Algorithmic and Data bias in machine learning model decision making.

By Victorine Jeptoo

DECLARATION:

I Godfrey Victorine Jeptoo declares that this is my original work and has not been presented anywhere else at the best of my knowledge.

Sign _____.

Supervisor confirmation:

I _____ confirm that this report has not been presented anywhere to the best of my knowledge.

Sign _____.

DEDICATION:

I would like to dedicate this report to Dr Ngure Nyaga who made sure I was well trained during my attachment period in the area of machine learning, my family for their continued support and lecturers for guidance.

ACKNOWLEDGEMENT:

The report would not have been successful without the cooperation of a number of people who enabled me to successfully finish this proposal.

First, I would like to thank the Almighty God for the charitable time, strength and attitude that enabled me to complete my attachment period.

I wish to acknowledge madam Gakii for continued support and guidance.

Finally, I would like to thank my loving family for their support and encouragement.

ABSTRACT:

This research contains detailed information on two of the major issues affecting the field of AI in recent times. (This section is written last as it is a summary of all the write-up)

TABLE OF CONTENTS

Contents

List of Graphical Figures

AI -Artificial Intelligence.

CHAPTER ONE

INTRODUCTION

The term “artificial intelligence” was popularized at a conference at Dartmouth College in the United States (1956)that brought together researchers on a broad range of topics, from language simulation ,data science to learning machines. Despite periods of significant scientific advances in the six decades since AI has often failed to live up to the hype that surrounded it. Decades were spent trying to describe human intelligence precisely, and the progress made did not deliver on the earlier excitement. Since the late 1990s, however, technological progress has gathered pace, especially in the past decade. Machine-learning algorithms have progressed, especially through the development of deep learning and reinforcement-learning techniques based on neural networks.

Several other factors have contributed to recent progress. Exponentially more computing capacity has become available to train larger and more complex models; this has come through silicon-level innovation including the use of graphics processing units and tensor processing units, with more on the way. This capacity is being aggregated in hyper-scale clusters, increasingly being made accessible to users through the cloud. Another key factor is the massive amounts of data being generated and now available to train AI algorithms. Some of the progress in AI has been the result of system-level innovations. Autonomous vehicles are a good illustration of this: they take advantage of innovations in sensors, LIDAR, machine vision, mapping and satellite technology, navigation algorithms, and robotics all brought together in integrated systems. Despite the progress, many hard problems remain that will require more scientific breakthroughs. This includes bias in today's Machine Learning models. Machine learning bias, also known as algorithm bias or AI bias, is a phenomenon that occurs when an algorithm produces results that are systematically prejudiced due to erroneous assumptions in the machine learning process. Algorithms can have built-in biases because they are created by individuals who have conscious or unconscious preferences that may go undiscovered until the algorithms are used, and potentially amplified, publicly. High bias is a reflection of problems related to the gathering or usage of data, where systems draw improper conclusions about data sets. This is often due to human intervention or the researchers' lack of cognitive assessment. Types of cognitive bias that can be inadvertently applied to algorithms are stereotyping, bandwagon effects, confirmation bias, priming and selective perception. Machine learning, a subset of artificial intelligence, depends on the quality, objectivity and size of learning data sets. Since machine-learning algorithms and pattern recognition abilities operate in the world defined by the data used to calibrate them, a lack of truly random or complete data can conclude in bias. Eliminating harmful biases is essential because machine learning is often applied to decisions with business implications, such as which individuals to approve for a loan and which applicants to offer a job interview, and personal implications, such as diagnostics in medical environments. One example of machine learning bias was observed in the initial rollout of Google's facial recognition feature as users of varying race were often incorrectly tagged as inhuman or ignored completely.

BACKGROUND TO THE STUDY

In the current age of complexity in problems facing we, humans have come up with ways in which we can solve them optimum and with high speed. One of the solutions we have is using machines to handle work that would otherwise be too large or complex to be handled by humans. One of the most revolutionary ideas is to use artificial intelligence where you gather data and use it to teach a model and then give it a problem to solve.

But AI still faces many practical challenges, though new techniques are emerging to address them. Machine learning can require large amounts of human effort to label the training data necessary for supervised learning. In-stream supervision, in which data can be labelled in the course of natural usage, and other techniques could help alleviate this issue. Obtaining data sets that are sufficiently large and comprehensive to be used for training—for example, creating or obtaining sufficient clinical-trial data to predict healthcare treatment outcomes more accurately—is also often challenging.

The “black box” complexity of deep learning techniques creates the challenge of “explainability,” or showing which factors led to a decision or prediction, and how. This is particularly important in applications where trust matters and predictions carry societal implications, as in criminal justice applications or financial lending. Some nascent approaches, including local interpretable model-agnostic explanations (LIME), aim to increase model transparency. Bias is another big challenge facing AI. Try as we might have data that is an absolute fact, there is inevitable bias when you explore the depths to which AI might be used. Forbes India explains the inherent bias in data, “An inherent problem with AI systems is that they are only as good – or as bad – as the data, they are trained on. Bad data is often laced with racial, gender, communal or ethnic biases. Proprietary algorithms are used to determine who is called for a job interview, who’s granted bail, or whose loan is sanctioned. If the bias lurking in the algorithms that make vital decisions goes unrecognized, it could lead to unethical and unfair consequences…In the future, such biases will probably be more accentuated, as many AI recruiting systems will continue to be trained using bad data. Hence, the need of the hour is to train these systems with unbiased data and develop algorithms that can be easily explained. Microsoft is developing a tool that can automatically identify bias in a series of AI algorithms.” An example of a biased AI is Facebook facial detection Artificial intelligence which identified a black man as a monkey one day after it was released. This is an example of how artificial intelligence can face trust issues with humans, in spite of its ability to cut down on tasks. It is basic human psychology that we often neglect something that we don’t understand. We as humans tend to stay away from anything complicated. And artificial intelligence is related to huge data, data science and algorithms, there are times when users do not grasp these concepts. Intelligence comes from learning, whether you’re human or machine. Systems usually have a training phase in which they "learn" to detect the right patterns and act according to their input. Once a system is fully trained, it can then go into the test phase, where it is hit with more examples and we see how it performs. Obviously, the training phase cannot cover all possible examples that a system may deal with in the real world. These systems can be fooled in ways that humans wouldn't be. For example, random dot patterns can lead a machine to “see”

things that aren't there. If we rely on AI to bring us into a new world of labor, security and efficiency, we need to ensure that the machine performs as planned and that people can't overpower it to use it for their own ends. Though artificial intelligence is capable of speed and capacity of processing that's far beyond that of humans, it cannot always be trusted to be fair and neutral. Google and its parent company Alphabet are one of the leaders when it comes to artificial intelligence, as seen in Google's Photos service, where AI is used to identify people, objects and scenes. But it can go wrong, such as when a camera missed the mark on racial sensitivity, or when a software used to predict future criminals showed bias against black people. We shouldn't forget that AI systems are created by humans, who can be biased and judgmental. Once again, if used right, or if used by those who strive for social progress, artificial intelligence can become a catalyst for positive change. No AI system can be universally fair or unbiased. But we can design these systems to meet specific fairness goals, thus mitigating some of the perceived unfairness and creating a more responsible system overall. (ALGORITHMS OF OPPRESSION by Safiya Umoja Noble)

PROBLEM STATEMENT

Bias in machine learning models often affects crucial decision making negatively by downgrading its trust. This also makes people being targeted for biased decisions made by the models. Hence everyone who's is concerned in the development of models from data scientists to knowledge engineers should adopt methods to minimize if not prevent bias.

OBJECTIVES

1. To research and find out the reason behind artificial intelligence being biased.
2. To find how bias in machines affects its usability.
3. To find out ways of detecting biased data before it is fed into AI models.
4. To find out the bias reduction methods of minimizing bias in AI model after it has been trained.

SCOPE

The research will focus on how we gather data and process it in order to be used to train the artificial intelligent models. It will mainly focus on data science methods which directly and actively affect the field of AI. The research will also focus on how decisions are made in teaching AI models and who makes them because this is the most important equation in solving the problem of biased machine models.

JUSTIFICATION

1. The research will help bring about change in building biased models like the one for Facebook which identified a black man as a monkey.
2. The research will help identify if the data gatherer, scientist and engineer are biased in their field of work hence making models that are biased and prone to affect the society negatively.
3. The research will help data scientists make decisions between quality and quantity of data needed for training artificial intelligence.

CHAPTER TWO

LITERATURE REVIEW:

There are those who praise the AI as the solution to some of humankind's gravest problems, and those who demonize AI as the world's greatest existential threat. Of course, these are two ends of the spectrum, and AI, surely, presents exciting opportunities for the future, as well as challenging problems to be overcome. The AI has been in existence for more than 60 years, and includes two main areas of research. One is based on rules, logic, and symbols; it is explainable; and it always finds a correct solution for a given problem, if that problem has been correctly specified. However, it can be used only when all possible scenarios for the problem at hand can be foreseen. The other area of research is based on examples, data analysis, and correlation. It can be applied in cases where there is an incomplete or ill-defined notion of the problem to be solved. However, this type of AI requires a lot of data, is usually less explainable, and there is always a small margin of error. These two lines of research and ways of thinking about AI are increasingly being combined in order to maximize the advantages of both and to mitigate their drawbacks.

In recent years, many successful applications of AI have been built, mainly because of the convergence of improved algorithms, vast computing power, and massive amounts of data. This provides AI systems with human-level perception capabilities, such as speech-to-text, text understanding, image interpretation, and others, for which machine-learning methods are suitable. These abilities make it possible to deploy AI systems in real-life scenarios that typically have a high degree of uncertainty. Still, current consumer-oriented AI applications where a service is provided to users—from navigation systems to voice-activated "smart" homes— barely scratch the surface of the tremendous opportunity that AI represents for businesses and other institutions.

The main purpose of what can be called enterprise AI is to augment humans' capabilities and to allow humans to make better, more informed and grounded decisions. At this point, AI and humans have very complementary capabilities, and it is when their capabilities are combined that we find the best results. Typical applications in enterprise AI are decision-support systems for doctors, educators, financial service operators, and a host of other professionals who need to make complex decisions based on lots of data. All this being said it is important to mention why Bias is one of the most challenging problems facing AI and to be precise machine learning. Machine learning is a subset of artificial intelligence and can be defined as an application of artificial **intelligence** (AI) that provides systems with the ability to automatically learn and improve from experience without being explicitly programmed. **Machine learning** focuses on the development of computer programs that can access data and use it to learn for themselves. This by itself is the most important part of machine learning to be able to add more to its own intelligence without any human interaction. This being said it is important to know that the efficiency of the machine is as good as its developer and the data used. This brings us into one of the top four problems challenging AI today and that is bias in AI.

According to Cambridge dictionary bias is preferring or disliking someone or something more than someone or something else, in a way that means that they are treated unfairly while Wikipedia

terms it as disproportionate weight in favor of or against an idea or thing, usually in a way that is closed-minded, prejudicial, or unfair. The most important factor about bias is that it innates or learned. If you understand both definitions of bias you will notice that they tend to lean more on the human aspect. In this case, they are also applicable in Artificial intelligence since the algorithms used and models trained are created by humans either directly or indirectly. Human biases are well-documented, from implicit association tests that demonstrate biases we may not even be aware of, to field experimentation that demonstrate how much these biases can affect outcomes. According to Judgment_under_uncertainty (Daniel Kahneman, Paul Slovic, Amos Tversky), many decisions are based on beliefs concerning the likelihood of uncertain events such as the outcome of an election. These beliefs are expressed in numerical forms such as odds or subjective probabilities. The research shows how people rely on a limited number of heuristic principles which reduce the complete tasks of assessing probabilities and predicting values to simpler judgmental operations. In general, these heuristics are useful but they may lead to severe and systematic errors. Over the past few years, society has started to wrestle with just how much these human biases can make their way into artificial intelligence systems with harmful results. At a time when many companies are looking to deploy AI systems across their operations, being acutely aware of those risks and working reducing them is an urgent priority. Has a lot of research has been done in the field of bias in Artificial Intelligence particularly machine learning? According to Corinne Bernstein Machine learning bias, also known as algorithm bias or AI bias, is a phenomenon that occurs when an algorithm produces results that are systematically prejudiced due to erroneous assumptions in the machine learning process. These Algorithms can have built-in biases because they are created by individuals who have conscious or unconscious preferences that may go undiscovered until the algorithms are used, and potentially amplified, publicly. A new wave of decision-support systems is being built today using AI services that draw insights from data (like text and video) and incorporate them in human-in-the-loop assistance. However, just as we expect humans to be ethical, the same expectation needs to be met by automated systems that increasingly get delegated to act on their behalf. A very important aspect of ethical behavior is to avoid (intended, perceived, or accidental) bias. Bias occurs when the data distribution is not representative enough of the natural phenomenon one wants to model and reason about. The possibly biased behavior of service is hard to detect and handle if the AI service is merely being used and not developed from scratch since the training data set is not available (AIES '18 Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society Pages 284-289).

Bias detection and mitigation are also fundamental in achieving trust in AI. Bias can be introduced through training data when it is not balanced and inclusive enough, but it can also be injected in the AI model in many other ways. Moreover, among the many notions of fairness, it is important to choose the most appropriate given the specific application context. It is also important to help developers become aware of what is available and can be used in current AI systems because of the abundance of bias metrics, notions of fairness, and bias mitigation and detection algorithms. The global community of data scientists and developers can and should continue to improve upon these capabilities in a collaborative way. To that end, IBM has made available to the open-source community a toolkit called “AI Fairness 360” to help developers and data scientists check for and

mitigate bias in AI models using bias-handling solutions, and supporting them with guidelines, datasets, tutorials, metrics, and algorithms.

Identifying and mitigating bias in AI systems is essential to building trust between humans and machines that learn. As AI systems find, understand, and point out human inconsistencies in decision making, they could also reveal ways in which we are partial, parochial, and cognitively biased, leading us to adopt more impartial or egalitarian views. In the process of recognizing our bias and teaching machines about our common values, we may improve more than AI. We might just improve ourselves (IBM researcher Francesca Rossi). Research in machine learning over the last ten years has been particularly concerned with experimental comparisons and the relative performance of the different class. In machine learning, bias refers to any basis for choosing one generalization over another, other than strict consistency with the instances" (Mitchell, 1980). technically, if an algorithm produces results that are on average skewed or incorrect with respect to the population it is being used to analyses, then the conclusions are considered to be biased. Colloquially, however, algorithmic bias is more commonly used to describe
systematic discrimination on the basis of these results. Generally speaking, discrimination can be defined as an "unjustified distinction of individuals based on their membership, or perceived membership, in a certain group or category". Therefore, a reasonable definition of algorithmic bias in the sense we are using it here is the unfair treatment of a group (e.g. an ethnic minority, gender or type of worker) that can result from the use of an algorithm to support decision-making.

Discrimination and fairness are central issues here. In order to be useful, algorithms must filter or discriminate between individuals in a population (e.g. they must be able to provide a reasonable assessment of someone's creditworthiness). The central question is whether they can do this fairly.

At the most basic level, a distinction between procedural and outcome fairness is often made. Procedural fairness is concerned with the fairness of the steps, input data, and evaluations made in a decision-making process. In a data science context, this could mean an algorithm which processes data about individuals in the same way, regardless of characteristics such as gender and ethnicity. Procedural fairness also encompasses other issues, such as the input of stakeholder groups in rule-making and revision processes, and the ability for individuals to appeal decisions and easily subject them to legal scrutiny. On the other side, outcome fairness addresses the equity of the outcomes of a decision-making process, and how they are distributed across individuals and social groups within the population. It is often discussed in terms of discrimination and the denial of opportunities or services to specific groups. A major problem is that these approaches to fairness are often fundamentally incompatible, meaning that they require judgement to choose between the most appropriate approach for a given task. For example, an employer may emphasize procedural fairness to ensure that all job applicants are treated equally, but then end up with a shortlist biased for or against certain social groups due to the use of selection criteria (e.g. education) that are proxies for membership of one group or another(Bias in Algorithmic decision-making by DR.Michael rovatsos,dr Brent Mittelstadt, Dr. Ansgar Koene).People are often unclear on the nature of the algorithms controlling large portions of their lives. Decisionmakers and policy analysts increasingly rely on algorithms as they try to make timely effective decisions in a data-

rich world. Their use of algorithms (or artificial agents more generally) as decision aids encapsulates details that are important but not pertinent to the decision. This is a strong benefit of algorithmic aids for decision-making. A properly functioning algorithm frees up the decisionmaker's cognitive capacity for other important deliberations. But the opacity of algorithms makes it harder to judge correctness, evaluate risk, and assess fairness in social applications. It can also obscure the causal understanding behind decisions. These issues might be harmless if algorithms were (near) infallible. But most algorithms have only probabilistic guarantees of accuracy. And this is in the best possible scenarios, in which the right models and algorithms are applied appropriately, with the best intention to "perfect" data. Algorithm designers and users rarely have the luxury of such perfect scenarios. They must rely on assumptions that can fail and lead to unexpected results. The fallibility of algorithms is an easy point to make. This includes systematic algorithmic errors, not just the statistical inaccuracies inherent to many algorithms. There are many examples in public policy–oriented applications. As a concrete example of significant error, Google's Flu Trends tool is famous for repeatedly misdiagnosing nationwide flu trends (Lazer et al., 2014). Many risk-estimation algorithms were based on incorrect probabilistic models and failed to react properly just before the 2008 U.S. financial crash (Salmon, 2012). One city implemented algorithms intended to optimally detect street potholes based on passively collected data from smartphone users. The demographic breakdown of smartphone users at the time would have led to blind spots, causing some communities to be underserved (Crawford,

2013). This would have had the effect of depriving less-affluent citizens' access to city repair services. Another city decided to use algorithmic approaches to direct its law-enforcement activities. The justification was that predictive policing algorithms were more objective as they only relied on objective "multi-variable equations," not on subjective human decisions (quoted in Tett, 2014). Reporting on another criminal justice application, Angwin et al. (2016) demonstrated systematic bias in a criminal risk assessment algorithm used in sentencing hearings across the United States.

.

Research Design

EXPERIMENTAL DESIGN.

A blueprint of the procedure that enables the researcher to maintain control over all factors that may affect the result of an experiment. In doing this, the researcher attempts to determine or predict what may occur. Experimental Research is often used where there is time priority in a causal relationship (cause precedes effect), there is consistency in a causal relationship (a cause will always lead to the same effect), and the magnitude of the correlation is great. The classic experimental design specifies an experimental group and a control group. The independent variable is administered to the experimental group and not to the control group, and both groups are measured on the same dependent variable. Subsequent experimental designs have used more groups and more measurements over longer periods. True experiments must have control, randomization, and manipulation.

Justification

1. Experimental research will allow the researcher to control the situation at hand. In so doing, it allows me to answer the question, "what causes algorithmic bias to occur?"
2. This design Permits me to identify cause and effect relationships between variables and to distinguish placebo effects from treatment effects.
3. Experimental research supports the ability to limit alternative explanations and to infer direct causal relationships in the study.
4. This Kind of approach provides me with the highest level of evidence for single studies.

Research tools and procedures

TOOLS

Data Collection

1.Case Studies

In order to answer a combination of 'what' and 'why' questions i will conduct survey on the subject case studies generally involve a mix of quantitative (i.e., surveys, usage statistics, etc.) and qualitative (i.e., interviews, focus groups, extant document analysis, etc.) data collection techniques. Most often, the researcher will analyze quantitative data first and t

2.Interviews

In-Depth Interviews include both individual interviews (e.g., one-on-one) as well as "group" interviews (including focus groups). The data can be recorded in a wide variety of ways including stenography, audio recording, video recording or written notes. In depth interviews differ from

direct observation primarily in the nature of the interaction. In interviews it is assumed that there is a questioner and one or more interviewees. The purpose of the interview is to probe the ideas of the interviewees about the phenomenon of interest.

Data analysis.

Hypothesis testing

Also known as "T Testing," this analysis method lets you compare the data you have against hypotheses and assumptions you've made about your operations. It also helps you forecast how decisions you could make will affect your organization. T Testing lets you compare two variables to find a correlation and base decisions on the findings. For instance, you may assume that more hours of work are equivalent to higher productivity. Before implementing longer work hours, it's important to ensure there's a real connection to avoid an unpopular policy

Content analysis

This method helps to understand the overall themes that emerge in qualitative data. Using techniques like color coding specific themes and ideas helps parse textual data to find the most common threads. Content analyses can work well when dealing with data such as user feedback, interview data, open-ended surveys, and more. This can help identify the most important areas to focus on for improvement.

Grounded theory: This refers to using qualitative data to explain when physical items. When to use this method depends on the research questions. Content any a certain phenomenon happened. It does this by studying a variety of similar cases in different settings and using the data to derive causal explanations. Researchers may alter the explanations or create new ones as they study more cases until they arrive at an explanation that fits all cases.

Prescriptive Analysis

Prescriptive Analysis combines the insight from all previous Analysis to determine which action to take in a current problem or decision. Most data-driven companies are utilizing Prescriptive Analysis because predictive and descriptive analysis is not enough to improve data performance. Based on current situations and problems, they analyze the data and make decisions.

System Requirements

NONE.

Result and Discussion.

HOW DOES ARTIFIAL INTELLIGENCE BECOME BIASED

1. DATASETS

Today there are more than one hundred and eighty human biases that have been defined and classified and many of them are evident in the artificially intelligent systems of today. These kinds of misclassifications happen because the data used does not represent truly distribution in all different subgroups. Example a study conducted and written on a paper called “no classification without representation” found out that most studied and used data set for images in the world come from western countries. Machine learning algorithms are statistical estimation methods. Their measures of estimation error often vary in inverse proportion with data sample sizes. This means that these methods will typically be more error-prone on low-representation (An Intelligence in Our Image by Rand cooperation)

Issues with Data Sets

. Unseen Cases

Much of the advantage of AI systems is their ability to generalize solutions with robustness to varied input. However, this can become a disadvantage when the system is faced with a class for which it was not trained. For example, a neural network trained to classify texts as German or English will still provide an answer when given a text in French rather than saying “I don’t know”. Such issues can lead to “hidden” or “silent” mispredictions, which can then propagate to cause additional harm to the business application.

Mismatched Data Sets

If data seen in production differs significantly from that used in training, the model is unlikely to perform well. Extending the point above, commercial facial recognition systems trained on mostly fair-skinned subjects have vastly different accuracies for different populations: 0.8% for lighter-skinned men and 34.7% for darker-skinned women. Even if the model is originally trained on a dataset that matches production use, production data can change over time due to various effects from seasonal changes to external trigger events. Any such change can bring about hidden effects generated by mismatched data sets.

Manipulated Data

Training data can be manipulated to skew the results as was exemplified by the short-lived chatbot Tay, which quickly mimicked the hate speech of its Twitter correspondents. Systems trained on small, public data sets are especially vulnerable to this form of attack. Similarly, data poisoning is a known security challenge for AI systems.

Unlearned Cases

Even well-trained models do not have 100% accuracy; indeed, such high accuracy would likely result from overfitting the data and indicate that the model is not likely to generalize well to new cases. As a result, even well-trained models will have classes of samples for which they perform poorly. Studies have shown that facial recognition datasets that do not adequately represent across ethnic groups can cause trained models to display vastly different accuracies across race.

Non-Generalizable Features

Due to the practical difficulties in creating large, labelled training sets, model developers may rely on training from well-preened subsets of their expected production data sets. This can result in granting importance to features that are particular to the training set and not generalizable to broader data sets. For example, shows how text classifiers, which were trained to classify articles as “Christian” or “atheist” on standard newsgroup training sets, emphasize non-relevant words like “POST” in making their classifications

Irrelevant Correlations

If the training data contains correlations between irrelevant input features and the result, it may produce incorrect predictions as a result. For example, Ribeiroetal. trained a classifier to differentiate between wolves and dogs with images of wolves surrounded by snow and dogs without snow. After training, the model sometimes predicts that a dog surrounded by snow is a wolf. Unlike non-generalizable features, the distribution of irrelevant correlations may not be particular to the training set but may occur in real-world data as well. It may well be that wolves are more likely to be found in snow than dogs. However, it would be incorrect for the feature to impact the prediction; a wolf is still a wolf even when it is in Grandmother’s house.

2. Biased Algorithmic Engineering Process.

This means AI systems always contain a degree of human error since it is built by humans. So what is an algorithm? An algorithm is a detailed series of instructions for carrying out an operation or solving a problem. For the non-programmers, is a set of instructions that take an input, A, and provide an output, B, that changes the data involved in some way. Technically, computers use algorithms to list the detailed instructions for carrying out an operation. Today Algorithms have a wide variety of applications. In math, they can help calculate functions from points in a data set, among much more advanced things. Aside from their use in programming itself, they play major roles in things like file compression and data encryption and in our context machine learning. **Algorithmic bias** describes systematic and repeatable errors in a computer system that create unfair outcomes, such as privileging one arbitrary group of users over others. Bias can emerge due to many factors, including but not limited to the design of the algorithm or the unintended or unanticipated use or decisions relating to the way data is coded,

Types of Algorithmic Bias

Pre-existing

Pre-existing bias in an algorithm is a consequence of underlying social and institutional ideologies. Such ideas may influence or create personal biases within individual designers or programmers. Such prejudices can be explicit and conscious, or implicit and unconscious. Poorly selected input data will influence the outcomes created by machines. Encoding pre-existing bias into software can preserve social and institutional bias, and without correction, could be replicated in all future uses of that algorithm.

Technical

Technical bias emerges through limitations of a program, computational power, its design, or other constraint on the system. Such bias can also be a restraint of design, for example, a search engine that shows three results per screen can be understood to privilege the top three results slightly more than the next three, as in an airline price display. Another case is software that relies on randomness for fair distributions of results. If the random number generation mechanism is not truly random, it can introduce bias, for example, by skewing selections toward items at the end or beginning of a list.

A *decontextualized algorithm* uses unrelated information to sort results, for example, a flight-pricing algorithm that sorts results by alphabetical order would be biased in favor of American Airlines over United Airlines. The opposite may also apply, in which results are evaluated in contexts different from which they are collected. Data may be collected without crucial external context: for example, when facial recognition software is used by surveillance cameras, but evaluated by remote staff in another country or region, or evaluated by non-human algorithms with no awareness of what takes place beyond the camera's field of vision. This could create an incomplete understanding of a crime scene, for example, potentially mistaking bystanders for those who commit the crime.

Lastly, technical bias can be created by attempting to formalize decisions into concrete steps on the assumption that human behavior works in the same way. For example, software weighs data points to determine whether a defendant should accept a plea bargain, while ignoring the impact of emotion on a jury. Another unintended result of this form of bias was found in the plagiarism-detection software Turnitin, which compares student-written texts to information found online and returns a probability score that the student's work is copied. Because the software compares long strings of text, it is more likely to identify non-native speakers of English than native speakers, as the latter group might be better able to change individual words, break up strings of plagiarized text, or obscure copied passages through synonyms. Because it is easier for native speakers to evade detection as a result of the technical constraints of the software, this creates a scenario where Turnitin identifies foreign-speakers of English for plagiarism while allowing more native-speakers to evade detection

Emergent

Emergent bias is the result of the use and reliance on algorithms across new or unanticipated contexts. Algorithms may not have been adjusted to consider new forms of knowledge, such as new drugs or medical breakthroughs, new laws, business models, or shifting cultural norms. This may exclude groups through technology, without providing clear outlines to understand who is responsible for their exclusion. Similarly, problems may emerge when training data (the samples "fed" to a machine, by which it models certain conclusions) do not align with contexts that an algorithm encounters in the real world.

In 1990, an example of emergent bias was identified in the software used to place US medical students into residencies, the National Residency Match Program (NRMP). The algorithm was designed at a time when few married couples would seek residencies together. As more women entered medical schools, more students were likely to request a residency alongside their partners. The process called for each applicant to provide a list of preferences for placement across the US, which was then sorted and assigned when a hospital and an applicant both agreed to a match. In the case of married couples where both sought residencies, the algorithm weighed the location choices of the higher-rated partner first. The result was a frequent assignment of highly preferred schools to the first partner and lower-preferred schools to the second partner, rather than sorting for compromises in placement preference.

Correlations

Unpredictable correlations can emerge when large data sets are compared to each other. For example, data collected about web-browsing patterns may align with signals marking sensitive data (such as race or sexual orientation). By selecting according to certain behavior or browsing patterns, the end effect would be almost identical to discrimination through the use of direct race or sexual orientation data. In other cases, the algorithm draws conclusions from correlations, without being able to understand those correlations. For example, one triage program gave lower priority to asthmatics who had pneumonia than asthmatics who did not have pneumonia. The program algorithm did this because it simply compared survival rates: asthmatics with pneumonia are at the highest risk. Historically, for this same reason, hospitals typically give such asthmatics the best and most immediate care.

Unanticipated uses

Emergent bias can occur when an algorithm is used by unanticipated audiences. For example, machines may require that users can read, write, or understand numbers, or relate to an interface

using metaphors that they do not understand. These exclusions can become compounded, as biased or exclusionary technology is more deeply integrated into society.

Apart from exclusion, unanticipated uses may emerge from the end user relying on the software rather than their own knowledge. In one example, an unanticipated user group led to algorithmic bias in the UK, when the British National Act Program was created as a proof-of-concept by computer scientists and immigration lawyers to evaluate suitability for British citizenship. The designers had access to legal expertise beyond the end users in immigration offices, whose understanding of both software and immigration law would likely have been unsophisticated. The agents administering the questions relied entirely on the software, which excluded alternative pathways to citizenship, and used the software even after new case laws and legal interpretations led the algorithm to become outdated. As a result of designing an algorithm for users assumed to be legally savvy on immigration law, the software's algorithm indirectly led to bias in favor of applicants who fit a very narrow set of legal criteria set by the algorithm, rather than by the broader criteria of UK immigration law.

Feedback loops

Emergent bias may also create a feedback loop, or recursion, if data collected for an algorithm results in real-world responses which are fed back into the algorithm. For example, simulations of the predictive policing software (Predpol), deployed in Oakland, California, suggested an increased police presence in black neighborhoods based on crime data reported by the public. The simulation showed that the public reported crime based on the sight of police cars, regardless of what police were doing. The simulation interpreted police car sightings in modeling its predictions of crime, and would in turn assign an even larger increase of police presence within those neighborhoods. The Human Rights Data Analysis Group, which conducted the simulation, warned that in places where racial discrimination is a factor in arrests, such feedback loops could reinforce and perpetuate racial discrimination in policing.

Recommender systems such as those used to recommend online videos or news articles can create feedback loops. When users click on content that is suggested by algorithms, it influences the next set of suggestions. Over time this may lead to users entering a Filter Bubble and being unaware of important or useful content.

3.AI Can Magnify Bias

Machine learning based on biased datasets often amplifies those biases. In one example, a photo dataset had 45 percent more women than men in photos involving cooking, but the algorithm amplified that bias to 55 percent. This means that AI without proper guidance all configured set of rules which gives it the scope of how it learns then it is prone to self-bias.

How Machine Bias Affect its Usability.

Unfair algorithmic biases and unintended negative side effects of machine learning (ML) are gaining attention—and rightly so. We know that machine translation systems trained on existing historical data can make unfortunate errors that perpetuate gender stereotypes. We know that voice devices underperform for users with certain accents, since speech recognition isn't necessarily trained on all speaker dialects. We know that recommendations can amplify existing inequalities. However, pragmatic challenges stand in the way of practitioners committed to addressing these issues. There are no clear guidelines or industry-standard processes that can be readily applied in practice on what biases to assess or how to address them. While researchers have begun to create a rich discourse in this space, the translation of research discussions into practice is challenging. Barriers to action are threefold: understanding the issues, inventing approaches to address the issues, and confronting organizational/institutional challenges to implementing solutions at scale.

- **The moral component —** The level of intelligence and "morality" that a machine exerts is a direct result of the data it receives. One consequence is that, based on the data input, machines may train themselves to work against the interest of some humans or be biased. Failure to erase bias from a machine algorithm may produce results that are not in line with the moral standards of society. Yet not all researchers, scientists and experts believe that AI will be hurtful to society. Some believe that AI can be developed to mirror the human brain and obtain human moralistic psychology to enhance society.
- **Accuracy of risk assessments —** Risk assessments are used in many areas of society to evaluate and measure the potential risks that may be involved in specific scenarios. The increasing popularity of using AI risk assessments to make important decisions on behalf of people is a direct result of the growing trust between humans and machines. However, there are serious implications to note when using a machine learning system to make risk assessments. A quantitative analyst estimates that some machine learning strategies may fail up to 90 percent when tested in a real-life setting. The reason is that while algorithms used in machine learning are based on an almost infinite number of items, much of this data is very similar. For these machines, finding a pattern would be easy, but finding a pattern that will fit every real-life scenario would be difficult.
- **Transparency of algorithms —** Supporters of creating transparency in AI advocate for the creation of a shared and regulated database that is not in possession of any one entity that has the power to manipulate the data; however, there are many reasons why corporations are not encouraging this. While transparency may be the solution to creating trust between users and machines, not all users of machine learning see a benefit there.

detecting biased data before it is fed into AI models.

Google's What-If tool

Google's What-If Tool (WIT) is an interactive tool that allows a user to visually investigate machine learning models. WIT is now part of the open source Tensor Board web application and

provides a way to analyze data sets in addition to trained TensorFlow models. One example of WIT is the ability to manually edit examples from a data set and see the effect of those changes through the associated model. It can also generate partial dependence plots to illustrate how predictions change when a feature is changed. WIT can apply various fairness criteria to analyze the performance of the model (optimizing for group unawareness or equal opportunity). WIT is straightforward to use and includes a number of demonstrations to get users up to speed quickly.

Human in the loop

In many cases, machine learning models are black box. You can feed them inputs and look at their outputs, but how they map those inputs to outputs is concealed within the trained model.

Explainable models can help to bring to light how machine learning models come to their conclusions, but until these are commonplace, an alternative is *human in the loop.*

Human in the loop is hybrid model that couples traditional machine learning with humans monitoring the results of the machine learning model. This allows them to observe when algorithmic or other data set biases come into play. Recall that Microsoft used crowdsourcing to validate their word embedding bias discoveries, which indicates that it is a useful hybrid model to employ.

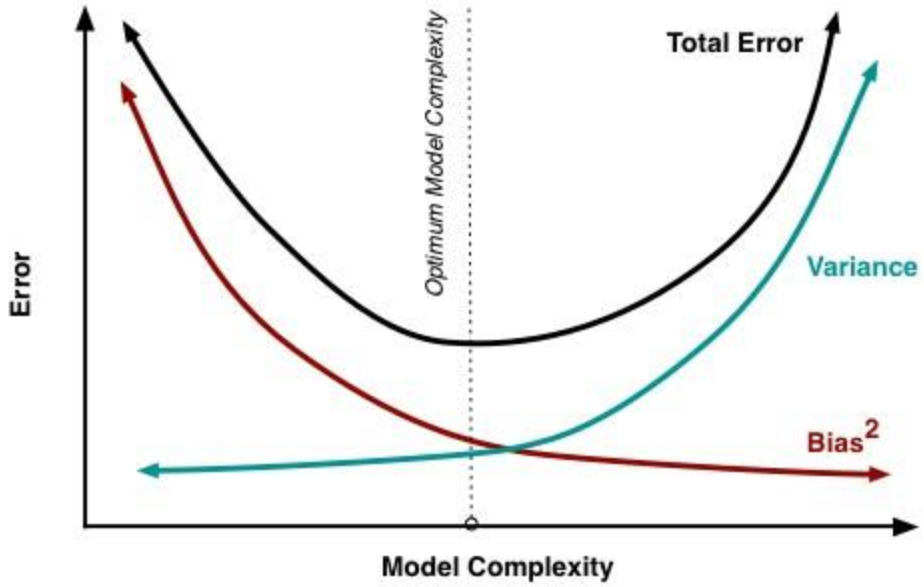
measuring Bias.

Bias reduction methods in AI

1. ensemble-learning

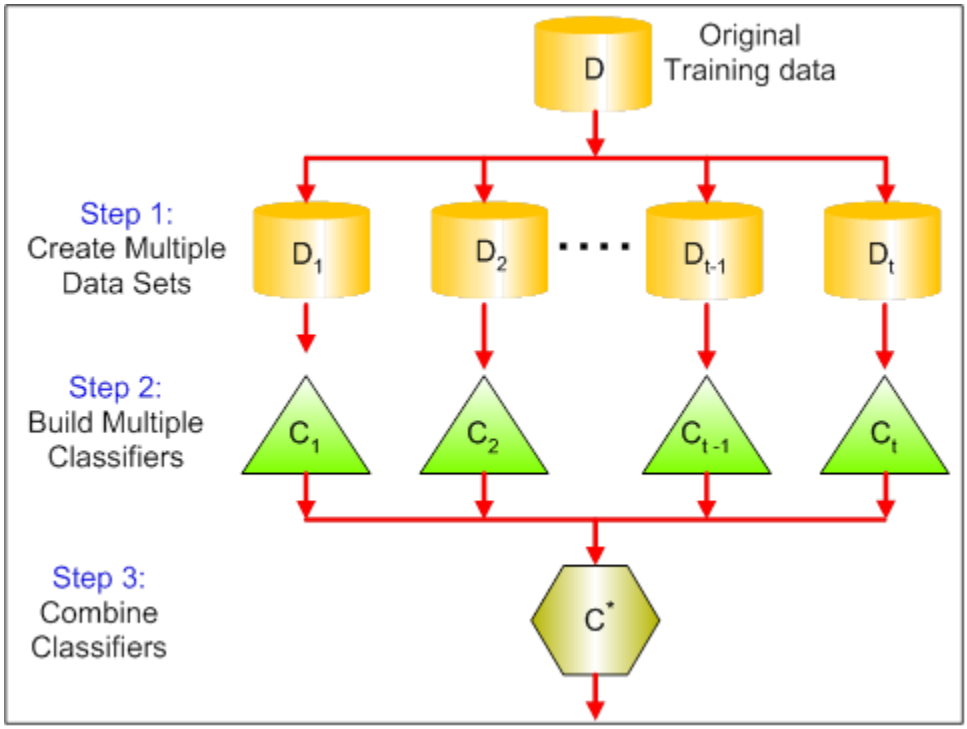
Ensemble is the art of combining diverse set of learners (individual models) together to improvise on the stability and predictive power of the model. example, the way we combine all the predictions together will be termed as Ensemble Learning. This is applicable where we have weak machine learning models which are required to solve a problem beyond their scope

Error in Ensemble Learning (Variance vs. Bias)

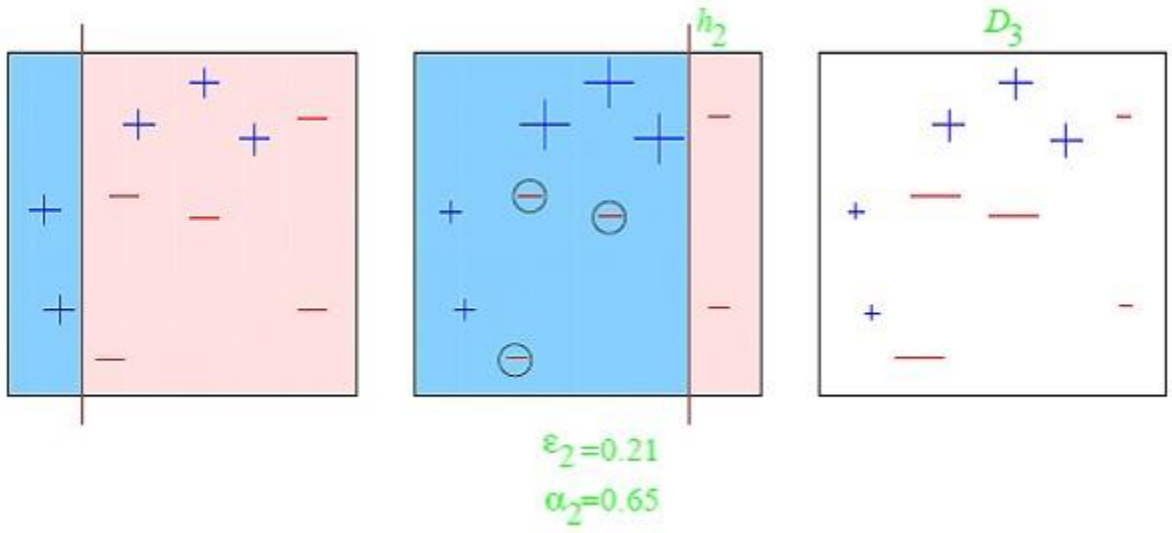


Some Commonly used Ensemble learning techniques

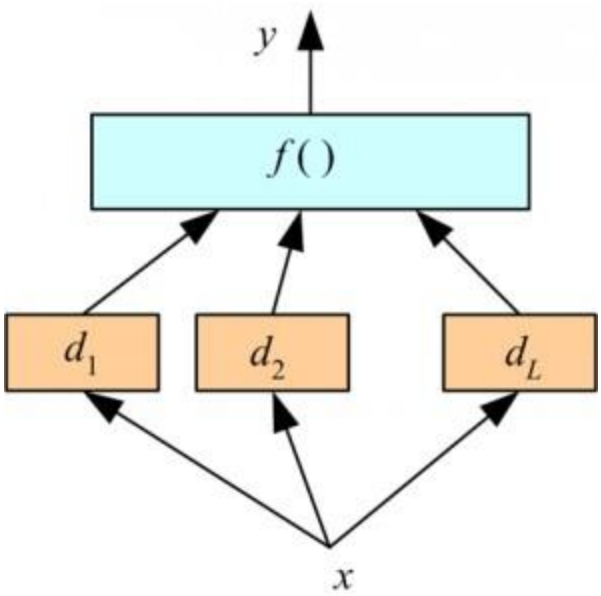
1. **<u>Bagging</u>**: Bagging tries to implement similar learners on small sample populations and then takes a mean of all the predictions. In generalized bagging, you can use different learners on different population. As you can expect this helps us to reduce the variance error.



2. **Boosting**: Boosting is an iterative technique which adjust the weight of an observation based on the last classification. If an observation was classified incorrectly, it tries to increase the weight of this observation and vice versa. Boosting in general decreases the bias error and builds strong predictive models. However, they may sometimes over fit on the training data.



3. **Stacking** : This is a very interesting way of combining models. Here we use a learner to combine output from different learners. This can lead to decrease in either bias or variance error depending on the combining learner we use.



2. oversampling and undersampling

Oversampling and **undersampling** in data analysis are techniques used to adjust the class distribution of a data set (i.e. the ratio between the different classes/categories represented). These terms are used both in statistical sampling, survey design methodology and in machine learning.

Oversampling and undersampling are opposite and roughly equivalent techniques.

Motivation for using the methods

Both oversampling and under sampling involve introducing a bias to select more samples from one class than from another, to compensate for an imbalance that is either already present in the data, or likely to develop if a purely random sample were taken. Data Imbalance can be of the following types:

1. *Under-representation of a class in one or more important predictor variables.* Suppose, to address the question of gender discrimination, we have survey data on salaries within a particular field, e.g., computer software. It is known women are under-represented considerably in a random sample of software engineers, which would be important when adjusting for other variables such as years employed and current level of seniority. Suppose only 20% of software engineers are women, i.e., males are 4 times as frequent as females. If we were designing a survey to gather data, we would survey 4 times as many females as males, so that in the final sample, both genders will be represented equally.
2. *Under-representation of one class in the outcome (dependent) variable.* Suppose we want to predict, from a large clinical dataset, which patients are likely to develop a particular disease (e.g., diabetes). Assume, however, that only 10% of patients go on to develop the

disease. Suppose we have a large existing dataset. We can then pick 1/9th the number of patients who did not go on to develop the disease for every one patient who did.

The end-result of over-/under-sampling is the creation of a *balanced dataset*. Many machine-learning techniques, such as neural networks, make more reliable predictions from being trained with balanced data. Certain analytical methods, however, notably linear regression and logistic regression, do not benefit from a balancing approach.

Oversampling is generally employed more frequently than undersampling, especially when the detailed data has yet to be collected by survey, interview or otherwise. Undersampling is employed much less frequently. Overabundance of already collected data became an issue only in the "Big Data" era, and the reasons to use undersampling are mainly practical and related to resource costs. Specifically, while one needs a suitably large sample size to draw valid statistical conclusions, the data must be cleaned before it can be used. Cleansing typically involves a significant human component, and is typically specific to the dataset and the analytical problem, and therefore takes time and money. For example:

- Domain experts will suggest dataset-specific means of validation involving not only intra-variable checks (permissible values, maximum and minimum possible valid values, etc.), but also inter-variable checks. For example, the individual components of a differential white blood cell count must all add up to 100, because each is a percentage of the total.

- Data that is embedded in narrative text (e.g., interview transcripts) must be manually coded into discrete variables that a statistical or machine-learning package can deal with. The more the data, the more the coding effort. (Sometimes, the coding can be done through software, but somebody must often write a custom, one-off program to do so, and the program's output must be tested for accuracy, in terms of false positive and false negative results.)

For these reasons, one will typically cleanse only as much data as is needed to answer a question with reasonable statistical confidence (see Sample Size), but not more than that.

Oversampling techniques for classification problems

Random oversampling

Random Oversampling involves supplementing the training data with multiple copies of some of the minority classes. Oversampling can be done more than once (2x, 3x, 5x, 10x, etc.) This is one of the earliest proposed methods, that is also proven to be robust. Instead of duplicating every sample in the minority class, some of them may be randomly chosen with replacement.

SMOTE

There are a number of methods available to oversample a dataset used in a typical classification problem (using a classification algorithm to classify a set of images, given a labelled training set of images). The most common technique is known as SMOTE: Synthetic Minority

Over-sampling Technique. To illustrate how this technique works consider some training data which has ***s*** samples, and ***f*** features in the feature space of the data. Note that these features, for simplicity, are continuous. As an example, consider a dataset of birds for classification. The feature space for the minority class for which we want to oversample could be beak length, wingspan, and weight (all continuous). To then oversample, take a sample from the dataset, and consider its *k* nearest neighbors (in feature space). To create a synthetic data point, take the vector between one of those *k* neighbors, and the current data point. Multiply this vector by a random number *x* which lies between 0, and 1. Add this to the current data point to create the new, synthetic data point.

Many modifications and extensions have been made to the SMOTE method ever since its proposal.

ADASYN

The adaptive synthetic sampling approach, or ADASYN algorithm, builds on the methodology of SMOTE, by shifting the importance of the classification boundary to those minority classes which are difficult. ADASYN uses a weighted distribution for different minority class examples according to their level of difficulty in learning, where more synthetic data is generated for minority class examples that are harder to learn.

Undersampling techniques for classification problems Random undersampling

---

Randomly remove samples from the majority class, with or without replacement. This is one of the earliest techniques used to alleviate imbalance in the dataset, however, it may increase the variance of the classifier and may potentially discard useful or important samples.

Cluster

Cluster centroids is a method that replaces cluster of samples by the cluster centroid of a K-means algorithm, where the number of clusters is set by the level of undersampling.

Tomek links

Tomek links remove unwanted overlap between classes where majority class links are removed until all minimally distanced nearest neighbor pairs are of the same class. A Tomek link is defined as follows: given an instance pair , where and is the distance between and , then the pair is called a Tomek link if there's no instance such that or . In this way, if two instances form a Tomek link then either one of these instances is noise or both are near a border. Thus, one can use Tomek links to clean up overlap between classes. By removing overlapping examples, one can establish well-defined clusters in the training set and lead to improved classification performance.

3.Hyperparameter Tuning

Hyperparameters in Machine Learning are user-controlled “settings” of your ML model. They influence how your model’s parameters will be updated and learned during training. Of course, the output of your model depends on its learned parameters, and its learned parameters are constantly updated and determined during the training phase. That updating is controlled by the

model's training, which is in turn influenced by the hyperparameters! Thus, if you can set the right hyperparameters, your model will learn the most optimal weights that it possibly can with a given training algorithm and data.

Finding the best hyper-parameters is usually done manually. It's a simple task of trial and error, with some intelligent guesstimating. You'll simply try as many hyperparameter settings as you have time for, and see which one gives you the best results. You can narrow your search space just by having some rough idea of what good parameters might be. That's a matter of domain knowledge, insight into your data, and experience with machine learning. Find the best hyperparameters you can and the model's performance might boost by a few percentage points!

Approaches

a) Grid search

The traditional way of performing hyperparameter optimization has been *grid search*, or a *parameter sweep*, which is simply an exhaustive searching through a manually specified subset of the hyperparameter space of a learning algorithm. A grid search algorithm must be guided by some performance metric, typically measured by cross-validation on the training set or evaluation on a held-out validation set. Since the parameter space of a machine learner may include real-valued or unbounded value spaces for certain parameters, manually set bounds and discretization may be necessary before applying grid search. For example, a typical soft-margin SVM classifier equipped with an RBF kernel has at least two hyperparameters that need to be tuned for good performance on unseen data: a regularization constant C and a kernel hyperparameter γ. Both parameters are continuous, so to perform grid search, one selects a finite set of "reasonable" values for each, say

Grid search then trains an SVM with each pair (C, γ) in the Cartesian product of these two sets and evaluates their performance on a held-out validation set (or by internal cross-validation on the training set, in which case multiple SVMs are trained per pair). Finally, the grid search algorithm outputs the settings that achieved the highest score in the validation procedure.

Grid search suffers from the curse of dimensionality, but is often embarrassingly parallel because the hyperparameter settings it evaluates are typically independent of each other.[2]

b) Random search

Random Search replaces the exhaustive enumeration of all combinations by selecting them randomly. This can be simply applied to the discrete setting described above, but also generalizes to continuous and mixed spaces. It can outperform Grid search, especially when only a small number of hyperparameters affects the final performance of the machine learning algorithm. In this case, the optimization problem is said to have a low intrinsic dimensionality. Random Search

is also embarrassingly parallel, and additionally allows the inclusion of prior knowledge by specifying the distribution from which to sample.

c)Bayesian optimization

Bayesian optimization is a global optimization method for noisy black-box functions. Applied to hyperparameter optimization, Bayesian optimization builds a probabilistic model of the function mapping from hyperparameter values to the objective evaluated on a validation set. By iteratively evaluating a promising hyperparameter configuration based on the current model, and then updating it, Bayesian optimization, aims to gather observations revealing as much information as possible about this function and, in particular, the location of the optimum. It tries to balance exploration (hyperparameters for which the outcome is most uncertain) and exploitation (hyperparameters expected close to the optimum). In practice, Bayesian optimization has been shown to obtain better results in fewer evaluations compared to grid search and random search, due to the ability to reason about the quality of experiments before they are run.

D)Gradient-based optimization

For specific learning algorithms, it is possible to compute the gradient with respect to hyperparameters and then optimize the hyperparameters using gradient descent. The first usage of these techniques was focused on neural networks. Since then, these methods have been extended to other models such as support vector machines or logistic regression.

A different approach in order to obtain a gradient with respect to hyperparameters consists in differentiating the steps of an iterative optimization algorithm using automatic differentiation.

e)Evolutionary optimization

Evolutionary optimization is a methodology for the global optimization of noisy black-box functions. In hyperparameter optimization, evolutionary optimization uses evolutionary algorithms to search the space of hyperparameters for a given algorithm. Evolutionary hyperparameter optimization follows a process inspired by the biological concept of evolution:

1. Create an initial population of random solutions (i.e., randomly generate tuples of hyperparameters, typically 100+)
2. Evaluate the hyperparameters tuples and acquire their fitness function (e.g., 10-fold cross-validation accuracy of the machine learning algorithm with those hyperparameters)
3. Rank the hyperparameter tuples by their relative fitness
4. Replace the worst-performing hyperparameter tuples with new hyperparameter tuples generated through crossover and mutation
5. Repeat steps 2-4 until satisfactory algorithm performance is reached or algorithm performance is no longer improving

Evolutionary optimization has been used in hyperparameter optimization for statistical machine learning algorithms, automatedmachine learning, deep neural

network architecture search, as well as training of the weights in deep neural networks. f) Population-based

Population Based Training (PBT) learns both hyperparameter values and network weights. Multiple learning processes operate independently, using different hyperparameters. Poorly performing models are iteratively replaced with models that adopt modified hyperparameter values from a better performer. The modification allows the hyperparameters to evolve and eliminates the need for manual hyper tuning. The process makes no assumptions regarding model architecture, loss functions or training procedures.

4.Feature Engineering

Feature engineering involves the careful selection and possible manipulation of your data's features. The purpose of this is to feed your model only the most optimal form of input. If you can consistently give your model only the parts of the data it needs to make accurate predictions, then it doesn't have to deal with any extra noise that comes from the rest of the data. If you apply Principal Component Analysis and find that one of your features have very low correlation with the output, then you probably don't need to be processing it. Some features are going to be intuitively not useful, such as the ID or perhaps recording date. Or maybe you only want certain features to be considered in the first place. To give your model the best your data has to offer, do some data exploration to find out what information and features are actually needed for predictions. Often times, you'll find that your dataset comes with some extra features that are either redundant or don't contribute to the prediction at all.

1. (Bias in Algorithmic decision-making by dr Michael rovatsos,dr Brent Mittelstadt,dr Ansgar Koene)

2.No classification without representation

3.Amodei, Dario, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané, "Concrete Problems in AI Safety," Ithaca, N.Y.: Cornell University Library, 2016. As of February 2, 2017:

4.https://arxiv.org/abs/1606.06565

5.Angwin, Julia, Jeff Larson, Surya Mattu, and Lauren Kirchner, "Machine Bias: There's Software Used Across the Country to Predict Future Criminals. And It's Biased Against Blacks," ProPublica, May 23, 2016. As of December 5, 2016:
https://www.propublica.org/article/ machine-bias-risk-assessments-in-criminal-sentencing

6.Arndt, A. B., "Al-Khwarizmi," Mathematics Teacher, Vol. 76, No. 9, 1983, pp. 668–670.

7.Athey, Susan, "Machine Learning and Causal Inference for Policy Evaluation,"

8.Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, Australia, August 10–13, 2015, pp. 5–6. Autor, David, "The Polarization of Job Opportunities in the U.S. Labor Market: Implications for Employment and Earnings," Washington, D.C.: Center for American Progress and the Hamilton Project, April 2010.

9.Baldus, David C., Charles Pulaski, and George Woodworth, "Comparative Review of Death Sentences: An Empirical Study of the Georgia Experience," Journal of Criminal Law and 10.Criminology, Vol. 74, No. 3, Autumn 1983, pp. 661–753. Baldus, David C., Charles A. 11.Pulaski, George Woodworth, and Frederick D. Kyle, "Identifying Comparatively Excessive 12.Sentences of Death: A Quantitative Approach," Stanford Law Review, Vol. 33, No. 1, 13.November 1980, pp. 1–74. Barocas, Solon, and Helen Nissenbaum, "Big Data's End Run 14.Around Procedural Privacy Protections," Communications of the ACM, Vol. 57, No. 11, 2014, pp. 31–33.

15.Barocas, Solon, and Andrew D. Selbst, "Big Data's Disparate Impact," California

16. *ALGORITHMS OF OPPRESSION by Safiya Umoja Noble | Kirkus Reviews*.

17.Kevin Eykholt, I. E. (2018). Robust Physical-World Attacks on Deep Learning Visual Classification. *CVPR.*

18.Langer, E. B. (1978). The mindlessness of Ostensibly Thoughtful Action: The Role of "Placebic" Information in Interpersonal Interaction. *Journal of Personality and Social Psychology, 36(6)*, 635–642.

19. Friedman, B. and Nissenbaum, H. Bias in computer systems. *ACM Trans. Inf. Syst. 14*, 3 (1996), 330–347.

20. FATML; https://www.fatml.org/resources/principles-for-accountable-algorithms

21. ACM Conference on Fairness, Accountability, and Transparency (ACM FAT*); https://fatconference.org

22. AI Now institute. Algorithmic impact assessments: A practical framework for public agencyaccountability. Apr. 2018; https://ainowinstitute.org/aiareport2018.pdf

23. Data & Society. Algorithmic accountability primer. Apr. 2018;https://datasociety.net/wp-content/uploads/2018/04/Data_Society_Algorithmic_Accountability_P rimer_FINAL-4.pdf

24 World Wide Web Foundation. Algorithmic accountability report, 2017; https://webfoundation.org/docs/2017/07/Algorithms_Report_WF.pdf

24. Gebru, T., Morgenstern, J., Vecchione, B., Wortman Vaughan, J., Wallach, H., Daumeé III, H., and Crawford, K. Datasheets for datasets. 2018; https://arxiv.org/abs/1803.09010

26. Olteanu, A., Castillo, C, Diaz, F., and Kiciman, E. Social data: Biases, methodologicalpitfalls, and ethical boundaries. 2016; http://dx.doi.org/10.2139/ssrn.2886526

27. Baeza-Yates, R. Data and algorithmic bias in the web. *Proc. of the 8th ACM Conference on Web Science.* ACM, New York, 2016; https://doi.org/10.1145/2908131.2908135

28. Springer, A. and Cramer, H. "Play PRBLMS": Identifying and correcting less accessiblecontent in voice interfaces. *Proc. of CHI '18.* ACM, New York, 2018.

29. Narayanan, A. FAT* 2018 tutorial: 21 fairness definitions and their politics.