Identifying software project risks using a causal machine learning approach

by

Blen Assefa

Bachelor Thesis in Computer Science

Submission: May 16, 2023 Supervisor: Prof. Dr.-Ing. Hendro Wicaksono

Constructor University | School of Computer Science and Engineering

Statutory Declaration

Family Name, Given/First Name	Assefa, Blen
Matriculation number	30004088
Kind of thesis submitted	Bachelor Thesis

English: Declaration of Authorship

I hereby declare that the thesis submitted was created and written solely by myself without any external support. Any sources, direct or indirect, are marked as such. I am aware of the fact that the contents of the thesis in digital form may be revised with regard to usage of unauthorized aid as well as whether the whole or parts of it may be identified as plagiarism. I do agree my work to be entered into a database for it to be compared with existing sources, where it will remain in order to enable further comparisons with future theses. This does not grant any rights of reproduction and usage, however.

This document was neither presented to any other examination board nor has it been published.

German: Erklärung der Autorenschaft (Urheberschaft)

Ich erkläre hiermit, dass die vorliegende Arbeit ohne fremde Hilfe ausschließlich von mir erstellt und geschrieben worden ist. Jedwede verwendeten Quellen, direkter oder indirekter Art, sind als solche kenntlich gemacht worden. Mir ist die Tatsache bewusst, dass der Inhalt der Thesis in digitaler Form geprüft werden kann im Hinblick darauf, ob es sich ganz oder in Teilen um ein Plagiat handelt. Ich bin damit einverstanden, dass meine Arbeit in einer Datenbank eingegeben werden kann, um mit bereits bestehenden Quellen verglichen zu werden und dort auch verbleibt, um mit zukünftigen Arbeiten verglichen werden zu können. Dies berechtigt jedoch nicht zur Verwendung oder Vervielfältigung.

Diese Arbeit wurde noch keiner anderen Prüfungsbehörde vorgelegt noch wurde sie bisher veröffentlicht.

May 16, 2023

Date, Signature

Abstract

Decision-making based on data has been traditionally practiced in many industries. A Data Scientist's primary task is to determine what adjustments need to be made in order to achieve the best results possible for customers or clients[60]. This research focuses on quantifying the assumptions and demonstrating which factors influence some of the variables in software production systems from a project manager's perspective.

The thesis intends to go beyond correlation and delve into the fundamental mechanisms driving project risks by merging causal inference and machine learning, providing project managers with valuable insights for efficient risk mitigation techniques. Here we show how to apply causal machine learning within the context of management and uncover the causal relationships between variables in the software development process for identifying project risks. The comprehensive methodology combines diverse project attributes and utilizes the DoWhy framework to estimate causal effects and identify key risk factors.

This thesis shows how decision-making can benefit from causal machine learning. The findings highlight the critical role of contribution complexity, and communication in influencing project risks. Our work is to contribute to the field of causal machine learning by providing practical implications for project managers to mitigate risks and improve project outcomes proactively. We show different variables' cause-and-effect relationships for managers for better planning. Our findings also paves the way for further investigation and development of causal machine-learning methods for risk management in software development.

Contents

1 Introduction

Managing software projects can be complex and challenging, requiring careful planning, execution, and monitoring. Identifying and mitigating risk could ensure in making the best decision for planning, executing, and monitoring projects [117]. Over the years, numerous techniques have been used to identify and mitigate project risks, including traditional risk management approaches and more advanced machine learning techniques. Traditional risk management approaches rely on expert judgment and experience, which can be subjective and prone to errors [50]. The use of various artificial intelligence (AI) and machine learning (ML) approaches to enhance the precision and efficacy of risk management in project management has gained popularity in recent years [36].

This research paper presents a study on applying machine learning techniques to identify project risks in software development. Specifically, the ML technique is Causal Machine Learning (CausalML). Causal Machine learning is a term used to cover a broad category of machine learning methods to formalize the data-generation process as a structural causal model [59]. The study will also examine the challenges and limitations of using AI and ML techniques in project management and propose strategies to overcome these challenges.

This study aims to investigate using a causal machine learning approach to identify and mitigate project risks in large projects. The study will explore the potential benefits of using AI and ML techniques to improve the accuracy and effectiveness of risk management in project management.

The study focuses on two projects that were chosen at random from a dataset that contained approximately 200 projects gathered and made public for performing fault-proneness, defect prediction, or other analysis on them [40]. This research aims to perform causal inference and apply causal machine learning techniques using the DoWhy library[77] to find relationships between variables that can help software product project managers make informed decisions and commonalities within the software product development cycles.

In this study, causal machine learning helps identify critical factors contributing to selected project risks for developing effective mitigation strategies. We highlight the methods of incorporating causal inference techniques in software development projects to improve decision-making processes and reduce project risks. The practices are easily replicable because the datasets examined have a generic structure of the GitHub project.

The study will contribute to the existing literature on project risk management and provide insights into the effectiveness of using causal machine learning in identifying and mitigating project risks. The study will focus on software projects and the use machine learning algorithms to analyze project data and identify potential risks. The study will be limited to causal machine learning approaches and not cover other machine learning techniques.

The study has several limitations that need you should consider and keep in mind. Firstly, our study will be limited to software projects, and the findings may not apply to other type projects. Secondly, the study will only focus on using causal machine learning approaches and will not cover other machine learning techniques. Thirdly, the study will rely on project data, and the accuracy of the findings will depend on the quality of the data. Finally, the study will only cover some possible risks associated with large projects; some risks may be missed. Despite these limitations, our research will provide useful tips for the use of machine learning in project risk management and contribute to the existing literature on the topic.

1.1 Statement and Motivation of Research

In today's competitive environment, software project managers face several problems, and making the best decisions that can substantially impact project success is essential. Large-scale real-world studies, on the other hand, can be costly and time-consuming. As a result, this research aims to give an alternate strategy by using data analysis to find patterns and links between project features. We can estimate the factors project managers want to change using causal machine learning, resulting in more informed decision-making and better project outcomes.

The primary goal of this research is to provide a valuable tool for resolving fundamental difficulties within their businesses. Through a practical, data-driven approach, we hope to show the benefits of causal machine learning in delivering insightful information for well-informed decision-making processes and encouraging project success. Our research will produce a valuable tool that will assist project managers in resolving issues and coming to informed decisions that will improve the overall success of their projects.

1.1.1 Research objectives

This research aims to estimate the causes of various outcomes from different treatments on software projects. The study uses other tools to identify multiple characteristics from the dataset of the projects and uses causal machine learning to estimate some variables project managers want to improve. The cost of real-world experimentation by project managers is high. This research serves as a data-backed approach to solving some fundamental problem projects managers would want to answer within companies. The scenarios created are based on assumptions and can be adjusted by different project managers as they see fit.

Project managers have practiced decision-making by data for many years. This approach is a traditional approach for decision-making in different-sized companies. However, instead of correlation, we investigate cause and effect relationships between different variables of the software development process of specific companies in such a way medium size scale companies or start-ups get to see the patterns of big companies in their growth.

Considering the goals, we experiment with a real-life software product dataset from Github.

2 Background

2.1 Project management and risk analysis

In the past 30 years, it has become obvious that project management is a valuable and insightful way of handling novel or challenging tasks that are known as projects. [80].

Accoring to Munns, A. K. and Bjeirmi, B. F. (1996)

A project can be thought of as accomplishing a particular goal, which involves several resource-consuming tasks and activities. It must be finished according to a predetermined timeline with established start and end dates [80].

On the contrary, project management oversees accomplishing the project's objectives. It attempts to manage the project by using a collection of tools and approaches while utilizing the existing organizational structures and resources without negatively impacting the business as usual [80]. Project management tasks include defining the needed work, determining its scope, allocating the necessary resources, planning how the work will be carried out, keeping track of its progress, and addressing deviations from the plan [80].

Some of the common metrics to measure project management is their effective development of strategies like initiations, plannings, interventions, executions, regulations and risk identifications and mitigations [2]. There are several techniques for ensuring minimal disruption while meeting specified deliverables defined as a goal to accomplish. To guarantee that there are as few disruptions as possible while the project is in progress, risk analysis and management are crucial in the project management strategy formulation.

Our studies focus on risk management and identification of in project management, specifical software project management. While it is impossible to forecast the future with absolute certainity, several tools have been used traditionally to identify some potential risks and reduce the likelihood that they will arise. This can increase a project's success and decrease the risks' adverse effects [64]. Most challenges arise in the formulation of project management that avoids risks. The challenging aspect of it is, is to know beforehand the outcomes of using specific methods.

To further understand, we look into what risk is the first step in implementing any risk management plan.

A risk is an uncertain event or circumstance that, if it occurs, has a positive or negative impact on a project objective [98]. Thus, a risk is not an event or occurrence already occurring during a project. It is a possible occurrence [98].

In any activity, there is always a certain level of risk involved. It is essential to understand that risk is not an event that has already happened but rather an event that might occur [98, 13]. Risk is composed of three elements: the risk event, its potential consequences, and the likelihood of it happening [98, 13]. A clearly defined risk event is necessary to comprehend the issue entirely.

During project planning, risks can be categorized as known, unknown, or unknowable. Known risks are recognized by many project personnel from the start, while unknown risks are only known by a limited group of people and need to be identified during planning. Unknowable risks are completely unexpected and impossible to predict [98, 13].

The key objective of the risk management process is to detect potential risks that might not have been acknowledged before. The process is iterative and requires a feedback loop.By following the risk management process, we can be able minimize unexpected events during a project and prepare for unknown factors. This helps in addressing potential issues proactively [98, 13].

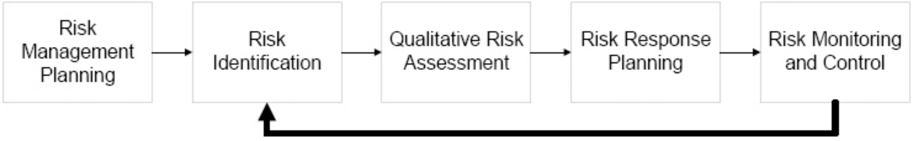


Figure 1: A process for managing risks as the “systematic process of identifying, analyzing, and responding to project risks” [98, 13]

2.2 Artificial Intelligence and Machine Learning in Project Managment

The integration of artificial intelligence (AI) and machine learning (ML) in project management is a topic that has been discussed in various sources. According to a Harvard Business Review article, AI and ML can bring significant benefits to project management, including automation of tasks and improved decision-making [62]. According to Gartner, by 2030, AI will be responsible for 80% of project management tasks. This will be made possible by utilizing big data, machine learning, and natural language processing [62]. Other sources, such as Brookings Institution and Red Hat, discuss the broader impact of AI and ML on society and businesses [116, 96]. Microsoft Azure provides a comparison between AI and ML [10], while Business News Daily explains how AI is already changing the business landscape[38].

2.2.1 Artificial Intelligence

Several definitions of artificial intelligence (AI) have emerged during the previous few decades [52]. John McCarthy's 2004 paper defines artificial intelligence as the precise science and engineering of fabricating intelligent machines, which primarily involves computer programs[76]. These fabricated intelligent machines, Artificial Intelligence (AI), are designed to surpass the boundaries of natural human observation by utilizing computer technology to comprehend and replicate human intelligence and outperform [76, 52].

Machines can perform tasks that typically require human intelligence, such as comprehending speech and decision-making, through the use of AI [29, 8, 55]. AI is continuously evolving to benefit many different industries, including healthcare, finance, and transportation. However, AI also raises important questions about ethics, privacy, and governance, and it is crucial to ensure that AI is designed and used in a way that benefits society as a whole [111, 56, 52, 119].

2.2.2 Machine Learning (ML)

Machine learning, a discipline at the intersection of AI and computer science, employs algorithms and data to replicate human learning processes, continually enhancing accuracy [51]. One application of AI is machine learning. Back in the 1950s, Arthur Samuel, a pioneer in AI, coined the term ”the field of study that enables computers to learn without the need for explicit programming” [106].

There are three types of machine learning algorithms [111]:

1. **Supervised learning:** Supervised learning is a method of teaching a computer system how to map input variables (X) to output variables (Y), and then using that

mapping to predict the output of new data. This is an essential technique in machine learning, particularly for multimedia data processing [24].

2. **Unsupervised learning:** Without any supervision or direction from a human expert, unsupervised learning is a subfield of machine learning that tries to enable computers learn and represent input patterns that reflect the statistical structure of a group of input patterns as a whole [27]. This approach is particularly useful when dealing with large and complex datasets, where it is difficult or impractical to manually label or classify the data.

3. **Reinforcement learning:** Reinforcement Learning (RL) is another form of learning that is guided by a specified goal. When you're in a new situation, the best way to figure things out is by trying things and seeing what happens. It's like how little kids learn - they do something and then watch what happens next. Same idea. The person uses the feedback it receives from the environment—which takes the form of rewards or penalties—to train itself and gain experience and understanding of the environment [81].

   The learner must independently determine the order of acts that would maximize reward, which is determined not only by the present reward but also by any potential delayed rewards. This is what is called Reinforcement learning(RL). RL is a powerful algorithm that can learn the behaviors that eventually lead to success in an unobserved environment without assistance from a supervisor.

2.2.2.1 What category is Causal ML?

Causal machine learning is not unsupervised, supervised, or reinforced learning, but rather a separate category of machine learning. Causal machine learning uses special models to understand how data is created. It helps researchers see how different things are connected and figure out what causes certain patterns. This can be useful in fields like finance, healthcare, marketing, and social sciences to make better predictions and decisions [59]. It is an attempt to make machine learning capable of causality, which is largely impossible at present [65].

2.2.3 Natural Language Processing (NLP)

According to IBM [53], Natural Language Processing (NLP) is an area of computer science and artificial intelligence that deals with how computers and humans communicate using natural language. It merges computational linguistics with statistical, machine learning, and deep learning models. NLP is employed to examine, comprehend, and produce human language, and it has many uses, such as chatbots, sentiment analysis, and machine translation.

2.2.4 Deep Learning (DL)

Deep learning is a powerful way to analyze data and make predictions. It is machine learning inspired by how our brains work, using complex networks to learn and improve over time [86, 57, 87, 49, 105]. Deep learning algorithms have the ability to learn from vast amounts of data, making them useful for a range of tasks such as recognizing images and speech, processing natural language, and enabling autonomous driving [57, 105] . The use of deep learning has led to significant advances in many fields, including computer vision, speech recognition, and natural language processing[105].

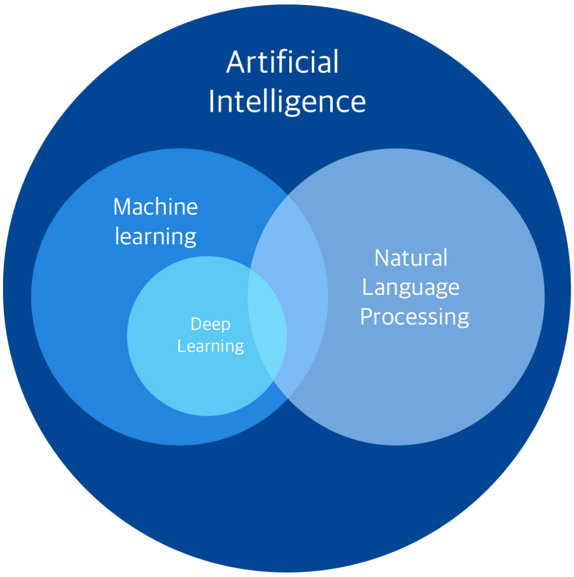


Figure 2: Illustration of the relationships between AI, ML, NLP, DL [115]

2.2.5 Relationships between AI, ML, NLP, DL and their application

AI is currently being used in a variety of industries, including healthcare [88], finance [19], and transportation [1]. Artificial intelligence (AI) has the potential to significantly impact project management, as the field relies on various human factors such as customer needs, team dynamics, and developer performance. AI can provide project analytics for estimating and predicting risks, automating repetitive tasks, offering recommendations, and making decisions, revolutionizing productivity and success rates [25].

To understand AI subfields and their potential project management applications, examining them closely is essential. One such subfield is machine learning, which involves developing algorithms and statistical models that allow computers to learn from data and make predictions or decisions without explicit programming [85, 89, 109, 58]. Machine learning algorithms can be used in project management for scheduling, resource allocation, risk identification and risk assessment [9]. For instance, a machine learning model could be trained to predict which team members are most likely to complete a project successfully and unscccessfully, allowing project managers to take proactive measures to mitigate those risks [33].

AI has a subfield called natural language processing (NLP) that deals with how computers interact with human language. NLP involves analyzing, comprehending, and creating natural language text and speech [22]. In project management, NLP can be used for tasks such as text analysis, which can help project managers quantify results for further data analysis.

By automating data analysis and causal inference, project managers can make decisions by being aware of the causes. AI can quickly identify patterns and trends that humans might overlook, allowing project managers to make more informed decisions. However, challenges such as the lack of standardization and the cost of implementing AI in project management must also be considered [84]. Understanding the different subfields of AI and their potential applications in project management can help project managers choose the right technology for their projects and stay competitive in an ever-evolving technological world.

2.3 Causal Inference and Causal Machine Learning

2.3.1 Causality

To understand causality, using a counterfactual or potential outcome approach is effective. But first, we need to define some terminology before formally explaining causal inference.

Assume that W is a random variable that represents the desired outcome and B is a random variable that represents the desired treatment. Individual i is represented as $b_i = 1$ if exposed and $b_i = 0$ if unexposed.

The difference between $W_i^{b=1}$, and $W_i^{b=0}$ is consequently used to determine the specific causal effect that one gets from a treatment B on an outcome W. The terms ”potential outcomes” or ”counterfactual outcomes” are used to describe the variables ”$W_i^{b=1}$” and ”$W_i^{b=0}$”. Causal inference is frequently viewed as a missing data issue because only one of these two outcomes can potentially be observed [17].

From the above explanation, we can define the average causal effect expression across all individuals.

$$E(W_i^{b=1} - W_i^{b=0}) \tag{1}$$

From the above equation, we can see that $E()$ is the expectation or average. The $E()$ can be calculated if the control or unexposed group is the same as to the exposed group in every respect, with the apparent exception of the exposure, as is the case, for instance, in a randomized trial [17].

The linearity of expectation allows us to give the following equation.

$$E(W_i^{b=1} - W_i^{b=0}) = E(W_i^{b=1}) - E(W_i^{b=0}) \tag{2}$$

We can reasonably assume that the average potential outcome for the exposed is then equal to the resulat we observe from the exposed grounp. Then

$$E(W^{b=1}|B = 1) = E(W|B = 1) \tag{3}$$

To continue further, if it happens that the potential outcome of the exposure in our experimental group is the same as the potentual outcome of the exposure for those exposed, then equation is

$$E(W^{b=1}) = E(W^{b=1}|B = 1) \tag{4}$$

and thus,

$$E(W^{b=1}) = E(W|B = 1) \tag{5}$$

or, alternatively, the observed outcome for the exposed is identical to the potential outcome for everyone who is exposed.

Notice here that, there is only a causal effect if $W^{b=1} \neq W^{b=0}$.

Similarly, we can infer that the observed outcome for non-exposes is equal to the potential outcome with no exposure for everyone [17].

$$E(W^{b=0}) = E(W|b = 0) \tag{6}$$

The equations for potential outcomes of non-exposed and exposed can be unified to get the average causal effect, $E(W_i^{b=1} - W_i^{b=0})$, and it can be estimated by the observed data $E(W|B = 1) - E(W|B = 0)$[17].

Given that:

$$E(W^{b=1}) = E(W|B = 1) \tag{7}$$

and

$$E(W^{b=0}) = E(W|B = 0) \tag{8}$$

To put it another way, causal association as stated by possible outcomes, $E(W_i^{b=1}) - E(W_i^{b=0})$ is the same as the observed association, $E(W|B = 1) - E(W|B = 0)$, assuming there is no confounding, as in a large randomized trial [17].

This is a basic introduction to causality and we should also know that other types of causal effects exist. For example, Causal relative risk: $E(W^1/W^0)$, Causal effect of treatment on the treated $E(W^1 - W^0|B = 1)$ or causal effects of how well treatment works among treated people [48].

In order to accurately calculate the average causal effect (ACE) in an observational study, it is important that the ignorability assumption is met. This means that the potential outcomes must be independent of treatment assignment. However, in many cases, ignorability is not automatically met in observational studies. It may only hold within certain groups defined by other variables, C, which is referred to as ”conditional ignorability.” If (conditional) ignorability is present, it is possible to obtain a fair and consistent estimate of the ACE [17, 71, 110, 94, 69].

Confounding variables are the variables or known similarities between groups that can be used to categorize them to avoid affecting the results. Different adjustment methods, such as propensity score matching, inverse probability weighting, and doubly robust estimation, can be used for confounding adjustment in the analysis of observational studies.

How can one determine whether (conditional) ignorability holds?

Statistical inference aims to simplify data into a concise mathematical representation of how observed variables are related. While statistical processes offer helpful insights into the data, they do not explain how it was generated. In contrast, causal inference aims to go beyond data description and uncover the underlying processes that produce the data, allowing for investigating interesting causal questions. You can find some guidelines for causal inference in the table 1.

Causation cannot be inferred just from statistical significance
Every task involving causal inference must rely on subjective, extra-data hypotheses or other scientifi research.
Assumption-free causal inference does not exist; rather, the issue is the quality of the assumptions, not their existence.
There are methods for mathematically expressing those assumptions and testing their implications.
There is a mathematical mechanism that can take those assumptions, integrate them with facts, and generate answers to interesting problems.

Table 1: conventions of causal inferencee [17]

2.3.1.1 What is the fundamental problem of causal inference?

The main issue with causal inference is that we can only see one possible result for each individual. However, we can calculate the average (population level) causal effects under certain conjectures.

Some variables can be estimated from data and thus they are considered identifiable. Identifiability of causal effects require making some untestable assumptions. They are called **causal assumptions**.

2.3.2 Statistical inference's incapacity to determine causality

Observed data and results might differ completely when experiment sub-groups are combined. A paradox named ”Simpson's paradox [20]” explain this phenomenon. It suggests that combined results suggest an outcome that makes no sense for individual groups of the experiment. For example, a drug may work differently for men and women, individually, but when combined, the results might be different.

We can avoid such conflicting results by knowing the treatments' characteristics and using those variables for sub-grouping the test subjects. This leads us to directed acyclic graphs (DAGs), which are used to simplify and visualize our approach. They help us when and if statistical associations are not the solutions.

2.3.3 DAG

A Directed Acyclic Graph (DAG) is a graphical tool used to represent causal relationships among a set of variables [39, 26, 61, 114, 45]. Causal diagrams or DAGs are highly effective in determining causal structures that align with the data available. This leads to logical inferences regarding statistical correlations. DAGs are instrumental in comprehending confounding, selection bias, covariate selection, and over-adjustment, as well as in avoiding analytical errors when dealing with these matters. DAGs are used to map out researchers' prior assumptions about the relationships between and among variables in causal structures [61, 17]

A set of variables' joint distribution is visually represented through DAGs, where each variable is depicted as a node linked by arrows pointing in a specific direction. DAGs are acyclic, meaning they contain no closed loops, and missing lines between nodes indicate variable independence. To create an accurate DAG, it is crucial to include all common causes of any two variables and all variables that play a role in data generation, whether observed or unobserved.

DAGs help in displaying the different associations that can exist between two variables. B and W [17]:

1. Random variations
2. B caused W
3. W caused B
4. A common cause is shared between W and B
5. Conditioning on a common impact of B and W resulted in the statistical correlation (Selection bias).

The first four statistical associations mentioned above are obvious but the fifth is when there are many independent causes for an effect (i.e. a common effect), conditioning on this common effect, or selecting only a subset of this common effect, results in a false connection between those causes [17]. The diagram below can assist illustrate this occurrence 3.

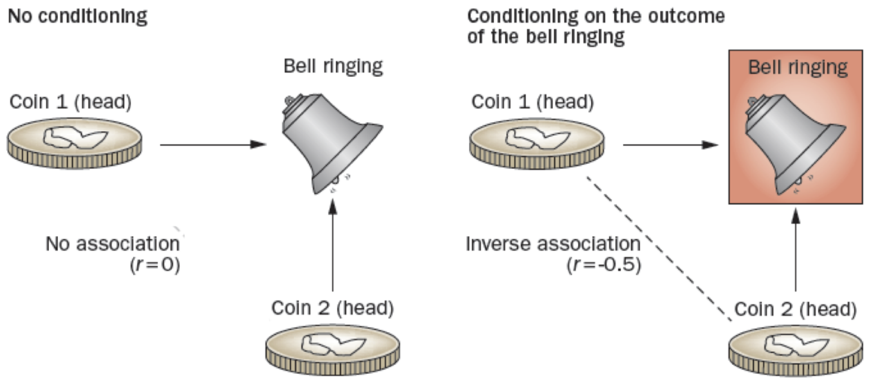


Figure 3: Collider stratification bias [16, 17]

To explain the concept of common effects, let me give you a simple example. In the image on the left, two coins are tossed independently. If one or both coins show heads, a bell rings. This means that the bell ringing is a common effect of the heads appearing on either coin, and the tosses are not related (with a correlation coefficient of 0).

In the Directed Acyclic Graph (DAG), we portray this by converging two causal arrows into a common effect variable or "collider." However, if we concentrate on the bell ringing and observe the image on the right, we can see that the two coin tosses are no longer independent. Instead, there is a -0.5 correlation between them. This correlation stems from the fact that if the first coin is tails, we know that the second coin (B) must be heads because the bell rang. In the DAG world, we represent conditioning on a common cause with a box around the variable name and spurious association with a dotted line between the variables. However, conditioning on a common cause may cause index event bias or collider stratification bias.

Another example, consider a study that aims to estimate the effect of a new drug on dementia. The study measures the drug treatment, outcome on dementia, and age of the participants. Age is a potential confounder because it is associated with both the treatment and the outcome. A DAG can be used to represent the causal relationships between these variables, with an arrow pointing from age to the treatment and the outcome. The DAG of the example can be as shown below:

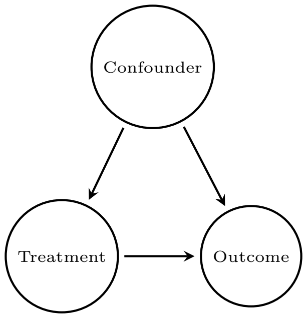


Figure 4: Example of a DAG for the Backdoor criterion.

In this example, the DAG shows that age is a confounder that affects both the treatment and the outcome. The arrow pointing from confounder(age) to the treatment and the outcome indicates that confounder(age) is a causal variable that needs to be adjusted for when estimating the causal effect of the drug on dementia.

2.3.4 Bausial Networks

Bayesian Networks (BNs) are models that use directed acyclic graphs (DAG) to represent probabilistic relationships between random variables [93]. These models make it possible to represent causal relationships between variables directly. Each node in the graph represents a random variable, while the edges between the nodes indicate the dependencies among them. These models are also known as causal probabilistic models or belief networks [15].Their structure makes it easy to combine observed data with our prior knowledge, which frequently takes the form of causal relationships.

2.3.5 Causal Bausial Networks

The causal relationships between variables are depicted using probabilistic graphical models called Causal Bayesian Networks (CBNs). CBNs can be used to anticipate the outcome of interventions, infer causal links between variables, and pinpoint the system's most essential variables. Causal Bayesian Networks (CBNs) provide a visual yet mathematically accurate framework that is a flexible and practical tool for facilitating more equitable decision-making in various fields, including healthcare, finance, and education [92, 21].

CBNs are founded on the premise that the probability of an event can be altered by other events that are causally related to it. A directed acyclic graph (DAG) can be used to model the structure of a CBN, with each node denoting a variable and the edges denoting the causal connections between them. The conditional probability distribution of the effect variable given the cause variable(s) represents the strength of the causal relationship.

Several algorithms, including the PC algorithm, the Greedy Equivalence Search algorithm, and the Hill-Climbing algorithm, can be used to learn CBNs from data[63, 104].

2.3.6 Causal effect

An independent variable and a dependent variable have a relationship known as a causal effect when the independent variable changes the dependent variable. Three concepts in causal inference that help to identify and understand the relationships between variables are: Confounding, mediating, and collider [66].

Third-variable effects, such as confounding and mediation, provide the theoretical basis for analyzing how risk variables affect behavior and how interventions change behavior. [66, 74].

The idea that conditioning on a variable C will change linkages between its causes is a key one in this literature. For instance, B and W will be connected within at least a single group of a variable that they both affect if they are minimally independent (i.e. unassociated prior to stratification) [91].

Let's take a look into the three concepts in detail:

1. **Mediator:** A mediator effect is a type of causal effect in which a third variable, called a mediator, explains the mechanism or process by which an independent variable affects a dependent variable [41]. Over the past 30 years, the case where a third variable acts as a mediator has seen extensive application and development in prevention and other fields because it provides an answer to the fundamental question of how a mediator transfers the causal effect of an independent variable to a dependent variable [74].

   The mediator is a variable affected by the independent variable and, in turn, affects the dependent variable, as depicted in the following figure 5.

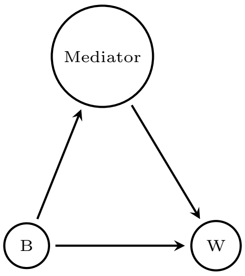


Figure 5: Mediator

2. **Confounder:** Confounding is a causal concept defined in terms of the data generation model as a variable that causally influences the dependent variable while also being linked with the independent variable, resulting in an erroneously relationship [97].

   More recent definitions place confounding in the paradigm of potential outcomes or counterfactuals, where as older definitions considered it a bias described as a mixing of effects of extraneous factors (called confounders) with the effect of interest [18].

   If confounding variables are not taken into account in statistical analysis, causal assertions cannot be made. They are typically problematic because a confounding variable, which influences both dependent and independent variables, must be taken into account in order to produce an accurate assessment of the causal effect [31, 74]. The figure 6 depictes this. By using randomization, restriction, or matching during the design phase, confounding control can take place.

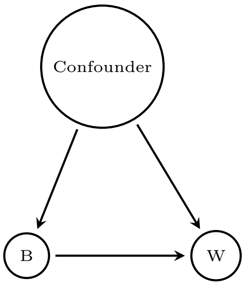


Figure 6: Confounder

3. **Collider:** A collider is a variable that is influenced by both the treatment and the result and can inadvertently link the two [91]. A collider variable is one that is caused by both the dependent and independent variables, as opposed to a confounder, which affects both the dependent and independent variables [74]. Adjustment for a collider conceals causal effects by creating bias in the estimate of the dependent-independant relationship, making collider variables challenging to comprehend, in contrast to adjustment for a mediator or confounder, which explains causal effects [74]. Figure 7 shows a straightforward digaram to represent colliders.

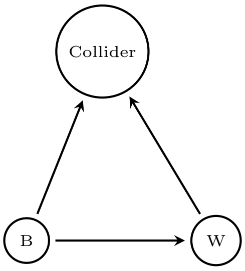


Figure 7: Collider

2.3.6.1 Identification of causal effects

1. **The Backdoor criterion:** In the realm of causal inference, the backdoor criterion is a valuable tool that aids in the identification of a set of variables that, when conditioned upon, can effectively eliminate any backdoor paths between a treatment and outcome variable within a causal graph.

   This criterion plays a significant role in enabling researchers to estimate the causal effect of a treatment on an outcome by allowing them to adjust for the variables that may have otherwise confounded the results. The backdoor criterion serves as an essential tool in the analysis of observational data, as it helps to block any potential bias by preventing backdoor paths between the treatment and outcome variables. The backdoor criterion is a key concept in causal inference and is widely used in the analysis of observational data [92].

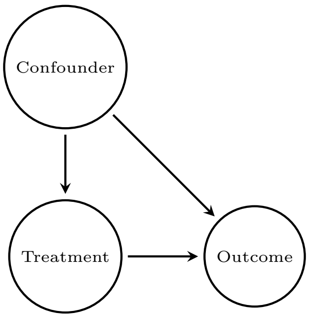


Figure 8: Example of a DAG illustrating the Backdoor criterion.

2. **The Frontdoor criterion:** The front-door criterion is a way to figure out if a treatment has a causal effect on an outcome, even if other factors could affect the outcome. It relies on a specific type of cause-and-effect relationship between the treatment and the outcome.

   The front-door criterion is a helpful tool for figuring out causal effects in studies where randomized controlled trials are not possible or ethical [44, 14, 35, 123, 46]. It involves three steps:

   1. Finding a variable that is affected by the treatment and affects the outcome,
   2. Finding a variable that affects the treatment but is not affected by the outcome, and
   3. Estimating the effects of the treatment and intermediate variables on the outcome

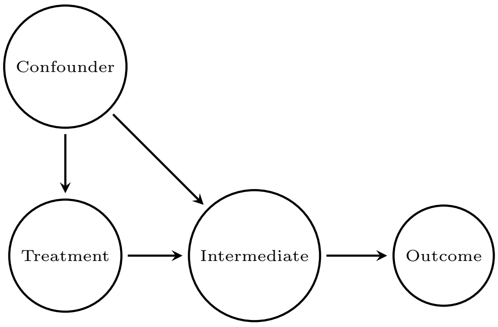


Figure 9: Example of a DAG for the Frontdoor criterion.

3. **The Instrumental variable:** The instrumental variable (IV) method is a useful tool for determining causal relationships in situations where controlled experiments are not feasible or where a treatment does not affect all units in a randomized experiment. This method helps overcome confounding and measurement errors in observational studies and allows for confident drawing of causal inferences.

   To do this, the IV method identifies a variable that is related to the treatment variable but not to the outcome variable, except through the treatment variable. This approach is effective in minimizing significant threats to internal validity, such as omitted variable bias and endogeneity.

   The IV method is widely used in econometrics, social sciences, epidemiology, and related disciplines to estimate causal relationships between variables when randomized controlled trials are not feasible or ethical [101, 113, 120, 124].

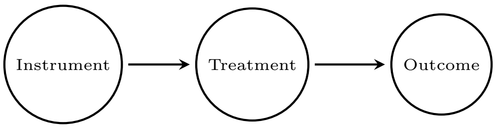


Figure 10: Example of a DAG for the Instrumental variable.

4. **Mediation (Direct and indirect effect identification):** Mediation analysis is a statistical tool that helps to understand how one variable affects another through an intermediary variable. This intermediary variable is called a mediator. The mediator explains the relationship between the independent and dependent variables [43].

   Mediation is a crucial process that involves the use of a mediator variable to explain the connection between the independent and dependent variables. This approach is essential in comprehending how different variables are interrelated and can provide valuable insights into the complex relationships that exist between them [43, 107, 32, 121, 112].

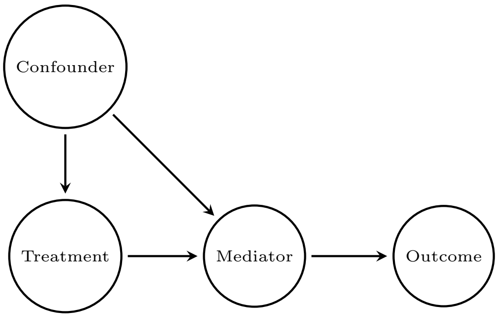


Figure 11: Example of a DAG for mediation.

2.3.7 Causal Estimands

Causal estimands are formulas used in causal inference to estimate the effect of an intervention on an outcome of interest [54]. They define the causal effect of an intervention on an outcome of interest and include different types such as the average treatment effect (ATE), the average treatment effect for the treated (ATT), and direct and indirect (or mediated) effects [54, 90, 68].

Causal estimands are used to estimate the causal effect of an intervention on an outcome of interest in both randomized controlled trials and observational studies [70]. The study's objectives, related primary outcome(s), how, when, and by whom the outcome(s) will be assessed, as well as the measurements of intervention effects—the major causal estimands—should all be clearly stated in the protocol for the trial [95, 75, 68].

The DoWhy[102] library defines a few techniques based on various goals. For instance:

- Techniques for calculating the treatment assignment [102]:
  - ➟ Propensity-based Stratification
  - ➟ Propensity Score Matching
  - ➟ Inverse Propensity Weighting

- Techniques for estimating the outcome model [102]:
  - ➟ Linear Regression
  - ➟ Generalized Linear Models
- Techniques for instrumental variables identification [102]:
  - ➟ Binary Instrument/Wald Estimator
  - ➟ Two-stage least squares
  - ➟ Regression discontinuity
- Techniques for front-door criterion and general mediation [102]:
  - ➟ Two-stage linear regression

2.3.8 Causal Refuters (Validating causal assumptions and estimators)

Recent work on creating validation tests for causal estimators can be broadly categorized into two groups [103].

1. **First type:** This method is similar to cross validation loss metric [3, 42, 7, 83, 34, 30, 103]. It develops a metric to indicate the causation estimate's quality using observed data. However, complementary models for quality assessment can be predicted through additional estimators without the need for experimental data from a randomized controlled trial or actual causal effects from ground truth data. [103].

   Accurate metrics for determining causal effect rely on precise auxiliary estimators that are non-parametrically consistent and highly accurate. However, meeting these requirements can be just as challenging as fitting the final causal model [103].

2. **Second type:** Another approach [82] involves creating a simulated dataset, in which the entire data-generating process (DGP) is understood, allowing for knowledge of the true causal effect to be passed down to the new created dataset.

   With the expectation that the magnitude of errors on the new dataset will also be transferred to the original dataset, candidate estimators are evaluated based on their error on the new dataset. The challenge is to create a simulated dataset that is close enough to the real dataset to enable for the identification of relevant data-generating process components to ascertain the true causal influence. A ranking of potential estimators may differ depending on different simulated datasets [103].

For our research, we utilized the DoWhy library to conduct validation tests. This library allowed us to create simulated datasets that closely resembled the original data. By doing so, we were able to determine the true causal effect, which can be difficult to establish using traditional methods due to certain assumptions and limitations [103]. Along with domain knowledge, it also offers sensitivity analyses to aid in assessing causal estimators [103].

The fact that causal hypotheses cannot be completely proven is crucial to remember in this situation [103]. Instead, the goal is to validate a few essential assumptions that follow and then eliminate models that do not meet those assumptions. We list the validated tests that are supported below [103].

- **Replacing treatment or outcome:** These tests generate a new dataset with a trivially known causal impact by substituting the treatment or outcome variable [103]. These tests are generic in that they can detect errors caused by any aspect of the analysis, including poor identification, inaccurate estimators, and even faults in the implementation, but they do not reveal which assumption was broken [103].

  1. **Placebo Treatment [103]:** The estimated causal effect should be zero if we substitute an independent random variable for the actual treatment variable.

  2. **Dummy Outcome [103]:** The predicted causal effect should be zero if we substitute an independent random variable for the genuine outcome variable.

  3. **Simulated Outcome[82, 103]:** If we replicate the data-generating process that produced the given dataset to create a simulation, the expected causal impact should correspond to the effect parameter of the original data-generating process.

- **Adding "unobserved" confounders:** In order to investigate how sensitive an estimate is to unobserved confounding, these strategies either introduce correlated confounders or randomly add confounders in areas where it is expected that the effect will remain the same.

  1. **Add Random Common Cause [103]:** One way to improve the accuracy of estimates in studies is by adding ”unobserved” confounders. This involves introducing random or correlated confounders to assess the sensitivity of the results.

  2. **Add Unobserved Common Causes [103]:** Adding a synthetic independent random variable as a common cause should not impact the estimated causal effect.

  3. **Simulated Outcome [103]:** Adding a common cause (confounder) to the dataset that is correlated with both the treatment and outcome should not significantly affect the effect estimate.

- **Creating subsets of the dataset:** To check for the variance of the estimator, we can create subsets or bootstrap samples of the dataset.

  1. **Data Subsets Validation:** We randomly select a subset of the dataset and replace the original dataset with it. The estimated effect should remain relatively unchanged.

  2. **Bootstrap Validation:** We replace the dataset with bootstrapped samples from the same dataset, and the estimated effect should also stay relatively stable.

Many of the techniques mentioned above, such as modeling, identification, and inference, aim to negate full causal analysis. Some utilize dummy findings or placebo treatments to reject causal analyses, whilst others use estimate processes, data subsets, and bootstrap testing for only particular phases [103]. To borrow terminology from software testing literature, the former is sometimes referred to as ”integration testing” and the latter as ”unit testing” [103].

The methods mentioned above are used in causal inference to test the validity of a causal claim. They are known as causal refuters because they make an effort to challenge or disprove a causal assertion by offering different explanations for the relationship that has been found between the treatment and outcome variables [23].

2.3.9 Causal machine learning

Causal inference and causal machine learning (CML) are important areas of research in machine learning that focus on understanding the causal relationships between variables, rather than just predicting outcomes based on correlations. Causal inference is the process of identifying the causal relationships between variables, while CML is the application of machine learning techniques to causal inference problems. CML can be used to extract low-dimensional representations of high-dimensional data, learn complex nonlinear relationships between variables, and process unstructured data in healthcare and other domains [4, 118].

CausalML is not a category of machine learning but a type of machine learning that uses a model to show how data is created. This helps us see what would have happened if we knew things beforehand.

Distinguishing between correlation and causation is a crucial challenge in causal inference and CML. Correlation pertains to the statistical relationship between two variables, while causation deals with the connection between cause and effect. Causal inference and CML aim to identify causal relationships by controlling for confounding variables and using techniques such as randomized controlled trials, instrumental variables, and natural experiments [12, 73].

2.4 Relevance of causal machine learning

Causal machine learning is a growing field that combines machine learning with causal inference to improve the accuracy of predictive models. The causal inference has the advantage of implying or inferencing the cause of one outcome from one treatment or cause. But what if the treatment or cause was different? Can it have the same outcome? This is the type of question causal machine learning helps answer. By taking advantage of machine learning approaches and combing it with causal inference, we introduce a way for an intervention to happen and update our predictive model on the go to infer a certain outcome.

Machine learning struggles with causality [28], since it is primarily reliant on massive predetermined sets of data [100]. Likewise, if there were multiple scenarios added to the predictive model, the accuracy would increase. Rather than depending on predefined correlations between data sets, these models allow the AI system to understand the causal variables and their impact on the environment. This would enable the system to detect objects despite minor alterations.

The research paper titled ”Towards Causal Representation Learning” [99] describes how artificial intelligent systems can learn causal representations and how the lack of such representations in machine learning algorithms and models is causing problems. In the paper, some limitations and further considerations are added.

Research is ongoing in integrating causal machine learning within different sectors like social and health sciences and finance, project management and others. For example, in a research paper titled: ”Mapping of machine learning approaches for description, prediction, and causal inference in the social and health sciences” [67], they describe the mapping of ML approaches to research purposes of description, prediction, and causal inference, mapping appropriate ML methods and giving empirical examples. More on the value added of machine learning on causal inference and vice versa is still being discovered [11].

2.5 Application of causal machine learning in project risk management

Causal machine learning (CML) is a powerful tool that can be applied in project risk management to identify causal relationships between project variables and potential risks. One application of CML is in risk assessment and management, where it can be used to predict the likelihood of risks occurring and to identify the factors that contribute to those risks [122]. Another application is in risk response planning, where CML can be used to select an appropriate set of a priori risk response strategies based on the causal relationships between project variables and risks[12].

In order to apply CML in project risk management, it is important to have a clear understanding of the different types of machine learning approaches and their applications. Recent research has mapped machine learning approaches into four categories: description, prediction, counterfactual prediction, and causal structural learning[125]. CML falls under the category of causal structural learning, which involves identifying the causal relationships between variables and using that information to make predictions for an informed decision-making. By using CML in project risk management, organizations can gain a deeper understanding of the factors that contribute to project risks and develop more effective risk response strategies.

3 Related works

Studies have looked at using machine learning to find risks in software projects. Here are a few examples.

1. ”Software Risk Prediction: Systematic Literature Review on Risk Causality and Risk Management in Software Projects” [5]: This project sets up a structure for examining the causes of risk in software projects and provides a plan for managing those risks.
2. ”Software project risk assessment using machine learning approaches” [72]: This paper mainly focuses on assessing the risks associated with software projects by utilizing various machine learning techniques to predict potential failures early on.
3. ”Applying Causal Learning to Improve Software Cost Estimation” [78]: In this project, they investigate methods for managing uncertainty in project planning and analyze traditional cost estimation models through causal learning.
4. ”An Analysis of the State of the Art of Machine Learning for Software Engineering Risk Management” [37]: This paper discusses managing risks in software projects. It covers traditional and newer machine learning methods for identifying and handling risks.
5. ”Risk Prediction Applied to Global Software Development using Machine Learning Methods” [79]: This study uses machine learning to predict risks in global software development and tests the model’s accuracy with real-world data.

These publications provide valuable insights on utilising causal machine learning techniques to identify and mitigate risks within software projects. The materials cover a broad range of risk management aspects, including identifying risk factors, risk assessment, cost estimation, and risk management strategies within a global software development context.

4 Description of the Investigation

4.1 Research design

The research is designed based on an objective: we create scenarios that we could answer, and these scenarios are most likely impact the overall software project development. Likewise, our first step was identifying the questions and problems project managers need help answering. In this case, the focus is on understanding the relationship between specific outcomes in the dataset and various factors that may contribute to them. This will involve selecting appropriate datasets and determining the variables and elements used in the analysis. Causal machine learning algorithms will be used to identify the causal relationship between variables/treatments and other factors such as duration and contribution complexity.

Once the causal relationships have been identified, the next step is to develop strategies to mitigate the project risks. We leave the design of the mitigation strategies to the respective managers.

A library called DoWhy is used for the experimentation and estimations [102].

4.2 Implementation Details

There are generally four steps we need to do. The figure below illustrates them 12.

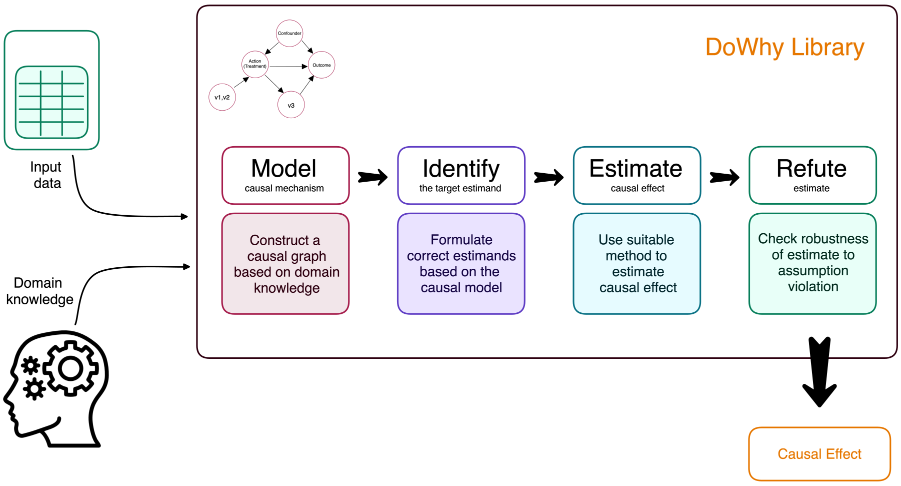

DoWhy Library
Confounder
Action (Treatment)
Outcome
v1,v2
v3
Input data
Domain knowledge
Model
causal mechanism
Identify
the target estimand
Estimate
causal effect
Refute
estimate
Construct a causal graph based on domain knowledge
Formulate correct estimands based on the causal model
Use suitable method to estimate causal effect
Check robustness of estimate to assumption violation
Causal Effect

Figure 12: Own creation based on Sharma, A., & Kiciman, E. (2020) [77]. DoWhy: An end-to-end library for causal inference

The picture explains the steps that dowhy chooses to use to return the causal effect. The steps are explained in our background, and we won't discuss the details of them from here on out.

In order to carry out the experiments, we needed to have a dataset and domain knowledge. The next section explains the formulation of the inputs.

4.3 Data preprocessing and cleaning

As mentioned, our experiment is conducted on randomly selected two projects from a dataset containing approximately 200 projects [40]. The dataset is basically the projects GitHub commit history. The data is both in JSON and CSV format [40].

Since our research focuses on identifying risks from the history of the software project development history, from the manager's perspective, we will preprocess the data.

The datasets are preprocessed in the following order.

Step	Description
1	Get a list of all the developers associated with the project
2	Obtain developers' experience over time since the project's creation date
3	Use NLP to categorize and retrieve task information from each individual commit
4	Set task information along with contribution complexity, duration, corrective commit probability and average cyclomatic complexity
5	Clean up data (convert values to a number)

Table 2: Steps for obtaining and processing developer and task data

Our target parameters were the following, and we managed to create them using different sources. Refere to the following table for a brief overview 4.

Table 3: Breif overview of the columns of my dataset

No.	Column	Description
1	task_type	The type of task associated with the commit.
2	has_bug_fixing	Whether the commit has bug fixing or not
3	has_code_refactoring	Whether the commit has code refactoring or not
4	loc	The lines of code changed in the commit
5	files_touched	The number of files affected by the commit
6	author	The author of the commit
7	committer	The committer of the commit
8	authored_at	The date and time the commit was authored
9	commited_at	The date and time the commit was committed
10	commit_time_in_day	The time a in day that commit was added
11	author_experience	The experience of the author at the time of the commit
12	committer_experience	The experience of the committer at the time of the commit

Continued on next page

Table 3: Breif overview of the columns of my dataset (Continued)

No.	**Column**	**Description**
13	code_quality	The corrective commit probablity as a code quality metrics for the commit
14	contribution_complexity	The complexity of the contribution in the commit
15	communication	If the author and the committer were different people, then we are sure there was communication on the commit
16	total_nloc	Total number of lines of code in the program
17	avg_nloc	Average number of lines of code per function or method
18	avg_ccn	Average cyclomatic complexity number, which measures the number of independent paths through the code
19	avg_token	Average number of tokens per line of code
20	fun_cnt	Total number of functions or methods in the program
21	warning_cnt	Total number of warnings generated by the code analysis tool
22	fun_rate	Ratio of functions or methods to the total number of lines of code.
23	nloc_rate	Ratio of non-comment lines of code to the total number of lines of code.
24	duration	The duration of the task associated with the commit

4.3.1 Parameter values

Column	Description
task_type has_bug_fixing has_code_refactoring	We used the commit message to get the task type, bug fixing status, and code refactoring status. The procedure includes using a natural language processor library for analyzing the text. The steps can be summarized as follows: 1. Define a set of keywords for each task category. These keywords represent phrases that are often used in commit messages related to each category. For example, we can use ”add”, ”new”, ”functionality” as keywords for the ”Feature requests or new feature” category. 2. Define a set of task categories that we want to identify in the commit messages. These categories can be based on the type of work being done, such as adding new features, fixing bugs, or improving code quality. 3. For each commit message, we analyze the message by checking if any of the keywords related to each category are present in the message. We can use a natural language processing tool like spaCy to tokenize and analyze the commit message. 4. For each category, we keep a count of the number of times that its keywords appear in the commit message. 5. We then identify the category with the highest count and use it as the task type for the commit message. If none of the categories have any keywords in the message, we classify the task type as ”Other”. 6. We can also identify if the task has bug fixing or code refactoring by counting the number of keywords in the corresponding keyword set that appear in the commit message. If the count is greater than zero, we consider that the task has bug fixing or code refactoring. The value of the has_bug_fixing and has_code_refactoring is a boolean, 0 (false) or 1(True). Whereas, the task type is a number and it maps to one of the following categories: (1) represents Feature requests or new feature. (2) represents Bug fixes. (3) represents Code Refactoring. (4) represents Documentation. (5) represents Testing. (6) represents Deployment. (7) represents Maintenance. (8) represents Other

Column	Description
loc	Since no standard convention exists to count lines of code and every tool that measures them will be slightly different, we devised our way to estimate from a commit to get the line of code changes [108]. Be advised that we used a formula for calculating the number of lines of code (LOC) as the sum of the number of lines of code added and the number of lines of code deleted where the source is ambiguous [108]. The total sum should not include blank or comment lines, but since they include them in a commit log of additions and deletions, we decided to keep the LOC number as just an estimate and not an absolute value. Despite this, we utilized the method for approximating the overall number of modified lines of code in a commit.The value is a number.
files_touched	We used the *files_changed_total* value from the original dataset for this value. The value is a number.
author committer	The author and committer values are represented by numbers, which correspond to the developer's index in our developers datasheet. This datasheet is created by us, using the orginal datasheet. It contains a list of developers with an associated index column.
committed_at authored_at	The value is a Date type. It contains the date and time the commit was committed or authored.
commit_time_in_day	We created this value based on the assumption that a developer's productivity changes at different times of the day, and thus, it may affect the overall commit as well. This variable can then be used as a feature in statistical models to determine if there is a correlation between the time of day a commit was made and the likelihood of bugs or other issues. By including this variable, we can better understand how a developer's productivity is affected by the time of day and make recommendations accordingly. The value is a number. It used the committed at timestamp and maps it to one of the following four numbers. (1) represents the time slot from 00:00 to 06:00 (2) represents the time slot from 06:00 to 12:00 (3) represents the time slot from 12:00 to 18:00 (4) represents the time slot from 18:00 to 23:59

Column	Description
author_experience committer_experience	To determine the level of experience of authors or committers at the time of their code commit, we used their past contributions as a reference. For instance, a value of 0 indicates it was their first time, while a value of 1 means they had committed or authored code the previous year.
code_quality	The corrective code probability (CCP) code quality metric is a tool we used here to estimate the value of the code quality at the time of contribution. The tool is used to approximate the probability that the commit has been made for corrective maintenance. We first analysed the commit messages to know whether or not that commit had corrections or bug fixes in them. We used an NLP library for the analysis of the messages. The metrics may suggest useful information for mangers at the time of the commit. For example, it indicates the frequency of corrective-related tasks. Thus, project managers can use it to identify the areas of code that need improvement and address them on time. The formula for CCP is: $$CCP = \frac{corrective\_commits}{corrective\_commits + non\_corrective\_commits} \quad (9)$$ where - ***corrective_commits*** is the number of corrective commits - ***corrective_commits + non_corrective_commits*** is the total number of commits. The CCP is a tool used to evaluate code quality by measuring the likelihood of corrective maintenance. It ranges from 0 to 1, and a higher value means a higher chance of corrective maintenance. It helps show the overall effort that is estimated to be used for improvement, which can be used as a risk predictor for project managers. These metrics claimed that "the bottom 10% of quality projects spend at least six times more effort fixing bugs than the top 10%"[6]. The metric also suggested that the higher the quality, the lower the code corrective probability"[6]. It is associated with many other factors, such as smaller files, lower coupling, fewer developers, lower developer churn, better onboarding, better productivity and others"[6]. At last, we used a complement of the probability to get an estimated code quality value. The formula for code quality is then: $$code\ quality = 1 - (corrective\ code\ probability) \quad (10)$$

Column	Description
contrinution_complexity	We used a tool that computes the complexity of a specified contribution using the commit hash and the source code. We needed to download the specific branch and source code from our original dataset. It returns a value with a scale that ranges from low to high [47]. This complexity value is an integer and can be used to determine the difficulty level involved in integrating a set of changes into the system. For instance, a low complexity value suggests that the changes were simple, while a high complexity value indicates that the changes were intricate and challenging to integrate. By analyzing the complexity of a contribution, we can better understand the effort required to incorporate it into the system and make more informed decisions regarding the development process. The value is a number and has the following indication of the complexity. (1) represents *low* (2) represents *moderate* (3) represents *medium* (4) represents *elevated* (5) represents *high*
communication	This is a boolean value indicating whether or not there the author of the commit and the commiter of the commit were different people. The value determines if the author and the approver/merger were different people. An *true* value would indicate that the committer, responsible for approving and merging, differed from the original author and a *false* value would indicate otherwise.
duration	The duration value in this context refers to the time taken to add the contribution commit into a branch, and is measured in hours. A duration value of 0 indicates that the merge was successful and did not take much time. However, a duration value above 0 indicates that an error or bug may have occurred during the merge, as the code was presumably checked and approved at the time of the merge. Therefore, a duration value above 0 is a good indication of an error or bug in the code.

Column	Description
total_nloc avg_nloc avg_ccn avg_token fun_cnt warning_cnt fun_rate nloc_rate	These parameters are used to measure the complexity and maintainability of software code. They include the total number of lines of code, the average number of lines of code per function or method, the average cyclomatic complexity number, the average number of tokens per line of code, the total number of functions or methods in the program, the total number of warnings generated by the code analysis tool, ratio of functions or methods to the total number of lines of code, and ratio of non-comment lines of code to the total number of lines of code. These metrics can identify potential issues in the code, such as functions with high cyclomatic complexity or low maintainability. We used a code analysis tool called ”Lizard” to generate these metrics for our code base. We had to go to each state of the commit to get the actual values of each task on the code base. These metrics we got using lizard will accurately describe a task’s impact on the overall code base.

Table 4: Explanitions of the dataset and how the values are generated

At last, we created our dataset with this values to use *code_quality*, *contribution_complexity* and *duration* as target outcomes and the other parameters as treatments for estimating causal effect.

4.4 Proposed approch and implementation

We created two scenarios for our experimentations and we made our assumption for the scenarios to create our diagrams. This is where the domain knowledge comes to play. However, it is important to acknowledge that these assumptions may lead to incorrect inferences. We advise the reader to confirm any speculations.

4.5 Experiment 1

What is the effect of contribution complexity on the cyclomatic complexity?

- Task: Effect estimation

Step 1. Create a Causal Graph

Using the parameters described from our dataset, we can use some general domain knowledge to define the cause-effect relationships in the form of a directed acyclic graph, which is represented in the following causal graph. The graph is as follows 13:

In the given diagram, the arrow pointing from B to W (B → W) represents a direct causal relationship, indicating that B is the cause of W. Based on this scenario, we have converted the following list of assumption into a causal diagram.

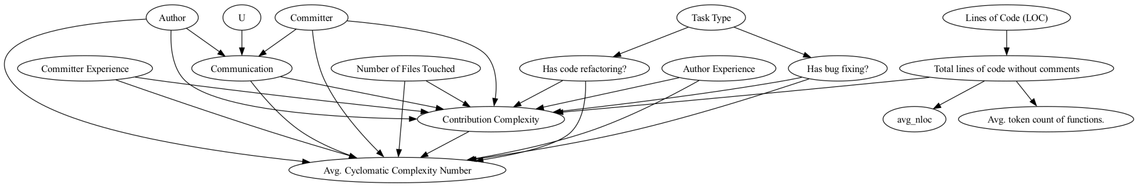


Figure 13: DAG using our domain knowledge to create assumptions

Our assumptions:

Task type impacts:

- Has bug fixing?: Certain task types may require bug fixing.
- Has code refactoring?: Some task types involve code refactoring.

Has bug fixing? impacts:

- Avg. Cyclomatic Complexity Number: Bug fixing involves complex logic, leading to higher CCN.
- Contribution complexity: Bug fixing adds complexity to an individual's contributions.

Has code refactoring? impacts:

- Avg. Cyclomatic Complexity Number: Code refactoring can affect control flow, influencing CCN.
- Contribution complexity: Code refactoring adds complexity to an individual's contributions.

Line of code (LOC) impacts:

- Total line of code without comments: Lines of code impact the total non-commented lines of code.

Total line of code without commit impacts:

- Avg. token count of functions: More lines of code generally result in a higher number of tokens.
- Contribution complexity: Larger codebases contribute to higher complexity in contributions.

Number of files touched impacts:

- Contribution complexity: Modifying multiple files adds complexity to contributions.
- Avg. Cyclomatic Complexity Number: Working with multiple files may involve complex logic.

Author impacts:

- Avg. Cyclomatic Complexity Number: Different authors have varying coding styles and expertise.
- Contribution complexity: Authors' experience and practices affect contribution complexity.
- Communication: Authors' communication impacts collaboration and understanding.

Committer impacts:

- Avg. Cyclomatic Complexity Number: Different committers have varying coding styles and expertise.
- Contribution complexity: Committers' experience and practices affect contribution complexity.
- Communication: Committers' communication impacts collaboration and understanding.

Author Experience impacts:

- Avg. Cyclomatic Complexity Number: More experienced authors often have better coding practices.
- Contribution complexity: Authors' experience influences the complexity of their contributions.

Committer Experience impacts:

- Avg. Cyclomatic Complexity Number: More experienced committers often have better coding practices.
- Contribution complexity: Committers' experience influences the complexity of their contributions.

Communication impacts:

- Contribution complexity: Effective communication reduces complexities in contributions.
- Avg. Cyclomatic Complexity Number: Collaborative communication can lead to code improvements and lower CCN.

Contribution complexity impacts:

- Avg. Cyclomatic Complexity Number: Contributions with higher complexity often have higher CCN.

Step 2. Identify Cause

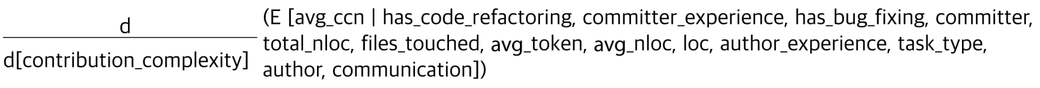
If modifying Treatment results in a change in Outcome while leaving everything else constant, we say that Treatment causes Outcome. So, in this stage, we determine the causal impact to be assessed by using features of the causal graph.

The cause calculated with the DoWhy library is the following:

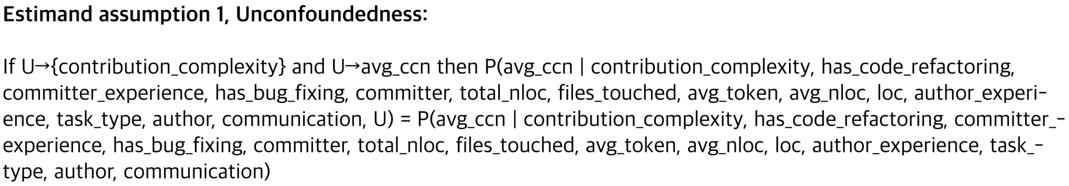
Estimand type: EstimandType.NONPARAMETRIC_ATE

Estimand: 1
Estimand name: backdoor

Estimand expression:



$$\frac{d}{d[\text{contribution\_complexity}]}(E[\text{avg\_ccn} \mid \text{has\_code\_refactoring, committer\_experience, has\_bug\_fixing, committer, total\_nloc, files\_touched, avg\_token, avg\_nloc, loc, author\_experience, task\_type, author, communication}])$$



Estimand assumption 1, Unconfoundedness:

If U→{contribution_complexity} and U→avg_ccn then P(avg_ccn | contribution_complexity, has_code_refactoring, committer_experience, has_bug_fixing, committer, total_nloc, files_touched, avg_token, avg_nloc, loc, author_experience, task_type, author, communication, U) = P(avg_ccn | contribution_complexity, has_code_refactoring, committer_experience, has_bug_fixing, committer, total_nloc, files_touched, avg_token, avg_nloc, loc, author_experience, task_type, author, communication)

Estimand: 2
Estimand name: iv

Estimand expression:

$$E\left[\frac{d}{d[total\_nloc]}(avg\_ccn)\cdot\left(\frac{d}{d[total\_nloc]}([contribution\_complexity])\right)^{-1}\right]$$

Estimand assumption 1, As-if-random:
If U→→avg_ccn then ¬(U →→{total_nloc})

Estimand assumption 2, Exclusion:
If we remove {total_nloc}→{contribution_complexity}, then ¬({total_nloc}→avg_ccn)

Estimand: 3
Estimand name: frontdoor

No such variable(s) found!

Step 3. Estimate the identified estimand

Realized estimand

Estimand : 1
Estimand name: backdoor

```
    causal_estimate_backdoor = model.estimate_effect(identified_estimand,
                                                 test_significance=True,
                                             method_name="backdoor.
                                             linear_regression", target_units="ate
                                             ")
```

b: duration ~ communication + author_experience + task_type + contribution_complexity + author + loc + avg_ccn + files_touched + total_nloc + avg_token + committer_experience + committer + code_quality + communication * author_experience + communication * task_type + communication * contribution_complexity + communication * avg_ccn + communication * loc + communication * files_touched + communication * total_nloc + communication * avg_token + communication * committer_experience + communication * code_quality

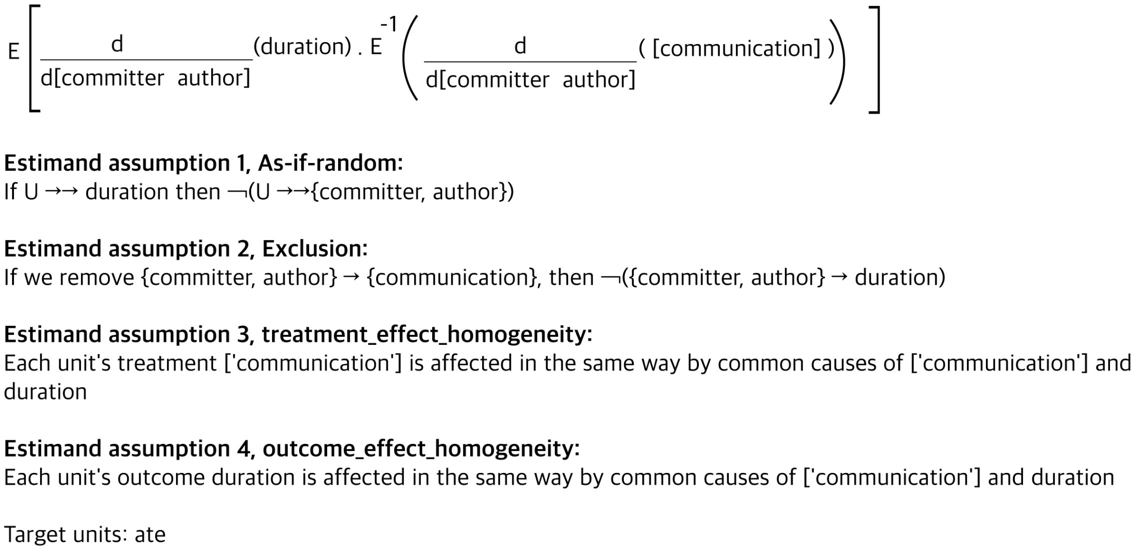
Target units: ate

Estimand : 2
Estimand name: backdoor

```
    causal_estimate_iv = model.estimate_effect(identified_estimand,
                                   method_name="iv.instrumental_variable",
                                   method_params = {'iv_instrument_name': '
                                        total_nloc'})
```

Realized estimand: Wald Estimator
Realized estimand type: EstimandType.NONPARAMETRIC_ATE

Estimates



$$E\left[\frac{d}{d[committer\ \ author]}(duration)\ .\ E^{-1}\left(\frac{d}{d[committer\ \ author]}(\ [communication]\ )\right)\right]$$

Estimand assumption 1, As-if-random:
If U →→ duration then ¬(U →→{committer, author})

Estimand assumption 2, Exclusion:
If we remove {committer, author} → {communication}, then ¬({committer, author} → duration)

Estimand assumption 3, treatment_effect_homogeneity:
Each unit's treatment ['communication'] is affected in the same way by common causes of ['communication'] and duration

Estimand assumption 4, outcome_effect_homogeneity:
Each unit's outcome duration is affected in the same way by common causes of ['communication'] and duration

Target units: ate

	Vue.js	**Brave-Browser**
Number of developers	291	60
Commits analyzed	1436	1458
Years	2017 - 2019	2016 - 2019
Branch	dev	master
Estimand name: backdoor	Mean value: 0.3512096141238845 Causal Estimate is 0.3512096141238845	Mean value: -0.00016735359078268175 Causal Estimate is -0.00016735359078268175
Estimand name: iv	Mean value: inf Causal Estimate is inf	Mean value: -12.136888881791265 Causal Estimate is -12.136888881791265

Step 4. Refute obtained results

Here we will use three refuters and only one of the estimate. For vue, we will look at it's backdoor estimate since it's the valid one. For Brave-browsers, we choose to look at the iv estimate since the value is greater and we will confirm whether infact the results are valid.

1. Bootstrap refuter:- The causal estimate should not differ considerably if our initial assumption was correct in this refutation.

Vue.js	**Brave-Browser**
Refute: Bootstrap Sample Dataset **Estimated effect:** 0.35120961412388452 **New effect:** 0.38756287923543886 **p value:** 0.5	**Refute:** Bootstrap Sample Dataset **Estimated effect:** -12.136888881791265 **New effect:** -12.741323654028136 **p value:** 0.94

2. Placebo Treatment Refuter:- If our assumptions were right, this newly discovered estimate should be 0.

   If the p-value is very small (usually less than 0.05), we can conclude that the observed outcome was not caused by chance alone, and we reject the null hypothesis in favor of the alternative hypothesis that the treatment has a true effect.

   It is crucial to emphasize, however, that statistical significance (as demonstrated by a low p-value) does not always imply that the observed effect is large or clinically significant.It's also worth mentioning that the p-value isn't the only factor to consider when analyzing study results. Other considerations should include the study methodology, sample size, and potential sources of bias.

Vue.js	**Brave-Browser**
Refute: Use a Placebo Treatment **Estimated effect:** 0.3512096141238845 **New effect:** 0.3512096141238844 **p value:** 0.0	**Refute:** Use a Placebo Treatment **Estimated effect:** -12.136888881791265 **New effect:** -18.073409110510624 **p value:** 0.94

3. Data Subset Refuter:- There shouldn't be much variety if our assumptions were appropriate.

Vue.js	**Brave-Browser**
Refute: Use a subset of data **Estimated effect:** 0.3512096141238845 **New effect:** 0.3506013251584047 **p value:** 0.96	**Refute:** Use a subset of data **Estimated effect:** -12.1368888817912655 **New effect:** -12.4139384313234 **p value:** 1.0

4.6 Experiment 2

What is the effect of communication on the time delay (bugs)?

- Task: Effect estimation

Step 1. Create a Causal Graph

We can define cause-effect relationships using general domain knowledge and the parameters from our dataset. The graph can be found in Figure 14.

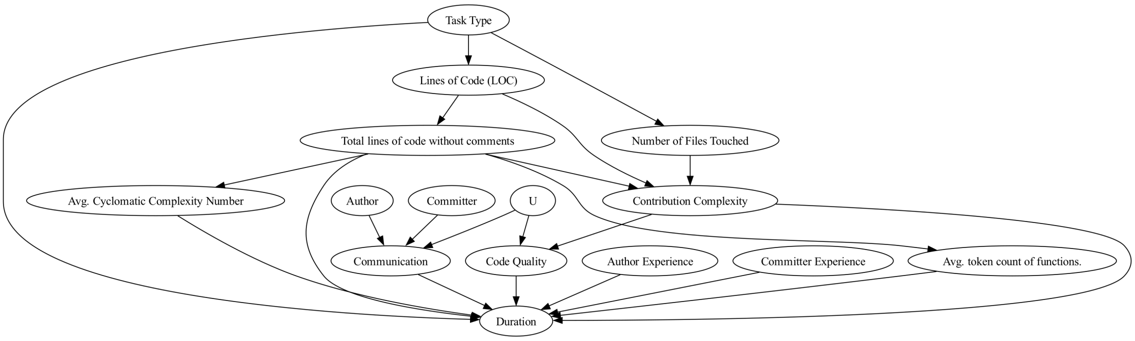


Figure 14: DAG using our domain knowledge to create assumptions

Our assumptions:

Unobserved confounders impact:

- Code quality: Unobserved confounders may influence the overall quality of the code.
- Communication: Unobserved confounders can affect the effectiveness of communication.

Task type impacts:

- Lines of code (LOC): Different task types can influence the number of lines of code.
- Number of files touched: Task type can influence the number of files involved.
- Duration: Different task types may require varying durations to complete.

Lines of code (LOC) impact:

- Total lines of code without comments:

LOC affects the total non-commented lines of code.

Total lines of code without comments impact:

- Contribution complexity: More lines of code contribute to higher complexity.
- Avg. Cyclomatic Complexity Number: LOC can influence the average CCN.
- Avg. token count of functions: LOC may affect the average token count.
- Duration: The size of the codebase can impact the overall duration of the task.

Avg. token count of functions impacts:

- Duration: Functions with a higher token count may require more time to complete.

Avg. Cyclomatic Complexity Number impacts:

- Duration: Higher CCN values can potentially increase the duration of the task.

Code quality impacts:

- Duration: Higher code quality standards may lead to reduced task duration.

Author impacts:

- Communication: The author's communication skills affect collaboration and understanding.

Committer impacts:

- Communication: The committer's communication skills impact collaboration and understanding.

Communication impacts:

- Duration: Effective communication can lead to smoother and faster task completion.

Lines of code (LOC) impact:

- Contribution complexity: LOC can contribute to the overall complexity of contributions.

Number of files touched impacts:

- Contribution complexity: Modifying multiple files adds complexity to contributions.

Author experience impacts:

- Duration: More experienced authors may complete tasks more efficiently.

Committer experience impacts:

- Duration: More experienced committers may complete tasks more efficiently.

Contribution complexity impacts:

1. Code quality: Higher contribution complexity can potentially lead to lower code quality.
2. Duration: More complex contributions may require additional time to complete.

Step 2. Identify Cause

If modifying Treatment results in a change in Outcome while leaving everything else constant, we say that Treatment causes Outcome. So, in this stage, we determine the causal impact to be assessed by using features of the causal graph.

The cause calculated with the DoWhy library is the following:

Estimand type: EstimandType.NONPARAMETRIC_ATE

Estimand: 1

Estimand name: backdoor

Estimand expression:

$$\frac{d}{d[\text{communication}]}\left(E[\text{duration} \mid \text{author\_experience, task\_type, contribution\_complexity, author, loc, avg\_ccn, files\_touched, total\_nloc, avg\_token, committer\_experience, committer, code\_quality}]\right)$$

Estimand assumption 1, Unconfoundedness:
If U → {communication} and U → duration then P (duration | communication, author_experience, task_type, contribution_complexity, author, loc, avg_ccn, files_touched, total_nloc, avg_token, committer_experience, committer, code_quality, U) = P(duration | communication, author_experience, task_type, contribution_complexity, author, loc, avg_ccn, files_touched, total_nloc, avg_token, committer_experience, committer, code_quality)

Estimand: 2

Estimand name: iv

Estimand expression:

$$E\left[\frac{d}{d[\text{committer} \quad \text{author}]}(\text{duration}) \cdot \left(\frac{d}{d[\text{committer} \quad \text{author}]}([\text{communication}])\right)^{-1}\right]$$

Estimand assumption 1, As-if-random:
If U →→ duration then ¬(U →→ {committer, author})

Estimand assumption 2, Exclusion:
If we remove {committer, author} → {communication}, then ¬({committer, author} → duration)

Estimand: 3

Estimand name: frontdoor

No such variable(s) found!

Step 3. Estimate the identified estimand

Realized estimand

Estimand : 1

Estimand name: backdoor

```
    causal_estimate_backdoor = model.estimate_effect(identified_estimand,
                                                    test_significance=True,
                                             method_name="backdoor.
                                             linear_regression", target_units="ate
                                             ")
```

b: duration ~ communication + author_experience + task_type + contribution_complexity + author + loc + avg_ccn + files_touched + total_nloc + avg_token + committer_experience + committer + code_quality + communication * author_experience + communication * task_type + communication * contribution_complexity + communication * avg_ccn + communication * loc + communication * files_touched + communication * total_nloc + communication * avg_token + communication * committer_experience + communication * code_quality

Target units: ate

Estimand : 2

Estimand name: backdoor

```
    causal_estimate_iv = model.estimate_effect(identified_estimand,
                                               method_name="iv.instrumental_variable",
                                               method_params = {'iv_instrument_name':
                                                              ['committer','author']})
```

Realized estimand: Wald Estimator
Realized estimand type: EstimandType.NONPARAMETRIC_ATE

$$E\left[\frac{d}{d[committer\ \ author]}(duration)\ .\ E^{-1}\left(\frac{d}{d[committer\ \ author]}(\ [communication]\ )\right)\right]$$

Estimand assumption 1, As-if-random:
If U →→ duration then ¬(U →→{committer, author})

Estimand assumption 2, Exclusion:
If we remove {committer, author}→{communication}, then ¬({committer, author}→duration)

Estimand assumption 3, treatment_effect_homogeneity:
Each unit's treatment ['communication'] is affected in the same way by common causes of ['communication'] and duration

Estimand assumption 4, outcome_effect_homogeneity:
Each unit's outcome duration is affected in the same way by common causes of ['communication'] and duration

Target units: ate

Estimates

	Vue.js	**Brave-Browser**
Number of developers	291	60
Commits analyzed	2770	1458
Years	2016 - 2019	2016 - 2019
Branch	dev	master

	Vue.js	**Brave-Browser**
Estimand name: backdoor	Mean value: -3.420785152629305 Causal Estimate is -3.420785152629305	Mean value: 193.8871873021942 Causal Estimate is 193.8871873021942
Estimand name: iv	Mean value: 49.81438882412487	Mean value: 289.33500209857243

Step 4. Refute obtained results

Here, once again, we used the three refuters:

1. Bootstrap refuter
2. Placebo Treatment Refuter
3. Data Subset Refuter

We choose to use the iv estimate since that is the one that we can confirm is the valid one with our refutation.

The results can be seen below.

Vue.js	Brave-Browser
Refute: Bootstrap Sample Dataset **Estimated effect:** 49.81438882412487 **New effect:** 50.77970383447133 **p value:** 0.96	**Refute:** Bootstrap Sample Dataset **Estimated effect:** 289.33500209857243 **New effect:** 294.4189151432695 **p value:** 0.84
Refute: Use a Placebo Treatment **Estimated effect:** 49.81438882412487 **New effect:** 89.41810936890319 **p value:** 0.0	**Refute:** Use a Placebo Treatment **Estimated effect:** 289.33500209857243 **New effect:** 198.80127364377583 **p value:** 0.0
Refute: Use a subset of data **Estimated effect:** 49.81438882412487 **New effect:** 49.6795514744566 **p value:** 1.0	**Refute:** Use a subset of data **Estimated effect:** 289.33500209857243 **New effect:** 290.40153792438304 **p value:** 0.98

5 Evaluation of the Investigation

Although more data and more analysis are needed to come to certain conclusions, the variance of the causal estimates on the same model suggests that a specific data analysis on individual software projects is most suitable for practical software project risk identification.

The causal estimate value indicates the covert impact the treatment may have on the specific outcome.

By analyzing causal relationships among various project factors, it becomes possible to identify early warning signs and indicators that may lead to future risks. This early detection empowers project teams to proactively mitigate risks, allocate resources effectively, and implement appropriate preventive measures, thus reducing the overall project vulnerability.

We assess the investigation conducted using DoWhy to estimate the causal effects fo variables derived from GitHub projects, specifically Vue and Brave-browser. The investigation aimed to answer two important questions: 1) What is the effect of communication on the time delay (bugs)? and 2) What is the effect of contribution complexity on cyclomatic complexity? The evaluation considers the success of the investigation, the reliability of the findings, and the implications of the results.

1. **Investigation Methodology:** DoWhy, a Python package for causal inference, is used, which denotes a thorough and organized method of estimating causal effects. DoWhy makes use of both statistical methods and graphical models to infer causation from observational data. The study ensures a strong foundation for its analysis by using this approach.

2. **Causal Effect Estimation:** Estimating the causal relationships between communication and time delay (bugs) and contribution complexity and cyclomatic complexity was the main goal of the project. It is important to highlight that some of the refutation was successful, showing that the model suited the data and there was little chance that the variables were unrelated.

3. **Communication Effect on Time Delay:** Using the dataset generated from GitHub projects, the analysis effectively gave insights into the causal effect of communication on time delay (bugs). This data suggests that developer collaboration may result in shorter or even longer bug resolution times. It implies that efficient teamwork and open lines of communication have a beneficial effect on the growth process or may hinder the growth, depending on the years and project types.

4. **Contribution Complexity on Cyclomatic Complexity:** The inquiry also looked at the relationship between cyclomatic complexity and contribution complexity. The findings provide insight into the relationship between the complexity of contributions and the complexity of the codebase. The relationship between code complexity and the work needed for maintenance, debugging, and overall program quality can be understood with the help of this information.

5.1 Limitations

There are several limitations that must be recognized, just like with any inquiry. The study, which potentially include biases and confounding variables, uses observational data from GitHub projects as its first point of departure. Additionally, it's possible that only programs like Vue and Brave-browser can use the findings. Since domain knowledge is a crucial component of the inquiry, replication of this model for different project datasets could not be precise. Other specific limitations should also be considered like:

1. Data Quality: The research is aware of the potential data quality limitations. The accuracy and dependability of the results may be impacted by insufficient or missing data. In this situation, the ability to estimate causal effects precisely may be constrained if the dataset utilized for analysis contains missing values or lacks key variables. Furthermore, as it depends on the accessibility and availability of pertinent data, gathering thorough and high-quality data from GitHub projects can be difficult.

2. Interpretability and Transparency: Another issue is the interpretability and transparency of the investigation's sophisticated models. While DoWhy provides a rigorous framework for causal inference, the intricacy of the models utilized may make interpreting and understanding the causal linkages revealed difficult. This constraint may impede the ability to effectively convey findings to stakeholders and make informed decisions based on the findings. In such circumstances, justifying and communicating the risks connected with the findings may also be difficult.

3. Generalizability and Context Dependency: The study recognizes the potential limitations of generalizability and context reliance. Causal correlations discovered in the Vue and Brave-browser projects may not be applicable to other projects or domains. Different projects may have distinct features, development techniques, team dynamics, and contextual factors that influence the applicability of causal estimations. As a result, when extending the findings to diverse circumstances, caution should be given.

5.1.1 Further research

Further research in the field of software development and software project management should attempt to solve the limits indicated in this study while also exploring new avenues of inquiry. Future study could include the investigation of additional elements that could influence software development outcomes. Investigating the impact of team size, developer experience, or project complexity, for example, on key metrics like as code quality, issue resolution time, or project success, could reveal valuable insights. Furthermore, conducting follow-up research to explore the long-term effects of communication techniques, contribution complexity, and other elements on software development projects may aid in our understanding of their influence over time.

More study might be conducted to investigate specific tactics or interventions to improve communication and minimize complexity, resulting in more efficient and effective software development should be considered.

Another important area for future research is the development of intervention studies. By implementing customized interventions aimed at increasing communication skills or managing contribution complexity inside software development teams, researchers can evaluate the causal influence of tailored interventions on project success. Such research would provide software development companies with actionable insights and evidence-based recommendations for refining their processes and increasing overall project success. Furthermore, investigating the interaction effects of various variables and identifying potential trade-offs or synergies between them would result in a more thorough understanding of the complex dynamics within software development projects.

6 Conclusions

Identifying software project risks using a causal machine learning approach to increase in project mangament is one that we aimed to aid here. Using DoWhy to estimate the causal effects of variables derived from GitHub projects, specifically Vue and Brave-browser, we were able to provid valuable insights into the relationship between communication, time delay (bugs), contribution complexity, and cyclomatic complexity.

The research satisfactorily answered the questions regarding the effects of communication on time delay and contribution complexity on cyclomatic complexity using the supplied dataset. The model created by general knowledge, passed some of the refutation test, indicating that it fit the data well and that the possibility of variables being related is present.

However, it is important to acknowledge the investigation's limits. The significance of data quality, interpretability, and generalizability should be addressed in future study. Improving data quality, improving the interpretability of complex models, and undertaking replication studies across other project datasets will increase the findings' dependability. As for generalization, we suggest domain specific knowledge should be used for the adjustments of the model. Furthermore, future research should evaluate the causal influence of treatments aiming at increasing communication and managing contribution complexity within software development teams by exploring additional factors and conducting intervention studies.

In conclusion, the investigation serves as a basis for further research in this field. More research should expand on the findings, rectify the constraints, and investigate new areas of study to better understand the causal links in software development projects. Finally, this research will help to advance best practices and techniques, resulting in more efficient and effective software development processes.

References

[1] Rusul Abduljabbar et al. "Applications of artificial intelligence in transport: An overview". In: *Sustainability* 11.1 (2019), p. 189.

[2] Team AdaptiveWork. *What are the objectives of project management?* Apr. 2022. URL: https://blog.planview.com/objectives-of-project-management/.

[3] Ahmed Alaa and Mihaela Van Der Schaar. "Validating causal inference models via influence functions". In: *International Conference on Machine Learning*. PMLR. 2019, pp. 191–201.

[4] Ahmed M Alaa and Mihaela van der Schaar. "Causal machine learning in health-care: a review". In: *arXiv preprint arXiv:1801.06618* (2018).

[5] Tiago Alves, Ana Gomes, and João M Fernandes. "Software Risk Prediction: Systematic Literature Review on Risk Causality and Risk Management in Software Projects". In: *IEEE Access* 7 (2019), pp. 102722–102739.

[6] Idan Amit and Dror G Feitelson. "The Corrective Commit Probability Code Quality Metric". In: *arXiv preprint arXiv:2007.10912* (2020).

[7] Susan Athey and Stefan Wager. "Estimating treatment effects with causal forests: An application". In: *Observational Studies* 5.2 (2019), pp. 37–51.

[8] No author. *Artificial Intelligence*. Definition. Accessed on April 30, 2023. No year. URL: https://www.techtarget.com/search/query?q=artificial+intelligence.

[9] Amin Azari, Mustafa Ozger, and Cicek Cavdar. "Risk-aware resource allocation for URLLC: Challenges and strategies with machine learning". In: *IEEE Communications Magazine* 57.3 (2019), pp. 42–48.

[10] Microsoft Azure. "Artificial intelligence (AI) vs. machine learning (ML)". In: *Microsoft Azure* (2021). URL: https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/artificial-intelligence-vs-machine-learning/.

[11] Anna Baiardi and Andrea A. Naghi. *The Value Added of Machine Learning to Causal Inference: Evidence from Revisited Studies*. 2021. arXiv: 2101.00878 [econ.GN].

[12] Elias Bareinboim and Judea Pearl. "Causal inference and the data-fusion problem". In: *Proceedings of the National Academy of Sciences* 113.27 (2016), pp. 7345–7352.

[13] G. M. Becker. "A practical risk management approach". In: *PMI® Global Congress 2004—North America*. Anaheim, CA: Project Management Institute, 2004. URL: https://www.pmi.org/learning/library/practical-risk-management-approach-8248.

[14] Marc F Bellemare and Jeffrey R Bloem. *Estimating Treatment Effects Using the Front-Door Criterion*. 2020. URL: https://www.canr.msu.edu/afre/events/Bellemare%20and%20Bloem%20(2020).pdf.

[15] Irad Ben-Gal. "Bayesian Networks". In: *Encyclopedia of Statistics in Quality and Reliability*. John Wiley & Sons, Ltd, 2008. ISBN: 9780470061572. DOI: https://doi.org/10.1002/9780470061572.eqr089. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/9780470061572.eqr089. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470061572.eqr089.

[16] James Brophy. *Causal Inference*. Bookdown. 2019. URL: https://bookdown.org/jbrophy115/bookdown-clinepi/causal.html.

[17] James Brophy. *Causal Inference*. https://bookdown.org/jbrophy115/bookdown-clinepi/causal.html. Accessed on May 3, 2023, Uses reference: Pearl, J, M Glymour, and NP Jewell. 2016. Causal Inference in Statistics. John Wiley. Book. 2021.

[18] James Brophy. *Confounding*. https://bookdown.org/jbrophy115/bookdown-clinepi/confounding.html. Accessed: April 30, 2023. 2018.

[19] Longbing Cao. "Ai in finance: challenges, techniques, and opportunities". In: *ACM Computing Surveys (CSUR)* 55.3 (2022), pp. 1–38.

[20] Bruce W. Carlson. *Simpson's paradox*. https://www.britannica.com/topic/Simpsons-paradox. Accessed 29 April 2023. 2023.

[21] Silvia Chiappa and William S. Isaac. "A Causal Bayesian Networks Viewpoint on Fairness". In: *Privacy and Identity Management. Fairness, Accountability, and Transparency in the Age of Big Data*. Springer International Publishing, 2019, pp. 3–20. DOI: 10.1007/978-3-030-16744-8_1. URL: https://doi.org/10.1007%2F978-3-030-16744-8_1.

[22] Abhimanyu Chopra, Abhinav Prashar, and Chandresh Sain. "Natural language processing". In: *International journal of technology enhancements and emerging engineering research* 1.4 (2013), pp. 131–134.

[23] Louis Anthony Cox Jr. "Improving causal inferences in risk analysis". In: *Risk analysis* 33.10 (2013), pp. 1762–1771.

[24] Pádraig Cunningham, Matthieu Cord, and Sarah Jane Delany. "Supervised learning". In: *Machine learning techniques for multimedia: case studies on organization and retrieval* (2008), pp. 21–49.

[25] Hoa Khanh Dam et al. "Towards effective AI-powered agile project management". In: *2019 IEEE/ACM 41st international conference on software engineering: new ideas and emerging results (ICSE-NIER)*. IEEE. 2019, pp. 41–44.

[26] UC Davis. *Directed Acyclic Graphs (DAGs) and Regression for ...* https://health.ucdavis.edu/hsr/education/courses/hsr210/lecture-notes/hsr210-lecture-6.pdf. Accessed: April 30, 2023. 2021.

[27] Peter Dayan, Maneesh Sahani, and Grégoire Deback. "Unsupervised learning". In: *The MIT encyclopedia of the cognitive sciences* (1999), pp. 857–859.

[28] Ben Dickson. *Machine learning and causality: Why it matters*. https://bdtechtalks.com/2021/03/15/machine-learning-causality/. Mar. 2021.

[29] Filip Karlo Došilović, Mario Brčić, and Nikica Hlupić. "Explainable artificial intelligence: A survey". In: *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. 2018, pp. 0210–0215. DOI: 10.23919/MIPRO.2018.8400040.

[30] Raaz Dwivedi et al. "Stable discovery of interpretable subgroups via calibration in causal studies". In: *International Statistical Review* 88 (2020), S135–S178.

[31] Felix Elwert and Christopher Winship. "Endogenous selection bias: The problem of conditioning on a collider variable". In: *Annual review of sociology* 40 (2014), pp. 31–53.

[32] Research with Fawad. *Mediation Analysis using SMART-PLS.* 2021. URL: https://researchwithfawad.com/index.php/lp-courses/basic-and-advance-data-analysis-using-smart-pls/mediation-analysis-interpretation-and-reporting-using-smart-pls/.

[33] Alexsandro Souza Filippetto, Robson Lima, and Jorge Luis Victória Barbosa. "A risk prediction model for software project management based on similarity analysis of context histories". In: *Information and Software Technology* 131 (2021), p. 106497.

[34] Dylan J Foster and Vasilis Syrgkanis. "Statistical learning with a nuisance component". In: *Conference on Learning Theory*. PMLR. 2019, pp. 1346–1348.

[35] Isabel Fulcher et al. "Robust Inference on Population Indirect Causal Effects: The Generalized Front Door Criterion". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 82 (Nov. 2019). DOI: 10.1111/rssb.12345.

[36] A Deiva Ganesh and P Kalpana. "Future of artificial intelligence and its influence on supply chain risk management–A systematic review". In: *Computers & Industrial Engineering* (2022), p. 108206.

[37] Ana Garcia et al. "An Analysis of the State of the Art of Machine Learning for Software Engineering Risk Management". In: *IEEE Access* 7 (2019), pp. 10798–10814.

[38] Sammi Caramela Gonzalez. "How Artificial Intelligence Will Transform Businesses". In: *Business News Daily* (2021). URL: https://www.businessnewsdaily.com/9402-artificial-intelligence-business-trends.html.

[39] Sander Greenland, Jude Pearl, and James M. Robins. "Using directed acyclic graphs to guide analyses of family studies". In: *Statistics in medicine* 21.11 (2002), pp. 1731–1740.

[40] Marios-Stavros Grigoriou and Kostas Kontogiannis. "Dataset for published paper Project Features That Make Machine-Learning Based Fault Proneness Analysis Successful". In: (Sept. 2022). DOI: 10.6084/m9.figshare.21044443.v2. URL: https://figshare.com/articles/dataset/Project_Features_That_Make_Machine-Learning_Based_Fault_Proneness_Analysis_Successful/21044443.

[41] Douglas Gunzler et al. "Introduction to mediation analysis with structural equation modeling". In: *Shanghai archives of psychiatry* 25.6 (2013), p. 390.

[42] Alula Hadgu and William Miller. "Using a combination of reference tests to assess the accuracy of a diagnostic test by A. Alonzo and M. Pepe, Statistics in Medicine 1999; 18: 2987–3003". In: *Statistics in medicine* 20.4 (2001), pp. 656–658.

[43] Andrew F Hayes. "Best (but oft-forgotten) practices: mediation analysis". In: *Journal of counseling psychology* 59.2 (2017), pp. 10–18. URL: https://doi.org/10.3945/ajcn.117.152546.

[44] Hua He, Pan Wu, Ding-Geng Chen, et al. *Statistical causal inferences and their applications in public health research*. Springer, 2016.

[45] National Institutes of Health. *Use of Directed Acyclic Graphs*. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2794468/. Accessed: April 30, 2023. 2009.

[46] Agency for Healthcare Research and Quality. *Chapter 7. Covariate Selection*. 2021. URL: https://effectivehealthcare.ahrq.gov/sites/default/files/ch-7-user-guide-to-ocer_130129.pdf.

[47] HelgeCPH. *Tool to compute contribution complexity*. https://github.com/HelgeCPH/contribution-complexity. 2023.

[48] Miguel Hernan and Jamie Robins. *A Crash Course in Causality: Inferring Causal Effects from Observational Data*. https://www.coursera.org/learn/crash-course-in-causality. Accessed on May 3, 2023. 2019.

[49] Wenmei Hu et al. "A deep learning method to estimate independent source number". In: *2017 4th International Conference on Systems and Informatics (ICSAI)*. 2017, pp. 1055–1059. DOI: 10.1109/ICSAI.2017.8248441.

[50] Douglas W Hubbard. *The failure of risk management: Why it's broken and how to fix it*. John Wiley & Sons, 2020.

[51] IBM. *Machine Learning*. https://www.ibm.com/topics/machine-learning. accessed on April 30, 2023.

[52] IBM. *What is Artificial Intelligence (AI)*. https://www.ibm.com/cloud/learn/what-is-artificial-intelligence. 2023.

[53] IBM. "What is Natural Language Processing?" In: *IBM* (2023). URL: https://www.ibm.com/cloud/learn/natural-language-processing.

[54] SAS Institute Inc. "Causal Effect Estimands: Interpretation, Identification, and Guidance for Selection". In: *Proceedings of the SAS Global Forum* (2020), pp. 1–12. URL: https://support.sas.com/resources/papers/proceedings20/4322-2020.pdf.

[55] Investopedia. *Artificial Intelligence: What It Is and How It Is Used*. https://www.investopedia.com/terms/a/artificial-intelligence-ai.asp. Accessed: April 30, 2023. 2023.

[56] Investopedia. *Artificial Intelligence: What It Is and How It Is Used*. https://www.investopedia.com/terms/a/artificial-intelligence-ai.asp. 2023.

[57] Surbhi Jain and Joydip Dhar. "Image based search engine using deep learning". In: *2017 Tenth International Conference on Contemporary Computing (IC3)*. 2017, pp. 1–7. DOI: 10.1109/IC3.2017.8284301.

[58] Christian Janiesch, Patrick Zschech, and Kai Heinrich. "Machine learning and deep learning". In: *Electronic Markets* 31.3 (2021), pp. 685–695.

[59] Jean Kaddour et al. *Causal Machine Learning: A Survey and Open Problems*. 2022. arXiv: 2206.15475 [cs.LG].

[60] Skylar Kerzner. *A complete guide to causal inference*. Feb. 2022. URL: https://towardsdatascience.com/a-complete-guide-to-causal-inference-8d5aaca68a47.

[61] Nancy Krieger and George Davey Smith. "Directed acyclic graphs: An underutilized tool for child ..." In: *Social Science & Medicine* 44.6 (1997), pp. 859–866.

[62] Rajesh Kumar. "How AI Will Transform Project Management". In: *Harvard Business Review* 101.1 (Feb. 2023), pp. 78–85.

[63] Matt J Kusner et al. "Counterfactual fairness". In: *Advances in neural information processing systems* 30 (2017).

[64] N Lavanya and T Malarvizhi. "Risk analysis and management: a vital key to effective project management". In: Project Management Institute. 2008.

[65] Paola Lecca. "Machine Learning for Causal Inference in Biological Networks: Perspectives of This Challenge". In: *Frontiers in Bioinformatics* 1 (2021). ISSN: 2673-7647. DOI: 10.3389/fbinf.2021.746712. URL: https://www.frontiersin.org/articles/10.3389/fbinf.2021.746712.

[66] David J Lederer et al. "Control of confounding and reporting of results in causal inference studies. Guidance for authors from editors of respiratory, sleep, and critical care journals". In: *Annals of the American Thoracic Society* 18.1 (2021). Accessed: April 30, 2023, pp. 22–28. DOI: 10.1513/AnnalsATS.202008-1012PS. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8967310/.

[67] Anja K Leist et al. "Mapping of machine learning approaches for description, prediction, and causal inference in the social and health sciences". In: *Science Advances* 8.42 (2022), eabk1942.

[68] Heng Li et al. "Estimands in observational studies: Some considerations beyond ICH E9 (R1)". In: *Pharmaceutical Statistics* 21.5 (2022), pp. 835–844.

[69] Xiaochun Li and Changyu Shen. "Doubly robust estimation of causal effect: upping the odds of getting the right answers". In: *Circulation: Cardiovascular Quality and Outcomes* 13.1 (2020), e006065.

[70] Ilya Lipkovich, Bohdana Ratitch, and Craig H Mallinckrodt. "Causal inference and estimands in clinical trials". In: *Statistics in Biopharmaceutical Research* 12.1 (2020), pp. 54–67.

[71] Yueran Liu, Peng Zhang, and Peter B. Gilbert. "Unbiased Causal Inference From an Observational Study". In: *Circulation: Cardiovascular Quality and Outcomes* 12.9 (2019), e006065. DOI: 10.1161/CIRCOUTCOMES.119.006065. eprint: https://doi.org/10.1161/CIRCOUTCOMES.119.006065. URL: https://doi.org/10.1161/CIRCOUTCOMES.119.006065.

[72] Yuhang Liu et al. "Software project risk assessment using machine learning approaches". In: *2018 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C)*. IEEE. 2018, pp. 1–8.

[73] Sara López-Pintado and James M Robins. "Towards a general and automatic approach to statistical inference from causal effects". In: *European Journal of Epidemiology* 31.9 (2016), pp. 879–887.

[74] David P MacKinnon and Sophia J Lamp. "A unification of mediator, confounder, and collider effects". In: *Prevention Science* 22.8 (2021), pp. 1185–1193.

[75] CH Mallinckrodt et al. "Aligning estimators with estimands in clinical trials: putting the ICH E9 (R1) guidelines into practice". In: *Therapeutic innovation & regulatory science* 54 (2020), pp. 353–364.

[76] John McCarthy. "WHAT IS ARTIFICIAL INTELLIGENCE?" In: (2004). Retrieved on April 23, 2023 from https://www-formal.stanford.edu/jmc/whatisai.pdf. URL: http://www-formal.stanford.edu/jmc/.

[77] Microsoft. *DoWhy: A Python library for causal inference*. https://github.com/microsoft/dowhy. 2021.

[78] Leandro L Minku and Xin Yao. "Applying Causal Learning to Improve Software Cost Estimation". In: *2013 IEEE International Conference on Software Maintenance*. IEEE. 2013, pp. 70–79.

[79] Leandro L Minku and Xin Yao. “Risk prediction applied to global software development using machine learning methods”. In: *2012 IEEE 23rd International Symposium on Software Reliability Engineering*. IEEE. 2012, pp. 211–220.

[80] Andrew K Munns and Bassam F Bjeirmi. “The role of project management in achieving project success”. In: *International journal of project management* 14.2 (1996), pp. 81–87.

[81] Muddasar Naeem, Syed Tahir Hussain Rizvi, and Antonio Coronato. “A Gentle Introduction to Reinforcement Learning and its Application in Different Fields”. In: *IEEE Access* 8 (2020), pp. 209320–209344. DOI: 10.1109/ACCESS.2020.3038605.

[82] Brady Neal, Chin-Wei Huang, and Sunand Raghupathi. “Realcause: Realistic causal inference benchmarking”. In: *arXiv preprint arXiv:2011.15007* (2020).

[83] Xinkun Nie and Stefan Wager. “Quasi-oracle estimation of heterogeneous treatment effects”. In: *Biometrika* 108.2 (2021), pp. 299–319.

[84] Fred Niederman. “Project management: openings for disruption from AI and advanced analytics”. In: *Information Technology & People* (2021).

[85] B Nithya and V Ilango. “Predictive analytics in health care using machine learning tools and techniques”. In: *2017 International Conference on Intelligent Computing and Control Systems (ICICCS)*. IEEE. 2017, pp. 492–499.

[86] Abdul Mannan Omi et al. “Multiple Authors Identification from Source Code Using Deep Learning Model”. In: *2021 International Conference on Electronics, Communications and Information Technology (ICECIT)*. 2021, pp. 1–4. DOI: 10.1109/ICECIT54077.2021.9641497.

[87] Wanli Ouyang, Xiao Chu, and Xiaogang Wang. “Multi-source Deep Learning for Human Pose Estimation”. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 2337–2344. DOI: 10.1109/CVPR.2014.299.

[88] Trishan Panch, Heather Mattie, and Leo Anthony Celi. “The “inconvenient truth” about AI in healthcare”. In: *NPJ digital medicine* 2.1 (2019), p. 77.

[89] Trishan Panch, Peter Szolovits, and Rifat Atun. “Artificial intelligence, machine learning and health systems”. In: *Journal of global health* 8.2 (2018).

[90] Álvaro Parafita and Jordi Vitrià. “Estimand-Agnostic Causal Query Estimation With Deep Causal Graphs”. In: *IEEE Access* 10 (2022), pp. 71370–71386. DOI: 10.1109/ACCESS.2022.3188395.

[91] Jude Pearl. “Quantifying biases in causal models: classical confounding vs collider-stratification bias”. In: *Epidemiology* 14.3 (2003). Accessed: April 30, 2023, pp. 300–306. DOI: 10.1097/01.EDE.0000057649.41132.22. URL: https://journals.lww.com/epidem/Fulltext/2003/05000/Quantifying_Biases_in_Causal_Models__Classical.9.aspx.

[92] Judea Pearl. *Causality*. Cambridge university press, 2009.

[93] Judea Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan kaufmann, 1988.

[94] Steffi Pohl et al. “Unbiased Causal Inference From an Observational Study: Results of a Within-Study Comparison”. In: *Educational Evaluation and Policy Analysis* 31.4 (2009), pp. 463–479. DOI: 10.3102/0162373709343964. eprint: https://doi.org/10.3102/0162373709343964. URL: https://doi.org/10.3102/0162373709343964.

[95] Bohdana Ratitch et al. "Choosing estimands in clinical trials: putting the ICH E9 (R1) into practice". In: *Therapeutic innovation & regulatory science* 54 (2020), pp. 324–341.

[96] Deb Richardson. "What is AI/ML and why does it matter to your business?" In: *Red Hat* (2021). URL: https://www.redhat.com/en/blog/what-aiml-and-why-does-it-matter-your-business.

[97] Julia M Rohrer. "Thinking clearly about correlations and causation: Graphical causal models for observational data". In: *Advances in methods and practices in psychological science* 1.1 (2018), pp. 27–42.

[98] Ken Rose. *Book Review: A Guide to the Project Management Body of Knowledge (PMBOK® Guide)-2000 Edition*. 2001.

[99] Bernhard Schölkopf et al. "Toward Causal Representation Learning". In: *Proceedings of the IEEE* 109.5 (2021), pp. 612–634. DOI: 10.1109/JPROC.2021.3058954.

[100] Bernhard Schölkopf et al. *Towards Causal Representation Learning*. 2021. arXiv: 2102.11107 [cs.LG].

[101] ScienceDirect. *Instrumental Variable Analysis - an overview*. 2021. URL: https://www.sciencedirect.com/topics/medicine-and-dentistry/instrumental-variable-analysis.

[102] Amit Sharma and Emre Kiciman. "DoWhy: An End-to-End Library for Causal Inference". In: *arXiv preprint arXiv:2011.04216* (2020).

[103] Amit Sharma et al. "Dowhy: Addressing challenges in expressing and validating causal assumptions". In: *arXiv preprint arXiv:2108.13518* (2021).

[104] Ilya Shpitser and Judea Pearl. "Identification of joint interventional distributions in recursive semi-Markovian causal models". In: *Proceedings of the National Conference on Artificial Intelligence*. Vol. 21. 2. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999. 2006, p. 1219.

[105] Shashi Pal Singh et al. "Machine translation using deep learning: An overview". In: *2017 International Conference on Computer, Communications and Electronics (Comptelix)*. 2017, pp. 162–167. DOI: 10.1109/COMPTELIX.2017.8003957.

[106] MIT Sloan. *Machine learning, explained*. https://mitsloan.mit.edu/ideas-made-to-matter/machine-learning-explained. Accessed: April 30, 2023. 2023.

[107] Statistics Solutions. *Mediator Variable*. 2021. URL: https://www.statisticssolutions.com/dissertation-resources/descriptive-statistics/mediator-variable/.

[108] *Stack Overflow*. https://stackoverflow.com/questions/353380/is-there-a-standard-way-to-count-lines-of-code. Accessed on May 4, 2023.

[109] Harry Surden. "Machine learning and law". In: *Wash. L. Rev.* 89 (2014), p. 87.

[110] Dingke Tang et al. "Variable Selection for Doubly Robust Causal Inference". In: *arXiv preprint arXiv:2007.14190* (2020).

[111] TechTarget. *What is artificial intelligence (AI)? - AI definition and how it ...* https://www.techtarget.com/searchenterpriseai/definition/AI-Artificial-Intelligence. 2023.

[112] John Tolan et al. "On the Interpretation and Use of Mediation: Multiple Perspectives on Mediation Analysis". In: *Frontiers in Psychology* 8 (2017), p. 1984. URL: https://doi.org/10.3389/fpsyg.2017.01984.

[113] Columbia University. *Instrumental Variables.* 2021. URL: https://www.publichealth.columbia.edu/research/population-health-methods/instrumental-variables.

[114] Stanford University. *Conceptual Frameworks and Directed Acyclic Graphs (DAGs).* https://med.stanford.edu/biostatistics/education/DAGs.html. Accessed: April 30, 2023. 2021.

[115] Unknown. *Detection of Offensive Language in Social Media Posts.* Scientific Figure on ResearchGate. [accessed 28 Apr, 2023]. Unknown. URL: https://www.researchgate.net/figure/Relationship-between-AI-ML-DL-and-NLP-7_fig8_343079524.

[116] Darrell M West and John R Allen. "How artificial intelligence is transforming the world". In: *Brookings Institution* (2018). URL: https://www.brookings.edu/research/how-artificial-intelligence-is-transforming-the-world/.

[117] Jason Westland. *The project management life cycle: A complete step-by-step methodology for initiating planning executing and closing the project.* Kogan Page Publishers, 2007.

[118] Jenna Wiens et al. "Do no harm: a roadmap for responsible machine learning for health care". In: *Nature Medicine* 25.9 (2019), pp. 1337–1340.

[119] Wikipedia. *Artificial intelligence.* https://en.wikipedia.org/wiki/Artificial_intelligence. 2023.

[120] Wikipedia. *Instrumental Variables Estimation.* 2021. URL: https://en.wikipedia.org/wiki/Instrumental_variables_estimation.

[121] Wikipedia. *Mediation (statistics).* 2021. URL: https://en.wikipedia.org/wiki/Mediation_(statistics).

[122] Dan Wu and David L Olson. "Risk assessment and risk management: Review of recent advances on their foundation". In: *International Journal of Production Research* 58.7 (2020), pp. 1957–1973.

[123] Ziqi Xu et al. *Causal Effect Estimation with Variational AutoEncoder and the Front Door Criterion.* 2023. arXiv: 2304.11969 [cs.LG].

[124] Yanyan Zhang et al. "Instrumental Variable Analysis in the Presence of Endogenous and Exogenous Moderators". In: *Journal of Research in Health Sciences* 18.2 (2018), e00420. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5994515/.

[125] Yudong Zhang and Yufei Zhang. "A method for risk response planning in project portfolio management based on mathematical optimization". In: *Journal of Intelligent & Fuzzy Systems* 36.1 (2019), pp. 1–10.