A New Science for Reliability

IEEE RAMS - Tutorial 35

Robert W. Stoddard

Robert W. Stoddard
Software Engineering Institute
4500 Fifth Ave (office 3110),
Pittsburgh, PA, 15367, USA
e-mail: rws@sei.cmu.edu

Copyright 2020 Carnegie Mellon University.

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

The view, opinions, and/or findings contained in this material are those of the author(s) and should not be construed as an official Government position, policy, or decision, unless designated by other documentation.

References herein to any specific commercial product, process, or service by trade name, trade mark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by Carnegie Mellon University or its Software Engineering Institute.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

[DISTRIBUTION STATEMENT A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

Internal use:* Permission to reproduce this material and to prepare derivative works from this material for internal use is granted, provided the copyright and “No Warranty” statements are included with all reproductions and derivative works.

External use:* This material may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other external and/or commercial use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

* These restrictions do not apply to U.S. government entities.

Carnegie Mellon® is registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

DM20-0769

SUMMARY AND PURPOSE

Although software reliability estimation based on results of software testing has been the subject of decades of software reliability publications, software reliability prediction using measures from the design and development of software has lagged. Software reliability prediction from Rome Air Development Center (RADC) in the 1980's enjoyed some popularity and then fell into disuse. Such prediction attempts were built on multiple regression models which, at times, suffered in precision and accuracy. Subsequent use of experimental research proved elusive and industry use of software reliability prediction modeling is almost non-existent. However, in 2018 with the publication of Dr Judea Pearl's book "The Book of Why", a complete new scientific approach to gaining knowledge of cause-effect without controlled experimentation became practical. Indeed, the ability to take observational data to create causal graphs and then quantify direct, indirect, mediated and moderated causal effects represents a major paradigm shift in research and specifically, reliability modeling of software and humans, as well as in risk analysis and prognostics and health management. Causal learning goes beyond traditional correlation to distinguish spurious correlation versus cause and effect. While traditional statistical regression and most forms of machine learning depend on correlation and association, causal learning will enable intelligence approaching what is termed "General AI".

This tutorial will summarize the landscape of causal learning in more detail and specifically expose participants to open source tooling that performs causal search. Participants will be able to download software and files to follow along in performing a causal search on a provided data set. Participants will gain a personal experience in causal search and realize why causal search and causal graphs should always proceed traditional regression modeling by determining which factors should be included versus not included in the regression model. We desire that participants become educated consumers of causal learning and prompt wider use of causal learning in various fields of reliability analysis and modeling.

Robert W. Stoddard

Robert Stoddard is a Principal Researcher with the Software Engineering Institute, a federally-funded research and development center (FFRDC) managed by Carnegie Mellon University in Pittsburgh, PPA. Robert's primary research interests cover AI methods and causal learning, along with Bayesian probabilistic modeling and simulation of complex, software intensive systems, and AI-enabled metamorphic testing at-scale. Robert has proposed, won and led over $10M in DoD SEI research, as well as, secured and executed over $6M in SEI customer funded transition work. Before joining the SEI in 2005, Robert served for 25 years in distinguished managerial and technical roles within Texas Instruments and Motorola as a Director of Quality and Reliability, and Motorola Six Sigma Master Black Belt. Robert is active as a Fellow of the American Society for Quality and Senior Member of the IEEE. Robert is currently serving a 3-year, elected position within the IEEE Reliability Adcom committee with a focus on software reliability, AI, and innovation.

Table of contents

1. CHALLENGE OF PREDICTING SOFTWARE RELIABILITY

This tutorial briefly discusses the historical themes of software reliability research and suggests the adoption challenge remains rooted in a lack of a scientific foundation of cause and effect. This is reinforced with a reminder of the differences between research grounded in correlation studies only versus studies grounded in experimental design and/or causal modeling.

1.1 Historical Research in Software Reliability

Much of the early research into software reliability focused on modeling the results of software testing before release to the field. Software reliability engineers in the 1980's and 1990's were educated in such methodology through quite familiar texts such as by Musa (18, 19) and Lyu (14). During those two decades, software reliability practitioners waited till software test to begin analyzing and modeling data to help a product development group. As a consequence, the perceived value of software reliability engineering remained low due to how late feedback was provided in the software life cycle. Most projects had little time to react when facing incredible pressure to ship product on time.

Despite the primary focus on modeling software test results, pockets of excellence also pursued providing earlier feedback in the software life cycle. Notable efforts here involved the Rome Air Development Center (RADC) TR-83-176 "A Guidebook for Software Reliability Assessment" (6). For many of those practicing early lifecycle software reliability modeling and prediction, this became the only tool in the toolkit. However, the checklists and regressions equations in the guidebook modeled aerospace software and not other types of commercial and defense software. As such, users had to caveat the results and hoped to replicate the multivariable regression modeling using their own corporate data. At this time, corporate investment into this type of modeling seemed rare and only a handful of practitioners succeeded with a corporate version of the RADC guidebook model.

1.2 Barriers in Distinguishing Correlation from Causation

For many of us practitioners engaged in building our own corporate statistical models of software reliability during the early phases of software development, the identification of significant, independent predictive factors of software reliability seemed elusive. Many models began as experience-based hypotheses while others began with tribal knowledge within the organization. Whether the practitioners were trained in reliability modeling, statistical modeling or Six Sigma, almost all employed different forms of statistical linear regression. As will be shown later in this tutorial, the predominant use of statistical linear regression built on correlation resulted in many statistically-significant models that were not practically-significant. Beyond the raging debates surrounding the use and misuse of p values (Null Hypothesis Significance Testing NHST) (7), most of these models did not coach changes in early life cycle software factors that resulted in expected business improvement. These factors included a wide variety of process factors, product artifact factors, people factors, environmental factors, customer and supplier factors, marketing factors, manufacturing factors and supply chain factors. Only in recent years have I learned why an overwhelming majority of the statistically significant factors did not offer an ability to improve business results – my models were correlation-based and not causal-based. Only causal-based models should be used to coach interventions with an expectation of improving business results. My hindsight experience has been that about 80% of the statistically-significant factors represented spurious correlation and not cause-effect relationships.

1.3 Software DFR Needs Causal Based Models

During these years, a few efforts arose to define and implement Design for Reliability of computer-based systems. Notably, publications by Martin Shooman (24) and Kishor Trivedi (29) served to provide modeling and insight to factors driving/causing software reliability. These DFR approaches were mindful of the need for modeling that could drive improved performance and reliability based on controllable factors or design interventions. As will be evident later in this tutorial, causal models greatly complement and boost the intent of DFR for software engineering and computer-based systems.

2. THE NEW SCIENCE OF RELIABILITY – CAUSAL INFERENCE

"The New Science of Reliability" is not an original term or title. The title comes from the sub-title of Dr. Judea Pearl's latest book, "The Book of Why – The New Science of Cause and Effect"(22). This book represents a culmination of Dr. Pearl's life work ending with the pursuit of causality. Although considered one of the fathers of Bayesian networks, Dr. Pearl made his mark on history and won the Alan Turing award (11) based on his innovative contributions in the area of causal modeling.

In a nutshell, "The Book of Why" defines the "ladder of causality" consisting of three rungs: Rung 1 (Association) encompasses what most researchers practice by way of correlation studies and regression prediction models, Rung 2 (Intervention) encompasses the causal machinery to answer causal queries of intervention in a system, and Rung 3 (Counterfactuals) encompasses the causal machinery to answer counterfactual questions often a product of one's imagination. Dr. Pearl and his disciples believe that scientific advances should and will be based on more than Rung 1 of the ladder. I posit that dramatic advances in early life cycle software reliability prediction have yet to occur and will likely only occur through research practice of Rungs 2 & 3 of the causal ladder.

2.1 Recent Publications Advance Theory and Practice

A number of key research pioneers have shaped the evolving field of causal learning. Beyond Dr. Pearl's seminal publication on "Causality" (20) and his latest book, "The Book of Why", other authors have published books all aligned to this modern approach to causality. Although not comprehensive in the list, notables include Dr. Stephen Morgan (16, 17), Dr. Jonas Peters (23) and Dr. Miguel Hernan & Dr. James Robins (9). Other authors with causal specializations will be mentioned in later sections of this tutorial.

2.2 Methods Rooted Back to Beginnings of Experimental Design

Modern causal learning has roots reaching back to before the advent of Sir Ronald Fisher's statistical experimental design. Prior to his method of experimental design, matching methods were used in an attempt to quantify different conditions' influence on outcomes. As Fisher's experimental design methods, based on orthogonality and randomization, became quite popular, the traditional matching methods quietly evolved outside of the departments of statistics and computer science due to the political nature of the Fisher statistical movement. I first learned about modern causal methods from Carnegie Mellon University faculty located within the Department of Philosophy in the Dietrich School of Humanities.

2.3 How Correlation is Misinterpreted and Misused

You may be asking how correlation can be spurious and thus mislead and harm research results. Spurious correlation mostly seems to occur when there is a third, unmeasured confounder that affects both of the highly correlated factors. For example, one can show that ice cream sales are highly correlated with shark attacks, but the spurious correlation occurs because both factors are driven by a confounder, called temperature. Hot temperatures drive up ice cream sales and shark attacks. Although simple in this explanation, most researchers often miss the point that an unmeasured confounder could be explaining the spurious correlation that they are excitedly publishing. To reinforce this point, there is at least one website (https://tylervigen.com/spurious-correlations) which publishes a litany of spurious correlations along with the supporting data sets and publications.

Because most forms of regression are based on a measure of correlation, most forms of regression are also at risk of misuse and misinterpretation. In some cases, results of statistical linear regression without context or awareness of a causal model, can actually conclude the opposite of the truth in a given situation. An example of this wrongful conclusion from a regression model (28) is shown in an example used by the CMU Department of Philosophy to demonstrate the fallacy of regression without context of a causal model. As a consequence, the audience is encouraged to embrace a causal model before proceeding with a statistical regression model.

2.4 Evolution and Basics of Modern Causal Inference

As alluded to earlier, modern causal learning methods originated with Sewal Wright path modeling in the 1920's and continued evolving with structural equation modeling in the 1930's, social science path models in the 1960's, and Bayesian networks in the 1980's. Although outside of the scope of this tutorial and paper, the audience may be interested in further reading the social and political dynamics beginning in Fisher's time that led to the separate developments and publishing circles of traditional statistics and modern causal methodology. By the late 1990's, both the CMU researchers pursuing causal search methods and Dr. Pearl (http://bayes.cs.ucla.edu/jp_home.html) and UCLA associates (https://causalai.net/) pursuing causal estimation methods, published a litany of papers, books and some automation tools.

This tutorial then discusses some of the basics surrounding causal structures and causal estimation. Key concepts include the use of Directed Acyclic Graphs (DAGS) with three key patterns used in causal estimation: indirect connections, common causes, and common effects (aka colliders). Key causal actions to quantify causal effects include identifying causal and non-causal paths in a causal structure (DAG) and blocking or adjusting non-causal paths so that only causation may be identified and quantified. These specific concepts help to distinguish factors that should participate versus not participate in the attempt of statistical regression prediction modeling.

A final concept of the Markov Blanket is discussed to show the audience how a simple subset of factors can be identified to fully explain the causal inputs and outputs of a given outcome node.

2.5 The Causal Learning Landscape

The causal learning landscape remains difficult to discern from simple google searches. After reading many of the books shared earlier in this tutorial, it became apparent that different researchers were attacking different parts of the same elephant. The tutorial then explains a landscape of causal learning that begins with data and prior knowledge, if any, of a given outcome and related causal factors. One can then proceed in several directions. First, one can make use of causal search methodology and tools to discover the causal structure representing the data generating behavior from a given set of data. Second, one could instead hypothesize a causal structure from experience and prior literature to be tested for applicability to the data generating process represented in the data. Arriving at a Directed Acyclic Graph (aka Causal Structure) either way, a host of causal estimation machinery may then be employed to quantify indirect and direct causal effects.

2.5.1 Causal Search

Researchers within the CMU Department of Philosophy have led the charge on publication of causal search algorithms (15, 26). One of the earliest and most widely known search algorithm, the PC algorithm (25), was named after the two collaborating CMU researchers, Peter Spirtes and Clark Glymour. As will be seen, the Tetrad tool, among other tools, helps to automate the search algorithms when provided an input data set and any prior knowledge.

The causal tool, called Tetrad, may be accessed as open source via https://www.ccd.pitt.edu/, a center operated jointly by CMU and the University of Pittsburgh. Seemingly each year, doctoral students publish and automate several new search algorithms available by one or more of the different forms of the Tetrad tool.

The advantages and peculiarities of causal search algorithms still tend to be a best kept secret. Many skeptics see the algorithms as imperfect in nature and hence, the need to use a collection of search algorithms to support a given search journey seems to cause concern. Causal search algorithms do not produce absolute, singular answers or results of a causal structure. Often, minor tweaks in search algorithm parameters and prior knowledge can dramatically alter the resulting causal structure. Because of this, many view causal search as a mix of science and art. Every year, more of the art is converted to science. In the past several years, bootstrapping has been added to Tetrad to gain confidence levels into the causal edges. Tribal knowledge of combining constraint-based search algorithms with score-based search algorithms produces increased information from the resulting causal graphs.

One noteworthy aspect of causal search algorithms is the CMU shared story of how the traditional sense of sample size does not affect the search algorithms. In the example, a data set of only 47 observations in which each observation had a captured value of each of 21,326 genes was subjected to a causal search algorithm. The causal search algorithm identified 9 of the genes to be the candidates for regulating the flowering of a certain plant. Subsequent greenhouse studies showed that 4 of the 9 genes were true regulators of the plant flowering.

2.5.2 Causal Estimation

Separate from causal search, a tremendous amount of research and results exist within causal estimation, aka the quantification of the causal effects of one variable on an outcome. In this tutorial, we briefly name and summarize the key concept of each method to give the audience an appreciation of this landscape. Please remember that each of these topics require 3-10 days of intense training (https://statisticalhorizons.com/) to reach some degree of practical mastery. As with any tool or method, one can learn the basics and nominal case and then iterate studies to learn the more sophisticated aspects of the method.

Before jumping into each estimation method, one should recognize that causal estimation comes with new measures. For example, causal estimation distinguishes from direct and indirect causal influences. Additionally, causal estimation models can include both causal mediation and causal moderation, and yes, you can have mediated moderation as well as moderated mediation! Instead of a dominant focus on r and r-squared, estimation methods focus on Total Causal Effect (TCE), Individual Level Causal Effect (ICE), and Average Causal Effect (ACE).

2.5.2.1 Structural Equation Modeling

For those that are not familiar with structural equation models (10), you can think of them as a group of simultaneous regression models. Within this simultaneous group, some dependent factors can also serve as independent factors. Structural equation models are most often viewed graphically with nodes and edges. Structural equation models can have formative and reflective structures inside the model. Formative structures are likely what most people are used to. A small collection of factors are used to predict an outcome node, and hence have arrows pointing to the outcome node. On the other hand, reflective SEM models include latent factor structures. With a latent factor structure, you have an unmeasured latent factor that represents underlying truth. Then you have other measurable factors (often called manifest factors) that are representing the observable nature of the truth. In this structure, instead of the measurable factors having arrows pointing to the outcome, you instead have arrows pointing from the latent factor of truth to the measurable factors representing what is seen based on the truth. Reflective SEM models are some of the most popular SEM models in use because they further combat measurement error of the individual measurable factors associated with the truth factor. One last mention of advantage of SEM over traditional regression modeling is that traditional regression modeling has one error term and assumes all factors have zero measurement error. In SEM modeling, each factor has an associated measurement error term and thus is deemed more realistic to the world we wish to model.

2.5.2.2 Propensity Scoring

Propensity scoring is a modern and sometimes controversial technique for causal estimation. An excellent book by Shenyang Guo (8) provides not only a thorough treatment of the technique but offers a host of job aids, decision flow charts and Stata routines to automate most of the steps of propensity scoring. Without going into exquisite detail, one can think of a propensity score as a calculated probability that an observation (record of data) would be at a given level of an independent variable in the data set. Using the propensity score, the data records are matched 1 from the treatment and 1 from the control based on similar propensity scores. After matching, the leftover unmatched data records are disregarded. Final causal analysis occurs on the remaining data set.

2.5.2.3 Instrumental Variables

The instrumental variables approach (22) is quite helpful in certain situations to derive what otherwise would seem to be a non-identifiable causal situation. With this approach, a third factor Z is added to the data set. Z must be highly correlated with the independent factor X but have no causal relationship with the outcome factor Y. Even in the face of unmeasured confounders U of the X and U, one can determine the unconfounded causal effect of X on Y. The downfall of this method remains finding a good candidate factor for use as the instrumental variable. This is often difficult to impossible to find.

2.5.2.4 Front Door Adjustment

At a high concept level, the Front Door adjustment technique (22) is useful when you have a mediation of causal effect of Smoking on Cancer through the Tar factor. In context of the mediated causal link of Smoking to Tar to Cancer, you also have an unmeasured confounding effect of a Smoking Gene on both Smoking and on Cancer. The Front Door Adjustment, using causal algebra rules, can still distinguish the unconfounded causal effect of Smoking on Cancer.

2.5.2.5 Back Door Adjustment

In the Back Door Adjustment technique (22), all noncausal paths between X and Y are blocked. This stops the confounding of spurious correlation and enables the estimation of causal effects of X on Y.

2.5.2.6 Do-Calculus Causal Algebra

Do-Calculus (21) causal algebra was originally envisioned by Dr. Pearl and then subsequently proven mathematically by one of his disciples, Dr. Elias Bareinboim (Columbia University). Both Dr. Pearl and Dr. Bareinboim have published countless papers on the uses of Do-Calculus. Essentially, Do-Calculus represents a new causal algebra in which an intervention (Do(x)) may be modeled and solved. There are three simple rules behind the new causal algebra which can then be used along with normal rules of algebra to handle causal queries of any single factor causal intervention or change influencing an outcome.

2.5.2.7 Sigma-Calculus Causal Algebra

Published by Dr. Bareinboim this year (3), Sigma-Calculus represents a proven extension of Do-Calculus that can handle causal queries involving several simultaneous factor interventions or changes.

2.6 Practical Toolkit for Causal Learning

The author of this tutorial has immersed in publicly available professional training in the above methods for the past 7 years. The above methods can be handled with a portfolio of only three tools: Tetrad, Mplus and Stata. Each tool is briefly described in the tutorial.

2.6.1 Tetrad

As mentioned earlier, Tetrad (https://www.ccd.pitt.edu/) is an open source tool originally created by the CMU Department of Philosophy and currently maintained in collaboration with the University of Pittsburgh via the Center for Causal Discovery. This tool is on GitHub and has an active user community and roughly 3-4 new releases each year. The tool is available as a service on the web utilizing a super computer in the background, as well as a desktop GUI application and a Java environment programmable environment.

2.6.2 Mplus

Mplus (https://www.statmodel.com/) is discussed in the tutorial as one of many SEM tools available in the market. Mplus is a commercial tool but there are open source and free SEM tools. However, the author has found the Mplus tool to cover much more realistic data situations and refined modern SEM techniques as compared to the other tools. Mplus is a covariance-based SEM tool. There are other tools which implement Partial Least Squares SEM, notably a tool from Germany called PLS.

2.6.3 Stata

Stata (https://www.stata.com/) rounds out the author's causal learning toolkit because of the unique library of Propensity Scoring routines created by Dr. Shenyang Guo and his students which are not available on any other statistical platform.

3. USE CASES OF THE NEW SCIENCE

In this part of the tutorial, suggestions for both generic and reliability-specific uses of causal learning are discussed. As the author perceives causal learning to be almost 100% new to the reliability community, this part of the tutorial seeks to enlighten and motivate the community towards accelerated adoption of causal learning. In the next section on Demonstrations of Causal Inference, several recently completed causal studies by the author and his fellow researchers are shared. As will be seen, application of causal learning in the reliability field is just beginning and we have a rich target environment with software reliability.

3.1 Generic Use Cases

A number of generic research use cases exist for causal learning techniques. In this tutorial, we will discuss a handful that stand out in the author's mind.

a) Reproduce and confirm/clarify prior research

b) Create actionable models of performance

c) Root cause analyze problems using Big Data

d) Reduce time for experimentation through causal search of historical data

e) Use causal structures to better understand and test system behavior

f) Use causal structures to explain and assure AI solutions

3.2 Multivariate Causal Models of Software Reliability

Similar to the initial work by RADC in the mid 1980's, one could revisit a host of correlational studies within software engineering and software reliability and help distinguish spurious correlation from real cause-effect relationships. One could also propose new collaborative data collection strategies as part of the new agile, continuous development continuous integration (CD/CI) software lifecycle and conduct causal learning studies on those modern factors. The author would find it refreshing to begin reading software reliability causal research results on an annual basis leading to a more credible, discipline of software reliability engineering.

3.3 Alternative to Reliability Experiments

Within the reliability and software reliability domains, it continues to be expensive, if not prohibitive, to conduct traditional statistically-designed experiments. Because software reliability is driven by software engineering, a human intellectual dominant activity, traditional experimentation may be quite infeasible, illegal or impossible. Causal learning is not dependent on traditional statistically designed experimentation and as such, is free of many of the barriers to experimentation.

3.4 Digest Observational Data

The ability to digest purely observational data not necessarily cultivated by experimental design remains the number one strength of causal learning. This opens the door to causal modeling for software reliability within corporations and across the industry. The author hopes this incentive will increase data sharing, or at least sharing of causal results for software reliability.

3.5 Recover Poor Experimental Design and/or Execution

This tutorial will quickly discuss recent publications (1) showing how causal learning may be used to recover from a bad experimental design or from a poorly-executed experiment or experiment with loss of controls.

3.6 Account for Sample Data Bias

Causal learning has also recently been shown to provide a capable approach to identify and deal with sample data bias (2). Data bias is an often neglected topic in research and reliability modeling and needs to be properly dealt with for confident reliability modeling results. An interesting example (https://www.trevorbragdon.com/blog/when-data-gives-the-wrong-solution) dating back to WW II will be shared to demonstrate data bias.

3.7 Identify and Deal with Simpson's Paradox

Causal learning has also recently been shown to help identify and deal with Simpson's paradox (https://en.wikipedia.org/wiki/Simpson%27s_paradox). The tutorial will include an example of Simpson's paradox so that the audience will be sensitive to that in their future modeling.

3.8 Determine Robustness of Reliability Models and AI Solutions

As AI and machine learning have evolved and entered the marketplace, many skeptics ask whether the solutions are biased, ethical or trustworthy. Recent publications (1) demonstrate leading thoughts and approaches to use of causal learning frameworks to identify when AI solutions are not robust or trustworthy, and need retraining. This need will become even more paramount as AI solutions enter the reliability modeling space.

3.9 Conduct Reliability Exploratory Models

The combined use of causal learning and SEM offers much in the area of exploratory modeling for reliability. Specifically, exploratory SEM seeks to identify latent factors and relationships between latent factors in a model. Causal search can be employed in parallel to the exploratory SEM to compare and contrast with the SEM result. Causal learning might be of help in screening the factors to participate in an exploratory SEM.

3.10 Conduct Reliability Confirmatory Models

In a similar fashion, causal learning can assist in determining which factors should participate in a confirmatory SEM which seeks to confirm the measurable factors associated with each latent factor. Of note would be the agreement or lack of agreement between causal search and the SEM result.

3.11 Support Feature Engineering including Latent Factor Modeling

Similar to the Exploratory SEM use case, causal search can help scan large volumes of measurable factors for a given outcome and distill down the set most likely to produce a latent factor based on causality.

4. DEMONSTRATIONS OF CAUSAL INFERENCE

This section of the tutorial will discuss a number of recent case study results applying causal search. Some are related to software reliability and others are outside software reliability but inside the software engineering domain. At this point, examples of causal search and estimation within the software domain and the reliability domain are indeed scarce. We share the following as further incentive to collaborate on future software reliability studies.

4.1 Early Software Causal Studies

The tutorial will summarize early software causal studies (13) related to: controlling software size, controlling software complexity, controlling software architecture violations, and controlling software team performance. These particular causal studies transcend the DoD, commercial and academic software projects.

4.2 Architecture Pattern Violations Causing Vulnerabilities

This particular case study (12, 27) came from a research study into architecture pattern violations and the subsequent rate of experienced security vulnerabilities. The study focused on the opensource Chrome project and it's subsystems. The audience will see the comparison of the traditional correlation results versus what causal search came up with.

4.3 Audience Practice Demonstration of Causal Search

At this point in the tutorial and given there is sufficient time, the tutorial participants will be invited to conduct a short causal search exercise on a provided data set. Links to both the Tetrad tool jar file and the data set will be provided to participants at the beginning of the tutorial so they can download and ensure the Tetrad tool is launching correctly. During this interactive exercise, the tutorial presenter will go live with the Tetrad tool and ask interested participants to follow along with the exercise on their computer. Separately, a handout of the screen shots of the step-by-step process will be provided so that audience members may conduct the exercise on their own after the tutorial session.

5. CALL TO ACTION

This final section of the tutorial reminds the audience of several key points:

a) Dr. Pearl has researched all the major statistical breakthroughs in history and concluded that, on average, it takes 50 years for a breakthrough to become widespread in usage. His desire is that it not take 50 years for causal learning to become widespread in research and industry.
b) As a reliability professional, we must all become educated consumers of the results of causal learning technology; we must ask for the causal basis of research and modeling results; we must demand more than a correlation study.
c) As a reliability practitioner who leads by example, we must all learn more about causal learning; peruse or read some of the latest books; access a growing content of tutorials and education on YouTube; experiment with the techniques in your own reliability work.
d) Question all models for underlying causal structure; question robustness of models; question data bias; question mediated and moderated causal effects; question the possible existence of unmeasured confounders.
e) Seek out expert coaching and services; almost all of these book authors coach and provide services; they work with both students and industry professionals.

6. REFERENCES

1. Bareinboim, E., & Pearl, J. (2013). A general algorithm for deciding transportability of experimental results. arXiv preprint arXiv:1312.7485.
2. Bareinboim, E., Tian, J., Pearl, J. (2014). Recovering from Selection Bias in Causal and Statistical Inference. AAAI-14. In Proceedings of the 28th AAAI Conference on Artificial Intelligence.
3. Correa, J. D., & Bareinboim, E. (2020). A Calculus for Stochastic Interventions: Causal Effect Identification and Surrogate Experiments. In AAAI (pp. 10093-10100).
4. Correa, J., Tian, J., Bareinboim, E. (2019). Identification of Causal Effects in the Presence of Selection Bias. AAAI-19. In Proceedings of the 33rd AAAI Conference on Artificial Intelligence.
5. Dohi, Tadashi, Trivedi, Kishor, Avritzer, Alberto. (2020). Handbook of Software Aging and Rejuvenation: Fundamentals, Methods, Applications, and Future Directions. WSPC, New Jersey.
6. Goel, A. L. (1983). A guidebook for software reliability assessment' Rep. RADC-TR-83-176, Syracuse University.
7. Goodman, S. N. (1999). Toward evidence-based medical statistics. 1: The P value fallacy. Annals of internal medicine, 130(12), 995-1004.
8. Guo, S., & Fraser, M. W. (2014). Propensity score analysis: Statistical methods and applications (Vol. 11). SAGE publications.
9. Hernán, M. A., & Robins, J. M. (2020). Causal inference: what if. Boca Raton: Chapman & Hill/CRC, 2020.
10. Hoyle, R. H. (Ed.). (2012). Handbook of structural equation modeling. Guilford press.
11. Hyman, P. (2012). in honor of alan turing. Communications of the ACM, 55(9), 20-23.
12. Kazman, R., Stoddard, R., Danks, D., & Cai, Y. (2017, May). Causal Modeling, Discovery, & Inference for Software Engineering. In 2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C) (pp. 172-174). IEEE.
13. Konrad, M., Stoddard, R., Nichols, W., Zubrow, D. (2109). Causal Models for Software Cost Prediction & Control (SCOPE). Software Engineering Institute Research Review.
14. Lyu, M. R. (1996). Handbook of software reliability engineering (Vol. 222). CA: IEEE computer society press.
15. Malinsky, D., & Danks, D. (2018). Causal discovery algorithms: A practical guide. Philosophy Compass, 13(1), e12470.
16. Morgan, S. L., & Winship, C. (2015). Counterfactuals and causal inference. Cambridge University Press.
17. Morgan, S. L. (Ed.). (2013). Handbook of causal analysis for social research. New York, NY: Springer.
18. Musa, J. D., Iannino, A., & Okumoto, K. (1987). Software reliability: Measurement, prediction, application. McGrawHill, New York.
19. Musa, J. D. (2004). Software reliability engineering: more reliable software, faster and cheaper. Tata McGraw-Hill Education.
20. Pearl, J. (2009). Causality. Cambridge university press.
21. Pearl, J. (2012). The do-calculus revisited. arXiv preprint arXiv:1210.4852.
22. Pearl, J., & Mackenzie, D. (2018). The book of why: the new science of cause and effect. Basic Books.

23. Peters, J., Janzing, D., & Schölkopf, B. (2017). Elements of causal inference (p. 288). The MIT Press.
24. Shooman, M. L. (2002). Reliability of computer systems and networks. John Wiley & Sons, Incorporated.
25. Spirtes, P., Glymour, C. An algorithm for fast recovery of sparse causal graphs. Social Science Computer Review, 9(1):67–72, 1991.
26. Spirtes, P., Glymour, C. N., Scheines, R., & Heckerman, D. (2000). Causation, prediction, and search. MIT press.
27. Stoddard, R., Nichols, B., Konrad, M., & Kazman, R. (2019). Do Software Architecture Patterns Reduce Security Vulnerabilities Insight from Causal Learning. Carnegie Mellon University Software Engineering Institute Pittsburgh United States.
28. Timberlake, M. and Williams, K. (1984). Dependence, political exclusion, and government repression: Some cross-national evidence. American Sociological Review 49, 141-146.
29. Trivedi, K. S., & Bobbio, A. (2017). Reliability and availability engineering: modeling, analysis, and applications. Cambridge University Press.