Do not Blindly Trust Your Data

Filip, René

SS 2018

Abstract

In Data Science and Machine Learning one does not simply apply an algorithm and reads out the result to solve a problem. Often, it includes tasks like data collection, data cleaning and data interpretation. In all these steps, many things can go wrong and manipulate the final result. Because people trust more and more the results of an algorithm, it is important to understand what can go wrong, why a result might be biased and how such a wrong result affects other people. In this paper, we will analyze some factors and argue about their importance and influence. At the end we reason why data science is more than only applying mathematical operations.

1 Introduction

Today, the role of data science and machine learning becomes more and more important because it gives us either new insights or it enables us to write programs that sometimes even outperform human beings. In the industry, there is a high demand of data scientists that can effectively explain given data sets and solve certain tasks like classification, prediction and generation. Machine Learning already proved its strength in industries like pharmacy and health Faggella [2018]. Usually, data is extracted from some source, cleaned from errors or noise and then fed into a fitting algorithm producing a result which needs to be interpreted at the end. However, many things can go wrong if one is not careful enough. For any scientist or engineer it is important to understand the challenges in each step of the pipeline in order not to give wrong or even dangerous solutions.

We can ask ourselves: What are the things we need to be aware of in order to make our analysis as accurate as possible? Why is there sometimes a bias in our data sets and what can we do against it? And what are the consequences of our (wrong) results on the society? In this essay we are going to answer these question by taking a closer look at algorithms and their unexpected behavior. Then, we observe an bias if experiments are conducted in public or

anonymously. Last but not least we raise awareness of being critical towards publicized papers.

2 Dangers of Trusting an Algorithm blindly

Algorithm can refer to many different things depending on which person one talks to Peters [2016]. A computer scientist might refer to the definition of an algorithm using a Turing machine while a journalist might refer to the behaviour of a certain software. Nevertheless, in both cases one needs to be aware of what the algorithm exactly does and what not.

2.1 ML Algorithms do not Always Solve the Task

An algorithm solves a given problem by performing predefined steps in a certain order. The result of an algorithm is a solution to the given problem. There are many types of algorithms. Deterministic algorithms like the sorting algorithm "Bubble Sort" will *always* provide the same output for a given input. In contrast, a probabilistic algorithm like the "Fast Johnson Lindenstrauss Transformation" returns a solution that only holds with a certain probability. Machine Learning algorithms like "Linear Regression" describe another type of algorithms which differ even more: Let us assume we want to predict the housing price of a flat in Munich if one gives us the square meter size of it. The algorithm *itself* does not give us an answer to that but instead returns a mathematical model which needs to be asked instead. In order to do so, the algorithm needs "training data" to train the model that we want to return. Otherwise, the model does not know anything about Munich or even about housing prices in general and would return random-like estimates instead.

This is a problem. The first two categories of algorithms we mentioned above did not rely on training data because they were solving the problem explicitly. However, machine learning algorithms generate a solution by looking at existing solutions and then try to make one on their own, implicitly. Suddenly, the performance of a learning algorithm depends on the training data set.

Solving a problem implicitly is not a bad thing per se. Actually, it is often much more easier to solve a problem that way than describing a solution explicitly. But, it adds a danger that one can easily overlook: The algorithm itself always produces a model (even for bad or wrong training data) and just because it returns does not mean it also returns the right solution! While for "classical" algorithms there often exist proofs guaranteeing correctness, users of machine learning algorithms must be aware that these do not exist in the same way. The user must not have a false sense of security just because the algorithm returned successfully and the underlying challenge is how to evaluate a model.

There are ways to perform the evaluation and we will look at them shortly. But the consequence of this is quite depressing: In the real world there might be bad machine learning models that do not act the way which was intended. For example, in Dailymail.com [2017] a soap dispenser was trained on detecting hands in order to dispose soap. It turned out that the model was trained on a data set that only contained white colored hands and at the end it could not detect the hand gesture by a black person. If we translate this problems where human life is in danger, the magnitude of it becomes clear. Imagine a self-driving car that cannot detect a group of humans because the model expected a different physical appearance. These kind of issues are not obvious from the start and might surprise.

For evaluation, papers often rely on Cross Validation (CV) to evaluate a model. Simply spoken, the technique works by splitting the source data into a training data set and a test data set. Obviously, the training only happens on the training data set and after training one can evaluate the performance by using the test data set. If it performs well, we usually say that the model generalizes and recreates the true data distribution. There is a problem though: If the initial data set is already biased, the evaluation is as well. The soap dispenser from the example above failed to identify hands by black people. Cross Validation did not solve this issue because the test set did not contain black hands.

2.2 Algorithms at a Broader View

More generally, when talking about an algorithm one can refer to the combination of several algorithms and data transformations instead of a specific one. When people criticize some piece of technology in big companies like Facebook or Google, the companies often defend themselves by stating an algorithm was responsible for a certain action and not the company or its employees. For example, the Google auto-complete functionality sometimes relates negative or damaging attributes with certain people if enough people search for such a term. Suddenly, the term "Bettina Wulff" was standing next to "Escort" (engl. escort) or "Rotlicht" (engl. redlight) Kuri [2013]. Google claimed not to be responsible for this behaviour because the algorithm behind it was doing the intended work.

The more complex the system becomes, the more difficult it gets to detect errors. At some point, the system could become so difficult to understand that no one can successfully fix errors or find out a root cause of a problem. At the same time, the end user does not know how easy or complex a system is and if it works in most of the cases, the user will trust the system without much hesitation.

2.3 Machine Learning's Unexpected Behaviour

Reinforcement Learning is a category of machine learning algorithms where an agent learns a behaviour (a policy) by interacting with the environment. For each performed action it receives a positive or negative reward. After some time the agent learns which action will maximize the expected return in the long run. These kind of algorithms are often applied in robotics or in optimal control because tuning a complicated controller manually is not tractable. Instead, if a robot could learn how to behave by only telling him which actions are right and which are wrong, then it could learn by itself just like a child. This approach has been successfully applied many times, for example when a small helicopter learned how to flight stunts Ng et al. [2006] or when Google's "Alpha Go" program beat the world champion in a game of the Japanese board game Go Wang et al. [2016]. Interestingly enough, the computer sometimes comes up with solutions that were not expected by the human in the first place. In OpenAI [2017] the authors present unexpected policies in a boat game. It shows a policy that *increases* the reward but does not solve the actual problem at all.

One cannot simply compare the learning of a computer with the learning of a human. A human has years of previous experience in many different fields that he attained during his childhood. He can successfully apply experiences made in one area to another. For example, if we let a human play a new video game, then it usually does not take too long until the player understands the fundamentals and basics of it. In contrast, a computer has much more difficulties until it "understands" how to solve a problem. In most cases it does not have previous experience at all and doing transfer learning is even more difficult. While a human might already learn after 1 to 10 tries, a computer might need over a million game plays Mnih et al. [2013]. Because of that, the computer tries out actions which seem to be inefficient or bad by the human. However, some of these actions sometimes lead to a situation that the computer then exploits. The program can only see the reward it gets and if it receives a reward even when it is *not* solving the problem, it will also do so.

This is a problem because we do not know when the agent performs well behaved and when it does not. Imagine a robot that interacts with humans and suddenly performs action that are unexpected and dangerous. What can we do against it? Sufficient testing is necessary but not sufficient. It is well known that "program testing can be a very effective way to show the presence of bugs, but is hopelessly inadequate for showing their absence." - Edsger Wybe Dijkstra. In machine engineering more sophisticated methods already exist, like "Formal Verification". Here, an expert states properties one would like to formally (meaning mathematically) verify such that these properties hold. Or, in aviation each incident is carefully reported and analyzed in order to prevent future incidents of the same type. Thus, an aircraft manufacturing company can learn from another and always improve.

In Machine Learning we need to establish similar practices. In fact, researchers

already dealt with security in Reinforcement Learning, for instance in Garcıa and Fernández [2015]. However, Machine Learning Formal Verification seems to be still emerging.

3 The Factor of Anonymity

Data does not always come from machines. In psychology, researchers conduct experiments with humans and then observe their behaviours, feelings and thoughts. In Natural Language Processing (NLP) researchers rely on text written by humans, for example the Twitter feed for a given hashtag. Surveys determine the opinion of a group of people to a topic, for instance which party would people vote if there are elections next week. In all given examples the presence or absence of anonymity influences the results.

There exists different kinds of anonymity: In non-anonymous environments it is clear from the start for everyone who is participating. In the case of a (political) debate, all participants can refer to ones job positions, accomplishments, relations and previous statements issued by the participant. In addition, people will judge an opinion by a well known person (for example a politician in a high position) differently than an opinion by someone less well known. In contrast, in an anonymous forum one can only refer to the forum post one writes and sometimes it is not even possible to match previous post to one user. In the most extreme case of anonymity not even the state or any other organization can know the identity. This is often accomplished with special kind of encryption software like the Tor network. There are also mixtures of these degrees: On Twitter some profiles "verified" their true identity and they might interact with profiles which make it hard or impossible to fully identify.

Anonymity has advantages as well as disadvantages. On the one hand, it enables the user to fully express their opinions, regardless how conventional or contentious these are. The user does not have to fear social repression in the real world because it is not possible to link an opinion to the actual real world person. Also, peer pressure plays a minor role because in an anonymous setting because one acts alone and not as a group. Discussions can become much more honest because participants do not have to fear consequences like losing your job or facing legal consequences. In addition, the background of a person becomes unimportant and people solely judge the content one gives. Suddenly, the opinion of someone with a lot of influence in the real world is as important as the opinion of someone without. And, identity traits do not matter as well. People will judge an opinion by someone independent of race or gender due to the fact that they are not known.

On the other hand, anonymity can be responsible for abuse and misuse. People can impersonate other people and damage their reputation. On Twitter it is not too difficult to create an account that impersonates another one because Twitter

does not perform a sufficient identity check and anyone can create an account without verification. There are internet forums that do not require any registration like the Japanese text board "2channel". If a registration process does not exist, then it is very difficult to remove a misbehaving user from a website. Due to this fact, "internet trolling" is a problem and can damage a discussion. It even encourages illegal behavior as anonymity defers legal consequences.

Due to the characteristics of anonymity, it heavily affects human-generated data and one needs to be very careful whether anonymity should be or was part of an experiment. For instance, a data analyst might not notice that most tweets to a hashtag were generated by a computer or by only a small circle of people that utilize multiple accounts.

Sometimes, people propose to remove anonymity completely to prevent illegal behaviour, bullying and fake news. Regardless whether this can be archived from a legal perspective, we believe that this step would not remove the anonymity bias of data due to social repression. Stephens-Davidowitz [2017] argued that people generally lie in public and are more honest in anonymous settings. Here, we do not propose a solution for the problem of removing a bias. However, a researcher should keep in mind that in both cases (anonymous vs. non-anonymous) a bias exist and the researcher should design the experiments in such a way that the bias is as small as possible. Official elections in Germany do a good job here for instance. They guarantee anonymity but establish rules such that every person can only vote once and cannot manipulate the process.

4 Fabrication of Data

In some cases, researchers commit scientific misconduct as it is described in McCook [2016]. Test results are fabricated entirely or manipulated in such a way that it supports the proposed method by the author. Not seldom, the culture within a research environment can be quite demanding or even harmful and some researchers might take a "short cut" by fabricating their results. Even worse, in some fields funding for research depends on previous publications and success and there is a huge pressure of showing progress. Last but not least, many authors would like to be considered an expert in a field, even if this expertise is not valid and only appears to be so.

While plagiarism can be detected with special kind of software, doing the same for fabricated data is not so easy and one needs to be skeptical about results. There are are a few indications that there could be a scientific misconduct within a paper: First, data collection can sometimes be quite difficult. Getting a high number of participants is often not possible without a huge amount of effort or financial support. One needs to evaluate whether the given sample size sounds realistic or not. In addition, one could check from where the data comes from. Second, data relies on certain distributions and one can check for such

properties. For example, Benford's law states that the digit "1" appears more often than the digit "9". Fabricated data might not have this property. Third, if one author published papers that were retracted in the past, one should be at least skeptical. The researcher "Yoshitaka Fujii" holds the record with 183 retractions during his career ret [2018].

The damage of scientific misconduct is huge. Researchers invest in methods that build on already existing ones. However, if the foundation is not working, then it is likely that the method build on top of it does not work as well. Once the experiment fail, time and money has been spent already and it is not obvious from the start, why it failed. One might doubt his own method first before searching more broadly. But fabricated data is not only wasting time and money, it can also put humans to a risk. A new medication might actually cause harm to a human than curing someone if one does not properly test it.

Data scientist have the tools to find out the *true* root distribution of data and they need to make usage of those. Not only because data fabrication also happens in the field of Machine Learning but also because they can prevent frauds in other fields as well and raise awareness.

5 Summary

In this paper we showed that there are several reasons why to be skeptical about the underlying data or certain results. We argued that there is more than only applying machine learning algorithms to data sets. It includes finding out biases in training data or experiments and thinking about the consequences of an analysis to the (data science) society.

References

The retraction watch leaderboard, Sep 2018. URL `https://retractionwatch.com/the-retraction-watch-leaderboard/`.

Sage Lazzaro For Dailymail.com. Soap dispenser only responds to white skin, Aug 2017. URL `http://www.dailymail.co.uk/sciencetech/article-4800234/Is-soap-dispenser-RACIST.html`.

Daniel Faggella. Machine learning healthcare applications - 2018 and beyond, Aug 2018. URL `https://www.techemergence.com/machine-learning-healthcare-applications/`.

Javier Garcıa and Fernando Fernández. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, 16(1):1437–1480, 2015.

Jürgen Kuri. Bgh zu autocomplete: Google muss in suchvorschläge eingreifen, May 2013. URL https://www.heise.de/newsticker/meldung/BGH-zu-Autocomplete-Google-muss-in-Suchvorschlaege-eingreifen-1862062.html.

Alison McCook. Why do scientists commit misconduct?, Aug 2016. URL https://retractionwatch.com/2016/08/29/why-do-scientists-commit-misconduct/.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

Andrew Y Ng, Adam Coates, Mark Diel, Varun Ganapathi, Jamie Schulte, Ben Tse, Eric Berger, and Eric Liang. Autonomous inverted helicopter flight via reinforcement learning. In *Experimental Robotics IX*, pages 363–372. Springer, 2006.

OpenAI. Faulty reward functions in the wild, Mar 2017. URL https://blog.openai.com/faulty-reward-functions/.

Benjamin Peters. *Digital keywords: a vocabulary of information society and culture*. Princeton University Press, 2016.

Seth Stephens-Davidowitz. *Everybody lies: what the Internet can tell us about who we really are*. Bloomsbury, 2017.

Fei-Yue Wang, Jun Jason Zhang, Xinhu Zheng, Xiao Wang, Yong Yuan, Xiaoxiao Dai, Jie Zhang, and Liuqing Yang. Where does alphago go: From church-turing thesis to alphago thesis and beyond. *IEEE/CAA Journal of Automatica Sinica*, 3(2):113–120, 2016.