Causal Models for Software Cost Control (SCOPE)

Recent results from five different studies

How can we better control costs in software development and sustainment? This project is collaborating with systems and software researchers in applying causal learning to program datasets to better understand which factors can reduce costs.

DoD Problem

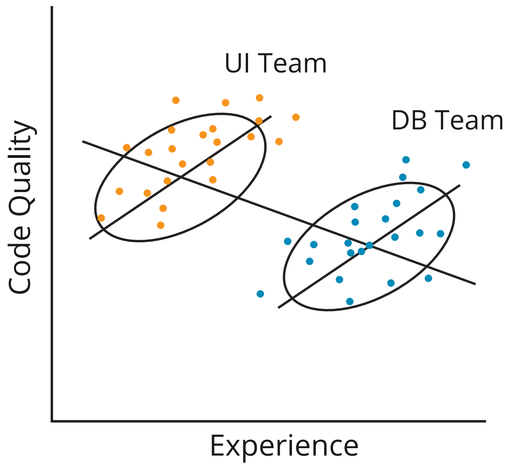
- DoD leadership continues to ask "Why does software cost so much?"
- DoD program offices need to know where to intervene to control software costs

Our Solution

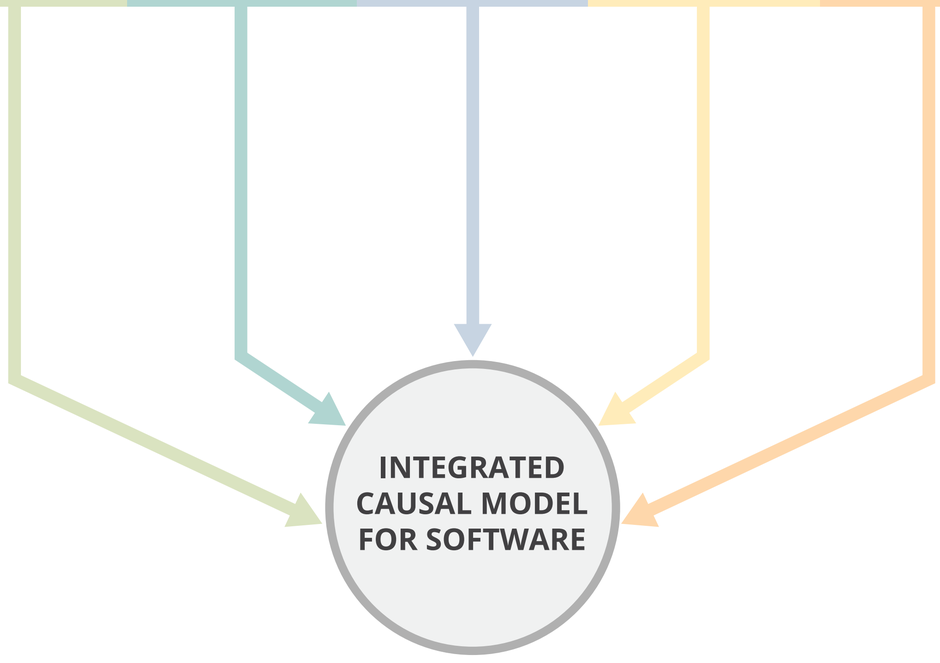
An actionable, full causal model of software cost factors immediately useful to DoD programs and contract negotiators

Causation Vs. Correlation

To reduce costs, what *causes* code quality to be good or bad needs to be understood. Correlations are insufficient. For example, in the figure below, would increasing experience level improve code quality?

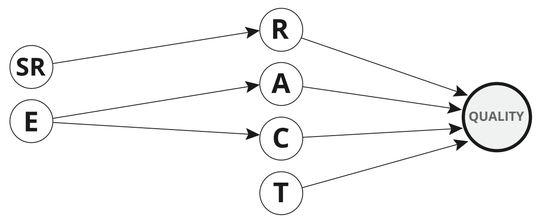


Practitioner	Software Size	Architecture	Complexity	Leadership
Challenge: Which factors affect a programmer's coding effort and quality?	**Challenge:** Which approaches to measuring code size most reflect factors affecting total effort?	**Challenge:** How might a project manager decide which areas of code to prioritize for maintenance?	**Challenge:** Which program and system complexity factors most affect cost, schedule, and performance?	**Challenge:** For action planning, which attributes of teaming and leadership improve team performance?
Approach: Apply Causal Discovery to data from students coding to the same ten requirements specifications.	**Approach:** Apply Causal Discovery to USC's Unified Code Count (UCC) project dataset.	**Approach:** Apply Causal Discovery to the results of a static code and design structure analysis to determine which type of architectural pattern violation most affects code quality.	**Approach:** Apply Causal Discovery to an existing project-survey dataset.	**Approach:** Apply Causal Discovery to results of 18 months of weekly surveys of software engineers from across a DoD organization to determine which factors most affect cost, schedule, and quality.
Results: To achieve precision, software estimation models should include both objective measures of requirements size as well as programmer-specific coding and defect factors.	**Results**: For IT-type systems, only COSMIC Function Points, Programmer Capability, and Documentation-Aligns-with-Lifecycle-Needs repeatedly recur as direct causes of total effort.	**Results:** Cyclic dependency was the single architecture pattern violation affecting code quality.	**Results:** The original analysis identified difficult requirements, stakeholder relationships, and cognitive fog; causal discovery confirmed only cognitive fog.	**Results:** Of the 20+ factors found to be highly correlated with cost, schedule, and quality, direct causal relationships were found for only two: Good Improvement Data and Stress From Overtime.



Technical Approach

Working with collaborators, we will identify and prepare datasets for causal learning to establish key cause-effect relationships among project factors and outcomes. For example, for Quality, we might have this causal graph:



The resulting causal models will then be "stitched" using CMU algorithms to create a universal causal model, but estimated and calibrated for lifecycle and super-domain. These estimated models will be the basis for improved program management.

Collaborative Approach

First, the SEI trains each collaborator to perform causal searches on their own proprietary datasets. The SEI then only needs to be provided with information about what dataset and search parameters were used as well as the resulting causal graph, which is sufficient for integrating into a universal causal model.

Summary

Causal learning has come of age from both a theoretical and tooling standpoint and provides better basis for program control than models based on correlation. Application to cost estimation requires large amounts of quality data. Now is the time to engage the larger community of systems and software researchers in deriving improved cost models that enable improved program control.

Dr. Michael Konrad | mdk@sei.cmu.edu
Robert Stoddard | rws@sei.cmu.edu

Distribution Statement A: Approved for Public Release; Distribution is Unlimited

Copyright 2018 Carnegie Mellon University. All Rights Reserved.
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.
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.
DM18-1145