Software Engineering Institute
Carnegie Mellon University

Investigating Causal Effects of Software and Systems Engineering Effort

Jim Alstad, Anandi Hira, A Winsor Brown; Mike Konrad
Alstad@acm.org, A.Hira@usc.edu, AWBrown@usc.edu; MDK@sei.cmu.edu
University of Southern California; Software Engineering Institute

ICEAA 2021 Online Workshop
May 17-20, 2021

USC CSSE

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.

DM21-0445

Presentation Outline

- Motivation
- Causal Analysis in SW Cost

- Intro to Causal Inference
- Algorithms and Tool Used

- Datasets – COCOMO® II, COSYSMO 3.0

- Approaches and Results

- Conclusions and Questions

Motivation

- Managers are frequently faced with issues of controlling project costs
  - My estimated cost is too high. What project aspects can I modify that would most likely reduce the cost?
  - I have some money to improve my organization's performance. Changing which organizational aspects would be most likely to improve cost performance?
  - As an acquirer, I need to add a new stakeholder and remove flexibility in modifying requirements. Is that likely to have a significant influence on project cost?
- Causal Analysis is a modern technique that analyzes datasets to determine causal relationships among its variables
- **The Goal of this Research:** Identify factors of software and systems engineering costs that are direct causes
  - To help manage real projects

History of Causal Analyses for Effort

Boehm – COCOMO® Models [1]

- In-depth behavioral analyses for effort drivers
- Including COSYSMO models

Evidence-Based SW Engineering [2]

- Suggests running Experiments to identify causal relationships:
  - Cause precedes effect
  - Cause covaries with effect
  - Alternative explanations are implausible

Unified Code Count Maintenance [3]

- Software maintenance and upgrade data
- Project data has limited scope
  - Similar projects, from a single environment

Our difference: 2 calibration datasets (observational data) with varying values of cost drivers, application types, and project types

Presentation Outline

- Motivation
- Causal Analysis in SW Cost

- Intro to Causal Inference
- Algorithms and Tool Used

- Datasets – COCOMO® II, COSYSMO 3.0

- Approaches and Results

- Conclusions

We Employ Causal Inference as the Basis of Our Research

Causal Inference

- **Causal Search/Discovery** — Algorithms and domain knowledge on observational data
- **Causal Estimation** — Algorithms to quantify causal influence; structural equation modeling (SEM)

Causal Search Algorithm Results

Result is a "causal graph", with each box representing a variable and each edge representing a causal relationship.

Here are the different possible types of edge:

X_1 directly causes X_2

X_1 directly causes X_2 or X_2 directly causes X_1

No directly causal relationship between X_1 and X_2

Tetrad Tool

- Implements causal algorithms
- Implemented and maintained by Center for Causal Discovery, primarily run by Carnegie Mellon University (CMU) and University of Pittsburgh
- https://www.ccd.pitt.edu for information, tutorials and tools
- We acknowledge the Center for Causal Discovery, supported by grant U54HG008540, in maintaining the algorithms and Tetrad tool used in this research.

* https://github.com/cmu-phil/tetrad for Java source code, latest release (as a Jar file), and the Tetrad Manual

* Our results come from Tetrad versions 6.5.4 (earlier results) and 6.7.0 (recent results)

Causal Search Algorithms [3]

PC Search

- Constraint-based algorithm
- PC's strengths are: (1) independence is an intuitive concept; (2) PC is modular, allowing different tests to be employed, to match assumptions about data distributions.

FGES Search

- Score-based algorithm
- FGES's strengths are: (1) resulting graph has almost all of its edges oriented; (2) score is similar to that used for model estimation.

Bootstrapping

- One bootstrap: draw a random sample (typically of size 90%) from the original dataset, with replacement
  - Run search algorithm on this sample
- Repeat this 100 or more times, and aggregate the results for all detected edges into the **edge probability table (EPT)**
  - In the EPT, each edge will show the percentage of times it was found, which reflects the fraction of data points that has this direct-causal relationship
- Stronger causal relationships appear higher in the EPT.
  - Entries further down the EPT are more likely to be due to accidental correlations
- Reduces sensitivity to small changes in dataset, improving generalizability

Bootstrap Results for COSYSMO 3.0

Graph Edges:

1. LogSize --> LogEffort [LogEffort <-- LogSize]:1.0000;
2. LSVC --> LogEffort [LSVC --> LogEffort]:0.6300;
   [no edge]:0.3700;
3. LogSize --> SCHED [LogSize <-- SCHED]:0.2300;
   [LogSize --- SCHED]:0.0800;
   [LogSize --> SCHED]:0.4900;[no edge]:0.2000;
4. DOCU --> LogEffort [DOCU --> LogEffort]:0.3800;
   [no edge]:0.6200;
5. TEAM --> ROPM [ROPM <-- TEAM]:0.3800;
   [ROPM --- TEAM]:0.1000;
   [ROPM --> TEAM]:0.2800; [no edge]:0.2400;

.

.

.

152. TRSK --> INST [INST <-- TRSK]:0.0100; [no edge]:0.9900;

Causal Estimation

- **Causal estimation** involves parameterizing the relationships appearing in the causal search graph and then determining what values to assign to these parameters.
  - Enables making predictions about the values that variables will attain as a result of hypothesized events; i.e., allows making an estimating model.
  - Causal estimation, when applied to just a single variable and its direct causes works like ordinary linear regression: **coefficients** are assigned to each edge
  - A *one-unit* change in a direct cause, with all other variables held constant, results in a change in the child of *coefficient* units.
- The resulting model is then evaluated for **model fit**.
  - **Model fit statistics** include: Chi square (per degrees of freedom), Bayesian Information Criterion (BIC), Comparative Fit Index (CFI), and Root Mean Square Error of Approximation (RMSEA).
- More information can be found in [4, 5]

Presentation Outline

- Motivation
- Causal Analysis in SW Cost

- Intro to Causal Inference
- Algorithms and Tool Used

- Datasets – COCOMO® II, COSYSMO 3.0

- Approaches and Results

- Conclusions

Datasets

COCOMO® II Calibration Dataset

- 16 organizations, various application types
- Variability in all 26 variables
- 161 projects
- See [6] for more details

COSYSMO 3.0 Calibration Dataset

- Covers various types of systems
  - > 2 orders of magnitude size variation
- Variability in all 18 variables
- 68 projects
- See [7] for more details

Each dataset is reasonably representative of projects of its type

Presentation Outline

- Motivation
- Causal Analysis in SW Cost

- Intro to Causal Inference
- Algorithms and Tool Used

- Datasets – COCOMO® II, COSYSMO 3.0

- Approaches and Results

- Conclusions

Our Methodology for Causal Search on Small Samples

- Problem: Search did not produce structures that were very informative
  - A consequence of having relatively few data points (projects)
- Solution: We invented an approach called weak-signal analysis (WSA), which consists of these steps, some based on the PNE (probability of no edge):
  1. Inject **null variables**: For each original variable, add a "null variable" column, a copy of the original variable values randomly sorted.
  2. Do causal search with **bootstrap**: determine for each edge terminating on a null variable (a "random edge") its PNE
  3. Set a **trim threshold** at the **10th percentile** of random edge PNEs (i.e., 90% of random edges will have a higher PNE)
  4. Discard all edges among original variables whose PNE > trim threshold. Also, discard all null variables and random edges.
- (The causal graph resulting from applying this to COSYSMO 3.0 data is shown on slide 19.)

We Found These Factors To Be Direct Causes of Effort

Intervening on These in a Project May Improve Outcomes

COCOMO® II - Effort

- Size (SLOC)
- Team Cohesion (TEAM)
- Platform Volatility (PVOL)
- Reliability (RELY)
- Storage Constraints (STOR)
- Time Constraints (TIME)
- Product Complexity (CPLX)
- Process Maturity (PMAT)
- Risk and Architecture Resolution (RESL)

COCOMO® II - Schedule

- Size (SLOC)
- Platform Experience (PLEX)
- Schedule Constraint (SCED)
- Effort (Log_PM)

COSYSMO 3.0 - Effort

- Size
- Level of Service Requirements (LSVC)

Using Tetrad to Derive Mini-Models to Produce Plausible Cost Estimates

Guided by the existing COCOMO® II and COSYSMO 3.0 estimating models' structure:

1. The structure of the estimating models does not directly conform to that needed by Tetrad. Therefore transformed the structure of each estimating equation by taking logarithms.
2. Forced cost predictors to be independent of each other (Tetrad Knowledge box).
3. Applied WSA to obtain a plausible causal graph, discarding any variables without edges.
4. Used the Tetrad Estimation capability to obtain coefficients and intercepts on the resulting graph. The mini-model was obtained by extracting the mini-estimating equation from the resulting graph.
5. However, intercepts need further work (below).

Resulting COSYSMO 3.0 Estimation

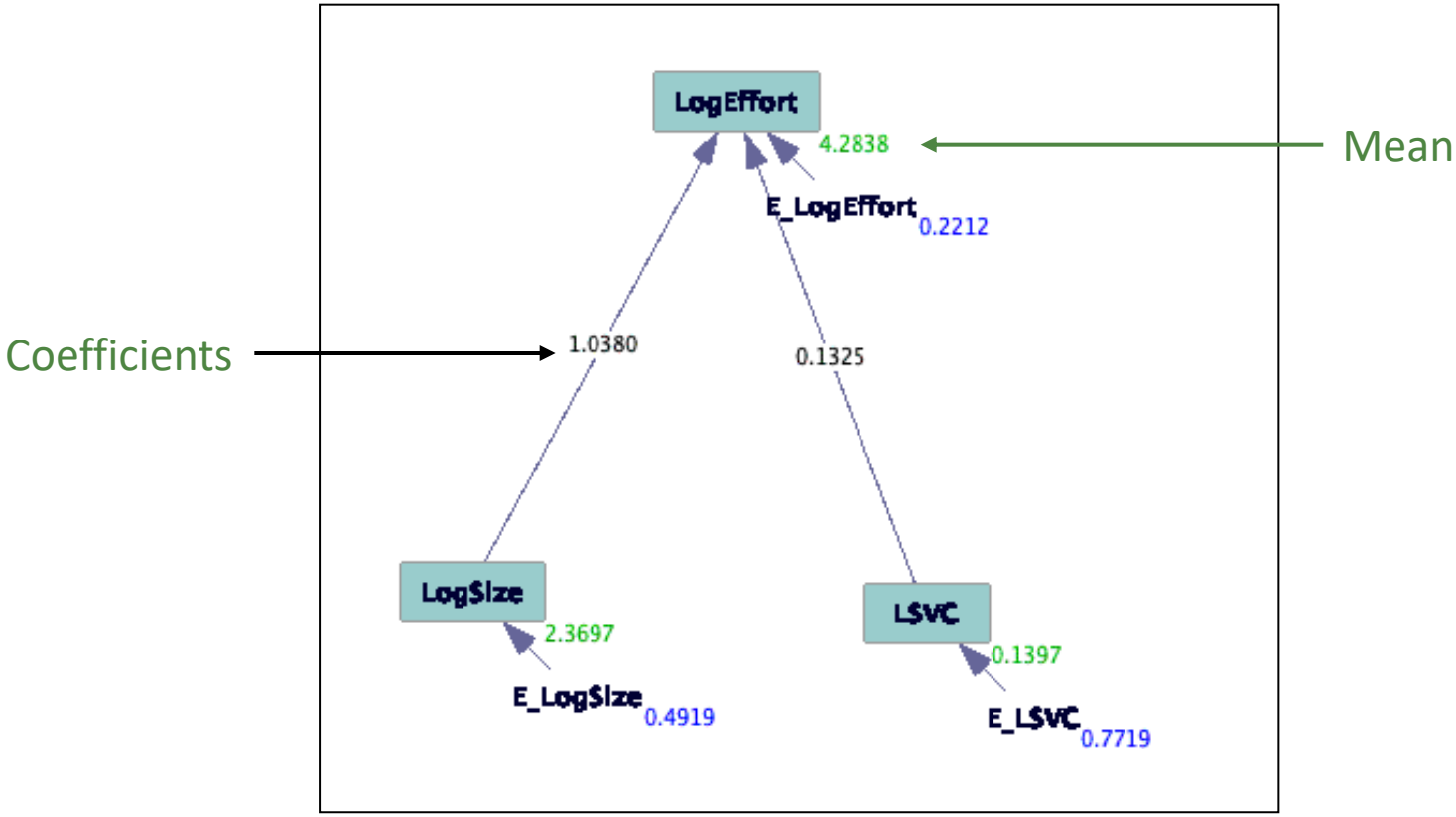
Overall Model Fit Statistics

Chi-Square Test(s) of Model Fit:
P-Value = 2.3667E-5
(want > 0.01, or at the very least > 0);
Chi-Square/DF = 17.8688 (want < 5)

RMSEA (Root Mean Square Error Of Approximation):
RMSEA = 0.5018 (want < 0.08)

CFI (Comparative Fit Index):
0.9967 (want > 0.95)

Conclusion: Model fit is Poor-to-Fair.



How to Get a Mini-Model from that Tetrad Estimation Model

- Reading off from the Tetrad estimation model, a mini-model would be:

  LogEffort = 1.0380 * LogSize + 0.1325 * LSVC + 4.2838

- First attempt at Effort estimation:

  $$\text{Effort} = \text{Size}^{1.0380} * 1.357^{\text{LSVC}} * 10^{4.2838}$$

- That, however, doesn't work
  - The problem is that 4.2838 is the mean of the LogEffort values; however, raising 10 to that power does not yield the mean of the Effort values.
- One has to do a separate linear regression of LogEffort against LogSize and LSVC
  - That yields an exponent for 10 of 1.805, which gives this estimating equation:

    $$\text{Effort} = 63.834 * \text{Size}^{1.0380} * 1.357^{\text{LSVC}}$$

Resulting COCOMO® II Estimation

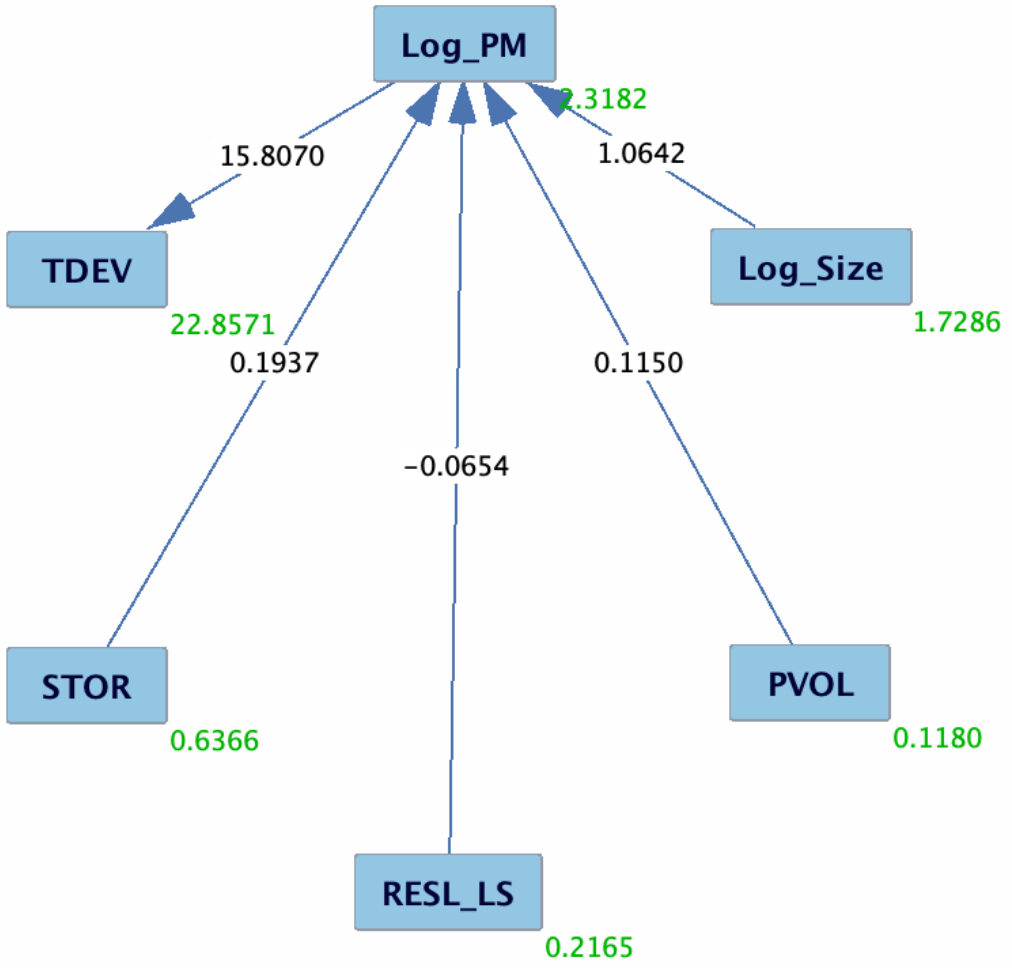
Overall Model Fit Statistics

Chi-Square Test(s) of Model Fit:
P-Value = 8.9571E-5
(want > 0.01, or at the very least > 0);
Chi-Square/DF = 3.58 (want < 5)

RMSEA (Root Mean Square Error Of Approximation):
RMSEA = 0.1271 (want < 0.08)

CFI (Comparative Fit Index):
0.9993 (want > 0.95)

Conclusion: Model fit is Fair-to-Good.



Prediction Accuracy: Mini-Models vs Estimating Models

COSYSMO 3.0 - Effort

	Mini-Model	Original
Max MRE	285.4%	234.8%
MMRE	45.9%	57.3%
PRED(25)	41.2%	23.5%
PRED(30)	48.5%	23.5%

COCOMO® II - Effort

	Mini-Model	Original
Max MRE	455.4%	229.41%
MMRE	38.64%	25.67%
PRED(25)	44.72%	67.08%
PRED(30)	52.8%	74.53%

COCOMO® II - Schedule

	Mini-Model*	Original
Max MRE	628.6%	130.95%
MMRE	42.28%	50.88%
PRED(25)	45.34%	9.94%
PRED(30)	52.8%	12.42%

*Note: Analysis done with TDEV; but realized Log(TDEV) might have been better.

Presentation Outline

- Motivation
- Causal Analysis in SW Cost

- Intro to Causal Inference
- Algorithms and Tool Used

- Datasets – COCOMO® II, COSYSMO 3.0

- Approaches and Results

- Conclusions

Conclusion – Causal Search

- Straightforward use of causal search algorithms may result in little information about cost-causing factors
  - Relatively small datasets (# of cases) compared to # of variables
- Weak-Signal Approach (WSA) enhanced results
  - Identified additional causes of effort and duration, while minimizing spurious correlations
  - Established a principled approach (methodology) to determining what cutoff to use for trimming results of a bootstrapped search (based on null variables and EPT)
- We identify (on slides 17, 19, and 21) specific direct causes, where action has been shown statistically to cause the cost or schedule
  - The data we used considered multiple application types and multiple organizations
  - We also investigated choice of Tetrad search algorithm and parameter values

Conclusion – Causal Estimation

- We developed a methodology (slide 16) for generating cost estimation mini-models based on datasets that deliver plausible results
- Observation
  - Modestly fitting with somewhat inferior predictions compared to original model
- Further Research
  - More investigation in alternative estimation approaches could produce more effective Tetrad-based models for use

How You Can Get Started with Causal Analysis

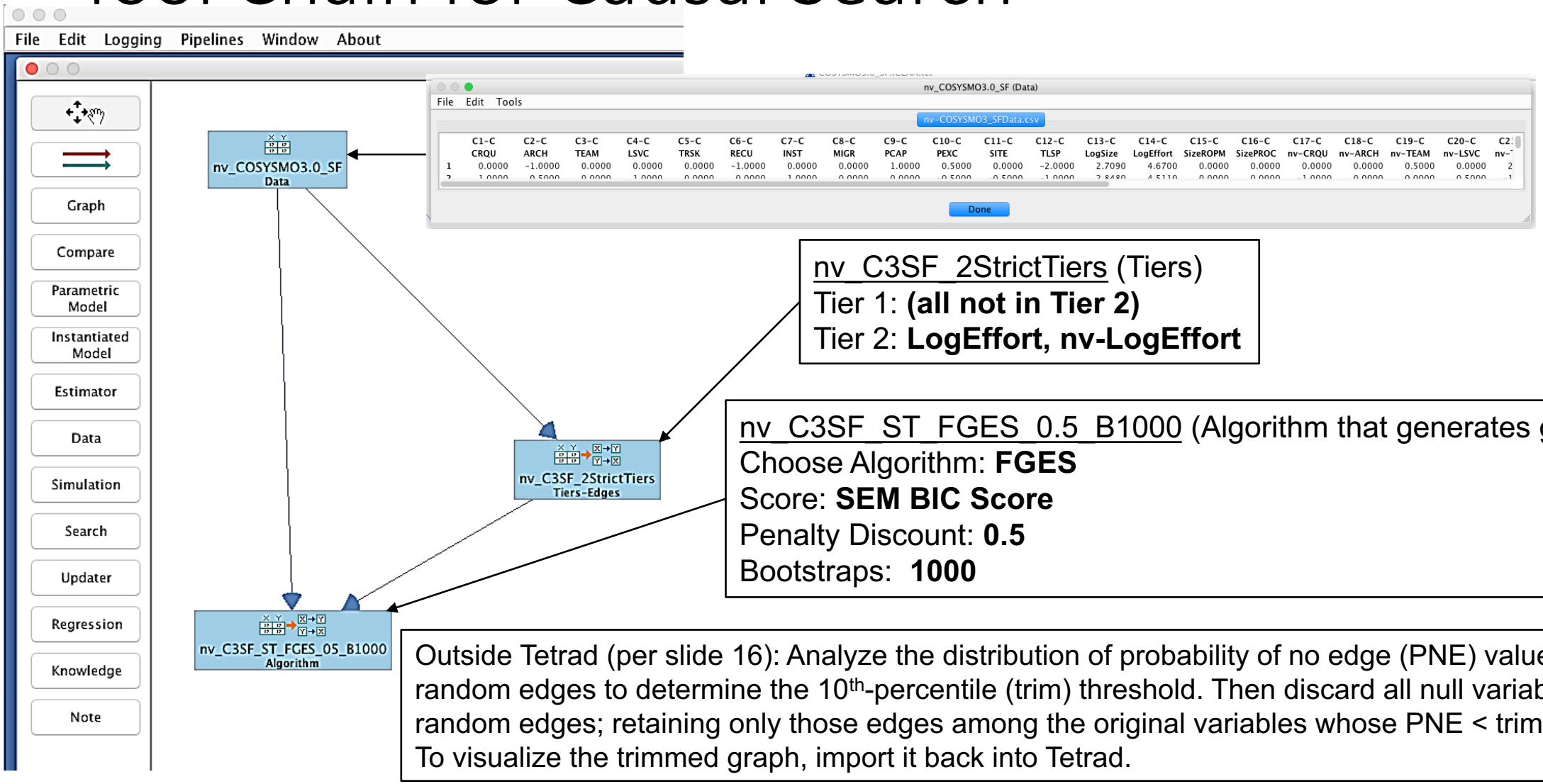
- We encourage you to try Tetrad, and WSA if warranted, on your data
  - Find out what the most important (i.e., causal) factors are
- Applicable to an organizational project database of moderate or larger size.
- Get Tetrad and the "Tetrad Manual" (see slide 9)
- Obtain training in Causal Search and Tetrad:
  - From those who pioneered Tetrad: https://www.ccd.pitt.edu/video-tutorials/
  - From the SEI: contact Mike Konrad (also the source for WSA Python scripts)

Bibliography

1. Boehm, B. W. (1981). *Software Engineering Economics.* Upper Saddle River, NJ: Prentice-Hall, Inc.
2. Kitchenham, B. A., Dyba, T., & Jorgensen, M. (2004, May). Evidence-based software engineering. In *Proceedings. 26th International Conference on Software Engineering* (pp. 273-281). IEEE.
3. Hira, A., Boehm, B., Stoddard, R., & Konrad, M. (2018). Further Causal Search Analyses With UCC's Effort Estimation Data.
4. Spirtes, P. (2010). Introduction to causal inference. *Journal of Machine Learning Research*. 11, May (2010), 1643–1662.
5. Kline, R. (2015). Principles and Practice of Structural Equation Modeling, Fourth Edition. Series: Methodology in the Social Sciences. The Guilford Press.
6. Boehm, B. W., et al. (2000). *Software Cost Estimation with COCOMO II.* Upper Saddle River, NJ: Prentice-Hall, Inc.
7. Alstad, JP. (2019). “Development of COSYSMO 3.0”, *Procedia Computer Science 153*

Backup Slides

Tool Chain for Causal Search

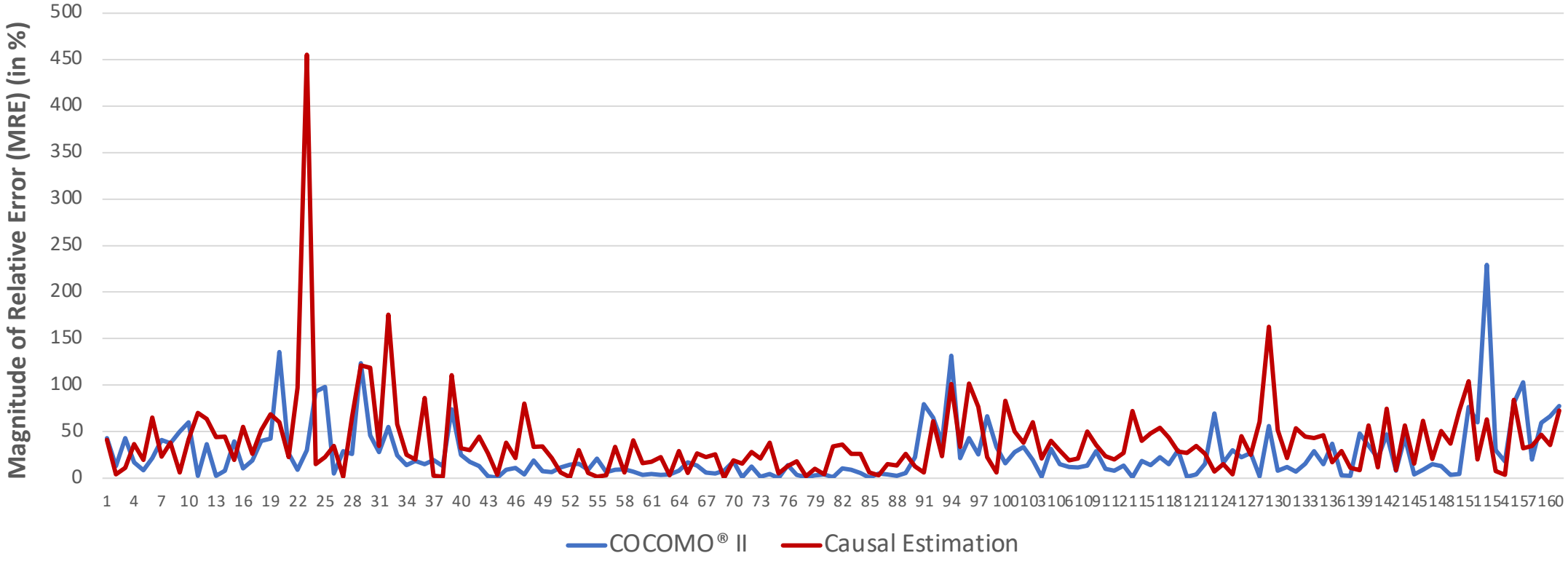


nv_C3SF_2StrictTiers (Tiers)
Tier 1: **(all not in Tier 2)**
Tier 2: **LogEffort, nv-LogEffort**

nv_C3SF_ST_FGES_0.5_B1000 (Algorithm that generates graph)
Choose Algorithm: **FGES**
Score: **SEM BIC Score**
Penalty Discount: **0.5**
Bootstraps: **1000**

Outside Tetrad (per slide 16): Analyze the distribution of probability of no edge (PNE) values for random edges to determine the 10th-percentile (trim) threshold. Then discard all null variables and random edges; retaining only those edges among the original variables whose PNE < trim threshold. To visualize the trimmed graph, import it back into Tetrad.

COCOMO® II vs Mini-Model Effort MRE Comparison



Using more factors for a cost estimate (as with the full model) tends to reduce the frequency of way-off predictions (of course, on any given project, either model might be more accurate). The advantage of the mini-model is that it uses just the factors, among many, that are more likely to drive cost and schedule.