CauseMap: Fast inference of causality from complex time series

M. Cyrus Maher, Ryan D. Hernandez

Background: Establishing health-related causal relationships is a central pursuit in biomedical research. Yet, the interdependent non-linearity of biological systems renders causal dynamics laborious and at times impractical to disentangle. This pursuit is further impeded by the dearth of time series that are sufficiently long to observe and understand recurrent patterns of flux. However, as data generation costs plummet and technologies like wearable devices democratize data collection, we anticipate a coming surge in the availability of biomedically-relevant time series data. Given the life-saving potential of these burgeoning resources, it is critical to invest in the development of open source software tools that are capable of drawing meaningful insight from vast amounts of time series data.

Results: Here we present CauseMap, the first open source implementation of convergent cross mapping (CCM), a method for establishing causality from long time series data (> ~25 observations). Compared to existing time series methods, CCM has the advantage of being model-free and robust to unmeasured confounding that could otherwise induce spurious associations. CCM builds on Takens' Theorem, a well-established result from dynamical systems theory that requires only mild assumptions. This theorem allows us to reconstruct high dimensional system dynamics using a time series of only a single variable. These reconstructions can be thought of as shadows of the true causal system. If the reconstructed shadows can predict points from the opposing time series, we can infer that the corresponding variables are providing views of the same causal system, and so are causally related. Unlike traditional metrics, this test can establish the directionality of causation, even in the presence of feedback loops. Furthermore, since CCM can extract causal relationships from times series of, e.g. a single individual, it may be a valuable tool to personalized medicine. We implement CCM in Julia, a high-performance programming language designed for facile technical computing. Our software package, CauseMap, is platform-independent and freely available as an official Julia package.

Conclusions: CauseMap is an efficient implementation of a state-of-the-art algorithm for detecting causality from time series data. We believe this tool will be a valuable resource for biomedical research and personalized medicine.

M. Cyrus Maher[1*], Ryan D. Hernandez[2,3,4]

[1]Department of Epidemiology and Biostatistics, University of California, San Francisco, 185 Berry Street, Lobby 5, Suite 5700 San Francisco, CA 94107
[2]Department of Bioengineering and Therapeutic Sciences,
[3]Institute for Human Genetics,
[4]Institute for Quantitative Biosciences (QB3),
University of California, San Francisco
San Francisco, California, USA

[*] To whom correspondence should be addressed.

E-mail: cyrusmaher@gmail.com

Address: 191 Castro St. Mountain View, CA 94041

Phone: (858) 249-7500

Introduction

Establishing health-related causal relationships is a pivotal objective in biomedical research. Yet, the interdependent non-linearity of biological systems often impedes a thorough understanding of causal dynamics. Existing and forthcoming time series data will likely play an important role in taming this complexity. Traditional cross-sectional sampling have the limitation that they may average out non-linear patterns by pooling heterogeneous signals across subjects. Long time series from a single source, on the other hand, can allow us to understand dynamic and context-specific patterns of change.

We are just beginning to grasp the biomedical relevance of such a dynamical systems perspective. Consider for example the human microbiome. Dysbiosis in the gut has been implicated in, e.g. irritable bowel disease (IBD), obesity, diabetes, asthma, anxiety, and depression (Foster & McVey Neufeld, 2013; Arrieta et al., 2014). Meanwhile, recent studies on microbiome dynamics have found that the ecological makeup of the human microbiome is dynamic and individual-specific . These dynamics may also interact with pathogens in interesting and therapeutically important ways. For example, there is evidence that ecological time series dynamics within the body may play a role in the progression from HIV to AIDS .

Complex, dynamically evolving interdependent systems such as the microbiome pose a significant challenge to existing time series methods. Several metrics exist for detecting static non-linear relationships. These include: spearman correlation , distance correlation (Spearman, 1904), and mutual information content . Causal relationships, on

the other hand, can be examined using methods such as time-lagged regression , instrumental variables , and dynamical Bayesian networks (Granger, 1969).

These causal methods are heavily model-based, however. As a result, they often falter when examining arbitrary non-linear or context-dependent relationships. Furthermore, the approaches mentioned above cannot adequately handle feedback loops, and they frequently generate both false positives and false negatives due to the influence of unmeasured confounders . These are significant liabilities, particularly in biomedicine, where relationships are usually embedded within a broad network of incompletely observed interactions.

In this paper, we present the first publicly available, open source implementation of convergent cross mapping (CCM), a model-free approach to detecting dependencies and inferring causality in complex non-linear systems (even in the presence of feedback loops and unmeasured confounding; Sugihara et al., 2012). CCM derives this power from explicitly capturing time-dependent dynamics through a technique known as state-space reconstruction (SSR). SSR has demonstrated utility for problems as diverse as wildlife management and cerebral autoregulation (Vanderweele & Arah, 2011). In practice, this analysis typically requires at least 25 data points, measured with relatively high accuracy and with sufficient density to capture system dynamics. One benefit of this approach is that, unlike most causal inference methods, the performance of CCM improves for increasingly non-linear systems. In addition, CCM can properly disentangle causal relationships that involve feedback loops, provided that strong forcing from external variables does not overwhelm the dynamics of the relationships of interest.

CCM leverages the fact that time series can be viewed as projections of higher-dimensional system dynamics (Sugihara *et al.*, 2012). As a logical result of this property, the time series of individual variables must contain information about the full causal system. Causal dynamics (conceptualized as the state space, or manifold) can then be reconstructed using individual time series. These reconstructions can be thought of as shadows of the true causal system. If the shadows reconstructed from distinct variables can be used to predict points from each other's time series, we can infer that these variables provide views of the same causal system and so are causally related. Since these relationships are fundamentally asymmetric, this test can also establish the directionality of causation.

Further details on CCM are available in the supplementary material of this paper, as well as in that of Sugihara *et al.* 2012. Additional explanatory resources can also be accessed through the project website (http://cyrusmaher.github.io/CauseMap.jl).

Materials and Methods

Convergent cross mapping algorithm

Consider time series of hypothetical variables X and Y. Convergent cross mapping (CCM) employs time-lagged coordinates of each of these variables to produce shadow versions of their respective source manifolds. To illustrate, suppose the time series for X were {1, 2, 3, 4}. Reconstructing a two-dimensional shadow manifold for X using a time lag of one would yield the following path: (2, 1) → (3, 2) → (4, 3). For sufficiently long time series, the path of this shadow manifold is expected to reveal important properties of the full causal system.

We will refer to the shadow manifolds reconstructed from X and Y as M_x and M_y, respectively. To test whether X causes Y, CCM applies the following logic: because manifold reconstruction preserves important structural components of the original system (i.e. the Lyapunov exponents; Casdagli, Eubank, Farmer, & Gibson, 1991), if X causes Y, then time points that are close in M_y should also be close in M_x. Since M_x is constructed from lags of the observations of X, the points that are close in M_x will also have similar values in the corresponding time series. Therefore, if X causes Y, then M_y can tell us which observations of X should best predict a given held-out point from X. Furthermore, predictability should increase with the number of manifold points in M_y that are considered.

Assessing predictive skill

To test whether X causes Y, M_y is used to infer the points in X that will best predict a given held-out point from X. We measure this performance using predictive skill, quantified by ρ_{ccm} as follows. To begin, we withhold a point from X that we will then attempt to predict. We use M_y to infer the points in M_x that will be closest to this point of interest. This is accomplished using relative pairwise distances of corresponding points in M_y. We then perform a weighted average of the corresponding observations in X using exponential weights derived from these pairwise distances in M_y. We similarly produce predicted values for each held-out point in X. ρ_{ccm} is then calculated as the Pearson correlation between held-out and predicted points. The cross validated nature of this measure serves to reduce over-fitting with respect to the model's tuning parameters described below. To examine whether the signal converges as expected for a causal relationship, these steps are repeated using increasing numbers of points from M_y and M_x.

CauseMap is fast

CauseMap implements CCM in Julia, a high-performance programming language designed for facile technical computing. Via intelligent JIT (just in time) compilation, Julia offers much of the speed of low-level, low-productivity languages like C, while also providing the ease of use and platform independence of much slower high-level languages like Python, R, or Matlab.

At the core of CauseMap is the calculation of distances between a large number of manifold points in potentially high dimensional spaces. To optimize efficiency, CauseMap precomputes all necessary manifolds and pairwise distances using a state-of-the-art, BLAS-based protocol (for benchmarks, see: https://github.com/JuliaStats/Distance.jl).

To illustrate the speed of CauseMap as a function of time series length, we present below the runtimes for successive catenations of the time series presented in Figure 1. For our time series of length 71, CauseMap finishes in approximately 10 seconds. For a time series of over 400 observations, CauseMap still finishes in less than 20 minutes on a single CPU. Note that for this dataset, predictive skill was nearly perfect at a time series length of 213. This calculation finished in less than two minutes.

Time series length	Runtime (s)
71	10.2
142	40.4
213	116.6
284	317.2

355	534.7
426	1080.5

Table 1. Runtime versus time series length. Results are presented for one to six catenations of the dataset presented in Figure 1. Runtime values are for comprehensive parameter optimizations on a single 2.6 GHz Intel Core i7 processor,

Tuning parameter values aid causal interpretation

Beyond the speed and comparative simplicity resulting from cutting-edge JIT compilation, CauseMap offers a number of conveniences and performance enhancements. For CCM, it is particularly important to optimize two tuning parameters: E and τ_p. E is the number of dimensions of the reconstructed shadow manifold. If E_{max} is the optimal embedding dimension, Whitney's Theorem tells us that the dimensionality of the full causal system must be between $(E_{max}-1)/2$ and E_{max}, inclusive (Eelles & Toledo, 1992). . τ_p denotes the time delay of the causal effect of interest. By examining the optimal values of these two parameters, we may place bounds on the number of variables involved in the full causal system, gain insight into the timeframe of causal effects, and obtain a built-in sensitivity analysis of the final results.

CauseMap optimizes and visualizes tuning parameters

E and τ_p are then optimized by multiple iterations of cyclic coordinate descent . Typically convergence of the cross map signal as a function of the time series length (L) alone is taken as the practical criterion for causality. However, measuring the dependence of this signal on E and τ_p is also useful for evaluating whether the result is suitably specific with respect to the assumed structure of the causal system. CauseMap therefore also includes a

plotting function to visualize the dependence of the predictive skill (ρ_{ccm}) on L, as well as on the joint values of E and τ_p.

Interpretation of output

The systematic increase of predictive skill (ρ_{ccm}) with L constitutes a practical, qualitative criterion for causality (Sugihara et al., 2012). Generally, non-causal ρ_{ccm} curves are flat with respect to L, while ρ_{ccm} signals associated with causal signals show striking convergence given sufficient data. One exception is in the case of strong external forcing. An outside variable can introduce a cross map correlation between two quantities if it exerts a sufficiently strong influence over both. We speculate that such situations can produce ρ_{ccm} values that, compared to true causal relationships, have a noisier or less interpretable dependence on E and τ_p. Furthermore, it is necessary to inspect the dependence of the cross map correlation on the joint distribution of E and τ_p in order to properly understand the meaning of the maximal values of these two variables. Note that for high throughput analyses, convergence with respect to L and sensitivity to E and τ_p could be assessed with, e.g. relative difference- and entropy-based measures, respectively.

CauseMap is easy to use

Beyond the tuning parameters mentioned above, CCM requires one to specify a range of library sizes, as well as the window of time points for which cross mapping should be performed. Valid values for these parameters depend in turn on *E* and τ_p. To reduce complexity for the user, CauseMap calculates intelligent defaults for these parameters, while also offering the option of specifying them directly.

Caveats and considerations

The strengths and weaknesses of CCM make it nicely complementary to the existing tools for causal inference. Unlike most algorithms for this task, the performance of CCM improves for increasingly non-linear systems. This capacity depends upon relatively long time series, however. CCM requires at least 25 data points, measured with relatively high accuracy and with sufficient density to capture system dynamics.

There are also theoretical and practical limitations to the types of relationships that CCM can disentangle. For example, if both X and Y are almost entirely determined by a third variable Z, we would be at risk of inferring a spurious relationship between X and Y (as we would be with any other causal inference method). If the forcing from Z is relatively weak however, CCM is expected to provide a lower false positive rate relative to other methods (Sugihara et al., 2012). Finally, CCM performs best with complete data sampled at regular intervals. This is particularly important for inferring the time lag of the causal effect. This limitation can be partially addressed through filtering or appropriate interpolation of input data.

Results and Discussion

To demonstrate CauseMap's functionality and performance, we examined the predator-prey relationship between *Paramecium aurelia* and *Didinium nasutum* (Heskamp et al., 2013). Observations were collected every 12 hours for 30 days, yielding a total of 60 data points (Veilleux, 1976). Plotted in Figure 1 is the CauseMap visualization of the dependence of predictive skill (ρ_{ccm}) on *L, E,* and τ_p. In Figure 1A, we observe convergence in ρ_{ccm} with respect to *L*, the number of data points used for prediction of

187 held-out observations. This convergence is a practical criterion for causality and the
188 source of the name *convergent* cross mapping.

189 The interpretation of this result is that the causal relationship between *P. aurelia*
190 and *D. nasutum* is bi-directional. That is, the number of predators influences the number
191 of prey, and vice-versa. Furthermore, relative strengths of convergence indicate that the
192 top-down influence of the predator (*D. nasutum*) is stronger than the bottom-up influence
193 of the prey (*P. Aurelia)*. As pointed out by Sugihara *et al.*, this finding is consistent with
194 experimental results and illustrates the ability of CCM to investigate asymmetrical bi-
195 directional coupling in non-linear systems.

196 Figures 1B and 1C show the dependence of the max ρ_{ccm} on E (the dimensionality
197 of the reconstructed system), and the supposed time lag of the causal effect (τ_p). Overall,
198 the patterning of these heatmaps demonstrates that max ρ_{ccm} has a reasonable and
199 moderately specific dependence on the dimensionality of the reconstructed system (*E*)
200 and on the time lag of the causal effect (τ_p). We expect this built-in sensitivity analysis to
201 rule out some cases of spurious convergent signal caused by external forcing. In addition,
202 this analysis can alert the researcher when alternative combinations of *E* and τ_p explain
203 the data approximately as well as the optimal values of *E* and τ_p.

204 For the system presented in Figure 1, while the max ρ_{ccm} is relatively insensitive
205 to the assumed dimensionality, the best-performing τ_p values correspond to either
206 immediate causal effects, or those delayed by five days. Note that τ_p=5 corresponds to the
207 principal frequency of the *Paramecium aurelia* and *Didinium nasutum* time series, as
208 determined by fourier transform analysis (see supplemental materials for further details).

209 This suggests that the peak at τ_p=5 is artifactual. Therefore, we are able to infer from the
210 data that, as we would expect, predator and prey populations exert bidirectional effects in
211 real-time.

212

213 ## Performance

214 Approximately 100 CCM evaluations were conducted to produce Figure 1. These
215 calculations finished in approximately 10 seconds on a single 2.6 GHz processor. Each of
216 these evaluations involved the prediction of over 60,000 points, compiled across all
217 sliding windows of libraries of varying lengths. At an average of 1.7 microseconds per
218 prediction, this is a highly efficient implementation given the computational challenges.

219 ## Dependence of predictive skill on time series length

220 CauseMap is designed to examine causal relationships in time series with 25 or more
221 observations. In order to illustrate the effects of shorter time series, we thinned the
222 *Paramedium-Didinium* data set by one-half and by one-third, yielding series of 30 and 20
223 observations, respectively. Figure 2 demonstrates the effect of this reduction on the
224 convergence of predictive skill (ρ_{ccm}). We see that the 1/2 thinned data set recapitulates
225 the trends observed in the full series, including the relative magnitudes of ρ_{ccm} between
226 the mappings of *Didinium* to *Paramecium* and vice versa. The 1/3 thinned sample set, on
227 the other hand, no longer demonstrates convergence. In addition, compared to the longer
228 sets, it exhibits the opposite trend in relative predictive skill between the two mappings.
229 Patterns in max ρ_{ccm} versus E and τ_p are approximately conserved, however (fig. S1).

This example illustrates that CCM performance drops off sharply between 20 and 30 data points. This behavior is partially due to the fact that the predictive skill for a given library size is averaged across sliding windows of that size. As time series get shorter, there are fewer windows of appropriate size across which to average, so the estimate for predictive skill becomes much less reliable.

Potential biomedical applications

Despite its requirement for relatively long time series (>25 observations), CauseMap has the advantage of requiring only a single time series for each variable. In dynamical systems with widely varying or context-specific behavior, this would allow researchers to draw conclusions that are tailored to, e.g. a given patient. Rather than acting on population averages, biomedical researchers would be free to fully personalize therapy to the unique biology and ecology of the patient. One example of this is in the treatment of microbiome dysbiosis. Imbalances in the microbiome have been implicated in, e.g. irritable bowel disease (IBD), obesity, diabetes, asthma, anxiety, and depression . While fecal transplantation therapy is effective in treating specific types of dysbiosis , next generation therapeutics may offer a blend of purified strains, tailored to the gut ecology of the patient. We believe CauseMap has the potential to be a valuable tool for designing such breakthrough therapies.

Additional examples include understanding patient-to-patient variability in drug response using time series metabolomics, and examining the basis of e.g. influenza seasonality using global time series. We expect that such applications will continue to proliferate as the costs of data collection decrease over the coming years. For this reason,

252 we believe it is vitally important that the biomedical research community have access to
253 an efficient implementation of CCM that is user-friendly and available for immediate
254 field testing.

255 Planned future development

256 In future versions, we will include S-map calculations to evaluate the non-linearity of the
257 causal system . We will also add a bootstrap-based procedure for library selection, as
258 opposed to the current approach using sliding windows. This has been shown to reduce
259 the effect of secular trends on the cross map correlation (Hao Ye, George Sugihara,
260 *personal communication*). In addition, we will re-implement the plotting functionality in
261 Julia, removing the requirements of Python and matplotlib for visualization. Finally, we
262 will design Python and R wrappers for CauseMap functions so that our codebase can be
263 easily leveraged from those environments as well. User suggestions will also be
264 considered as we decide how best to develop the tool.

265 Conclusions

266 CauseMap provides a fast, user-friendly implementation of CCM, a powerful new
267 method for exploring dependencies and even establishing causality in complex, highly
268 non-linear datasets with many unobserved variables. We believe that CCM holds a great
269 deal of promise for a wide range of applications, including personalized microbiome
270 therapy and metabolic dynamics analysis. As novel time series datasets continue to
271 emerge, it is our hope that CauseMap will allow researchers to uncover interesting and
272 biomedically actionable causal relationships using this next-generation time series
273 method.

274

Availability and Requirements

Project name: CauseMap

Project home page: http://cyrusmaher.github.io/CauseMap.jl/

Operating system(s): Platform independent

Programming language: Julia

Other requirements: Python and matplotlib (for graphing)

License: MIT

Any restrictions to use by non-academics: No

List of abbreviations

Convergent cross mapping (CCM), State space reconstruction (SSR)

Competing interests

The authors had no competing interest to declare.

Author's contributions

MCM and RDH conceived the project and drafted the manuscript. MCM implemented the algorithm and built the project website.

Author's Information

M. C. M. is a University of California, San Francisco (UCSF) graduate student with an emphasis in statistical computing. After graduation, he will be working as a Software Engineer for Human Longevity Inc., a San Diego-based biotechnology startup. R. D. H.

294 is a Bioengineering & Therapeutics Sciences professor at UCSF. He is also the author of
295 SFS_CODE a popular program for flexible simulation of population genetic evolution.

296 Acknowledgements

297 We would like to thank George Sugihara, Hao Ye, and Ethan Deyle for their invaluable
298 help in understanding the core details of the CCM algorithm, and Lawrence Uricchio,
299 Nicolas Strauli, and Raul Torres for comments on this manuscript.

REFERENCES

Arrieta M-C, Stiemsma LT, Amenyogbe N, Brown EM, Finlay B. 2014. The Intestinal Microbiome in Early Life: Health and Disease. *Frontiers in Immunology* 5:427.

Casdagli M, Eubank S, Farmer JD, Gibson J. 1991. State space reconstruction in the presence of noise. *Physica D: Nonlinear Phenomena* 51:52–98.

Eelles J, Toledo D. 1992. *Collected Papers of Hassler Whitney*. Boston: Birkhauser Publications.

Foster JA, McVey Neufeld K-A. 2013. Gut-brain axis: how the microbiome influences anxiety and depression. *Trends in neurosciences* 36:305–12.

Granger CWJ. 1969. Investigating Causal Relations by Econometric Models and Cross-spectral Methods Title. *Econometrica* 37:424–438.

Heskamp L, Meel-van den Abeelen A, Katsogridakis E, Panerai R, Simpson D, Lagro J, Claassen J. 2013. Convergent cross mapping: a promising technique for future cerebral autoregulation estimation. *CEREBROVASCULAR DISEASES* 35:15–16.

Spearman C. 1904. The proof and measurement of association between two things. *American Journal of Psychology* 15:72–101.

Sugihara G, May R, Ye H, Hsieh C, Deyle E, Fogarty M, Munch S. 2012. Detecting causality in complex ecosystems. *Science (New York, N.Y.)* 338:496–500.

Vanderweele TJ, Arah OA. 2011. Bias formulas for sensitivity analysis of unmeasured confounding for general outcomes, treatments, and confounders. *Epidemiology (Cambridge, Mass.)* 22:42–52.

Veilleux BG. 1976. The analysis of a predatory interaction between Didinium and Paramecium. Master's thesis, University of Alberta.

Figure legends

Fig. 1. An example visualization from CauseMap using abundances of *Paramecium aurelia* and *Didinium nasutum* (see supplemental materials for more information on this system). A.) For optimal parameter values, the convergence of the cross-map correlation with library size. B-C.) The dependence of the maximum cross-map correlation on assumed dimensionality (measured by E) and the time lag of the causal effect (measured by τ_p). Note that the second maximum at τ_p=5 corresponds to the principal frequency of the *P. aurelia* and *D. nasutum* time series, as determined by fourier transform analysis.

334 **Fig. 2. The effect of time series length on ρ_{ccm} convergence.** Black, blue, and red lines
335 illustrate ρ_{ccm} for the full, 1/2 thinned, and 1/3 thinned datasets, respectively. For a given color,
336 darker lines show ρ_{ccm} for the test of whether *Didinium* abundance influences *Paramecium*
337 abundance. Lighter lines examine the converse.

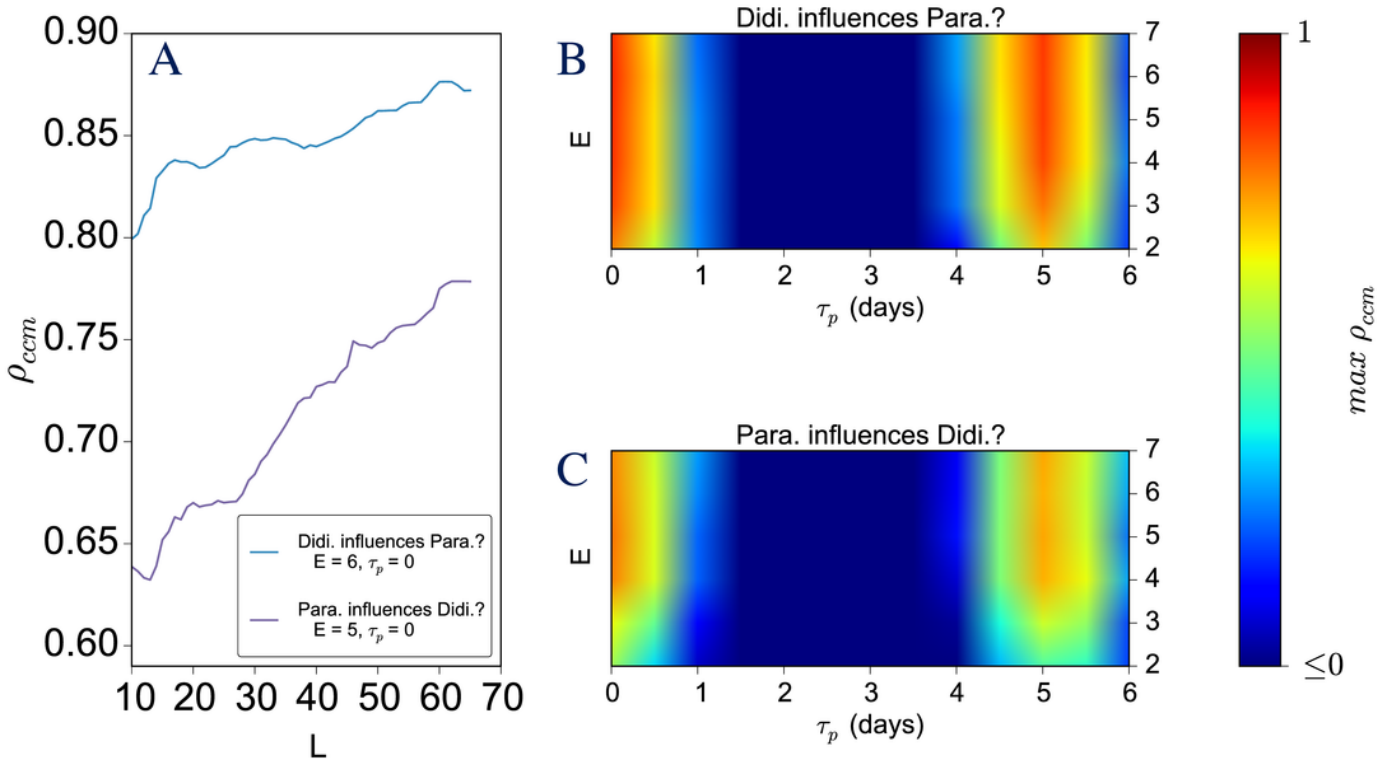
338

339

1

An example visualization from CauseMap using abundances of *Paramecium aurelia* and *Didinium nasutum*

See supplemental materials for more information on this system. A.) For optimal parameter values, the convergence of the cross-map correlation with library size. B-C.) The dependence of the maximum cross-map correlation on assumed dimensionality (measured by E) and the time lag of the causal effect (measured by τ_p). Note that the second maximum at τ_p=5 corresponds to the principal frequency of the *P. aurelia* and *D. nasutum* time series, as determined by fourier transform analysis.



2

The effect of time series length on ρ_{ccm} convergence.

Black, blue, and red lines illustrate $\boldsymbol{\rho_{ccm}}$ for the full, 1/2 thinned, and 1/3 thinned datasets, respectively. For a given color, darker lines show ρ_{ccm} for the test of whether *Didinium* abundance influences *Paramecium* abundance. Lighter lines examine the converse.

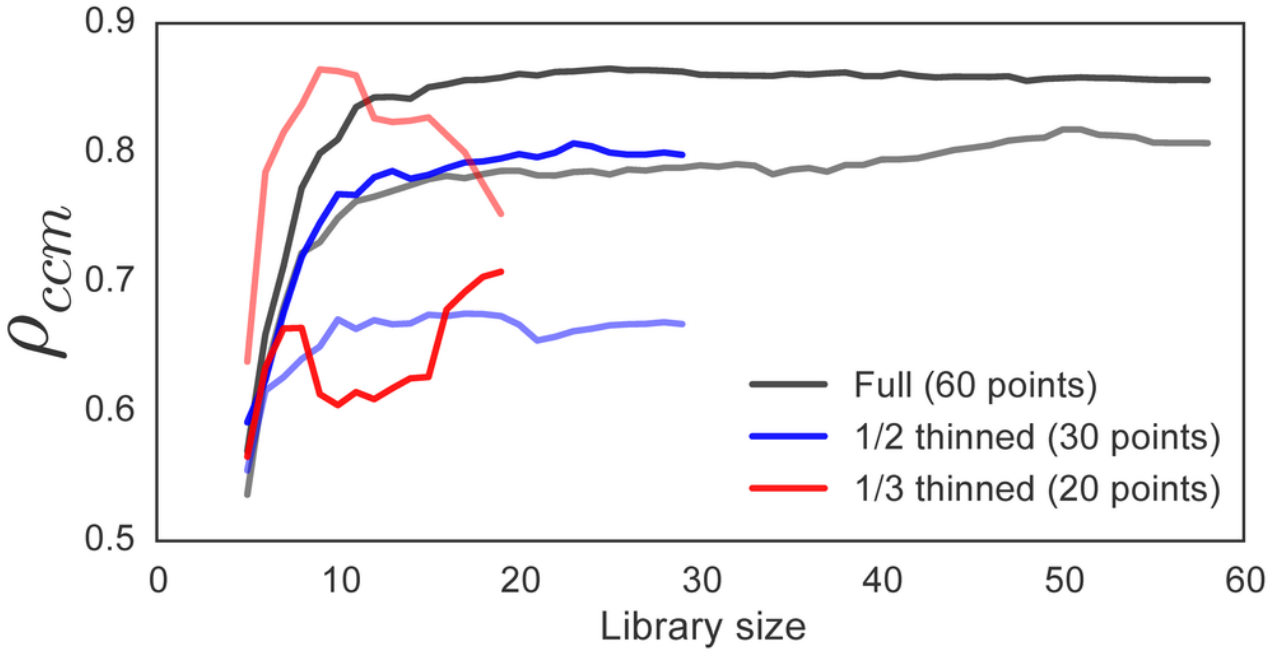


Table 1(on next page)

Table 1. Runtime versus time series length.

Results are presented for one to six catenations of the dataset presented in Figure 1. Runtime values are for comprehensive parameter optimizations on a single 2.6 GHz Intel Core i7 processor

Time series length	Runtime (s)
71	10.2
142	40.4
213	116.6
284	317.2
355	534.7
426	1080.5

Table 1. Runtime versus time series length. Results are presented for one to six catenations of the dataset presented in Figure 1. Runtime values are for comprehensive parameter optimizations on a single 2.6 GHz Intel Core i7 processor