correct answer. Then, given a new test set, the algorithm uses the learned model to determine an answer for it. In unsupervised learning, no correct answers are provided. Instead, the algorithm resorts to pre-coded measures that serve as good indicators of "success".

8. There are many learning tasks ML algorithms may fulfil. For illustration purposes, consider the following set of learning task examples. In what follows we use the term data instance to identify one sample datum from the data the algorithm receives, e.g., a report on gasoline price:

[8] A. Gal. Tutorial: Uncertain entity resolution. PVLDB, 7(13):1711{1712, 2014.

- **Classification**: given a fixed set of categories, the algorithm aims at correctly associate a data instance with a category. For example, given a set of TV shows a viewer watches, the algorithm aims at classifying a viewer as belonging to one of the following categories: baby, child, teenager, young adult, adult, and senior.
- **Clustering**: organizing a set of data instances (that is, one sample datum from the data the algorithm receives) into one group, based on some sense of similarity. For example, putting together all the bank accounts of a single customer. Unlike classification, here the set of categories is not pre-determined.
- **Prediction**: providing an assessment for a future value. For example, predicting what will be the price of gasoline at a competitor's gas station and when would the next price increase occur. To illustrate the prediction task, consider Figure 1, which presents graphically the outcome of an ML algorithm, which is part of a specific category of learning algorithms called decision trees. Algorithms in this category create a decision tree, which is constructed from decision nodes and edges that direct the decision from one node to the next. Given a new data instance, the algorithm starts from the root of a tree and follows the relevant node, which determines what should one do. For example, Figure 1 illustrates a decision tree that predicts an outcome (say, wait time) based on a single feature, called Start Hour. If the start hour is before 9:52, the value is predicted to be 72. After 9:52 and before 15:13, the value is predicted to be 50 and afterwards, 60.

Figure 1. Decision Tree Example



3. Algorithms Learn from Features

9. Now that we have established that ML algorithms learn from data, let us focus on how, in principle, such algorithms learn. For that consider one typical way of formatting an input to be used by an ML algorithm, a format we shall refer to as a relational log, or simply a log[9]. A relational log is constructed from rows (also termed tuples), where each row is a data instance, as defined above. A row is constructed from a set of features (also termed attributes) where each feature is a specific type of data observation. As an example, consider a relational log that represents sale prices of gasoline in a gas station. A row in such a log may contain features like: day, time, station, gasoline type, and price. It may also contain features like temperature, road congestion, and even the colour of the flag on top of a competing gas station. Some features may be the result of mix-and-match

[9] A. Gal, A. Mandelbaum, F. Schnitzler, A. Senderovich, and M. Weidlich. Travel-ing time prediction in scheduled transportation with journey segments. Inf. Syst.,64:266{280, 2017.

of other features, using information-theoretic mathematical functions (like value smoothing functions) to which we cannot necessarily attach meaning in the domain of discourse.

10. ML algorithms determine which of the features at their disposal has the highest impact in forecasting a value that is close to the one conceived to be correct. Accordingly, a classification algorithm seeks to determine a set of features whose values can best determine the correct class for a data instance. Alternatively, for prediction, an ML algorithm may look for a linear combination of feature values (each with a different weight) that yields a value that is as close as possible to the value to be predicted.

11. In many of the ML algorithms it is easy to observe not only what they have learned (e.g., a category, a cluster, or a predicted value) but also how they reached their conclusion (which features were used in the learning process and how much weight what assigned to their values). To illustrate this point, let us look again at decision trees. The example in Figure 1 introduced a decision tree with a single feature, Start Hour, yet decision trees may be constructed from multiple features. Take, for example, an algorithm that determines the prices to be charged for gasoline. Learning from the data, the algorithm may base the price on features such as road congestion and the external temperature, the effects of which on demand for gas is not easily observable without careful analysis of the data. Once a complete path of the tree is completed, a leaf node is reached with a decision (e.g., increase gasoline price by 2 cents).

12. Nowadays, many consider deep learning[10], a class of computerized neural networks- based algorithms, to be a separate, subclass of ML algorithms. This is true. One of the things that sets them apart from other ML algorithms is their limited ability to explain their decision making. In deep learning, features are created as a (possibly complex) computation over multiple features, making such algorithms' decision-making hard to explain.

4. Regulating with Features

13. More and more decision-making is transferred to algorithms. Algorithms advise us which path to take to reach our destination. Algorithms tell us which TV series we should watch next, which book to purchase, and how generous we should be when attending a wedding. Algorithms advise on health issues and recommend physical activities. Algorithms also become prominent in business decision-making, enabling an efficient analysis of relevant data.

14. This raises the following question: to what extent can the regulator expect to under- stand the decisions made by algorithms in order to determine any wrongdoing? Equipped with the understanding of how ML algorithms operate, we can now provide educated answers to this regulatory question. An important element that affects the regulatory process involves the level of access to the algorithm. Such access can be placed on a spectrum, from black box to white box. Black box access[11] means that a regulator will be able to execute the code and gain access to its output, without accessing

[10] L. Deng and D. Yu. Deep learning: Methods and applications. Foundations and Trends in Signal Processing, 7(34):197{387, 2014.

[11] https://en.wikipedia.org/wiki/Black-box_testing

the code itself. The other extreme of the spectrum is the white box access[12] where a regulator is allowed to access and analyse the code itself. Black box access allows organizations to keep their trade secrets, keeping in mind that the algorithm's comparative advantage may not necessarily lie in the "science" of the algorithm but rather in its "engineering," including the way the algorithm is tuned and the methods that it uses for improved performance. Therefore, the terms under which access to the algorithm is granted should be set to allow regulatory oversight, without risking the organization's intellectual property or trade secrets.

15. The second element that affects the ability to regulate algorithms, involves the access of the regulatory body to the dataset used by the algorithm. This dataset is important for two reasons. First, once the dataset and the algorithm are available, regulators can check whether a specific outcome was indeed the outcome of the specific algorithm. To do so, regulators can use a tool borrowed from empirical sciences, which is known as repeatability. Repeatability measures the "closeness of the agreement between the results of successive measurements of the same measure and carried out under the same conditions of measurement."[13] It is worth noting that for repeatability, using a black box access is sufficient.

16. Repeatability provides a "sanity check," to validate the reports of a firm. Yet, it still leaves us with questions regarding the actual decision-making of the algorithm. This leads to the second reason why access to data is important. Where a firm uses an algorithm that exposes its selection, e.g., in the form of a decision tree, the regulator can determine whether the algorithm uses features that it considers legitimate. If, for example, a firm's algorithm uses temperature and congestion in determining gasoline prices, the regulator may be less concerned than if the main feature is the colour of flag of a neighbouring gas station (which may be a covert channel for collusion). Where such features are unclear, the availability of the data enables the regulator to run a set of existing ML algorithms on the data and on the outcomes observed, in order to determine which features were used to reach the outcome. Such know-how requires familiarity with state-of-the-art ML algorithms and the ability to use existing frameworks (e.g., Google's TensorFlow[14] and Python's Scikit-learn[15]). Put differently, by using publicly available ML algorithms, it is possible to re-create the results yielded by the algorithms (measured by using statistical measures such as RMSE[16]) and gather from the results the prominent features used by the firm's proprietary algorithm.

5. Conclusions

17. We have provided an introduction to learning algorithms, explained how algorithms learn from data and introduced the concept of features, a central concept to the way algorithms learn. We then discussed the possibilities of regulatory agencies to

[12] https://en.wikipedia.org/wiki/White-box_testing

[13] http://www.pitt.edu/~jdnorton/teaching/1702_jnrsnr_sem/docs/Reproducibility/reproducibility.html

[14] https://www.tensorflow.org/

[15] http://scikit-learn.org/stable/

[16] R. J. Hyndman and A. B. Koehler. Another look at measures of forecast accuracy. International Journal of Forecasting, 22(4):679 { 688, 2006.

understand which features were used in decision-making processes. Knowing which features were used by an algorithm reveal the basic elements that were used as the basis for the algorithm's decision-making. Equipped with such knowledge a regulatory body may decide, for example, to forbid the use of certain features.